YSNO: A YIELD-SOURCE NEURAL OPERATOR FOR INVERSE MODELING OF SLOPE MECHANICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Frequent landslide disasters underscore the need for monitoring-driven models that can assess slope risk before catastrophic failure. A key capability underlying such models is to infer the spatially varying strength of a slope from observed deformation, which in turn requires a differentiable forward model capable of resolving both localized yielding and its nonlocal mechanical consequences. However, existing neural operators and their benchmarks have been developed predominantly for fluid dynamics and generic PDEs, leaving heterogeneous elastoplastic slope mechanics and inversion largely underexplored. We introduce the Yield-Source Neural Operator (YSNO), an inverse-oriented neural operator for spatial material-field inference in slopes, together with SlopeBench, a paired finite-element benchmark for elastoplastic field prediction and material inversion. YSNO combines a differentiable finite-element physics lift, a plasticity-gated local operator for localized nonlinear corrections, and a finite-rank global integral operator that propagates yield-related source information across the domain. The resulting fully differentiable surrogate can be frozen and directly embedded into gradient-based inversion, establishing an end-to-end computational pathway from displacement observations to spatially distributed strength estimates. On SlopeBench, YSNO reduces full-setting displacement relative-L_2 error by approximately 44.9% relative to the strongest evaluated baseline and yields substantially more accurate material-field reconstruction in inverse tasks. Cross-resolution and out-of-distribution experiments further demonstrate useful generalization beyond the standard training regime. By providing a benchmark for jointly evaluating forward accuracy and inverse suitability in elastoplastic slope mechanics, SlopeBench fills an underexplored regime in scientific operator learning. More broadly, YSNO provides a computational foundation for transforming deformation monitoring into interpretable estimates of internal slope strength, with the potential to support more reliable landslide risk assessment, early warning, and disaster mitigation for vulnerable communities worldwide.

1 INTRODUCTION

Landslides are a global hazard shared by communities in mountainous and hilly regions. They destroy homes and infrastructure, interrupt transport and lifelines, and can turn a short period of rainfall or ground disturbance into a mass-casualty event. A global inventory recorded 4,862 distinct non-seismic fatal landslides and 55,997 deaths from 2004–2016; averaged over those 13 years, this is about 374 recorded fatal events and 4,307 deaths per year (Froude & Petley, 2018). Recent disasters in Zhenxiong, Ethiopia's Gofa zone, and Wayanad illustrate that the same risk is encountered across continents (Wu et al., 2025; Shano et al., 2024; Krishnapriya et al., 2024). Extreme rainfall, mountain development, and increasing exposure continue to raise the consequences. Rescue is time-critical after a failure, while prevention and early warning require observations before the most damaging motion occurs. InSAR, GNSS, LiDAR, and ground sensors now provide long and spatially resolved deformation records, and machine-learning systems use them for detection, forecasting, and warning (Casagli et al., 2023; Tehrani et al., 2022; Zhou et al., 2022; 2024; Zeng et al., 2024; Lv et al., 2025). However, such methods primarily address how slope deformation evolves, whereas the magnitude of deformation alone does not reliably reflect landslide risk across different slopes. A more precise risk assessment therefore requires answering a further question: what internal strength degradation and changes in the mechanical state are associated with the observed deformation?

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.

This problem naturally becomes an inverse problem in which material parameters are inferred from monitored responses. Finite-element analysis establishes the physical mapping between material parameters and displacement, strain, and stress, but high-dimensional inversion requires repeated solves of an expensive nonlinear forward model (Liu et al., 2022; Cai et al., 2024). Two difficulties arise. First, elastoplastic slope responses combine localized yielding with global redistribution. Second, under sparse observations, different material fields may generate nearly indistinguishable responses, making the inverse problem strongly non-unique. An inversion surrogate must therefore be efficient and accurate while preserving, as far as possible, the response sensitivities and gradient structure induced by material variation.

Neural operators provide a natural modeling framework for this problem. DeepONet, the Fourier Neural Operator (FNO), and subsequent methods such as PINO, CNO, GNOT, and Transolver directly learn function-to-function mappings and have shown strong computational efficiency and generalization in fluid dynamics, aerodynamics, wave propagation, and parameterized PDEs (Lu et al., 2021; Li et al., 2021a; Kovachki et al., 2023; Li et al., 2021b; Raonic et al., 2023; Hao et al., 2023; Wu et al., 2024). More recently, DPOT, Poseidon, Universal Physics Transformers, and The Well have advanced large-scale PDE pretraining and cross-physics modeling (Hao et al., 2024; Herde et al., 2024; Alkin et al., 2024; Ohana et al., 2024). Benchmarks including PDEBench, PDEArena, AirfRANS, LagrangeBench, and OpenFWI provide common evaluation environments for neural solvers across parameterized physical systems (Takamoto et al., 2022; Gupta & Brandstetter, 2022; Bonnet et al., 2022; Toshev et al., 2023; Deng et al., 2022). However, these resources leave limited coverage of geotechnical settings in which complex geometry, spatially heterogeneous materials, elastoplastic state transitions, and parameter inversion occur simultaneously.

Existing mechanics-learning studies indicate that embedding known physical structure can substantially reduce the learning burden. EPNN incorporates elastoplastic relations into a constitutive surrogate; finite-element–DeepONet coupling uses neural operators to replace expensive local solves; physics-infused constitutive networks and finite-element-integrated neural networks combine mechanics discretizations with learned models; and NeuberNet uses low-fidelity elastic solutions to predict local elastoplastic responses (Eghbalian et al., 2023; Yin et al., 2022; Roy et al., 2024; Zhang et al., 2025; Grossi et al., 2025). These works suggest that, for problems with clear low-order physical structure, a reliable physics reference should be supplied as an input so that the network can focus on the genuinely difficult nonlinear correction instead of relearning the complete solution. A fuller comparison with displacement-based slope inversion, neural operators for irregular geometries, mechanics-informed surrogates, and scientific-ML benchmarks is provided in the Related Work section at the beginning of the appendix (Liu et al., 2022; Cai et al., 2024; Hao et al., 2023; Wu et al., 2024; Eghbalian et al., 2023; Grossi et al., 2025; Takamoto et al., 2022; Ohana et al., 2024).

To advance real-time landslide early warning while retaining interpretability and generalization, we study inversion of spatially varying slope-material parameter fields. Specifically, for slope computations governed by elastoplastic constitutive laws, we ask: how can one construct a high-fidelity computational operator that remains suitable for gradient-based optimization in the inverse problem? We propose a forward-solution and differentiable-inversion framework based on the Yield-Source Neural Operator (YSNO). YSNO is a slope-computation neural operator that integrates elastoplastic yield features to efficiently predict displacement, strain, and stress fields and to support differentiable Bayesian inference. Its design (a) extracts effective yield features to inject physical information, (b) combines local and nonlocal operators to infer global response fields, and (c) uses multi-response compatibility training to improve generalization. In parallel, we construct SlopeBench for realistic slope scenarios, an evaluation dataset for elastoplastic slope computation and inversion that supports response-prediction assessment in landslide settings. Our main contributions are as follows.

(1) Yield-Source Neural Operator. We propose YSNO for elastoplastic field prediction. By injecting yield-function features, combining local and nonlocal operators, and enforcing compatibility during training, YSNO performs inference over displacement, strain, and stress response fields.

(2) SlopeBench. We construct SlopeBench for neural-operator evaluation and parameter inversion. The dataset covers varying geometries, spatially heterogeneous material fields, and two-stage multi-field responses, providing a unified benchmark for forward prediction and inverse problems in geotechnical mechanics.

(3) Differentiable multi-field prediction and Bayesian inversion. We establish a differentiable framework that jointly predicts displacement, strain, and stress with field-compatibility constraints, and uses a frozen YSNO surrogate for high-dimensional material-parameter identification through Bayesian inversion.

(4) Empirical evaluation. On SlopeBench, YSNO achieves relative improvements of 44.9% in forward prediction and 15.7% in full-field inversion, while also exhibiting computational efficiency, scalability, and out-of-distribution generalization.

2 METHOD

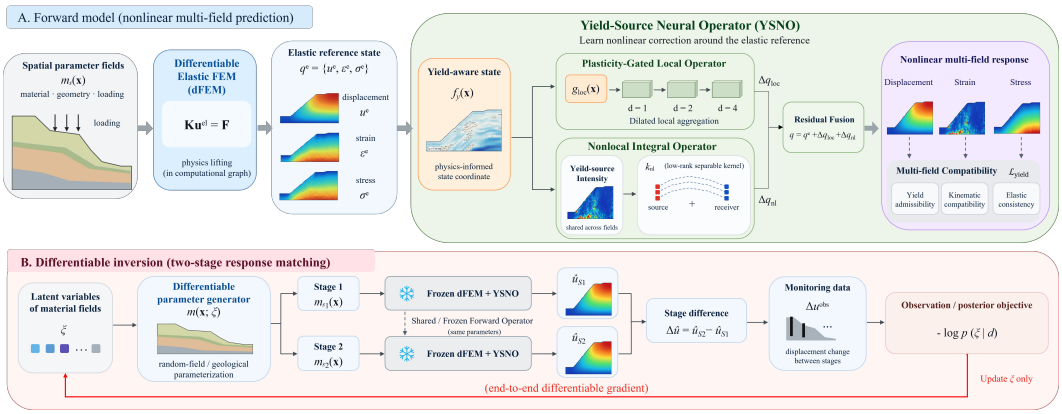

Figure 1: YSNO workflow for staged field prediction and latent-field inversion, with a frozen surrogate differentiated against borehole or full-field displacements.

2.1 PROBLEM AND DIFFERENTIABLE ELASTIC LIFTING

Let $\Omega \subset \mathbb{R}^2$ be a bounded slope domain and let $M : \Omega \to [0, 1]$ be the fixed geometry mask on the surrounding computational grid. We use $s \in \{1, 2\}$ for the recorded analysis stages. The stage input $\mathbf{a}_s$ contains material fields, geometry, prescribed loading, and the stage protocol; the target field is

$$\mathbf{q}_s = (\mathbf{u}_s, \boldsymbol{\varepsilon}_s, \boldsymbol{\sigma}_s), \qquad \mathcal{S} : \mathbf{a}_s \mapsto \mathbf{q}_s, \qquad \Delta\mathbf{q} = \mathbf{q}_2 - \mathbf{q}_1. \tag{1}$$

Here $d_u = 2$ and $d_\varepsilon = d_\sigma = 3$ denote the numbers of in-plane displacement, engineering-strain, and stress channels. The map is conditional on the prescribed loading path and history; it is not a memoryless map of the current strength field alone. We reserve $\Delta\mathbf{q}$ for the stage difference and use $\delta\mathbf{q}_{\mathrm{loc}}$ and $\delta\mathbf{q}_{\mathrm{nl}}$ below for learned correction fields.

The differentiable elastic lift supplies a reference field $\mathbf{q}_s^e = \mathcal{E}(\mathbf{a}_s)$. After essential boundary conditions are imposed, the reduced finite-element system and its sensitivity with respect to a scalar input parameter a_j are

$$K(\mathbf{a}_s)\mathbf{u}_s^e = \mathbf{f}(\mathbf{a}_s), \qquad K\frac{\partial \mathbf{u}_s^e}{\partial a_j} = \frac{\partial \mathbf{f}}{\partial a_j} - \frac{\partial K}{\partial a_j}\mathbf{u}_s^e. \tag{2}$$

The identity assumes a nonsingular reduced stiffness matrix and differentiable assembly.

The complete predictor retains the material channels together with the elastic lift:

$$\widehat{\mathcal{S}}_\theta = \mathcal{G}_\theta \circ \widetilde{\mathcal{E}}, \qquad \widetilde{\mathcal{E}}(\mathbf{a}) = (\mathbf{a}, \mathcal{E}(\mathbf{a})), \qquad \widehat{\mathbf{q}} = \mathbf{q}^e + \delta\mathbf{q}_{\mathrm{loc}} + \delta\mathbf{q}_{\mathrm{nl}}. \tag{3}$$

The augmented input is needed because the correction depends explicitly on strength parameters that are absent from linear elasticity.

2.2 YIELD-AWARE LOCAL CORRECTION AND SHARED SOURCE

We construct a dimensionless trial-state feature from the elastic stress and the strength fields:

$$r(\mathbf{x}) = \frac{\mathcal{Y}(\boldsymbol{\sigma}^e(\mathbf{x}), c(\mathbf{x}), \phi(\mathbf{x}))}{\max\{c(\mathbf{x}), c_{\min}\}}, \qquad c_{\min} > 0. \tag{4}$$

The numerator $\mathcal{Y}$ is the in-plane Mohr–Coulomb trial function defined in Appendix A.9; its stress units match those of cohesion c, so r is dimensionless. A positive value indicates that the trial state is outside the selected yield envelope. It is a routing feature, not a constitutive return-mapping variable or a label of accumulated plasticity.

Let $v = v_\theta(\mathbf{a}, \mathbf{q}^e, r) : \Omega \to \mathbb{R}^{d_v}$ denote the shared encoded feature field, and use v as shorthand for its field argument in all heads below. With $0 \le g_0 < 1$ and $T_g > 0$, define

$$\begin{aligned} g_{\mathrm{loc}}(\mathbf{x}) &= M(\mathbf{x})\{g_0 + (1 - g_0)\,\mathrm{sigmoid}(T_g r(\mathbf{x}))\}, \\ \delta\mathbf{q}_{\mathrm{loc}}(\mathbf{x}) &= g_{\mathrm{loc}}(\mathbf{x})\mathcal{R}_\theta(v)(\mathbf{x}). \end{aligned} \tag{5}$$

The local residual network uses dilation factors $1, 2, 4, 8$. The gate scales a vector correction and is not itself the prediction. The floor g_0 allows a correction away from the trial yield boundary.

The elastic trial state does not determine the final redistribution. We therefore define the nonnegative yield-source intensity λ_θ as the relative source strength at each location that drives the propagation of the global nonlinear response:

$$r_{\mathrm{eff}} = r + \delta r_\theta(v), \tag{6}$$

$$\begin{aligned} g_s(\mathbf{x}) &= M(\mathbf{x})\{g_{s,0} + (1 - g_{s,0})\,\mathrm{sigmoid}(T_s r_{\mathrm{eff}}(\mathbf{x}))\}, \\ \lambda_\theta(\mathbf{x}) &= g_s(\mathbf{x})\,\mathrm{softplus}(h_\theta(v)(\mathbf{x})) \ge 0, \end{aligned} \tag{7}$$

where $0 \le g_{s,0} < 1$, $T_s > 0$, and $\mathrm{softplus}(t) = \log(1 + e^t)$. The displacement, strain, and stress branches share λ_θ.

2.3 Source-conditioned nonlocal integral

Fix an integer $R \ge 1$, the number of source–receiver modes, and let $k \in \{1, \dots, R\}$. For each output field $b \in \{u, \varepsilon, \sigma\}$, its channel dimension is $d_b \in \{2, 3, 3\}$. The shared feature $v(\mathbf{x}) \in \mathbb{R}^{d_v}$ is mapped to: a scalar receiver basis $\rho_{b,k}(\mathbf{x}; v) \in \mathbb{R}$, a nonnegative scalar source basis $\varphi_{b,k}(\mathbf{y}; v) \ge 0$, and a vector value map $V_{b,k}(\mathbf{y}; v) \in \mathbb{R}^{d_v}$. The learned output projection is $W_b \in \mathbb{R}^{d_b \times d_v}$. These maps are heads of the shared encoder; they are not additional physical variables. With the source λ_θ from Eq. (7), define the scalar source mass and vector source moment

$$\begin{aligned} A_{b,k}(v) &= \int_\Omega \lambda_\theta(\mathbf{y})\varphi_{b,k}(\mathbf{y}; v)\,\mathrm{d}\mathbf{y}, \\ B_{b,k}(v) &= \int_\Omega \lambda_\theta(\mathbf{y})\varphi_{b,k}(\mathbf{y}; v)V_{b,k}(\mathbf{y}; v)\,\mathrm{d}\mathbf{y}. \end{aligned} \tag{8}$$

The fixed constant $\epsilon_c > 0$ has the units of $A_{b,k}$ and protects the normalization when a source mode has negligible mass. The nonlocal correction is

$$\delta\mathbf{q}_{\mathrm{nl},b}(\mathbf{x}) = \frac{M(\mathbf{x})}{\sqrt{R}} W_b \sum_{k=1}^{R} \rho_{b,k}(\mathbf{x}; v) \frac{B_{b,k}(v)}{\max\{A_{b,k}(v), \epsilon_c\}}. \tag{9}$$

Thus $\lambda_\theta\varphi_{b,k}$ weights each domain integral, and $\rho_{b,k}$ redistributes the resulting moment to a receiver location. In a grid implementation, nonnegative quadrature weights ω_j approximate these integrals; the corresponding aggregation and redistribution cost is $O(NRd_v)$ for N grid points, or $O(NR)$ when the feature width d_v is treated as fixed. No explicit $N \times N$ kernel is formed.

For fixed v, λ_θ, bases, and value maps, the branch is a sum of R spatially separable moment functionals. For scalar values, its frozen transport matrix has rank at most R. For vector values the output range is bounded by $R\ \mathrm{rank}(W_b) \le R \min\{d_b, d_v\}$; it is therefore inaccurate to call the full vector-valued branch rank-R without this qualification. The complete map remains nonlinear because the source and all heads depend on v.

Proposition 1 (Continuity and quadrature consistency)**.** *On a fixed bounded domain with fixed mask, suppose that λ_θ, $\rho_{b,k}$, $\varphi_{b,k}$, and $V_{b,k}$ are continuous in space and locally Lipschitz in the uniform norm on a compact encoded-input set, and let $\epsilon_c > 0$ be fixed. Then the branch in Eq. (9) is locally Lipschitz in that set. If the encoded fields and the quadrature rules for all displayed integrands converge uniformly, the discrete branch converges uniformly to the integral branch.*

Appendix A proves the source-normalized moment inequality and the fixed-reference discretization estimate in Theorem A.8. Lemma A.2 and Proposition A.3 provide the finite-sum form used by the implementation. For the complete resolution-indexed YSNO family, Theorem A.11 bounds the cross-resolution error by the grid-transfer error, the receptive-field mismatch, and the implementation discrepancy. Under the physical-receptive-field and resolution-stability assumptions of Corollary A.12, these terms vanish and the complete YSNO converges uniformly to the continuous reference operator.

2.4 Multi-field supervision and compatibility

Let $\mathcal{D}$ denote the surrounding grid and set $\mu_M = \int_{\mathcal{D}} M(\mathbf{x})\,\mathrm{d}\mathbf{x} > 0$. For each field b, let D_b be a fixed positive diagonal matrix that converts its channels to dimensionless training units, and define

$$\ell_b(y, \bar{y}) = \frac{1}{\mu_M} \int_{\mathcal{D}} M(\mathbf{x}) \|D_b(y(\mathbf{x}) - \bar{y}(\mathbf{x}))\|_2^2 \,\mathrm{d}\mathbf{x}. \tag{10}$$

The stage-shared predictor is trained on the stage difference and on both stage fields:

$$\mathcal{L}_{\mathrm{data}} = \sum_b \ell_b(\Delta\widehat{\mathbf{q}}_b, \Delta\mathbf{q}_b) + \frac{\alpha}{2} \sum_{s=1}^{2} \sum_b \ell_b(\widehat{\mathbf{q}}_{s,b}, \mathbf{q}_{s,b}), \qquad \alpha \geq 0. \tag{11}$$

With a fixed elastic lift and the same D_u, the displacement-correction difference has exactly the same squared error as the total displacement difference.

Field heads can disagree even when each supervised error is small. We use a yield penalty and a kinematic penalty, where $s_\varepsilon > 0$ is a fixed strain scale:

$$\begin{aligned} \mathcal{L}_y &= \frac{1}{\mu_M} \int_{\mathcal{D}} M(\mathbf{x})\,\mathrm{ReLU}(\widehat{r}(\mathbf{x}))^2 \,\mathrm{d}\mathbf{x}, \qquad \widehat{r} = \frac{\mathcal{Y}(\widehat{\boldsymbol{\sigma}}, c, \phi)}{\max\{c, c_{\min}\}}, \\ \mathcal{L}_{u\varepsilon} &= \frac{1}{\mu_M} \int_{\mathcal{D}} M(\mathbf{x}) \left\| \frac{\widehat{\boldsymbol{\varepsilon}}(\mathbf{x}) - D\widehat{\mathbf{u}}(\mathbf{x})}{s_\varepsilon} \right\|_2^2 \mathrm{d}\mathbf{x}, \\ D\widehat{\mathbf{u}} &= (\partial_x \widehat{u}_x, \partial_y \widehat{u}_y, \partial_y \widehat{u}_x + \partial_x \widehat{u}_y)^\top. \end{aligned} \tag{12}$$

The last component is engineering shear strain. These penalties regularize predictions but do not guarantee equilibrium or a constitutively admissible state. In particular, $\widehat{r} < 0$ does not imply $\widehat{\boldsymbol{\sigma}} = C\widehat{\boldsymbol{\varepsilon}}$: an unloaded point can retain plastic strain. Elastic consistency requires a known zero-plastic-strain region or the relation $\boldsymbol{\sigma} = C(\boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^p)$ with a consistent plastic strain. The coordinate scaling and small-strain limits of the implemented kinematic penalty are stated in Appendix A.9.

