# Inductive Knowledge Graph Wave Networks

**Giuseppe Pirrò**
DeMaCS, University of Calabria
giuseppe.pirro@unical.it

## Abstract

Inductive knowledge graph reasoning (IKGR) requires generalization to entities and relation types not seen during training. We introduce Knowledge Graph Wave Networks (KGWN), an IKGR approach that learns adaptive per-relation wave operators to control relation-dependent propagation, along with velocity normalization ensuring stability at any propagation depth. Experiments demonstrate state-of-the-art results on inductive benchmarks and competitive transductive performance.

## 1 Introduction

Knowledge graphs (KGs) empower various applications [Hogan *et al.*, 2021]. A central challenge is learning KG representations that generalize not only within a fixed graph (*transductive* setting), but also to entities and relations unseen during training (*inductive* setting) [Lee *et al.*, 2023]. Consider a KG where `Oppenheimer` is an *unseen entity* at training time. To answer the query $(\texttt{Oppenheimer}, \texttt{directed\_by}, ?)$, a model cannot rely on memorized entity embeddings; instead, it must infer the missing director by propagating evidence through relation patterns (e.g., shared collaborators). Likewise, for $(\texttt{Oppenheimer}, \texttt{genre}, ?)$, evidence can flow through thematic connections (e.g., films), enabling genres prediction. Inductive KG reasoning (IKGR) thus demands *relation-dependent propagation*: selectively gathering evidence through heterogeneous, directed relations while preserving distinct semantic signals. This is challenging under *relational heterophily* [Qian *et al.*, 2025], where homophilic relations like `same_genre` connect same-type entities and benefit from low-frequency alignment, while heterophilic relations like `directed_by` connect different entity types requiring contrast-preserving propagation for type distinctions.

**Related Work.** GraIL [Teru *et al.*, 2020], NBFNet [Zhu *et al.*, 2021], SNRI [Xu *et al.*, 2022], MulGA [Shang *et al.*, 2024], and RED-GNN [Zhang and Yao, 2022] reason over query-conditioned local subgraphs to accommodate unseen entities under known relations. IKGR-ME [Cui *et al.*, 2022] extends entity induction to multi-batch emerging entities. InGram [Lee *et al.*, 2023] supports both new entities and relations at test time. All these methods rely on discrete, neighbor-centric aggregation that is typically biased toward low-pass smoothing, lacking an explicit mechanism to control when low-frequency alignment versus contrast-preserving high-frequency behavior should be emphasized. As depth increases, this can lead to oversmoothing that washes out relational cues crucial for inductive inference [Qian *et al.*, 2025; He *et al.*, 2025]—particularly problematic for heterophilic relations requiring sharp cross-type distinctions. Diffusion models like DiffusionE [Cao *et al.*, 2024] treat propagation as continuous-time heat flow, progressively smoothing node signals; however, diffusion behaves primarily as a low-pass filter, blurring signals in heterophilic relations. GWN [Yue *et al.*, 2025] and EWGNN [Wu *et al.*, 2025] offer conservative alternatives that maintain high-frequency signals, but both are limited to *homogeneous, transductive* settings.

| Method | KG | Entity Induction | Relation Induction | Heterophily-aware Propagation | Stability Guarantees |
|---|---|---|---|---|---|
| GraIL (ICML'20) | ✓ | ✓ | ✗ | ✗ | ✗ |
| NBFNet (NeurIPS'21) | ✓ | ✓ | ✗ | ✗ | ✗ |
| SNRI (IJCAI'22) | ✓ | ✓ | ✗ | ✗ | ✗ |
| RED-GNN (WebConf'22) | ✓ | ✓ | ✗ | ✗ | ✗ |
| IKGR-ME (CIKM'22) | ✓ | ✓ | ✗ | ✗ | ✗ |
| InGram (ICML'23) | ✓ | ✓ | ✓ | ✗ | ✗ |
| MulGA (IJCAI'24) | ✓ | ✓ | ✗ | ✗ | ✗ |
| GRAND (NeurIPS'21) | ✗ | ✗ | ✗ | ✗ | ✓ |
| DiffusionE (KDD'24) | ✓ | ✓ | ✗ | ✗ | ✗ |
| GWN (WebConf'25) | ✗ | ✗ | ✗ | ✗ | ✓ |
| EWGNN (IJCAI'25) | ✗ | ✗ | ✗ | ✗ | ✓ |
| **KGWN (Ours)** | ✓ | ✓ | ✓ | ✓ | ✓ |

Table 1: Comparison of KGWN to relevant approaches.

**Our approach.** Table 1 shows that existing approaches do not explicitly combine stable wave-based propagation with relation-dependent frequency control in the full IKGR setting. Thus, we introduce Knowledge Graph Wave Networks (KGWN), defining relation-specific propagation channels in which each relation type induces a learnable *wave operator* with two key parameters: (i) *propagation speeds* controlling signal strength and reach, and (ii) *frequency gates* that bias relations toward low-frequency alignment or contrast-preserving propagation. Since mode parameters are learned per relation, they transfer directly to unseen entities. For unseen relations, KGWN computes relation representations using a *relation graph*.

An extended version of the paper, including appendices, is available online at: https://github.com/giuseppepirro/KGWN

**Contributions and outline.** We contribute: **(1)** a multi-relational wave equation for directed KGs using symmetric operators (§3.1) and velocity normalization ensuring bounded propagation at any depth with uniform stability guarantees (§3.2); **(2)** relation-adaptive frequency control explicitly addressing relational heterophily (§3.3); and **(3)** a wave-based framework for inductive KG reasoning that generalizes to unseen entities *and* relations (§4).

## 2 Preliminaries

A knowledge graph is a tuple $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$, where $\mathcal{V}$ contains $N$ entities, $\mathcal{R}$ is a set of relation types, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ contains directed triples.

### 2.1 Knowledge Graph Reasoning Settings

We focus on *link prediction*, which operates under two broad settings: (i) *transductive*, where all entities and relations at test time were present during training, and the model predicts missing edges among known elements; and (ii) *inductive*, where the model must generalize to new entities and/or relations without retraining. We distinguish three inductive cases. In *entity-level induction*, test triples contain unseen entities but relations are drawn from $\mathcal{R}_{\text{train}}$. In *relation-level induction*, test triples may contain unseen relations, while the protocol provides enough support information or textual metadata to embed those relations at test time. In *full induction*, test triples may contain both unseen entities and unseen relations; KGWN assumes that unseen elements have textual metadata and are inserted into the corresponding entity or relation graph at test time without gradient updates.

Embedding initializations. We initialize entity and relation embeddings using pretrained text encoders [Sanh *et al.*, 2019] applied to their textual names and descriptions:

$$\mathbf{X}^{(0)} = \text{TextEnc}\big(\{\text{name}_v, \text{desc}_v\}_{v\in\mathcal{V}}\big),$$
$$\mathbf{R}^{(0)} = \text{TextEnc}\big(\{\text{name}_r, \text{desc}_r\}_{r\in\mathcal{R}}\big).$$

## 3 Knowledge Graph Wave Networks

We now introduce *Knowledge Graph Wave Networks* (KGWN) for inductive knowledge graph reasoning. Fig. 1 gives an overview. After initial embeddings, KGWN defines symmetric wave operators with controlled spectrum to enable multi-relational wave propagation (§3.1), then multi-relational aggregation with stability guarantees (§3.2), and finally relation-specific frequency adaptation via *relation gates*, allowing the model to learn whether each relation should bias propagation toward low-frequency alignment or contrast-preserving behavior (§3.3).

