RATIO: A RELATION-AWARE TOKEN INTERACTION OPERATOR FOR DEFORMABLE SOLID DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

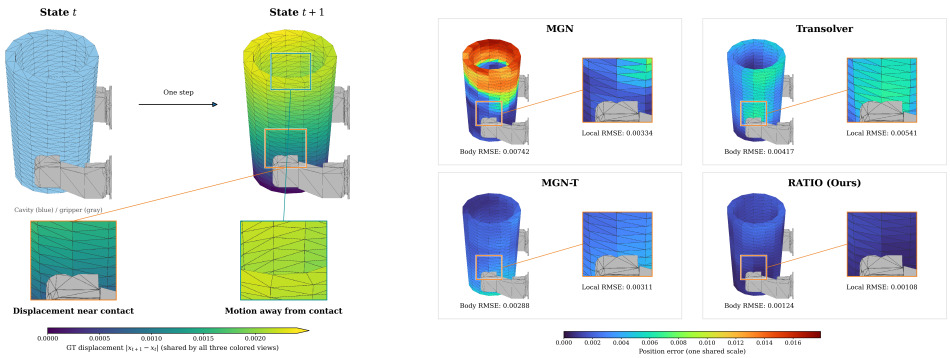
Deep learning has enabled efficient simulation of deformable solids, yet accurately modeling contact dynamics requires capturing both localized physical interactions and long-range deformation responses. Graph-based methods rely on multi-hop message passing to capture long-range dependencies, incurring substantial computational overhead, whereas methods that compress physical states into a small set of latent tokens may lose the fine-grained information needed to resolve local interactions. Incorporating local physical structure directly into compact global representations provides a natural route toward combining localized interaction modeling with global information exchange. For deformable dynamics, however, this requires capturing and distinguishing heterogeneous intra-body and contact relations. Graph-based simulators naturally represent such relations through connectivity and relation-specific edge attributes, motivating a relation-aware formulation for global modeling. In this paper, we propose RATIO, a Relation-Aware Token Interaction Operator that jointly models nodal states, intra-body mesh relations, and inter-body contact relations through token interactions. RATIO aggregates nodal states and the two types of edge features into separate groups of tokens, applies attention across these groups, and maps the resulting information back to update node and edge states. This design allows intra-body and contact relations to directly participate in global evolution. RATIO can further be coupled with local graph modules for enhanced local interaction modeling. Experiments on deformable solid dynamics with contact demonstrate state-of-the-art accuracy.

1 INTRODUCTION

Deformation and contact of solids are prevalent in a wide range of applications, including industrial manufacturing (Clausen et al., 2000; Tao et al., 2025a), aeronautical engineering (Phanden et al., 2021), and robotic manipulation tasks (Huang et al., 2022; Scheikl et al., 2023). Classical numerical methods, such as the Finite Element Method (Dhatt et al., 2012) and the Material Point Method (Bardenhagen et al., 2004), provide established tools for solid simulation, but resolving complex deformation and evolving contact can require substantial computational resources. Recent advances in deep learning have enabled efficient simulation by learning the nonlinear evolution of physical systems directly from data (Luo et al., 2025; Tao et al., 2025b; Pfaff et al., 2020). Learned simulators can therefore predict state changes over larger time intervals and generate trajectories in fewer steps while maintaining stable autoregressive rollouts (Alkin et al., 2025).

At the same time, contact introduces a distinctive challenge in deformable simulation, as localized interactions can induce deformation far beyond the contact region within a single prediction step, particularly when larger time steps are used. As illustrated in Fig. 1(a), contact is confined to the gripper–object interface, while deformation between consecutive states extends across the body. *Accurate autoregressive simulation, therefore, requires resolving local contact interactions alongside the broader deformation response.* Existing learned simulators address these demands through different mechanisms for representing and exchanging physical information. Conventional graph-based simulators explicitly encode mesh connectivity and contact relations through distinct edge types (Linkerhägner et al., 2023; Dubey et al., 2026), but their reliance on local message passing makes it challenging to capture the long-range dependencies underlying the overall deformation response (Iparraguirre et al., 2026). Token-based neural operators, such as Transolver, aggregate mesh-point features into compact latent tokens and apply attention among them to enable efficient

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.



(a) Local contact and body-wide displacement. (b) Comparison of rollout errors across learned models.

Figure 1: Capturing local contact and global response in a cavity grasping scenario.

global communication (Wu et al., 2024; Alkin et al., 2024), but this compression can blur fine-scale spatial variations, making local physical details harder to resolve (Luo et al., 2025). In the example shown in Fig. 1(b), graph-based MGN better preserves deformation near the contact interface but exhibits larger deviations in the overall shape, whereas token-based Transolver more closely recovers the overall shape but loses local contact detail.

To capture both local interactions and long-range responses, recent studies have explored graph structure refinement and local–global integration. Hierarchical graph methods (Yu et al., 2024; Wang et al., 2026) shorten propagation paths through coarser representations, while physics-informed rewiring (Li et al., 2026; Yu et al., 2025) adjusts connectivity to facilitate information exchange along physically relevant paths. A complementary line of work combines local operators for neighborhood interactions with global modules for information exchange across the domain (Ju et al., 2026; Liu et al., 2025). These approaches retain graph-based operators for local interaction modeling. On the other hand, local structural information can be incorporated directly into the representations processed by global operators through tokenization, a strategy that has been widely explored in graph learning (Kim et al., 2022; Gao et al., 2024; Zhao et al., 2026). This allows local relations to guide global information exchange without restricting interactions to graph neighborhoods. Despite this progress, the paradigm remains underexplored in neural operators for deformable simulation.

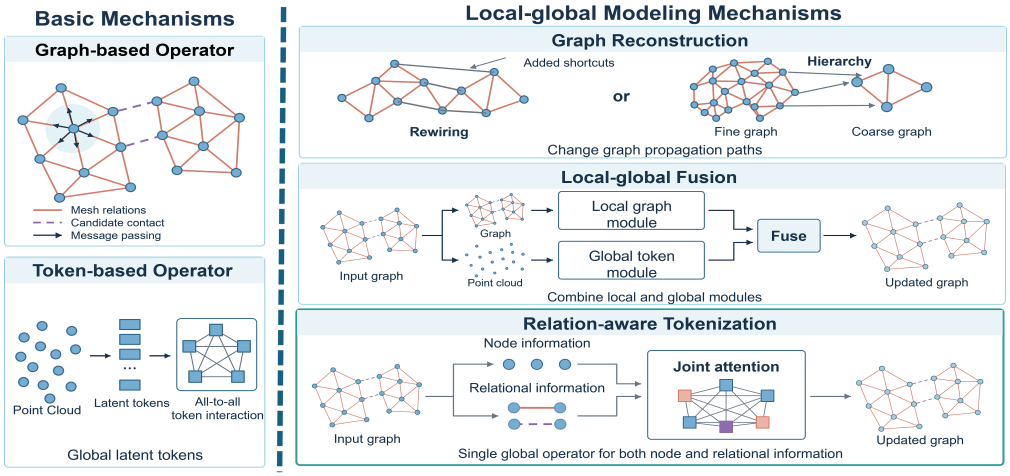


Figure 2: Comparison of interaction mechanisms and local–global modeling paradigms.

Inspired by this perspective, we seek a global operator that can directly incorporate the local interaction structure of deformable solids. A key challenge is that deformable dynamics involve heterogeneous physical relations, with intra-body interactions governing deformation within each

object and inter-body relations characterizing potential contact. These two types of interactions play fundamentally different roles in the system evolution. Graph-based simulators provide a natural representation of such relations, where *graph connectivity identifies the interacting entities, while edge attributes encode their relation-specific geometric and physical properties* (Pfaff et al., 2020). Building on this observation, we introduce a *relation-aware* paradigm for compact global modeling, as illustrated in Fig. 2. Unlike graph reconstruction or local–global fusion, relation-aware tokenization directly brings heterogeneous physical relations into the global representation space. Intra-body and contact relations are represented separately and processed jointly with nodal information, allowing their distinct physical roles to directly shape global information exchange without altering the underlying graph structure.

