DAG-MAMBA: AN ORDER-ROBUST STRUCTURAL INTERFACE FOR STATE SPACE MODELING ON DIRECTED ACYCLIC GRAPHS

Anonymous authors
Paper under double-blind review

ABSTRACT

The directed acyclic graph (DAG) induces a partial order, but does not induce a unique node sequence, which causes a structural mismatch between it and the order-sensitive state space model (SSM). Arbitrary topological serialization may make the loop model depend on the relative order of incomparable nodes, even if the order is not part of the graph. We study how to align order-sensitive sequence models with DAGs without committing to an arbitrary linear extension. We propose DAG-Mamba, a sequential robust structural interface, which separates the node-level dependency information from the coarse-grained depth sequence. The local and bounded reachability branches retain the direct relationship and the predecessor-follower relationship, and the nodes with the same longest path depth are pooled into a norm-front token sequence ; we then use Mamba as a learned cross-frontier mixer. The frontier represents that the node index and the selected topology order remain unchanged, but is intentionally designed to be non-injective, so it complements rather than replaces the graph structure. On the code, reference, and neural architecture DAG, complete instantiation is significantly superior to sequential naive Mamba on several tasks and remains competitive with strong graph/DAG baselines. A diagnostic ablation further demonstrated that replacing the learned Mamba transfer with an identical leading-edge mixer on three ablation tasks could match or exceed the complete model. This distinguishes two confusing effects : sequential robustness is provided by the DAG interface, and the marginal value of the learned cross-frontier loop depends on the task. Therefore, we position DAG-Mamba as a Mamba instantiation of a sequential robust DAG interface, rather than as evidence that selective scanning is generally necessary for DAG learning. The simulation code can be obtained in `https://github.com/zongjin130/DAG-MAMBA`.

1 INTRODUCTION

Directed acyclic graphs (DAGs) appear in source code, neural architecture, computational graphs, reference networks, and workflows. Their edges induce a kind of partial order relation : some node pairs are precursor-successor pairs, while others are incomparable. This makes DAG richer in structure than sequences, but more ordered than general graphs.

Message passing GNNs utilize local connectivity (Gilmer et al., 2017), while graph Transformers achieve long-range interaction at the expense of potential secondary dense attention (Vaswani et al., 2017). The DAG-specific model uses dependent order, reachability, or depth references (Zhang et al., 2019; Thost & Chen, 2021; Dong et al., 2022; Luo et al., 2023b). The selective state space model provides another way to the global context: Mamba (Gu & Dao, 2024) uses input-dependent recursion and selective scanning instead of pairwise attention, and graph variants explore graph-aware tokenization and ordering (Wang et al., 2024; Behrouz & Hashemi, 2024).

For DAGs, the scheduling problem is particularly acute. A directed acyclic graph (DAG) can accommodate many effective topological classes, and the relative order of uncomparable nodes is

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.

arbitrary. Therefore, when two uncomparable nodes are exchanged, the order-sensitive recursion can be changed even if the graph is invariant. Our design separates two roles that may be confused by a single serialization. *structural path* runs at node resolution through local edges and bounded reachability. *global path* uses the longest path depth to divide the nodes into boundaries and pool each boundary invariablely, resulting in a typical depth sequence on which an order-sensitive mixer can be run. In the DAG-Mamba, we instantiate the mixer with Mamba. For node index and topology linearization, the frontier sequence is normative, but it is intentionally rough : depth consistency does not mean reachability, and different DAGs can share the same frontier summary. Therefore, local propagation and reachable propagation are complementary structural channels, rather than a supposedly lossless sequential repair.

This distinction also defines the scope of our empirical claims. The main question is not whether Mamba is generally superior to other front-end mixers, but whether a DAG-aware interface eliminates any node order dependence while retaining useful structure and global context. We use Mamba instantiation to learn cross-industry transitions and include an identity mixer control to separate the contribution of the interface from the learned recursive contribution. Identity control matches or improves the complete configuration on the three ablation tasks, indicating that the structural interface is a more consistently supported component under this protocol.

