ENTROMESH: MECHANICS-CONDITIONED STRUCTURAL ENTROPY FOR MULTI-SOLID STAMPING

Anonymous authors
Paper under double-blind review

ABSTRACT

Learned simulation of multi-solid stamping requires long-range reasoning over irregular finite-element meshes, where load, contact, material transitions, and localized plasticity govern inter-element couplings. Hierarchical mesh simulators reduce graph diameters, yet their coarse-level meshes are commonly derived from topology, geometry, algebraic connectivity, or standalone learned scores. These static coarsening schemes ignore instantaneous, state-dependent mechanical couplings and introduce substantial prediction error for evolving physical states. We introduce EntroMesh, a mechanics-conditioned structural-entropy hierarchy for multi-solid mesh prediction. EntroMesh converts each finite-element state into a dimensionless weighted graph, strengthens edges that carry stiffness, load, boundary, and strain-energy evidence, weakens edges across large stress jumps, and constructs a fixed-budget coding tree for a coarse-to-fine V-cycle graph simulator. Our hierarchy is physically interpretable, state-causal, and enables fair comparison under identical node budgets. Evaluated on the two large-scale long-time Unisoma stamping benchmarks against the state-of-the-art Unisoma model, EntroMesh reduces the sum of six-field relative-L_2 prediction error from 65.08 to 62.62 on BILATERALSTAMPING and from 64.19 to 53.65 on UNILATERALSTAMPING. Experiments demonstrate that mechanics-aware structural entropy is an effective hierarchical prior for large-scale multi-solid learned stamping simulation.

1 INTRODUCTION

Nonlinear solid mechanics plays a central role in modeling complex material and structural responses under large deformation, contact, and inelastic behavior. In virtual manufacturing, robotic manipulation, surgical planning, and structural design, one often needs to predict how solids deform, transmit loads, concentrate stress, and enter plastic regimes as geometries, materials, contacts, loads, and boundary conditions vary. Finite-element methods (FEMs) provide the standard high-fidelity discretization for these problems, but repeated nonlinear solves are costly when a design loop or data-driven study must evaluate many cases (Larson & Bengzon, 2013). This cost has made learned surrogate solvers attractive. Neural operators learn mappings between function spaces, with Fourier Neural Operators and DeepONet providing influential formulations for parametric PDE families (Li et al., 2021; Lu et al., 2021). Graph Network-based Simulators and MeshGraphNets instead express physical dynamics as relational computation on graph discretizations, making them natural for irregular finite-element meshes and multi-object systems whose contacts and connectivities are already graph-structured (Sanchez-Gonzalez et al., 2020; Pfaff et al., 2021). Prescribed multiscale meshes and breadth-first coarsening shorten propagation paths by fixing or traversing a mesh hierarchy (Fortunato et al., 2022; Cao et al., 2023); contact-aware attention and adaptive mesh refinement reshape the hierarchy around interactions or evolving physical features (Yu et al., 2024; Perera & Agrawal, 2024); learned hierarchy evolution and AMG-based diffusion refinement further make coarse propagation depend on data or operator structure (Deng et al., 2025; Würth et al., 2025).

A useful learned simulator for nonlinear solids has to satisfy a demanding mix of requirements, and the hardest one is not local message design alone but long-range mechanical communication. Boundary constraints, load paths, material transitions, and plastic zones can influence distant mesh regions, while sharp stress fronts must remain localized. Flat message-passing networks carry such signals through local hops, so increasing communication range by adding layers raises cost and can blur high-gradient responses. Hierarchical graph simulators reduce this burden by adding coarse

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.

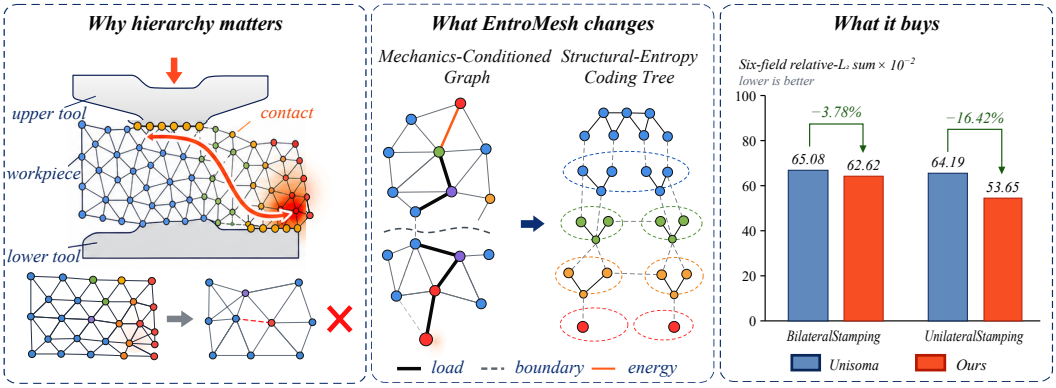


Figure 1: Overview of mechanics-conditioned structural entropy in EntroMesh and its benchmark gains.

routes, but mesh compression still determines which fine-scale relations remain coupled. Existing hierarchies often privilege geometric proximity, topological adjacency, algebraic connectivity, or learned scores; pure topology and geometry can miss state-dependent mechanical roles, while algebraic multigrid principles and low-energy-mode preservation do not directly yield an interpretable, state-conditioned hierarchy objective for fixed-budget neural comparison (Xu & Zikatanov, 2017). This gap is most visible in long-time multi-solid stamping, where metal and rubber responses are jointly shaped by tools, contacts, loads, and localized plastic strain.

EntroMesh addresses this compression problem by constructing the hierarchy from a dimensionless mechanics graph rather than from raw mesh adjacency alone, as summarized in Figure 1. Each finite-element state is converted into a weighted graph whose edges combine geometric proximity with stiffness, loads, boundary evidence, strain-energy proxies, and stress consistency. We then minimize structural entropy on this mechanics graph and use the resulting coding tree as the hierarchy for a fixed-budget V-cycle graph simulator. Structural entropy is useful here because it measures how efficiently a coding tree describes random-walk uncertainty on a weighted graph (Li & Pan, 2016); after mechanics conditioning, the tree favors edges that carry load, boundary, energy, or stress-consistency evidence while allowing separations across sharp stress changes. The paper makes three contributions: 1) We formulate a mechanics-conditioned structural-entropy hierarchy for multi-solid mesh prediction, turning current-state physical evidence into a dimensionless weighted graph before entropy-guided coarsening. 2) We integrate the hierarchy into a fixed-budget V-cycle simulator, enabling controlled comparisons between hierarchy rules without changing neural capacity or coarse-node budgets. 3) We evaluate the approach on the two largest Unisoma long-time stamping benchmarks, where EntroMesh reduces aggregate six-field error over the strongest published SOTA baseline by 3.78% on BILATERALSTAMPING and 16.42% on UNILATERALSTAMPING.

2 Related Work

Learned PDE and mesh simulators. Physics-informed neural networks impose differential-equation residuals during training (Raissi et al., 2019), while neural operators such as FNO and DeepONet learn solution operators from paired simulations (Li et al., 2021; Lu et al., 2021). Geometry-aware and graph/operator hybrids extend operator learning beyond regular grids and simple domains (Li et al., 2023c;d; Anandkumar et al., 2019; Li et al., 2020; Wu et al., 2023; Cao, 2021; Li et al., 2023b;a; Tran et al., 2023; Xiao et al., 2024; Wu et al., 2024; Serrano et al., 2023). Recent mechanics-oriented neural operators further connect learned solvers to finite-element, solid-mechanics, and large-scale 3D CAE settings (Yin et al., 2022; Kaewnuratchadasorn et al., 2024; Zeng et al., 2026). Message-passing and refinement-based neural PDE solvers provide complementary routes for graph-like discretizations and long-rollout stability (Brandstetter et al., 2022; Lippe et al., 2023). Graph Network-based Simulators and MeshGraphNets keep finite-element connectivity explicit, making them a strong fit for unstructured domains with variable discretizations (Sanchez-Gonzalez et al., 2020; Pfaff et al., 2021). Later mesh simulators incorporate physical in-

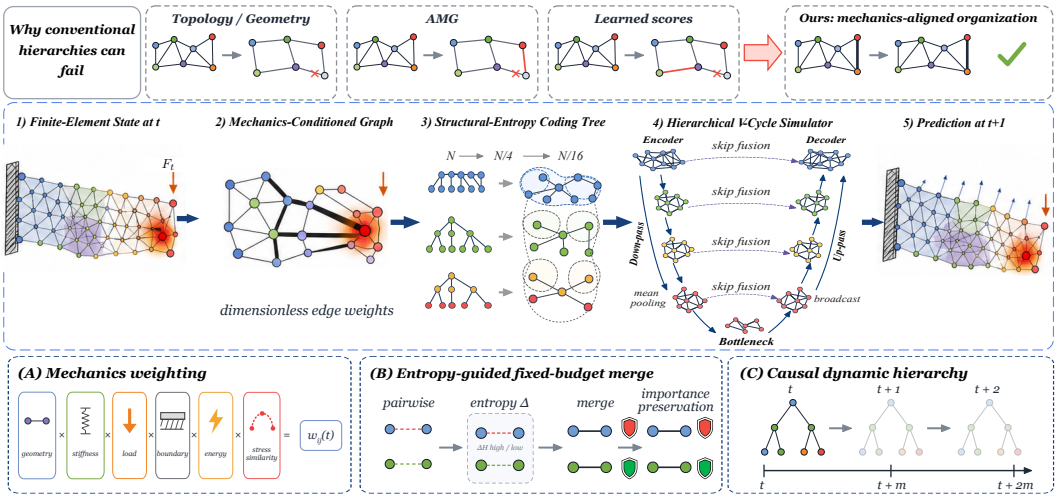


Figure 2: Overview of EntroMesh. A finite-element state is converted into a mechanics-conditioned graph, compressed by structural entropy, and processed by a V-cycle simulator. Thick black edges mark high-weight mechanical couplings to preserve during coarsening, while faded or crossed links indicate weak or discontinuous relations that may be separated.

puts, finite-element losses, or domain-specific structure for garments and solids (Linkerhägner et al., 2023; Würth et al., 2024; Grigorev et al., 2023). EntroMesh follows this mesh-simulator setting, but its main design target is the hierarchy that controls long-range information flow.