### 3.1 Symmetric Multi-Relational Operators

The standard wave equation $\ddot{\mathbf{X}}(t) = \mathbf{L}\mathbf{X}(t)$ propagates signals using a *single symmetric* graph operator $\mathbf{L}$, yielding wave-like dynamics that can preserve high-frequency components and mitigate oversmoothing on homogeneous graphs [Yue *et al.*, 2025]. In contrast, knowledge graphs are *multi-relational* and *directed*, with edges typed by relations $r \in \mathcal{R}$. To retain the stability and spectral control afforded by wave dynamics, KGWN assigns each relation its own *symmetric* propagation operator $\mathbf{L}_r$, and combines them into a controlled multi-relational operator for propagation. To handle directionality, we define two KGWN variants.

**1. KGWN-S (Symmetric Channel).** For each relation $r \in \mathcal{R}$, $\tilde{\mathbf{A}}_r := \mathbf{A}_r + \mathbf{A}_r^\top$ is its symmetrized adjacency, where $\mathbf{A}_r$ is the directed adjacency matrix containing edges of type $r$. Let $\mathbf{D}_r = \text{diag}(\tilde{\mathbf{A}}_r\mathbf{1})$ denote the corresponding degree matrix. We construct the normalized operator

$$\mathbf{P}_r = \mathbf{D}_r^{-1/2}\,\tilde{\mathbf{A}}_r\,\mathbf{D}_r^{-1/2}, \qquad \mathbf{L}_r := \mathbf{P}_r - \mathbf{I}. \tag{1}$$

For zero-degree nodes under relation $r$, the corresponding entries of $\mathbf{D}_r^{-1/2}$ are set to zero. Symmetrization treats each triple $(u, r, v)$ as bidirectional, pooling information from both head-to-tail and tail-to-head. Each $\mathbf{L}_r$ is symmetric with spectrum $\text{spec}(\mathbf{L}_r) \subseteq [-2, 0]$: symmetrization yields a symmetric normalized adjacency $\mathbf{P}_r$ with $\text{spec}(\mathbf{P}_r) \subseteq [-1, 1]$, and the shift $\mathbf{L}_r = \mathbf{P}_r - \mathbf{I}$ moves the spectrum to $[-2, 0]$ (Lemma 1 Appendix A). This makes the continuous-time dynamics $\ddot{\mathbf{X}} = \mathbf{L}\mathbf{X}$ oscillatory rather than exponentially growing, and provides a spectral bound used for stability.

**2. KGWN-L (Lifted Channel).** Symmetrization discards directional information, which may be semantically important (e.g., `parent_of` vs. `child_of`). To preserve direction while maintaining symmetry, we use a lifted construction. For each relation $r$, we construct the adjacency

$$\mathbf{B}_r := \begin{bmatrix} \mathbf{0} & \mathbf{A}_r \\ \mathbf{A}_r^\top & \mathbf{0} \end{bmatrix} \in \mathbb{R}^{2N\times 2N}, \tag{2}$$

which encodes directed edges through role-specific block structure: the upper-right block contains forward source-to-target edges, while the lower-left block contains the corresponding reverse role edges. Let $\mathbf{D}_r^{(\text{lift})} = \text{diag}(\mathbf{B}_r\mathbf{1})$ be the degree matrix for this lifted graph. We then define

$$\mathbf{P}_r^{(\text{lift})} = (\mathbf{D}_r^{(\text{lift})})^{-1/2}\,\mathbf{B}_r\,(\mathbf{D}_r^{(\text{lift})})^{-1/2}, \quad \mathbf{L}_r^{(\text{lift})} := \mathbf{P}_r^{(\text{lift})} - \mathbf{I}. \tag{3}$$

Again, zero-degree entries in the inverse square-root degree matrix are set to zero. Consider a simple example with two entities $\{E_1, E_2\}$ and one directed edge $(E_1, r, E_2)$ (e.g., $r$ = `parent_of`). The lifted graph duplicates each entity into forward and backward copies: $\{E_{1,f}, E_{1,b}, E_{2,f}, E_{2,b}\}$. The block matrix $\mathbf{B}_r$ encodes directed edges via its blocks: (i) in the upper-right block, $E_{1,f} \to E_{2,b}$ represents the original source-to-target role; and (ii) in the lower-left block, $E_{2,f} \to E_{1,b}$ represents the reverse role induced by $\mathbf{A}_r^\top$. Although propagation on the lifted graph is symmetric, the two partitions encode different head/tail roles, so directionality is preserved through role-specific states rather than through an asymmetric operator. After propagation on this $2N$-node bipartite graph, the final embedding for entity $E_i$ fuses the two context-specific states $E_{i,f}$ and $E_{i,b}$ via a fusion layer. Despite encoding directed edges, $\mathbf{B}_r$ is symmetric by construction, so $\mathbf{L}_r^{(\text{lift})}$ inherits the same spectral properties as the symmetric variant: spectrum in $[-2, 0]$, enabling stable wave propagation. The lifted construction operates on state vectors $\mathbf{X}^{(n)} \in \mathbb{R}^{2N\times d}$ that maintain separate representations for each entity in forward and backward contexts (see Lemma 2 in Appendix A).

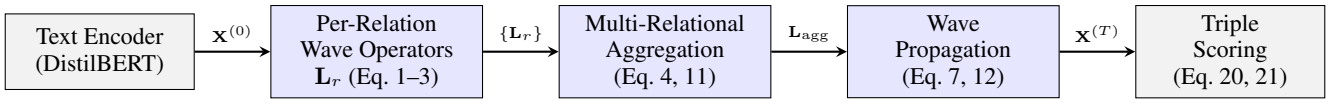


Figure 1: KGWN pipeline overview.

In what follows, we use $\mathbf{L}_r$ to denote the relation operator under either construction. We next show how to combine per-relation operators into a unified wave equation.

### 3.2 Multi-Relational Aggregation and Stability

To combine the per-relation operators $\{\mathbf{L}_r\}_{r\in\mathcal{R}}$ into a single multi-relational wave equation, a natural choice would be a weighted sum $\mathbf{L}_{\text{agg}} = \sum_{r\in\mathcal{R}} w_r \mathbf{L}_r$. The key issue is stability: even if each $\mathbf{L}_r$ is individually well behaved, unconstrained weights can increase the spectral radius of $\mathbf{L}_{\text{agg}}$, which in turn can make the update in Eq. (7) unstable. We therefore need a condition, and a normalization, that controls $\text{spec}(\mathbf{L}_{\text{agg}})$. Let $\{a_r > 0\}_{r\in\mathcal{R}}$ be learnable relation velocities controlling each relation's propagation strength. We define the aggregated operator

$$\mathbf{L}_{\text{agg}} = \sum_{r\in\mathcal{R}} a_r^2 \mathbf{L}_r, \tag{4}$$

yielding the multi-relational wave equation

$$\ddot{\mathbf{X}}(t) = \mathbf{L}_{\text{agg}} \mathbf{X}(t). \tag{5}$$

The squared velocities $a_r^2$ ensure non-negativity, which is essential for spectral control: since each $\mathbf{L}_r$ has spectrum in $[-2, 0]$ and velocities are nonnegative, Rayleigh quotient bounds (Theorem 1, Appendix A) give

$$\text{spec}(\mathbf{L}_{\text{agg}}) \subseteq [-2\nu^2, 0], \quad \text{where} \quad \nu^2 = \sum_{r\in\mathcal{R}} a_r^2. \tag{6}$$

To implement wave propagation computationally, we discretize Eq. (5) in time using the *leapfrog* scheme [Hairer *et al.*, 2006]. The second derivative is approximated by a centered finite difference, leading to the update

$$\mathbf{X}^{(n+1)} = \left(2\mathbf{I} + \tau^2 \mathbf{L}_{\text{agg}}\right) \mathbf{X}^{(n)} - \mathbf{X}^{(n-1)}, \tag{7}$$

where $\tau > 0$ is the time step and $\mathbf{X}^{(n)} \approx \mathbf{X}(n\tau)$ denotes the state at discrete time $n\tau$. Unless otherwise stated, we use zero initial velocity, implemented as $\mathbf{X}^{(-1)} = \mathbf{X}^{(0)}$; the same convention is used for relation-space propagation.