Our contributions are:

- **Order-robust DAG-to-sequence interface.** We formalize the sensitivity of any topological serialization and construct a depth-boundary sequence that removes the inner-boundary ordering and is independent of the selected linear extension before applying any sequence mixer.
- **Structural/global decomposition with explicit information loss.** The local and reachable channels preserve the node-level DAG relationship, while the front path provides a standardized rough global summary. We prove that boundary compression is non-injective, which clarifies why the global sequence must complement rather than replace the graph structure.
- **Diagnostic empirical characterization.** In several DAG domains, Mamba instantiation competes with strong baselines and often has a substantial improvement over uncommanded Mamba. Controlled identity mixer ablation separates the values of sequential robust interfaces from the learned cross-industry recursion and indicates that the latter is task-dependent.

2 Background and Related Work

DAG structure. Let $G = (V, E, X)$ be a DAG with $n = |V|$ nodes, directed edges $E \subseteq V \times V$, and node features $X \in \mathbb{R}^{n \times d}$. We write $u \preceq v$ if a directed path exists from u to v. This reachability relation defines a partial order. A source has no predecessors and a sink has no successors. Following prior DAG work (Luo et al., 2023b), we define node depth by

$$\mathrm{depth}(v) = \begin{cases} 0, & v \text{ is a source,} \\ 1 + \max_{(u,v)\in E} \mathrm{depth}(u), & \text{otherwise.} \end{cases} \tag{1}$$

The depth of G is $D = \max_{v \in V} \mathrm{depth}(v)$.

Selective state space models. A discrete state space model maps a sequence $x_1, \ldots, x_T$ to hidden states and outputs via

$$h_t = \bar{A}_t h_{t-1} + \bar{B}_t x_t, \qquad y_t = C_t h_t + D x_t. \tag{2}$$

Mamba makes the key SSM parameters related to the input, and realizes the result recursion through the hardware-aware selective scanning (Gu & Dao, 2024). This allows content-dependent propagation and forgetting without constructing an explicit $T \times T$ attention matrix. For a fixed state size, the cyclic mixing cost is linear in sequence length.

Mamba on graphs. The Mamba method of general graphs must transform an unordered neighborhood or graph into one or more sequences. Graph-Mamba (Wang et al., 2024) uses graph-centric

node prioritization and replacement strategies, while Graph Mamba Networks (GMNs) (Behrouz & Hashemi, 2024) combines neighborhood labeling, sorting, bidirectional selective SSMs, and local encoding. These methods motivate SSM-based graph learning, but their ranking mechanisms are designed for general graphs. On the contrary, it utilizes a property unique to DAG: partial order provides a typical order on the depth boundary, although it does not provide a typical total order on the node.

Transformers over DAGs. DAG-Transformer (Luo et al., 2023b) uses reachability to constrain attention and encodes node depth, which greatly improves the multi-graph Transformer backbone. PACE (Dong et al., 2022) also seeks parallel DAG coding by transforming the DAG structure into a representation suitable for attention. Our work also exploits reachability and depth, but uses them to define a structural interface for order-sensitive global sequence operators. The reported model instantiates this operator with Mamba on a compressed depth-frontier sequence; identity mixer ablation separately tests whether the learned cross-industry dynamics add value beyond the interface itself.

3 DAG-MAMBA

3.1 ORDER-SENSITIVE MODELS ON A PARTIAL ORDER

A topological sort $\pi = (v_1, \ldots, v_n)$ satisfies $u \preceq v \Rightarrow \pi(u) \leq \pi(v)$, but a DAG generally has many valid linear extensions.

Proposition 1 (topological serialization is not invariant in general). For any order-sensitive sequence operator S, there exists a DAG and two effective topological orders such that different outputs can be generated under S.

