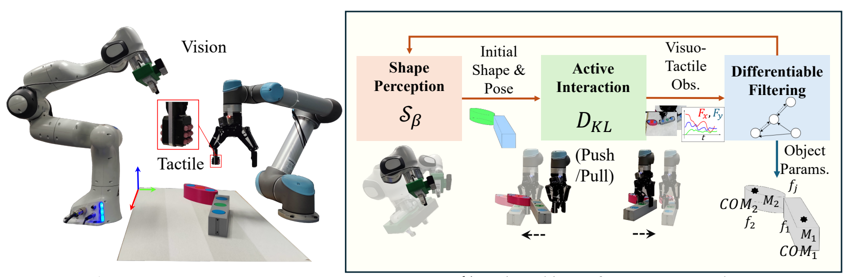
# Interactive Visuo-Tactile Learning to Estimate Properties of Articulated Objects

**Anonymous Author(s)**
Affiliation
Address
email

**Abstract:** Robotic systems operating in unstructured environments must infer key physical properties of objects, such as stiffness, mass, center of mass, friction, and shape, to ensure stable manipulation. Accurate estimation of these properties is crucial for predicting and effective planning manipulation outcomes. In this work, we present a novel framework for identifying the properties of challenging objects which are articulated through versatile, non-prehensile push-pull actions and using visuo-tactile observation. Our approach introduces a differentiable filtering method that incorporates embedding interaction physics into graph neural networks, enabling the system to actively learn object-robot interactions and consistently infer both directly observable pose information and indirectly observable physical parameters. Experimental results on real robotic systems show that our method outperforms existing baselines in efficiency and accuracy.

**Keywords:** Perception for Grasp & Manipulation, Visuo-Tactile Sensing, Active Learning, Interaction Dynamics




a) Interactive Perception Setup

b) Active Object Inference Framework

## 1 Introduction

Robotic systems engaged in contact-rich object manipulation tasks need to perceive the physical properties of the object, such as mass, center of mass, and surface friction, to perform effectively. However, these properties are difficult to estimate, as they are not directly observable in static environments and become salient only during specific object-robot interactions [1]. Current visual or tactile perception frameworks struggle to handle previously unseen objects [2, 3], necessitating the use of simple and robust interaction strategies to infer these physical properties prior to manipulation tasks [4, 5]. In this study, we propose a novel interactive learning and perception framework for inferring the properties of articulated objects using both vision and tactile sensing seamlessly using versatile push-pull interactions.

## 2 Related Work

Estimating inertial and surface properties of rigid objects is a long-standing problem in control theory, particularly for rigid body identification [6, 7]. The early methods relied on rigidly attached

Submitted to the 8th Conference on Robot Learning (CoRL 2024). Do not distribute.

objects to manipulators [8, 9], limiting their applicability in unstructured environments due to specialized mechanisms and known object geometry. Interactive techniques like grasping or pushing [10, 11, 12] tried to overcome these issues but relied on simplified assumptions. Recent research explores data-driven [13, 14, 15] and physics-based approaches [16], with studies [17, 18] showing the potential of graph networks to capture object-robot interactions. However, current GNNs focus on spatial relationships and kinematics but fail to capture contact forces influenced by physical properties and robot actions. This highlights the need for a graph-based model incorporating tactile information with a stronger inductive bias. Moreover, existing data-driven methods require extensive training and often lack strategic interaction, limiting their use to simulation environments. This motivates us to investigate possible active/informative interaction techniques [19, 20, 21], which is addressed in this work. Furthermore, prior research has relied mainly on visual [1, 22] or tactile [12, 11] methods to estimate physical properties, each with limitations. Tactile sensing can infer multiple properties of objects, but requires precise information and prior knowledge, while vision offers a limited range of observable properties, but provides a global view of the shape and movement of an object. Recent works [23, 24] combining vision and tactile approaches have shown improvements in pose estimation and contact-rich manipulation tasks. Building on these advances, our framework integrates sensing modalities with active exploratory actions: *non-prehensile pushing*, and *prehensile pulling* to enhance object perception. By encoding object-robot interactions into Probabilistic Markov Models and using a learned interaction model (differentiable filter [25, 22]), our system predicts visuo-tactile observations and estimates key physical parameters of articulated objects in a Bayesian Inference setting. The learned models capture not only the complex interaction dynamics but also modality-specific noise, improving the efficiency of inference.