**Stability note.** The leapfrog update in Eq. (7) is second-order accurate in $\tau$ and has good long-time energy behavior, but it is stable only if every eigenmode stays bounded. Let $\lambda$ be an eigenvalue of $\mathbf{L}_{\text{agg}}$ and consider the corresponding scalar mode $x^{(n)}$. Substituting $x^{(n)} = \rho^n$ into Eq. (7) gives the characteristic equation $\rho^2 - (2 + \tau^2\lambda)\rho + 1 = 0$. Boundedness is equivalent to the standard condition (Theorem 2, Appendix B):

$$|2 + \tau^2\lambda| \le 2 \qquad \text{for all } \lambda \in \text{spec}(\mathbf{L}_{\text{agg}}). \tag{8}$$

Using Eq. (6), the worst case is the most negative eigenvalue $\lambda = -2\nu^2$. Plugging this into Eq. (8) yields

$$\tau^2\nu^2 < 2, \tag{9}$$

which means that the time step $\tau$ must be small enough to resolve the fastest oscillations. Rather than hand-tuning $\tau$ or manually constraining the learned velocities, we enforce Eq. (9) via a global rescaling:

$$\tilde{a}_r = \frac{a_r}{\max\left(1,\ \frac{\tau\nu}{\sqrt{2-\delta_v}}\right)}, \tag{10}$$

where $\delta_v \in (0, 1)$ is a small safety margin and $\nu = \sqrt{\sum_{r\in\mathcal{R}} a_r^2}$. Let $\tilde{\nu}^2 = \sum_{r\in\mathcal{R}} \tilde{a}_r^2$ denote the post-rescaling aggregate velocity. By construction, $\tau^2\tilde{\nu}^2 \le 2 - \delta_v$, so every eigenmode remains bounded for any propagation depth. We denote the post-normalization velocities by $\tilde{a}_r$ for clarity, and use these rescaled values in all subsequent propagation steps. For notational simplicity in later equations, we write $a_r$ with the understanding that it refers to the normalized quantity $\tilde{a}_r$ from Eq. (10). See Appendix B for detailed derivations.

While stability prevents numerical blow-up, it does not address a key KG challenge: relations are heterogeneous and may require different spectral responses, such as low-frequency alignment for homophilic relations and contrast preservation for heterophilic ones.

### 3.3 Relation-Level Frequency Adaptation

The aggregated operator in Eq. (4) weights relations by strength via the learned velocities $\{a_r\}$, but it does not allow different relations to exhibit different spectral behaviors. Hence, we introduce *relation gates* that separate (i) *importance* and (ii) *frequency behavior*. Let $\beta_r \ge 0$ with $\sum_{r\in\mathcal{R}} \beta_r = 1$ be learned importance weights (e.g., $\beta_r = \text{softmax}(\gamma_r)$), and let $\alpha_r \in [-1, 1]$ be signed frequency gates. Positive gates preserve stable low-frequency alignment, while negative gates invert the spectral response and induce contrast-enhancing dynamics associated with high-frequency components. We retain the learned velocities $a_r > 0$ from Eq. (4) to control coupling strength. The adaptive aggregated operator becomes

$$\tilde{\mathbf{L}}_{\text{agg}} := \sum_{r\in\mathcal{R}} \beta_r\, \alpha_r\, a_r^2\, \mathbf{L}_r. \tag{11}$$

Allowing $\alpha_r < 0$ can flip the sign of some terms and introduce positive eigenvalues (Lemma 3, Appendix C). Positive eigenvalues in $\tilde{\mathbf{L}}_{\text{agg}}$ induce exponential growth in the corresponding continuous-time modes, and can destabilize the discrete update. Constraining $\alpha_r \ge 0$ would eliminate this issue but would also remove the model's ability to learn contrast-enhancing propagation along heterophilic relations. We therefore stabilize signed-gate dynamics using two mechanisms: operator rescaling and anchoring toward the initialization. Let

$$\kappa := \sum_{r\in\mathcal{R}} \beta_r\, |\alpha_r|\, a_r^2.$$

Since $\|\mathbf{L}_r\|_2 \leq 2$ for both constructions (Appendix C), we have $\|\tilde{\mathbf{L}}_{\text{agg}}\|_2 \leq 2\kappa$. We define the rescaled adaptive operator

$$\widehat{\mathbf{L}}_{\text{agg}} := \eta\, \tilde{\mathbf{L}}_{\text{agg}}, \qquad \eta = \min\left(1, \frac{\varepsilon - \delta_a}{2\tau^2\kappa}\right), \tag{12}$$

with anchoring strength $\varepsilon \in [0, 1)$ and margin $\delta_a \in (0, \varepsilon)$. This ensures $\tau^2\|\widehat{\mathbf{L}}_{\text{agg}}\|_2 \leq \varepsilon - \delta_a$. The anchored update is

$$\mathbf{X}^{(n+1)} = \left(2\mathbf{I} + \tau^2\widehat{\mathbf{L}}_{\text{agg}}\right)\mathbf{X}^{(n)} - \mathbf{X}^{(n-1)} + \varepsilon\big(\mathbf{X}^{(0)} - \mathbf{X}^{(n)}\big). \tag{13}$$

For any eigenvalue $\mu$ of $\widehat{\mathbf{L}}_{\text{agg}}$, the scalar mode coefficient is $2 - \varepsilon + \tau^2\mu$. Since $|\mu| \leq \|\widehat{\mathbf{L}}_{\text{agg}}\|_2$, the rescaling condition implies $-\varepsilon < \tau^2\mu < \varepsilon$, and hence $2 - 2\varepsilon < 2 - \varepsilon + \tau^2\mu < 2$. Therefore the leapfrog boundedness condition is satisfied for $\varepsilon \in [0, 1)$ (Theorem 3 Appendix C). The parameters have distinct roles: $\beta_r$ controls *how much* relation $r$ contributes overall, while $\alpha_r$ controls *how* it acts spectrally. If we restrict $\alpha_r \in [0, 1]$, then $\tilde{\mathbf{L}}_{\text{agg}}$ remains negative semidefinite, and leapfrog stability follows the condition in §3.2. When allowing signed gates $\alpha_r \in [-1, 1]$, the anchor in Eq. (13) stabilizes modes induced by positive eigenvalues while retaining contrast-enhancing expressiveness.

# 4 Inductive Reasoning

KGWN works in both entity-inductive and full scenarios.