Proof. Take an inverse chain, and its node characteristics form a sequence to witness the order sensitivity of S. Each permutation is an efficient topological order, so two efficient orders can produce different outputs. □

Therefore, the topological validity itself does not provide the invariance of the equivalent linear extension.

3.2 CANONICAL DEPTH FRONTIERS

Using longest-path depth, define

$$F_\ell = \{v \in V \mid \mathrm{depth}(v) = \ell\}, \qquad \ell = 0, \ldots, D. \tag{3}$$

Lemma 1 (frontiers are antichains). If $u, v \in F_\ell$ and $u \neq v$, then neither $u \preceq v$ nor $v \preceq u$.

Proof. If a positive-length path connects u to v, longest-path depth gives $\mathrm{depth}(v) > \mathrm{depth}(u)$, contradicting equal depth. □

Depth is only a rough stratification: $u \prec v$ means $\mathrm{depth}(u) < \mathrm{depth}(v)$, and vice versa. Therefore, we do not regard depth as the partial order itself.

Let $h_v^{(l)} \in \mathbb{R}^d$ be a node state. We form one token per frontier,

$$z_\ell^{(l)} = \mathrm{Pool}\left(\{h_v^{(l)} : v \in F_\ell\}\right), \tag{4}$$

with permutation-invariant Pool, and define $Z^{(l)} = (z_0^{(l)}, \ldots, z_D^{(l)})$.

Proposition 2 (canonical frontier interface). For a fixed node representation, Z is invariant for permutations within each boundary and is independent of the selected node-level topology ordering. Under the graph-preserving node relabeling, the boundary token is unchanged.v

Proof. The depth is determined by the graph structure, the formula equation 4 is permutation invariant, and the sequence index is depth rather than node order. □

Proposition 3 (frontier compression is non-injective). The depth-and-pooling representation alone cannot uniquely encode DAG edge structure. For example, with sources a, b, depth-one nodes c, d, sink e, and identical node representations, the edge sets $\{a \to c, b \to d, c \to e, d \to e\}$ and $\{a \to c, a \to d, b \to d, c \to e, d \to e\}$ induce the same frontier multisets but different graphs. □

This non-injectivity is deliberate : the frontier channel is a normative global summary, not a full coding of the DAG. Importantly, the order robustness of this interface is mixer-unknown: any deterministic sequence operator that consumes only Z is independent of the selected node-level topology sort. Mamba is the learning sequence operator studied in this paper, and the node-level structure channel retains the information lost by the front compression.

3.3 Mamba as a frontier-level mixer

The above interface defines the sequence $Z^{(l)}$ independent of any node-level topology serialization. We instantiate its learned global mixer with Mamba. Odd layer scan source to sink, even layer scan Sink-to-Source:

$$\widetilde{Z}^{(l)} = \mathrm{Mamba}_{\rightarrow}(Z^{(l)}), \qquad l \text{ odd}, \tag{5}$$

$$\widetilde{Z}^{(l)} = \mathrm{reverse}\Big(\mathrm{Mamba}_{\rightarrow}(\mathrm{reverse}(Z^{(l)}))\Big), \qquad l \text{ even}. \tag{6}$$

Each node receives the contextualized token of its depth,

$$h^{(l)}_{v,\mathrm{frontier}} = \mathrm{Fuse}_{\mathrm{frontier}}\Big(h^{(l)}_{v}, \widetilde{z}^{(l)}_{\mathrm{depth}(v)}\Big). \tag{7}$$

Alternating direction is an implementation choice, not an optimality requirement for each layer of bidirectional or bidirectional alternatives. Cross-depth state flow does not mean node-level reachability ; the accurate predecessor-successor relationship is handled separately.