2.5 Differentiable material inversion

We denote the frozen-surrogate, gradient-based reconstruction protocol by DI (differentiable inversion). Let $\mathcal{Z} \subset \mathbb{R}^m$ be the feasible latent space and let $\mathcal{T} : \mathcal{Z} \to \mathcal{M} \times \mathcal{M}$ be a differentiable constrained decoder:

$$(\mathbf{m}_1(\mathbf{z}), \mathbf{m}_2(\mathbf{z})) = \mathcal{T}(\mathbf{z}), \qquad \mathbf{z} \in \mathcal{Z}, \qquad \mathbf{m}_s = (c_s, \phi_s), \qquad c_2 \leq c_1,\ \phi_2 \leq \phi_1. \tag{13}$$

Here, $\mathbf{m}_s$ denotes the strength fields at the two analysis stages, while geometry, loading, and the elastic companion fields are fixed. Let $\mathcal{G}_{\theta,u}$ be the frozen displacement head of the forward surrogate. The predicted observation is

$$\widehat{\mathbf{d}}(\mathbf{z}) = \mathcal{H}[\mathcal{G}_{\theta,u}(\mathbf{a}_2(\mathbf{z}), \mathbf{q}_2^e) - \mathcal{G}_{\theta,u}(\mathbf{a}_1(\mathbf{z}), \mathbf{q}_1^e)], \tag{14}$$

where $\mathcal{H}$ extracts either the full displacement field or the available borehole observations. The measured response is modeled as

$$\mathbf{d} = \mathcal{H}[\mathbf{u}_2^\star - \mathbf{u}_1^\star] + \boldsymbol{\eta}, \qquad \boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \Gamma), \qquad \Gamma \succ 0. \tag{15}$$

During inversion, θ, $\mathcal{H}$, and the elastic companion fields are frozen, so gradients propagate through the decoder $\mathcal{T}$ and the neural surrogate. Given a prior density $p(\mathbf{z})$, the reported estimate is obtained by minimizing the negative log-posterior:

$$\mathbf{z}_{\mathrm{MAP}} = \arg\min_{\mathbf{z} \in \mathcal{Z}} \left\{ \frac{1}{2} \big(\widehat{\mathbf{d}}(\mathbf{z}) - \mathbf{d}\big)^\top \Gamma^{-1} \big(\widehat{\mathbf{d}}(\mathbf{z}) - \mathbf{d}\big) - \log p(\mathbf{z}) \right\}. \tag{16}$$

This differentiable formulation enables efficient high-dimensional material identification with either full-field or sparse displacement observations.

3 SLOPEBENCH

The dataset uses a two-dimensional slope profile as the basic independent unit. The main benchmark contains 197 2D slope profiles, including 110 engineered slopes and 87 natural slopes. We generate up to 50 material variants for each profile; after removing variants without complete two-stage outputs, the dataset contains 9,828 paired samples. For each variant, material parameters are generated using spatial random fields with depth-dependent correlation lengths, and one weak band is superimposed to the model potential weak interlayer under realistic geological conditions. Each sample contains two material stages with different material-property parameters. Appendix B specifies the generation and acceptance process.

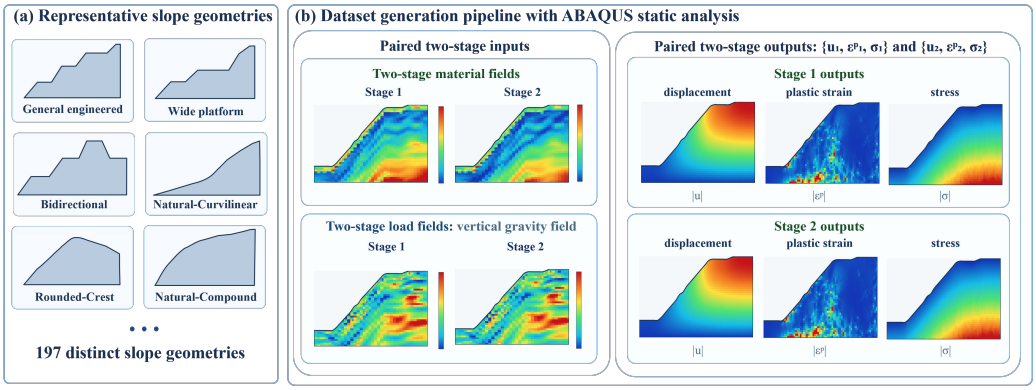

Figure 2: SlopeBench generation pipeline: 197 slope geometries, paired two-stage inputs, and Abaqus-generated multi-field responses.

Task and sample format. We construct *SlopeBench* to evaluate elastoplastic field prediction and material-parameter inversion for spatially heterogeneous slopes. Each sample consists of a two-stage material variant defined on the same slope profile. For each stage $s \in \{1, 2\}$, the input fields include the material parameters (E, ν, c_s, ϕ_s), the unit-weight field γ, the physical coordinates, and a geometric mask. The supervised targets are the displacement field $\mathbf{u}_s$ (2 channels), the in-plane strain field $\boldsymbol{\varepsilon}_s$ (3 channels), and the stress field $\boldsymbol{\sigma}_s$ (3 channels). In addition to stage-wise prediction, the paired samples support evaluation of stage differences.

Finite-element responses. We perform two-dimensional plane-strain finite-element analyses with a Mohr–Coulomb constitutive model and record the displacement, strain, and stress fields at both stages. The corresponding elastic companion solves preserve the same geometry, mesh, boundary conditions, and loading history, providing reference fields for YSNO. The stage labels are defined as follows: the Stage-1 response is the final state after applying the Stage-1 body load, whereas the Stage-2 response is the final state after changing the strength parameters and applying the Stage-2 body load. Consequently, the stage differences in the dataset jointly reflect the effects of strength changes, load removal, and the preceding loading path. The finite-element outputs are interpolated onto a regular physical grid with a spacing of $1\,\mathrm{m}$, and a binary mask identifies the interior of the slope body.

Additional physical regimes. To examine distribution shifts, we additionally provide two-stage variants based on Mohr–Coulomb without a weak layer (7,794 cases), Drucker–Prager with a weak layer (7,799 cases), and Drucker–Prager without a weak layer (7,800 cases). Each set is generated and filtered from 156 parent profiles, with 50 material variants per profile, and is used exclusively for its corresponding supplementary evaluation. To assess the mesh-size dependence of the proposed operator, we further generate three variants of the original SlopeBench dataset with mesh spacings of $h/2$, $2h$, and $4h$, where h denotes the baseline mesh spacing.

4 EXPERIMENTS

4.1 FORWARD FIELD PREDICTION

Protocol. We evaluate displacement, strain, and stress prediction on the valid computational domain. All material variants associated with the same slope profile are assigned to a single split to prevent profile-level leakage. We report results under Full, Small, and geometry-OOD settings, which progressively reduce training coverage or evaluate transfer to unseen slope geometries. Metrics are first averaged over material variants and then over slope cases. Relative L_2 error measures field fidelity, while IoU@25% measures the overlap of regions with the highest plastic-strain response.

YSNO transfers across data regimes. Across the Full, Small, and OOD settings, YSNO maintains the strongest displacement and stress fidelity together with the most reliable localization of high-response regions. Although YSNO does not achieve the best performance for strain prediction, displacement is the quantity directly observed in engineering monitoring and is therefore the primary target for monitoring and inversion. The stress field is mainly used to assess the locations of potential internal weak zones, making the strain field relatively less central in this application. Moreover, YSNO maintains a substantial advantage in the more important task of localizing plastic-strain regions. The same performance pattern is observed under full-data training, limited-data training, and natural-slope transfer, suggesting that the yield-aware reference–correction design provides both strong generalization and favorable scalability. Additional qualitative examples for this evaluation are provided in Appendix C.2. Further OOD prediction examples are reported in Appendix C.3.

Table 1: Forward prediction on Full, Small, and OOD. Entries are mean ± standard deviation; Δu, $\Delta\varepsilon$, and $\Delta\sigma$ are relative L_2 errors. IoU@25% measures localization overlap for the top-25% plastic-strain response region. Full/Small/OOD use 120/10/80 training slopes; OOD trains on artificial and tests on natural slopes.

Setting	Model	Δu	$\Delta\varepsilon$	$\Delta\sigma$	IoU@25%
Full	FNO (Li et al., 2021a)	3.194 ± 0.892	4.769 ± 1.295	4.815 ± 1.210	0.187 ± 0.016
Full	GINO (Li et al., 2023)	0.666 ± 0.035	$\mathbf{0.639 \pm 0.016}$	0.498 ± 0.032	0.377 ± 0.021
Full	LNO (Cao et al., 2024)	3.107 ± 0.979	3.610 ± 0.844	3.128 ± 0.541	0.230 ± 0.077
Full	TRANSOLVER (Wu et al., 2024)	0.819 ± 0.067	0.987 ± 0.200	0.652 ± 0.086	0.210 ± 0.016
Full	U-NET (Ronneberger et al., 2015)	0.748 ± 0.011	0.737 ± 0.033	0.503 ± 0.020	0.203 ± 0.018
Full	MAMBANO (Zheng et al., 2024)	0.639 ± 0.004	0.801 ± 0.057	0.470 ± 0.016	0.373 ± 0.013
Full	**YSNO (ours)**	$\mathbf{0.352 \pm 0.029}$	0.668 ± 0.088	$\mathbf{0.181 \pm 0.020}$	$\mathbf{0.746 \pm 0.027}$
Small	FNO (Li et al., 2021a)	2.030 ± 0.841	3.217 ± 1.034	3.608 ± 1.736	0.228 ± 0.065
Small	GINO (Li et al., 2023)	0.810 ± 0.259	$\mathbf{0.691 \pm 0.231}$	0.676 ± 0.329	0.409 ± 0.075
Small	LNO (Cao et al., 2024)	1.898 ± 0.933	2.677 ± 0.995	2.543 ± 1.108	0.159 ± 0.023
Small	TRANSOLVER (Wu et al., 2024)	1.151 ± 0.478	1.871 ± 1.329	1.093 ± 0.764	0.417 ± 0.115
Small	U-NET (Ronneberger et al., 2015)	0.753 ± 0.185	0.725 ± 0.149	0.531 ± 0.091	0.244 ± 0.078
Small	MAMBANO (Zheng et al., 2024)	0.684 ± 0.069	0.792 ± 0.245	0.501 ± 0.077	0.376 ± 0.041
Small	**YSNO (ours)**	$\mathbf{0.396 \pm 0.192}$	0.770 ± 0.585	$\mathbf{0.209 \pm 0.107}$	$\mathbf{0.700 \pm 0.126}$
OOD	FNO (Li et al., 2021a)	3.916 ± 1.393	5.029 ± 1.612	5.032 ± 1.350	0.206 ± 0.042
OOD	GINO (Li et al., 2023)	0.747 ± 0.060	$\mathbf{0.716 \pm 0.069}$	0.559 ± 0.069	0.191 ± 0.006
OOD	LNO (Cao et al., 2024)	2.164 ± 0.798	2.424 ± 0.831	2.050 ± 0.936	0.210 ± 0.017
OOD	TRANSOLVER (Wu et al., 2024)	0.859 ± 0.112	0.865 ± 0.090	0.602 ± 0.030	0.191 ± 0.019
OOD	U-NET (Ronneberger et al., 2015)	0.816 ± 0.047	0.867 ± 0.154	0.569 ± 0.032	0.212 ± 0.019
OOD	MAMBANO (Zheng et al., 2024)	0.681 ± 0.039	0.824 ± 0.173	0.504 ± 0.025	0.229 ± 0.022
OOD	**YSNO (ours)**	$\mathbf{0.480 \pm 0.128}$	0.923 ± 0.404	$\mathbf{0.233 \pm 0.087}$	$\mathbf{0.395 \pm 0.044}$

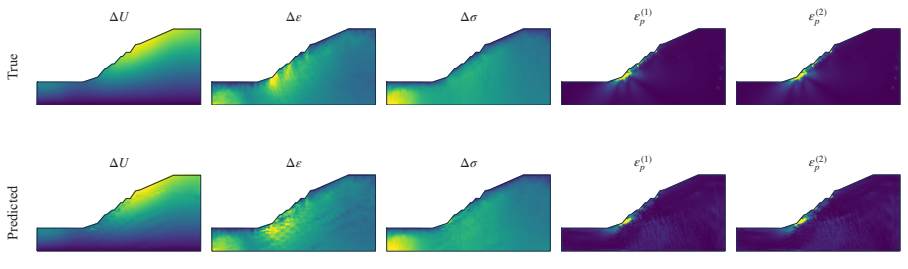

Figure 3: YSNO predictions versus finite-element ground truth for profile 046 (artificial, $v0006$). The rows show true and predicted fields; columns report stage-difference displacement, strain, stress, and plastic-strain fields.

4.2 SUITABILITY FOR MATERIAL INVERSION

Protocol. We evaluate material inversion on four held-out parent cases, with all variants associated with these cases kept unseen by every compared surrogate. All DI entries use the same reconstruction procedure and a fixed budget of evaluations of the frozen surrogate. In the Borehole setting, the observation model provides 1–3 displacement measurements selected according to a predefined rule based on slope size; in the Full setting, all displacement values in the valid domain are provided. Methods that do not use gradients perform inversion through Abaqus iterations and are reported separately as a solver-based comparison, keeping the computational costs of Abaqus solves and surrogate evaluations separately interpretable.

Table 2: Material inversion on four unseen cases. Borehole uses 1–3 size-adapted displacement points; Full field uses all displacement values. Columns report NMAE and time (s); non-gradient and DI rows use 200 Abaqus iterations and 200 surrogate evaluations, respectively.

	Borehole		Full field	
Method	NMAE ↓	Time (s) ↓	NMAE ↓	Time (s) ↓
Non-gradient solvers				
GA (Holland, 1992)	0.317 ± 0.033	50.55	0.43 ± 0.017	59.25
DREAM-ZS (Vrugt et al., 2009)	0.47 ± 0.095	50.46	0.646 ± 0.085	55.96
BUS (Straub & Papaioannou, 2015)	0.536 ± 0.006	49.20	0.489 ± 0.014	55.32
CMA-ES (Hansen & Ostermeier, 2001)	0.571 ± 0.036	65.36	0.503 ± 0.048	71.41
TEKI (Iglesias et al., 2013)	0.665 ± 0.201	51.57	1.074 ± 0.025	64.98
Differentiable inversion (DI) with frozen surrogates				
DI+FNO (Li et al., 2021a)	0.338 ± 0.016	1.91	0.339 ± 0.006	1.64
DI+GINO (Li et al., 2023)	0.424 ± 0.059	1.80	0.507 ± 0.015	1.82
DI+LNO (Cao et al., 2024)	0.355 ± 0.013	5.86	0.343 ± 0.03	5.73
DI+TRANSOLVER (Wu et al., 2024)	0.323 ± 0.009	2.46	0.33 ± 0.011	2.42
DI+U-NET (Ronneberger et al., 2015)	0.371 ± 0.005	**1.24**	0.392 ± 0.011	**1.17**
DI+MAMBANO (Zheng et al., 2024)	0.32 ± 0.03	220.40	0.324 ± 0.025	222.49
DI+**YSNO (ours)**	$\mathbf{0.301 \pm 0.017}$	2.51	$\mathbf{0.273 \pm 0.02}$	2.70

DI+YSNO improves the accuracy–efficiency trade-off. Relative to the strongest non-gradient solver, DI+YSNO reaches lower material-recovery error with a substantially shorter reconstruction time in both borehole and Full-field settings. Among differentiable operator backbones, it also provides the most accurate displacement reconstruction while remaining in the same practical runtime scale as the main alternatives. The result supports using a high-fidelity forward operator inside gradient-based inversion: engineering warning depends on the reconstructed displacement response, so lower forward and inverse error is more valuable than a marginal runtime advantage.

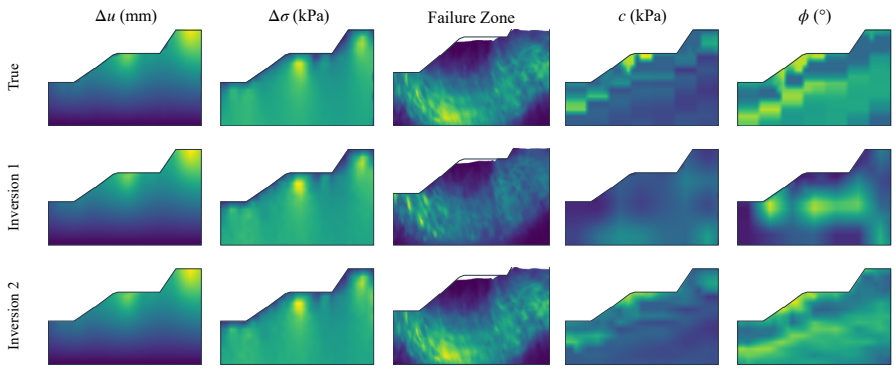

Figure 4: Material-field inversion for an artificial slope. The true fields are compared with two inverse solutions, showing similar displacement and stress responses despite different recovered material fields.

The inverse fields expose non-uniqueness. Figure 4 compares the ground-truth material fields with the effective reconstructions obtained by YSNO in two inversion trials. In both trials, YSNO successfully reconstructs the displacement field, while the predicted stress field and final failure-zone field closely match the ground truth. Notably, in Inversion 1, the reconstructed material field does not

exactly coincide with the ground truth, yet it achieves the same optimization objective. This result reveals the inherent non-uniqueness of material-parameter inversion: different estimates of cohesion and friction angle can produce highly consistent displacement and stress responses. It also further supports the need for a high-fidelity surrogate operator in the inversion loop.

4.3 Ablation and mesh-resolution scalability

Every physics module contributes. The ablation study removes one YSNO ingredient at a time under the same Full evaluation. Removing the elastic FEM reference or the plastic operator causes the clearest degradation, while removing the analytic Mohr–Coulomb input, learned yield margin, or compatibility loss also harms at least one field or localization measure. The consistent pattern across components supports the value of combining physical lifting, yield-aware routing, and multi-field compatibility rather than relying on a single architectural modification. Additional qualitative ablation results are presented in Appendix C.4.

Table 3: YSNO ablation on Full. Entries are relative L_2 errors and IoU@25% overlap (mean $\pm$ standard deviation).

Variant	Δu	$\Delta \varepsilon$	$\Delta \sigma$	IoU@25%
YSNO (ours)	$\mathbf{0.352 \pm 0.029}$	$\mathbf{0.668 \pm 0.088}$	$\mathbf{0.181 \pm 0.020}$	$\mathbf{0.746 \pm 0.027}$
w/o Plastic Operator	0.703 ± 0.014	0.875 ± 0.055	0.185 ± 0.015	0.383 ± 0.008
w/o Elastic FEM Input	0.843 ± 0.034	1.776 ± 0.109	0.850 ± 0.061	0.249 ± 0.001
w/o Analytic MC Input	0.545 ± 0.005	1.082 ± 0.124	0.202 ± 0.014	0.415 ± 0.010
w/o Learned Yield Margin	1.015 ± 0.049	1.053 ± 0.118	0.258 ± 0.020	0.400 ± 0.019
w/o Compatibility Loss	0.446 ± 0.022	1.092 ± 0.066	0.261 ± 0.006	0.400 ± 0.010

YSNO scales across independently solved meshes. On datasets with target mesh spacings of $h/2$, h, $2h$, and $4h$, displacement, strain, and stress errors follow a consistent trend and high-response plastic-region localization remains stable. This direct multi-mesh comparison shows that YSNO transfers across solver resolutions and supports its scalability beyond a single finite-element discretization.

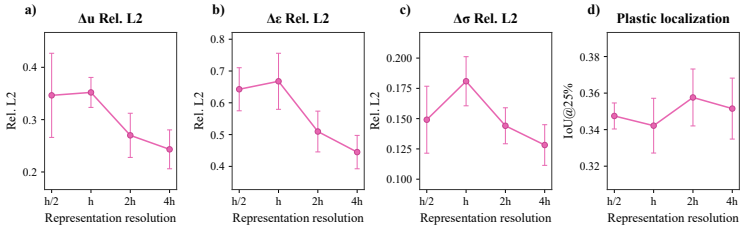

Figure 5: Cross-resolution Abaqus evaluation of YSNO.

5 Conclusion

This paper presents YSNO, a yield-aware neural operator for forward prediction and differentiable material-field inversion in elastoplastic slopes, together with SlopeBench, a benchmark for evaluating both tasks across diverse slope geometries and heterogeneous materials. By combining an elastic reference with local and nonlocal corrections and multi-field compatibility, YSNO outperforms the evaluated baselines in displacement and stress prediction and material inversion. Experiments demonstrate strong generalization across full-data, limited-data, and out-of-distribution settings, together with computational efficiency and scalability across the tested Abaqus mesh resolutions. These results offer a promising direction for landslide risk assessment and early warning. Inversion nevertheless remains non-unique: distinct material fields can produce similar displacement responses. The current benchmark does not include engineered reinforcement structures, coupled fluid–solid processes, or time-dependent slope evolution. Future work will extend the framework to reinforced slopes and transient hydromechanical simulations to better represent the conditions encountered in practical landslide monitoring.

AI USE STATEMENT

This work used AI-assisted tools to translate the Chinese draft provided by the authors into English and to revise the wording and organization of the manuscript. AI tools also assisted with the review of related work, verification of equations and citations, and development of the conditional integral-branch proposition and its proof. For the code, AI tools contributed to the initial implementation and debugging.

REFERENCES

Benedikt Alkin, Andreas Fürst, Simon Schmid, Lukas Gruber, Markus Holzleitner, and Johannes Brandstetter. Universal physics transformers: A framework for efficiently scaling neural operators. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-0793. URL https://doi.org/10.52202/079017-0793.

