GRAPHLET MPNNS: EXTENDING MESSAGE-PASSING NEURAL NETWORKS WITH GRAPHLET INFORMATION

Anonymous authors
Paper under double-blind review

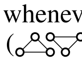
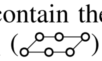
ABSTRACT

We propose Graphlet Message-Passing Neural Networks (MPNNs) as an extension of classical MPNNs in which vertex and edge graphlet information is taken into account. In this way, the distinguishing power of MPNNs is increased in a natural way. We introduce Graphlet MPNNs in quite some generality, hereby encompassing recent proposals. Our main result is a complete characterization of the distinguishing power of Graphlet MPNNs. We conclude this paper by outlining some interesting directions for future research.

1 INTRODUCTION

Xu et al. (2019) and Morris et al. (2019) showed that vertex embeddings, computed by Message-Passing Neural Networks (MPNNs), are inherently limited in expressive power. This limitation can be elegantly formalized, as follows. Let $\mathcal{M}$ be an MPNN and denote by $\mathcal{M}(G, v)$ and $\mathcal{M}(G)$ the embedding (in some $\mathbb{R}^d$), computed by $\mathcal{M}$, of vertex v in the graph G, and of the graph itself, respectively. For a graph X, denote by X^x a rooted version of X, where x is a vertex in X. Then:

Proposition 1 ((Dvořák, 2010; Dell et al., 2018))**.** For any two graphs G and H, vertices v in G and w in H, and *any* MPNN $\mathcal{M}$: $\mathcal{M}(G, v) = \mathcal{M}(H, w)$ if for any rooted tree T^r, $\mathsf{hom}(T^r, G^v) = \mathsf{hom}(T^r, H^w)$. Similarly, $\mathcal{M}(G) = \mathcal{M}(H)$ if for any (unrooted) tree T, $\mathsf{hom}(T, G) = \mathsf{hom}(T, H)$.

Here, $\mathsf{hom}(X, Y)$ denotes the number of edge-preserving vertex mappings from graph X to Y (called homomorphisms). In case X and Y are rooted, homomorphisms should additionally map the root of X to the root of Y. Proposition 1 thus tells that MPNNs cannot distinguish vertices and graphs, based on the computed embeddings, whenever they contain the same tree-like structures. For example, the classical example graphs G_1 () and H_1 () cannot be distinguished by any MPNN. We also remark that the class of MPNNs is powerful enough such that there exists an MPNN $\mathcal{M}$ for which $\mathcal{M}(G, v) = \mathcal{M}(H, w)$ implies that $\mathsf{hom}(T^r, G^v) = \mathsf{hom}(T^r, H^w)$ for any rooted tree T^r, and similarly for graph embeddings (Xu et al., 2019; Morris et al., 2019).

Inspecting the example graphs G_1 and H_1 reveals what graph information could be used for distinguishing them. Indeed, G_1 contains two triangles (3-cliques), whereas H_1 does not contain any. Embedding G_1 and H_1 in $\mathbb{R}$ in terms of 3-clique counts thus suffices to distinguish them. In general, subgraph counts were used by Przulj (2007) to define vertex embeddings (graphlet degree vectors), and by Shervashidze et al. (2009) to define graph embeddings (graphlet kernel). In those works, graphlets refer to small graphs and for each graphlet, the number of induced subgraphs isomorphic to the graphlet, is taken as a feature of a vertex or graph.

Natural questions are now: How can graphlet information be integrated in MPNNs? Can MPNNs benefit from having such additional graphlet information? Can Proposition 1 be extended to such "Graphlet MPNNs"? We will address these questions in quite some generality, by allowing a flexible way to define how graphlets can be tied to the underlying graph data.

2 GRAPHLET MPNNS

Graphlet MPNNs are inspired by the Graph Substructure Networks (GSNs) of Bouritsas et al. (2021) and the $\mathcal{P}$-MPNNs of Barceló et al. (2021). In those works, graph structural information is used to *augment* the initial vertex and edge features, by using (induced) subgraph counts and homomorphism counts of graphlets, respectively. Subsequently, standard MPNNs are executed on such "augmented" graphs. Graphlet MPNNs can be regarded as a unifying model for these two approaches and further allow to define variants using *general counting mechanisms* of graphlets. We first explain what we mean by such counting mechanisms and then define Graphlet MPNNs.

Counting vertex- and edge-graphlets. For a graph X, we denote by V_X and E_X its set of vertices and edges, respectively. Moreover, $X^{x,y}$ denotes a two-rooted graph X with $(x,y) \in E_X$. A *graphlet* P is just a small graph, a *vertex-graphlet* P^r is single-rooted graphlet, and an *edge-graphlet* Q^{r_1,r_2} is a two-rooted graphlet. Let P^r be a vertex-graphlet and G a graph. We define the following set of vertex mappings: $\mathsf{Map}_\varphi(P^r, G^v) := \{f : V_P \to V_G \mid f(r) = v, f \text{ satisfies } \varphi\}$. Similarly, for an edge-graphlet Q^{r_1,r_2} we define $\mathsf{Map}_\psi(Q^{r_1,r_2}, G^{v,w}) := \{f : V_P \to V_G \mid f(r_1) = v, f(r_2) = w, f \text{ satisfies } \psi\}$. Crucial in these definitions are the *conditions* φ and ψ which restrict the allowed vertex mappings. Example conditions are:

- If $\varphi = \forall\, p,q \in V_P \big((p,q) \in E_P \to (f(p), f(q)) \in E_G\big)$, then $\mathsf{Map}_\varphi(P^r, G^v)$ consists of *homomorphisms* from P^r to G^v;
- If $\varphi = \forall\, p,q \in V_P \big(\big((p,q) \in E_P \to (f(p), f(q)) \in E_G\big) \wedge \big((p \neq q) \to f(p) \neq f(q)\big)\big)$, then $\mathsf{Map}_\varphi(P^r, G^v)$ consists of *subgraph isomorphisms* from P^r to G^v; and
- $\varphi = \forall\, p,q \in V_P \big(\big((p,q) \in E_P \leftrightarrow (f(p), f(q)) \in E_G\big) \wedge \big((p \neq q) \to f(p) \neq f(q)\big)\big)$, then $\mathsf{Map}_\varphi(P^r, G^v)$ consists of *induced subgraph isomorphisms* from P^r to G^v.

Similar conditions ψ can be used for edge-graphlets. Depending on the graph learning task at hand, any other condition can be used. Given these sets of mappings, we next turn them into *quantitative* vertex and edge features. More precisely, we obtain features by *counting* graphlets as follows:[1]

$$\#\varphi(P^r, G^v) := |\mathsf{Map}_\varphi(P^r, G^v)| \text{ and } \#\psi(Q^{r_1,r_2}, G^{v,w}) := |\mathsf{Map}_\varphi(Q^{r_1,r_2}, G^{v,w})|.$$

With these vertex and edge features at hand, we can now define Graphlet MPNNs.

Graphlet MPNNs. Consider a set $\mathcal{P} = \{P_1^r, \ldots, P_p^r\}$ of vertex-graphlets, a set $\mathcal{Q} = \{Q_1^{r_1,r_2}, \ldots, Q_q^{r_1,r_2}\}$ of edge-graphlets, and conditions φ and ψ used to restrict vertex mappings for vertex-graphlets and edge-graphlets, respectively, as explained above. Then, a *Graphlet* MPNN is formally specified as a $(\mathcal{P}_\varphi, \mathcal{Q}_\psi)$-MPNN in which:

- The initial vertex labels (lab_v) are augmented with vertex-graphlet counts, and edge-graphlet counts are used for the edge labels. That is, the initial vertex embedding and edge labels are:
$$\mathbf{l}_v^{(0)} := \big(\mathsf{lab}_v, \#\varphi(P_1^r, G^v), \ldots, \#\varphi(P_p^r, G^v)\big)$$
$$\mathbf{l}_{vw} := \big(\#\psi(Q_1^{r_1,r_2}, G^{v,w}), \ldots, \#\psi(Q_q^{r_1,r_2}, G^{v,w})\big).$$
- Then, the vertex embedding $\mathbf{l}_v^{(t)}$ is updated just as in standard MPNNs (Gilmer et al., 2017):
$$\mathbf{l}_v^{(t+1)} := \mathsf{Update}^{(t)}\left(\mathbf{l}_v^{(t)}, \mathsf{Aggregate}^{(t)}\big(\{\!\{(\mathbf{l}_w^{(t)}, \mathbf{l}_{vw}) \mid w \in N_G(v)\}\!\}\big)\right),$$
where $N_G(v)$ denotes the set of vertices adjacent to v in G, and $\mathsf{Aggregate}$ and Update are arbitrary functions operating on multisets of vectors (denoted by $\{\!\{\,\}\!\}$) and vectors, respectively.
- Finally, to obtain a graph embedding, a general *readout* function can be applied after T updates. More precisely:
$$\mathbf{l}_G := \mathsf{Readout}\left(\{\!\{\mathbf{l}_v^{(T)} \mid v \in V_G\}\!\}\right).$$

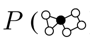
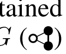
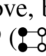
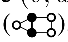
With Graphlet MPNNs in place, we next endeavor to provide a characterization of their distinguishing power. More precisely, we show a generalization of Proposition 1 for Graphlet MPNNs.