3.4 Node-level structural channels

The local branch aggregates on the original edges,

$$H^{(l)}_{\mathrm{local}} = \mathrm{LocalAgg}\Big(H^{(l)}, E\Big). \tag{8}$$

For $k \in \mathbb{N} \cup \{\infty\}$, define

$$\mathcal{N}^{\mathrm{reach}}_{k}(v) = \{u : u \preceq_k v\} \cup \{u : v \preceq_k u\}, \tag{9}$$

where $u \preceq_k v$ denotes a directed path of length at most k. The reachability branch aggregates only over these predecessor–successor pairs:

$$h^{(l)}_{v,\mathrm{reach}} = \mathrm{ReachAgg}\Big(\Big\{(h^{(l)}_{u}, e^{\mathrm{dist}}_{r_{uv}}, e^{\mathrm{dir}}_{d_{uv}}) : u \in \mathcal{N}^{\mathrm{reach}}_{k}(v)\Big\}\Big), \tag{10}$$

where r_{uv} is the shortest path distance and d_{uv} is the direction. The two structural branches retain the node-level relationship that the collected boundary markers cannot retain.

3.5 Layer composition and equivariance

We combine the three channels by

$$H^{(l+1)} = \mathrm{Norm}\Big(H^{(l)} + \alpha_l H^{(l)}_{\mathrm{local}} + \beta_l H^{(l)}_{\mathrm{reach}} + \gamma_l H^{(l)}_{\mathrm{frontier}}\Big), \tag{11}$$

where $\alpha_l, \beta_l, \gamma_l$ is a scalar gate for layer-by-layer learning. They provide coarse-grained component weighting rather than adaptive routing of nodes or graph conditions. Graph-level tasks use permutation invariant readout ; node-level tasks use the final node state.

Proposition 4 (node permutation equivariance). If the local aggregation and reachable aggregation are permutation-equivalent, and the front pooling is permutation-invariant, then the a DAG-Mamba layer is permutation-equivalent under the node relabeling.v

Table 1: Dataset statistics. n_∞ is the average number of nodes reachable from a node when the reachability radius is unbounded. Statistics follow the DAG-Transformer benchmark construction (Luo et al., 2023b).

	code2	NA	Self-cit.	Cora	Citeseer	Pubmed
# graphs	452,741	19,020	1,000	1	1	1
Avg. # nodes	125.2	8.0	59.1	2,708	3,327	19,717
Avg. n_∞	9.78	7.00	4.30	20.88	5.33	60.56

Proof sketch. The branch of the structure is interoperable with the relabeling; proposition 2 makes the boundary label independent of the index within the boundary, and the output of the broadcast node is consistent through deep relabeling. Residual addition and normalized retention are equivariant. □

3.6 Complexity

Let $m = |E|$, $r_k = |\{(u, v) : u \in \mathcal{N}_k^{\text{reach}}(v)\}|$, D be maximum depth, and s the SSM state size. Ignoring feature-projection costs common to the blocks, local propagation costs $\mathcal{O}(md)$, reachability costs $\mathcal{O}(r_k d)$, and frontier mixing costs $\mathcal{O}(Dds)$. Hence the structure-dependent mixing cost is

$$\mathcal{O}(md + r_k d + Dds). \tag{12}$$

Only the frontier mixer is guaranteed to be linear in the number of depth levels. The full model is not necessarily subquadratic: under the transitive closure, r_k can be $\Theta(n^2)$. Bound k controls this individual reachable term.

4 Experiments

We evaluate three questions. First, by placing Mamba behind DAG-aware structural interfaces, can we improve unsorted Mamba while maintaining competition with established graph/DAG baselines? Secondly, controllable ablation can distinguish the contribution of node hierarchy and normative frontier interface from the contribution of cross-industry recursion ? Third, how to calculate the quality of transaction prediction with bounded reachable radius? The organization intentionally separates claims for interfaces from claims for specific sequence mixers.