Florent Bonnet, Jocelyn Ahmed Mazari, Paola Cinnella, and Patrick Gallinari. AirfRANS: High fidelity computational fluid dynamics dataset for approximating Reynolds-averaged Navier–Stokes solutions. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/94ab7b23a345f93333eac8748a66c763-Paper-Datasets_and_Benchmarks.pdf.

Jing-Sen Cai, Tian-Chyi Jim Yeh, E-Chuan Yan, and Rui-Xuan Tang. Exploiting excavation-induced displacement for slope heterogeneity characterization and stability analysis. *Computers and Geotechnics*, 170:106292, 2024. doi: 10.1016/j.compgeo.2024.106292. URL https://doi.org/10.1016/j.compgeo.2024.106292.

Qianying Cao, Somdatta Goswami, and George Em Karniadakis. Laplace neural operator for solving differential equations. *Nature Machine Intelligence*, 6:631–640, 2024. doi: 10.1038/s42256-024-00844-4. URL https://doi.org/10.1038/s42256-024-00844-4.

Nicola Casagli, Emanuele Intrieri, V. Tofani, G. Gigli, and F. Raspini. Landslide detection, monitoring and prediction with remote-sensing techniques. *Nature Reviews Earth & Environment*, 4:51–64, 2023. doi: 10.1038/s43017-022-00373-x. URL https://doi.org/10.1038/s43017-022-00373-x.

Chengyuan Deng, Shihang Feng, Hanchen Wang, Xitong Zhang, Peng Jin, Yinan Feng, Qili Zeng, Yinpeng Chen, and Youzuo Lin. Openfwi: Large-scale multi-structural benchmark datasets for full waveform inversion. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 6007–6020. Curran Associates, Inc., 2022. doi: 10.52202/068431-0435. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/27d3ef263c7cb8d542c4f9815a49b69b-Paper-Datasets_and_Benchmarks.pdf.

Mahdad Eghbalian, Mehdi Pouragha, and Richard Wan. A physics-informed deep neural network for surrogate modeling in classical elasto-plasticity. *Computers and Geotechnics*, 159:105472, 2023. doi: 10.1016/j.compgeo.2023.105472. URL https://doi.org/10.1016/j.compgeo.2023.105472.

Gordon A. Fenton and D. V. Griffiths. *Risk Assessment in Geotechnical Engineering*. John Wiley & Sons, 2008. doi: 10.1002/9780470284704. URL https://doi.org/10.1002/9780470284704.

Melanie J. Froude and David N. Petley. Global fatal landslide occurrence from 2004 to 2016. *Natural Hazards and Earth System Sciences*, 18(8):2161–2181, 2018. doi: 10.5194/nhess-18-2161-2018. URL https://doi.org/10.5194/nhess-18-2161-2018.

Tommaso Grossi, Marco Beghini, and Matteo Benedetti. Neubernet: A neural operator solving elastic-plastic partial differential equations at v-notches from low-fidelity elastic simulations. *Communications Engineering*, 4:218, 2025. doi: 10.1038/s44172-025-00549-5. URL https://doi.org/10.1038/s44172-025-00549-5.

Jayesh K. Gupta and Johannes Brandstetter. Towards multi-spatiotemporal-scale generalized PDE modeling. *arXiv preprint arXiv:2209.15616*, 2022. doi: 10.48550/arXiv.2209.15616. URL https://arxiv.org/abs/2209.15616.

Nikolaus Hansen and Andreas Ostermeier. Completely derandomized self-adaptation in evolution strategies. *Evolutionary Computation*, 9(2):159–195, 2001. doi: 10.1162/106365601750190398.

Zhongkai Hao, Zhengyi Wang, Hang Su, Chengyang Ying, Yinpeng Dong, Songming Liu, Ze Cheng, Jian Song, and Jun Zhu. GNOT: A general neural operator transformer for operator learning. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 12556–12569. PMLR, 2023. URL https://proceedings.mlr.press/v202/hao23c.html.

Zhongkai Hao, Chang Su, Songming Liu, Julius Berner, Chengyang Ying, Hang Su, Anima Anandkumar, Jian Song, and Jun Zhu. DPOT: Auto-regressive denoising operator transformer for large-scale PDE pre-training. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 17616–17635. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/hao24d.html.

Maximilian Herde, Bogdan Raonić, Tobias Rohner, Roger Käppeli, Roberto Molinaro, Emmanuel de Bézenac, and Siddhartha Mishra. Poseidon: Efficient foundation models for pdes. In *Advances in Neural Information Processing Systems*, volume 37, 2024. doi: 10.52202/079017-2311. URL https://doi.org/10.52202/079017-2311.

John H. Holland. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press, 04 1992. ISBN 9780262275552. doi: 10.7551/mitpress/1090.001.0001. URL https://doi.org/10.7551/mitpress/1090.001.0001.

Marco A. Iglesias, Kody J. H. Law, and Andrew M. Stuart. Ensemble Kalman methods for inverse problems. *Inverse Problems*, 29(4):045001, 2013. doi: 10.1088/0266-5611/29/4/045001. URL https://doi.org/10.1088/0266-5611/29/4/045001.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to pdes. *Journal of Machine Learning Research*, 24(89):1–97, 2023. URL http://jmlr.org/papers/v24/21-1524.html.

V. K. Krishnapriya, A. Rajaneesh, K. S. Sajinkumar, Thomas Oommen, Ali P. Yunus, Nikhil Nedumpallile Vasu, R. B. Binoj Kumar, and S. Adarsh. A rapid run-out assessment methodology for the 2024 wayanad debris flow. *npj Natural Hazards*, 1(1), 2024. doi: 10.1038/s44304-024-00044-5. URL https://doi.org/10.1038/s44304-024-00044-5.

Samuel Lanthaler, Siddhartha Mishra, and George E. Karniadakis. Error estimates for DeepONets: A deep learning framework in infinite dimensions. *Transactions of Mathematics and Its Applications*, 6(1):tnac001, January 2022. ISSN 2398-4945. doi: 10.1093/imatrm/tnac001. URL https://doi.org/10.1093/imatrm/tnac001.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021a. URL https://openreview.net/forum?id=c8P9NQVtmnO.

Zongyi Li, Hongkai Zheng, Nikola Kovachki, David Jin, Haoxuan Chen, Burigede Liu, Kamyar Azizzadenesheli, and Anima Anandkumar. Physics-informed neural operator for learning partial differential equations. *arXiv preprint arXiv:2111.03794*, 2021b. URL https://arxiv.org/abs/2111.03794.

Zongyi Li, Nikola Kovachki, Chris Choy, Boyi Li, Jean Kossaifi, Shourya Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, and Animashree Anandkumar. Geometry-informed neural operator for large-scale 3d pdes. In A. Oh, T. Naumann,

A. Globerson, K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information Processing Systems*, volume 36, pp. 35836–35854. Curran Associates, Inc., 2023. doi: 10.52202/075280-1556. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/70518ea42831f02afc3a2828993935ad-Paper-Conference.pdf.

Yibiao Liu, Weizhong Ren, Chenchen Liu, Simin Cai, and Wenhui Xu. Displacement-based back-analysis frameworks for soil parameters of a slope: Using frequentist inference and bayesian inference. *International Journal of Geomechanics*, 22(4):04022026, 2022. doi: 10.1061/(ASCE)GM.1943-5622.0002318. URL https://ascelibrary.org/doi/abs/10.1061/%28ASCE%29GM.1943-5622.0002318.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021. doi: 10.1038/s42256-021-00302-5. URL https://doi.org/10.1038/s42256-021-00302-5.

Jichao Lv, Rui Zhang, Xin Bao, Renzhe Wu, Ruikai Hong, Xu He, and Guoxiang Liu. Time-series insar landslide three-dimensional deformation prediction method considering meteorological time-delay effects. *Engineering Geology*, 350:107986, 2025. doi: 10.1016/j.enggeo.2025.107986. URL https://doi.org/10.1016/j.enggeo.2025.107986.

Ruben Ohana, Michael McCabe, Lucas Meyer, Rudy Morel, Fruzsina J. Agocs, Miguel Beneitez, Marsha Berger, Blakesley Burkhart, Stuart B. Dalziel, Drummond B. Fielding, Daniel Fortunato, Jared A. Goldberg, Keiya Hirashima, Yan-Fei Jiang, Rich R. Kerswell, Suryanarayana Maddu, Jonah Miller, Payel Mukhopadhyay, Stefan S. Nixon, Jeff Shen, Romain Watteaux, Bruno Régaldo-Saint Blancard, François Rozet, Liam Parker, Miles Cranmer, and Shirley Ho. The well: a large-scale collection of diverse physics simulations for machine learning. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 44989–45037. Curran Associates, Inc., 2024. doi: 10.52202/079017-1430. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/4f9a5acd91ac76569f2fe291b1f4772b-Paper-Datasets_and_Benchmarks_Track.pdf.

Thomas O'Leary-Roseberry, Peng Chen, Umberto Villa, and Omar Ghattas. Derivative-informed neural operator: An efficient framework for high-dimensional parametric derivative learning. *Journal of Computational Physics*, 496:112555, 2024. doi: 10.1016/j.jcp.2023.112555. URL https://doi.org/10.1016/j.jcp.2023.112555.

Christopher J. Paciorek and Mark J. Schervish. Nonstationary covariance functions for Gaussian process regression. In *Advances in Neural Information Processing Systems 16*, pp. 273–280. MIT Press, 2003. URL https://proceedings.neurips.cc/paper/2003/hash/326a8c055c0d04f5b06544665d8bb3ea-Abstract.html.

Kok-Kwang Phoon and Fred H. Kulhawy. Characterization of geotechnical variability. *Canadian Geotechnical Journal*, 36(4):612–624, 1999. doi: 10.1139/t99-038. URL https://doi.org/10.1139/t99-038.

Bogdan Raonic, Roberto Molinaro, Tim De Ryck, Tobias Rohner, Francesca Bartolucci, Rima Alaifari, Siddhartha Mishra, and Emmanuel de Bézenac. Convolutional neural operators for robust and accurate learning of PDEs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 77187–77200. Curran Associates, Inc., 2023. doi: 10.52202/075280-3376. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/f3c1951b34f7f55ffaecada7fde6bd5a-Paper-Conference.pdf.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. URL https://arxiv.org/abs/1505.04597.

Arunabha M. Roy, Suman Guha, Veera Sundararaghavan, and Raymundo Arröyave. Physics-infused deep neural network for solution of non-associative drucker–prager elastoplastic constitutive model. *Journal of the Mechanics and Physics of Solids*, 185:105570, 2024. doi: 10.1016/j.jmps.2024.105570. URL https://doi.org/10.1016/j.jmps.2024.105570.

Leulalem Shano, Bisirat Gisila, Wondwosen Jerene, Desta Ekaso, Tigabu Baye, Tariku Degife, Geberemedin Chameno, Zeleke Dosa, and Muralitharan Jothimani. Fatal landslides in kencho, shacha gozdi villages, gofa zone, ethiopia: A detailed investigation (geological, geotechnical, geophysical geospatial) of the july 22, 2024 catastrophe and its socioeconomic repercussions. *Quaternary Science Advances*, 16:100241, 2024. ISSN 2666-0334. doi: https://doi.org/10.1016/j.qsa.2024.100241. URL https://www.sciencedirect.com/science/article/pii/S2666033424000790.

Daniel Straub and Iason Papaioannou. Bayesian updating with structural reliability methods. *Journal of Engineering Mechanics*, 141(3):04014134, 2015. doi: 10.1061/(ASCE)EM.1943-7889.0000839. URL https://ascelibrary.org/doi/abs/10.1061/%28ASCE%29EM.1943-7889.0000839.

Andrew M. Stuart. Inverse problems: A Bayesian perspective. *Acta Numerica*, 19:451–559, 2010. doi: 10.1017/S0962492910000061. URL https://doi.org/10.1017/S0962492910000061.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Dan MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems*, volume 35, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/0a9747136d411fb83f0cf81820d44afb-Abstract-Datasets_and_Benchmarks.html.

Farshad S. Tehrani, Michele Calvello, Z. Liu, L. Zhang, and S. Lacasse. Machine learning and landslide studies: Recent advances and applications. *Natural Hazards*, 114(2):1197–1245, 2022. doi: 10.1007/s11069-022-05423-7. URL https://doi.org/10.1007/s11069-022-05423-7.

Artur P. Toshev, Gianluca Galletti, Fabian Fritz, Stefan Adami, and Nikolaus A. Adams. Lagrangebench: A lagrangian fluid mechanics benchmarking suite. In *Advances in Neural Information Processing Systems*, volume 36, 2023. doi: 10.52202/075280-2830. URL https://doi.org/10.52202/075280-2830.

Jasper A. Vrugt, C. J. F. ter Braak, C. G. H. Diks, Bruce A. Robinson, James M. Hyman, and Dave Higdon. Accelerating markov chain monte carlo simulation by differential evolution with self-adaptive randomized subspace sampling. *International journal of nonlinear sciences and numerical simulation*, 10(3):18, 2009. ISSN 1565-1339. doi: 10.1515/IJNSNS.2009.10.3.273. is peer reviewed.

Christopher K. I. Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, volume 13, pp. 682–688. MIT Press, 2001. URL https://proceedings.neurips.cc/paper/2000/file/19de10adbaa1b2ee13f77f679fa1483a-Paper.pdf.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for PDEs on general geometries. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53681–53705. PMLR, 2024. URL https://proceedings.mlr.press/v235/wu24r.html.

Mingyang Wu, Anhui Li, Zhi Li, Ningsheng Chen, Shufeng Tian, Runing Hou, Jules Maurice Habumugisha, and Na Huang. Frost-heaving may triggered the catastrophic landslide in Zhenxiong on January 22, 2024. *Landslides*, 22(4):1153–1166, 2025. doi: 10.1007/s10346-024-02401-x. URL https://doi.org/10.1007/s10346-024-02401-x.

Tianju Xue, Shuheng Liao, Zhengtao Gan, Chanwook Park, Xiaoyu Xie, Wing Kam Liu, and Jian Cao. JAX-FEM: A differentiable GPU-accelerated 3D finite element solver for automatic inverse design and mechanistic data science. *Computer Physics Communications*, 291:108802, 2023. doi: 10.1016/j.cpc.2023.108802. URL https://doi.org/10.1016/j.cpc.2023.108802.

Minglang Yin, Enrui Zhang, Yue Yu, and George E. Karniadakis. Interfacing finite elements with deep neural operators for fast multiscale modeling of mechanics problems. *Computer Methods in Applied Mechanics and Engineering*, 402:115027, 2022. doi: 10.1016/j.cma.2022.115027. URL https://doi.org/10.1016/j.cma.2022.115027.

Peng Zeng, Bing Feng, Keren Dai, Tianbin Li, Xuanmei Fan, and Xiaoping Sun. Can satellite InSAR innovate the way of large landslide early warning? *Engineering Geology*, 342:107771, 2024. doi: 10.1016/j.enggeo.2024.107771. URL https://doi.org/10.1016/j.enggeo.2024.107771.

Ning Zhang, Kunpeng Xu, Zhen Yu Yin, Kai-Qi Li, and Yin-Fu Jin. Finite element-integrated neural network framework for elastic and elastoplastic solids. *Computer Methods in Applied Mechanics and Engineering*, 433:117474, 2025. doi: 10.1016/j.cma.2024.117474. URL https://doi.org/10.1016/j.cma.2024.117474.

Jianwei Zheng, Wei Li, Ni Xu, Junwei Zhu, Xiaoxu Lin, and Xiaoqin Zhang. Alias-free mamba neural operator. In *Advances in Neural Information Processing Systems*, volume 37, pp. 52962–52995, 2024. doi: 10.52202/079017-1678. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/5ee553ec47c31e46a1209bb858b30aa5-Paper-Conference.pdf.

Chao Zhou, Ying Cao, Lulu Gan, Yue Wang, Mahdi Motagh, Sigrid Roessner, Xie Hu, and Kunlong Yin. A novel framework for landslide displacement prediction using MT-InSAR and machine learning techniques. *Engineering Geology*, 334:107497, 2024. doi: 10.1016/j.enggeo.2024.107497. URL https://doi.org/10.1016/j.enggeo.2024.107497.

Fan Zhou, Rongfan Li, Qiang Gao, Goce Trajcevski, Kunpeng Zhang, and Ting Zhong. Dynamic manifold learning for land deformation forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(4):4725–4733, 2022. doi: 10.1609/aaai.v36i4.20398. URL https://doi.org/10.1609/aaai.v36i4.20398.

Related Work

Displacement-based slope inversion. Liu et al. (2022) study frequentist and Bayesian displacement-based back-analysis with finite-element surrogates, including spatial random fields in the Bayesian setting. Cai et al. (2024) use excavation-induced displacement to characterize slope heterogeneity and assess stability. These studies establish that displacement can inform material identification under an appropriate mechanical model. Our emphasis is a geometry-conditioned, multi-field surrogate and a paired evaluation resource for comparing forward prediction with frozen-surrogate material recovery.

Field learning and neural operators. DeepONet uses branch and trunk representations for nonlinear operator learning (Lu et al., 2021); FNO introduces global mixing through Fourier-space kernels (Li et al., 2021a). The neural-operator framework formalizes compositions of pointwise maps and integral operators (Kovachki et al., 2023), while CNO studies convolutional representations across discretizations (Raonic et al., 2023). These methods motivate field-to-field learning, but their standard formulations do not encode the staged strength transition or the paired elastic reference used here.

Local, geometric, and attention-based operators. Operator architectures have also been designed for locality, irregular geometry, and heterogeneous inputs. GNOT uses heterogeneous normalized attention and geometric gating for irregular meshes and multiscale problems (Hao et al., 2023); Transolver uses physics-aware attention on general geometries (Wu et al., 2024); and multiscale operator formulations target long-range spatial and temporal interactions (**?**). YSNO shares the goal of avoiding a dense all-to-all kernel, but its nonlocal branch is explicitly conditioned on a yield-state source and is analyzed separately from the fixed-grid local backbone.

Physics-guided learning and differentiable mechanics. Physics-informed neural operators incorporate PDE residuals or known constraints into training (Li et al., 2021b), while differentiable finite-element systems expose sensitivities through a discretized mechanics model (Xue et al., 2023). These approaches make the forward map more physically structured, but a residual penalty alone does not identify the strength field that caused an observed deformation. We use a recorded elastic companion and explicit field-compatibility penalties, and we state the assumptions under which the source-conditioned branch is continuous.

Elastoplastic surrogates and neural constitutive models. EPNN embeds constitutive structure in an elastoplastic surrogate (Eghbalian et al., 2023); finite-element–DeepONet coupling interfaces local numerical solutions with an operator (Yin et al., 2022); FE-integrated neural networks learn elastic and elastoplastic solids (Zhang et al., 2025); and NeuberNet maps low-fidelity elastic solutions to local elastic-plastic fields (Grossi et al., 2025). These studies support using a reliable physics reference, but their geometries, constitutive settings, and inverse interfaces differ from the staged slope benchmark considered here. The correction in YSNO is consequently described as nonlinear minus elastic, not as a learned plastic internal variable.

Scientific-ML benchmarks and inverse problems. PDEBench covers multiple PDE families; PDEArena and The Well organize broad scientific simulation suites; AirfRANS provides aerodynamic simulations; LagrangeBench targets particle dynamics; and OpenFWI targets seismic full-waveform inversion (Takamoto et al., 2022; **?**; Ohana et al., 2024; Bonnet et al., 2022; Toshev et al., 2023; Deng et al., 2022). Bayesian inverse formulations describe uncertainty through a likelihood and prior (Stuart, 2010), while derivative-informed operators target the sensitivities needed in outer-loop inverse problems (O'Leary-Roseberry et al., 2024). SlopeBench complements these resources with staged elastoplastic response and heterogeneous strength inversion. Its independent units are geometries, not individual material realizations, and its simulated labels are not measurements from thousands of independent field sites. We therefore distinguish the probabilistic formulation from the point-reconstruction evidence reported here.

A Operator Construction, Proofs, and Inverse-Protocol Details

A.1 Operator construction and scope

We analyze YSNO by separating a continuous function-space map from the finite-grid arrays used to evaluate it. Our construction follows an encoding–network–reconstruction decomposition, a useful distinction in operator learning (Kovachki et al., 2023; Lanthaler et al., 2022). The approximation bounds below are proved directly for the stated YSNO realization; no universal approximation theorem is assumed.

Two mesh scales play different roles. The internal reference-grid scale $\ell > 0$ is fixed together with the network parameters θ. The external input/output scale $h > 0$ tends to zero. We denote the resulting realization by $G_{\theta,\ell}$, and call it the *fixed-reference realization* of YSNO. This construction is distinct from a physical-offset architecture whose local layers are evaluated on each external grid, including YSNO-ρ. It is also distinct from directly applying the original fixed-pixel CNN at each new resolution.

A.1.1 Domain, inputs, and physical target

Let $\Omega \subset \mathbb{R}^2$ be a bounded polygonal Lipschitz domain of positive area. Fix its geometry, boundary conditions, initial state, and loading protocol. Let

$$X = L^2(\Omega; \mathbb{R}^{d_a}), \qquad Y = L^2(\Omega; \mathbb{R}^{d_u}), \qquad K \Subset X. \tag{17}$$

Here K is a nonempty compact set of admissible, nondimensionalized input fields, and $d_u = 2$ for displacement. All norms on physical fields use the same domain and physical area measure. Different input channels may have fixed positive scaling factors, absorbed into the definition of X.

For the material/load interpretation, write $a = (m, f)$, where $m = (E, \nu, c, \phi, \varrho)$ and the prescribed load is $f(t, x) = \lambda(t) f(x)$. The initial state and λ are the same for all inputs. This removes varying load history as an unrecorded input, but does not by itself establish existence or uniqueness of the terminal state.