Hierarchical graph simulation. Hierarchical graph networks reduce communication distance by introducing coarse levels. MultiScale MeshGraphNets, BSMS-GNN, and HCMT construct these levels from mesh hierarchies, breadth-first traversal, or contact-aware attention (Fortunato et al., 2022; Cao et al., 2023; Yu et al., 2024). Adaptive-mesh and learned-hierarchy approaches refine or evolve the graph according to predicted physical needs (Perera & Agrawal, 2024; Deng et al., 2025). ROBIN uses root-node AMG together with diffusion refinement for nonlinear solid mechanics (Würth et al., 2025). Together, these works show that hierarchy design can be a first-order factor in simulator accuracy. EntroMesh contributes a complementary route: a fixed-budget hierarchy with an interpretable objective conditioned on the current mechanical state.

Multi-solid simulation. Graph-based simulators have been widely used for learning dynamics of interacting rigid and deformable bodies (Sanchez-Gonzalez et al., 2020; Pfaff et al., 2021). Recent methods further improve contact modeling through learned constraints, face-level interactions, and hierarchical contact propagation (Rubanova et al., 2022; Allen et al., 2023; Yu et al., 2024). Other works explicitly target deformable contact and forming processes (Dubey et al., 2026; Zhao et al., 2026). Unisoma extends this direction to general multi-solid systems with object-aware representations and dedicated contact modeling (Tao et al., 2025). However, scalable hierarchy construction that preserves mechanically important inter-solid couplings remains less explored.

Structural entropy. Structural entropy quantifies the code length needed to describe a graph through a hierarchy (Li & Pan, 2016). SEP uses this principle for graph pooling (Wu et al., 2022), and multi-relational structural entropy extends the idea to heterogeneous relations (Cao et al., 2024). EntroMesh uses structural entropy for a different purpose: the coding tree becomes a mechanics-aware simulator hierarchy rather than a generic representation-learning pooling operator.

3 METHOD

EntroMesh separates hierarchy construction from neural message passing. Given a finite-element state, the method first constructs a mechanics-conditioned weighted graph, then greedily builds a structural-entropy coding tree at fixed node budgets, and finally runs a V-cycle simulator on the

resulting hierarchy. This separation is deliberate: the hierarchy has an explicit physical objective and can be compared against alternative hierarchy rules without changing the neural architecture.

3.1 Problem Formulation

Each case is represented as a finite-element mesh with node positions $\mathbf{x}_i \in \mathbb{R}^D$, element-derived undirected edges E, node features $\mathbf{a}_i^t$, current displacement $\mathbf{u}_i^t$, stress σ_i^t, loads $\mathbf{f}_i^t$, and boundary indicators $\mathbf{b}_i^t$. The simulator predicts the target physical fields specified by the benchmark protocol from the retained input state. For transition-style data, the same architecture predicts displacement and stress increments; for long-time stamping, it predicts the retained target state directly.

3.2 Mechanics-Conditioned Graph Construction

The hierarchy is built from a mechanics graph rather than raw mesh adjacency. We first canonicalize finite-element connectivity into a simple undirected graph. For triangles, quadrilaterals, tetrahedra, and hexahedra, each cell contributes its standard element edges; duplicate undirected edges are merged. Each edge receives a positive weight w_{ij} that combines geometric proximity with current-state mechanical evidence. Let $\ell_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2$, and let $\bar{\ell}$ be the median positive edge length in the current mesh. The geometry term is

$$g_{ij} = \frac{\bar{\ell}}{\max(\ell_{ij}, \epsilon_\ell)}, \tag{1}$$

renormalized to unit positive median. Loads, boundary indicators, stress magnitudes, and energy proxies are robustly normalized by their 95th percentile and clipped to $[0, 1]$, using ϵ_z in each percentile denominator. If element or nodal strain energy is not available, we use the causal proxy $e_i = |\sigma_i^t| \, \|\mathbf{u}_i^t\|_2$. We denote the resulting normalized load, boundary, energy, and stress scores by q_i^f, q_i^b, q_i^e, and q_i^σ. The mechanics amplification for edge (i, j) is

$$a_{ij} = 1 + c_K K_{ij} + c_f \frac{q_i^f + q_j^f}{2} + c_b \frac{q_i^b + q_j^b}{2} + c_e \frac{q_i^e + q_j^e}{2}, \tag{2}$$

where K_{ij} is an optional stiffness weight and is zero when stiffness is unavailable. Large stress jumps are treated as candidate partition boundaries through

$$s_{ij} = \exp\left(-c_\sigma \frac{\big||\sigma_i^t| - |\sigma_j^t|\big|}{\max(Q_{0.95}(|\sigma^t|), \epsilon_\sigma)} \right). \tag{3}$$

The final edge weight is

$$w_{ij} = \frac{g_{ij} a_{ij} s_{ij}}{\frac{1}{|E|} \sum_{(p,q)\in E} g_{pq} a_{pq} s_{pq}}, \tag{4}$$

clipped below by ϵ_w. In the implementation, $\epsilon_\ell = \epsilon_z = \epsilon_\sigma = \epsilon_w = 10^{-6}$; the separate symbols identify the geometric, percentile-normalization, stress, and final-weight safeguards in the derivation. This normalization makes the weighting dimensionless and largely invariant to uniform coordinate and unit scaling. Large weights encourage the coding tree to keep load-bearing, boundary-relevant, and energy-concentrated neighborhoods together; the stress-consistency term lowers weights across sharp stress changes so that the hierarchy may place module boundaries at likely physical discontinuities. The node importance used during agglomeration is

$$r_i = \mathrm{clip}_{[0,1]}\Big(c_f q_i^f + c_b q_i^b + c_e q_i^e + c_\sigma q_i^\sigma \Big). \tag{5}$$

All quantities are computed from current or earlier fields, so hierarchy construction remains causal and does not use target-state information.

Proposition 1 (Causality and scale invariance)**.** *Consider the mechanics graph in Eq. (4) on a finite mesh with at least one edge. Assume nonnegative coefficients $c_K, c_f, c_b, c_e, c_\sigma$, a nonnegative dimensionless stiffness term K_{ij}, positive robust normalizers for the physical evidence fields, and non-degenerate numerical floors so that the $\epsilon_\ell, \epsilon_z, \epsilon_\sigma, \epsilon_w$ safeguards are inactive under the rescalings considered. Then the constructed edge weights are positive, dimensionless, and causal with respect to the current finite-element state. Moreover, they are invariant to uniform coordinate rescaling and to positive unit rescaling of the normalized load, stress, and energy evidence fields. The node-importance scores r_i are also causal, dimensionless, and invariant to the same positive unit rescalings.*

Proposition 1 formalizes the intended role of the mechanics graph: it changes with the physical state, but not with arbitrary mesh units or coordinate scale. The proof is given in Appendix A.

3.3 Structural-Entropy Coding Tree

For a weighted graph $G = (V, E, w)$, let $d_i = \sum_j w_{ij}$ and $\mathrm{vol}(G) = \sum_i d_i$. The one-dimensional structural entropy is

$$H^1(G) = -\sum_{i \in V} \frac{d_i}{\mathrm{vol}(G)} \log \frac{d_i}{\mathrm{vol}(G)}. \tag{6}$$

Given a partition $\mathcal{P}$, the two-dimensional partition entropy contains a between-module term induced by cut weights and a within-module term induced by node degrees:

$$H^{\mathcal{P}}(G) = -\sum_{C \in \mathcal{P}} \frac{g_C}{\mathrm{vol}(G)} \log \frac{\mathrm{vol}(C)}{\mathrm{vol}(G)} - \sum_{C \in \mathcal{P}} \sum_{i \in C} \frac{d_i}{\mathrm{vol}(G)} \log \frac{d_i}{\mathrm{vol}(C)}, \tag{7}$$

where g_C is the total cut weight leaving module C. EntroMesh builds a nested coding tree by greedy agglomeration. Starting from singleton modules, adjacent modules are merged according to the entropy-change score plus an importance-preservation term:

$$\Delta(C_a, C_b) = \Delta H^{\mathcal{P}}(C_a, C_b) \; + \; \lambda \frac{\max(r_{C_a}, r_{C_b})\big(\mathrm{vol}(C_a) + \mathrm{vol}(C_b)\big)}{\mathrm{vol}(G)}. \tag{8}$$

Snapshots of this process define the hierarchy at fixed node budgets. The node budgets are external to the entropy objective, which allows the same construction rule to operate under any fixed communication budget. The tree is fixed for the forward pass of a retained state, which avoids target leakage while still adapting the hierarchy to current loads, constraints, and stress features.

3.4 Hierarchical V-Cycle Simulator

The neural model is an encoder–processor–decoder graph simulator with an explicit V-cycle. An MLP encodes fine-level node features into hidden states. At each level, a message-passing block forms symmetric directed messages on every undirected edge. The edge features are relative position, normalized distance, and log normalized mechanics weight:

$$\mathbf{e}_{ij} = \left[\frac{\mathbf{x}_j - \mathbf{x}_i}{\bar{\ell}}, \frac{\|\mathbf{x}_j - \mathbf{x}_i\|_2}{\bar{\ell}}, \log \frac{w_{ij}}{\bar{w}} \right]. \tag{9}$$

Messages are averaged at receivers, fused with node states through an MLP, and applied with residual layer normalization. During the downward pass, fine states are pooled to parent modules by mean aggregation. During the upward pass, coarse states are broadcast back to fine modules, fused with stored skip states, and refined by one message-passing block. A decoder predicts the normalized target field. The V-cycle therefore gives each fine node both local mesh messages and coarse contextual messages, while the skip fusion preserves the fine-level state needed for localized stress and PEEQ prediction.