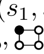
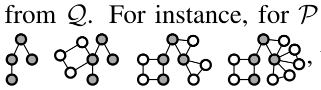
3 Distinguishing power of Graphlet MPNNs

In order to generalize Proposition 1 to Graphlet MPNNs, we take inspiration from Barceló et al. (2021). In that paper, the authors introduced $\mathcal{P}$-MPNNs, for a set $\mathcal{P}$ of vertex-graphlets and considered homomorphism counts. As such, a $\mathcal{P}$-MPNN is a $(\mathcal{P}_\varphi, \emptyset)$-MPNNs with φ the condition

[1] One can further generalize this by allowing other functions on the cardinalities of the sets of mappings.

corresponding to homomorphisms, as described earlier. Barceló et al. (2021) show that the distinguishing power of $\mathcal{P}$-MPNNs can be expressed in terms of homomorphism counts of so-called $\mathcal{P}$-pattern trees. Furthermore, it was observed that this pattern tree characterization can be extended to a sub-class of GSNs (Bouritsas et al., 2021) in which only vertex-graphlets and induced subgraph counts are used. This subclass corresponds to $(\mathcal{P}_\varphi, \emptyset)$-MPNNs with φ the condition corresponding to induced subgraph isomorphisms, as described earlier. In this section, we generalize the pattern tree characterization for general Graphlet MPNNs, which can use general counting mechanisms (expressed through the conditions φ and ψ), and that incorporate both vertex- and edge-graphlets. We start by describing pattern trees, followed by our main result.

Pattern trees. Given a graph G, a vertex $v \in V_G$ and a vertex-graphlet P^r, the join graph $G \star P$ is obtained by taking the disjoint union of G and P, followed by identifying the root vertex r with v. For example, the join of G () and P () along the marked vertices is $G \star P$ (). Similarly, for a graph G, an edge $(v, u) \in E_G$ and an edge-graphlet Q^{r_1,r_2}, the join graph $G \star Q$ is obtained as above, but this time by identifying the edge (v, u) with (r_1, r_2). For example, the join of G () and Q () along the marked edges is $G \star Q$ ().

Given a collection $\mathcal{P}$ of vertex-graphlets and $\mathcal{Q}$ of edge-graphlets, we next define a $(\mathcal{P}, \mathcal{Q})$*-pattern tree* T^r: It is a rooted tree, obtained from a standard rooted tree $S^r = (V_S, E_S)$, called the *backbone* of T^r, followed by joining every vertex $s \in V_S$ with any number of copies of vertex-graphlets from $\mathcal{P}$ and by joining every edge (s_1, s_2) in E_S with any number of copies of edge-graphlets from $\mathcal{Q}$. For instance, for $\mathcal{P} = \{$, $\}$ and $\mathcal{Q} = \{$$\}$, examples of $(\mathcal{P}, \mathcal{Q})$-pattern trees are: , where we marked vertices in the backbone in gray.

Main result. We are now ready to state our generalization of Proposition 1. In this generalization, trees will be replaced by $(\mathcal{P}, \mathcal{Q})$-pattern trees and we will use a revised notion of homomorphism, based on the conditions φ and ψ used in Graphlet MPNNs.

More precisely, let T^r be a $(\mathcal{P}, \mathcal{Q})$-pattern tree with backbone S^r, and consider G^v for graph G. A (φ, ψ)*-homomorphism* f from T^r to G^v is defined as follows: It is mapping $f : V_T \to V_G$ satisfying (i) $f(r) = v$; (ii) the restriction $f\big|_S$ of f to the backbone S is a standard homomorphism; (iii) the restriction $f\big|_P$ of f to each (copy of a) joined vertex-graphlet P in T^r is an element in $\mathsf{Map}_\varphi(P^r, G^v)$; and finally, (iv) the restriction $f\big|_Q$ of f to each (copy of a) joined edge-graphlet Q in T^r is an element in $\mathsf{Map}_\psi(Q^{r_1,r_2}, G^{v,w})$. In accordance with the notation used in Proposition 1, we denote the number of (φ, ψ)-homomorphisms from T^r to G^v by $\mathsf{hom}_{\varphi,\psi}(T^r, G^v)$. Finally, a (φ, ψ)-homomorphism from an unrooted $(\mathcal{P}, \mathcal{Q})$-pattern tree T to G is defined as a (φ, ψ)-homomorphism from a rooted version T^r of T to any rooted version G^v of G, for $v \in V_G$. We use $\mathsf{hom}_{\varphi,\psi}(T, G)$ to denote the number of such homomorphisms. Our main result is as follows:

Theorem 2. For any two graphs G and H, vertices v in G and w in H, and *any* $(\mathcal{P}_\varphi, \mathcal{Q}_\psi)$-MPNN $\mathcal{M}$, we have the following:

- $\mathcal{M}(G, v) = \mathcal{M}(H, w)$ if for any rooted $(\mathcal{P}, \mathcal{Q})$-pattern tree T^r, we have that $\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(T^r, H^w)$; and
- $\mathcal{M}(G) = \mathcal{M}(H)$ if for any (unrooted) $(\mathcal{P}, \mathcal{Q})$-pattern tree T, $\mathsf{hom}_{\varphi,\psi}(T, G) = \mathsf{hom}_{\varphi,\psi}(T, H)$.

Just as for classical MPNNs, the class of $(\mathcal{P}_\varphi, \mathcal{Q}_\psi)$-MPNNs is rich enough such that there exists a $(\mathcal{P}_\varphi, \mathcal{Q}_\psi)$-MPNN for which $\mathcal{M}(G, v) = \mathcal{M}(H, w)$ implies that $\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(T^r, H^w)$ for any rooted $(\mathcal{P}, \mathcal{Q})$-pattern tree T^r, and similarly for graph embeddings.

Theorem 2 tells that, intuitively, even when starting from a small number of vertex- and edge-graphlets, when integrated in MPNNs, more complex graph patterns can be detected. Indeed, one can show that a T-layered Graphlet MPNN can use $(\mathcal{P}, \mathcal{Q})$-pattern trees of depth at most T (where depth is defined in terms of the tree's backbone). This is in contrast to, say graphlet degree vectors or graph kernels, where one needs to explicitly provide all graphlets of interest. As such, Graphlet MPNNs are more flexible and require less hand-crafted features (the initial set of graphlets). The proof of Theorem 2 is a modification of the one in Grohe (2020a) and Barceló et al. (2021), but generalized to edge-graphlets and general conditions φ and ψ (see supplementary material).

We remark that when $\mathcal{P}$ and $\mathcal{Q}$ are empty, $(\mathcal{P}, \mathcal{Q})$-pattern trees are just trees, and (φ, ψ)-homomorphisms are just homomorphisms. Hence, Theorem 2 encompasses Proposition 1 as a special case. Similarly, it also encompasses the results by Barceló et al. (2021) for the special cases of Graphlet MPNNs described in the beginning of this section.

4 Looking ahead

We conclude by pointing out some interesting directions for future research and by describing some related preliminary results. We defer details to the supplementary material.

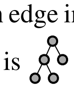
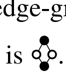
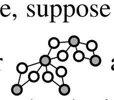
Comparison of Graphlet MPNNs. In designing Graphlet MPNNs one has to decide what graphlets to include and what kind of mappings (conditions φ and ψ) are of interest. A natural question is how different instantiations of Graphlet MPNNs compare to each other, in terms of distinguishing power or performance. For a theoretical comparison, Theorem 2 may turn useful. For example, one can show that there are graphs that can be distinguished by GSNs but not by $\mathcal{P}$-MPNNs when using the same graphlets, and vice versa. This shows that the choice of φ, i.e., homomorphisms versus subgraph isomorphisms, makes a difference. A related question is what the impact is of using vertex-graphlets versus edge-graphlets. One intuitively would expect that edge-graphlets result in stronger features since vertex-graphlet features "summarize" all information into single vertices. We have the following result, where we use edge-graphlets in $\mathcal{Q}$ also as vertex-graphlets by ignoring one its roots.