## 4.1 Entity Induction

Consider a link prediction query $(h, r, ?)$ or $(?, r, t)$, where $h$ or $t$ may be unseen during training. Rather than relying on entity-specific embeddings, KGWN performs reasoning over the test-time graph structure. For a candidate tail $t$, KGWN extracts the $k$-hop enclosing subgraph around the pair $(h, t)$:

$$\mathcal{V}_{h,t}^{(k)} = \mathcal{N}_k(h) \cup \mathcal{N}_k(t),$$

where $\mathcal{N}_k(v)$ denotes the set of nodes within $k$ hops of $v$ in the test graph $\mathcal{G}_{\text{test}}$, computed on the undirected view for extraction. During evaluation, the target test triple and its inverse are removed before subgraph extraction to avoid label leakage. The induced subgraph $\mathcal{G}_{h,t}^{(k)} = (\mathcal{V}_{h,t}^{(k)}, \mathcal{E}_{h,t}^{(k)}, \mathcal{R})$ is then the domain for wave propagation, with

$$\mathcal{E}_{h,t}^{(k)} = \{(u, r', v) \in \mathcal{E}_{\text{test}} : u, v \in \mathcal{V}_{h,t}^{(k)}\}.$$

The crucial observation is that KGWN's dynamics depend only on relation types and textual/structural features, not on memorized entity identities. On the extracted subgraph, we construct per-relation operators $\mathbf{L}_r^{(k;h,t)}$ following the same recipe as Eqs. (1)–(3), then aggregate:

$$\mathbf{L}_{\text{agg}}^{(k;h,t)} = \sum_{r \in \mathcal{R}} a_r^2\, \mathbf{L}_r^{(k;h,t)}. \tag{14}$$

The relation parameters $\{a_r\}$, and optionally $\{\alpha_r, \beta_r\}$, come directly from training—no fine-tuning required. Wave propagation proceeds via the standard leapfrog update:

$$\mathbf{X}^{(n+1)} = \big(2\mathbf{I} + \tau^2\mathbf{L}_{\text{agg}}^{(k;h,t)}\big)\mathbf{X}^{(n)} - \mathbf{X}^{(n-1)}, \tag{15}$$

with zero initial velocity and initial states obtained from the text encoder. Since text embeddings are available for any entity with a name or description, unseen elements can be encoded without retraining.

## 4.2 Relation and Full Induction

When test queries involve unseen relation types, entity induction alone does not suffice. KGWN addresses this by propagating wave dynamics over a *relation graph* $\mathcal{G}_{\text{rel}} = (\mathcal{R}, \mathcal{E}_{\text{rel}})$, where edges connect relations that co-occur in triples or have sufficiently similar textual descriptions (Appendix D). In our implementation, relation-graph edges are created when relation pairs exceed a co-occurrence threshold $\theta_c$ or a text-similarity threshold $\theta_s$. For unseen relations, text similarity provides the default connection mechanism; when the benchmark provides support triples, co-occurrence edges are computed only from support information and never from held-out target triples. We initialize relation embeddings $\mathbf{R}^{(0)} \in \mathbb{R}^{|\mathcal{R}| \times d_r}$ from text and propagate via the same leapfrog scheme with normalized Laplacian $\mathbf{L}_{\text{rel}}$:

$$\mathbf{R}^{(m+1)} = \big(2\mathbf{I} + \tau_{\text{rel}}^2\mathbf{L}_{\text{rel}}\big)\mathbf{R}^{(m)} - \mathbf{R}^{(m-1)}. \tag{16}$$

We use zero initial velocity, $\mathbf{R}^{(-1)} = \mathbf{R}^{(0)}$, and choose or rescale $\tau_{\text{rel}}$ analogously to satisfy leapfrog stability. After $M$ propagation steps, learnable projections $(\mathbf{w}_\cdot, b_\cdot)$ decode each embedding $\mathbf{r}_r^{(M)}$ into the wave parameters:

$$\begin{aligned} a_r &= \text{softplus}(\mathbf{w}_a^\top \mathbf{r}_r^{(M)} + b_a), \\ \alpha_r &= \tanh(\mathbf{w}_\alpha^\top \mathbf{r}_r^{(M)} + b_\alpha), \\ \beta_r &= \text{softmax}_r(\mathbf{w}_\beta^\top \mathbf{r}_r^{(M)} + b_\beta). \end{aligned} \tag{17}$$

The activations enforce the constraints in §3.3: softplus ensures $a_r > 0$, tanh bounds $\alpha_r \in (-1, 1)$, and softmax normalizes the mixing weights $\{\beta_r\}$ across relations. For unseen relations $r_{\text{new}}$, we insert their text embeddings into $\mathcal{G}_{\text{rel}}$, connect them to similar known relations using the above criterion, rerun Eq. (16), and decode parameters via Eq. (17) without gradient updates. These inferred parameters drive propagation in the query subgraph:

$$\tilde{\mathbf{L}}_{\text{agg}}^{(k;h,t)} = \sum_{r \in \mathcal{R}_{h,t}} \tilde{\beta}_r\, \alpha_r\, a_r^2\, \mathbf{L}_r^{(k;h,t)}, \tag{18}$$

where $\mathcal{R}_{h,t}$ denotes relations in the subgraph and $\tilde{\beta}_r$ renormalizes the mixing weights over $\mathcal{R}_{h,t}$. Entity embeddings then evolve via the adaptive update in Eq. (13).

**Scoring.** For *link prediction* in the entity induction setting, we use MLP scoring to match DiffusionE [Cao *et al.*, 2024]:

$$s(h, r, t) = \text{MLP}\Big(\big[\mathbf{x}_h^{(T)} \,\|\, \mathbf{x}_t^{(T)} \,\|\, (\mathbf{x}_h^{(T)} \odot \mathbf{x}_t^{(T)}) \,\|\, \mathbf{r}_r^{(M)}\big]\Big). \tag{19}$$

For the full inductive setting, we adopt the bilinear scoring function to match InGram [Lee *et al.*, 2023]:

$$s(h, r, t) = {\mathbf{x}_h^{(T)}}^\top \text{diag}(\mathbf{W}\mathbf{r}_r^{(M)})\, \mathbf{x}_t^{(T)}, \tag{20}$$

where the diagonal matrix weights each embedding dimension according to the relation, enabling efficient yet expressive head–tail matching.

## 4.3 Training

For *entity induction*, each positive triple $(h, r, t) \in \mathcal{T}^+$ is paired with $K$ corrupted triples $\mathcal{T}^-$ obtained by replacing the

head or tail. We minimize the binary cross-entropy:

$$\mathcal{L}_{\text{ent}} = -\sum_{(h,r,t)\in\mathcal{T}^+}\Bigg[\log\sigma\big(s(h,r,t)\big) + \frac{1}{K}\sum_{(h',r,t')\in\mathcal{T}^-}\log\big(1-\sigma(s(h',r,t'))\big)\Bigg]. \tag{21}$$

In the *full inductive* setting, following InGram [Lee *et al.*, 2023], we train with margin $\gamma > 0$ to rank candidates:

$$\mathcal{L}_{\text{f}} = \sum_{(h,r,t)\in\mathcal{T}^+}\frac{1}{K}\sum_{(h',r,t')\in\mathcal{T}^-}\max\big(0,\gamma - s(h,r,t) + s(h',r,t')\big). \tag{22}$$

## 5 Experimental Evaluation

We evaluate KGWN across inductive/transductive settings:

- **Main Results:** Entity induction (RQ1), relation induction (RQ2), and transductive reasoning (RQ3)
- **Analysis:** Component ablations and hyperparameter sensitivity (RQ4).