The training objective is a field-balanced relative-L_2 loss over deformable objects:

$$\mathcal{L} = \mathcal{L}_{\mathrm{geom}} + \mathcal{L}_{\mathrm{stress}} + \mathcal{L}_{\mathrm{PEEQ}}, \tag{10}$$

where the three terms compare geometry, stress, and PEEQ targets over deformable-object nodes. This loss balances the physical fields used by the simulator without changing the mechanics-conditioned hierarchy construction.

4 Experiments

4.1 Experimental Setup

Implementation. All EntroMesh runs and timing measurements are conducted on NVIDIA GeForce RTX 3090. The simulator uses hidden width 128, four message-passing steps, two coarse-attention layers with eight heads, dropout 0, AdamW with cosine learning-rate decay from 3×10^{-4}

Table 1: Long-time multi-solid stamping results using relative L_2 error($\times 10^{-2}$); lower is better.

BILATERALSTAMPING						
Method	Geo.-M	Geo.-R	Stress-M	Stress-R	PEEQ-M	PEEQ-R
Geo-FNO	1.25	1.17	3.90	16.12	42.69	47.31
GINO	2.99	2.43	5.43	26.21	55.23	64.09
GNO	2.85	2.81	4.63	25.24	58.30	59.09
LSM	5.98	6.15	5.56	36.16	70.01	73.47
Galerkin	5.99	5.94	5.63	34.42	69.62	73.35
FactFormer	2.97	1.05	4.59	13.78	62.66	42.18
OFormer	3.34	2.68	4.95	31.78	55.51	51.57
ONO	3.01	2.75	4.64	32.91	51.76	52.31
Transolver	0.83	0.81	3.46	14.25	35.92	32.76
GraphSAGE	2.27	2.29	4.40	21.96	49.36	44.95
MGN	1.95	1.97	4.21	20.59	46.46	41.41
Unisoma	0.57	0.52	2.78	10.39	28.17	**22.65**
EntroMesh	**0.50**	**0.51**	**2.57**	**10.18**	**26.14**	22.71

UNILATERALSTAMPING						
Method	Geo.-M	Geo.-R	Stress-M	Stress-R	PEEQ-M	PEEQ-R
Geo-FNO	1.09	1.54	11.13	11.24	37.71	35.74
GINO	1.88	2.19	12.62	14.38	46.51	48.85
GNO	4.26	4.24	13.51	18.14	64.13	68.59
LSM	3.67	3.57	13.27	13.35	58.89	60.26
Galerkin	10.07	10.10	17.25	27.94	79.86	85.55
FactFormer	1.78	1.66	10.95	10.35	38.86	30.22
OFormer	1.19	0.98	11.03	9.81	32.69	26.75
ONO	4.33	4.37	13.77	13.11	51.40	47.48
Transolver	0.79	0.75	8.39	8.62	29.38	26.93
GraphSAGE	2.75	2.60	10.59	11.89	54.41	51.06
MGN	3.91	3.83	12.52	14.17	58.05	56.94
Unisoma	0.71	0.63	8.07	7.73	25.91	21.14
EntroMesh	**0.57**	**0.51**	**7.56**	**6.80**	**20.75**	**17.47**

to 3×10^{-5}, weight decay 10^{-5}, and gradient clipping at 1. Hierarchy snapshots use level ratios 1, 0.25, and 0.0625; random-hierarchy and position-attention controls use the same budgets. EntroMesh stamping rows are evaluated in physical space after inverse standardization, with displacement, stress, and PEEQ metrics computed under the benchmark field definitions. Training uses 100 epochs with global batch size 1. Evaluation follows the source benchmark: relative L_2 error is reported separately for metal and rubber geometry, stress, and PEEQ, and all table values are shown in 10^{-2} units.

Datasets. We evaluate EntroMesh on the two largest Unisoma long-time stamping datasets. BILATERALSTAMPING contains two deformable solids, two rigid solids, two loads, five contacts, and 13,714 discretized points on average. UNILATERALSTAMPING contains two deformable solids, seventeen rigid solids, sixteen loads, eighteen contacts, and 49,386 points on average (Tao et al., 2025). For both tasks, the goal is to predict the final high-load state from the retained input state. The dataset is split into training, validation, and test sets with proportions of 83.33%, 8.33%, and 8.33%, respectively.

Baselines. We use Geo-FNO(Li et al., 2023c), GINO(Li et al., 2023d), GNO(Anandkumar et al., 2019), LSM(Wu et al., 2023), Galerkin Transformer(Cao, 2021), FactFormer(Li et al., 2023b), OFormer(Li et al., 2023a), ONO(Xiao et al., 2024), Transolver(Wu et al., 2024), GraphSAGE(Hamilton et al., 2017), MGN(Pfaff et al., 2021), and Unisoma(Tao et al., 2025) as baselines.

4.2 MAIN RESULTS

Under the published Unisoma field metrics, this section analyzes the effect of the mechanics-conditioned hierarchy on long-time multi-solid prediction. Table 1 shows that the main advantage is not a uniform rescaling of all errors, but a field-dependent improvement concentrated on quantities that require long-range mechanical communication. On the larger and more contact-rich UNILATERALSTAMPING task, EntroMesh improves all six fields and reduces the aggregate six-field sum from 64.19 to 53.65, a 16.42% reduction over the strongest published baseline. Most of this gain comes from the inelastic response channels: metal PEEQ decreases by 5.16 and rubber PEEQ by 3.67, which together account for approximately 8.83 of the 10.54 aggregate reduction. The two stress channels also improve by 0.51 and 0.93, while geometry gains are smaller in absolute value because the strongest baseline is already below 0.8 on both geometry fields. This pattern is consistent with the intended role of the hierarchy: the coarse graph should preserve load-bearing and stress-consistent couplings, where local message passing alone is most likely to lose information.

On BILATERALSTAMPING, the results provide a complementary view of the same mechanism. EntroMesh reduces the aggregate sum from 65.08 to 62.62 and improves five fields: geometry and stress for both materials, together with metal PEEQ. The gains on the two stress channels (2.78 to 2.57 and 10.39 to 10.18) and on metal PEEQ (28.17 to 26.14) indicate that mechanics-conditioned coarsening does more than reduce geometric discrepancy; it also improves state variables tied to load transfer, stress propagation, and plastic accumulation. Rubber PEEQ is effectively tied, with Unisoma lower by 0.06 (22.65 versus 22.71). Thus, the bilateral gain mainly comes from geometry, stress, and metal PEEQ rather than a uniform decrease in all plastic quantities. Together with the all-field improvement on UNILATERALSTAMPING, this result indicates that incorporating stiffness, load, boundary, strain-energy, and stress-consistency evidence into the structural-entropy objective biases the coarse graph toward the mechanical couplings that matter during stamping.

Additional rubber-PEEQ and difficult-case diagnostics are provided in Appendix B.1, while field-wise error distributions and spatial prediction examples are provided in Appendix D.

4.3 HIERARCHY ABLATION

Table 2 compares hierarchy rules under the same architecture and node budgets. On BILATERALSTAMPING, EntroMesh reaches 65.21 ± 0.90, compared with 79.31 ± 0.87 for random coarsening and 124.61 ± 1.31 for position attention, reducing error by 17.78% and 47.67%, respectively. On UNILATERALSTAMPING, EntroMesh reaches 58.00 ± 1.28, compared with 59.57 ± 0.95 for random coarsening and 57.84 ± 0.60 for position attention. The hierarchy-rule advantage is strongest on the bilateral task, while the unilateral result remains close to position attention.

Statistical significance. The bilateral advantage is consistent at the case level. Across 100 validation cases, EntroMesh improves the six-field error over random hierarchy by 12.81 on average, with a paired bootstrap 95% confidence interval of $[11.22, 14.46]$, and performs better on 94% of cases. Against position attention, the mean improvement is 56.07, with a 95% confidence interval of $[48.81, 63.50]$, and EntroMesh performs better on all cases. Holm-corrected paired Wilcoxon tests confirm significant improvements in both comparisons. Figure 3 further reports the field-wise paired results.

Table 2: Fixed-budget hierarchy ablation. Values are mean $\pm$ standard deviation of the six-field relative-L_2 error in 10^{-2} units. Lower is better.

Hierarchy	Bilateral	Unilateral
Position attention	124.61 ± 1.31	$\mathbf{57.84} \pm 0.60$
Random hierarchy	79.31 ± 0.87	59.57 ± 0.95
EntroMesh	$\mathbf{65.21} \pm 0.90$	58.00 ± 1.28

Parameter sensitivity. Figure 4 examines one hierarchy parameter at a time while keeping the validation cases, node budgets, and neural simulator fixed. The coarse-node budget is the dominant control in both tasks. Increasing coarse tokens per type from 16 to 64 increases bilateral hotspot purity from 0.372 to 0.526 and the high-gradient boundary rate from 0.088 to 0.111. The remaining mechanics coefficients exhibit comparatively small variations, indicating that the default weighting operates in a locally stable regime.