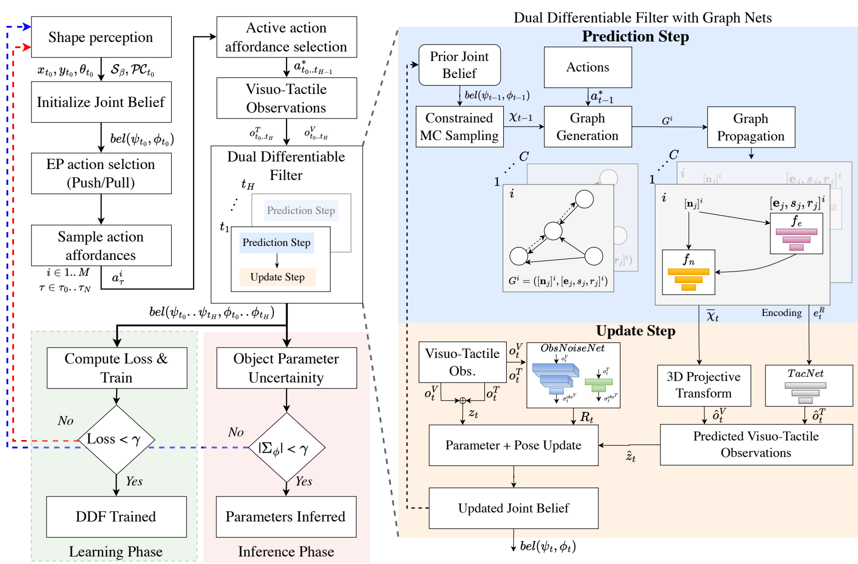



Figure 1: Proposed framework for interactively learning & inferring the properties of articulated objects using visuo-tactile sensing.

# 3 Methods

## 3.1 Problem Formulation

We tackle estimating the state $s$ of an unknown rigid object on a support surface using visual ($o^V$) and tactile ($o^T$) inputs along with actions ($a$). The objects are articulated, with multiple links connected through rotational joints. At time $t$, the state of the object $s_t$ is composed of $l \in 1, ..., L$ links, expressed as $s_t = \{s_t^1 ... s_t^L\}$. The state of each link $l$, $s_t^l = \{\psi_t^l, \phi^l\}$, includes time-varying elements: the 2D *pose* and *twist*, $\psi_t^l = \{x_t, y_t, \theta_t, v_{x_t}, v_{y_t}, \omega_t\}$, and time-invariant elements $\phi^l$, involving *inertial parameters* like $\{m, CoM_x, CoM_y\}$ mass and center of mass vector, and *interaction parameters* $\{f, f_r, f_j\}$ for friction with the table, robot, and adjacent link. The visual data $o_t^V$

includes RGB-D images of the robot-object interface, while the tactile data $o_t^T$ is 2D contact forces from the robotic gripper's interaction (fingertip forces). The push/pull action is defined by the tuple *contact point* $(cp)$, *direction* $(pd)$ and *velocity* $(u)$. In addition, for autonomous and seamless exploration of the object, the shape of each link $\mathcal{S}^l$ is estimated via superquadrics [26]. The belief about the current state of the object $s_t$ is represented by a distribution conditioned on previous actions $a_{1:t}$ and observations $o_{1:t}$ and employs recursive Bayesian filtering.

$$bel(s_t) = p(s_t|o_{1:t}, a_{1:t}) = \eta p(o_t|s_t, a_t) \int p(s_t|s_{t-1}, a_{t-1}) bel(s_{t-1}) ds_{t-1} \tag{1}$$

where $\eta$ is a normalizing factor. We employ a data-driven strategy to learn the process, observation, and noise models. Since object pose intricately relies on inertial and interaction parameters, joint filtering for pose and parameters [27] is found to be ineffective and we adopt a dual filter design to maintain consistent filtering and infer object parameters.