YSNO also consumes elastic trial fields. The analysis permits either of two precise input contracts: those fields are supplied as additional channels of a, or a fixed-dimensional, locally Lipschitz trial-state computation is included in the internal map below. In the first contract, any theorem in terms of material/load inputs alone additionally requires control of the map producing the supplied channels. Errors in that computation are accounted for in Section A.4. Geometry masks and coordinate conventions are fixed metadata, not thresholded functions of the variable input a.

Assumption A.1 (Physical solution map, used only for physical-error bounds)**.** For every $a \in K$, the prescribed evolution has a specified unique terminal displacement $\mathcal{S}(a) \in Y$. When a finite-sample uniform bound is used, $\mathcal{S} : K \to Y$ is continuous.

This is a premise about the physical problem, not a consequence of the network architecture. In particular, smooth loads and a fixed loading path do not prove this premise for a non-associated elastoplastic model. The network consistency results below do not require Assumption A.1. We do not invoke an elasticity coercivity argument to establish it for the elastoplastic evolution.

A.1.2 Fixed-reference realization of the complete network

Choose a fixed reference partition $\{Q_j^\ell\}_{j=1}^{N_\ell}$ of Ω with positive cell areas. Define channelwise cell-average encoding by

$$(E_\ell a)_j = \frac{1}{|Q_j^\ell|} \int_{Q_j^\ell} a(x)\, dx, \qquad \|z\|_\ell^2 = \sum_{j=1}^{N_\ell} |Q_j^\ell|\, |z_j|^2. \tag{18}$$

Fixed reshaping, fixed padding, and insertion of geometry/coordinate arrays are included in the finite-dimensional network $F_{\theta,\ell} : U_\ell \to \mathbb{R}^{M_\ell d_u}$, where $U_\ell \subset \mathbb{R}^{N_\ell d_a}$ is an open admissible set. It contains the local backbone, source router, normalized integral branches, output projections, and the prescribed elastic lifting under the chosen input contract. Internal convolutions, dilation, normalization groups, padding, and denominator floors are held fixed as h changes.

Choose fixed continuous, piecewise affine or bilinear shape functions $\psi_j^\ell \in H^1(\Omega)$ on a conforming reference mesh, and let

$$D_\ell z = \sum_{j=1}^{M_\ell} z_j \psi_j^\ell, \qquad G_{\theta,\ell} = D_\ell F_{\theta,\ell} E_\ell : X_{\mathrm{ad}} \longrightarrow Y, \tag{19}$$

Here $X_{\mathrm{ad}} = E_\ell^{-1}(U_\ell)$; the assumptions below ensure $K_* \subset X_{\mathrm{ad}}$. A globally defined network permits $X_{\mathrm{ad}} = X$, but no extension outside the admissible set is needed for our results. The input and output reference meshes need not have the same indexing; $F_{\theta,\ell}$ explicitly maps between their coefficient arrays. Homogeneous displacement constraints can be enforced by choosing basis functions with the required zero trace. A fixed nonhomogeneous lifting can be added to D_ℓ; its H^1 seminorm must then be included in the output bound. For simplicity, subsequent formulas use the homogeneous case.

Equation (19) defines an operator on L^2 equivalence classes: it uses cell averages, not point evaluation of arbitrary L^2 inputs. It is a mathematical realization of the complete finite network. It does not assert that the released inference pipeline already implements these particular encoding and decoding maps.

A.2 Stability of the YSNO map

A.2.1 Source-normalized moments

At a fixed reference grid, one integral branch has the form

$$A_r(z) = \sum_j \omega_j w_{rj}(z), \qquad B_r(z) = \sum_j \omega_j w_{rj}(z) V_{rj}(z),$$
$$m_r(z) = \frac{B_r(z)}{\max\{A_r(z), \epsilon_\ell\}}, \qquad k_i(z) = W_o \frac{1}{\sqrt{R}} \sum_{r=1}^{R} \rho_{ri}(z) m_r(z) + b_o. \tag{20}$$

Here $\omega_j > 0$ are fixed weights, $w_{rj} = q_j \varphi_{rj} \geq 0$ is the source weight, V_{rj} is a vector of source values, and ρ_{ri} is the receiver weight. All masks can be incorporated in w and the final fixed output map. The source router uses nonnegative gates and softplus values; the source and receiver bases use softmax weights. The floor $\epsilon_\ell > 0$ is fixed. Both $\omega_j = 1$ (the unweighted code sum) and physical cell areas are allowed for this fixed-grid result, but they define different maps unless the floor is rescaled consistently.

Lemma A.2 (Stable normalized moment). *For $A, A' \geq 0$, vector numerators B, B', and $\epsilon > 0$,*

$$\left\| \frac{B}{\max\{A, \epsilon\}} - \frac{B'}{\max\{A', \epsilon\}} \right\| \leq \frac{\|B - B'\|}{\epsilon} + \frac{\|B'\|}{\epsilon^2} |A - A'|. \tag{21}$$

Consequently, if A_r, B_r are Lipschitz with constants $L_{A,r}, L_{B,r}$ and $\|B_r\| \leq M_{B,r}$ on a set, m_r is Lipschitz there with constant $L_{B,r}/\epsilon_\ell + M_{B,r} L_{A,r}/\epsilon_\ell^2$. Moreover, if $\|V_{rj}\| \leq M_V$, then $\|m_r\| \leq M_V$.

Proof. Set $d = \max\{A, \epsilon\}$ and $d' = \max\{A', \epsilon\}$. Insert B'/d, use $d, d' \geq \epsilon$, and note $|d - d'| \leq |A - A'|$. This proves (21). The Lipschitz claim follows by substitution. Finally, positivity of the weights gives $\|B_r\| \leq A_r M_V$, and $A_r / \max\{A_r, \epsilon_\ell\} \leq 1$. □

Bounded Lipschitz source/value/receiver maps and the finite sum in (20) therefore give a bounded Lipschitz branch on bounded admissible feature sets. A strictly positive lower bound on source mass is unnecessary here: the fixed positive floor already controls the quotient. Such a mass bound becomes relevant for a different limit in which an effective floor tends to zero.

Area weights and the code floor. For a uniform grid with cell area q_ℓ, the exact identity is

$$\frac{\sum_j w_j V_j}{\max\{\sum_j w_j, \epsilon_{\mathrm{code}}\}} = \frac{q_\ell \sum_j w_j V_j}{\max\{q_\ell \sum_j w_j, q_\ell \epsilon_{\mathrm{code}}\}}. \tag{22}$$

Thus matching a fixed physical floor ϵ_c requires $\epsilon_{\mathrm{code}} = \epsilon_c / q_\ell$. In the present construction q_ℓ never changes with h. If one instead evaluates at native grid scale h, its effective floor is $q_h \epsilon_{\mathrm{code}}$; a

consistency claim then requires a separate quadrature argument. On nonuniform grids, unweighted sums cannot generally be converted by one scalar cell area.

Proposition A.3 (Finite-rank structure and exact initialization)**.** *For fixed features z, the transport in (20), viewed as a linear map of the source values with all source/receiver coefficients frozen, factors through the R moment vectors. For scalar values its matrix has rank at most R. If every added integral output projection has $W_o = 0$ and $b_o = 0$, then the complete model equals its retained local baseline, using identical lifting, masks, encoding, and decoding.*

Proof. Flatten the rank-specific scalar source values into columns indexed by (r, j). The transport matrix before the output projection is

$$T_{i,(r,j)}(z) = \frac{1}{\sqrt{R}} \rho_{ri}(z) \frac{\omega_j w_{rj}(z)}{\max\{A_r(z), \epsilon_\ell\}}. \tag{23}$$

Every column with the same rank index r is a multiple of the receiver vector $(\rho_{ri})_i$. The column space is therefore contained in the span of these R vectors, so its rank is at most R. Zero output weights and bias make each added correction identically zero. Addition to the unchanged baseline, followed by the same decoder, gives the second assertion. □

The full branch is nonlinear because its coefficients depend on the input. The frozen-coefficient rank statement is not a rank bound on its Jacobian or an assertion that all outputs lie in one input-independent R-dimensional subspace. Exact initialization equality is an algebraic statement; it does not guarantee nondegradation after training updates.

A.2.2 From the finite network to function-space stability

Let P_h^{in} denote the external cell-average projection defined in Section A.3, and set

$$K_* = K \cup \bigcup_{0<h\le h_{\max}} P_h^{\mathrm{in}} K. \tag{24}$$

Assumption A.4 (Admissible finite-dimensional computation)**.** There is a compact convex set $C_\ell \subset U_\ell$ containing $E_\ell K_*$, and $F_{\theta,\ell}$ is locally Lipschitz on U_ℓ. All network parameters, preprocessing conventions, geometry metadata, normalization floors, and internal grid dimensions are fixed.

The neighborhood condition excludes singular preprocessing. In particular, material transformations must stay away from poles of $\tan\phi$; normalizing denominators must be bounded away from zero or clamped; and input-dependent changes of masks or padding cannot be silently included. Locally Lipschitz maps on a neighborhood of a compact convex set have a finite Lipschitz constant on that set: cover it by finitely many local Lipschitz neighborhoods and partition each line segment into sufficiently short pieces. Denote such a constant by $L_{F,\theta,\ell}$, measured from $\|\cdot\|_\ell$ to the Euclidean output-coefficient norm.

Lemma A.5 (Encoder stability and decoder regularity)**.** *The encoder satisfies $\|E_\ell a\|_\ell \le \|a\|_X$. There are finite constants $C_{D,0}, C_{D,1}$, depending on the fixed decoder, such that*

$$\|D_\ell z\|_Y \le C_{D,0}\|z\|_2, \qquad |D_\ell z|_{H^1(\Omega)} \le C_{D,1}\|z\|_2. \tag{25}$$

Proof. Jensen's inequality on each reference cell gives $|Q_j^\ell|\,|(E_\ell a)_j|^2 \le \int_{Q_j^\ell} |a|^2$; sum over cells. For decoding, the triangle and Cauchy–Schwarz inequalities give, for $k = 0, 1$, $\|\sum_j z_j \psi_j^\ell\|_{H^k} \le \|z\|_2 (\sum_j \|\psi_j^\ell\|_{H^k}^2)^{1/2}$. The same argument with gradients proves the seminorm bound. □

Proposition A.6 (Continuity, stability, and output regularity)**.** *Under Assumption A.4, $G_{\theta,\ell}$ is Lipschitz on K_* with a constant $L_{\theta,\ell} = C_{D,0} L_{F,\theta,\ell}$. Furthermore, there is a finite $B_{\theta,\ell}$ such that*

$$\sup_{a\in K} |G_{\theta,\ell}(a)|_{H^1(\Omega)} \le B_{\theta,\ell}. \tag{26}$$

Proof. Apply the encoder bound, the Lipschitz inequality for $F_{\theta,\ell}$ on C_ℓ, and the L^2 decoder bound in succession. Since $F_{\theta,\ell}$ is continuous on compact C_ℓ, $M_F = \sup_{z\in C_\ell} \|F_{\theta,\ell}(z)\|_2 < \infty$. Equation (25) gives $B_{\theta,\ell} = C_{D,1} M_F$. □

This output regularity concerns the reconstructed network field, not the unknown physical displacement. The constants may depend strongly on ℓ, θ, and the admissible set; they are only required to be independent of the external mesh scale h.

Checking the finite network assumption. On a fixed array, affine convolutions are bounded linear maps. Under a compatible discrete ℓ^2 norm, a convolution with kernels W_s has norm at most $\sum_s \|W_s\|$ when its shift/padding operators have norm at most one; otherwise their norms must be included. Lipschitz activations multiply layer constants, and a residual block $z + H(z)$ has constant at most $1 + L_H$. Products of bounded Lipschitz features are Lipschitz, and Lemma A.2 controls the source-normalized moments.

Group normalization on a *fixed* group of n scalars also admits a direct bound. With $P = I - \mathbf{1}\mathbf{1}^\top/n$, its normalized part is $g(z) = Pz/(\|Pz\|_2^2/n+\epsilon_N)^{1/2}$, $\epsilon_N > 0$. The radial and tangential eigenvalues of the derivative with respect to Pz are bounded by $\epsilon_N^{-1/2}$. Since $\|P\|_2 \le 1$, affine channel scaling by γ gives a Lipschitz bound $\|\gamma\|_\infty/\sqrt{\epsilon_N}$. Equivalence of finite-dimensional norms gives the corresponding weighted bound. This establishes stability on the fixed internal array; it does not establish consistency of spatial statistics under changing padding or native grid size.

If a deformable layer is used, its interpolation must be Lipschitz both in feature values and in sampling coordinates on the relevant bounded set. On a fixed reference grid, continuous bilinear interpolation with a stable boundary extension has this property, with a coordinate constant depending on neighboring feature differences and ℓ^{-1}. A Lipschitz offset predictor then gives a Lipschitz composite. Weight norms alone do not control such a layer.

A.3 External discretization and cross-grid convergence

Let $\mathcal{T}_h$ be conforming partitions of Ω into convex cells of maximum diameter at most h. Define the cell-average projection and piecewise constant reconstruction as one map

$$(P_h v)|_Q = \frac{1}{|Q|}\int_Q v(x)\,dx, \qquad Q \in \mathcal{T}_h. \tag{27}$$

Use this definition channelwise for $P_h^{\mathrm{in}} : X \to X$ and $P_h^{\mathrm{out}} : Y \to Y$. Distinct input/output meshes are permitted; using one symbol h means their maximum diameters are both at most h. All discrete outputs below are identified with reconstructed functions.

Lemma A.7 (Projection estimates). *For $v \in L^2(\Omega)$ and $u \in H^1(\Omega)$,*

$$\|P_h v\|_{L^2} \le \|v\|_{L^2}, \qquad \|P_h u - u\|_{L^2} \le C_P h |u|_{H^1}. \tag{28}$$

Moreover, $P_h v \to v$ for every $v \in L^2(\Omega)$, uniformly when v ranges over a compact subset of $L^2(\Omega)$.

Proof. Jensen's inequality proves contraction. On each convex cell, the mean-zero Poincare inequality bounds $\|u - u_Q\|_{L^2(Q)} \le C_P \operatorname{diam}(Q)\|\nabla u\|_{L^2(Q)}$ with a cell-independent constant. Square and sum to obtain the second estimate. Approximate any L^2 function by an H^1 function and use the contraction to get strong convergence. Finally, cover a compact set by finitely many L^2 balls of radius η. For a center v_i within η of v, contraction gives $\|P_h v - v\| \le 2\eta + \|P_h v_i - v_i\|$. The finite maximum tends to zero; then let $\eta \downarrow 0$. □

Define the exact externally discretized realization by

$$\widehat{G}_{\theta,\ell,h}(a) = P_h^{\mathrm{out}} G_{\theta,\ell}(P_h^{\mathrm{in}} a), \qquad \alpha_K(h) = \sup_{a\in K} \|P_h^{\mathrm{in}} a - a\|_X. \tag{29}$$

For piecewise constant input data, $E_\ell P_h^{\mathrm{in}} a$ can be computed by weighted overlap areas between external and reference cells. Output cell averages are integrals of the fixed decoder basis. Approximating these transfers rather than computing them exactly adds an implementation error treated below.

Theorem A.8 (Uniform discretization error for fixed-reference YSNO). *Under Assumption A.4, for every external mesh,*

$$\sup_{a\in K} \|\widehat{G}_{\theta,\ell,h}(a) - G_{\theta,\ell}(a)\|_Y \le \mathcal{E}_{\theta,\ell}(h),$$
$$\mathcal{E}_{\theta,\ell}(h) := L_{\theta,\ell}\alpha_K(h) + C_P B_{\theta,\ell} h \longrightarrow 0. \tag{30}$$

If, in addition, $\alpha_K(h) \le Ah^s$ *for some* $0 < s \le 1$*, then*

$$\mathcal{E}_{\theta,\ell}(h) \le L_{\theta,\ell} A h^s + C_P B_{\theta,\ell} h. \tag{31}$$

Proof. Abbreviate $G = G_{\theta,\ell}$. Add and subtract $P_h^{\mathrm{out}} G(a)$:

$$\begin{aligned} \|P_h^{\mathrm{out}} G(P_h^{\mathrm{in}} a) - G(a)\|_Y &\le \|P_h^{\mathrm{out}}[G(P_h^{\mathrm{in}} a) - G(a)]\|_Y + \|(P_h^{\mathrm{out}} - I)G(a)\|_Y \\ &\le L_{\theta,\ell} \|P_h^{\mathrm{in}} a - a\|_X + C_P h |G(a)|_{H^1}. \end{aligned} \tag{32}$$

Here both a and $P_h^{\mathrm{in}} a$ belong to K_*, the output projection is a contraction, and Proposition A.6 applies. Take the supremum over K and use Lemma A.7. The additional rate assumption gives (31). □

For a nonorthogonal output transfer with norm at most C_R, the first term in (32) acquires the factor C_R, and its own output approximation estimate must replace (28). Nodal interpolation cannot be substituted for P_h without these checks.

Corollary A.9 (Comparison of two external grids)**.** *For any two external grids* h, k *under the same assumptions,*

$$\sup_{a \in K} \|\widehat{G}_{\theta,\ell,h}(a) - \widehat{G}_{\theta,\ell,k}(a)\|_Y \le \mathcal{E}_{\theta,\ell}(h) + \mathcal{E}_{\theta,\ell}(k). \tag{33}$$

Proof. Insert the common continuous field $G_{\theta,\ell}(a)$ and apply Theorem A.8 twice. □

The conclusion is asymptotic consistency, not exact equality on finite meshes. Refinement of h does not refine the internal grid ℓ, increase network capacity, or retrain θ.

A.3.1 When an algebraic input rate is available

For uniformly H^1-bounded input channels, the same projection estimate gives $\alpha_K(h) \le Ah$, hence an $O(h)$ network discretization bound. More generally, on shape-regular meshes a uniform fractional Sobolev bound with $0 < s \le 1$ gives a corresponding cell-average estimate $\alpha_K(h) \le Ah^s$. Compactness alone supplies convergence, not a power-law rate.

For layered materials, a useful separate case is a family of inputs that are piecewise H^1, uniformly bounded in value, and have uniformly bounded broken H^1 seminorms. Assume additionally that the union U_h of cells intersecting material interfaces satisfies $|U_h| \le C_\Gamma h$ uniformly over the input family. This geometric condition holds, for example, for a uniformly controlled finite collection of regular interfaces and compatible mesh families. On uncut cells use Poincare; on cut cells a uniform value bound controls the squared error by a constant times $|U_h|$. Therefore

$$\|P_h^{\mathrm{in}} a - a\|_X^2 \le C_1 h + C_2 h^2, \qquad \alpha_K(h) = O(h^{1/2}). \tag{34}$$

This is an upper bound under stated uniform conditions, not a claim that every experiment has exactly half-order convergence. Interface-fitted meshes may yield a stronger bound. We analyze reconstruction inside Ω; discontinuous zero extension across its boundary need not belong to H^1 on an enclosing rectangle.

A.4 Implementation and elastic-input errors

Let $\widetilde{G}_{\theta,h} : K \to Y$ denote a deployed model after its specified reconstruction to the physical domain. Define its discrepancy from the analyzed realization by

$$\eta_{\theta,\ell}(h) := \sup_{a \in K} \|\widetilde{G}_{\theta,h}(a) - \widehat{G}_{\theta,\ell,h}(a)\|_Y, \qquad \mathcal{D}_{\theta,\ell}(h) := \eta_{\theta,\ell}(h) + \mathcal{E}_{\theta,\ell}(h). \tag{35}$$

Whenever $\eta_{\theta,\ell}(h)$ is finite, the triangle inequality gives

$$\sup_{a \in K} \|\widetilde{G}_{\theta,h}(a) - G_{\theta,\ell}(a)\|_Y \le \mathcal{D}_{\theta,\ell}(h). \tag{36}$$

This identity-based error decomposition is useful only when the implementation discrepancy is independently bounded. It does not prove that $\eta_{\theta,\ell}(h)$ vanishes for an arbitrary implementation.

For a quantitative example, suppose the internal encoded array has error at most $e_{\mathrm{enc}}(h)$ in $\|\cdot\|_\ell$, both the exact and perturbed arrays belong to C_ℓ, the finite network evaluation has coefficient error at most $e_{\mathrm{net}}(h)$, and output transfer contributes at most $e_{\mathrm{dec}}(h)$ in Y. Then

$$\eta_{\theta,\ell}(h) \le C_{D,0}\big(L_{F,\theta,\ell} e_{\mathrm{enc}}(h) + e_{\mathrm{net}}(h)\big) + e_{\mathrm{dec}}(h). \tag{37}$$

To prove this, insert the exact network evaluation at the perturbed array, then the exact evaluation at the unperturbed array, use the decoder bound, and finally add the transfer error. Approximate elastic input channels can be included in e_{enc} after encoding. For raw material/load inputs, their error must first be bounded in the chosen input norm; differentiating an L^2 displacement estimate does not provide a strain error estimate.

Why native-resolution evaluation is a separate question. A fixed-pixel convolution on a uniformly continuous field has the form

$$(C_h v)(x) = \sum_s W_s v(x + h\xi_s) \longrightarrow \Big(\sum_s W_s\Big) v(x), \tag{38}$$

uniformly on locations where a common uniformly continuous boundary extension is available. The assertion follows from the bound $\|C_h v - (\sum_s W_s)v\|_\infty \le \sum_s \|W_s\|\, \omega_v(h \max_s |\xi_s|)$, where ω_v is a modulus of continuity. Thus a fixed pixel stencil can have a continuum limit, but that limit collapses its physical neighborhood. It is generally not the fixed-reference map in (19). Native-grid normalization and padding introduce further differences. Our full-network theorem therefore applies to (29). Native-grid inference is covered by the resolution-indexed result below only after its additional discrepancy in (35) and the stated uniformity assumptions are controlled.