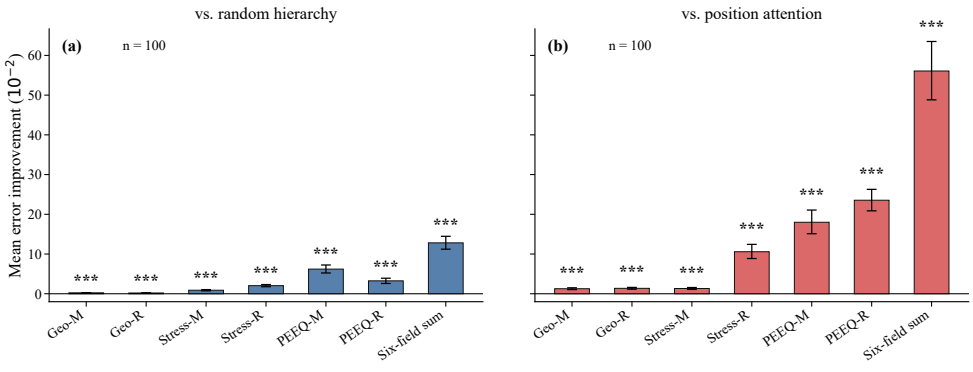


Figure 3: Bilateral validation significance analysis. Bars show mean baseline-minus-EntroMesh improvement in 10^{-2} units. Error bars denote paired bootstrap 95% confidence intervals, and $***$ indicates Holm-corrected $p < 10^{-3}$.

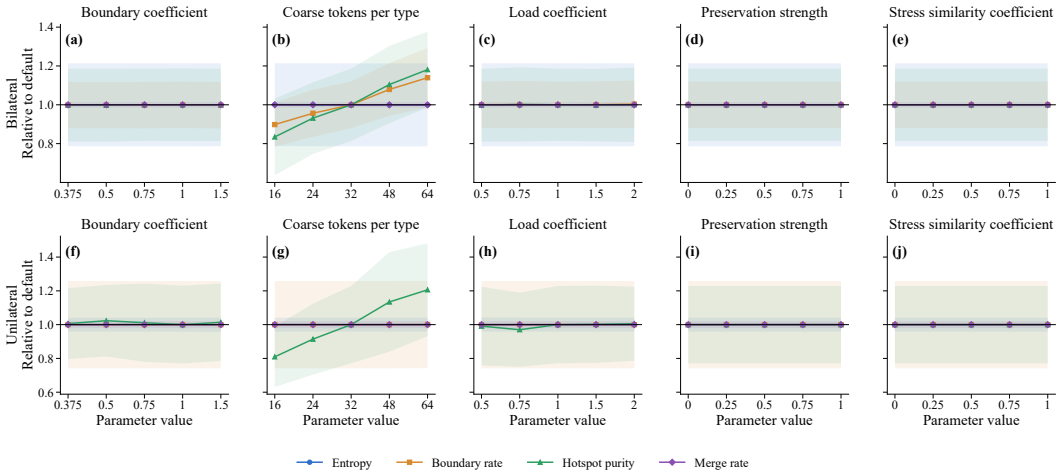


Figure 4: Parameter sensitivity on the dataset. Curves are normalized by the default setting, and shaded bands show one standard deviation.

Mechanics-conditioned hierarchy. Figure 5 follows tool and constraint information through a bilateral validation case. The input signal is concentrated around tools in panel (a), strengthening associated graph edges in panel (b). Structural-entropy coarsening uses the weighted graph to form coarse modules in panel (c), while panel (d) records the module assignments of high target-stress-gradient edges. Initial stress is zero, so the visible modulation reflects tool loads and constraints. Figure 6 in Appendix B.2 extends this analysis to the multi-tool unilateral case.

4.4 Efficiency

On NVIDIA GeForce RTX 3090, end-to-end inference including hierarchy construction requires 1.07 s per case and 183 MiB of GPU memory on BilateralStamping, and 5.95 s and 511 MiB on UnilateralStamping. The unilateral workload contains 3.60× more points on average, with 5.54× higher latency and 2.80× higher memory usage. Throughput decreases from 0.93 to 0.17 cases per second. Since the two tasks also differ in their numbers of tools and contacts, these measurements characterize end-to-end workload cost rather than controlled mesh-size scaling.

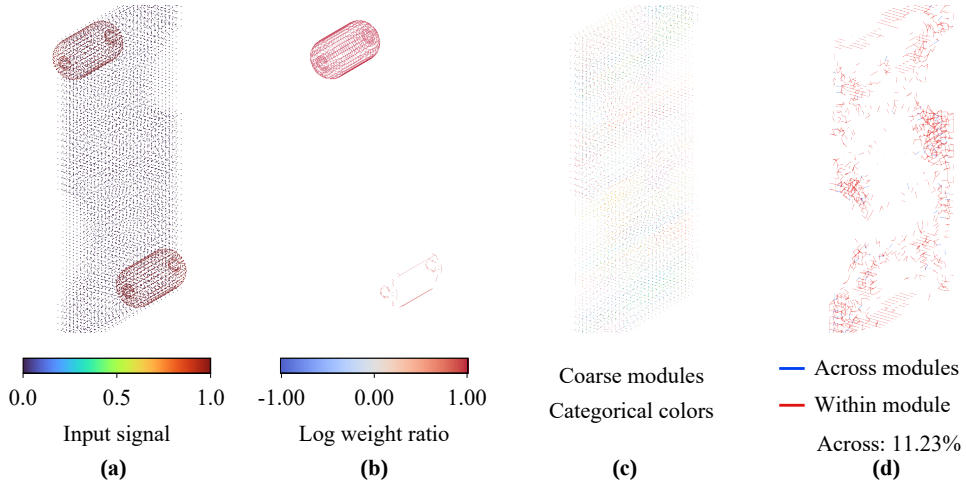


Figure 5: Tool- and constraint-conditioned hierarchy on BILATERALSTAMPING. (a) Normalized input signal. (b) Log mechanics-to-geometry edge-weight ratio for strongly modulated edges. (c) EntroMesh coarse modules. (d) High target-stress-gradient edges crossing modules (blue) or remaining within a module (red). Target stress is used only for post-hoc diagnosis.

5 DISCUSSION

EntroMesh constructs a mechanics-weighted graph and fixed-budget structural-entropy hierarchy for V-cycle simulation, reducing six-field prediction error on both stamping benchmarks, especially UNILATERALSTAMPING. Unlike geometry-, traversal-, or score-based hierarchies (Fortunato et al., 2022; Cao et al., 2023; Yu et al., 2024; Würth et al., 2025), EntroMesh derives coarse structure from current stiffness, load, boundary, strain-energy, and stress-consistency evidence. The gains are not uniform: rubber PEEQ on BILATERALSTAMPING remains slightly worse than Unisoma, and position attention is marginally better on UNILATERALSTAMPING, indicating that rubber-contact plasticity remains challenging.

EntroMesh also gives structural entropy a mechanics-specific role by using mechanical evidence to define transition weights and compress state-dependent communication (Li & Pan, 2016; Wu et al., 2022; Cao et al., 2024). The current evaluation is limited to two stamping protocols and discrete proxy-based coarsening. Future work will explore contact-aware weights, richer finite-element operators, broader benchmarks, and adaptive hierarchy updates.

6 CONCLUSION

We introduced EntroMesh, a mechanics-conditioned structural-entropy hierarchy for multi-solid stamping prediction. It converts current finite-element states into dimensionless physical edge weights, constructs a fixed-budget coding tree, and uses the hierarchy in a V-cycle graph simulator. On the two largest Unisoma stamping benchmarks, EntroMesh reduces the six-field error sum relative to the strongest published SOTA baseline by 3.78% on BILATERALSTAMPING and 16.42% on UNILATERALSTAMPING. The result gives structural entropy a concrete role in mechanics simulation: preserving state-dependent coupling during graph compression.

Ethics Statement

This work uses numerical surrogate data for solid mechanics and does not involve human subjects or personally identifiable data; predictions should still be independently verified before safety-critical engineering use.

Reproducibility Statement

We release an anonymized GitHub repository with code and scripts for end-to-end reproduction of all experiments and parameters. The link is available at: https://anonymous.4open.science/r/EntroMesh-13F5

AI Use Statement

ChatGPT was used only for language polishing, including sentence rephrasing, grammar checking, readability improvement, and flow refinement. It was not involved in idea generation, methodology, experimental design, data generation, result calculation, hyperparameter selection, or scientific analysis. All concepts, experiments, analyses, and conclusions were developed, conducted, and verified by the authors, who take full responsibility for the manuscript.

References

Kelsey R. Allen, Yulia Rubanova, Tatiana Lopez-Guevara, William Whitney, Alvaro Sanchez-Gonzalez, Peter W. Battaglia, and Tobias Pfaff. Learning rigid dynamics with face interaction graph networks. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=naBI4kpD6D.

Anima Anandkumar, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Nikola Kovachki, Zongyi Li, Burigede Liu, and Andrew Stuart. Neural operator: Graph kernel network for partial differential equations. In *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, 2019. URL https://openreview.net/forum?id=fg2ZFmXFO3.

Johannes Brandstetter, Daniel E. Worrall, and Max Welling. Message passing neural PDE solvers. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25–29, 2022*. OpenReview.net, 2022. URL https://openreview.net/forum?id=vSix3HPYKSU.

Shuhao Cao. Choose a transformer: Fourier or galerkin. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6–14, 2021, Virtual*, pp. 24924–24940, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/d0921d442ee91b896ad95059d13df618-Abstract.html.

Yadi Cao, Menglei Chai, Minchen Li, and Chenfanfu Jiang. Efficient learning of mesh-based physical simulation with bi-stride multi-scale graph neural network. In *Proceedings of the 40th International Conference on Machine Learning, ICML 2023, 23–29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 3541–3558. PMLR, 2023. URL https://proceedings.mlr.press/v202/cao23a.html.