### 3.2 Dual Differentiable Filter

For the dual filter formulation, we explicitly represent the state of the object (joint distribution of pose and twist) via Multivariate Gaussian distribution:

$$bel(\psi_t, \phi_t) \doteq \mathcal{N}(\psi_t, \phi_t|\mu_t, \Sigma_t), \quad \mu_t = \begin{pmatrix} \mu_{\psi_t} \\ \mu_{\phi_t} \end{pmatrix}, \quad \Sigma_t = \begin{pmatrix} \Sigma_{\psi_t} & \Sigma_{\psi_t \phi_t} \\ \Sigma_{\phi_t \psi_t} & \Sigma_{\phi_t} \end{pmatrix} \tag{2}$$

with dimensions $\mu_t \in \mathbb{R}^{11L-1}$ and $\Sigma_t \in \mathbb{R}^{(11L-1)\times(11L-1)}$. The dual filter as shown in Fig.1 follows the structure of a Kalman filter with a *prediction step* and an *update step*, with the proposed novelty explained in this section.

#### 3.2.1 Prediction Step

In the prediction step, the next joint belief is predicted based on the prior belief and actions. Since the object's inertial and interaction parameters have physical constraints (e.g., $m, f, f_j > 0$, $CoM_x$, $CoM_y$ must lie within the object boundary), constrained Monte Carlo sigma point sampling is performed to maintain these constraints and the Gaussian variance. A differentiable sampling method [28] is used to sample $C$ sigma points $\chi_{t-1}^i, i = \{1..C\}$ from the joint distribution $bel(\psi_{t-1}, \phi_{t-1})$, with an associated weight $w_{t-1}^i = 1/C$.

We employ Graph Neural Networks (GNNs) to model the interaction between the object, the support surface, and the robot. Using the sigma points $\chi_{t-1}^i$ and the robot action $a_{t-1}$, a directed graph $G_t^i = (\{\mathbf{n}_l\}, \{\mathbf{e}_j, s_j, r_j\})$ is constructed, where $\mathbf{n}_l$ represents the nodes for each link of the object, the robot and the support surface, and $\mathbf{e}_j$ represents the directed edges. Each node $\mathbf{n}_l \in \mathbb{R}^{L+2}$ contains features including dynamic (pose, twist) and static (inertial) parameters, populated from the sigma points for the object links, with default values for the robot and surface. The edges $\mathbf{e}_j \in \mathbb{R}^{3L}$ capture the interaction between the links between the objects, the robot and the support surface, with features such as friction coefficients. To update node and edge features from time $t-1$ to $t$, we use a novel graph propagation algorithm (see the Appendix) with two functions: $f_n$ for node updates and $f_e$ for edge updates.

#### 3.2.2 Update Step

The dual filter employs a separate update of the parameter belief similar to the parameter update presented in [29] and the conditional pose belief update based on the UKF update [30]. To reduce the complexity of predicting raw RGB-D images, we use the initial segmented point cloud $\mathcal{PC}_{t_0}$ from the shape perception method to transform it using the predicted pose and generate expected RGB-D images using the standard 3D to 2D projective transformation approach [31] involving the intrinsic and extrinsic values of the camera, also avoiding generalization issues. For the tactile counterpart, a three-layer feedforward network is utilized to predict the contact force information from the edge encoding directed towards the robot. The filtering step is used end-to-end for both learning and inference.