**Evaluation protocol.** We evaluate link prediction in the filtered setting [Bordes *et al.*, 2013]. For each test triple $(h, r, t)$, we rank tail candidates $(h, r, t')$ and head candidates $(h', r, t)$ for all $t', h' \in \mathcal{V}_{\text{test}}$. For each candidate, we extract its $k$-hop enclosing subgraph (default $k = 2$), propagate with KGWN, and compute the score. We train KGWN using Adam (lr $5 \times 10^{-4}$), batch size 128, 50 epochs with early stopping (patience 3) on validation MRR. Default hyperparameters ($d$=128, $k$=2, $T$=3, $\tau$=0.05, $\varepsilon$=0.2, $\delta$=0.15) are identical across all benchmarks without per-dataset tuning (full table in Appendix E); sensitivity is analysed in §5.4. Text features use DistilBERT [Sanh *et al.*, 2019]. Implementation details including Algorithm 1 (pseudocode) and complexity analysis are in Appendix D. We report ranking metrics Hits@1/10 and MRR. Code is available online*.

The two settings use different datasets because they follow distinct community protocols: entity splits (V1–V4) were introduced by [Teru *et al.*, 2020]; NELL/Wikidata68K/Freebase overlap splits were introduced by [Lee *et al.*, 2023] as the only available benchmarks with disjoint unseen-relation test sets, which the entity-induction datasets do not provide.

### 5.1 Entity-Level Induction (RQ1)

We used the FB15k-237, WN18RR, and NELL-995 benchmarks and splits by [Teru *et al.*, 2020]. We compare against recent KG-based methods evaluated on the same splits: GraIL [Teru *et al.*, 2020], SNRI [Xu *et al.*, 2022], NBFNet [Zhu *et al.*, 2021], RED-GNN [Zhang and Yao, 2022] (R-GNN), MulGA [Shang *et al.*, 2025], and DiffusionE [Cao *et al.*, 2024] (DiffE). For conciseness, we omit baselines that underperformed based on the recent evaluation by [Cao *et al.*, 2024]. Table 2 reports entity-inductive link prediction across four increasingly difficult splits per benchmark.

*https://github.com/giuseppepirro/KGWN

| Method | WN18RR | | | | FB15k-237 | | | | NELL-995 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | V1 | V2 | V3 | V4 | V1 | V2 | V3 | V4 | V1 | V2 | V3 | V4 |
| **MRR** | | | | | | | | | | | | |
| GraIL | .627 | .625 | .323 | .553 | .279 | .276 | .251 | .227 | .481 | .297 | .322 | .262 |
| SNRI | .576 | .539 | .305 | .486 | .186 | .227 | OM | OM | .363 | .200 | OM | OM |
| NBFNet | .741 | .704 | .452 | .641 | .422 | .514 | .476 | .453 | .584 | .410 | .425 | .287 |
| R-GNN | .701 | .690 | .427 | .651 | .369 | .469 | .445 | .442 | .637 | .419 | .436 | .363 |
| MulGA | .747 | .705 | .476 | .655 | .485 | .538 | .492 | .472 | .812 | .525 | .510 | .429 |
| DiffE | .727 | .724 | .460 | .660 | .310 | .474 | .456 | .446 | .673 | .424 | .458 | .308 |
| KGWN-S | .738 | .726 | .469 | .668 | .328 | .489 | .468 | .458 | .702 | .448 | .475 | .332 |
| **KGWN-L** | **.761** | **.732** | **.491** | **.678** | **.502** | **.556** | **.511** | **.489** | **.831** | **.542** | **.528** | **.447** |
| **Hits@1** | | | | | | | | | | | | |
| GraIL | 55.4 | 54.2 | 27.8 | 44.3 | 20.5 | 20.2 | 16.5 | 14.3 | 42.5 | 19.9 | 22.4 | 15.3 |
| SNRI | 47.9 | 44.8 | 23.4 | 39.9 | 10.7 | 14.4 | OM | OM | 18.5 | 12.8 | OM | OM |
| NBFNet | 69.5 | 65.1 | 39.2 | 60.8 | 33.5 | 42.1 | 38.4 | 36.0 | 50.0 | 27.1 | 26.2 | 23.3 |
| R-GNN | 65.3 | 63.3 | 36.8 | 60.6 | 30.2 | 38.1 | 35.1 | 34.0 | 52.5 | 31.9 | 34.5 | 25.9 |
| MulGA | 70.8 | 65.2 | 42.4 | 61.1 | 40.2 | 44.4 | 39.2 | 37.2 | 75.0 | 42.5 | 42.2 | 34.1 |
| DiffE | 68.5 | 67.8 | 41.0 | 61.5 | 22.8 | 38.2 | 36.8 | 35.4 | 59.2 | 32.8 | 36.5 | 24.6 |
| KGWN-S | 69.8 | 68.2 | 41.8 | 62.4 | 24.5 | 39.6 | 37.8 | 36.2 | 62.5 | 35.2 | 38.4 | 26.8 |
| **KGWN-L** | **72.4** | **68.6** | **44.8** | **64.2** | **42.5** | **46.8** | **42.1** | **39.5** | **77.3** | **44.8** | **44.6** | **36.5** |
| **Hits@10** | | | | | | | | | | | | |
| GraIL | 76.0 | 77.6 | 40.9 | 68.7 | 42.9 | 42.4 | 42.4 | 38.9 | 56.5 | 49.6 | 51.8 | 50.6 |
| SNRI | 75.0 | 69.8 | 43.2 | 65.2 | 33.2 | 38.8 | OM | OM | 55.5 | 33.3 | OM | OM |
| NBFNet | 82.6 | 79.8 | 56.8 | 69.4 | 57.4 | 68.5 | 63.7 | 62.7 | 79.5 | 63.5 | 60.6 | 59.1 |
| R-GNN | 79.9 | 78.0 | 52.4 | 72.1 | 48.3 | 62.9 | 60.3 | 62.1 | 86.6 | 60.1 | 59.4 | 55.6 |
| MulGA | 82.7 | 80.1 | 56.0 | 73.6 | 62.2 | 70.7 | 66.2 | 65.2 | 93.5 | 70.9 | 67.9 | 57.1 |
| DiffE | 85.7 | 84.3 | 60.5 | 73.7 | 52.5 | 66.1 | 62.7 | 62.5 | 87.3 | 66.0 | 62.0 | 52.3 |
| KGWN-S | 85.9 | 84.5 | 61.2 | 74.5 | 54.2 | 67.8 | 64.2 | 63.8 | 89.5 | 68.2 | 64.5 | 54.8 |
| **KGWN-L** | **86.4** | **85.1** | **62.3** | **76.8** | **64.5** | **72.4** | **68.9** | **67.8** | **95.2** | **73.2** | **70.5** | **59.8** |

Table 2: Entity-inductive link prediction. V1-V4 denote increasingly difficult test splits with decreasing entity overlap with training. Results for competitors are from *Cao et al., 2024*; MulGA from *Shang et al., 2024*. Best results in bold. KGWN results are averaged over 3 runs with 3 seeds; standard deviations and significance tests are in Appendix E.