Yuwei Cao, Hao Peng, Angsheng Li, Chenyu You, Zhifeng Hao, and Philip S. Yu. Multi-relational structural entropy. In *Proceedings of the Fortieth Conference on Uncertainty in Artificial Intelligence, UAI 2024, Barcelona, Spain, July 15–19, 2024*, volume 244 of *Proceedings of Machine Learning Research*, pp. 532–546. PMLR, 2024. URL https://proceedings.mlr.press/v244/cao24a.html.

Huayu Deng, Xiangming Zhu, Yunbo Wang, and Xiaokang Yang. EvoMesh: Adaptive physical simulation with hierarchical graph evolutions. In *Proceedings of the 42nd International Conference on Machine Learning, ICML 2025, Vancouver, Canada, July 13–19, 2025*, volume 267 of *Proceedings of Machine Learning Research*, pp. 13327–13346. PMLR, 2025. URL https://proceedings.mlr.press/v267/deng25j.html.

Vijay K. Dubey, Collin E. Haese, Osman Gültekin, David Dalton, Manuel K. Rausch, and Jan Fuhg. Graph neural network surrogates for contacting deformable bodies with necessary and sufficient contact detection. *Computer Methods in Applied Mechanics and Engineering*, 448: 118413, 2026. doi: 10.1016/j.cma.2025.118413. URL https://doi.org/10.1016/j.cma.2025.118413.

Meire Fortunato, Tobias Pfaff, Peter Wirnsberger, Alexander Pritzel, and Peter Battaglia. MultiScale MeshGraphNets. In *ICML 2022 2nd AI for Science Workshop*, 2022. URL https://openreview.net/forum?id=G3TRIsmMhhf.

Artur Grigorev, Michael J. Black, and Otmar Hilliges. HOOD: Hierarchical graphs for generalized modelling of clothing dynamics. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2023, Vancouver, BC, Canada, June 17–24, 2023*, pp. 16965–16974. IEEE, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Grigorev_HOOD_Hierarchical_Graphs_for_Generalized_Modelling_of_Clothing_Dynamics_CVPR_2023_paper.html.

William L. Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4–9, 2017, Long Beach, CA, USA*, pp. 1024–1034, 2017. URL https://proceedings.neurips.cc/paper/2017/hash/5dd9db5e033da9c6fb5ba83c7a7ebea9-Abstract.html.

Chawit Kaewnuratchadasorn, Jiaji Wang, and Chul-Woo Kim. Physics-informed neural operator solver and super-resolution for solid mechanics. *Computer-Aided Civil and Infrastructure Engineering*, 39(22):3435–3451, 2024. doi: 10.1111/mice.13292. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/mice.13292.

Mats G. Larson and Fredrik Bengzon. *The Finite Element Method: Theory, Implementation, and Applications*, volume 10 of *Texts in Computational Science and Engineering*. Springer, Berlin, Heidelberg, 2013. doi: 10.1007/978-3-642-33287-6.

Angsheng Li and Yicheng Pan. Structural information and dynamical complexity of networks. *IEEE Transactions on Information Theory*, 62(6):3290–3339, June 2016. ISSN 1557-9654. doi: 10.1109/TIT.2016.2555904.

Zijie Li, Kazem Meidani, and Amir Barati Farimani. Transformer for partial differential equations' operator learning. *Transactions on Machine Learning Research*, 2023a. URL https://openreview.net/forum?id=EPPqt3uERT.

Zijie Li, Dule Shu, and Amir Barati Farimani. Scalable transformer for PDE surrogate modeling. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10–16, 2023*, 2023b. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/e21e4e625b130a7d701f1f2836f189ad-Abstract-Conference.html.

Zongyi Li, Nikola B. Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Andrew M. Stuart, Kaushik Bhattacharya, and Anima Anandkumar. Multipole graph neural operator for parametric partial differential equations. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6–12, 2020, Virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/4b21cf96d4cf612f239a6c322b10c8fe-Abstract.html.

Zongyi Li, Nikola Borislavov Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew M. Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3–7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=c8P9NQVtmnO.

Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. Fourier neural operator with learned deformations for PDEs on general geometries. *Journal of Machine Learning Research*, 24:388:1–388:26, 2023c. URL https://jmlr.org/papers/v24/23-0064.html.

Zongyi Li, Nikola B. Kovachki, Christopher B. Choy, Boyi Li, Jean Kossaifi, Shourya Prakash Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, and Animashree Anandkumar. Geometry-informed neural operator for large-scale 3D PDEs. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10–16, 2023*, 2023d. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/f0ba4a0bf853b6b913b6036ff54071a5-Abstract-Conference.html.

Jonas Linkerhägner, Niklas Freymuth, Paul Maria Scheikl, Franziska Mathis-Ullrich, and Gerhard Neumann. Grounding graph network simulators using physical sensor observations. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=jsZsEd8VEY.

Phillip Lippe, Bas Veeling, Paris Perdikaris, Richard E. Turner, and Johannes Brandstetter. PDE-Refiner: Achieving accurate long rollouts with neural PDE solvers. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10–16, 2023*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d529b943af3dba734f8a7d49efcb6d09-Abstract-Conference.html.

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.

Roberto Perera and Vinamra Agrawal. Multiscale graph neural networks with adaptive mesh refinement for accelerating mesh-based simulations. *Computer Methods in Applied Mechanics and Engineering*, 429:117152, 2024. ISSN 0045-7825. doi: 10.1016/j.cma.2024.117152. URL https://www.sciencedirect.com/science/article/pii/S0045782524004080.

Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter W. Battaglia. Learning mesh-based simulation with graph networks. In *International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3–7, 2021*. OpenReview.net, 2021. URL https://openreview.net/forum?id=roNqYL0_XP.

M. Raissi, P. Perdikaris, and G. E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019. ISSN 0021-9991. doi: 10.1016/j.jcp.2018.10.045. URL https://www.sciencedirect.com/science/article/pii/S0021999118307125.

Yulia Rubanova, Alvaro Sanchez-Gonzalez, Tobias Pfaff, and Peter W. Battaglia. Constraint-based graph network simulator. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18844–18870. PMLR, 2022. URL https://proceedings.mlr.press/v162/rubanova22a.html.

Alvaro Sanchez-Gonzalez, Jonathan Godwin, Tobias Pfaff, Rex Ying, Jure Leskovec, and Peter W. Battaglia. Learning to simulate complex physics with graph networks. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13–18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8459–8468. PMLR, 2020. URL http://proceedings.mlr.press/v119/sanchez-gonzalez20a.html.

Louis Serrano, Lise Le Boudec, Armand Kassaï Koupaï, Thomas X. Wang, Yuan Yin, Jean-Noël Vittaut, and Patrick Gallinari. Operator learning with neural fields: Tackling PDEs on general geometries. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10–16, 2023*, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/df54302388bbc145aacaa1a54a4a5933-Abstract-Conference.html.

Shilong Tao, Zhe Feng, Haonan Sun, Zhanxing Zhu, and Yunhuai Liu. Unisoma: A unified transformer-based solver for multi-solid systems. In *Proceedings of the 42nd International Conference on Machine Learning, ICML 2025, Vancouver, Canada, July 13–19, 2025*, volume

267 of *Proceedings of Machine Learning Research*, pp. 59132–59153. PMLR, 2025. URL https://proceedings.mlr.press/v267/tao25b.html.

Alasdair Tran, Alexander Patrick Mathews, Lexing Xie, and Cheng Soon Ong. Factorized fourier neural operators. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1–5, 2023*. OpenReview.net, 2023. URL https://openreview.net/forum?id=tmIiMPl4IPa.

Haixu Wu, Tengge Hu, Huakun Luo, Jianmin Wang, and Mingsheng Long. Solving high-dimensional PDEs with latent spectral models. In *Proceedings of the 40th International Conference on Machine Learning, ICML 2023, 23–29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pp. 37417–37438. PMLR, 2023. URL https://proceedings.mlr.press/v202/wu23r.html.

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, ICML 2024, Vienna, Austria, July 21–27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53681–53705. PMLR, 2024. URL https://proceedings.mlr.press/v235/wu24r.html.

Junran Wu, Xueyuan Chen, Ke Xu, and Shangzhe Li. Structural entropy guided graph hierarchical pooling. In *Proceedings of the 39th International Conference on Machine Learning, ICML 2022, 17–23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pp. 24017–24030. PMLR, 2022. URL https://proceedings.mlr.press/v162/wu22b.html.

Tobias Würth, Niklas Freymuth, Clemens Zimmerling, Gerhard Neumann, and Luise Kärger. Physics-informed meshgraphnets (PI-MGNs): Neural finite element solvers for non-stationary and nonlinear simulations on arbitrary meshes. *Computer Methods in Applied Mechanics and Engineering*, 429:117102, 2024. ISSN 0045-7825. doi: 10.1016/j.cma.2024.117102. URL https://www.sciencedirect.com/science/article/pii/S004578252400358X.

Tobias Würth, Niklas Freymuth, Gerhard Neumann, and Luise Kärger. Diffusion-based hierarchical graph neural networks for simulating nonlinear solid mechanics. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems, NeurIPS 2025*, volume 38, 2025. doi: 10.52202/085713-4336. URL https://papers.nips.cc/paper_files/paper/2025/hash/bcd3c132bb19544815c5e660a2b044bd-Abstract-Conference.html.

Zipeng Xiao, Zhongkai Hao, Bokai Lin, Zhijie Deng, and Hang Su. Improved operator learning by orthogonal attention. In *Proceedings of the 41st International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21–27, 2024*, volume 235 of *Proceedings of Machine Learning Research*, pp. 54288–54299. PMLR, 2024. URL https://proceedings.mlr.press/v235/xiao24c.html.

Jinchao Xu and Ludmil Zikatanov. Algebraic multigrid methods. *Acta Numerica*, 26:591–721, 2017. doi: 10.1017/S0962492917000083.