4.1 Setup

Datasets. We use the same DAG benchmark family as the previous DAG-Transformer work (Luo et al., 2023b): ogbg-code2 (Hu et al., 2020), neural architecture(NA) (Zhang et al., 2019; Pham et al., 2018). Self-citep (Luo et al., 2023a), and directed reference networks Cora, Citeseer and Pubmed (Sen et al., 2008). The statistics of the dataset are shown in table 1.

Baselines and reporting. We compare it with message passing GNNs, DAG-specific encoders, Graph Transformers, Mamba, Graph-Mamba (Wang et al., 2024) and Graph Mamba Networks (Behrouz & Hashemi, 2024). When the data set is split and the metric matches (Luo et al., 2023b), the published non-Mamba values are taken from the corresponding benchmark table ; the Mamba-family entry as a result of our report was obtained from our experimental pipeline. Therefore, we conservatively explain the small cross-paper differences and do not view them as evidence of superiority unless they exceed the reported variation. The published baselines are not re-run on our hardware, so wall clock time is only used for comparison between runs generated under our own protocol, not for cross-paper speed declarations.v

Implementation environment. This experiment runs on Ubuntu 24.04.2 LTS (Linux kernel 6.8.0-31-generic). The processor is AMD EPYC 7763 64-Core, the system memory is 188 GiB, and the visual RTX 2080 Ti. The system reports 22,528 MiB GPU memory and uses NVIDIA driver 570.133.07. The software stack uses Python 3.9.18, PyTorch 2.0.1 and CUDA 11.8, PyTorch Geometric 2.0.4 and OGB 1.3.6. The operating system exposes 256 logical CPUs.

Table 2: ogbg-code2 function-name prediction. Published non-Mamba baselines are from Luo et al. (2023b); Mamba-family rows marked here are from our experimental runs.

Model	Valid F1 (%)	Test F1 (%)
GIN	13.7 ± 0.2	14.9 ± 0.2
GCN	14.0 ± 0.2	15.1 ± 0.2
DAGNN	16.1 ± 0.4	17.5 ± 0.5
PACE	16.3 ± 0.3	17.8 ± 0.2
Transformer	15.5 ± 0.2	16.7 ± 0.2
DAG+Transformer	17.4 ± 0.1	18.8 ± 0.2
GraphGPS	17.4 ± 0.2	18.9 ± 0.2
SAT	17.7 ± 0.2	19.4 ± 0.3
DAG+SAT	18.5 ± 0.1	20.2 ± 0.2
Mamba	17.3 ± 0.2	19.1 ± 0.2
Graph-Mamba	18.6 ± 0.2	20.3 ± 0.2
DAG-Mamba	18.2 ± 0.2	20.4 ± 0.2

Table 3: Self-citation node classification (%). Published baselines follow Luo et al. (2023b) where available.

Model	AP ↑	ROC-AUC ↑
PNA	62.4 ± 0.7	81.0 ± 0.4
DAGNN	61.2 ± 0.6	81.0 ± 0.3
Transformer	56.8 ± 1.8	78.7 ± 0.3
DAG+Transformer	63.8 ± 0.8	82.2 ± 0.5
GraphGPS	61.6 ± 2.6	81.3 ± 0.6
SAT	59.8 ± 1.7	79.8 ± 0.7
Mamba	46.2 ± 0.5	71.1 ± 0.2
Graph-Mamba	64.7 ± 1.1	82.5 ± 0.4
GMN	65.4 ± 1.3	82.5 ± 0.4
DAG-Mamba	64.0 ± 0.7	83.1 ± 0.3

4.2 Main results and discussion

Code graphs. On ogbg-code2 (Table 2), DAG-Mamba obtains 20.4% of test F1, while our original Mamba baseline is 19.1%, the published DAG+SAT is 20.2%, and Graph-Mamba is 20.3%. The 0.1-point gap with Graph-Mamba is lower than the reported mutation scale and is not considered a meaningful advantage. The results instead support a narrower comparison center for this work : the introduction of the DAG-aware structure significantly improves the unordered Mamba baseline while maintaining competitiveness with graph-aware alternatives.