### 3.2.3 Active Interaction: $N$-step Information Gain

To make the framework more sample efficient for real robot scenarios, we employ active action selection by formulating an $N$-step information gain criteria [32] under the filtering setting. We recursively use the prediction step of the dual differentiable filter without the update step to compute the expected Information Gain for both model learning and object parameter inference for each sampled non-prehensile pushing or prehensile pulling action $\pi^{[i]} = a^i_{\tau_0:\tau_N}$ over $N-$step in future $\tau = \tau_0..\tau_N$

$$IG_N(\pi^{[i]}) \approx -\mathbb{E}_{p(\psi_{\tau_N},\phi_{\tau_N}|\pi^{[i]})}[ln(\overline{bel}^{[i]}(\psi_{\tau_N},\phi_{\tau_N})) - ln(\overline{bel}^{[i]}(\psi_{\tau_0},\phi_{\tau_0}))] \tag{3}$$

where, $\overline{bel}^{[i]}(\psi_{\tau_N},\phi_{\tau_N})$ is the hypothetical predictive joint distribution after $N$-step by taking action $\pi^{[i]}$ without taking account the actual observation. At every step $\pi^* = \arg\max_{\pi^i} IG_N(\pi^{[i]})$ is selected for interaction.

# 4 Results & Conclusion

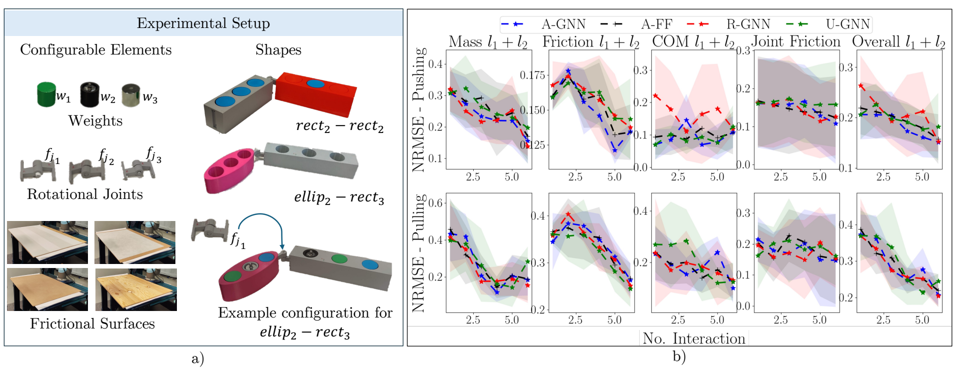



Figure 2: A) Experimental Setup with configurable objects B) Parameter estimation error across multiple interactions for articulated objects comparing proposed $A-GNN$ with $A-FF$, $R-GNN$ and $U-GNN$

| | Pulling | | Pushing | |
|---|---|---|---|---|
| | A | B | A | B |
| A-GNN | 2830 | $0.21 \pm 0.02$ | 4390 | $0.15 \pm 0.03$ |
| A-FF | 3410 | $0.22 \pm 0.04$ | 4810 | $0.16 \pm 0.05$ |
| R-GNN | 2922 | $0.21 \pm 0.03$ | 5295 | $0.15 \pm 0.02$ |
| U-GNN | 3405 | $0.25 \pm 0.02$ | 6000 | $0.18 \pm 0.07$ |

Table 1: Col. A) presents the no. of interactions required for training convergence, and B) presents $NRMSE$ of the overall parameter inference

We compare our proposed method, $A-GNN$, with the baseline $A-FF$ from [14] and conducted ablation studies to evaluate active action selection against uniform ($U-GNN$) and random ($R-GNN$) selection for model learning and inference. We designed 60 3D-printed articulated objects by varying weights, frictional surfaces, and joint friction (Fig.2a)). The networks were trained using negative log-likelihood ($\mathcal{L}_{NLL}$), mean squared error ($\mathcal{L}_{MSE}$), and observed noise log-likelihood ($\mathcal{L}^{obs}_{NLL}$). $\mathcal{L}_{NLL}$ and $\mathcal{L}_{MSE}$ compared ground truth and predicted poses, parameters, and forces, while $\mathcal{L}^{obs}_{NLL}$ was calculated using predicted and real observations. To account for different inertial and interaction parameters, we used normalized root mean squared error (NRMSE) to report estimation errors. Table 1.A shows that $A-GNN$ with active action selection improved data efficiency by 25% over uniform selection and 9% over $A-FF$, particularly for complex articulated objects and push interactions. Moreover, inference accuracy remains consistent with low SD and surpasses baseline methods (see Fig.2.B and Table 1.B).