KGWN-L achieves state-of-the-art on all configurations, while KGWN-S consistently outperforms DiffusionE. DiffusionE's first-order diffusion inherently smooths signals, losing discriminative power on heterophilic relations, whereas KGWN's second-order wave equation preserves high-frequency structure through oscillatory propagation. On WN18RR-V3, KGWN-L improves MRR by $+3.1$ pp over DiffusionE (.491 vs .460), showing that taxonomic relations benefit from bidirectional wave flow. KGWN-L's learnable per-relation parameters ($a_r$, $\alpha_r$) enable adaptive smoothing/sharpening, yielding MRR improvement over DiffusionE on V1 (.502 vs .310) and consistently outperforming MulGA across all splits. NELL-995 is extracted via noisy web extraction. KGWN-L achieves .831 MRR on V1, surpassing MulGA (.812) by leveraging text-initialized features that provide semantic grounding. The gap widens on harder splits ($+0.018$ points on both V3 and V4), where wave dynamics better separate signal from noise and KGWN scales gracefully while SNRI runs out-of-memory. KGWN-S offers a favorable accuracy-efficiency trade-off, matching or exceeding DiffusionE. Unlike DiffusionE's learned semantic modules, KGWN's wave operators are derived directly from adjacency structure, reducing learnable parameters to a few scalars per relation. On the hardest V3/V4 splits with sparser test subgraphs, KGWN-L improves over the next-best method by $+1.5$ pp and $+2.3$ pp, respectively.

| Method | NELL | | | | Wikidata | | | | Freebase | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 100 | 75 | 50 | 25 | 100 | 75 | 50 | 25 | 100 | 75 | 50 | 25 |
| **MRR** | | | | | | | | | | | | |
| GraIL | .135 | .096 | .162 | .216 | – | – | – | – | – | – | – | – |
| NBFNet | .096 | .137 | .225 | .283 | .014 | .072 | .062 | .154 | .072 | .089 | .130 | .224 |
| R-GNN | .212 | .203 | .179 | .214 | .096 | .172 | .058 | .170 | .121 | .107 | .129 | .145 |
| InGram | .309 | .261 | .281 | .334 | .107 | .247 | .068 | .186 | .223 | .189 | .117 | .133 |
| KGWN-S | .318 | .268 | .289 | .345 | .114 | .255 | .075 | .193 | .230 | .196 | .126 | .141 |
| **KGWN-L** | **.342** | **.291** | **.312** | **.371** | **.128** | **.278** | **.091** | **.215** | **.251** | **.218** | **.148** | **.168** |
| **Hits@1** | | | | | | | | | | | | |
| GraIL | .114 | .036 | .104 | .160 | – | – | – | – | – | – | – | – |
| NBFNet | .032 | .077 | .161 | .224 | .005 | .028 | .036 | .092 | .026 | .048 | .071 | .137 |
| R-GNN | .114 | .129 | .115 | .166 | .070 | .110 | .033 | .111 | .053 | .057 | .072 | .077 |
| InGram | .212 | .167 | .193 | .241 | .072 | .179 | .034 | .124 | .146 | .119 | .067 | .067 |
| KGWN-S | .219 | .174 | .200 | .250 | .079 | .186 | .041 | .131 | .153 | .126 | .075 | .076 |
| **KGWN-L** | **.241** | **.196** | **.223** | **.278** | **.095** | **.208** | **.058** | **.154** | **.176** | **.148** | **.096** | **.101** |
| **Hits@10** | | | | | | | | | | | | |
| GraIL | .173 | .205 | .288 | .366 | – | – | – | – | – | – | – | – |
| NBFNet | .199 | .255 | .346 | .417 | .026 | .172 | .105 | .301 | .154 | .166 | .259 | .410 |
| R-GNN | .385 | .353 | .280 | .266 | .136 | .290 | .093 | .263 | .263 | .201 | .251 | .284 |
| InGram | .506 | .464 | .453 | .501 | .169 | .362 | .135 | .309 | .371 | .325 | .218 | .271 |
| KGWN-S | .517 | .475 | .464 | .514 | .179 | .373 | .145 | .320 | .382 | .337 | .230 | .283 |
| **KGWN-L** | **.548** | **.506** | **.495** | **.547** | **.201** | **.398** | **.168** | **.348** | **.412** | **.368** | **.261** | **.315** |

Table 3: Relation-inductive results with different % relations shared with training. Results for the competitors are taken from *Lee at al., 2023*. KGWN results are averaged over 3 runs; see Appendix E for variance and significance analysis.

## 5.2 Relation-Level Induction (RQ2)

We evaluate now *both* unseen relations and entities at test time. We use the benchmarks from InGram [Lee *et al.*, 2023] spanning NELL [Xiong *et al.*, 2017], Wikidata68K [Gesese *et al.*, 2022], and Freebase [Bollacker *et al.*, 2008], with varying overlap (100%, 75%, 50%, 25%). We compare against GraIL [Teru *et al.*, 2020], NBFNet [Zhu *et al.*, 2021], RED-GNN [Zhang and Yao, 2022], and InGram [Lee *et al.*, 2023]. Table 3 shows that KGWN-L achieves state-of-the-art across all configurations. InGram pioneered relation-level induction via relation graph construction; KGWN builds on this foundation but replaces static message passing with wave-based dynamics. On NELL-25 (only 25% relations seen), KGWN-L improves MRR by $+11.1\%$ over InGram (.371 vs .334), showing that oscillatory propagation captures higher-order relational interactions missed by static aggregation. As known relations decrease (100→25), most methods collapse: NBFNet's MRR on Freebase jumps erratically (.072→.224), revealing brittleness. KGWN-L degrades gracefully, maintaining consistent improvements over InGram ($+11.1\%$ on NELL, $+15.6\%$ on Wikidata, $+26.3\%$ on Freebase at 25-split). On heterogeneous Wikidata and Freebase, where GraIL fails entirely and NBFNet/R-GNN struggle below .20 MRR, KGWN-L achieves .278 MRR on Wikidata-75 and .251 on Freebase-100 ($+12.6\%$ over InGram for both), confirming that relation-specific wave parameters ($a_r$, $\alpha_r$) adapt to diverse semantics without overfitting.

## 5.3 Transductive Reasoning (RQ3)

Table 4 reports transductive reasoning results. On *WN18RR*, DiffusionE achieves .577 MRR, the best prior method. KGWN-L improves to .598 MRR (+3.6% relative improvement), demonstrating that second-order wave propagation extracts complementary structural patterns even when entity embeddings are fully trained. The improvement is particularly evident in Hits@10 (69.8 vs 67.2, +2.6 pp). On *FB15k-237*, with 237 diverse relations, the best baseline is MulGA at .422 MRR. KGWN-L achieves .448 MRR (+6.2% relative), with per-relation wave parameters adapting propagation dynamics to each relation's homophilic or heterophilic profile. Hits@10 improves by 3.4 pp (63.4 vs 60.0). On *NELL-995*, MulGA leads prior methods at .564 MRR. KGWN-L reaches .589 MRR (+4.4% relative improvement) and 69.6 Hits@10 (+2.7 pp over MulGA's 66.9), confirming wave-based reasoning's effectiveness even in this noisy, web-extracted graph.

| Method | WN18RR | | | FB15k-237 | | | NELL-995 | | |
|---|---|---|---|---|---|---|---|---|---|
| | MRR | H@1 | H@10 | MRR | H@1 | H@10 | MRR | H@1 | H@10 |
| NBFNet | .551 | 49.7 | 66.6 | .415 | 32.1 | 59.9 | .525 | 45.1 | 63.9 |
| MulGA | .573 | 52.3 | 67.1 | .422 | 33.2 | 60.0 | .564 | 50.4 | 66.9 |
| RED-GNN | .533 | 48.5 | 62.4 | .374 | 28.3 | 55.8 | .543 | 47.6 | 65.1 |
| DiffusionE | .577 | 52.4 | 67.2 | .376 | 29.4 | 53.9 | .552 | 49.0 | 65.4 |
| KGWN-S | .583 | 53.2 | 68.1 | .428 | 33.5 | 60.8 | .568 | 50.8 | 67.5 |
| **KGWN-L** | **.598** | **54.8** | **69.8** | **.448** | **35.2** | **63.4** | **.589** | **52.8** | **69.6** |

Table 4: Transductive link prediction results. Results for competitors are taken from *Cao et al., 2024* while for MulGA are from *Shang et al., 2024*. Best results are in bold. KGWN results are averaged over 3 runs (see Appendix E).