Proposition 3. $(\emptyset, \mathcal{Q}_\psi)$-MPNNs can distinguish more graphs than $(\mathcal{Q}_\psi, \emptyset)$-MPNNs; and $(\mathcal{Q}_\psi, \mathcal{Q}_\psi)$-MPNNs are exactly as strong as $(\emptyset, \mathcal{Q}_\psi)$-MPNNs.

This result resolves an open problem of GSNs (Bouritsas et al., 2021): edge-graphlets are strictly more powerful than vertex-graphlets. Clearly, these results only provide an initial understanding of the entire space of Graphlet MPNNs.

Other uses of graphlets. So far, graphlets are used to augment vertex and edge features. We can, however, also use edge-graphlets for defining an extended notion of neighborhood, over which MPNNs can aggregate. Such an approach is recently proposed by Li et al. (2021). The idea is as follows: Given an edge-graphlet Q^{r_1,r_2} (where now r_1 and r_2 are abitrary) and corresponding condition ψ, define the extended neighborhood $N_{G,Q_\psi}(v) := \{u \in V_G \mid \exists f \in \mathsf{Map}_\psi(Q^{r_1,r_2}, G^{v,u})\}$. In other words, vertices $u \in V_G$ are treated as neighbors of v, when they are linked through the edge-graphlet via a mapping satisfying ψ. Using this, we can extend MPNNs by aggregating over this extended set of neighbors:

$$\mathbf{l}_v^{(t+1)} := \mathsf{Update}^{(t)}\left(\mathbf{l}_v^{(t)}, \mathsf{Aggregate}^{(t)}\big(\{\!\{(\mathbf{l}_u^{(t)}, \mathbf{l}_{vu}) \mid u \in N_G(v) \cup N_{G,Q_\psi}(v)\}\!\}\big)\right).$$

Furthermore, one can similarly revise $(\mathcal{P}_\varphi, \emptyset)$-MPNNs.[2] Let us denote the resulting extension by $\mathcal{P}_\varphi$-MPNNs^{+Q}. A natural question is how the use of extended neighborhoods affects the distinguishing power. Well, as it turns out, we can leverage Theorem 2 to gain insights. Indeed, consider a $\mathcal{P}$-pattern tree T^r (no edge-graphlets) underlying the characterization of $(\mathcal{P}_\varphi, \emptyset)$-MPNNs. Now, let Q^{r_1,r_2} be an edge-graphlet, which is used for the extended neighborhood. It now suffices to observe that the backbone S^r of T^r was a tree, because the standard neighbor set $N_G(v)$ was used. So, when using $N_{G,Q_\psi}(v)$ instead, it suffices to replace each edge in the backbone tree S^r by the edge-graphlet Q^{r_1,r_2}. For example, suppose that our backbone is  and we use the edge-graphlet is . Then, we need to consider  as backbone of our pattern tree instead. If additional vertex-graphlets are present, these need to be joined with the original (gray) backbone vertices. With this extended pattern-tree notion in place, Theorem 2 generalize to the case of extended neighborhoods.

To conclude, we believe that Graphlet MPNNs in all their forms and shapes are a promising and interesting way of extending the power of classical MPNNs. Moreover, pattern-tree based characterizations of distinguishing power are not only elegant, they are handy means for. reasoning over Graphlet MPNNs.

[2] Combining extended neighborhoods with edge-graphlet count features seems less natural.

REFERENCES

Pablo Barceló, Floris Geerts, Juan Reutter, and Maksimilian Ryschkov. Graph neural networks with local graph parameters. In *Advances in Neural Information Processing Systems*, volume 34, 2021. URL https://openreview.net/forum?id=yGKklt8wyV.

Giorgos Bouritsas, Fabrizio Frasca, Stefanos Zafeiriou, and Michael M. Bronstein. Improving graph neural network expressivity via subgraph isomorphism counting. *arXiv*, 2021. URL https://arxiv.org/abs/2006.09252.

Holger Dell, Martin Grohe, and Gaurav Rattan. Lovász meets weisfeiler and leman, 2018. URL https://arxiv.org/abs/1802.08876.

Zdeněk Dvořák. On recognizing graphs by numbers of homomorphisms. *Journal of Graph Theory*, 64(4):330–342, 2010. doi: https://doi.org/10.1002/jgt.20461.

Vijay Prakash Dwivedi, Chaitanya K Joshi, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. *arXiv preprint arXiv:2003.00982*, 2020.

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*, 2017. URL https://dl.acm.org/doi/10.5555/3305381.3305512.

Martin Grohe. word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data. In *Proceedings of the 39th Symposium on Principles of Database Systems*, pp. 1–16, 2020a. URL https://doi.org/10.1145/3375395.3387641.

Martin Grohe. Counting bounded tree depth homomorphisms. In *Proceedings of the 35th Symposium on Logic in Computer Science*, pp. 507–520, 2020b. URL https://doi.org/10.1145/3373718.3394739.

Andreas Krebs and Oleg Verbitsky. Universal covers, color refinement, and two-variable counting logic: Lower bounds for the depth. *arXiv*, 2015. URL https://arxiv.org/abs/1407.3175.

Huan Li, Boyuan Wang, Lixin Cui, Lu Bai, and Edwin R. Hancock. Lgl-gnn: Learning global and local information for graph neural networks. *Structural, Syntactic, and Statistical Pattern Recognition*, pp. 129–138, 2021. URL https://eprints.whiterose.ac.uk/173116/.

Christopher Morris, Martin Ritzert, Matthias Fey, William L. Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and Leman go neural: Higher-order graph neural networks. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence (AAAI)*, pp. 4602–4609, 2019. URL https://doi.org/10.1609/aaai.v33i01.33014602.

Natasa Przulj. Biological network comparison using graphlet degree distribution. *Bioinform.*, 23 (2):177–183, 2007. URL https://doi.org/10.1093/bioinformatics/btl301.

Nino Shervashidze, SVN Vishwanathan, Tobias Petri, Kurt Mehlhorn, and Karsten Borgwardt. Efficient graphlet kernels for large graph comparison. In *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, volume 5, pp. 488–495, 2009. URL http://proceedings.mlr.press/v5/shervashidze09a.html.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=ryGs6iA5Km.

A Appendix

A.1 Proof of main theorem

Proof of Theorem 2

Earlier we mentioned that classical MPNNs are completely characterized in expressive power by the Weisfeiler-Leman color refinement procedure. In Xu et al. (2019) this result is proven by demonstrating that every couple of graphs distinguishable by Weisfeiler-Leman will be embedded differently with the right chosen depth for a GNN provided that the aggregation and updating functions are injective. Moreover, a GNN architecture (called Graph Isomorphism Network or shortly GIN) is constructed that satisfies these conditions. While this result is originally formulated for the "basic" MPNNs without extensions, it also holds for MPNNs with extended features and the corresponding versions of the Weisfeiler-Leman procedure (see Barceló et al. (2021) for a full exposition on the extension of node features with homomorphism counts as an example). We remind the reader that the WL-procedure goes as follows:

$$l_v^{(t+1)} := \mathsf{Hash}\left(l_v^{(t)}, \{\{l_u^{(t)} \mid u \in \mathcal{N}_G(u)\}\}\right)$$

Any graph morphism types ϕ and ψ enable us to define a version of the Weisfeiler-Leman color refinement procedure that corresponds to MPNNs with extended node features and/or extended edge features based on structural information about the nodes and edges in the graph.

$$\begin{aligned} l_v^{(0)} &:= (\chi_v, \#\phi(P_1^r, G^v), \ldots, \#\phi(P_p^r, G^v) \\ l_{uv} &:= (\#\psi(Q_1^{r1r2}, G^{uv}), \ldots, \#\psi(Q_q^{r1r2}, G^{uv})) \\ l_v^{(t+1)} &:= \mathsf{Hash}\left(l_v^{(t)}, \left\{\!\!\left\{(l_u^{(t)}, l_{uv}) \mid u \in \mathcal{N}_G(v)\right\}\!\!\right\}\right) \end{aligned}$$

For pattern sets $\mathcal{P}, \mathcal{Q}$ and morphism types ϕ and ψ we call this the $(\mathcal{P}_\phi, \mathcal{Q}_\psi)$**-WL procedure**.

We show that for any finite collections $\mathcal{P}, \mathcal{Q}$ of patterns, graphs G and H, vertices $v \in V_G$ and $w \in V_H$, and $d \geq 0$:

$$(G, v) \equiv^{(d)}_{(\mathcal{P}_\phi, \mathcal{Q}_\psi)\text{-}\mathsf{WL}} (H, w) \iff \mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(T^r, H^w), \tag{1}$$

for every $(\mathcal{P}, \mathcal{Q})$-pattern tree T^r of depth at most d. Similarly,

$$G \equiv^{(d)}_{(\mathcal{P}_\phi, \mathcal{Q}_\psi)\text{-}\mathsf{WL}} H \iff \mathsf{hom}_{\varphi,\psi}(T, G) = \mathsf{hom}_{\varphi,\psi}(T, H), \tag{2}$$

for every (unrooted) $(\mathcal{P}, \mathcal{Q})$-pattern tree of depth at most d.