Minglang Yin, Enrui Zhang, Yue Yu, and George Em 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. ISSN 0045-7825. doi: 10.1016/j.cma.2022.115027. URL https://www.sciencedirect.com/science/article/pii/S0045782522002535. A Special Issue in Honor of the Lifetime Achievements of J. Tinsley Oden.

Youn-Yeol Yu, Jeongwhan Choi, Woojin Cho, Kookjin Lee, Nayong Kim, Kiseok Chang, ChangSeung Woo, Ilho Kim, SeokWoo Lee, Joon Young Yang, Sooyoung Yoon, and Noseong Park. Learning flexible body collision dynamics with hierarchical contact mesh transformer. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7–11, 2024*. OpenReview.net, 2024. URL https://openreview.net/forum?id=90yw2uM6J5.

Peijian Zeng, Guan Wang, Haohao Gu, Xiaoguang Hu, Tiezhu Gao, Zhuowei Wang, Aimin Yang, and Xiaoyu Song. LRQ-Solver: A transformer-based neural operator for fast and accurate solving of large-scale 3D PDEs. *Computer-Aided Design*, 200:104151, 2026. ISSN 0010-4485. doi: 10.1016/j.cad.2026.104151. URL https://www.sciencedirect.com/science/article/pii/S0010448526001211.

Yingxue Zhao, Qianyi Chen, Haoran Li, Haosu Zhou, Hamid Reza Attar, Tobias Pfaff, Tailin Wu, and Nan Li. Recurrent u-net-based graph neural network (RUGNN) for accurate deformation predictions in sheet material forming. *Advanced Engineering Informatics*, 69:104021, 2026. doi: 10.1016/j.aei.2025.104021. URL https://doi.org/10.1016/j.aei.2025.104021.

A Proof of Proposition 1

Proof. Let

$$\mathcal{F}_t = \left\{E, \mathbf{x}, K, \mathbf{u}^{\le t}, \sigma^{\le t}, \mathbf{f}^{\le t}, \mathbf{b}^{\le t}, \text{element evidence}^{\le t}\right\}$$

denote the information available at the retained state, with $\mathbf{y}_{\text{tar}} \notin \mathcal{F}_t$.

For a nonnegative evidence field $z = \{z_i\}_{i \in V}$, define

$$\mathcal{N}_\epsilon(z_i; z) = \text{clip}_{[0,1]}\left(\frac{z_i}{\max(Q_{0.95}(z), \epsilon_z)}\right), \qquad q_i^m = \mathcal{N}_\epsilon(z_i^m; z^m), \quad m \in \{f, b, e, \sigma\}.$$

For each edge $(i, j) \in E$, the quantities used by the mechanics graph are

$$\ell_{ij} = \|\mathbf{x}_i - \mathbf{x}_j\|_2, \quad \widetilde{\ell}_{ij} = \max(\ell_{ij}, \epsilon_\ell), \quad g_{ij} = \frac{\bar{\ell}}{\widetilde{\ell}_{ij}}, \tag{11}$$

$$\delta_{ij}^\sigma = \frac{\big||\sigma_i^t| - |\sigma_j^t|\big|}{\max(Q_{0.95}(|\sigma^t|), \epsilon_\sigma)}, \quad s_{ij} = \exp(-c_\sigma \delta_{ij}^\sigma), \tag{12}$$

$$a_{ij} = 1 + c_K K_{ij} + c_f \frac{q_i^f + q_j^f}{2} + c_b \frac{q_i^b + q_j^b}{2} + c_e \frac{q_i^e + q_j^e}{2}, \tag{13}$$

$$p_{ij} = g_{ij} a_{ij} s_{ij}, \quad \bar{p} = \frac{1}{|E|} \sum_{(r,s) \in E} p_{rs}, \tag{14}$$

$$w_{ij} = \max\left(\frac{p_{ij}}{\bar{p}}, \epsilon_w\right). \tag{15}$$

The node-importance score is

$$r_i = \text{clip}_{[0,1]}\left(c_f q_i^f + c_b q_i^b + c_e q_i^e + c_\sigma q_i^\sigma\right). \tag{16}$$

Causality. Every quantity in Eqs. (11)–(16) is computed exclusively from $\mathcal{F}_t$. Therefore, there exists a deterministic map Φ such that

$$\left(\{w_{ij}\}_{(i,j) \in E}, \{r_i\}_{i \in V}\right) = \Phi(\mathcal{F}_t).$$

In particular, neither the edge weights nor the node-importance scores depend on $\mathbf{y}_{\text{tar}}$. If the coding tree is produced by the deterministic fixed-budget rule

$$\mathcal{T} = \Psi(E, w, r, B),$$

then

$$\mathcal{T} = \Psi(E, \Phi(\mathcal{F}_t), B),$$

which is also independent of the target state. Hence the hierarchy construction is causal.

Positivity and boundedness. Since $\epsilon_\ell, \epsilon_z, \epsilon_\sigma, \epsilon_w > 0$, $c_K, c_f, c_b, c_e, c_\sigma \ge 0$, $K_{ij} \ge 0$, and $q_i^m \in [0, 1]$, we have

$$\widetilde{\ell}_{ij} > 0, \qquad g_{ij} > 0, \qquad a_{ij} \ge 1, \qquad 0 < s_{ij} \le 1.$$

Consequently,

$$p_{ij} > 0, \qquad \bar{p} > 0, \qquad w_{ij} > 0, \qquad 0 \le r_i \le 1.$$

Thus all quantities required by the weighted graph and coding-tree construction are well defined.

Dimensionlessness. Under the non-degenerate-floor assumption of Proposition 1, the robust normalizers are strictly larger than their corresponding numerical floors. Hence

$$[g_{ij}] = \frac{[\bar{\ell}]}{[\ell_{ij}]} = \frac{L}{L} = 1, \qquad [q_i^m] = \frac{[z_i^m]}{[Q_{0.95}(z^m)]} = 1.$$

By assumption, K_{ij} and the coefficients $c_K, c_f, c_b, c_e, c_\sigma$ are dimensionless. Moreover,

$$[\delta_{ij}^\sigma] = \frac{S}{S} = 1, \qquad [s_{ij}] = 1.$$

It follows that

$$[a_{ij}] = [p_{ij}] = [\bar{p}] = [w_{ij}] = [r_i] = 1.$$

Therefore, both the edge weights and node-importance scores are dimensionless.

Uniform coordinate rescaling. Consider a positive coordinate rescaling

$$\mathbf{x}_i' = \alpha \mathbf{x}_i, \qquad \alpha > 0.$$

Then

$$\ell_{ij}' = \alpha \ell_{ij}, \qquad \bar{\ell}' = \alpha \bar{\ell}.$$

When the length floor is inactive,

$$g_{ij}' = \frac{\bar{\ell}'}{\ell_{ij}'} = \frac{\alpha \bar{\ell}}{\alpha \ell_{ij}} = g_{ij}.$$

The normalized mechanical evidence and the dimensionless stiffness term remain unchanged, giving

$$q_i^{m\,\prime} = q_i^m, \qquad K_{ij}' = K_{ij}, \qquad a_{ij}' = a_{ij}, \qquad s_{ij}' = s_{ij}.$$

Therefore,

$$p_{ij}' = p_{ij}, \qquad \bar{p}' = \bar{p}, \qquad w_{ij}' = w_{ij}, \qquad r_i' = r_i.$$

Hence the mechanics graph is invariant to uniform coordinate rescaling.

Positive unit rescaling of mechanical evidence. Let a nonnegative evidence field be rescaled as

$$z' = \gamma z, \qquad \gamma > 0.$$

Since quantiles are positively homogeneous,

$$Q_{0.95}(\gamma z) = \gamma Q_{0.95}(z).$$

When the percentile floor is inactive,

$$q_i(\gamma z) = \mathrm{clip}_{[0,1]}\left(\frac{\gamma z_i}{Q_{0.95}(\gamma z)}\right) = \mathrm{clip}_{[0,1]}\left(\frac{\gamma z_i}{\gamma Q_{0.95}(z)}\right) = q_i(z). \tag{17}$$

The same argument applies to the stress-consistency term. Under $\sigma' = \gamma_\sigma \sigma$,

$$\delta_{ij}^{\sigma\,\prime} = \frac{\gamma_\sigma \big||\sigma_i^t| - |\sigma_j^t|\big|}{\gamma_\sigma Q_{0.95}(|\sigma^t|)} = \delta_{ij}^\sigma, \tag{18}$$

$$s_{ij}' = s_{ij}. \tag{19}$$

For the proxy strain-energy evidence

$$e_i = |\sigma_i^t| \|\mathbf{u}_i^t\|_2,$$

a positive change of stress and displacement units gives

$$\sigma_i^{t\,\prime} = \gamma_\sigma \sigma_i^t, \qquad \mathbf{u}_i^{t\,\prime} = \gamma_u \mathbf{u}_i^t.$$

Therefore,

$$e_i' = \gamma_\sigma \gamma_u e_i, \qquad Q_{0.95}(e') = \gamma_\sigma \gamma_u Q_{0.95}(e),$$

and Eq. (17) gives

$$q_i^{e\,\prime} = q_i^e.$$

Consequently,

$$a_{ij}' = a_{ij}, \qquad s_{ij}' = s_{ij}, \qquad r_i' = r_i,$$

which further implies

$$p_{ij}' = p_{ij}, \qquad \bar{p}' = \bar{p}, \qquad w_{ij}' = w_{ij}.$$

Thus the construction is invariant to positive unit rescaling of the normalized load, stress, and energy evidence.