## 5.4 Ablation Study (RQ4)

We analyze KGWN's components on FB15k-237 (entity-inductive, V2) unless otherwise noted.

Wave dynamics. Replacing the second-order wave equation with first-order diffusion ($\dot{\mathbf{X}} = \mathbf{L}\mathbf{X}$) drops MRR by $-7.9\%$, as diffusion smooths signals and loses discriminative power on heterophilic relations. A static 1-hop GNN baseline ($-14.0\%$) confirms that iterative propagation is essential, but wave dynamics capture patterns diffusion misses (Table 5a).

Directional encoding: KGWN-S (symmetric) underperforms KGWN-L (lifted) by 12.1% MRR. The lifted construction preserves asymmetric semantics (e.g., `hypernym` vs. `hyponym`) that symmetrization conflates, with gaps widening on taxonomic-heavy WN18RR (15.3%).

Frequency adaptation: fixing $\alpha_r = 1$ (all smoothing) costs 6.3% MRR; uniform importance weights cost 3.2%. Removing both degrades performance by 10.4%, confirming that learning relation-specific smoothing/sharpening is critical.

Stability mechanisms: without velocity normalization (Eq. 10), training becomes unstable after 3–4 layers (13.1% MRR drop). The anchor term ($\varepsilon$) prevents drift from negative gates; removing it costs 5.8% (Table 5a).

Hyperparameter sensitivity. KGWN is robust with graceful degradation outside optimal regions (Table 5b). Specifically: (1) $\tau$ too large violates stability (Eq. 9), causing numerical instability; (2) $\varepsilon$ too small provides insufficient regularization for negative gates, while too large over-constrains dynamics; (3) $T = 1$ underexplores multi-hop patterns, while $T > 4$ yields diminishing returns. Optimal values ($\tau = 0.05$, $\varepsilon = 0.2$, $T = 3$) work across all datasets.

Frequency gates initialization. Starting from zero (neutral) outperforms biased initializations (Table 5c), allowing the

**(a) Component Ablation**

| Variant | MRR | H@1 | H@10 | Δ |
|---|---|---|---|---|
| **KGWN-L (full)** | **.556** | **46.8** | **72.4** | – |
| *Wave dynamics* | | | | |
| Wave → Diffusion | .512 | 42.1 | 68.5 | −7.9 |
| Wave → Static | .478 | 38.4 | 64.2 | −14.0 |
| *Directional encoding* | | | | |
| Lifted → Symmetric | .489 | 39.6 | 67.8 | −12.1 |
| *Frequency adaptation* | | | | |
| No freq. gates | .521 | 43.2 | 69.8 | −6.3 |
| Uniform $\beta_r$ | .538 | 44.5 | 70.6 | −3.2 |
| Both removed | .498 | 40.8 | 67.1 | −10.4 |
| *Stability mechanisms* | | | | |
| No velocity norm | .483 | 38.9 | 65.4 | −13.1 |
| No anchor | .524 | 43.8 | 69.2 | −5.8 |

**(b) Hyperparameters**

| $\tau$ | | $\varepsilon$ | | $T$ | |
|---|---|---|---|---|---|
| 0.01 | .538 | 0.05 | .542 | 1 | .498 |
| 0.02 | .548 | 0.10 | .548 | 2 | .538 |
| **.05** | **.556** | **.20** | **.556** | **3** | **.556** |
| 0.10 | .551 | 0.30 | .552 | 4 | .552 |
| 0.20 | .542 | 0.50 | .545 | 5 | .545 |

**(c) Gate Initialization**

| $\alpha_r$ Initialization | MRR | Ep. |
|---|---|---|
| Positive (+0.5) | .538 | 42 |
| Negative (−0.5) | .524 | 48 |
| Random $\mathcal{U}(-1, 1)$ | .552 | 35 |
| **Zero (0)** | **.556** | **32** |

**(d) Learned Parameters per Dataset**

| Dataset | %Sharp. | %Smooth. | $\lvert\alpha\rvert$ | $a^2$ | $\beta$ |
|---|---|---|---|---|---|
| WN18RR | 62 | 38 | .71 | .82 | .18 |
| FB15k-237 | 48 | 52 | .58 | 1.05 | .08 |
| NELL-995 | 41 | 59 | .52 | 1.38 | .12 |

**(e) Computational Cost (FB15k-237 V2)**

| Variant | Time | Mem | Par. | MRR |
|---|---|---|---|---|
| Baseline | 22.4s | 3.2G | 223K | .498 |
| +Freq. gates | 23.5s | 3.3G | 224K | .521 |
| +KGWN-S | 28.8s | 3.8G | 224K | .489 |
| +KGWN-L | 35.4s | 5.4G | 257K | **.556** |
| *Overhead* | 1.58× | 1.69× | 1.15× | +11.6% |

Table 5: Ablation study. (a) Component contributions; (b) hyperparameter sensitivity; (c) gate initialization; (d) learned parameters across datasets; (e) computational overhead. Relation graph construction ablation (thresholds $\theta_c$, $\theta_s$) is in Appendix E.

model to discover relation-specific behaviors from data. Interestingly, random initialization in $[-1, 1]$ performs comparably, suggesting the optimization landscape is well-behaved.

Anchor strength vs propagation depth. Fig. 2 shows MRR across different $(\varepsilon, T)$ combinations. Without anchoring ($\varepsilon = 0$), performance degrades at $T \geq 3$ when negative gates introduce positive eigenvalues. Weak anchoring ($\varepsilon = 0.1$) provides partial stabilization but underperforms at the optimal depth. Moderate anchoring ($\varepsilon = 0.2$) achieves the best trade-off, stabilizing wave propagation while preserving expressiveness. Strong anchoring ($\varepsilon = 0.5$) over-regularizes, pulling representations too strongly toward initialization and limiting the model's ability to learn discriminative features.

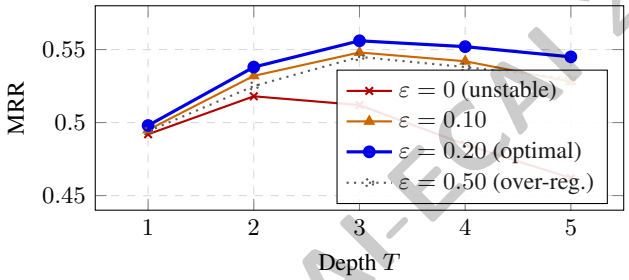


Figure 2: Anchor strength ($\varepsilon$) vs. propagation depth ($T$). Without anchoring, performance degrades at $T \geq 3$; moderate anchoring ($\varepsilon$=0.2) achieves the best trade-off.

Learned parameters. WN18RR learns the highest proportion of sharpening gates (62%), consistent with its heterophilic taxonomic structure (Table 5d). FB15k-237 shows balanced smooth/sharpen ratios reflecting its diverse relation types. NELL-995 learns higher velocities on average, compensating for its noisy extraction process. An additional analysis of per relation learned parameters is available in Appendix E.