For a given set $\mathcal{P} = \{P_1^r, \ldots, P_p^r\}$ of patterns and $\mathbf{s} = (s_1, \ldots, s_p) \in \mathbb{N}^p$, we denote by $\mathcal{F}^{\mathbf{s}}$ the graph pattern of the form $(P_1^{s_1} \star \cdots \star P_p^{s_p})^r$, that is, we join s_1 copies of P_1, s_2 copies of P_2 and so on.

Proof of equivalence (1). The proof is by induction on the number of rounds d.

$\boxed{\Longrightarrow}$ We first consider the implication $(G, v) \equiv^{(d)}_{(\mathcal{P}_\phi, \mathcal{Q}_\psi)\text{-}\mathsf{WL}} (H, w) \implies \mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(T^r, H^w)$ for every $(\mathcal{P}, \mathcal{Q})$-pattern tree T^r of depth at most d.

<u>Base case.</u> Let us first consider the base case, that is, $d = 0$. In other words, we consider $(\mathcal{P}, \mathcal{Q})$-pattern trees T^r consisting of a single root r adorned with a pattern $\mathcal{F}^{\mathbf{s}}$ for some $\mathbf{s} = (s_1, \ldots, s_p) \in \mathbb{N}^p$. We note that due to the properties of the graph join operator:

$$\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \prod_{i \in [p]} \left(\#\phi(P_i^r, G^v)\right)^{s_i}. \tag{3}$$

Since, $(G,v) \equiv^{(0)}_{(P,Q)\text{-WL}} (H,w)$, we know that $\chi_G(v) = \chi_H(w) = a$ for some $a \in \Sigma$ and $\#\phi(P_i^r, G^v) = \#\phi(P_i^r, H^w)$ for all $P_i^r \in \mathcal{P}$. This implies that the product in (3) is equal to

$$\prod_{i\in[p]} \big(\#\phi(P_i^r, H^w)\big)^{s_i} = \mathsf{hom}_{\varphi,\psi}(T^r, H^w)$$

as desired.

Inductive step. Suppose next that we know that the implication holds for $d-1$. We assume now $(G,v) \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi\text{-WL}} (H,w)$ and consider an $(\mathcal{P},\mathcal{Q})$-pattern tree T^r of depth at most d. Assume that in the backbone of T^r, the root r has m children $c_1,\dots,c_m$, and denote by $T_1^{c_1},\dots,T_m^{c_m}$ the $(\mathcal{P},\mathcal{Q})$-pattern trees (of depth $\leq d-1$ in T^r rooted at c_i. Furthermore, we denote by $T_i^{(r,c_i)}$ the $(\mathcal{P},\mathcal{Q})$-pattern tree obtained from $T_i^{c_i}$ by attaching r to c_i; $T_i^{(r,c_i)}$ has root r. Let $\mathcal{F}^\mathbf{r}$ be the pattern in T^r associated with r and let $\mathcal{E}_i^{r_1r_2}$ be the edge-wise join of patterns attached to backbone edge (r,c_i) . The following equality obviously follows from the definition of a (ϕ,ψ)-homomorphism:

$$\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(\mathcal{F}^\mathbf{r}, G^v) \prod_{i\in[m]} \Big(\sum_{v'\in N_G(v)} \mathsf{hom}_{\varphi,\psi}(T_i^{c_i}, G^{v'})\mathsf{hom}_{\varphi,\psi}(\mathcal{E}_i^{r_1r_2}, G^{vv'})\Big). \tag{4}$$

Recall now that we assume $(G,v) \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H,w)$ and thus, in particular, $(G,v) \equiv^{(0)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}}$ (H,w). Hence, by induction, $\mathsf{hom}_{\varphi,\psi}(S^r, G^v) = \mathsf{hom}_{\varphi,\psi}(S^r, H^w)$ for every $\mathcal{F}$-pattern tree S^r of depth 0. In particular, this holds for $S^r = \mathcal{F}^\mathbf{r}$ and hence

$$\mathsf{hom}_{\varphi,\psi}(\mathcal{F}^\mathbf{r}, G^v) = \mathsf{hom}_{\varphi,\psi}(\mathcal{F}^\mathbf{r}, H^w).$$

Furthermore, $(G,v) \equiv^{(d)}_{(\mathcal{P}_phi,\mathcal{Q}_\psi)\text{-WL}} (H,w)$ implies that there exists a bijection $\beta : N_G(v) \to N_H(w)$ such that $(G,v') \equiv^{(d-1)}_{(\mathcal{P}_phi,\mathcal{Q}_\psi)\text{-WL}} (H,\beta(v'))$ and $\#\psi(Q_i^{r_1r_2}, G^{vv'}) = \#\psi(Q_i^{r_1r_2}, H^{ww'})$ for every $v' \in N_G(v)$. By induction, for every $v' \in N_G(v)$ there thus exists a unique $w' \in N_H(w)$ such that $\mathsf{hom}_{\varphi,\psi}(S^r, G^{v'}) = \mathsf{hom}_{\varphi,\psi}(S^r, H^{w'})$ for every $\mathcal{F}$-pattern tree S^r of depth at most $d-1$. In particular, for every $v' \in N_G(v)$ there exists a $w' \in N_H(w)$ such that

$$\mathsf{hom}_{\varphi,\psi}(T_i^{c_i}, G^{v'}) = \mathsf{hom}_{\varphi,\psi}(T_i^{c_i}, H^{w'})$$
$$\mathsf{hom}_{\varphi,\psi}(\mathcal{E}_i^{r_1r_2}, G^{vv'}) = \prod_{j\in[q]} \#\psi(Q_i^{r_1r_2}, G^{vv'})^{y_{ij}}$$
$$= \prod_{j\in[q]} \#\psi(Q_i^{r_1r_2}, H^{ww'})^{y_{ij}} = \mathsf{hom}_{\varphi,\psi}(\mathcal{E}_i^{r_1r_2}, H^{ww'})$$

for each of the sub-trees $T_i^{c_i}$ in T^r. Hence, equation 4 is equal to

$$\mathsf{hom}_{\varphi,\psi}(\mathcal{F}^\mathbf{r}, H^w) \prod_{i\in[m]} \Big(\sum_{w'\in N_H(w)} \mathsf{hom}_{\varphi,\psi}(T_i^{c_i}, H^{w'})\mathsf{hom}_{\varphi,\psi}(\mathcal{E}_i^{r_1r_2}, H^{ww'})\Big)$$

which in turn is equal to $\mathsf{hom}_{\varphi,\psi}(T^r, H^w)$, as desired.

$\boxed{\Longleftarrow}$ We next consider the other direction, that is, we show that when $\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(S^r, H^w)$ holds for every (P,Q)-pattern tree T^r of depth at most d, then $(G,v) \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H,w)$ holds. This is again verified by induction on d. This direction is more complicated and is similar to techniques used in Grohe (2020b). In our induction hypothesis we further include that a *finite* number of $(\mathcal{P},\mathcal{Q})$-pattern trees suffices to infer $(G,v) \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H,w)$ for graphs G and H and vertices $v \in V_G$ and $w \in V_H$.

Base case. Let us consider the base case $d = 0$ first. We need to show that $\chi_G(v) = \chi_H(w)$ and $\#\phi(P_i^r, G^v) = \#\phi(P_i^r, H^w)$ for every $P_i^r \in \mathcal{P}$, since this implies $(G, v) \equiv^{(0)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, w)$.

We first observe that $\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(T^r, H^w)$ for every $(\mathcal{P}, \mathcal{Q})$-pattern tree T^r of depth 0, implies that v and w must be assigned the same label, say a, by χ_G and χ_H, respectively.

Indeed, if we take T^r to consist of a single root r labeled with a (and thus r is associated with the pattern $\mathcal{F}^{\mathbf{0}}$), then $\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(T^r, H^w)$ will be one if $\chi_G(v) = \chi_H(w) = a$ and zero otherwise. This implies that $\chi_G(v) = \chi_H(w) = a$.