A.5 Complete YSNO error under changing receptive fields

The fixed-reference result above isolates external projection error. To state a bound for the complete YSNO when it is evaluated on a sequence of native grids, we also track the physical locations sampled by each local stencil. This section is conditional: the assumptions below are the conditions needed to turn a native-grid checkpoint into a resolution-indexed operator family.

For a local layer ℓ and stencil entry s, let $\delta_{\ell,s} \in \mathbb{R}^2$ be the desired physical offset and let $\delta_{\ell,s,h}$ be the offset represented on a grid whose maximum cell diameter is h. Define

$$\varepsilon_{\mathrm{rf}}(h) := \max_{\ell,s} \|\delta_{\ell,s,h} - \delta_{\ell,s}\|_2. \tag{39}$$

Let $e_{\mathrm{grid}}(h)$ be a nonnegative bound for the sum of the uniform errors from input encoding and interpolation, physical quadrature, supplied elastic lifting, and output reconstruction. For the continuum intermediate feature fields $z_\ell(a)$ generated by the complete network, define their uniform modulus on K by

$$\omega_K(r) := \max_\ell \sup_{a\in K} \sup_{\|x-y\|_2 \le r} \|z_\ell(a)(x) - z_\ell(a)(y)\|_2. \tag{40}$$

Assumption A.10 (Resolution-stable native-grid family)**.** The admissible set K is compact. On the bounded feature sets generated from K, every local backbone block, router, source-normalized integral branch, elastic lift, and output head is Lipschitz with a constant independent of h. Interpolation and padding are uniformly stable, all normalization denominators are bounded below by an h-independent constant $\underline{\epsilon} > 0$, and the source/value/receiver fields in each integral branch are uniformly bounded. The feature modulus $\omega_K(r)$ is finite and satisfies $\omega_K(r) \to 0$ as $r \downarrow 0$. The non-offset parameters, channel dimensions, and normalization parameters are held fixed across h; any parameter-transfer error is included in $e_{\mathrm{grid}}(h)$ or $\eta_{\mathrm{impl}}(h)$. If a denominator floor is allowed to vanish with h, it is additionally required that every source mass is bounded below uniformly on K.

Let $\mathcal{N}_{\theta,\delta} : K \to Y$ denote the complete continuum YSNO with physical offsets $\{\delta_{\ell,s}\}$, and let $\mathcal{N}_{\theta,\delta_h,h}$ denote the corresponding grid realization with offsets $\{\delta_{\ell,s,h}\}$. Write S_h for the input transfer to the native grid and R_h for the output reconstruction. In the fixed-reference construction, these transfers reduce to the exact projection/reconstruction maps in (29). Here the complete network includes the local backbone, shared source router, all source-normalized integral branches, the prescribed elastic lift, and the output heads.

Theorem A.11 (Uniform error bound for complete YSNO)**.** *Under Assumption A.10, there are constants $C_{\mathrm{net}}, C_{\mathrm{rf}}$ independent of h such that*

$$\sup_{a\in K} \|R_h \mathcal{N}_{\theta,\delta_h,h} S_h a - \mathcal{N}_{\theta,\delta} a\|_Y \le C_{\mathrm{net}} e_{\mathrm{grid}}(h) + C_{\mathrm{rf}} \omega_K(\varepsilon_{\mathrm{rf}}(h)) . \tag{41}$$

If $\widetilde{\mathcal{N}}_{\theta,h}$ is the deployed implementation and

$$\eta_{\mathrm{impl}}(h) := \sup_{a\in K} \left\| R_h \widetilde{\mathcal{N}}_{\theta,h} S_h a - R_h \mathcal{N}_{\theta,\delta_h,h} S_h a \right\|_Y , \tag{42}$$

then the right-hand side of (41) is augmented by $\eta_{\mathrm{impl}}(h)$. Thus the complete deployed-model error is uniformly bounded whenever $e_{\mathrm{grid}}(h)$, $\omega_K(\varepsilon_{\mathrm{rf}}(h))$, and $\eta_{\mathrm{impl}}(h)$ are uniformly bounded.

Proof. For one local block, interpolation and the offset perturbation give a propagated error of the form

$$e_{\ell+1}(h) \le L_{\ell+1} e_\ell(h) + C_{\ell+1} \left(e_{\mathrm{grid}}(h) + \omega_K(\varepsilon_{\mathrm{rf}}(h)) \right) .$$

The constants are independent of h by Assumption A.10. Induction through the finite local backbone and router gives the same form for their outputs. Lemma A.2 and the uniform denominator bound propagate it through each source-normalized integral branch; the physical quadrature, elastic-lift, and reconstruction terms are included in $e_{\mathrm{grid}}(h)$. The finite output heads preserve the estimate, and summing the finite recurrence yields (41). The additional deployed term follows from the triangle inequality and (42). □

Corollary A.12 (Receptive-field adjustment and an algebraic rate)**.** *Suppose that*

$$\varepsilon_{\mathrm{rf}}(h) \le C_\delta h^\beta, \qquad \omega_K(r) \le C_\omega r^\gamma, \qquad e_{\mathrm{grid}}(h) \le C_g h^p, \qquad \eta_{\mathrm{impl}}(h) \le C_\eta h^q \tag{43}$$

for positive exponents p, q, β, γ. Then

$$\sup_{a\in K} \left\| R_h \widetilde{\mathcal{N}}_{\theta,h} S_h a - \mathcal{N}_{\theta,\delta} a \right\|_Y \le C_\eta h^q + C_{\mathrm{net}} C_g h^p + C_{\mathrm{rf}} C_\omega C_\delta^\gamma h^{\beta\gamma} = O\left(h^{\min\{p,q,\beta\gamma\}} \right) . \tag{44}$$

On a regular grid, choosing pixel offsets or interpolation points whose physical locations approximate fixed offsets gives $\beta = 1$. Hence the full YSNO error remains bounded and converges when the transfer and implementation errors vanish.

Corollary A.13 (Fixed-pixel receptive field without adjustment)**.** *If integer pixel offsets $\xi_{\ell,s}$ are kept fixed, then $\delta_{\ell,s,h} = h\xi_{\ell,s}$. Relative to the collapsed local limit $\delta_{\ell,s} = 0$, one has $\varepsilon_{\mathrm{rf}}(h) = O(h)$, so Theorem A.11 gives a uniform bound and convergence to that collapsed operator under Assumption A.10. Relative to a target with a fixed nonzero physical offset, however, $\varepsilon_{\mathrm{rf}}(h)$ approaches a nonzero receptive-field floor. The error can therefore remain bounded, but consistency with that fixed nonlocal physical operator is not established without adjusting the stencil or adding a separate implementation correction.*

The theorem is conditional for the released implementation. In particular, fixed-pixel convolutions, spatial GroupNorm, batch-dependent padding, and per-sample coordinate normalization require either a resolution-stable implementation or an independently verified bound on $\eta_{\mathrm{impl}}(h)$. The result supplies a complete-YSNO error certificate under the stated receptive-field, normalization, transfer, and implementation assumptions; it does not claim an unconditional mesh-independent theorem for an arbitrary native-grid checkpoint.

A.6 Error relative to the physical solution

Assume the existence part of Assumption A.1 and define the uniform model error, assumed finite, by

$$\varepsilon_{\mathrm{model}} := \sup_{a\in K} \|G_{\theta,\ell}(a) - \mathcal{S}(a)\|_Y . \tag{45}$$

Finiteness follows, in particular, if $\mathcal{S}$ is continuous on K.

Theorem A.14 (Model, discretization, and implementation error)**.** *Under the preceding assumptions,*

$$\sup_{a\in K} \|\widetilde{G}_{\theta,h}(a) - \mathcal{S}(a)\|_Y \le \varepsilon_{\mathrm{model}} + \mathcal{D}_{\theta,\ell}(h). \tag{46}$$

For exact implementation of (29), $\eta_{\theta,\ell}(h) = 0$. If $\alpha_K(h) \le Ah^s$, its bound becomes $\varepsilon_{\mathrm{model}} + L_{\theta,\ell} A h^s + C_P B_{\theta,\ell} h$.

Proof. For each a, insert $G_{\theta,\ell}(a)$ between the deployed prediction and $\mathcal{S}(a)$. Bound the two differences by (36) and (45), then take the supremum. □

Defining $\varepsilon_{\text{model}}$ does not estimate it from data. Neither smaller external cells nor parameter sharing implies $\varepsilon_{\text{model}} \to 0$. Approximation capacity, training, input information loss, and the fixed decoder can all contribute to this term.

Proposition A.15 (Fixed-encoding information barrier)**.** *If $a, b \in K$ satisfy $E_\ell a = E_\ell b$, then every map of the form (19) satisfies*

$$\max\{\|G_{\theta,\ell}(a) - \mathcal{S}(a)\|_Y, \|G_{\theta,\ell}(b) - \mathcal{S}(b)\|_Y\} \geq \tfrac{1}{2}\|\mathcal{S}(a) - \mathcal{S}(b)\|_Y. \tag{47}$$

Proof. Identical encodings imply identical predictions. Insert this common prediction between $\mathcal{S}(a)$ and $\mathcal{S}(b)$ and apply the triangle inequality. □

Consequently, the present consistency result is not a universal approximation theorem for a fixed internal grid. A capacity-growing approximation theorem would need its own hypotheses and control of the encoding and decoder as they change.

A.7 Connecting finite samples to a uniform physical-error bound

Let $a_1, \ldots, a_n \in K$ be a δ-net in the input norm, meaning

$$\forall a \in K, \quad \min_{1 \leq i \leq n} \|a - a_i\|_X \leq \delta. \tag{48}$$

Let $y_i \in Y$ be reconstructed numerical reference displacements with $\|y_i - \mathcal{S}(a_i)\|_Y \leq \eta_{\text{ref}}$. At a common observation grid h_0, set

$$\varepsilon_{\text{obs}} = \max_i \|\widetilde{G}_{\theta,h_0}(a_i) - y_i\|_Y. \tag{49}$$

Continuity of $\mathcal{S}$ on compact K gives the uniform modulus

$$\omega_{\mathcal{S}}(t) := \sup_{\substack{a,b \in K \\ \|a-b\|_X \leq t}} \|\mathcal{S}(a) - \mathcal{S}(b)\|_Y, \qquad \omega_{\mathcal{S}}(t) \longrightarrow 0 \quad (t \downarrow 0). \tag{50}$$

Theorem A.16 (Deterministic finite-sample bound)**.** *Under Assumptions A.1 and A.4, the projection assumptions, and the sample conditions above,*

$$\begin{aligned}\sup_{a \in K} \|\widetilde{G}_{\theta,h}(a) - \mathcal{S}(a)\|_Y \leq{}& \varepsilon_{\text{obs}} + \eta_{\text{ref}} + L_{\theta,\ell}\delta + \omega_{\mathcal{S}}(\delta) \\ &+ \mathcal{D}_{\theta,\ell}(h_0) + \mathcal{D}_{\theta,\ell}(h).\end{aligned} \tag{51}$$

If $\mathcal{S}$ is $L_{\mathcal{S}}$-Lipschitz on K, the two coverage terms can be replaced by $(L_{\theta,\ell} + L_{\mathcal{S}})\delta$.

Proof. For $a \in K$, choose a_i within distance δ. Follow the chain

$$\begin{aligned}\widetilde{G}_{\theta,h}(a) &\longrightarrow G_{\theta,\ell}(a) \longrightarrow G_{\theta,\ell}(a_i) \longrightarrow \widetilde{G}_{\theta,h_0}(a_i) \\ &\longrightarrow y_i \longrightarrow \mathcal{S}(a_i) \longrightarrow \mathcal{S}(a).\end{aligned}$$

The six successive differences are at most $\mathcal{D}_{\theta,\ell}(h)$, $L_{\theta,\ell}\delta$, $\mathcal{D}_{\theta,\ell}(h_0)$, ε_{obs}, η_{ref}, and $\omega_{\mathcal{S}}(\delta)$, respectively. Sum and take the supremum. The Lipschitz special case follows from $\omega_{\mathcal{S}}(\delta) \leq L_{\mathcal{S}}\delta$. □

If sample i has its own grid h_i and label-error bound $\eta_{\text{ref},i}$, replace $\varepsilon_{\text{obs}} + \eta_{\text{ref}} + \mathcal{D}_{\theta,\ell}(h_0)$ by

$$\max_i \big\{\|\widetilde{G}_{\theta,h_i}(a_i) - y_i\|_Y + \eta_{\text{ref},i} + \mathcal{D}_{\theta,\ell}(h_i)\big\}. \tag{52}$$

The same proof establishes this form.

This is a deterministic coverage bound, not a probabilistic generalization theorem. Compactness ensures that finite δ-nets exist; it does not establish that a given training set has a small covering radius. A mean squared training loss is not ε_{obs}: for $\text{MSE} = n^{-1}\sum_i \|\widetilde{G}_{\theta,h_0}(a_i) - y_i\|_Y^2$, one only has the generic inequality $\varepsilon_{\text{obs}} \leq \sqrt{n\,\text{MSE}}$. All sample norms must be reconstructed physical L^2 norms or consistent area-weighted approximations to them. Numerical reference error must include spatial discretization, load stepping, solver tolerance, geometry representation, and transfer wherever relevant; no rate for these errors is assumed or proved here.

Table 4: Scope of the conditional fixed-reference results in this appendix.

Result	Conditions and interpretation
Moment stability	Fixed positive denominator floor and bounded Lipschitz encoded maps.
Full-network consistency	Fixed internal grid; stable encoding/network; continuous H^1 decoder; consistent external projections.
Native-grid error bound	Resolution-stable layer and normalization constants; controlled receptive-field mismatch; bounded transfer and implementation errors.
Deployment consistency	Additionally, the implementation discrepancy $\eta_{\theta,\ell}(h)$ must vanish.
Physical-error bound	Additionally, a specified single-valued physical solution map and finite model error.
Finite-sample bound	Additionally, verified sample coverage, reference-error control, and continuity of the physical map.

A.8 Two-stage outputs and interpretation of the result

For two prescribed stages, let $\widetilde{u}_s$ and u_s denote predicted and target displacements, $s = 1, 2$. Independently of the architecture,

$$\|(\widetilde{u}_2 - \widetilde{u}_1) - (u_2 - u_1)\|_Y \le \|\widetilde{u}_2 - u_2\|_Y + \|\widetilde{u}_1 - u_1\|_Y. \tag{53}$$

This follows by expanding the difference and applying the triangle inequality. Applying Theorem A.14 or A.16 separately to each stage gives a bound on the predicted displacement change. A physically sequential loading history must be represented in the input contract; the inequality does not turn two independent terminal-state evaluations into a history-dependent solver.

The analysis extends componentwise to separately decoded strain or stress heads if their corresponding spaces, decoder regularity, and target-map assumptions are imposed. An L^2 displacement error alone does not control its derivative, constitutive stress, or consistency between separately predicted fields. Likewise, nonnegative latent sources do not establish equality with physical plastic multipliers or exact complementarity.

For numerical verification of the discretization statement, fix θ, ℓ, all normalization parameters, the physical domain, and the same continuous input fields. Encode those fields on nested external meshes and compare predictions after reconstruction to a common physical space. Keep the model weights and reference grid unchanged. Such tests can diagnose violations of the implementation contract; finitely many mesh comparisons do not independently certify a uniform bound over K. Native-resolution tests should be reported separately because they evaluate a different family. The resolution-indexed theorem above applies to that family only when Assumption A.10 and the bound on $\eta_{\mathrm{impl}}(h)$ are verified.

The proved conclusion is that the explicitly constructed fixed-reference YSNO is a continuous nonlinear function-space map with a consistent external discretization. The resolution-indexed theorem additionally gives a complete-YSNO error bound whose terms are grid-transfer error, receptive-field mismatch, and implementation discrepancy. The proof does not establish universal approximation for a fixed configuration, an unconditional continuum theorem for native-grid checkpoints, convergence of training, or well-posedness of the underlying elastoplastic problem.

A.9 Mechanical Conventions and Training Details

Yield feature. For tension-positive in-plane stresses, define the Mohr-circle center and radius by

$$p_\sigma = \tfrac{1}{2}(\sigma_{xx} + \sigma_{yy}), \qquad r_\sigma = \sqrt{\tfrac{1}{4}(\sigma_{xx} - \sigma_{yy})^2 + \sigma_{xy}^2}. \tag{54}$$

The implemented trial feature is

$$\mathcal{Y}(\boldsymbol{\sigma}, c, \phi) = r_\sigma + p_\sigma \sin\phi - c\cos\phi. \tag{55}$$

Angles in trigonometric operations are in radians. A small radicand floor is used in the numerical implementation. Division by cohesion in Eq. (4) makes the feature dimensionless. Under uniaxial

tension, the positive mean stress raises the feature; compression lowers it. This sign convention must be preserved when converting finite-element outputs. Equation (55) is an in-plane yield-related feature. The three predicted in-plane stress components do not supply the full out-of-plane elastoplastic stress, so the feature should not be presented as an exact three-dimensional Abaqus Mohr–Coulomb admissibility check.

Source regularization. For dimensionless $\widehat{r}$ and source $\lambda \geq 0$, the smoothed Fischer–Burmeister penalty used in the loss implementation is

$$\mathcal{L}_{\mathrm{FB}} = \left\langle \left[\sqrt{\lambda^2 + (-\widehat{r})^2 + \delta_{\mathrm{FB}}} - \lambda + \widehat{r} \right]^2 \right\rangle_M, \qquad \delta_{\mathrm{FB}} > 0. \tag{56}$$

Without smoothing, a zero Fischer–Burmeister residual expresses nonnegativity and complementarity of its two arguments. Here it is a soft penalty on a learned latent source, not proof that the source equals the constitutive plastic multiplier. An additional penalty suppresses sources in confidently elastic trial regions:

$$\mathcal{L}_{\mathrm{src}} = \left\langle \mathrm{sigmoid}(-4r)\lambda^2 \right\rangle_M . \tag{57}$$

The trial confidence is detached when evaluating this source penalty. The conservative loss defaults are

$$\mathcal{L}_{\mathrm{stage},s} = \mathcal{L}_{\mathrm{data},s} + 10^{-4}(\mathcal{L}_y + \mathcal{L}_{\mathrm{FB}} + \mathcal{L}_{u\varepsilon})_s + 10^{-5}\mathcal{L}_{\mathrm{src},s}, \tag{58}$$

and $\mathcal{L} = \mathcal{L}_\Delta + 0.5(\mathcal{L}_{\mathrm{stage},1} + \mathcal{L}_{\mathrm{stage},2})/2$. These are the documented implementation defaults; historical checkpoint configurations must be preserved when reproducing the reported comparisons. The equilibrium term has zero default weight, and there is no default total-stress–total-strain elastic consistency term.

Kinematics and scaling. In physical coordinates, small-strain kinematics gives $\boldsymbol{\varepsilon} = (\varepsilon_{xx}, \varepsilon_{yy}, \gamma_{xy})^\top$ with $\gamma_{xy} = 2\varepsilon_{xy}$, as in Eq. (12). If coordinates are normalized as $\xi = 2(x - x_0)/L_x - 1$ and $\eta = 2(y - y_0)/L_y - 1$, then

$$\partial_x = \frac{2}{L_x}\partial_\xi, \qquad \partial_y = \frac{2}{L_y}\partial_\eta. \tag{59}$$

A physical compatibility penalty must include these factors and consistent displacement/strain scales. The current loss implementation differentiates on a canonical grid with spacings $2/(W-1)$ and $2/(H-1)$; consequently its historical regularizer should be interpreted as a scaled training penalty rather than an exact physical residual. The nonlinear labels also use large-displacement finite-element analysis. An exact finite-strain compatibility test would require compatible strain measures and the appropriate nonlinear kinematics. Neither of these implementation limitations is removed by typesetting Eq. (12) in physical coordinates.

A.10 Material Parameterization and Inverse Protocol

Evaluated decoder. The reported inversion uses four channels on a 4×8 coarse grid, giving 128 latent variables. The decoder interpolates these fields to the observation grid and applies constrained transforms for $(c_1, \phi_1, c_2, \phi_2)$. Its channels represent log cohesion, a bounded friction coordinate, a cohesion ratio and a friction-drop coordinate. The decoder applies

$$\begin{aligned} c_1 &= c_{\mathrm{scale}} \exp(\mathrm{clip}(t_1, -8, 8)), & \phi_1 &= \phi_{\min} + (\phi_{\max} - \phi_{\min})\,\mathrm{sigmoid}(t_2), \\ c_2 &= c_1\,\mathrm{sigmoid}(t_3), & \phi_2 &= \max\{\phi_{\min}, \phi_1 - \Delta\phi_{\max}\,\mathrm{sigmoid}(t_4)\}. \end{aligned} \tag{60}$$

Here the t_j are interpolated latent fields. The implementation uses $c_{\mathrm{scale}} = 20$ in the stored cohesion units and $\Delta\phi_{\max} = 20^\circ$, with friction bounds of 5° and 45°; angles are converted consistently for constitutive features. The transforms are differentiable almost everywhere, including a piecewise-differentiable lower clamp. This representation enforces stage weakening but does not give an unrestricted representation of arbitrarily sharp interfaces.