Effect of numerical floors. The exact scale-invariance statements above require

$$\ell_{ij} > \epsilon_\ell, \qquad Q_{0.95}(z^m) > \epsilon_z, \qquad Q_{0.95}(|\sigma^t|) > \epsilon_\sigma$$

for the rescalings under consideration. If one of these safeguards becomes active, exact positive homogeneity need not hold. Nevertheless,

$$\widetilde{\ell}_{ij} > 0, \qquad q_i^m \in [0,1], \qquad a_{ij} \geq 1, \qquad 0 < s_{ij} \leq 1, \qquad w_{ij} > 0, \qquad r_i \in [0,1],$$

so the construction remains well defined and causal.

Combining the arguments above proves positivity, dimensionlessness, causality, and the stated coordinate- and unit-scale invariances under the assumptions of Proposition 1. □

B Additional Diagnostics

B.1 Rubber-PEEQ Variation and Difficult Cases

Table 3 complements Section 4.2 with rubber-PEEQ statistics from 100 bilateral validation cases. The mean and median are 22.07 and 21.31, whereas the 90th and 95th percentiles reach 30.01 and 32.07, and the maximum reaches 46.57, all in 10^{-2} units. This spread is visible in Figure 9. The published Unisoma test value 22.65 serves as a reference threshold, exceeded by 46 cases; these validation statistics are separate from the test-set comparison in Table 1.

Rubber-PEEQ error correlates with rubber stress ($r = 0.52$), metal PEEQ ($r = 0.51$), rubber geometry ($r = 0.48$), metal geometry ($r = 0.40$), and metal stress ($r = 0.37$). Its correlation with the six-field sum is $r = 0.90$, partly reflecting its inclusion in that sum. The field-to-field correlations indicate that difficult plasticity cases can also incur errors in other response fields, motivating the joint inspection of stress, PEEQ, and displacement in Figures 11 and 12.

Table 3: Bilateral rubber-PEEQ error diagnostic. Error values are reported in 10^{-2} units. The threshold is the published Unisoma rubber-PEEQ value; lower is better.

Threshold	Mean	Median	P90	P95	Max	Cases above threshold
22.65	22.07	21.31	30.01	32.07	46.57	46/100

Table 4 separates field-specific difficulty from aggregate error. Case 033 has the largest rubber-PEEQ error (46.57), but case 072 has the largest six-field sum among the listed cases (101.11) and a higher rubber-stress error (17.72 versus 7.86). Rubber geometry remains comparatively accurate in both cases (0.84 and 0.56). Figure 12 shows the spatial residuals for case 033, connecting the numerical ranking to the regions where prediction errors occur.

Table 4: Worst bilateral cases by rubber PEEQ. Error values are reported in 10^{-2} units. "Sum" is the six-field relative-L_2 error.

Case	Rubber PEEQ	Sum	Rubber stress	Rubber geometry
033	46.57	94.22	7.86	0.84
028	36.96	76.53	14.05	0.85
072	36.68	101.11	17.72	0.56
064	34.66	90.49	10.45	1.58
090	33.37	76.26	11.87	0.41

B.2 Tool- and Constraint-Conditioned Hierarchies

Figures 5 and 6 show how the available mechanical inputs enter EntroMesh hierarchy construction. The diagnostic covers twelve validation cases from each stamping dataset; the figures display one case per task. Initial stress is zero in these cases, and nonzero load and boundary signals occur on tool-related nodes rather than inside the deformable bodies. Panel a visualizes the maximum of the individually normalized current stress magnitude, force magnitude, boundary indicator, and displacement magnitude. The two-tool configuration produces localized input signals in the bilateral case, whereas the unilateral case distributes them across multiple tools.

Panel b displays $\log(w_{ij}^{\mathrm{mechanics}}/w_{ij}^{\mathrm{geometry}})$ on edges selected from the upper decile of absolute modulation. Positive values identify edges strengthened by the available mechanical information. Structural-entropy coarsening then operates on the resulting weighted graph, producing the coarse modules shown in panel c. These views make the construction sequence explicit: tool and constraint information enters the edge weights, and the weighted graph supplies the hierarchy used for coarse-level communication.

Panel d examines deformable-body edges in the upper decile of target-stress jumps. Blue edges join different coarse modules, while red edges have both endpoints within one module. The cross-module fractions are 11.23% for the displayed bilateral case and 9.25% for the unilateral case.

Most selected edges remain within a module; the panel records this partition property using target stress only after hierarchy construction. Input signals, weight modulation, and module membership therefore describe the construction mechanism, while prediction accuracy is evaluated separately in Table 1.

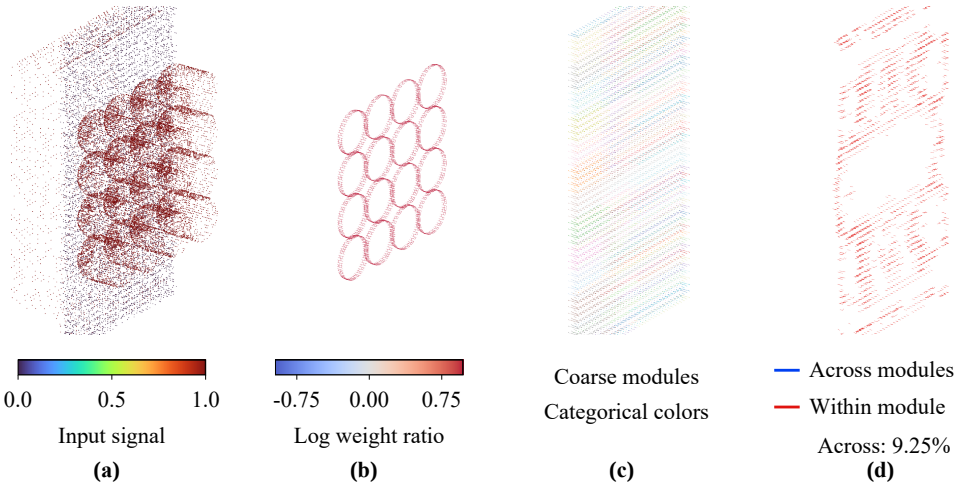


Figure 6: Tool- and constraint-conditioned hierarchy on UNILATERALSTAMPING. Panels show (a) normalized input signals, (b) edge-weight modulation, (c) EntroMesh coarse modules, and (d) high target-stress-gradient edges across modules (blue) or within a module (red), using the definitions in Figure 5.

B.3 INFERENCE RESOURCE PROFILE

Figure 7 details the end-to-end measurements in Section 4.4, including hierarchy construction. Latency, GPU memory, and throughput are measured over 20 validation cases per dataset, while panel a uses the aggregate test-set errors reported in Section 4.2. Unisoma error markers share the horizontal position of the corresponding EntroMesh point to display the within-task error reduction; they do not represent measured Unisoma latency.

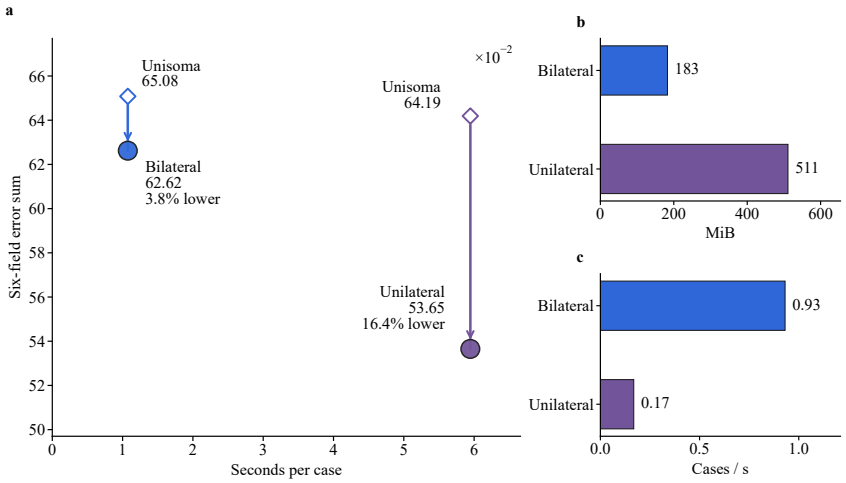


Figure 7: Efficiency profile on the stamping benchmarks. Panel a couples test-set error with inference latency; panels b and c report memory and throughput on RTX 3090.

C BASELINE REFERENCE SIGNIFICANCE

Across both tasks, all baselines yield a positive aggregate Δ, indicating lower validation error for EntroMesh under this reference calculation. The corresponding significance-test results are shown in Figure 8. Compared with the closest baseline, Unisoma, EntroMesh achieves aggregate improvements of 0.040340 on BILATERALSTAMPING and 0.102503 on UNILATERALSTAMPING. After Holm correction, statistically significant improvements are observed on five of seven metrics for BILATERALSTAMPING and six of seven metrics for UNILATERALSTAMPING. Together with the consistently positive aggregate and field-wise differences, these results provide statistical evidence that the improvements of EntroMesh are systematic rather than driven by a single response field.

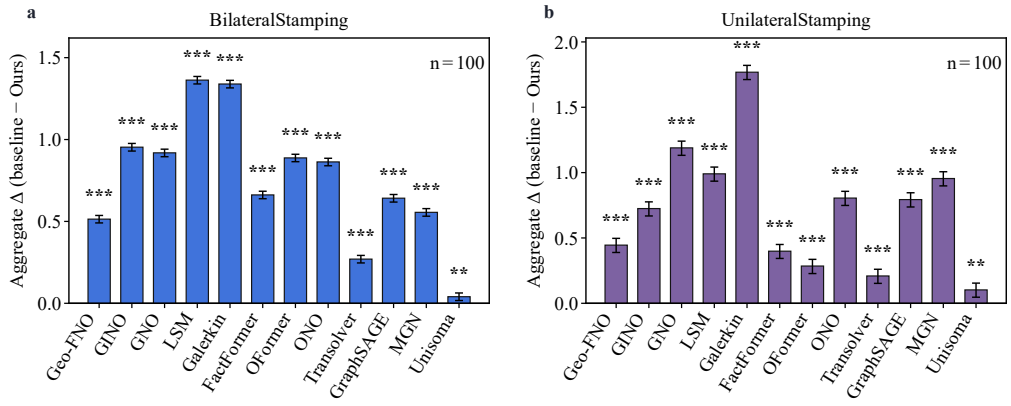


Figure 8: GraphPad-style reference significance audit. Bars show aggregate $\Delta = \mu_{\text{pub}} - \bar{e}_{EntroMesh}$, error bars show bootstrap 95% intervals, and stars denote Holm-corrected one-sided reference tests.