Next, we show that $\#\phi(P_i^r, G^v) = \#\phi(P_i^r, H^w)$ for every $P_i^r \in \mathcal{P}$. It suffices to consider the $\mathcal{P}$-pattern tree T_i^r consisting of a root r joined with a single copy of P_i^r:

$$\begin{aligned}\#\phi(P_i^r, G^v) &= \mathsf{hom}_{\varphi,\psi}(P_i^r, G^v)\\ = \mathsf{hom}_{\varphi,\psi}(P_i^r, H^w) &= \#\phi(P_i^r, H^w)\end{aligned}$$

We observe that we only need a finite number of $\mathcal{F}$-pattern trees to infer $(G, v) \equiv^{(0)}_{\mathcal{F}\text{-WL}} (H, w)$. Indeed, suppose that χ_G and χ_H assign labels $a_1, \ldots, a_L$, then we need L single vertex trees with no patterns attached and root labeled with one of these labels. In addition, we need one $\mathcal{F}$-pattern tree for each pattern $P_i^r \in \mathcal{F}$ and each label $a_1, \ldots, a_L$. That is, we need $L(p + 1)$ $(\mathcal{P}, \mathcal{Q})$-pattern trees of depth 0.

Inductive step. We now assume that the implication holds for $d - 1$ and consider trees of depth d. That is, we assume that if $\mathsf{hom}_{\varphi,\psi}(T^r, G^v) = \mathsf{hom}_{\varphi,\psi}(T^r, H^w)$ holds for every $\mathcal{F}$-pattern tree T^r of depth at most $d - 1$, then $(G, v) \equiv^{(d-1)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, w)$ holds. Furthermore, we assume that only a finite number K of $\mathcal{F}$-pattern trees $S_1^r, \ldots, S_K^r$ of depth at most $d - 1$ suffice to infer $(G, v) \equiv^{(d-1)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, w)$.

So, for d, let us assume that $\mathsf{hom}(T^r, G^v) = \mathsf{hom}(T^r, H^w)$ holds for every $\mathcal{F}$-pattern tree of depth at most d. We need to show $(G, v) \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, w)$ and that we can again assume that a finite number of $\mathcal{F}$-pattern trees of depth at most d suffice to infer $(G, v) \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, w)$.

By definition of $(G, v) \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, w)$, we can, equivalently, show that $(G, v) \equiv^{(d-1)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, w)$ and that there exists a bijection $\beta : N_G(v) \to N_H(w)$ such that $(G, v') \equiv^{(d-1)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} (H, \beta(v'))$ **and** $\#\psi(Q_i^{r_1 r_2}, G^{vv'}) = \#\psi(Q_i^{r_1 r_2}, H^{w\beta(v')})$ for every $v' \in N_G(v)$. That $(G, v) \equiv^{(d-1)}_{\mathcal{F}\text{-WL}} (H, w)$ holds, is by induction, since $\mathsf{hom}(T^r, G^v) = \mathsf{hom}(T^r, H^w)$ for every $(\mathcal{P}, \mathcal{Q})$-pattern tree of depth at most d and thus also for every $(\mathcal{P}, \mathcal{Q})$-pattern tree of depth at most $d - 1$. We may thus focus on showing the existence of the bijection β.

We know, by induction and the proof of the previous implication, that $(G, v) \equiv^{(d-1)}_{\mathcal{F}\text{-WL}} (H, w)$ if and only if $\mathsf{hom}_{\varphi,\psi}(S_i^r, G^v) = \mathsf{hom}_{\varphi,\psi}(S_i^r, H^w)$ for each $i \in K$. Denote by $R_1, \ldots, R_e$ the equivalence class on $V_G \cup V_H$ induced by $\equiv^{(d-1)}_{\mathcal{F}\text{-WL}}$ and the values of the counts $\#\psi(Q_i^{r_1 r_2}, G^{vv'})$ or $\#\psi(Q_i^{r_1 r_2}, H^{ww'})$. In other words: two vertices are in the same equivalence class if the K (ϕ, ψ)-homomorphism counts of the trees S_i and the q ψ-counts of the patterns Q_i are all the same. Furthermore, define $N_{j,X}(x) := N_X(x) \cap R_j$ and let $n_j = |N_{j,G}(v)|$ and $m_j = |N_{j,H}(w)|$ for $v \in V_G$ and $w \in V_H$, for each $j \in [e]$. If we can show that $n_j = m_j$ for each $j \in [e]$, then this implies the existence of the desired bijection.

Let $T_i^{r=a}$ be the $\mathcal{F}$-pattern tree of depth at most d obtained by attaching S_i^r to a new root vertex r labeled with a. We may assume that v and w both have label a, since their homomorphism counts for the single root trees with labels from Σ. The root vertex r is not joined with any $\mathcal{F}^{\mathbf{s}}$ (or alternatively it is joined with $\mathcal{F}^{\mathbf{0}}$). It will be convenient to denote the root of S_i^r by r_i instead of r. Then for each $i_1 \in [K]$:

$$\begin{aligned}\mathsf{hom}_{\varphi,\psi}(T_{i_1}^{r=a}, G^v) &= \sum_{v' \in N_G(v)} \mathsf{hom}_{\varphi,\psi}(S_i^{r_i}, G^{v'}) = \sum_{j\in[e]} n_j \mathsf{hom}_{\varphi,\psi}(S_i^{r_{i_1}}, G^{v'_j}) \\ &= \sum_{j\in[e]} m_j \mathsf{hom}_{\varphi,\psi}(S_{i_1}^{r_{i_1}}, H^{w'_j}) = \sum_{w'\in N_H(w)} \mathsf{hom}_{\varphi,\psi}(S_{i_1}^{r_{i_1}}, H^{w'}) \\ &= \mathsf{hom}_{\varphi,\psi}(T_{i_1}^{r=a}, H^w)\end{aligned}$$

where v'_j and w'_j denote arbitrary vertices in $N_{j,G}(v)$ and $N_{j,H}(w)$, respectively. Let us denote $\mathsf{hom}_{\varphi,\psi}(S_{i1}^{r_i}, G^{v'_j})$ by $a_{i_1 j}$ and observe that this is equal to $\mathsf{hom}_{\varphi,\psi}(S_{i1}^{r_{i1}}, H^{w'_j})$. Hence, we know that for each $i_1 \in [K]$:

$$\sum_{j\in[e]} a_{i_1 j} n_j = \sum_{j\in[e]} a_{i_1 j} m_j.$$

Similarly, for each $i_2 \in [q]$:

$$\begin{aligned}\mathsf{hom}_{\varphi,\psi}(Q_{i_2}^{r_1}, G^v) &= \sum_{v'\in N_G(v)} \mathsf{hom}_{\varphi,\psi}(Q_{i2}^{r_1 r_2}, G^{vv'}) = \sum_{j\in[e]} n_j \mathsf{hom}_{\varphi,\psi}(Q_{i2}^{r_1 r_2}, G^{vv'_j}) \\ &= \sum_{j\in[e]} m_j \mathsf{hom}_{\varphi,\psi}(Q_{i2}^{r_1 r_2}, H^{ww'_j}) = \sum_{w'\in N_H(w)} \mathsf{hom}_{\varphi,\psi}(Q_{i2}^{r_1 r_2}, H^{ww'}) \\ &= \mathsf{hom}_{\varphi,\psi}(T_{i_1}^{r=a}, H^w)\end{aligned}$$

and thus

$$\sum_{j\in[e]} q_{i_2 j} n_j = \sum_{j\in[e]} q_{i_2 j} m_j$$

for all $i_2 \in [q]$ if we denote $\mathsf{hom}_{\varphi,\psi}(Q_{i2}^{r_1 r_2}, G^{vv'_j}) = \mathsf{hom}_{\varphi,\psi}(Q_{i2}^{r_1 r_2}, H^{ww'})$ by $q_{i_2 j}$.

In what follows, we will denote the patterns $Q \in mathcalQ$ with $S_i^{r_i}$ with $i \in \{K+1, \ldots, K+q\}$ and the K $(d-1)$-depth pattern trees from the inductive step with $S_i^{r_i}$ with $i \in K$. This means that we can denote $q_{i,j} = a_{i+K,j}$ and that we won't need $q_i j$ in our notation and that the following holds for all $i \in [K+q]$:

$$\sum_{j\in[e]} a_{ij} n_j = \sum_{j\in[e]} a_{ij} m_j$$

We call a set $I \subseteq [K+q]$ compatible if the following conditions are satisfied:

- For all $i \in I \cap [K]$, the corresponding trees have the same label in the root.
- If the pattern set $\mathcal{Q}$ consists of labeled patterns (i.o.w.: if preserving labels is a condition of the ψ-type morphisms) : For all $i \in I \cap \{K+1, \ldots, K+q\}$ the corresponding patterns have label a in the first root and the same label as the roots of depth $d-1$ in the second root.