The inverse experiment holds geometry, E, ν, loads and recorded elastic companion fields fixed. Only strength fields are optimized. Therefore the reported inverse results verify use of material gradients through the correction model, not sensitivity through a changing elastic solve. The general dFEM sensitivity in Eq. (2) applies when its elastic inputs are varied.

Covariance-based fields. A distinct continuous-field construction is useful for the material generator and for possible future inverse decoders. Let $\zeta = (x, h)$ denote horizontal position and depth below the local ground surface, and let $\Sigma(\zeta)$ be a positive-definite matrix of squared local length scales. A Paciorek–Schervish covariance (Paciorek & Schervish, 2003) has the form

$$k(\zeta, \zeta') = s(\zeta)s(\zeta')\frac{|\Sigma(\zeta)|^{1/4}|\Sigma(\zeta')|^{1/4}}{|\overline{\Sigma}(\zeta, \zeta')|^{1/2}} \exp\left[-(\zeta - \zeta')^\top \overline{\Sigma}^{-1}(\zeta - \zeta')\right],$$
$$\overline{\Sigma}(\zeta, \zeta') = \tfrac{1}{2}[\Sigma(\zeta) + \Sigma(\zeta')]. \tag{61}$$

For support locations S, diagonalize the covariance matrix as $K_{SS} = V\Lambda V^\top$ and retain m positive eigenvalues. A truncated Gaussian field evaluated away from the support is

$$g(\zeta) = \mu(\zeta) + K_{\zeta S}V_m\Lambda_m^{-1/2}\mathbf{z}, \qquad \mathbf{z} \sim \mathcal{N}(0, I_m). \tag{62}$$

This is a finite-dimensional spectral construction with Nyström evaluation (Williams & Seeger, 2001); small eigenvalues require stabilization. It permits evaluation at new spatial locations, but this alone does not prove discretization consistency of an entire inverse pipeline. Smooth covariance fields also do not automatically represent discontinuous stratigraphy. Sharp interfaces require explicit interface variables, piecewise fields or another suitable representation. No results in Table 2 are attributed to this alternative decoder.

Cases, observations and computation. The reported comparison uses four parent cases that are jointly unseen by the compared models. It evaluates two observation modes, borehole displacement and the full displacement field. Every gradient-based reconstruction uses 200 forward evaluations. The five non-gradient rows in Table 2 use Abaqus for 200 iterations and are kept separate from the gradient rows. The table reports the resulting NMAE values and runtimes; this appendix does not add case identifiers, seed counts, or replay statistics that are not stated there.

For valid grid points Ω_h, the material recovery metric is

$$\text{NMAE} = \frac{1}{4}\sum_{j=1}^{4}\frac{\sum_{x\in\Omega_h}|\widehat{m}_j(x) - m_j(x)|}{\max\{\sum_{x\in\Omega_h}|m_j(x)|, \epsilon_m\}}. \tag{63}$$

Normalization uses the mean absolute reference magnitude for each material field, not its range. The table reports the mean $\pm$ standard deviation under the stated protocol; no additional aggregation or relative-improvement calculation is introduced here.

B SLOPEBENCH DATASET

B.1 DATASET SCOPE AND DESIGN

This appendix describes the completed construction of SLOPEBENCH, a geometry-conditioned finite-element dataset for learning staged elastoplastic responses of two-dimensional slopes. Following the organization used by scientific machine learning benchmarks such as PDEBench (Takamoto et al., 2022) and The Well (Ohana et al., 2024), we specify the data unit, stochastic input generator, numerical solver, stored fields, controlled variants, resolution protocol, quality checks, and intended tasks in one reproducible description. The benchmark is designed for forward response prediction, elastic-to-elastoplastic correction, and controlled material-field inversion. It is a simulation benchmark rather than a collection of independent field measurements.

We distinguish four levels of the data hierarchy. A *profile* is one closed two-dimensional slope domain. A *variant* is one deterministic realization of the material and loading generator on a profile. A *stage* is one of the two material states in the staged analysis. A *paired sample* contains the stage-1 and stage-2 records for one profile–variant pair, together with the elastic companion records and the corresponding elastoplastic Abaqus records. This distinction is important because the independent geometric sampling unit is a profile, whereas the 50 material variants of a profile are correlated conditional draws.

Table 5: Composition of the completed base corpus. The 9,828 paired samples are the grid-matched solver outputs used by the compact benchmark.

Quantity	Count	Description
Source profiles	200	Geometry bank before solver acceptance
Accepted profiles	197	Passed the final strengthened base solve
Excluded profiles	3	Unstable
Accepted engineered profiles	110	Designed, excavated, or benched surfaces
Accepted natural profiles	87	Natural-ground surfaces
Accepted digitized profiles	121	Derived from archived section drawings
Accepted procedural profiles	76	Generated to broaden geometry support
Complete paired samples	9,828	Stage-1/stage-2 grid-matched records
Nominal profile–variant pairs	9,850	197×50 before completeness filtering

The labels `artificial` and `natural` describe the surface type, not whether a profile is synthetic. An artificial profile denotes a designed, excavated, or benched engineering surface; a natural profile denotes existing or natural ground. Independently, each profile is tagged as drawing-derived or procedural. The excluded profiles remain in the geometry audit but are not treated as response samples; exclusion was based on solver completion and not on a surrogate or inverse-model score. Figure 6 illustrates several representative slope profiles from the dataset.

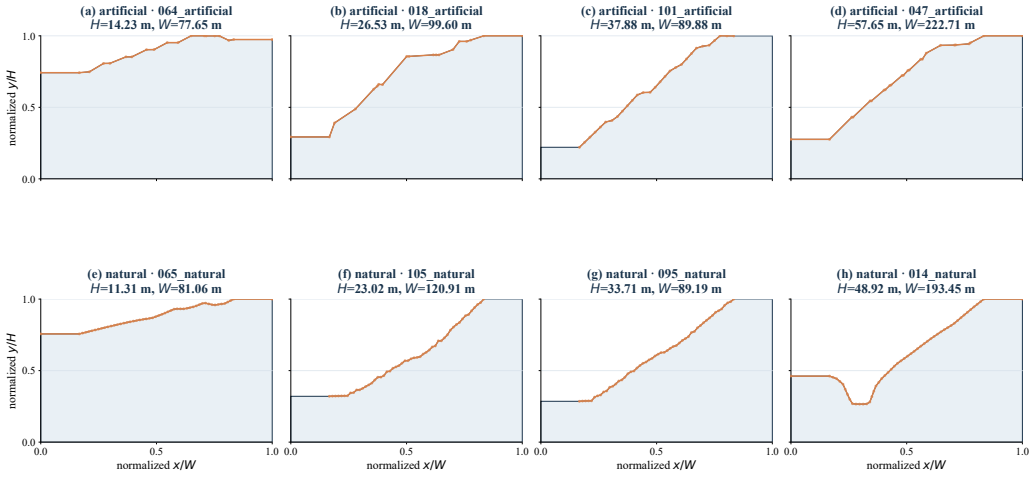

Figure 6: Drawing-derived artificial and natural slope profiles used in the dataset. The first row contains four artificial slope sections and the second row contains four natural slope sections.

B.2 Geometry Sources and Domain Curation

The drawing-derived profiles were digitized from engineering cross-section rasters in the project archive. Dimension chains and elevation axes were used for scale calibration when available; the per-profile metadata records the image path, scale anchors, control points, coordinate origin, modeling extensions, and warnings. Only the outer ground surface enters the present single-material domain. Retaining structures, drains, anchors, construction lines, text leaders, hatching, and geological lines that were not implemented as material interfaces are excluded from the surface polyline. These profiles are therefore geometry-conditioned simulation templates rather than surveyed as-built digital twins.

The procedural profiles augment the drawing bank with natural and excavated surfaces generated from a fixed seed. Natural profiles use a monotone elevation trend with low-amplitude sinusoidal perturbations. Excavated profiles use several inclined segments separated by benches; their height and width are sampled in a bounded engineering range and the polyline is resampled at approximately uniform arc length. Procedural profiles broaden the geometric support and do not represent known failure cases.

For both source groups, the visible surface is extended by horizontal side platforms, vertical side boundaries, and a horizontal bottom closure. The closed polygon is rigidly translated to a nonnegative local coordinate system. Translation changes neither dimensions nor mechanical response. The final domain is meshed as a two-dimensional plane-strain region. Across the accepted geometry bank, the mean slope height is 31.64 ± 14.20 m, the mean domain width is 131.25 ± 57.16 m, and the mean closed-domain area is $4{,}587 \pm 3{,}591$ m^2.

B.3 Nonstationary Geotechnical Random Fields

Spatial variability is modeled as a correlated random field rather than independent element-wise noise. This choice follows the geotechnical random-field view that marginal uncertainty and a scale of fluctuation jointly determine material variability (Phoon & Kulhawy, 1999; Fenton & Griffiths, 2008). For a physical point (x, y), let $y_s(x)$ denote the upper ground boundary and define the local depth

$$d(x, y) = \max\{0, y_s(x) - y\}, \qquad t(x, y) = \operatorname{clip}\left(\frac{d(x, y)}{H}, 0, 1\right), \tag{64}$$

where H is the profile height. The random field is evaluated in (x, d) coordinates so that its statistics follow the ground surface rather than the global Cartesian vertical direction. For a material zone, the query abscissa is its horizontal center and the query depth is the mean of the valid points in a fixed 3×3 zone stencil (with a centroid fallback); this is a deterministic representative query, not an area integral of the final element field.

B.4 Support Grid and Covariance

For each profile, a 34×28 Cartesian support lattice is clipped by the closed polygon. Each retained support point stores (x_i, d_i). The horizontal and depth correlation lengths and the marginal standard-deviation profile are

$$\ell_x(d) = 6 + 22t, \qquad \ell_d(d) = 0.45 + 3t, \tag{65}$$

$$\sigma(d) = 0.72 + 0.28\exp(-2.2t). \tag{66}$$

The covariance is a nonstationary anisotropic Gaussian kernel of the Paciorek–Schervish form (Paciorek & Schervish, 2003). With

$$\bar{\ell}^2_{x,ij} = \frac{\ell^2_{x,i} + \ell^2_{x,j}}{2}, \qquad \bar{\ell}^2_{d,ij} = \frac{\ell^2_{d,i} + \ell^2_{d,j}}{2}, \tag{67}$$

we use

$$K_{ij} = \sigma_i \sigma_j \frac{\sqrt{\ell_{x,i}\ell_{d,i}}\sqrt{\ell_{x,j}\ell_{d,j}}}{\sqrt{\bar{\ell}^2_{x,ij}\bar{\ell}^2_{d,ij}}} \tag{68}$$

$$\times \exp\left[-\frac{(x_i - x_j)^2}{\bar{\ell}^2_{x,ij}} - \frac{(d_i - d_j)^2}{\bar{\ell}^2_{d,ij}}\right] + 10^{-6}\delta_{ij}. \tag{69}$$

The small diagonal term stabilizes eigendecomposition and is recorded as part of the generator configuration.

B.5 Low-Rank Evaluation and Cross-Property Dependence

Let $K = V\Lambda V^\top$ be the eigendecomposition of the support covariance. Eigenvalues are sorted in descending order, and the smallest number of modes targeted to explain at least 95% of the support covariance energy is retained, with an upper cap of 320 modes. The achieved energy and retained rank are stored in case metadata. At arbitrary material-zone query points q, the truncated Nyström basis is

$$B(q) = K_{qS} V_m \Lambda_m^{-1/2}, \qquad z(q) = B(q)\xi. \tag{70}$$

The Nyström construction is used as a computational low-rank extension of the support covariance (Williams & Seeger, 2001); the retained rank is a numerical representation parameter, not a physical claim.

For the four latent channels (c, ϕ, E, γ), independent standard-normal coefficient vectors are mixed using the Cholesky factor of

$$R = \begin{bmatrix} 1.00 & 0.60 & 0.50 & 0.15 \\ 0.60 & 1.00 & 0.35 & 0.10 \\ 0.50 & 0.35 & 1.00 & 0.15 \\ 0.15 & 0.10 & 0.15 & 1.00 \end{bmatrix}. \tag{71}$$

Each latent channel is standardized over the current profile's material-zone query points. If $Z \in \mathbb{R}^{4\times N}$ is the raw latent matrix, then $\widetilde{Z}_{kj} = (Z_{kj} - \mu_k)/\max(s_k, 10^{-8})$, where μ_k and s_k are the channel mean and population standard deviation over the N queries. The delivered material fields are therefore correlated conditional fields, not four independent images. The finite realization need not have sample correlations exactly equal to the entries of R, because standardization, nonlinear transforms, clipping, and deterministic surface rules are applied after the correlated draw.

B.6 Material transformations

The standardized latent channels are mapped to physical parameters with depth trends, bounded variability, and explicit unit limits. The principal rules are summarized in Table 6. The case-shared reference modulus is sampled as $E_0 = 30{,}000 r_E$ kPa with $r_E \sim \mathcal{U}(0.70, 1.30)$; the final modulus field is bounded to $E_0(1 \pm 0.12)$. Poisson's ratio is fixed at $\nu = 0.30$ and the dilation angle at $\psi = 0°$.

Table 6: Physical material-field transformations used by the completed two-stage generator.

Field	Implemented background rule (before weak-band, surface, and profile stabilization)
c	Surface trend $c_0 \sim \mathcal{U}(9, 14)$ kPa; bottom target $c_b \sim \mathcal{U}(0.62, 0.88) c_{\max}$ with $c_{\max} = 30 + 0.6 d_{\max}$ kPa; approximately 15 m depth knots are made nondecreasing; correlated fluctuations have scale $3.4 + 2.2(d/d_{\max})$ kPa and multiplier $\mathcal{U}(0.65, 1.45)$; final range $[8, c_{\max}]$ kPa.
ϕ	Variant trend $\phi_0 \sim \mathcal{U}(22.5, 25.5)^\circ$; correlated fluctuation scale $2.6 r_\phi^\circ$ with $r_\phi \sim \mathcal{U}(0.65, 1.35)$; final range $[15, 32]^\circ$ before case stabilization.
E	Case-shared E_0; log-space depth-knot noise with standard deviation 0.035; correlated log-field scale $(0.055 + 0.025 e^{-2t}) r'_E$ with $r'_E \sim \mathcal{U}(0.55, 1.25)$; final range $E_0(1 \pm 0.12)$.
γ	Mean $18 + 0.75t$ kN/m^3, standard deviation $0.75 - 0.25t$ kN/m^3, and final range $[14, 24]$ kN/m^3.

Every profile–variant pair is deterministic. The seed is derived from the case identifier and variant tag using the project hash-seed routine. Independent tags are used for the background field, weak-band geometry, and stage severity; surface treatment is deterministic from the profile's artificial/natural label. Re-running the generator with the same profile, variant identifier, and source configuration reproduces the same zone-level parameters.

B.7 A Non-trained Differentiable Field Parameterization

The field generator has a useful differentiable core even though the offline solver pipeline also contains discrete acceptance operations. Let $b(q) \in \mathbb{R}^{1\times m}$ denote the Nyström row at a query point, let $L_R L_R^\top = R$, and let $\Xi \in \mathbb{R}^{4\times m}$ contain independent coefficient rows. For fixed profile geometry, support set, and retained eigenbasis, the four-channel latent field at q is

$$z(q; \Xi) = L_R \Xi b(q)^\top, \qquad \frac{\partial \operatorname{vec} Z}{\partial \operatorname{vec} \Xi} = B \otimes L_R. \tag{72}$$

where $B \in \mathbb{R}^{N\times m}$ stacks the query rows and $Z = L_R \Xi B^\top$. Channel-wise centering and standardization are differentiable wherever the standard deviations are nonzero; the Jacobian is obtained by composing Eq. 72 with that normalization. The differentiable core is therefore a non-trained low-rank random-field model: its basis is calculated from the prescribed covariance and its coefficients are sampled or optimized, rather than fitted by gradient descent on the dataset. Smooth depth trends, exponentials, affine material transforms, and the smooth weak-band intensity

$$I(q) = \exp\left[-\frac{r(q)^2}{2w^2}\right] \tag{73}$$

are differentiable wherever the geometry and active branch are fixed.

The distinction between the differentiable core and the delivered data pipeline is essential. Zone-query construction, hard clipping, monotone cohesion-knot correction, surface reinforcement, weak-band candidate rejection, solver convergence, and Abaqus's return mapping introduce piecewise or discrete operations. Thus the complete offline chain is reproducible and contains a differentiable parameterized field core, but it is not claimed to be an everywhere-differentiable end-to-end Abaqus simulator. This statement is consistent with the broader differentiable-FEM literature (Xue et al., 2023) while making no claim that SlopeBench itself was generated by a differentiable FEM solver.

B.8 Weak Bands, Stages, and Surface Treatment

Each base MC variant contains one through-going weak band whose centerline is sampled as an upward-opening parabola or a shallow circular arc. Its target mean depth is sampled uniformly from $0.05H$ to $0.60H$. Candidates are accepted only when the centerline remains inside the domain and satisfies the depth-range, slope-range, and curvature constraints recorded in the generation metadata. Let $r(q)$ be the approximate normal distance to the centerline. For the medium-severity baseline, the nominal width is $0.045H$ and is clipped to $[1, 3]$ m, followed by a multiplicative $\mathcal{U}(0.85, 1.20)$ variation. The intensity in Eq. 73 uses this width.

For stage 2, the medium-severity reduction is

$$\alpha_c = 0.34 r_c, \qquad r_c \sim \mathcal{U}(0.80, 1.30), \tag{74}$$

$$\alpha_\phi = 2.0 r_\phi \text{ degrees}, \qquad r_\phi \sim \mathcal{U}(0.70, 1.40), \tag{75}$$

and

$$c_2 = \mathrm{clip}\{c_{\mathrm{bg}}(1 - \alpha_c I), 8, c_{\max}\}, \qquad \phi_2 = \mathrm{clip}\{\phi_{\mathrm{bg}} - \alpha_\phi I, 15, 32\}. \tag{76}$$

Stage 1 uses a factor sampled from $[0.45, 0.55]$ multiplying the two weak-band reductions. Two additional correlated fields increase stage-1 c and ϕ locally by factors in $[1.02, 1.15]$. The construction enforces the intended stage ordering before final stabilization: $c_2 \le c_1$ and $\phi_2 \le \phi_1$.

For engineered or excavated surfaces, zones within 4 m of the ground surface receive deterministic treatment in both stages: cohesion is multiplied by three, friction is raised to at least 24°, and the modulus is raised toward $1.20E_0$ before the final global modulus bound. Natural profiles do not receive this treatment.

The initial uniform base material is $(E, \nu, c, \phi, \psi) = (30{,}000\ \mathrm{kPa}, 0.30, 20\ \mathrm{kPa}, 24^\circ, 0^\circ)$. A profile-level stabilization procedure increases c and ϕ by successive factors of 1.10 until the base analysis converges, applies one additional 1.10 factor, caps ϕ at 35°, and synchronizes the resulting factor to both stages of all variants on that profile. The final solver acceptance is independent of any learned-model error. Figure 7 shows material-parameter fields across representative slope geometries.

B.9 Finite-Element Construction and Constitutive Model

The completed ground-truth pipeline uses Abaqus/Standard in two-dimensional plane strain. The material assignment grid consists of nominal 5.0×0.5 m rectangular zones. A free mesh of linear plane-strain triangles (CPE3) is generated at a target edge size of 0.5 m. The zones are a field-assignment grid rather than conforming geometric partitions: each element receives the material associated with the zone containing its centroid, with a nearest-zone fallback for edge cases. Geometry, mesh topology, node sets, field-variable schedule, and loads are held fixed across material variants.

The nonlinear model combines isotropic linear elasticity with the Abaqus Mohr–Coulomb yield surface

$$\tau_f = c + \sigma_n \tan\phi, \tag{77}$$

with a single zero-plastic-strain hardening entry. The fixed $\psi = 0^\circ$ produces non-associated flow for $\phi > 0$. Large-deformation kinematics are enabled and the unsymmetric solver is used. The bottom boundary fixes $u_x = u_y = 0$; the left and right vertical boundaries fix $u_x = 0$ and remain free vertically. Body force is assigned zone by zone as a downward force per unit volume.

Three static steps are applied:

1. `Step-1`: stage-1 material and self-weight $-\gamma$;

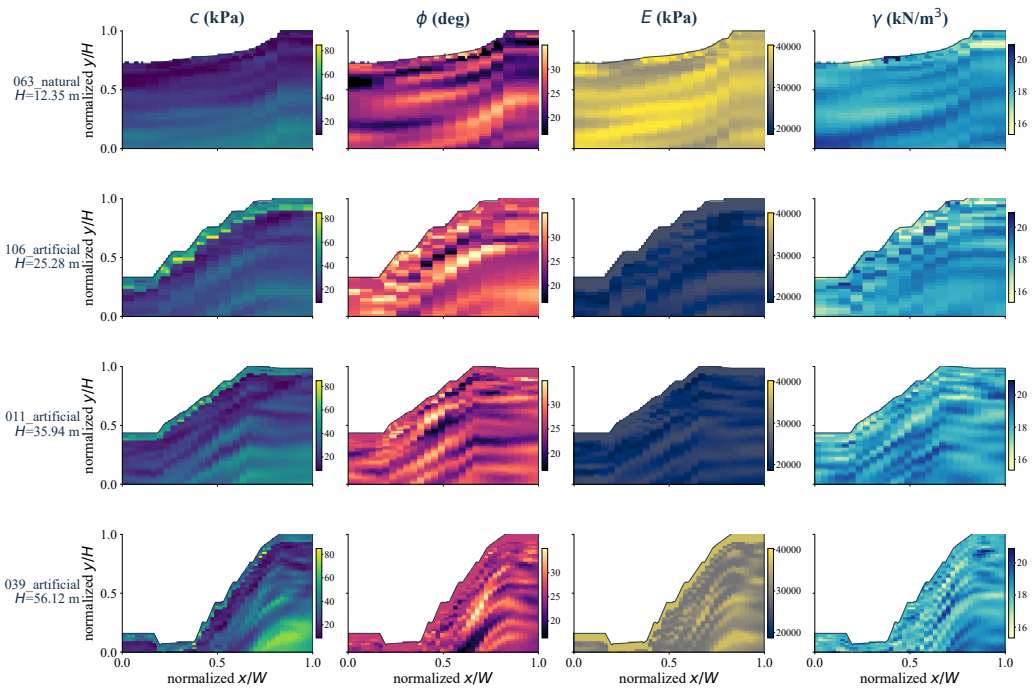

Figure 7: material-parameter fields across representative slope geometries.