Although this study considers a single object in isolation, future work will address more complex clutter scenarios and include interactive perception for prismatic-rotational joint identification. We also assumed the objects were planar and each articulated link was at most connected by two links. In conclusion, the proposed novel framework enables the robotic system to estimate the properties of intricate articulated objects autonomously using simple and efficient (active) interactive actions: non-prehensile push and prehensile pull.

## References


[1] Z. Xu, J. Wu, A. Zeng, J. B. Tenenbaum, and S. Song. Densephysnet: Learning dense physical object representations via multi-step dynamic interactions. *arXiv preprint arXiv:1906.03853*, 2019.

[2] N. Navarro-Guerrero et al. Visuo-haptic object perception for robots: an overview. *AuRo*, 47 (4):377–403, 2023.

[3] Q. Li et al. A review of tactile information: Perception and action through touch. *IEEE Trans. on Rob.*, 36(6):1619–1634, 2020. doi:10.1109/TRO.2020.3003230.

[4] P. Agrawal, A. V. Nair, P. Abbeel, J. Malik, and S. Levine. Learning to poke by poking: Experiential learning of intuitive physics. *Advances in neural information processing systems*, 29, 2016.

[5] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal, and G. S. Sukhatme. Interactive perception: Leveraging action in perception and perception in action. *IEEE Transactions on Robotics*, 33(6):1273–1291, 2017. doi:10.1109/TRO.2017.2721939.

[6] L. Ljung. *System Identification*, pages 163–173. Birkhäuser Boston, Boston, MA, 1998. ISBN 978-1-4612-1768-8. doi:10.1007/978-1-4612-1768-8_11. URL `https://doi.org/10.1007/978-1-4612-1768-8_11`.

[7] M. Niebergall and H. Hahn. Identification of the ten inertia parameters of a rigid body. *Nonlinear Dynamics*, 13:361–372, 1997.

[8] C. G. Atkeson, C. H. An, and J. M. Hollerbach. Estimation of inertial parameters of manipulator loads and links. *The International Journal of Robotics Research*, 5(3):101–119, 1986.

[9] C. Wang, X. Zang, X. Zhang, Y. Liu, and J. Zhao. Parameter estimation and object gripping based on fingertip force/torque sensors. *Measurement*, 179:109479, 2021.

[10] S. Tanaka, T. Tanigawa, Y. Abe, M. Uejo, and H. T. Tanaka. Active mass estimation with haptic vision. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 256–261. IEEE, 2004.

[11] K. Yao, M. Kaboli, and G. Cheng. Tactile-based object center of mass exploration and discrimination. In *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, pages 876–881, 2017. doi:10.1109/HUMANOIDS.2017.8246975.

[12] B. Sundaralingam and T. Hermans. In-hand object-dynamics inference using tactile fingertips. *IEEE Transactions on Robotics*, 37(4):1115–1126, 2021.

[13] C. Song and A. Boularias. A probabilistic model for planar sliding of objects with unknown material properties: Identification and robust planning. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5311–5318. IEEE, 2020.

[14] A. Dutta, E. Burdet, and M. Kaboli. Push to know!-visuo-tactile based active object parameter inference with dual differentiable filtering. In *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3137–3144. IEEE, 2023.

[15] A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, and P. Battaglia. Learning to simulate complex physics with graph networks. In *International conference on machine learning*, pages 8459–8468. PMLR, 2020.