D FIELD-WISE ERRORS AND SPATIAL PREDICTIONS

D.1 ERROR DISTRIBUTIONS

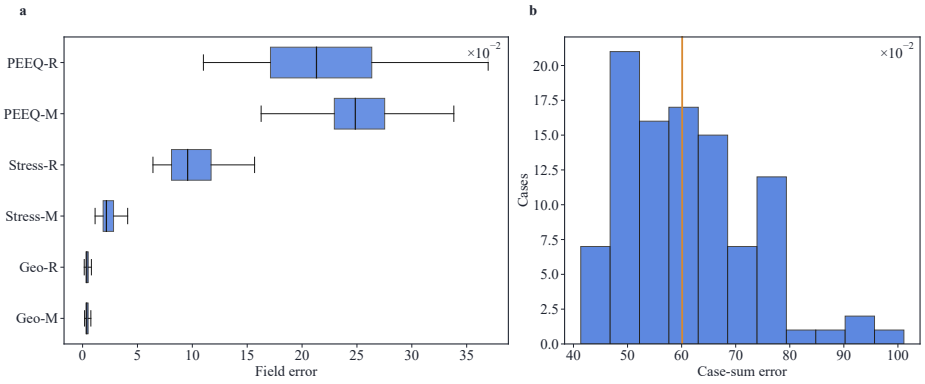


Figure 9: Field-wise box plots and per-case error sums for BILATERALSTAMPING.

Figure 9 separates the central error pattern from its long tail on BILATERALSTAMPING. The geometry fields remain tightly concentrated near zero, whereas the PEEQ fields account for most of the vertical spread, with medians of 24.85 for metal and 21.31 for rubber. The accompanying case-sum histogram has a median of 60.10 and a 90th percentile of 77.09, while the maximum reaches 101.11. Thus, the aggregate error is governed by a smaller set of difficult plasticity cases rather than by uniformly large errors across the validation set. This distribution-level view connects the summary in Table 1 to the rubber-PEEQ statistics in Table 3 and the spatial residuals in Figure 12.

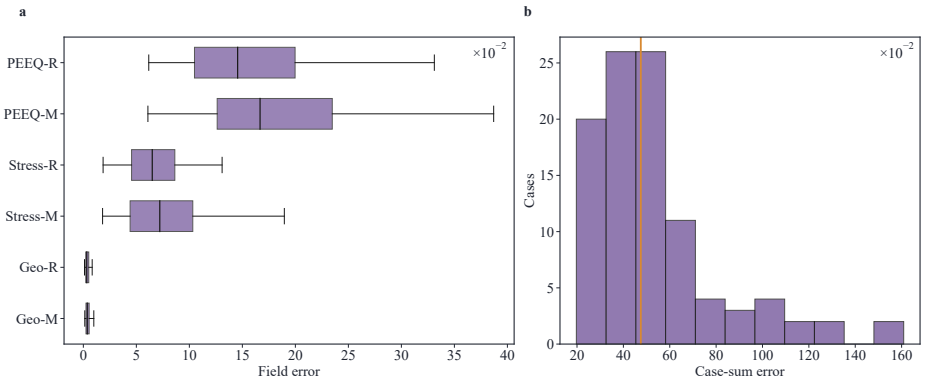


Figure 10: Field-wise box plots and per-case error sums for UNILATERALSTAMPING.

Figures 9 and 10 report the six field errors and their per-case sum over 100 validation cases per dataset, extending the field-wise analysis in Section 4.2. Boxes indicate the interquartile range, central lines indicate medians, and whiskers extend to the most extreme values within 1.5 interquartile ranges; outlier markers are omitted. The orange line in each histogram marks the median. Error values below use the same 10^{-2} units as the figures.

On BILATERALSTAMPING, both geometry medians are 0.38, while metal and rubber PEEQ medians are 24.85 and 21.31. The wider rubber-PEEQ distribution agrees with the quantiles in Table 3. On UNILATERALSTAMPING, geometry medians remain low (0.37 and 0.32), while metal and rubber PEEQ medians are 16.67 and 14.55. The unilateral six-field sum has a lower median than the bilateral sum (47.51 versus 60.10), but a higher 90th percentile (88.01 versus 77.09) and maximum (160.89 versus 101.11). Lower central error therefore coexists with a more extended upper tail. This distributional difference complements the aggregate test-set results in Section 4.2 and motivates examining local field errors in the cases below.

D.2 REPRESENTATIVE AND HIGH-ERROR PREDICTIONS

Figure 11 shows bilateral validation case 040, whose six-field error sum is 41.35. The prediction recovers the broad stress pattern, the two principal PEEQ regions, and the large-scale displacement pattern visible in the ground truth. Residuals vary across the surface, with PEEQ discrepancies concentrated around the active plastic regions. This example connects the low-error end of Figure 9 to spatially resolved predictions.

Figure 12 shows case 033 from Table 4, with a six-field sum of 94.22 and rubber-PEEQ error of 46.57. The overall deformation pattern is retained, but the displayed PEEQ maximum near the upper edge is underestimated. Stress residuals are also concentrated near surface transitions, while displacement residuals extend across a broader region. Compared with case 040, this example shows that recovering the large-scale response does not ensure accurate local plastic amplitudes. The weight-to-module views in Figures 5 and 6 describe how mechanical inputs enter the hierarchy; these prediction maps show the spatial errors that remain after simulation.

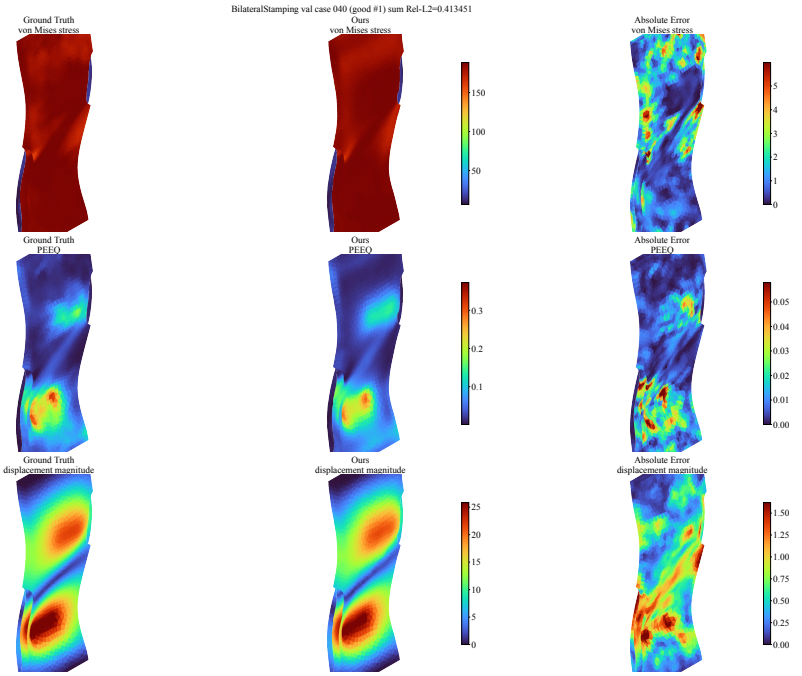


Figure 11: Representative BILATERALSTAMPING prediction with ground truth, EntroMesh, and error maps.

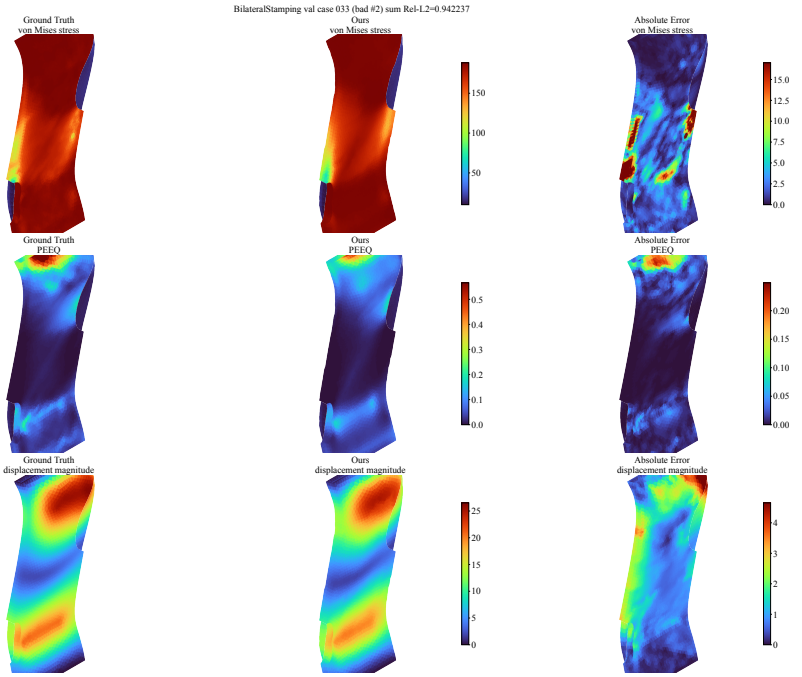


Figure 12: Representative high rubber-PEEQ-error BILATERALSTAMPING case.