Building on this paradigm, we introduce RATIO, a Relation-Aware Token Interaction Operator that incorporates local physical relations into compact global computation. RATIO represents nodal states, intra-body mesh relations, and inter-body contact relations as distinct token groups that interact through shared attention. RATIO can serve as a standalone global processor or be coupled with local graph modules in a local–global architecture to further improve local contact responses. RATIO achieves state-of-the-art performance across extensive evaluations of deformable solid dynamics with contact. The main contributions are summarized as follows:

- We propose RATIO, a unified and compact framework that incorporates local intra-body and contact interactions into global computation. Through type-specific tokenization and joint attention, RATIO brings nodal states and heterogeneous physical relations into a shared latent space for global modeling.
- We design a unified local–global architecture that couples RATIO with different local graph modules within a common processing structure. This architecture combines local neighborhood modeling with relation-aware global interaction to better capture localized contact responses and deformation across the physical domain.
- We compare RATIO, both as a standalone global processor and in local–global configurations, with existing methods on deformable solid simulation tasks. The evaluation examines autoregressive prediction accuracy, with ablations assessing the effects of edge sets and token allocation.

2 Related Work

Graph-based Local Physical Simulation. Graph neural networks provide a natural representation of physical systems on irregular discretizations (Mousavi et al., 2026; Würth et al., 2026). As a representative approach, MeshGraphNets (Pfaff et al., 2020) construct mesh edges from mesh connectivity and world edges between spatially nearby nodes to model intra-body and contact interactions and learn their dynamics through edge-based message computation and neighborhood aggregation. Recent work (Feng et al., 2026) provides richer representations of contact boundaries and volumetric structure. However, the limited receptive field of message passing makes it difficult to capture long-range physical dependencies (Alkin et al., 2024). Two lines of research address this limitation through graph design: hierarchical methods shorten communication paths through coarser graph representations (Yu et al., 2024; Deng et al., 2024), while physics-informed graph rewiring modifies connectivity to facilitate information exchange (Yu et al., 2025; Li et al., 2026).

Token-based Global Physical Simulation. An alternative approach compresses spatial representations into a small set of latent tokens and performs global interactions in this compact space (Wang & Wang, 2024; Wen et al., 2026). Transolver (Wu et al., 2024) aggregates mesh points into learnable physical-state slices, while UPT (Alkin et al., 2024) encodes spatial inputs into compact latent representations for global processing and flexible decoding. While these methods enable efficient global communication, capturing local heterogeneity and sharp spatial variations remains challenging (Ju et al., 2026). To better capture these localized physical features, recent methods combine token-based global computation with local operators. MGN-T (Iparraguirre et al., 2026) integrates physics attention with graph-based preprocessing and refinement, while APT (Ju et al., 2026) fuses local geometric aggregation with global attention. These methods retain graph-based operators to model local interactions within architectures that support global communication.

Relation-aware Tokenization. A complementary direction incorporates local structure directly into global token representations. TokenGT (Kim et al., 2022) and GraphsGPT (Gao et al., 2024) represent nodes and edges as tokens, using structural identifiers to encode graph connectivity for global Transformer interactions. In physical simulation, however, relations also carry geometric and physical attributes that characterize how entities interact. Unisoma (Tao et al., 2025b) incorporates nodal and intra-object edge features into shared object-level tokens. Deformable dynamics require intra-body and contact relations to be captured and distinguished within the same system, motivating a unified, *relation-aware* formulation for global modeling.

3 METHODOLOGY

3.1 PROBLEM FORMULATION

We represent interacting solids by a unified Lagrangian point set $\mathcal{V} = \{1, \dots, N\}$. Each point i has a position $\mathbf{p}_i^k \in \mathbb{R}^3$ at step k, a fixed reference position $\mathbf{X}_i = \mathbf{p}_i^0$, and an object identifier o_i. The positions form $\mathbf{P}^k \in \mathbb{R}^{N\times 3}$. Given reference-mesh adjacency A_{ij}, we define $\mathcal{G}^k = (\mathcal{V}, \mathcal{E}_m, \mathcal{E}_w^k)$ by

$$\begin{aligned} \mathcal{E}_m &= \{(i,j) \in \mathcal{V}\times\mathcal{V} : A_{ij} = 1,\ o_i = o_j\}, \\ \mathcal{E}_w^k &= \{(i,j) \in \mathcal{V}\times\mathcal{V} : o_i \neq o_j,\ \|\mathbf{p}_i^k - \mathbf{p}_j^k\|_2 < \rho\}. \end{aligned} \tag{1}$$

Mesh edges preserve native material connectivity; world edges represent potential contact between different solids within radius ρ. Reference coordinates $\mathbf{X} = [\mathbf{X}_1, \dots, \mathbf{X}_N]^\top = \mathbf{P}^0$ provide a fixed geometric baseline for deformation and, together with mesh connectivity and point labels, form the static information $\mathcal{M}$.

Given $\mathbf{P}^0$ and prescribed motions $\{\mathbf{C}^k\}_{k=0}^{T_{\text{roll}}-1}$, we predict $\{\mathbf{P}^0, \widehat{\mathbf{P}}^1, \dots, \widehat{\mathbf{P}}^{T_{\text{roll}}}\}$ at $t_k = k\Delta t$ through autoregressive rollout. Each $\mathbf{C}^k \in \mathbb{R}^{N\times 3}$ contains prescribed next-step displacements at controlled points and zeros elsewhere. The mask $\mathbf{m} \in \{0,1\}^{N\times 1}$ is one at predicted deformable points and zero at controlled points. A learned displacement predictor advances the configuration as

$$\widehat{\mathbf{P}}^{k+1} = \widehat{\mathbf{P}}^k + \mathbf{m} \odot \mathcal{F}_\theta(\widehat{\mathbf{P}}^{k-1}, \widehat{\mathbf{P}}^k, \mathbf{C}^k, \widehat{\mathcal{G}}^k; \mathcal{M}) + \mathbf{C}^k, \tag{2}$$

where $\mathcal{F}_\theta$ returns $N\times 3$ physical displacements and $\mathbf{m}$ is broadcast across coordinates. Initialization $\widehat{\mathbf{P}}^{-1} = \widehat{\mathbf{P}}^0 = \mathbf{P}^0$ gives zero initial backward displacement. At each step, $\widehat{\mathcal{G}}^k$ rebuilds world edges from $\widehat{\mathbf{P}}^k$, while mesh connectivity and reference coordinates remain fixed. A relatively large Δt may encompass both localized contact changes and body-wide deformation, requiring their joint modeling within each prediction step.

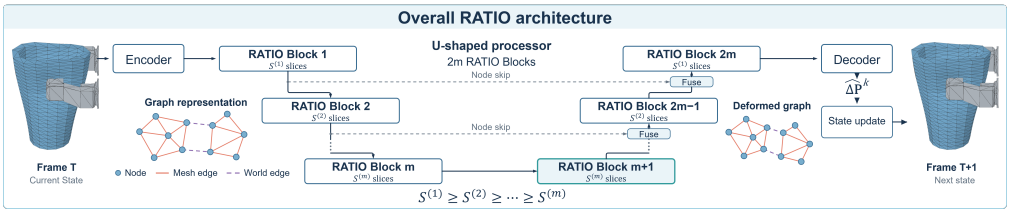
3.2 OVERALL ARCHITECTURE

Figure 3(a) illustrates the overall RATIO architecture and the core design of the RATIO processor. The architecture follows an encoder–processor–decoder design with a U-shaped processor built from RATIO-based blocks. Within each RATIO block, node, mesh-edge, and world-edge states are compressed into separate token groups, jointly processed through attention, and updated through redistribution with residual connections. We describe its overall computation below and detail the RATIO block and its local integration in the following subsections.

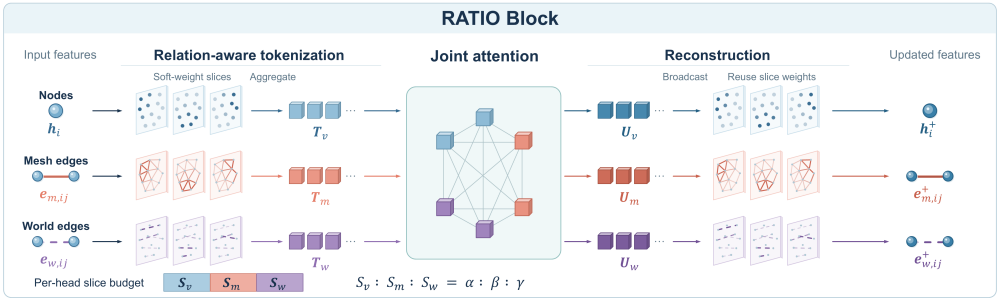
Encoder. At rollout step k, node features $\mathbf{f}_{v,i}^k$ encode current positions, backward displacements, prescribed motion, and point types. Edge features $\mathbf{f}_{m,ij}^k$ and $\mathbf{f}_{w,ij}^k$ use current offsets $\mathbf{r}_{ij}^k = \widehat{\mathbf{p}}_i^k - \widehat{\mathbf{p}}_j^k$; mesh edges include reference offsets $\mathbf{R}_{ij} = \mathbf{X}_i - \mathbf{X}_j$ additionally, giving

$$\begin{aligned} \mathbf{f}_{v,i}^k &= \big[\widehat{\mathbf{p}}_i^k,\ \widehat{\mathbf{p}}_i^k - \widehat{\mathbf{p}}_i^{k-1},\ (1-m_i)(\widehat{\mathbf{p}}_i^k + \mathbf{c}_i^k),\ \mathbf{c}_i^k,\ \boldsymbol{\tau}_i\big] \in \mathbb{R}^{21}, \\ \mathbf{f}_{m,ij}^k &= \big[\mathbf{r}_{ij}^k,\ \|\mathbf{r}_{ij}^k\|_2,\ \mathbf{R}_{ij},\ \|\mathbf{R}_{ij}\|_2\big] \in \mathbb{R}^8, \qquad \mathbf{f}_{w,ij}^k = \big[\mathbf{r}_{ij}^k,\ \|\mathbf{r}_{ij}^k\|_2\big] \in \mathbb{R}^4. \end{aligned} \tag{3}$$