Self-citation. In the case of self-citation (table 3), the AP / ROC-AUC of DAG-Mamba increased from $46.2/71.1$ for order-naive Mamba to $64.0/83.1$. Because the complete model changes several structural components at the same time, this comparison supports the DAG-specific decomposition of the whole, rather than the isolated frontier transition as the cause. Graph-Mamba and GMN are still competitive, so we do not advocate dominance in graph-aware alternatives.

Citation networks and neural architectures. Table 4 shows a mixed picture. Although DAG-Mamba is still lower than the most powerful listing models on Cora, Citeseer and Pubmed, DAG-Mamba significantly improves our unordered Mamba baseline on Cora, Citeseer and Pubmed. On NA, the root mean square error of DAG-Mamba is 0.313, and the Pearson correlation coefficient is 0.947, which are worse than Mamba and DAG+Transformer. We regard NA as an explicit boundary case, rather than attribute the gap to an unmeasurable mechanism. Combined with the following mixer ablation, this result shows that neither frontier-level context nor learned cross-industry recursion should be expected to be consistently helpful in DAG tasks.

Table 4: Results on citation networks (accuracy, %) and NA regression. Non-Mamba baselines follow published values in Luo et al. (2023b); DAG-Mamba/Mamba values are from our experimental runs.

	Cora	Cite.	Pubmed	RMSE ↓	r ↑
GCN	87.06	75.75	88.16	0.482	0.871
DAGNN	–	–	–	0.264	0.964
PACE	79.47	73.65	OOM	0.254	0.964
Transformer	75.92	72.23	OOM	0.285	0.957
DAG+Transformer	87.80	74.42	89.01	0.253	0.966
SAT	75.18	74.88	OOM	0.298	0.952
NodeFormer	88.80	76.33	89.32	–	–
DAG+NodeFormer	90.49	78.24	89.44	–	–
Mamba	71.78	69.80	87.22	0.254	0.964
DAG-Mamba	85.08	75.84	89.53	0.313	0.947

Table 5: Controlled interface/mixer ablations. "Frontier + identity mixer" retains the canonical frontier construction and fusion but removes learned cross-depth state transitions, allowing the interface to be separated from the Mamba recurrence.

	ogbg-code2		NA		Self-citation	
Ablation	Valid F1	Test F1	RMSE ↓	r ↑	AP ↑	AUC ↑
DAG-Mamba	0.1813 ± .0014	0.2020 ± .0014	0.2689 ± .002	0.9617 ± .001	0.6402 ± .008	0.8302 ± .005
w/o Local	0.1846 ± .002	0.2005 ± .001	0.2612 ± .003	0.9639 ± .001	0.6072 ± .011	0.7931 ± .003
w/o Reachability	0.1762 ± .001	0.1953 ± .001	0.2565 ± .002	0.9654 ± .001	0.6600 ± .028	0.8165 ± .018
w/o Frontier branch	0.1849 ± .003	0.1993 ± .003	0.2518 ± .003	0.9668 ± .001	0.6586 ± .014	0.8320 ± .010
Frontier + identity mixer	0.1855 ± .001	0.2034 ± .001	0.2521 ± .004	0.9667 ± .001	0.6598 ± .018	0.8369 ± .003

4.3 Ablation: structural interface versus sequence mixer

Table 5 aims to separate three effects : node-level structure propagation, specification frontier interface, and learned cross-industry recursion. Three leave-one-component output rows test the structure/global branch, while the final diagnostic row *Front + identity mixer* preserves the front structure, invariant pooling, broadcast/fusion, and local/reachable branches, but removes the learned cross-depth conversion by setting v

$$\widetilde{z}_\ell = z_\ell. \tag{13}$$

All rows share the same fixed ablation protocol and should be compared within this table.