Consider a vector $\mathbf{s} = (s_1, \ldots, s_K, \ldots, s_{K+q}) \in \mathbb{N}^{K+q}$ and define its support as $\mathsf{supp}(\mathbf{s}) := \{i \in [K+q] \mid s_i \neq 0\}$. We say that $\mathbf{s}$ is compatible if its support is. For such a compatible $\mathbf{s}$ we now define $T^{r=a,\mathbf{s}}$ to be the $\mathcal{F}$-pattern tree with root r labeled with a, with one child c which is joined with (and inheriting the label from) the following $\mathcal{F}$-pattern tree of depth $d-1$:

$$\star_{i\in\mathsf{supp}(\mathbf{s})\cap[K]} S_i^{s_i}.$$

The edge (r, c) is joined with the corresponding number of patterns from $\mathcal{Q}$:

$$\star_{i\in\mathsf{supp}(\mathbf{s})\cap\{K+1,\ldots,K+q\}} S_i^{s_i}.$$

In other words, we simply join together powers of the K $(d-1)$-deep pattern trees $S_i^{r_i}$'s that have roots with the same label, whereas we join the edge patterns on the edge (r,c). Then for every compatible $\mathbf{s} \in \mathbb{N}^{[K+q]}$:

Then for every compatible $\mathbf{s} \in \mathbb{N}^{[K+q]}$:

$$\begin{aligned}
&\mathsf{hom}_{\varphi,\psi}(T^{r=a,\mathbf{s}}, G^v)\\
&= \sum_{v' \in N_G(v)} \prod_{i\in[K+1,K+q]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_1r_2}, G^{vv'})\big)^{s_i} \prod_{i\in[K]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_i}, G^{v'})\big)^{s_i}\\
&= \sum_{j\in[e]} n_j \prod_{i\in[K+1,K+q]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_1r_2}, G^{vv'_j})\big)^{s_i} \prod_{i\in[K]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_i}, G^{v'_j})\big)^{s_i}\\
&= \sum_{j\in[K+q]} m_j \prod_{i\in[K+1,K+q]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_1r_2}, H^{ww'_j})\big)^{s_i} \prod_{i\in[K]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_i}, H^{w'_j})\big)^{s_i}\\
&= \sum_{w' \in N_H(w)} \prod_{i\in[K+1,K+q]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_1r_2}, H^{ww'})\big)^{s_i} \prod_{i\in[K+q]} \big(\mathsf{hom}_{\varphi,\psi}(S_i^{r_i}, H^{w'})\big)^{s_i}\\
&= \mathsf{hom}_{\varphi,\psi}(T_i^{r=a,\mathbf{s}}, H^w)
\end{aligned}$$

where, as before, v'_j and w'_j denote arbitrary vertices in $N_{j,G}(v)$ and $N_{j,H}(w)$, respectively. Hence, for any compatible $\mathbf{s} \in \mathbb{N}^{[K+q]}$:

$$\sum_{j\in[e]} n_j \prod_{i\in[K+q]} a_{ij}^{s_i} = \sum_{j\in[e]} m_j \prod_{i\in[K+q]} a_{ij}^{s_i}.$$

We now continue in the same way as in the proof of Lemma 4.2 in Grohe (2020b). We repeat the argument here for completeness. Define $\mathbf{a}_j^{\mathbf{s}} := \prod_{i\in[K+q]} a_{ij}^{s_i}$ for each $j \in [e]$. We assume, for the sake of contradiction, that there exists a $j \in [e]$ such that $n_j \neq m_j$. We choose such a $j_0 \in [e]$ for which $S = \mathsf{supp}(\mathbf{a}_{j_0})$ is inclusion-wise maximal.

We first rule out that $S = \emptyset$. Indeed, suppose that $S = \emptyset$. This implies that $\mathbf{a}_{j_0} = \mathbf{0}$. Now observe that $\mathbf{a}_j$ and $\mathbf{a}_{j'}$ are mutually distinct for all $j, j' \in [e]$, $j \neq j'$. Indeed, if they were equal then this would imply that $R_j = R_{j'}$. Hence, $\mathsf{supp}(\mathbf{a}_j) \neq \emptyset$ for any $j \neq j_0$. We note that $n_j = m_j$ for all $j \neq j_0$ by the maximality of S. Hence, $n_{j_0} = n - \sum_{j\neq j_0} n_j = n - \sum_{j\neq j_0} m_j = m_{j_0}$, contradicting our assumption. Hence, $S \neq \emptyset$.

Consider $J := \{j \in [e] \mid \mathsf{supp}(\mathbf{a}_j) = S\}$. For each $j \in J$, consider the truncated vector $\hat{\mathbf{a}}_j := (a_{ij} \mid i \in S)$. We note that $\hat{\mathbf{a}}_j$, for $j \in J$, all have positive entries and are mutually distinct. Lemma 4.1 in Grohe (2020b) implies that we can find a vector (with non-zero entries) $\hat{\mathbf{s}} = (\hat{s}_i \mid i \in S)$ such that the numbers $\hat{\mathbf{a}}_j^{\hat{\mathbf{s}}}$ for $j \in J$ are mutually distinct as well. We next consider $\mathbf{s} = (s_1, \ldots, s_K)$ with $s_i = \hat{s}_i$ if $i \in S$ and $s_i = 0$ otherwise. Then by definition of $\hat{\mathbf{s}}$, also $\mathbf{a}_j^{\mathbf{s}}$ for $j \in J$ are mutually distinct.

We next note that for every $p \in \mathbb{N}$, $\mathbf{a}_j^{p\mathbf{s}} = (\mathbf{a}_j^{\mathbf{s}})^p$ and if we define $\mathbf{A}$ to be the $|J| \times |J|$-matrix such that $A_{jj'} := \mathbf{a}_j^{j'\mathbf{s}}$ then this will be an invertible matrix (Vandermonde). We use this invertibility to show that $n_{j_0} = m_{j_0}$.

Let $\mathbf{n}_J := (n_j \mid j \in J)$ and $\mathbf{m}_J = (m_j \mid j \in J)$. If we inspect the j'th entry of $\mathbf{n}_J \cdot \mathbf{A}$, then this is equal to

$$\sum_{j\in J} n_j \mathbf{a}_j^{j'\mathbf{s}} = \sum_{j\in[e]} n_j \mathbf{a}_j^{j'\mathbf{s}} - \sum_{\substack{j\in[e]\\ S\not\subseteq \mathsf{supp}(\mathbf{a}_j)}} n_j \mathbf{a}_j^{j'\mathbf{s}} - \sum_{\substack{j\in[e]\\ S\subset \mathsf{supp}(\mathbf{a}_j)}} n_j \mathbf{a}_j^{j'\mathbf{s}}.$$

We want to reduce the above expression to

$$\sum_{j\in J} n_j \mathbf{a}_j^{j'\mathbf{s}} = \sum_{j\in[e]} n_j \mathbf{a}_j^{j'\mathbf{s}} - \sum_{\substack{j\in[e]\\ S\subset \mathsf{supp}(\mathbf{a}_j)}} n_j \mathbf{a}_j^{j'\mathbf{s}}.$$

To see that this holds, we verify that when $S \not\subseteq \mathsf{supp}(\mathbf{a}_j)$ then $\mathbf{a}_j^{j'\mathbf{s}} = 0$. Indeed, take an $\ell \in S$ such that $\ell \notin \mathsf{supp}(\mathbf{a}_j)$. Then, $\mathbf{a}_j^{j'\mathbf{s}}$ contains the factor $a_{\ell j}^{j's_\ell} = 0^{s_\ell}$ with $s_\ell = \hat{s}_\ell \neq 0$. Hence, $\mathbf{a}_j^{j'\mathbf{s}} = 0$.

Now, by the maximality of S, for all j with $S \subset \mathsf{supp}(\mathbf{a}_j)$ we have $n_j = m_j$ and thus

$$\sum_{\substack{j\in[e]\\ S\subset\mathsf{supp}(\mathbf{a}_j)}} n_j \mathbf{a}_j^{j'\mathbf{s}} = \sum_{\substack{j\in[e]\\ S\subset\mathsf{supp}(\mathbf{a}_j)}} m_j \mathbf{a}_j^{j'\mathbf{s}}.$$

Since $\sum_{j\in[e]} n_j \mathbf{a}_j^{j'\mathbf{s}} = \sum_{j\in[e]} m_j \mathbf{a}_j^{j'\mathbf{s}}$, we thus also have that

$$\sum_{j\in J} n_j \mathbf{a}_j^{j'\mathbf{s}} = \sum_{j\in J} m_j \mathbf{a}_j^{j'\mathbf{s}}.$$

Since this holds for all $j' \in J$, we have $\mathbf{n}_J \cdot \mathbf{A} = \mathbf{m}_J \cdot \mathbf{A}$ and by the invertibility of $\mathbf{A}$, $\mathbf{n}_J = \mathbf{m}_J$. In particular, since $j_0 \in J$, $n_{j_0} = m_{j_0}$ contradicting our assumption.

As a consequence, $n_j = m_j$ for all $j \in [e]$ and thus we have our desired bijection.