Here, $\mathbf{c}_i^k$ is row i of $\mathbf{C}^k$, $\boldsymbol{\tau}_i \in \mathbb{R}^9$ is the one-hot point-type encoding, and brackets denote feature concatenation. Separate MLPs map these features to d-dimensional latent states, forming $\mathbf{H}^0$, $\mathbf{E}_m^0$, and $\mathbf{E}_w^0$, respectively.



(a) Overall RATIO architecture.



(b) RATIO-based processor block.

Figure 3: Overall RATIO architecture and RATIO-based processor block.

Processor. We omit the latent time index and use ℓ for block depth. The processor updates

$$(\mathbf{H}^{\ell}, \mathbf{E}_m^{\ell}, \mathbf{E}_w^{\ell}) = \mathcal{B}_{\ell}(\widetilde{\mathbf{H}}^{\ell-1}, \mathbf{E}_m^{\ell-1}, \mathbf{E}_w^{\ell-1}; \widehat{\mathcal{G}}^k), \quad \ell = 1, \dots, 2m. \tag{4}$$

Inspired by the multiscale architecture (Ronneberger et al., 2015), we organize the processor into a U-shaped backbone with $2m$ blocks. The number of slices per attention head decreases as $S^{(1)} \geq \cdots \geq S^{(m)}$ through the first m blocks and increases in reverse order through the remaining m blocks, capturing information at various resolutions while preserving mesh resolution. Skip connections link corresponding early and late stages through learned node-wise residual fusion,

$$\widetilde{\mathbf{H}}^{\ell-1} = \begin{cases} \mathbf{H}^{\ell-1}, & 1 \leq \ell \leq m+1, \\ \mathbf{H}^{\ell-1} + \Psi_{\ell}\left([\mathbf{H}^{\ell-1}, \mathbf{H}^{2m-\ell+1}]\right), & m+2 \leq \ell \leq 2m, \end{cases} \tag{5}$$

where $\Psi_{\ell} = \mathrm{MLP}_{\ell} \circ \mathrm{LN}_{\ell}$ maps the concatenated $2d$-dimensional node features to d dimensions. Each processor block applies RATIO, optionally integrates a local operator, and ends with a residual node feed-forward update.

Decoder. A node-wise MLP $\mathrm{Dec}_{\theta}(\cdot)$ decodes the final node states to advance the configuration:

$$\widehat{\mathbf{P}}^{k+1} = \widehat{\mathbf{P}}^{k} + \mathbf{m} \odot \mathrm{Dec}_{\theta}(\mathbf{H}^{2m}) + \mathbf{C}^{k}. \tag{6}$$

3.3 Relation-Aware Token Interaction Block

Modeling interacting solids requires relating point motion to both material connectivity within each body and potential contact between bodies. RATIO preserves these distinct sources of information in separate token groups, then couples them through joint attention. This formulation incorporates local physical structure directly into compact global computation, allowing information exchange beyond graph-defined neighborhoods. Dedicated token groups reserve representational capacity for each physical role, including contact relations, whose number varies across rollout steps.

Figure 3(b) illustrates the RATIO block. Consider its application in processor block ℓ at rollout step k. For brevity, denote its input states by

$$(\mathbf{H}, \mathbf{E}_m, \mathbf{E}_w) = (\widetilde{\mathbf{H}}^{\ell-1}, \mathbf{E}_m^{\ell-1}, \mathbf{E}_w^{\ell-1}). \tag{7}$$

The entity sets are $\mathcal{I}_v = \mathcal{V}$, $\mathcal{I}_m = \mathcal{E}_m$, and $\mathcal{I}_w = \widehat{\mathcal{E}}_w^k$, where $\widehat{\mathcal{E}}_w^k$ is the world edge set of $\widehat{\mathcal{G}}^k$.

Relation-aware representations. We combine each edge state with the states of its two endpoints to form a relational representation before tokenization. Type-specific affine projections give

$$\mathbf{y}_{v,i} = \phi_v(\mathbf{h}_i), \qquad \mathbf{y}_{r,ij} = \phi_r([\mathbf{h}_i, \mathbf{e}_{r,ij}, \mathbf{h}_j]), \quad r \in \{m, w\}, \tag{8}$$

where all projected features lie in $\mathbb{R}^d$. This lets slicing assignments depend on both the relation and its endpoint states, with separate projections for material connections and potential contact. For clarity, we describe a single attention head using $\mathbf{Y}_r \in \mathbb{R}^{|\mathcal{I}_r| \times d_h}$, $r \in \{v, m, w\}$, where n_h is the number of attention heads and $d_h = d/n_h$ is the feature dimension of each head.

Typed slicing and allocation ratio. Nodes, mesh edges, and world edges encode point states, material connectivity, and potential contact, respectively. Their cardinalities differ, with world-edge counts varying across rollout steps. Joint pooling must account for these count differences when balancing contributions from different entity types (Tao et al., 2025b). Beyond this numerical imbalance, pooling physically distinct features into shared slices mixes their information during aggregation, without retaining separate token representations for their respective roles.

We therefore allocate a separate slice bank to each type, preserving these distinctions while controlling token capacity independently of entity counts. In particular, a fixed world-edge bank provides dedicated capacity for contact relations as their number varies across rollout steps. For each head, the three banks contain S_v, S_m, and S_w slices, respectively, satisfying $S = S_v + S_m + S_w$ and $S_v : S_m : S_w = \alpha : \beta : \gamma$, where S is the per-head slice budget of the current block and each S_r is a positive integer fixed across rollout steps. The allocation ratio $\alpha : \beta : \gamma$ is fixed across heads and processor depths, while assignments within each bank are learned. As a global operator for nodal dynamics, RATIO prioritizes node tokens while retaining dedicated capacity for both edge types. We adopt $\alpha : \beta : \gamma = 6 : 1 : 1$ as an empirical choice, with alternative allocations examined in Section 4.3. For each type $r \in \{v, m, w\}$, we compute the assignments and pooled tokens as

$$\mathbf{A}_r = \mathrm{softmax}(\psi_r(\mathbf{Y}_r)) \in \mathbb{R}^{|\mathcal{I}_r| \times S_r}, \quad \mathbf{T}_r = \mathbf{D}_r^{-1} \mathbf{A}_r^\top \mathbf{Y}_r \in \mathbb{R}^{S_r \times d_h}, \tag{9}$$

where $\psi_r : \mathbb{R}^{d_h} \to \mathbb{R}^{S_r}$ maps each entity representation to slice-assignment logits, and $\mathbf{D}_r$ is diagonal with entries $D_{r,ss} = \max(\sum_{u \in \mathcal{I}_r} A_{r,us}, \epsilon)$ for $\epsilon > 0$. The softmax distributes each entity over the slots of the corresponding type, and $\mathbf{D}_r$ normalizes each token by its accumulated assignment weight. World-edge pooling is performed independently for each frame, yielding S_w tokens per head without requiring edge correspondence across frames.

Joint token interaction. After type-specific pooling, joint attention couples the three token banks. We concatenate them along the token dimension and add a learned slot embedding $\mathbf{\Pi} \in \mathbb{R}^{S \times d_h}$,

$$\mathbf{B} = \mathrm{Concat}(\mathbf{T}_v, \mathbf{T}_m, \mathbf{T}_w) + \mathbf{\Pi}, \quad \mathbf{U} = \mathrm{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_h}}\right)\mathbf{V}, \tag{10}$$

where $\mathbf{Q} = \mathbf{B}\mathbf{W}_Q$, $\mathbf{K} = \mathbf{B}\mathbf{W}_K$, and $\mathbf{V} = \mathbf{B}\mathbf{W}_V$, with $\mathbf{W}_Q, \mathbf{W}_K, \mathbf{W}_V \in \mathbb{R}^{d_h \times d_h}$. Attention spans all S slots, allowing nodal states, material relations, and contact relations to exchange information. We then partition $\mathbf{U}$ into $\mathbf{U}_v$, $\mathbf{U}_m$, and $\mathbf{U}_w$ according to the original bank sizes.