Structural channels. Removing local branches has little effect on code2, but has a greater impact on self-citation, while removing reachability has the greatest impact on code2. The symbols on NA and self-citation are not uniform, so we do not think that each structural component is generally beneficial ; their roles are different and depend on the data set.

Frontier interface versus learned transition. Identity control is the core diagnostic because it preserves the same-order robust frontier structure and only removes the learned cross-depth dynamics. On code 2, the identity configuration reaches 0.2034 to test F1, while it is 0.1993 when the frontier branch is removed, and 0.2020 when Mamba changes. Therefore, even in the absence of learned relapses, the normative frontier path contributes to this task, and selective transfer alone does not explain this benefit. On NA, identity and boundless configuration are almost tied together (0.2521 vs.0.2518 RMSE), and the complete Mamba transition is even worse (0.2689). In self-citation, identity configuration is also at least competitive with the removal of frontier branches, and exceeds complete conversion in both AP and AUC. These comparisons support a decomposition of the evidence : sequential robust interfaces can be used independently of the learned cross-industry recursion, and the recursive values are task-dependent. Therefore, we do not use this ablation to claim that selective state-space conversion is indispensable.

Table 6: Effect of bounded reachability radius k on ogbg-code2. n_k is the average reachable-neighborhood size. Time is seconds per epoch measured on the common workstation described in the experimental setup and is intended for within-sweep comparison only.

k	Avg. n_k	Test F1 ↑	Time (s/epoch) ↓
1	1.97	0.1988 ± 0.0022	1497
2	3.91	0.2011 ± 0.0023	1601
3	5.69	0.2016 ± 0.0006	1891
4	7.16	0.2020 ± 0.0021	2007
5	8.29	0.2021 ± 0.0020	2101
6	9.01	0.2029 ± 0.0020	2187
7	9.40	0.2031 ± 0.0019	2226
8	9.60	$\mathbf{0.2041 \pm 0.0008}$	2256
∞	9.78	0.2000 ± 0.0015	2279

4.4 Impact of reachability radius

Setup. For example, in DAG-Transformer (Luo et al., 2023b), the reachable radius k controls the structural receptive field. For a node v, the reachable branch contains only the successors and successors of a directed path of length at most k connected to v. We use n_k to represent the average size of this reachable neighborhood. Therefore, the increase of k not only increases the number of explicit partial order information, but also increases the cost of reachability branch. Table 6 reports the ogbg-code2 scan. In the execution environment specified by Section 4.1, all epoch times in the scan are recorded; therefore, they are used to analyze the cost trend within the scan. We do not directly compare these times with published baselines running in different or unspecified hardware and software environments.

Reachability trade-off. From $k = 1$ to $k = 8$, test F1 rises from 0.1988 to 0.2041 while the average reachable set grows from 1.97 to 9.60 nodes; gains diminish after roughly $k = 4$, while recorded epoch time continues to increase. The unbounded setting is not best: $k = \infty$ gives 0.2000 F1. Thus k given by $k = \infty$ should be treated as a precision-calculating hyperparameter, rather than automatically set as a transitive closure. One explanation is that once there is a single global path at the frontier level, explicit remote reachability can become redundant, but this scan does not isolate the Mamba transition.

5 Limitations

The main empirical limitation is specific to the learned frontier mixer : in controlled ablation, the Mamba conversion is not superior to the identical mixer. Therefore, we propose a structure-interface declaration, rather than a declaration, that Mamba is indispensable for DAG learning. Boundary compression is also non-injective, sharing a global label across nodes, and can have different ancestors, descendants, and semantics ; depth is a rough order that conforms to accessibility, not the partial order itself. Efficiency is also conditional. The front mixer is linear in the number of depth levels, but the reachable term can become quadratic under the transitive closure. The timing scan in our study was measured on an ordinary workstation, but the published baseline was not re-run in the same environment, so this article does not make a speed statement across the paper wall clock. Finally, the experiment did not report the depth, boundary width, compression ratio or reachable density statistics, which will be used to identify the learned boundary recurrence as a beneficial structural mechanism. Alternating scan directions and layer-by-layer scalar gates should also be considered as simple design choices, rather than proven best choices.