2. `Step-Load`: stage-1 material, self-weight, and a depth-decaying rainfall-equivalent body-force increment;
3. `Step-2`: stage-2 material and self-weight $-\gamma$.

The field variable is 1.0 in the first two steps and 2.0 in the third. Rainfall is represented as an equivalent body-force increment over a depth $h_r = \min(8, 0.15H)$ m with decay length $\lambda_r = 0.5h_r$; this is not a coupled seepage or pore-pressure analysis.

For the DP variants, the same geometry, elastic parameters, loads, and stage schedule are retained while the strength surface is replaced by a plane-strain Drucker–Prager parameterization matched to (c, ϕ), with $K = 1$ and $\psi = 0°$. In the compression-positive convention used by the implementation, $p = -\operatorname{tr}(\boldsymbol{\sigma})/3$, $q = \sqrt{3J_2}$, and the linear surface is $F = q - p\tan\beta - d = 0$. The implementation uses

$$\beta = \arctan(\sqrt{3}\sin\phi), \qquad d = \sqrt{3}c\cos\phi, \tag{78}$$

consistent with the plane-strain matching convention in the Abaqus documentation. The DP variant changes the constitutive law and leaves the stochastic geometry/material realization otherwise paired.

For the baseline MC compact corpus, an elastic companion input is produced by removing only the Mohr–Coulomb and hardening blocks. Geometry, mesh, E, ν, γ, boundary conditions, field schedule, and body-force history are otherwise unchanged. These elastic solves provide a controlled reference response for learning elastoplastic corrections; they are not additional geometries or independent stochastic samples. The OOD material-field assets have family-specific schemas and should not be assumed to contain the same absolute elastic companion fields unless their manifest explicitly declares them.

B.10 Response Extraction and Compact Representation

The stored response is the final frame of `Step-Load` for stage 1 and the final frame of `Step-2` for stage 2. The paired displacement difference is therefore

$$\Delta\mathbf{u} = \mathbf{u}_{2,\text{Step}-2} - \mathbf{u}_{1,\text{Step}-\text{Load}}. \tag{79}$$

It includes the stage transition, removal of the temporary rainfall-equivalent increment, and the path-dependent state accumulated before stage 2. It is not a same-load, material-only degradation target.

For each selected frame, the extractor records element label, integration point, physical coordinates, displacement, in-plane stress (S_{xx}, S_{yy}, S_{xy}), and strain $(E_{xx}, E_{yy}, \gamma_{xy})$. Integration-point coordinates are used when available; otherwise the element centroid is used. Element displacement is the mean of its nodal values. For the forward compact corpus, detailed records are mapped to a 1 m physical representation grid by averaging all integration-point records that fall into the same valid cell. Cells outside the polygon are masked. The separate two-stage inverse contract retains the native material-zone grid (nominally 5 m in x by 0.5 m in y); it is not silently identified with the 1 m forward representation.

The compact sample stores the valid-cell flat indices, physical coordinate vectors, polygon boundary, and material/response arrays. Unpacking reconstructs the dense representation without changing the stored values. A field named `eps_p_target`, when present in the learning schema, is the nonlinear-minus-elastic total-strain correction; it is not a direct Abaqus PEEQ tensor and should not be interpreted as a plastic multiplier ground truth. Figure 8 shows empirical distributions of the generated dataset.

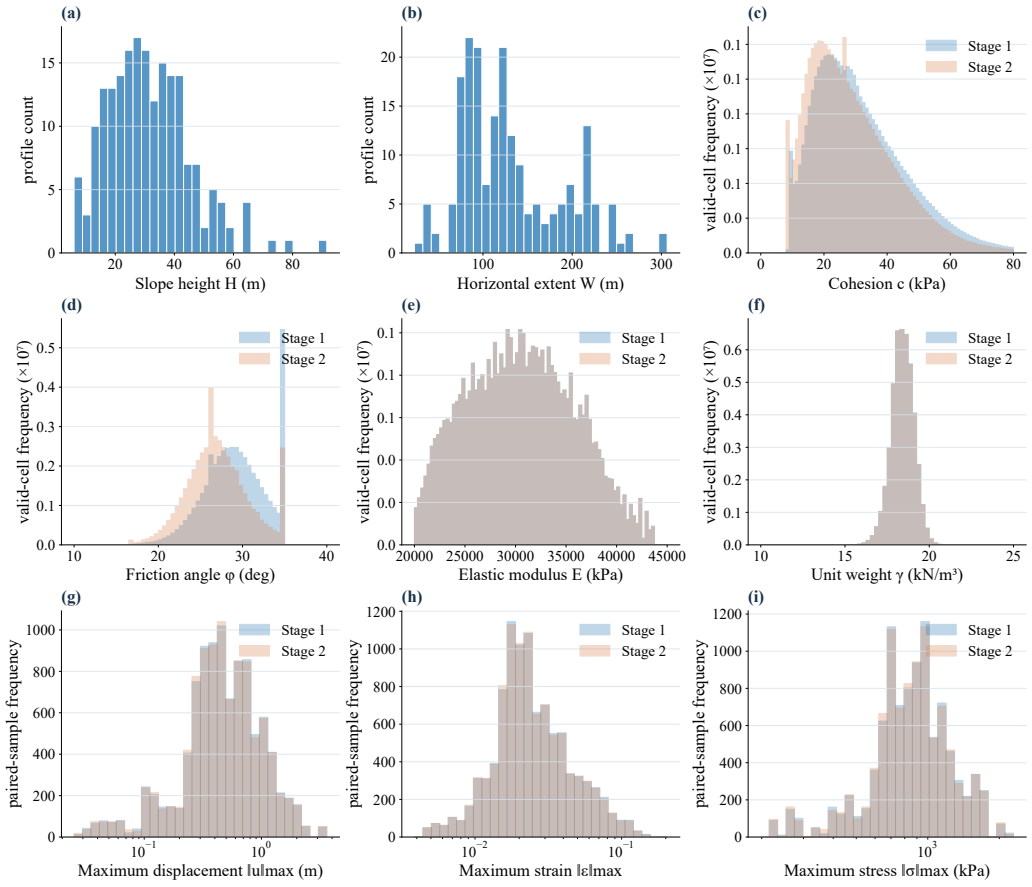

Figure 8: Empirical distributions of the generated dataset.

B.11 Controlled Variants

The completed physics and constitutive variants are summarized in Table 7. The four-way comparison is useful because it separates a material weak-band shift from a constitutive-law shift. The variants retain paired profile and variant identifiers whenever the source family contains the corresponding sample.

Table 7: Completed variant families and their current compact assets. Counts are samples after each family's own completeness filter.

Family	Samples	Parent profiles	Interpretation
MC-WB	9,828	197	Baseline Mohr–Coulomb with weak band
MC-NoWB	7,794	156	Mohr–Coulomb without weak band
DP-WB	7,799	156	Drucker–Prager with weak band
DP-NoWB	7,800	156	Drucker–Prager without weak band

The common four-family intersection contains 7,784 paired samples over 156 parent profiles. The family-specific counts differ because the OOD assets were generated and audited independently; a model evaluation must use the family manifest rather than infer completeness from the nominal 50-variant design. MC-NoWB removes the weak-band modification while retaining the background field and surface treatment. DP-WB retains the weak-band realization but changes the constitutive law. DP-NoWB combines both changes.

B.12 Resolution Variants and the Three Grid Levels

Three grids must be distinguished. The *finite-element grid* is the CPE3 solver mesh, whose target edge size is varied in the multi-resolution experiment. The baseline target edge is 0.5 m, and the tested mesh scales are $h/2$, h, $2h$, and $4h$. The *material-assignment grid* is the nominal 5.0×0.5 m zone grid used to write zone materials into the input file. The *learning representation grid* is the 1 m physical compact grid used to aggregate and store valid cells. These grids have different roles and are not interchangeable.

The random-field support lattice is a fourth, auxiliary discretization: its 34×28 points are clipped to the profile and used to construct the covariance basis, after which each material zone is evaluated at its fixed representative query and receives one (c, ϕ, E, γ) tuple. For every tested mesh scale, the CPE3 mesh is regenerated, the zone assignments are transferred to the new element centroids, and both nonlinear stages are rerun in Abaqus. Thus, the cross-resolution package changes the finite-element discretization while retaining the same physical case and loading contract; it is not produced by interpolating a single solver field.

The resolution package contains independently solved Abaqus mesh families at three non-baseline scales in addition to the baseline h corpus:

Table 8: Abaqus mesh-resolution families used for the cross-resolution evaluation. Each family contains independently rerun two-stage finite-element solutions at the stated target scale.

Label	Target mesh scale	Stage-1 samples	Stage-2 / two-stage samples
`h_div_2`	$h/2$	9,828	9,828 / 9,828
`h`	h	9,828	9,828 / 9,828
`h_mul_2`	$2h$	9,828	9,828 / 9,828
`h_mul_4`	$4h$	9,828	9,828 / 9,828

The package contains 88,452 compact files across the three tasks and the three non-baseline mesh resolutions. The physical domain extent, material realization, stage protocol, and case identifiers are preserved, while each resolution has its own Abaqus mesh, solver output, and field extraction. The multiresolution results therefore test response transfer across independently solved finite-element grids and support a mesh-resolution scalability claim for the evaluated family.

This distinction follows the resolution-aware organization used in neural-operator benchmarks (Li et al., 2021a; Gupta & Brandstetter, 2022): changing only the learning representation tests discretization sensitivity of a surrogate, whereas regenerating the finite-element mesh and rerunning the nonlinear solver tests transfer across physical solver resolutions. The current package implements the latter.

B.13 Splits, Tasks, and Quality Control

All variants of a profile inherit the same profile-level split. This prevents material variants of one polygon from appearing in different partitions. The split is case-disjoint, but it is not a claim of source-site or geographic independence because several profiles may share the same drawing archive or procedural generator. The 9,828-pair generation corpus is the source pool; each training or evaluation package records its own manifest, checkpoint identifiers, and task-specific split. We therefore report split counts from the manifest attached to a particular experiment rather than treating every variant as an independent site.

The main supported tasks are:

1. forward prediction of staged displacement, strain, and stress fields from geometry and material inputs;
2. prediction of elastoplastic corrections relative to the elastic companion;
3. cross-family OOD evaluation across weak-band and constitutive changes;
4. cross-resolution evaluation using independently solved Abaqus meshes; and
5. controlled material-field inversion under full-field or partial displacement observations.

The construction audit checks polygon closure, nonempty boundary sets, element-to-zone assignment, solver completion, stage pairing, grid and coordinate equality, finite stored values, material-range limits, and the intended stage ordering $c_2 \leq c_1$ and $\phi_2 \leq \phi_1$. Each compact manifest records the profile identifier, variant identifier, task, resolution label, source location, mesh configuration, solver status, and output path. The multiresolution audit additionally checks mesh validity, solver completion, field extraction, and finite values across the retained Abaqus resolutions.

B.14 Intended Use and Limitations

SlopeBenchis intended for controlled studies of geometry-conditioned surrogate operators, staged elastoplastic response prediction, and material-field inversion. The material priors are broad engineering simulation priors rather than fitted population distributions. The domains are two-dimensional plane-strain idealizations and omit explicit groundwater flow, consolidation, unsaturated behavior, dynamic loading, structural support elements, and three-dimensional effects. Rainfall is represented as an equivalent body-force increment rather than a coupled hydraulic analysis.

The source profiles carry digitization or procedural-geometry uncertainty, and multiple variants share one geometry. Consequently, a variant count is not a count of independent field sites. The mesh-resolution study covers the stated Abaqus mesh family and does not establish invariance to arbitrary meshing strategies. Finally, the staged displacement difference in Eq. 79 includes a load-path change and should not be interpreted as a pure material-degradation response. These limitations define the scope of the benchmark and the claims that can be supported by its completed assets.

C Additional Experimental Visualizations

This appendix provides qualitative visualizations that complement the quantitative tables in Section 5. The figures cover validation-loss dynamics, representative forward predictions, geometry- and distribution-shifted cases, and component ablations. Across all field-comparison panels, the columns report stage-difference displacement, strain, and stress together with the two stage-wise nonlinear-minus-elastic strain corrections. The rows identify the finite-element reference or the compared neural operators; the color scales are inherited from the supplied figures and are not re-normalized across cases.

C.1 Forward-prediction loss curves

The validation-loss curves summarize the optimization dynamics of the compared forward models over the training epochs. We report separate curves for displacement, strain, and stress so that convergence behavior can be inspected by output field. The curves are shown on a logarithmic ordinate, and lower values indicate smaller validation loss under the corresponding field normalization.

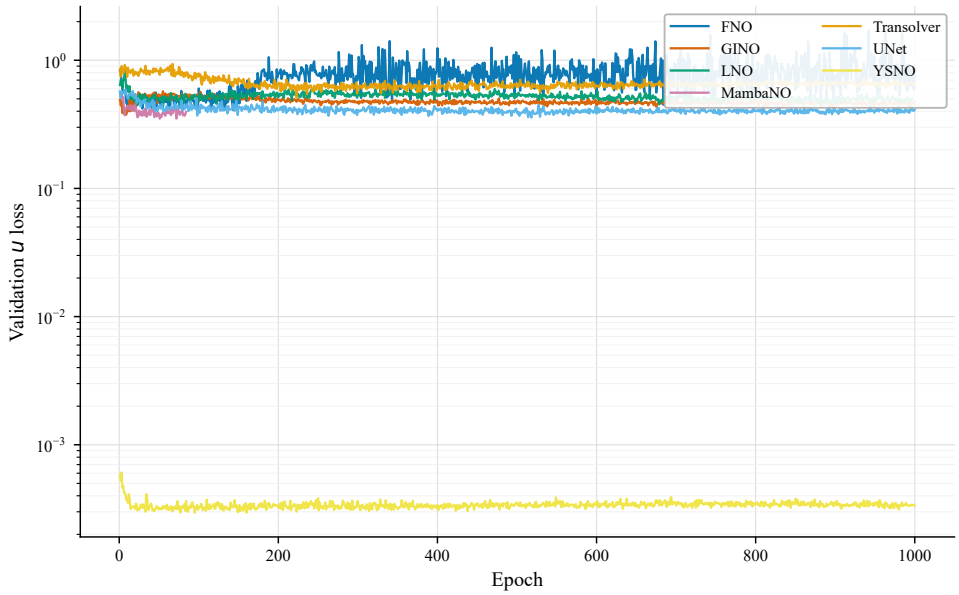

Figure 9: Validation loss for stage-difference displacement, Δu, across the compared forward models. Curves are plotted over training epochs on a logarithmic scale; lower values indicate smaller validation loss.

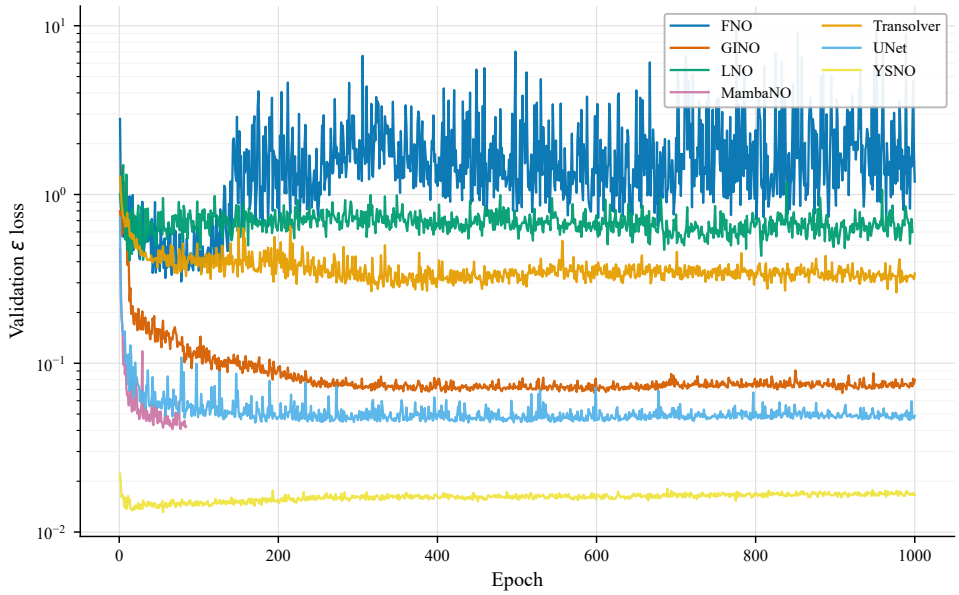

Figure 10: Validation loss for stage-difference strain, $\Delta \varepsilon$, across the compared forward models. The plot exposes field-specific convergence behavior under the same validation protocol.

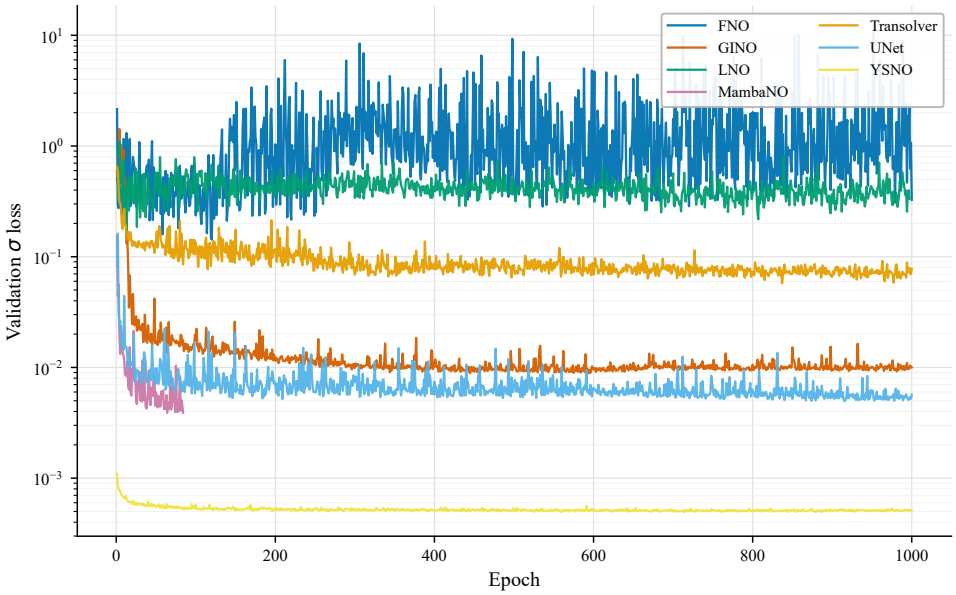

Figure 11: Validation loss for stage-difference stress, $\Delta\sigma$, across the compared forward models. Lower curves correspond to smaller normalized validation error for the stress field.

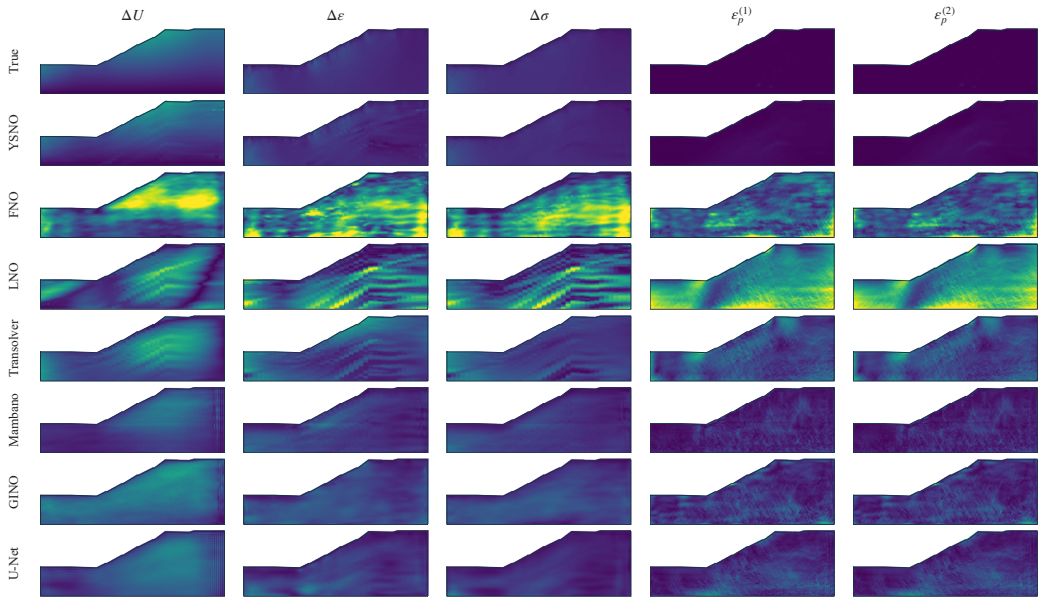

Figure 12: Qualitative forward-prediction comparison for artificial profile 014, variant $v0001$. Rows show the finite-element reference and predictions from YSNO, FNO, LNO, Transolver, MambaNO, GINO, and U-Net; columns show Δu, $\Delta\varepsilon$, $\Delta\sigma$, $\varepsilon_p^{(1)}$, and $\varepsilon_p^{(2)}$.