Typed redistribution and residual updates. The same assignments redistribute the updated tokens to their source entities within each frame. Restoring the head index a, we concatenate the head outputs and apply a type-specific affine projection $\omega_r : \mathbb{R}^d \to \mathbb{R}^d$ row-wise, $\Delta\mathbf{Z}_r = \omega_r\Big(\mathrm{Concat}\left\{\mathbf{A}_r^{(a)}\mathbf{U}_r^{(a)}\right\}_{a=1}^{n_h}\Big)$ for $r \in \{v, m, w\}$. Each update is then added to the corresponding input state through an independent identity path, $\mathbf{H}^+ = \mathbf{H} + \Delta\mathbf{Z}_v$, $\mathbf{E}_m^+ = \mathbf{E}_m + \Delta\mathbf{Z}_m$, and $\mathbf{E}_w^+ = \mathbf{E}_w + \Delta\mathbf{Z}_w$. With relatively few slices allocated to edges, these residual paths are particularly important for preserving edge-specific information alongside the compressed global updates. For a given graph and fixed head width and count, slicing, interaction, and redistribution cost $O(NS_v + |\mathcal{E}_m|S_m + |\widehat{\mathcal{E}}_w^k|S_w + S^2)$, excluding graph construction. The quadratic attention cost thus depends on the token budget S, rather than the number of graph entities.

3.4 Local–Global Fusion

RATIO incorporates mesh and contact relations into global token interactions, enabling global state updates informed by local physical structure. To further strengthen the modeling of localized interactions, we complement this global computation with explicit neighborhood propagation along

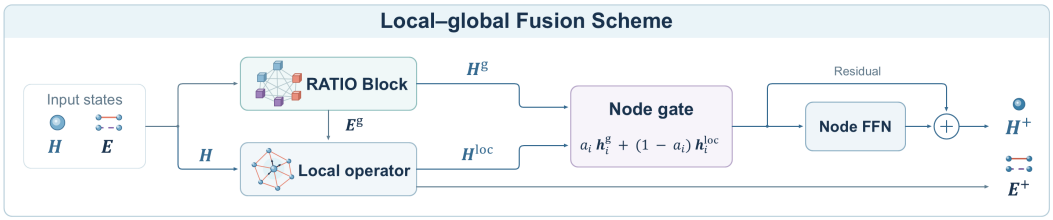


Figure 4: Local–global fusion within a RATIO-based processor block.

mesh and contact edges. This optional local augmentation is realized through a fusion block that couples RATIO with a local graph operator.

For fusion, we relabel the RATIO outputs as $\mathbf{H}^g$, $\mathbf{E}_m^g$, and $\mathbf{E}_w^g$, and reserve the superscript $+$ for the final block outputs. The local branch takes the input node states and the RATIO-updated edge states:

$$(\mathbf{H}^{\mathrm{loc}}, \mathbf{E}_m^+, \mathbf{E}_w^+) = \mathcal{P}_\ell(\mathbf{H}, \mathbf{E}_m^g, \mathbf{E}_w^g; \widehat{\mathcal{G}}^k), \tag{11}$$

where $\mathcal{P}_\ell$ is a local graph operator performing relation-specific aggregation with residual node and edge updates. A node-wise scalar gate combines the two node outputs, followed by a residual feed-forward update:

$$a_i = \sigma\big(\mathrm{MLP}([\mathbf{h}_i^g, \mathbf{h}_i^{\mathrm{loc}}])\big), \quad \overline{\mathbf{h}}_i = a_i \mathbf{h}_i^g + (1 - a_i)\mathbf{h}_i^{\mathrm{loc}}, \quad \mathbf{h}_i^+ = \overline{\mathbf{h}}_i + \mathrm{FFN}_\ell\big(\mathrm{LN}_\ell(\overline{\mathbf{h}}_i)\big). \tag{12}$$

Here, σ is the sigmoid function, and a_i is broadcast across feature channels. Both branches retain the input node state through residual connections, so their convex combination preserves a single identity contribution. The gate MLP is shared across processor blocks and initialized to give $a_i = 1/2$. Without local augmentation, we set $\overline{\mathbf{h}}_i = \mathbf{h}_i^g$.

3.5 Training Objective

We train the model with single-step supervision, perturbing the current positions of predicted points with Gaussian noise, $\widetilde{\mathbf{p}}_i^k = \mathbf{p}_i^k + m_i \boldsymbol{\epsilon}_i^k$, where $\boldsymbol{\epsilon}_i^k \sim \mathcal{N}(\mathbf{0}, \sigma_n^2 \mathbf{I}_3)$. The target displacement is measured from the perturbed current position to the clean next-step position, $\mathbf{p}_i^{k+1} - \widetilde{\mathbf{p}}_i^k$. Let $\widehat{\mathbf{z}}_i^k$ and $\mathbf{z}_i^k$ denote the predicted and target displacements normalized using fixed training-set statistics. We minimize $\mathcal{L}(\theta) = \mathbb{E}\left[\frac{\sum_{i=1}^N m_i \left\| \widehat{\mathbf{z}}_i^k - \mathbf{z}_i^k \right\|_2^2}{3 \sum_{i=1}^N m_i}\right]$, where the expectation is over training transitions and noise, and m_i selects predicted points as defined above.

4 Experiments

4.1 Experimental Setup

Datasets. To evaluate the effectiveness of RATIO in modeling deformable-solid dynamics with contact, we conduct experiments on two benchmarks: Cavity Grasping (CG) and Tissue Manipulation (TM) (Linkerhägner et al., 2023; Tao et al., 2025b; Feng et al., 2026). CG features a deformable cavity grasped by rigid fingers, whereas TM involves tissue manipulated by moving rigid grippers. The graphs contain 1,386 nodes for CG and 363 nodes for TM. Both benchmarks couple localized contact interactions with global deformation responses, making them well suited to evaluating RATIO's joint modeling of intra-body and contact relations. Their distinct geometries and mesh sizes further provide complementary settings for assessing its effectiveness.

Baselines. We compare RATIO against representative methods with different interaction mechanisms. Local graph baselines include MGN (Pfaff et al., 2020), GraphSAGE (Hamilton et al., 2017), and GATv2 (Brody et al., 2021). We adapt GraphSAGE and GATv2 to distinguish mesh and contact edges. Global baselines include Transolver (Wu et al., 2024), GNOT (Hao et al., 2023), and LSM (Wu et al., 2023). We further compare with local–global hybrids MGN-T (Iparraguirre et al., 2026) and AeroGTO (Liu et al., 2025), the hierarchical contact model HCMT (Yu et al., 2024), and the solid-aware interaction model Unisoma (Tao et al., 2025b).

Table 1: Test positional RMSE ($\times 10^{-3}$; lower is better). For each dataset, the best, second-best, and third-best results are indicated by boldface, underlining, and light gray shading, respectively.

Local / global baselines			Hybrid baselines / RATIO		
Method	CG	TM	Method	CG	TM
MGN	8.83	17.78	Unisoma	17.02	14.88
GraphSAGE	15.12	12.19	AeroGTO	17.85	12.19
GATv2	18.21	13.10	HCMT	7.30	16.35
Transolver	15.82	10.75	MGN-T	4.41	8.60
GNOT	17.48	13.36	RATIO-only	<u>4.10</u>	<u>7.31</u>
LSM	17.83	14.03	RATIO+MGN	**2.51**	**5.45**

Experimental settings. We evaluate all models on 104-step autoregressive test rollouts using the root mean square error (RMSE) of node positions. The reported errors are averaged over all rollout steps and test trajectories. For a fair comparison, all models are configured to have comparable parameter counts. Further experimental details are provided in the appendix.

4.2 Experimental Results

Prediction Accuracy. Table 1 presents the autoregressive rollout results of all models on CG and TM. RATIO consistently outperforms the compared baselines on both datasets. Even without an additional local graph module, standalone RATIO reduces positional RMSE by 7.0% on CG and 15.0% on TM relative to MGN-T, the strongest baseline on both datasets. These results support the effectiveness of incorporating mesh and contact relations into global token interactions. Integrating MGN further lowers the errors to 2.51 and 5.45, corresponding to reductions of 43.1% and 36.6% over MGN-T, respectively. Compared with standalone RATIO, local fusion reduces the errors by 38.8% on CG and 25.4% on TM. RATIO incorporates physical relations through compact global token interactions, while the local module complements this mechanism with direct neighborhood propagation. The observed gains suggest that such local refinement can further enhance RATIO's modeling of localized dynamics, with a more pronounced benefit on the CG dataset.