[16] J. Wu, I. Yildirim, J. J. Lim, B. Freeman, and J. Tenenbaum. Galileo: Perceiving physical object properties by integrating a physics engine with deep learning. *Advances in neural information processing systems*, 28, 2015.

[17] F. Paus, T. Huang, and T. Asfour. Predicting pushing action effects on spatial object relations by learning internal prediction models. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10584–10590, 2020. doi:10.1109/ICRA40945.2020.9197295.

[18] A. E. Tekden, A. Erdem, E. Erdem, T. Asfour, and E. Ugur. Object and relation centric representations for push effect prediction. *Robotics and Autonomous Systems*, page 104632, 2024.

[19] J. Xu, H. Lin, S. Song, and M. Ciocarlie. Tandem3d: Active tactile exploration for 3d object recognition. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10401–10407. IEEE, 2023.

[20] T. Parr, G. Pezzulo, and K. J. Friston. *Active inference: the free energy principle in mind, brain, and behavior*. MIT Press, 2022.

[21] M. Kaboli, K. Yao, D. Feng, and G. Cheng. Tactile-based active object discrimination and target object search in an unknown workspace. *Autonomous Robots*, 43:123–152, 2019.

[22] A. Kloss, M. Bauza, J. Wu, J. B. Tenenbaum, A. Rodriguez, and J. Bohg. Accurate vision-based manipulation through contact reasoning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6738–6744. IEEE, 2020.

[23] P. K. Murali, A. Dutta, M. Gentner, E. Burdet, R. Dahiya, and M. Kaboli. Active visuo-tactile interactive robotic perception for accurate object pose estimation in dense clutter. *IEEE Robotics and Automation Letters*, 7(2):4686–4693, 2022. doi:10.1109/LRA.2022.3150045.

[24] M. A. Lee, Y. Zhu, K. Srinivasan, P. Shah, S. Savarese, L. Fei-Fei, A. Garg, and J. Bohg. Making sense of vision and touch: Self-supervised learning of multimodal representations for contact-rich tasks. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8943–8950. IEEE, 2019.

[25] R. Jonschkowski, D. Rastogi, and O. Brock. Differentiable particle filters: End-to-end learning with algorithmic priors. *arXiv preprint arXiv:1805.11122*, 2018.

[26] W. Liu, Y. Wu, S. Ruan, and G. S. Chirikjian. Robust and accurate superquadric recovery: A probabilistic approach. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2676–2685, 2022.

[27] A. Kloss, G. Martius, and J. Bohg. How to train your differentiable filter. *Autonomous Robots*, 45(4):561–578, 2021.

[28] M. Wüthrich, C. G. Cifuentes, S. Trimpe, F. Meier, J. Bohg, J. Issac, and S. Schaal. Robust gaussian filtering using a pseudo measurement. In *2016 American Control Conference (ACC)*, pages 3606–3613. IEEE, 2016.

[29] J. Liu and M. West. Combined parameter and state estimation in simulation-based filtering. In *Sequential Monte Carlo methods in practice*, pages 197–223. Springer, 2001.

[30] S. Thrun. Probabilistic robotics. *Communications of the ACM*, 45(3):52–57, 2002.

[31] I. Nematollahi, E. Rosete-Beas, S. M. B. Azad, R. Rajan, F. Hutter, and W. Burgard. T3vip: Transformation-based 3D video prediction. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4174–4181, 2022. doi:10.1109/IROS47612.2022.9981187.

[32] K. Friston. The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11 (2):127–138, 2010.