6 Conclusion

We introduce the method as the order-robust structural interface between DAG and the order-sensitive sequence model, and instantiate it with Mamba. The local and reachable channels preserve the node-level dependencies, while the invariant depth-front pooling provides a regular but deliberately rough global sequence without any topological linearization. This analysis makes both

invariance and information loss explicit. Empirically, Mamba instantiation competes with powerful graph / DAG baselines and often has significant improvements over unordered Mamba. Identity mixer diagnostics further separates the sources of these gains : sequential robust interfaces are consistently supported, and learned cross-industry recursion is task-dependent under this protocol. The resulting conclusion is about the universality of interface design rather than mixers : when a sequence-sensitive model is applied to DAGs, the representation of the partial order must be separated from any selection of linear extensions.

REPRODUCIBILITY STATEMENT

Section 3 details the depth definition, frontier construction, Mamba instantiation, reachability neighborhood, layer composition, invariance assumption, non-injection results and complexity range. Section 4 separates the published baseline values from the measurements generated by our experimental pipeline and reports the controlled ablation and reachability radius analysis. We ran Ubuntu 24.04.2 LTS, AMD EPYC 7763 CPU, 188 GiB RAM, and RTX 2080 Ti, and the software version is listed in the experimental setup. Runtime values are only compared in our common environment ; the published baselines are not re-run on the same hardware.

AI USE STATEMENT

Produce AI tools to assist in drafting and rewriting manuscripts, organizing discussions of experimental measurements provided by authors, LaTeX recombination, conformance checking, and bibliographic organization. They are not used to generate, modify or reduce experimental measurements. The author is responsible for verifying the results of all technical claims, equations, references, and reports based on underlying code, logs, and source publications.

REFERENCES

Ali Behrouz and Farnoosh Hashemi. Graph mamba: Towards learning on graphs with state space models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 119–130, 2024. doi: 10.1145/3637528.3672044.

Zehao Dong, Muhan Zhang, Fuhai Li, and Yixin Chen. PACE: A parallelizable computation encoder for directed acyclic graphs. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 5360–5377. PMLR, 2022.

Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1263–1272. PMLR, 2017.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=tEYskw1VY2.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems*, volume 33, pp. 22118–22133, 2020.

Yuankai Luo, Lei Shi, Mufan Xu, Yuwen Ji, Fengli Xiao, Chunming Hu, and Zhiguang Shan. Impact-oriented contextual scholar profiling using self-citation graphs. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 4572–4583, 2023a.

Yuankai Luo, Veronika Thost, and Lei Shi. Transformers over directed acyclic graphs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 47764–47782, 2023b. doi: 10.52202/075280-2070.

Hieu Pham, Melody Guan, Barret Zoph, Quoc V. Le, and Jeff Dean. Efficient neural architecture search via parameter sharing. In *Proceedings of the 35th International Conference on Machine*

Learning, volume 80 of *Proceedings of Machine Learning Research*, pp. 4095–4104. PMLR, 2018.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. *AI Magazine*, 29(3):93–106, 2008.

Veronika Thost and Jie Chen. Directed acyclic graph neural networks. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=JbuYF437WB6.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Chloe Wang, Oleksii Tsepa, Jun Ma, and Bo Wang. Graph-mamba: Towards long-range graph sequence modeling with selective state spaces. *arXiv preprint arXiv:2402.00789*, 2024.

Muhan Zhang, Shali Jiang, Zhicheng Cui, Roman Garnett, and Yixin Chen. D-VAE: A variational autoencoder for directed acyclic graphs. In *Advances in Neural Information Processing Systems*, volume 32, pp. 1586–1598, 2019.