Local–Global Complementarity. To examine the interplay between local propagation and global interactions, we combine different local and global operators within our fusion framework.

Table 2: Test positional RMSE ($\times 10^{-3}$; lower is better) for different combinations of local and global operators. Within each dataset and global-operator column, the best and second-best results are marked in bold and underlined, respectively. "None" indicates the absence of the corresponding operator, and "n/a" denotes the configuration with neither operator. Trans. denotes Transolver.

	CG				TM			
Local	None	LSM	Trans.	RATIO	None	LSM	Trans.	RATIO
None	n/a	17.83	15.82	4.10	n/a	14.03	10.75	7.31
MGN	**8.83**	**3.89**	12.16	**2.51**	17.78	**9.07**	**9.30**	**5.45**
GraphSAGE	<u>15.12</u>	5.80	**4.23**	<u>3.50</u>	**12.19**	9.76	<u>9.58</u>	<u>6.76</u>
GATv2	18.21	<u>4.29</u>	<u>9.58</u>	8.37	<u>13.10</u>	<u>9.71</u>	13.11	8.39

Table 2 compares local–global combinations and standalone models with comparable parameter counts. Most fused configurations outperform both standalone counterparts, highlighting the benefits of combining local propagation with global interactions. Across the three local modules and two datasets, RATIO achieves the lowest error in five of six comparisons, demonstrating its effectiveness across different fusion configurations. At the same time, fusion gains vary across operator pairings, with GATv2 increasing RATIO's errors on both datasets. Notably, standalone accuracy does not directly determine fusion benefits: MGN is the weakest standalone local model on TM, yet provides the best local partner for all three global operators. We leave a more systematic investigation of operator compatibility and fusion design to future work.

Qualitative comparison. Figure 5 compares autoregressive predictions on CG and TM at three rollout stages. RATIO+MGN more closely reproduces the ground-truth configurations, with reduced positional errors along the cavity walls in CG and around the bent tissue region in TM. These

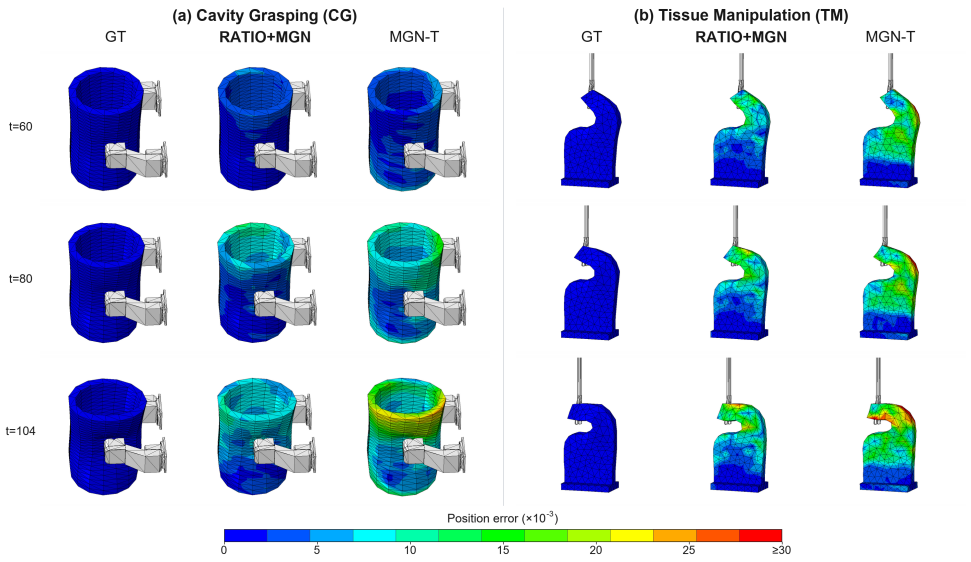


Figure 5: Qualitative comparison on CG (left) and TM (right).

improvements remain evident at later rollout stages, with approximately 34–44% lower nodal position RMSE than MGN-T across the displayed snapshots.

4.3 Ablation Studies

RATIO jointly models nodal states and heterogeneous edge relations through global token interactions. On TM, we ablate token allocation and relation contributions under a fixed total token budget. We compare node:mesh:world allocations of 6:1:1 and 2:1:1 with a Shared variant, and remove either mesh or world relations under each scheme. When a relation type is removed, its token quota is reassigned to the remaining edge type.

Table 3: Ablation results of standalone RATIO on TM ($\times 10^{-3}$; lower is better).

Relations retained	6:1:1	2:1:1	Shared
Nodes + mesh + world	7.31	8.28	14.17
Nodes + world	7.74	8.43	9.67
Nodes + mesh	8.30	9.16	11.80

Table 3 shows that type-specific tokenization outperforms Shared, suggesting that separating entities with unequal cardinalities and heterogeneous features helps balance their contributions. Among single-relation variants, retaining world edges yields lower error than retaining mesh edges, highlighting their importance in TM. The consistent advantage of 6:1:1 over 2:1:1 further supports allocating a larger share of token capacity to nodes.

5 Conclusion and Limitations

We introduced RATIO, a relation-aware global operator that jointly models nodal states, mesh relations, and contact relations through compact, type-specific token interactions. Experiments on deformable solid dynamics demonstrate strong predictive performance and the benefit of explicitly incorporating heterogeneous physical relations into global modeling. Several limitations remain. First, training with the current implementation is time-consuming, motivating more efficient optimization and implementation strategies. Second, the design space of local–global integration, including the choice and coupling of local and global operators, remains to be explored more systematically. Finally, the current evaluation is limited to relatively controlled benchmark settings. Future work will extend RATIO to more complex and realistic scenarios involving diverse geometries, materials, and contact conditions.

AI USE STATEMENT

We used generative AI tools primarily to revise and polish the manuscript and to assist with drafting auxiliary code for model implementation and experimental workflows. These tools also assisted in comparing and interpreting experimental results. The core methodological ideas and algorithmic design were developed by the authors. The authors reviewed all AI-assisted text, checked the quantitative comparisons against the experimental results, and verified the auxiliary code. We take full responsibility for the final content of this work, including all AI-assisted text and code.

REPRODUCIBILITY STATEMENT

The model architecture and experimental details are provided in the main text and appendix. We include source code in the supplementary materials to facilitate reproducibility.

REFERENCES

Benedikt Alkin, Andreas Fürst, Simon Schmid, Lukas Gruber, Markus Holzleitner, and Johannes Brandstetter. Universal physics transformers: A framework for efficiently scaling neural operators. *Advances in Neural Information Processing Systems*, 37:25152–25194, 2024.

Benedikt Alkin, Tobias Kronlachner, Samuele Papa, Stefan Pirker, Thomas Lichtenegger, and Johannes Brandstetter. Neuraldem for real time simulations of industrial particular flows. *Communications Physics*, 8(1):440, 2025.

Scott G Bardenhagen, Edward M Kober, et al. The generalized interpolation material point method. *Computer modeling in engineering and sciences*, 5(6):477–496, 2004.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28, 2015.

Shaked Brody, Uri Alon, and Eran Yahav. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*, 2021.

Arild H Clausen, Odd S Hopperstad, and Magnus Langseth. Stretch bending of aluminium extrusions for car bumpers. *Journal of Materials Processing Technology*, 102(1-3):241–248, 2000.

Huayu Deng, Xiangming Zhu, Yunbo Wang, and Xiaokang Yang. Evomesh: Adaptive physical simulation with hierarchical graph evolutions. *arXiv preprint arXiv:2410.03779*, 2024.

Gouri Dhatt, Emmanuel Lefrançois, and Gilbert Touzot. *Finite element method*. John Wiley & Sons, 2012.

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.

Zhe Feng, Shilong Tao, Haonan Sun, Shaohan Chen, Zhanxing Zhu, and Yunhuai Liu. Maven: A mesh-aware volumetric encoding network for simulating 3d flexible deformation. *arXiv preprint arXiv:2604.04474*, 2026.

Zhangyang Gao, Daize Dong, Cheng Tan, Jun Xia, Bozhen Hu, and Stan Z. Li. A graph is worth K words: Euclideanizing graph using pure transformer. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 14681–14701. PMLR, 2024.

Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. *Advances in neural information processing systems*, 30, 2017.

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 *International conference on machine learning*, pp. 12556–12569. PMLR, 2023.

Isabella Huang, Yashraj Narang, Clemens Eppner, Balakumar Sundaralingam, Miles Macklin, Ruzena Bajcsy, Tucker Hermans, and Dieter Fox. Defgraspsim: Physics-based simulation of grasp outcomes for 3d deformable objects. *IEEE Robotics and Automation Letters*, 7(3):6274–6281, 2022.

Mikel M Iparraguirre, Iciar Alfaro, David Gonzalez, and Elias Cueto. Meshgraphnet-transformer: Scalable mesh-based learned simulation for solid mechanics. *arXiv preprint arXiv:2601.23177*, 2026.

Xin Ju, Nok Hei, Yuyan Zhang, Carl Jacquemyn, Matthew Jackson, Randolph Settgast, Sally M Benson, Gege Wen, et al. Adaptive physics transformer with fused global-local attention for subsurface energy systems. *arXiv preprint arXiv:2602.11208*, 2026.

Jinwoo Kim, Dat Nguyen, Seonwoo Min, Sungjun Cho, Moontae Lee, Honglak Lee, and Seunghoon Hong. Pure transformers are powerful graph learners. *Advances in Neural Information Processing Systems*, 35:14582–14595, 2022.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Zenong Li, Wei Xian Lim, Wai Lee Chan, and Adams Wai-Kin Kong. Graph rewiring based on flow alignment for improving fluid simulation. In *International Conference on Machine Learning*, 2026.

Jonas Linkerhägner, Niklas Freymuth, Paul Maria Scheikl, Franziska Mathis-Ullrich, and Gerhard Neumann. Grounding graph network simulators using physical sensor observations. In *International conference on learning representations*, 2023.

Pengwei Liu, Pengkai Wang, Xingyu Ren, Hangjie Yuan, Zhongkai Hao, Chao Xu, Shengze Cai, and Dong Ni. Aerogto: An efficient graph-transformer operator for learning large-scale aerodynamics of 3d vehicle geometries. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 18924–18932, 2025.

Huakun Luo, Haixu Wu, Hang Zhou, Lanxiang Xing, Yichen Di, Jianmin Wang, and Mingsheng Long. Transolver++: An accurate neural solver for pdes on million-scale geometries. *arXiv preprint arXiv:2502.02414*, 2025.

Sepehr Mousavi, Shizheng Wen, Levi Lingsch, Maximilian Herde, Bogdan Raonic, and Siddhartha Mishra. Rigno: A graph-based framework for robust and accurate operator learning for pdes on arbitrary domains. *Advances in Neural Information Processing Systems*, 38:150039–150101, 2026.

Tobias Pfaff, Meire Fortunato, Alvaro Sanchez-Gonzalez, and Peter Battaglia. Learning mesh-based simulation with graph networks. In *International Conference on Learning Representations*, 2020.

Rakesh Kumar Phanden, Priavrat Sharma, and Anubhav Dubey. A review on simulation in digital twin for aerospace, manufacturing and robotics. *Materials Today: Proceedings*, 38:174–178, 2021.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 234–241. Springer, 2015.

Paul Maria Scheikl, BalĂAzs Gyenes, Rayan Younis, Christoph Haas, Gerhard Neumann, Martin Wagner, and Franziska Mathis-Ullrich. Lapgym-an open source framework for reinforcement learning in robot-assisted laparoscopic surgery. *Journal of Machine Learning Research*, 24(368): 1–42, 2023.

Shilong Tao, Zhe Feng, Haonan Sun, Zhanxing Zhu, and Yunhuai Liu. Ladeep: A deep learning-based surrogate model for large deformation of elastic-plastic solids. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 4901–4912, 2025a.

Shilong Tao, Zhe Feng, Haonan Sun, Zhanxing Zhu, and Yunhuai Liu. Unisoma: A unified transformer-based solver for multi-solid systems. In *Forty-second International Conference on Machine Learning*, 2025b.

Caoliwen Wang, Minghao Guo, Siyuan Chen, Heng Zhang, Mengdi Wang, Xingyu Ni, Hanson Sun, Kunyi Wang, Zherong Pan, Kui Wu, et al. Worldparticle: Unified world simulation of lagrangian particle dynamics via transformer. *arXiv preprint arXiv:2605.15305*, 2026.

Tian Wang and Chuang Wang. Latent neural operator for solving forward and inverse pde problems. *Advances in Neural Information Processing Systems*, 37:33085–33107, 2024.

Shizheng Wen, Arsh Kumbhat, Levi Lingsch, Sepehr Mousavi, Yizhou Zhao, Praveen Chandrashekar, and Siddhartha Mishra. Geometry aware operator transformer as an efficient and accurate neural surrogate for pdes on arbitrary domains. *Advances in Neural Information Processing Systems*, 38:155423–155501, 2026.

Haixu Wu, Tengge Hu, Huakun Luo, Jianmin Wang, and Mingsheng Long. Solving high-dimensional pdes with latent spectral models. *arXiv preprint arXiv:2301.12664*, 2023.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for pdes on general geometries. *arXiv preprint arXiv:2402.02366*, 2024.

Tobias Würth, Niklas Freymuth, Gerhard Neumann, and Luise Kärger. Diffusion-based hierarchical graph neural networks for simulating nonlinear solid mechanics. *Advances in Neural Information Processing Systems*, 38:130172–130201, 2026.

Youn-Yeol Yu, Jeongwhan Choi, Woojin Cho, Kookjin Lee, Nayong Kim, Kiseok Chang, Chang-Seung Woo, Ilho Kim, Seok-Woo Lee, Joon-Young Yang, et al. Learning flexible body collision dynamics with hierarchical contact mesh transformer. In *International Conference on Learning Representations*, 2024.

Youn-Yeol Yu, Jeongwhan Choi, Jaehyeon Park, Kookjin Lee, and Noseong Park. Piorf: Physics-informed ollivier-ricci flow for long-range interactions in mesh graph neural networks. *arXiv preprint arXiv:2504.04052*, 2025.

Yuming Zhao, Zangyueyang Xian, Qijian Zhang, Rendong Liang, Qin Jia, Ying He, and Junhui Hou. Seamflow: Structure-aware flow matching on edge probabilities for artist-like uv unwrapping. *arXiv preprint arXiv:2609.04751*, 2026.

A Datasets

We evaluate on Cavity Grasping (CG) and Tissue Manipulation (TM), two three-dimensional benchmarks for deformable-solid dynamics under prescribed manipulation (Linkerhägner et al., 2023; Tao et al., 2025b). Each dataset contains 840 trajectories, with 600 for training, 120 for validation, and 120 for testing. Each trajectory records 105 states, giving 104 autoregressive prediction steps. The prediction targets are the positions of the deformable vertices, while the rigid-body or grasping-point motion is provided as a boundary condition.

Cavity Grasping (CG). CG simulates a cone-shaped elastic cavity manipulated by a rigid gripper, as illustrated in Figure 6. The two gripper jaws follow different prescribed motions and are represented as separate rigid bodies. Cavity geometry and elastic properties vary across trajectories, exposing the simulator to different shape responses under manipulation. The benchmark description includes Poisson's ratios of -0.9, 0, and 0.49 (Tao et al., 2025b). Each scene contains 1,386 mesh vertices, comprising 750 cavity vertices and 636 gripper vertices. Given the initial configuration and the prescribed jaw trajectories, the task is to predict the evolving cavity geometry over the rollout. This requires capturing deformation near the contact regions together with the resulting shape changes across the curved cavity.

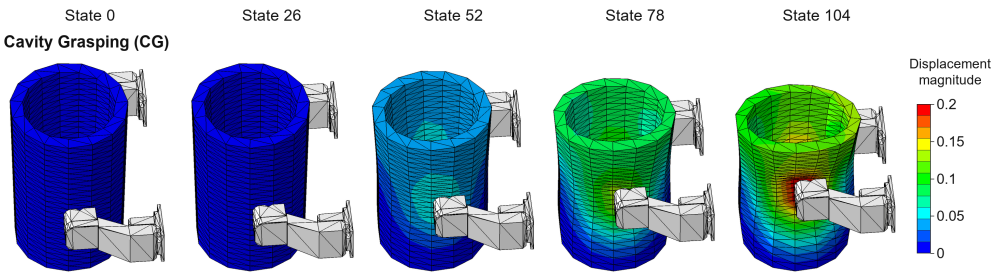


Figure 6: Visualization of the Cavity Grasping dataset.