# 5 Appendix

## 5.1 Graph Propogation Algorithm

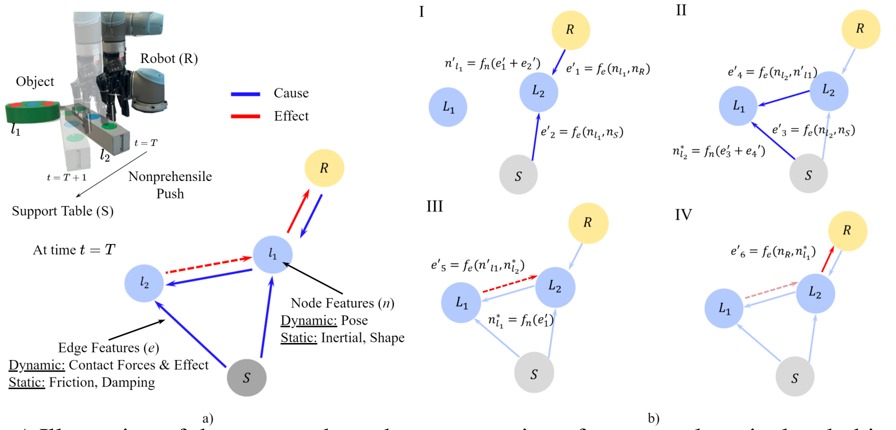



Figure 3: a) Illustration of the proposed graph representation of an example articulated object with two links b) Novel graph propagation for updating the graphical model from time $t-1$ to $t$ for the example object. The support edges $e_1, e_2$, the edge $e_6$ contains contact force or tactile information

---

**Algorithm 1** Graph Propagation Algorithm ($GP$)

**Input**: Graph $G_{t-1} = (\{\mathbf{n}_i\}, \{\mathbf{e}_j, s_j, r_j\})$

Initialize Stacks (LIFO)

$NTV \longleftarrow \mathbf{n}_R$ ▷ Nodes to visit

$NV \longleftarrow \emptyset$ ▷ Nodes visited

$EN \longleftarrow \emptyset$ ▷ End nodes

*Propagate cause*

**while do** $NTV \neq \emptyset$

  $\mathbf{n}_i$ = Pop $NTV$

  $\mathbf{n}_{r_j}$ = Gather receiver nodes of $\mathbf{n}_i$

  $\mathbf{n}_{r_j} = \mathbf{n}_{r_j} \setminus NV$ ▷ Remove nodes already visited

  **if** $\mathbf{n}_{r_j} \neq \emptyset$ **then**

    Push $\mathbf{n}_i \rightarrow NV$

    Push $\mathbf{n}_{r_j} \rightarrow NTV$

    **for** each node $\mathbf{n}_{r_j}$ **do**

      Compute causal edges, $\mathbf{e}^*_j = f_e(\mathbf{n}_i, \mathbf{n}_{r_j}, \mathbf{e}_{s_j})$

       ▷ $\mathbf{e}_{s_j}$ is static edge feature (friction values)

      Compute support edges, $\mathbf{e}^*_k = f_e(\mathbf{n}_S, \mathbf{n}_{r_j}, \mathbf{e}_{s_k})$

      Compute node features, $\mathbf{n}^*_i = f_n(\mathbf{n}_i, \mathbf{e}^*_j + \mathbf{e}^*_k)$

    **end for**

  **else**

    Push $\mathbf{n}_i \rightarrow EN$

  **end if**

**end while**

*Propagate effect*

**while do** $NV \neq \emptyset$

  $\mathbf{n}_i$ = Pop $NV$

  $\mathbf{n}^*_{s_j}$ = Gather sender nodes of $\mathbf{n}_i$

  $\mathbf{n}^*_{s_j} = \mathbf{n}^*_{s_j} \setminus NV$

  Aggregate effect edges, $\mathbf{e}^*_j = f_e(\mathbf{n}^*_{s_j}, \mathbf{n}_i, \mathbf{e}_{s_j})$

  Update node features, $\mathbf{n}^*_i = f_n(\mathbf{n}_i, \sum_{j/s_j} \mathbf{e}^*_j)$

**end while**

**Output**: Graph $G_t = (\{\mathbf{n}^*_i\}, \{\mathbf{e}^*_j, s_j, r_j\})$

---