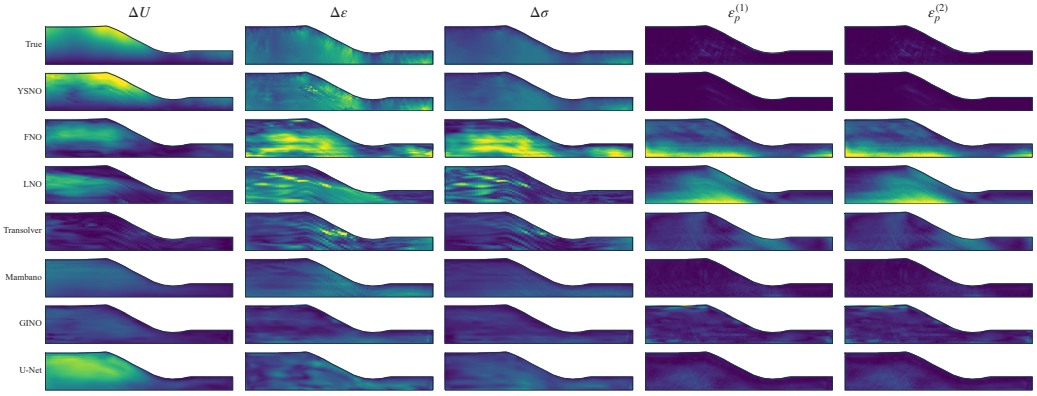

Figure 13: Qualitative forward-prediction comparison for natural profile 028, variant $v0001$. The panel reports the same five response fields and model ordering as Figure 12.

C.2 Qualitative forward predictions

The following panels compare the finite-element reference with predictions from YSNO and the baseline models on representative artificial and natural slope profiles. Each panel uses the same row ordering across models and the same field ordering across columns, allowing local response patterns and model-dependent artifacts to be inspected without reducing the comparison to a single scalar metric.

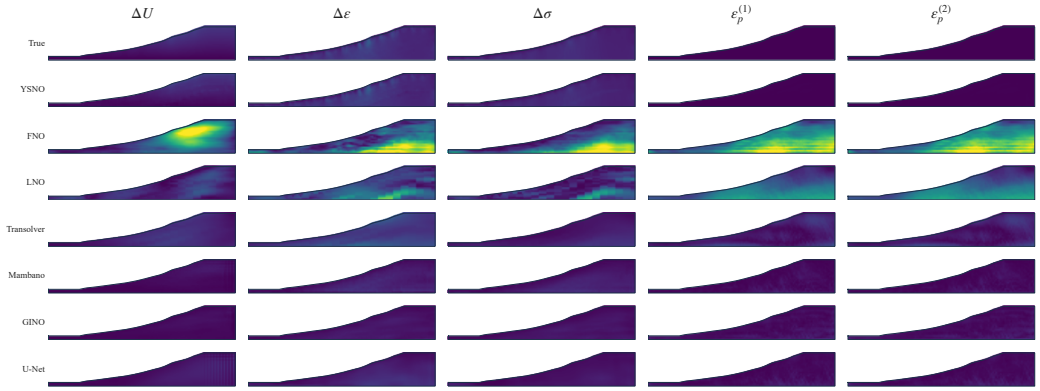

Figure 14: Qualitative forward-prediction comparison for natural profile 055, variant $v0001$. The visualization highlights how the compared operators represent the staged response over a natural-slope geometry.

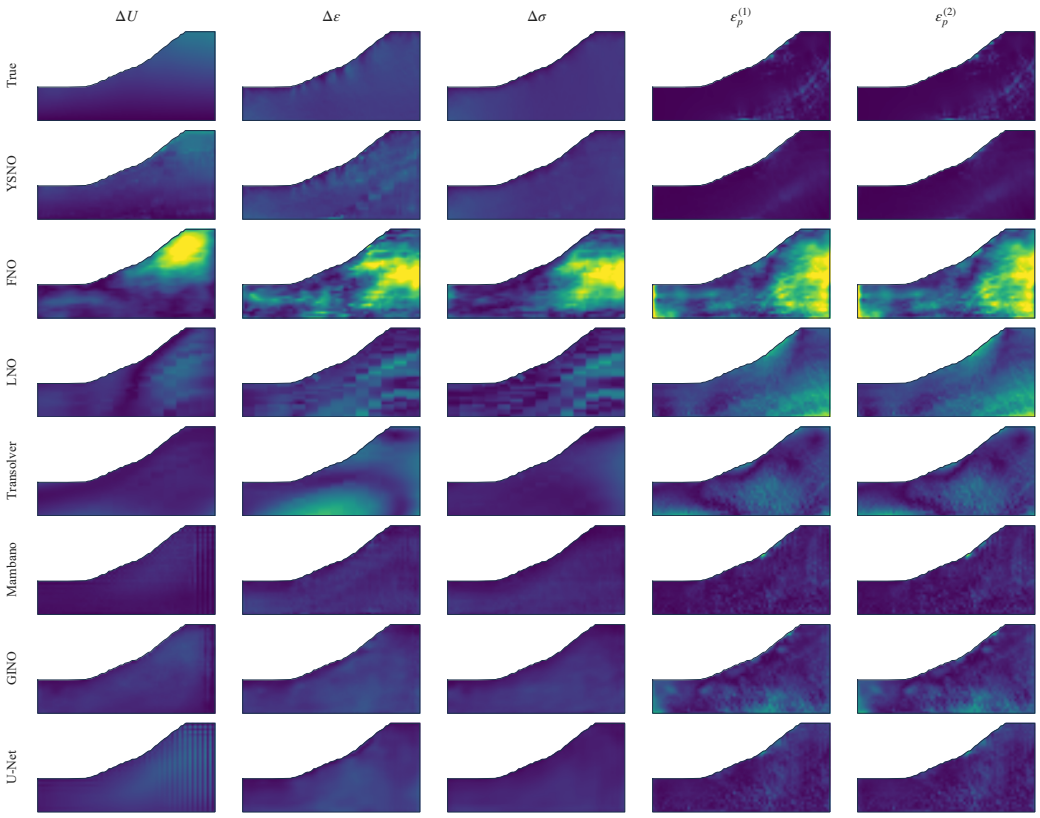

Figure 15: Qualitative forward-prediction comparison for natural profile 072, variant $v0001$. The five columns correspond to displacement, strain, stress, and the two stage-wise nonlinear-minus-elastic strain fields.

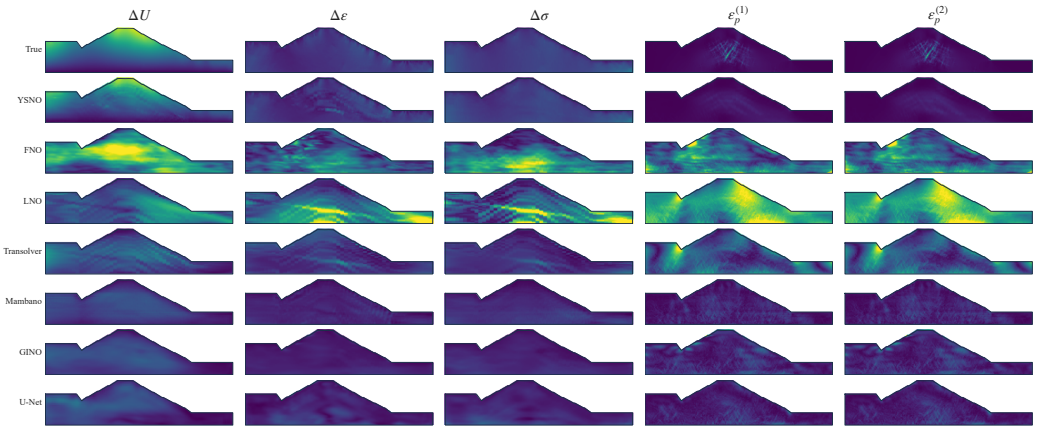

Figure 16: OOD forward-prediction comparison for artificial profile 052, variant $v0023$. The finite-element reference is compared with YSNO and the baseline operators across the five response-field columns.

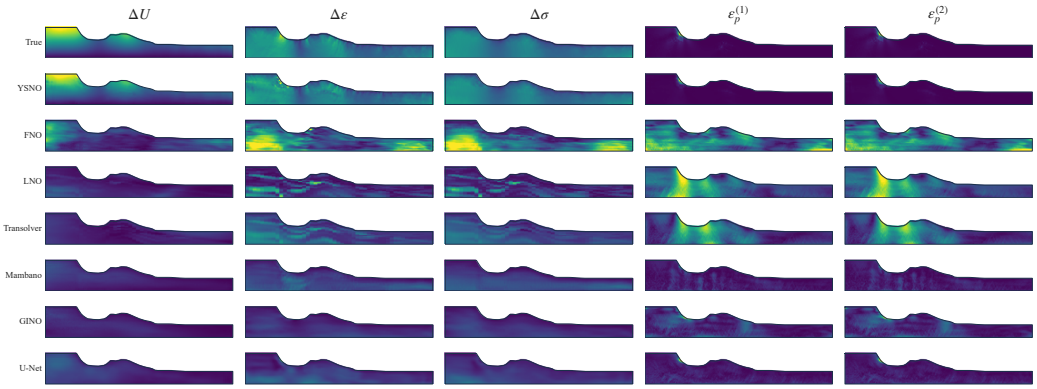

Figure 17: OOD forward-prediction comparison for natural profile 052, variant $v0021$. The panel emphasizes qualitative transfer of the predicted displacement, strain, stress, and stage-wise correction fields.

C.3 Out-of-distribution qualitative comparisons

These panels provide qualitative examples from the geometry-OOD evaluation. They complement the aggregate OOD metrics by showing how the predicted response fields preserve slope geometry, localization, and stage-dependent correction patterns when the evaluation cases differ from the training distribution.

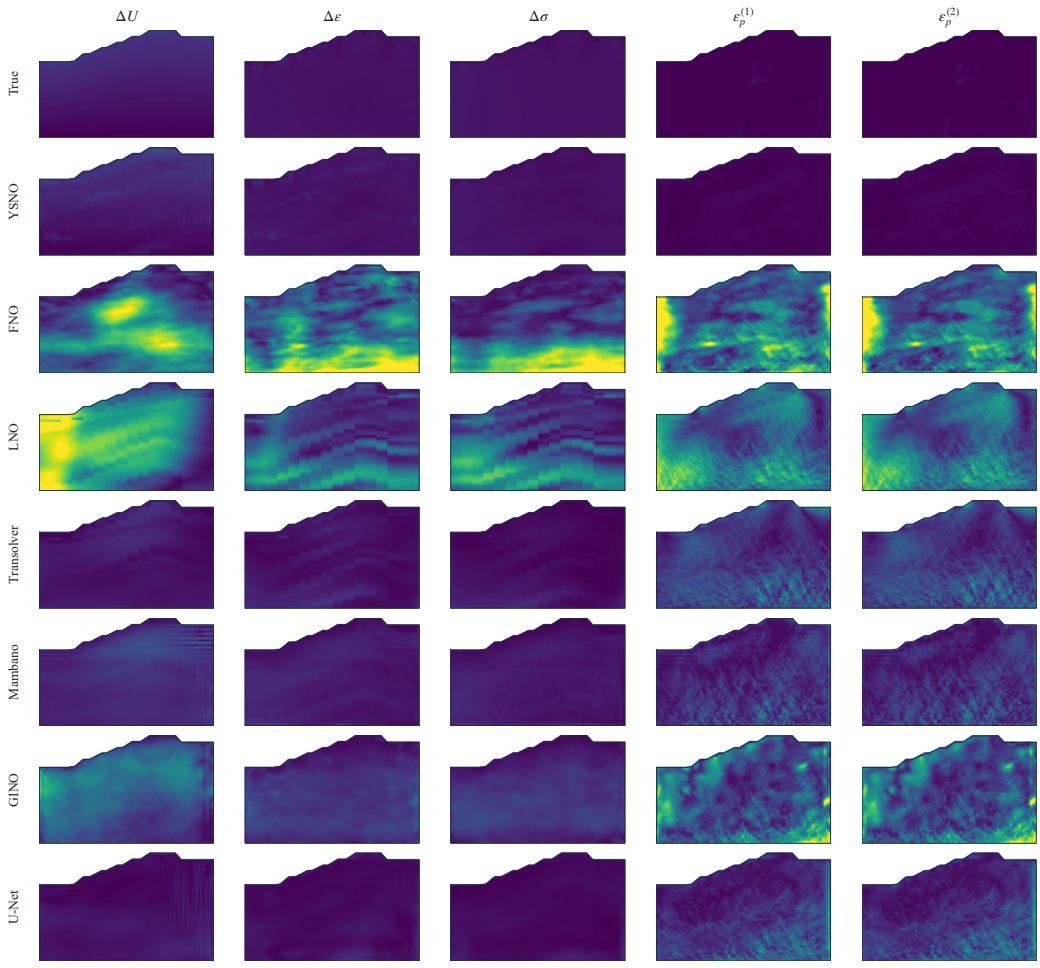

Figure 18: OOD forward-prediction comparison for artificial profile 063, variant $v0001$. Rows correspond to the reference and the compared models, while columns report the five predicted response fields.

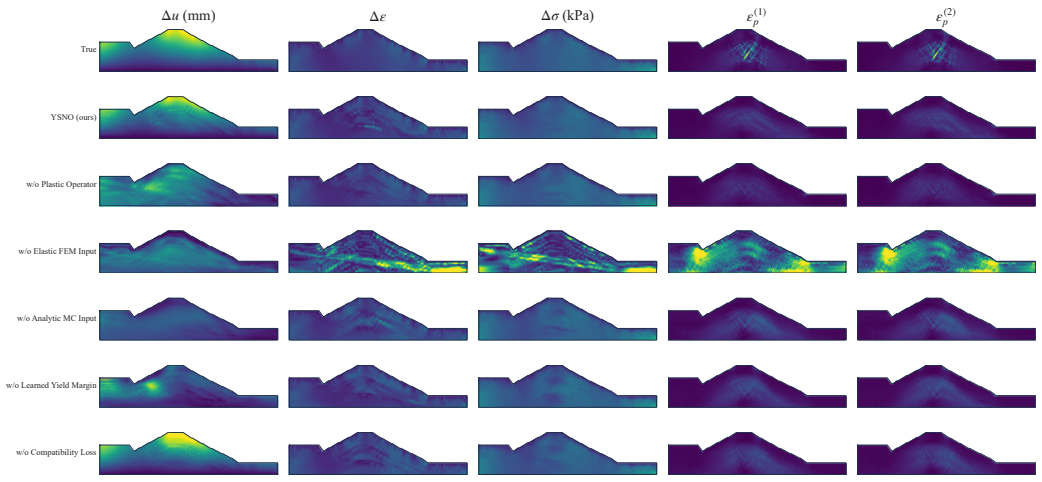

Figure 19: Ablation comparison for artificial profile 052, variant $v0023$. The rows compare YSNO with models that remove one physical or training component, and the columns report Δu, $\Delta\varepsilon$, $\Delta\sigma$, $\varepsilon_p^{(1)}$, and $\varepsilon_p^{(2)}$.

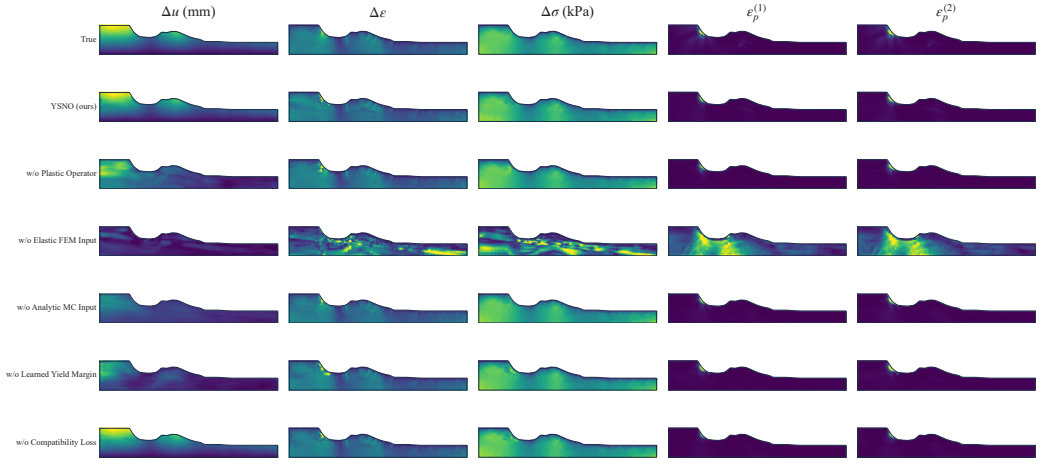

Figure 20: Ablation comparison for natural profile 052, variant $v0021$. The panel shows how removing individual YSNO modules changes the spatial structure of the predicted multi-field response.

C.4 Ablation visualizations

The ablation panels isolate the visual effect of removing one YSNO component at a time. In addition to the complete model, the rows show variants without the plastic operator, elastic FEM input, analytic Mohr–Coulomb input, learned yield margin, or compatibility loss. These visualizations complement the controlled scalar ablation results in Table 3.

The figures in this appendix are qualitative diagnostics. They use the same field definitions and evaluation conventions as the main experiments, but visual agreement alone does not replace the aggregate metrics, profile-level splits, or the independent Abaqus mesh-resolution evaluation reported in Section 5.

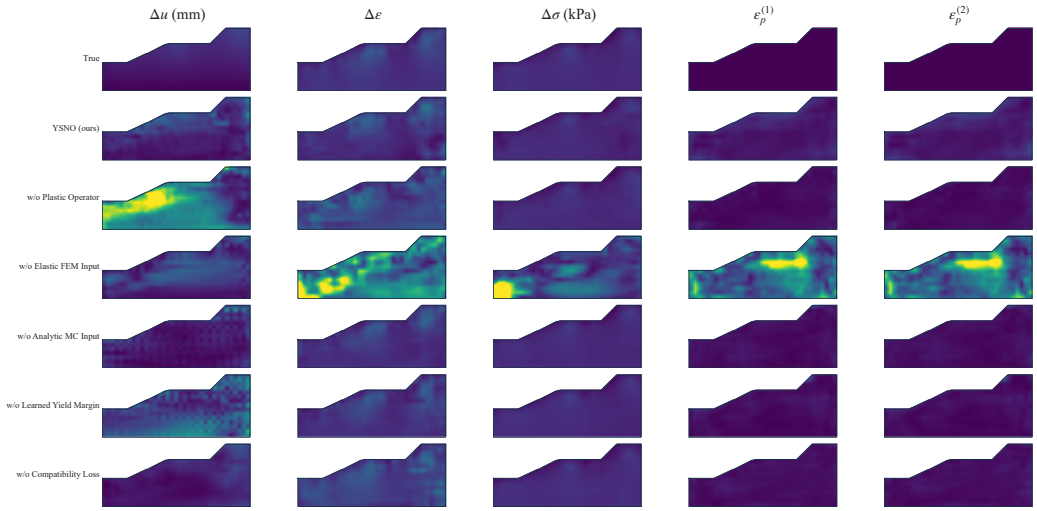

Figure 21: Ablation comparison for artificial profile 057, variant $v0001$. The complete YSNO prediction is followed by the five component-removal variants under the same field ordering.

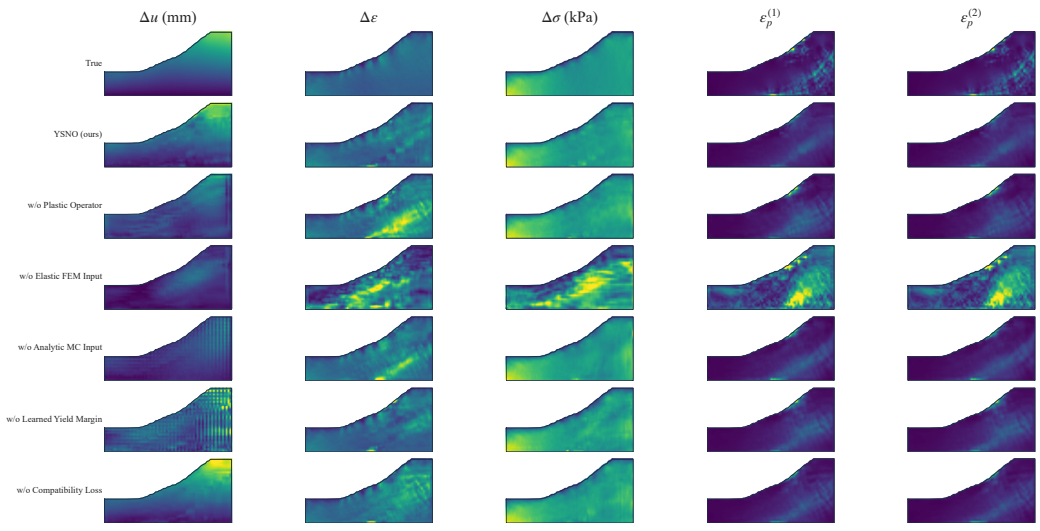

Figure 22: Ablation comparison for natural profile 072, variant $v0001$. The visualization provides a geometry-diverse qualitative check of the component-level effects reported quantitatively in Table 3.