It remains to verify that we only need a finite number of $\mathcal{F}$-pattern trees to conclude that $n_j = m_j$ for all $j \in [e]$. In fact, the above proof indicates that we just need to check test for each root label a, we need to check identities for the finite number of pattern trees used to define the matrix $\mathbf{A}$.

Proof of equivalence 2 This equivalence is shown just like proof of Theorem 4.4. in Grohe (2020a). in Grohe (2020b).

$\boxed{\Longrightarrow}$ We first show that $G \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} H$ implies $\mathsf{hom}_{\varphi,\psi}(T,G) = \mathsf{hom}_{\varphi,\psi}(T,H)$ for unrooted $(\mathcal{P},\mathcal{Q})$-pattern trees T of depth at most d.

Assume that $V_X \cap V_Y = \emptyset$ for $X, Y \in \{G, H\}$. For $x \in V_X$ and $y \in V_Y$, define $x \sim_d y$ if and only if $\mathsf{hom}(T^r, X^x) = \mathsf{hom}(T^r, Y^y)$ for all $\mathcal{F}$-pattern trees T^r of depth at most d. Let $R_1, \ldots, R_e$ be the $\sim_d$-equivalence classes and for each $j \in [e]$, let $p_j := |R_j \cap V_G|$ and $q_j := |R_j \cap V_H|$. Suppose that $G \equiv^{(d)}_{\mathcal{F}\text{-WL}} H$. This implies that $p_j = q_j$ for every $j \in [e]$.

Let T be an unrooted $(\mathcal{P},\mathcal{Q})$-pattern tree of depth at most d, let r be any vertex on the backbone of T, and let T^r be the rooted $(\mathcal{P},\mathcal{Q})$-pattern tree obtained from T by declaring r as its root. By definition, for $X \in \{G, H\}$, any $x \in V_X \cap R_j$, $\mathsf{hom}_{\varphi,\psi}(T^r, X^x)$ are all the same number, only dependent on $j \in [e]$. Hence,

$$\begin{aligned}\mathsf{hom}_{\varphi,\psi}(T,G) &= \sum_{v\in V(G)} \mathsf{hom}_{\varphi,\psi}(T^r,G^v) = \sum_{j\in[e]} p_j \mathsf{hom}_{\varphi,\psi}(T^r,G^{v_j}) \\ &= \sum_{j\in[e]} q_j \mathsf{hom}_{\varphi,\psi}(T^r,H^{w_j}) = \sum_{w\in V(H)} \mathsf{hom}_{\varphi,\psi}(T^r,H^w) = \mathsf{hom}_{\varphi,\psi}(T,H),\end{aligned}$$

where v_j and w_j are arbitrary vertices in $R_j \cap V_G$ and $R_j \cap V_H$, respectively, and where we used that $\mathsf{hom}_{\varphi,\psi}(T^r, G^{v_j}) = \mathsf{hom}_{\varphi,\psi}(T^r, H^{w_j})$ and $p_j = q_j$. Since this holds for any unrooted $(\mathcal{P},\mathcal{Q})$-pattern tree T of depth at most d, we have shown the desired implication.

$\boxed{\Longleftarrow}$ We next check the other direction. That is, we assume that $\mathsf{hom}_{\varphi,\psi}(T,G) = \mathsf{hom}_{\varphi,\psi}(T,H)$ holds for any unrooted $(\mathcal{P},\mathcal{Q})$-pattern tree T of depth at most d and verify that $G \equiv^{(d)}_{(\mathcal{P}_\phi,\mathcal{Q}_\psi)\text{-WL}} H$.

For $x \sim_d y$ to hold for $x \in V_X$, $y \in V_Y$ and $X, Y \in \{G, H\}$, we earlier showed that this corresponds to checking whether $\mathsf{hom}_{\varphi,\psi}(T_i^{r_i}, X^x) = \mathsf{hom}_{\varphi,\psi}(T_i^{r_i}, Y^y)$ for a *finite* number K rooted $(\mathcal{P},\mathcal{Q})$-pattern trees $T_i^{r_i}$. By definition of the R_j's, $a_{ij} := \mathsf{hom}_{\varphi,\psi}(T_i^{r_i}, X^x)$ for $x \in R_j$ is well-defined (independent of the choice of $X \in \{G, H\}$ $x \in V_X$). For the rooted $T_i^{r_i}$'s we denote by T_i its unrooted version. Similarly as before,

$$\mathsf{hom}_{\varphi,\psi}(T_i,G) = \sum_{j\in[e]} a_{ij} p_j = \sum_{j\in[e]} a_{ij} q_j = \mathsf{hom}_{\varphi,\psi}(T_i,H).$$

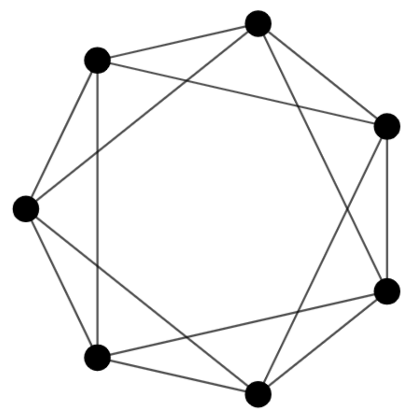


Figure 1: A circular skip graph with 7 nodes and skip length 2

We next show that $p_j = q_j$ for $j \in [e]$. In fact, this is shown in precisely the same way as in our previous characterisation and based on Lemma 4.2 in Grohe (2020b). That is, we again consider trees obtained by joining copies of the T_i's, to obtain, for compatible $\mathbf{s} \in \mathbb{N}^K$,

$$\sum_{j\in[e]} a_{ij}^{s_i} p_j = \sum_{j\in[e]} a_{ij}^{s_i} q_j.$$

It now suffices to repeat the same argument as before (details omitted). □

A.2 Detailed comparisons of different approaches

Before introducing the examples substantiating the comparisons between different extensions of MPNN models , we briefly define the *Circular Skip Graphs*. This is a graph consisting of n nodes forming a cycle and additional edges making 'skip connections' between nodes divided by k steps in the cycle. As an example, see the graph CSL(7,2) in figure 1.

The computed numbers of homomorphism and subgraph isomorphisms in the propositions of this section can be verified by self implemented code (available in [3]).

We will use the notation $Cl_1 > Cl_2$ to denote that an instance from class Cl_1 of MPNNs is able to distinguish a pair of vertices that any instance of class Cl_2 will not be able to distinguish.

At first, we prove that neither homomorphism or subgraph isomorphism counts of the same pattern set result in a strictly stronger model.

Proposition 4. *There exist vertices v_1 and v_2 in graphs G and H and a pattern set $\mathcal{P}$ such that v_1 and v_2 that can not be distinguished by $(\mathcal{P}_{\mathsf{sub}}, \emptyset)$-MPNNs but are distinguishable by some $(\mathcal{P}_{\mathsf{hom}}, \emptyset)$-MPNN, as well as the other way around.*

Proof. While such graphs are 4-regular graphs, CSL graphs of the same size but with mutually prime skip connection numbers k are not isomorphic. These graphs can be used to test the expressive power of GNN models (as in Dwivedi et al. (2020)), as every pair of CSL graphs with a fixed number of nodes is WL-indistinguishable. Moreover, any pair of vertices in such graphs are also WL-indistinguishable.

1. $(\mathcal{P}_{\mathsf{hom}}, \emptyset)$-MPNN $>$ $(\mathcal{P}_{\mathsf{sub}}, \emptyset)$-MPNNs
   Consider the graph $G = C_3 \cup C_3 \cup C_3 \cup C_3 \cup C_3$ and $H = C_5 \cup C_5 \cup C_5$. For any vertex v in G and w in H: $\mathsf{sub}(C_6^r, G^v) = \mathsf{sub}(C_6^r, G^v) = 0$, which means that those two are indistinguishable by $(\{C_6\}_{\mathsf{sub}}, \emptyset)$-MPNNs. However: $\mathsf{hom}(C_6^r, G^v) = 22 \neq 20 =$

[3]https://github.com/AnonSubmits/ICLRWorkshop2022

$\mathsf{hom}(C_6^r, H^w)$, which means that a $(\{C_6\}_{\mathsf{hom}}, \emptyset)$-MPNN exists that can distinguish those vertices.