Tissue Manipulation (TM). TM simulates elastic tissue manipulation motivated by robot-assisted surgery, as shown in Figure 7. One grasping point remains fixed while another follows a prescribed trajectory, deforming the tissue between the constrained and moving regions. Elastic properties vary across samples. In our representation, the tissue is discretized into 361 mesh vertices, and the fixed and moving grasping points are represented by two additional control nodes, giving 363 nodes in total. At each rollout step, the model uses the prescribed control motion to predict the subsequent tissue geometry. The task couples local deformation near the moving grasp with the response of tissue constrained by the stationary grasp, requiring coherent shape evolution throughout the manipulation sequence.

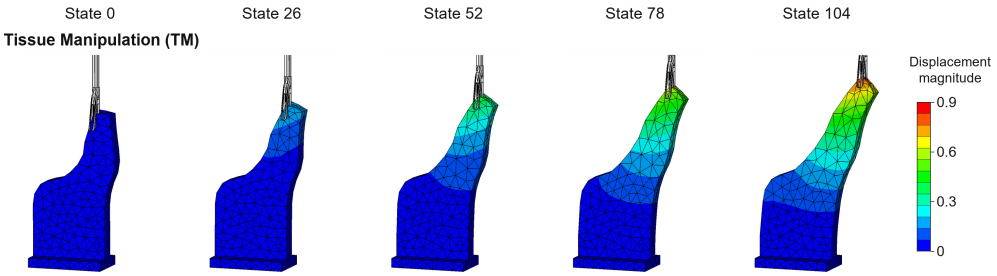


Figure 7: Visualization of the Tissue Manipulation dataset.

B Evaluation Metrics

We evaluate RATIO and all compared methods using positional root mean square error (RMSE) over autoregressive rollouts. For trajectory j at prediction step k, let $\hat{\mathbf{p}}_{j,i}^{k}$ and $\mathbf{p}_{j,i}^{k}$ denote the predicted and ground-truth positions of vertex i, respectively. The per-step error is

$$\mathrm{RMSE}_{j,k} = \sqrt{\frac{1}{|\mathcal{V}_j^{\mathrm{pred}}|} \sum_{i \in \mathcal{V}_j^{\mathrm{pred}}} \left\| \hat{\mathbf{p}}_{j,i}^{k} - \mathbf{p}_{j,i}^{k} \right\|_2^2}, \tag{13}$$

where $\mathcal{V}_j^{\mathrm{pred}}$ contains the vertices whose positions are predicted, excluding points with prescribed motion. The Euclidean norm measures the three-dimensional positional deviation of each vertex.

We average these per-step errors over all predicted steps and trajectories:

$$\mathcal{E}_{\mathrm{rollout}} = \frac{\sum_{j=1}^{M} \sum_{k=1}^{T_j} \mathrm{RMSE}_{j,k}}{\sum_{j=1}^{M} T_j}, \tag{14}$$

where M is the number of trajectories, and T_j is the number of prediction steps in trajectory j. Both CG and TM use $T_j = 104$, excluding the initial state.

Training and rollout evaluation. We train the model with single-step supervision by minimizing the mean squared error of normalized displacements, using ground-truth states perturbed with positional noise. This permits optimization over individual transitions without backpropagating through long autoregressive rollouts.

Sequence models can also be trained over multiple time steps using teacher forcing, which conditions each prediction on ground-truth previous states. However, this introduces a mismatch with autoregressive inference, where previous predictions become subsequent inputs. Errors can then shift the model away from the states encountered during training and accumulate over time (Bengio et al., 2015). Our single-step supervision likewise does not directly optimize errors accumulated along predicted trajectories, so a low training loss alone does not establish long-horizon accuracy.

We therefore perform validation and testing using 104-step autoregressive rollouts. Starting from the ground-truth initial configuration, the model recursively uses its predicted deformable positions, while prescribed motions remain available as boundary conditions. We select the checkpoint with the lowest validation rollout RMSE and report its test rollout RMSE. This evaluates performance under accumulated prediction errors and aligns model selection with the intended simulation setting.

C Experimental Details

Model Hyperparameters. We use fixed hidden widths and adjust model depths to control model size. Table 4 reports the configurations and parameter counts, with unused FFNs and their associated normalization layers excluded consistently across methods. These capacity controls, together with a common training and evaluation protocol, support a fair comparison.

Training Implementation. All models use the same data splits and a batch size of 13. We train for 400 epochs on CG and 600 epochs on TM using Adam (Kingma & Ba, 2014), with an initial learning rate of 3×10^{-4} and cosine annealing. Training minimizes the one-step MSE of normalized displacements. Checkpoints are selected by autoregressive validation RMSE and evaluated on the test set using the same rollout protocol.

Table 4: Model configurations and parameter counts for the current implementation on Cavity Grasping (CG) and Tissue Manipulation (TM). Counts include the encoder, processor, and decoder, excluding parameters unused by the forward computation.

		CG		TM	
Model	Width	Depth	Params (M)	Depth	Params (M)
MGN	128	10	2.615	10	2.615
GraphSAGE	128	7	2.575	7	2.575
GATv2	128	7	2.692	7	2.692
Transolver	128	14	2.602	14	2.602
GNOT	128	2	2.487	2	2.487
LSM	128	10	2.711	10	2.711
Unisoma	128	4	2.415	4	2.415
AeroGTO	128	5	2.776	5	2.776
HCMT	128	15	2.528	15	2.561
MGN-T	128	3	2.858	3	2.858
RATIO-only	96	9	2.075	9	2.075
Transolver + MGN	128	4	2.618	4	2.618
Transolver + GraphSAGE	128	5	2.797	5	2.797
Transolver + GATv2	128	5	2.880	5	2.880
LSM + MGN	128	5	2.730	5	2.730
LSM + GraphSAGE	128	6	2.466	6	2.466
LSM + GATv2	128	6	2.566	6	2.566
RATIO + MGN	96	6	2.195	6	2.195
RATIO + GraphSAGE	96	7	2.158	7	2.158
RATIO + GATv2	96	7	2.224	7	2.224

Depth denotes the architecture-specific processor depth; for local–global combinations, it counts composite blocks.

D Model Efficiency

We briefly discuss the computational efficiency of RATIO. Table 5 reports the inference and training runtimes of different models on CG and TM. RATIO incurs a higher inference cost than the evaluated conventional neural operators, but remains faster than fusion baselines such as MGN-T and HCMT, whereas RATIO+MGN achieves inference runtimes comparable to existing fusion models. Importantly, RATIO achieves substantially improved prediction accuracy at competitive inference costs, demonstrating a favorable accuracy–efficiency trade-off. Both configurations have higher training runtimes than most baselines. We attribute part of this overhead to the current implementation of world-edge slicing and deslicing. World-edge features are aggregated by batch index to accommodate variable-sized contact edge sets, while deslicing accumulates token contributions sequentially to avoid materializing a full edge–token–channel tensor during reconstruction. These operations introduce additional indexing and reduction overhead, making more efficient world-edge processing an important direction for future implementation optimization.

Table 5: Inference and training runtimes (ms) on CG and TM. Lower is better.

	CG		TM	
Model	Inference	Training	Inference	Training
MGN	24.44	61.49	20.05	54.81
Transolver	18.64	72.78	18.53	72.13
MGN-T	25.68	67.31	24.66	66.38
AeroGTO	27.25	63.23	21.02	51.99
HCMT	98.54	155.28	38.99	97.53
Unisoma	18.86	59.29	13.10	35.18
GNOT	11.31	27.91	8.45	25.23
LSM	20.62	55.59	18.82	56.62
GATv2	18.56	58.37	22.74	55.62
GraphSAGE	17.45	44.80	16.14	46.83
RATIO	22.30	90.19	21.82	83.76
RATIO+MGN	26.27	111.96	24.96	106.50

E Visualization

To examine the behavior of different models in greater detail, we present additional visualization examples beyond those in the main text.

Figure 8 shows three more trajectories from the Cavity Grasping (CG) dataset at multiple rollout states. Each model is visualized on its predicted geometry using a shared scale for nodal position error, enabling comparison of both geometric deviations and their spatial distribution. In these examples, Transolver develops spatially widespread errors and noticeable shape distortions as the rollout progresses. MGN exhibits a different pattern: errors remain relatively smaller near the grippers but become more pronounced in distant regions, particularly around the cavity opening and sidewalls. RATIO+MGN more closely follows the ground-truth deformation, with lower near-gripper and distant-region errors than MGN and MGN-T at the displayed states. Its improvement over RATIO varies across trajectories, illustrating that the benefit of combining the two components depends on the deformation sequence. These examples complement the aggregate rollout metrics by revealing how prediction errors develop and where they concentrate.