Text encoder robustness. Replacing DistilBERT [Sanh *et al.*, 2019] with BERT-base drops MRR by only −0.5%; hash features (no semantic text) recover most of the gap over random initialisation, confirming wave propagation is the primary source of representational power. End-to-end fine-tuning is impractical: text embeddings are precomputed once and reused across all per-query subgraph passes (fine-tuning would require a full encoder call per candidate). Full ablation results are in Appendix E.

Computational efficiency. Table 5e analyzes the cost of KGWN components on FB15k-237 (V2: 1,660 entities, 237 base relations, 474 with inverses). The lifted construction (KGWN-L) adds ∼50% time overhead due to doubled state dimension ($2N \times d$) and requires 15% more parameters (34K) for the two-stream fusion layer. Wave operators $\mathbf{L}_r$ are parameter-free, constructed from graph structure (Eqs. 1–3). Frequency adaptation adds minimal overhead.

Per-query inference cost. Per-query complexity is $O(T \cdot |\mathcal{E}_{\text{sub}}| \cdot d)$ (Appendix D); with $k$=2 on sparse graphs, $|\mathcal{E}_{\text{sub}}| \ll |\mathcal{E}|$. Subgraph extractions are cached: cold extraction takes ∼20 ms on FB15k-237 V2, cache hits <0.2 ms. KGWN-L end-to-end latency is 0.42 s/query vs. 0.31 s for DiffusionE (full per-dataset breakdown in Appendix D); KGWN-S runs at comparable speed to DiffusionE.

## 6 Conclusions and Future Work

Knowledge graph relations require different propagation behaviors. KGWN addresses this with stable second-order wave dynamics and signed frequency gates, enabling relation-specific smoothing or contrast preservation. Experiments show strong performance across inductive and transductive benchmarks, and the same formulation naturally extends from entity to relation reasoning.

Limitations. KGWN-L doubles the state dimension, which can limit scalability on very large graphs. Performance also drops in sparse-subgraph regimes and many-to-many relations. Finally, text features are frozen; fine-tuning them may help, but would add substantial cost.

Future work. We plan to extend KGWN to temporal and multi-modal KGs.

## References

[Bollacker *et al.*, 2008] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase:

A collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, pages 1247–1250. ACM, 2008.

[Bordes *et al.*, 2013] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. *Advances in Neural Information Processing Systems*, 26, 2013.

[Cao *et al.*, 2024] Zongsheng Cao, Jing Li, Zigan Wang, and Jinliang Li. DiffusionE: Reasoning on knowledge graphs via diffusion-based graph neural networks. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 222–230. ACM, 2024.

[Chamberlain *et al.*, 2021] Benjamin Paul Chamberlain, James Rowbottom, Maria I Gorinova, Michael Bronstein, Stefan Webb, and Emanuele Rossi. GRAND: Graph neural diffusion. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 1407–1418. PMLR, 2021.

[Cui *et al.*, 2022] Yuanning Cui, Yuxin Wang, Zequn Sun, Wenqiang Liu, Yiqiao Jiang, Kexin Han, and Wei Hu. Inductive knowledge graph reasoning for multi-batch emerging entities. In Mohammad Al Hasan and Li Xiong, editors, *Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022*, pages 335–344. ACM, 2022.

[Gesese *et al.*, 2022] Genet Asefa Gesese, Harald Sack, and Mehwish Alam. RAILD: Towards leveraging relation features for inductive link prediction in knowledge graphs. In *Proceedings of the 11th International Joint Conference on Knowledge Graphs, IJCKG 2022, Hangzhou, China, October 27-28, 2022*, pages 82–90. ACM, 2022.

[Hairer *et al.*, 2006] Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric Numerical Integration*. Springer, 2006.

[He *et al.*, 2025] Xin He, Yili Wang, Wenqi Fan, Xu Shen, Xin Juan, Rui Miao, and Xin Wang. Mamba-based graph convolutional networks: Tackling over-smoothing with selective state space. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 5345–5353, 2025. Main Track.

[Hogan *et al.*, 2021] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, Axel-Cyrille Ngonga Ngomo, Axel Polleres, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, and Antoine Zimmermann. *Knowledge Graphs*. Synthesis Lectures on Data, Semantics, and Knowledge. Morgan & Claypool Publishers, 2021.

[Lee *et al.*, 2023] Jaejun Lee, Chanyoung Chung, and Joyce Jiyoung Whang. InGram: Inductive knowledge graph embedding via relation graphs. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 18796–18809. PMLR, 2023.

[Qian *et al.*, 2025] Feifei Qian, Lu Bai, Lixin Cui, Ming Li, Hangyuan Du, Yue Wang, and Edwin Hancock. Exploring the over-smoothing problem of graph neural networks for graph classification: An entropy-based viewpoint. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 3235–3244, 2025. Main Track.

[Sanh *et al.*, 2019] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

[Shang *et al.*, 2024] Ziyu Shang, Peng Wang, Wenjun Ke, Jiajun Liu, Hailang Huang, Guozheng Li, Chenxiao Wu, Jianghan Liu, Xiye Chen, and Yining Li. Learning multi-granularity and adaptive representation for knowledge graph reasoning. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 2333–2341. International Joint Conferences on Artificial Intelligence Organization, 8 2024. Main Track.

[Shang *et al.*, 2025] Ziyu Shang, Peng Wang, Wenjun Ke, Jiajun Liu, Hailang Huang, Guozheng Li, Chenxiao Wu, Jianghan Liu, Xiye Chen, and Yining Li. Learning multi-granularity and adaptive representation for knowledge edge graph reasoning. *IEEE Transactions on Knowledge and Data Engineering*, 37(9):5360–5377, 2025.

[Teru *et al.*, 2020] Komal Teru, Etienne Denis, and William L Hamilton. Inductive relation prediction by subgraph reasoning. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 9448–9457. PMLR, 2020.

[Wu *et al.*, 2025] Peihan Wu, Hongda Qi, Sirong Huang, Dongdong An, Jie Lian, and Qin Zhao. Wave-driven graph neural networks with energy dynamics for over-smoothing mitigation. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence, IJCAI-25*, pages 6579–6587, 2025. Main Track.

[Xiong *et al.*, 2017] Wenhan Xiong, Thien Hoang, and William Yang Wang. DeepPath: A reinforcement learning method for knowledge graph reasoning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 564–573. Association for Computational Linguistics, 2017.

[Xu *et al.*, 2022] Xiaohan Xu, Peng Zhang, Yongquan He, Chengpeng Chao, and Chaoyang Yan. Subgraph neighboring relations infomax for inductive link prediction on knowledge graphs. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 2341–2347. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.

[Yue *et al.*, 2025] Juwei Yue, Haikuo Li, Jiawei Sheng, Yihan Guo, Xinghua Zhang, Chuan Zhou, Tingwen Liu, and Li Guo. Graph wave networks. In *Proceedings of the ACM Web Conference 2025 (WWW '25)*, pages 1365–1379. ACM, 2025.

[Zhang and Yao, 2022] Yongqi Zhang and Quanming Yao. Knowledge graph reasoning with relational digraph. In *Proceedings of the ACM Web Conference 2022 (WWW '22)*, pages 912–924. ACM, 2022.

[Zhu *et al.*, 2021] Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal Xhonneux, and Jian Tang. Neural Bellman-Ford networks: A general graph neural network framework for link prediction. In *Advances in Neural Information Processing Systems*, volume 34, 2021.