To further examine model behavior on Tissue Manipulation (TM), we present three selected trajectories at rollout steps 60, 80, and 104 in Figure 9. These examples exhibit pronounced bending of the upper tissue near the gripper, while the lower region remains comparatively stable. Under the shared position-error scale, MGN exhibits extensive regions of large error in the deforming upper tissue, with particularly pronounced error accumulation in trajectory 18. Transolver also shows noticeable deviations around the grasped and bent regions. MGN-T and standalone RATIO reduce these errors relative to MGN, although their relative accuracy varies across trajectories and rollout stages. RATIO+MGN more closely follows the ground-truth deformation and produces smaller errors around the bending region, achieving the lowest nodal position RMSE among the compared models in all nine displayed snapshots. These observations highlight its improved reconstruction of localized tissue deformation in the TM rollouts.

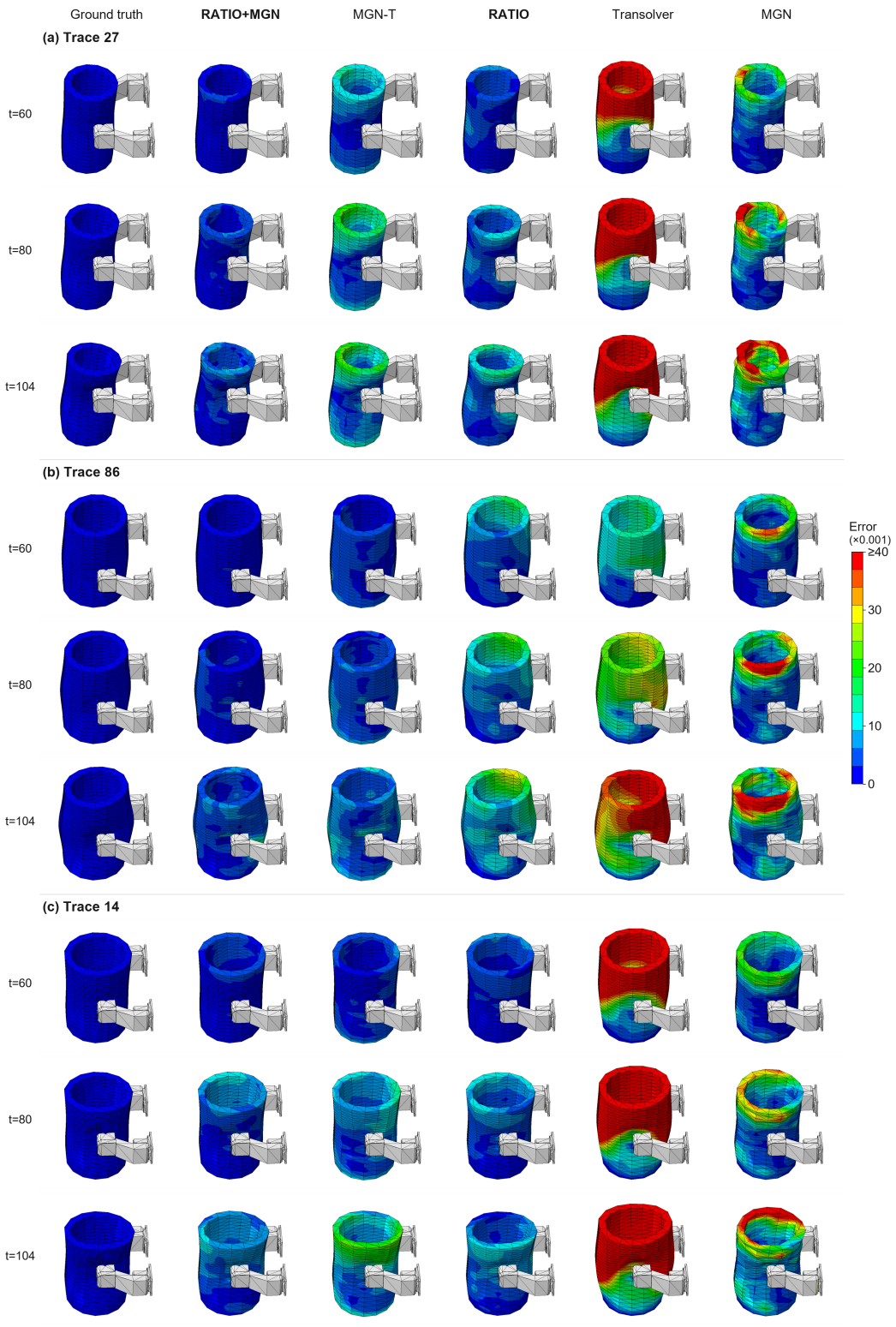


Figure 8: More qualitative comparison on Cavity Grasping dataset.

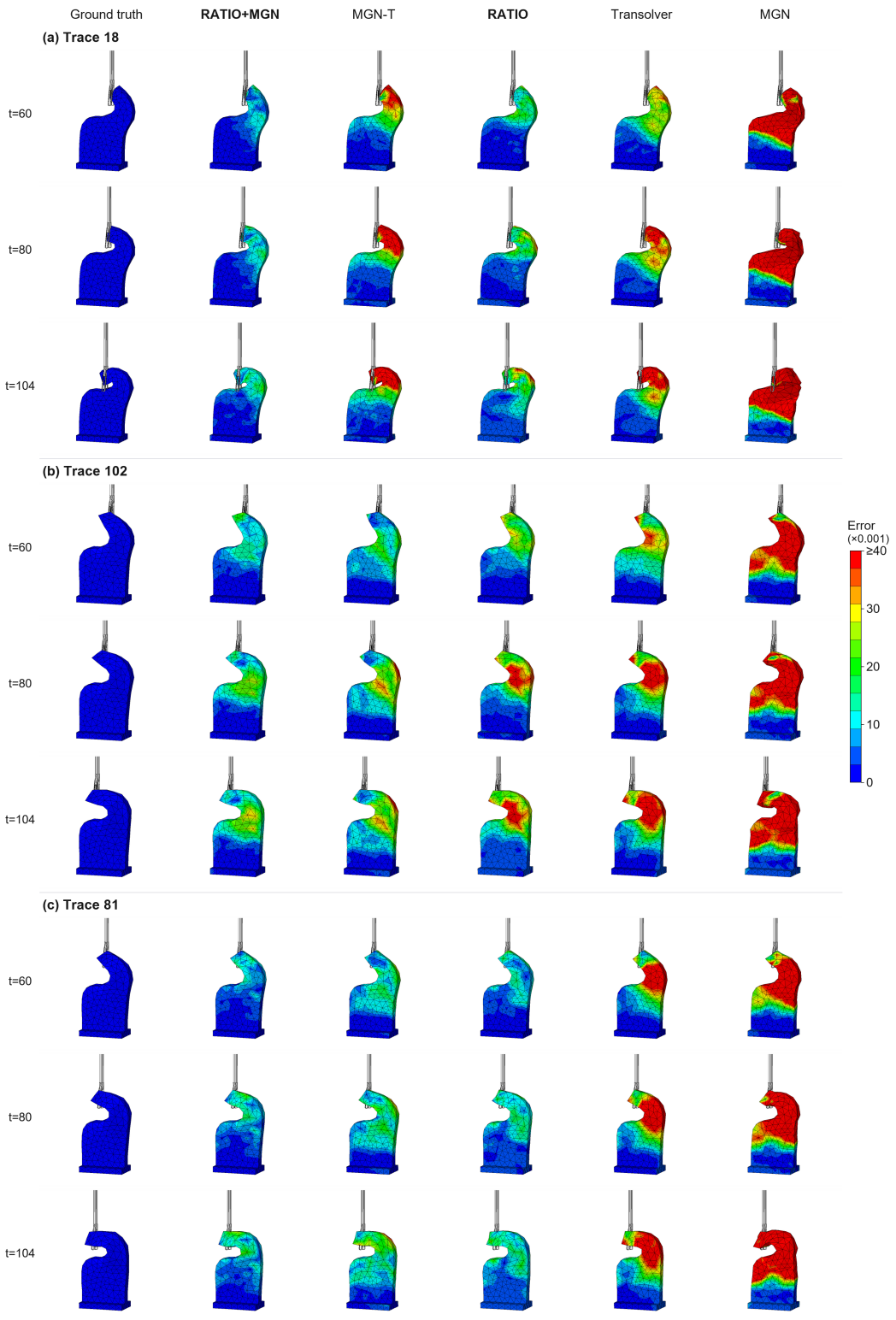


Figure 9: More qualitative comparison on Tissue Manipulation dataset.