2. $(\mathcal{P}_{\mathsf{sub}}, \emptyset)$-MPNN $>$ $(\mathcal{P}_{\mathsf{hom}}, \emptyset)$-MPNNs

   Consider the graph $G = \text{CSL}(7,2) \cup \text{CSL}(7,2)$ and $H = \text{CSL}(14,2)$. For any vertex v in G and any vertex w in H, $\mathsf{hom}(C_6^r, G^v) = \mathsf{hom}(C_6^r, H^w)$ but $\mathsf{sub}(C_6^r, G^v) \neq \mathsf{sub}(C_6^r, H^w)$. This means that these vertices are indistinguishable by $(\{C_{10}\}_{\mathsf{hom}}, \emptyset)$-MPNNs but not by $(\{C_{10}\}_{\mathsf{sub}}, \emptyset)$-MPNNs

□

Theorem 5. *1.* $(\emptyset, \mathcal{P}_\phi)$-MPNN*s are strictly stronger in distinguishing power than* $(\mathcal{P}_\phi, \emptyset)$-MPNN*s.*

2. $(\mathcal{P}_\phi, \mathcal{P}_\phi)$-$MPNNs$ *are exactly as strong as* $(\emptyset, \mathcal{P}_\phi)$-$MPNNs$

Proof. 1. We proof the first part of this proposition by demonstrating two things:

- $(\emptyset, \mathcal{P}_\phi)$-MPNNs $>$ $(\mathcal{P}_\phi, \emptyset)$-MPNNs
- Every pair of vertices undistinguishable by $(\emptyset, \mathcal{P}_\phi)$-MPNNs is indistinguishable by $(\mathcal{P}_\phi, \emptyset)$-MPNNs.

Consider the graph G and vertices v and w, illustrated in 2. The ”inner circle” is a 7-cycle with a skip link connection of length 2, the ”outer circle” a 7-cycle with a skip link connection of length 3. Node on corresponding position at the inside and the outside are joined with an edge. The result is a 5-regular graph of 14 nodes which implies that if all labels are identical no pair of vertices in this graph can be distinguished by the Weisfeiler-Leman procedure within any number of rounds. A computational verification shows that $\mathsf{sub}(C_4, G^v) = \mathsf{sub}(C_4, G^w) = 20$, we also observe that this means that $\mathsf{sub}(C_4, G^u) = 20$ for any vertex u in the graph as v and w are representatives of the two orbits of $V(G)$ induced by the action of the graph automorphism group. As a consequence, if $\mathcal{P} = C_4$ than v and w are indistinguishable by $(\mathcal{P}_{\mathsf{sub}}, \emptyset)$-$MPNNs$

On the other hand: if we consider the edges in v and w (depictioned in bold) and compute the numbers $\mathsf{sub}(C_4, G^{vv'})$ and $\mathsf{sub}(C_4, H^{ww'})$ we get the multisets $\{\{4,4,5,5,2\}\}$ and $\{\{6,6,3,4,2\}\}$. This means that v and w are distinguishable by $(\emptyset, \mathcal{P}_{\mathsf{sub}})$-MPNN.

We proof the general claim by induction. Specifically, we proof the following:

For any vertices v, w in the respective graphs G and H the following holds for all $d \in \mathbb{N}$

$$(G, v) \equiv^{(d)}_{(\emptyset, \mathcal{P}_\phi)\text{-WL}} (H, w) \Rightarrow (G, v) \equiv^{(d-1)}_{(\mathcal{P}_\phi, \emptyset)\text{-WL}} (H, w)$$

We start with the base case $d = 1$

Suppose $(G, v) \equiv^{(1)}_{(\emptyset, \mathcal{P}_\phi)\text{-WL}} (H, w)$: this means that

$$\begin{aligned} \big(l_v^0, \{\{(l_{v'}^0, l_{vv'}) \mid v' \in \mathcal{N}_G(v)\}\}\big) &= \big(l_w^0, \{\{(l_{w'}^0, l_{ww'}) \mid w' \in \mathcal{N}_H(w)\}\}\big) \\ &\Downarrow \\ \Big(l_v^0, \Big\{\Big\{(l_{v'}^0, (\#\phi(P_i^{r_1 r_2}, G^{vv'}))_{i\in[p]}) \mid v' \in \mathcal{N}_G(v)\Big\}\Big\}\Big) & \\ = \Big(l_w^0, \Big\{\Big\{(l_{w'}^0, (\#\phi(P_i^{r_1 r_2}, H^{ww'}))_{i\in[p]}) \mid w' \in \mathcal{N}_H(w)\Big\}\Big\}\Big) & \end{aligned}$$

where l_v^0 and l_w^0 denote the original labeling in the graph. It is immediately clear that this implies

$$\big(l_v^0, (\#\phi(P_i^{r_1}, G^v))_{i\in[p]}\big) = \big(l_w^0, (\#\phi(P_i^{r_1}, H^w))_{i\in[p]}\big)$$

as

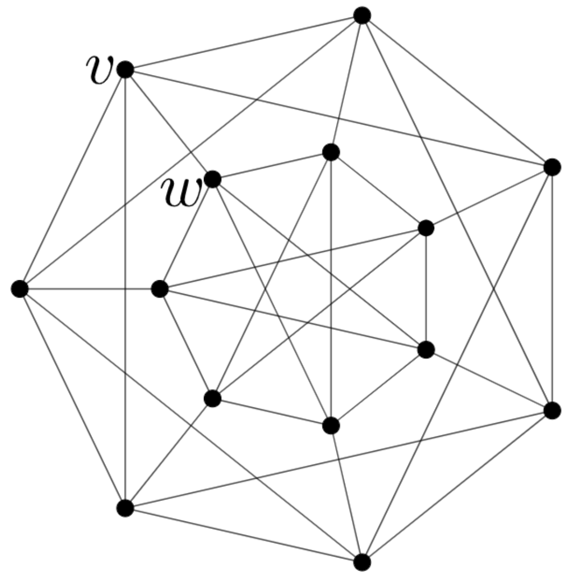


Figure 2: $(\emptyset, \mathcal{P}_\phi)$-MPNNs are strictly stronger in distinguishing power than $(\mathcal{P}_\phi, \emptyset)$-MPNNs

$$\forall i \in [p] : \#\phi(P_i^{r_1}, G^v) = \sum_{v' \in \mathcal{N}_G(v)} \#\phi(P_i^{r_1 r_2}, G^{vv'})$$
$$= \sum_{w' \in \mathcal{N}_G(w)} \#\phi(P_i^{r_1 r_2}, H^{ww'}) = \#\phi(P_i^{r_1}, H^w)$$

Hence, $(G, v) \equiv^{(0)}_{(\mathcal{P}_\phi,\emptyset)\text{-WL}} (H, w)$

In the inductive case we assume that $(G, v) \equiv^{(d+1)}_{(\emptyset,\mathcal{P}_\phi)\text{-WL}} (H, w)$.

This implies that $(G, v) \equiv^{(d)}_{(\emptyset,\mathcal{P}_\phi)\text{-WL}} (H, w)$ and the existence of a bijection β between $\mathcal{N}_G(v)$ and $\mathcal{N}_H(w)$ mapping every v' to a $\beta(v')$ such that $(G, v') \equiv^{(d)}_{(\emptyset,\mathcal{P}_\phi)\text{-WL}} (H, \beta(v'))$.

By the induction hypothesis this implies that $(G, v) \equiv^{(d-1)}_{(\mathcal{P}_\phi,\emptyset)\text{-WL}} (H, w)$ and $(G, v') \equiv^{(d-1)}_{(\mathcal{P}_\phi,\emptyset)\text{-WL}} (H, \beta(v')$ By the definition of the Weisfeiler-Leman procedure this means that $(G, v) \equiv^{(d)}_{(\mathcal{P}_\phi,\emptyset)\text{-WL}} (H, w)$.

2. It is obvious that $(\mathcal{P}_\phi, \mathcal{P}_\phi)$-WL distinguishes all pairs disitinguishable by $(\emptyset, \mathcal{P}_\phi)$-WL. The other direction is verified in exactly the same way as in the first part of this proposition.

□

A.3 Details and proof on Pattern-induced neighborhoods

In the classical Weisfeiler-Leman color refinement procedure, a node color in each next iteration is determined by aggregating the neighbors' colors from the previous round. In the main paper we described and $(\mathcal{P}_\phi \mathcal{Q}_\psi)$-MPNNs and the proof of theorem 2 introduced the corresponding color refinement procedure. While these extensions add local information to the node and edge labels information is still propagated along the edges of the graph. In both cases this results in the possibility to detect (albeit extended with pattern "leaves") tree-shaped structures. Suppose now that we would be interested to propagate information not along edges, but between couples of nodes who appear together in a specific pattern. This kind of label propagation gives raise to a color refinement procedure and corresponding message passing neural networks.

Li et al. (2021) combines the aggregation of features according to a classical MPNN updating rule with a "motif-neighborhood", which essentially comes down to the approach analyzed in this section. They note that this enables the model to simultaneously learn global graph information and local structural information pertaining to a particular nodes and to distill the necessary combination of this two types of information by introducing an attention mechanism.

In this section we would like to determine what kind of graph structures such frameworks would be able to detect. Let us firstly introduce the necessary concepts and notations to formalize this notion.

Let $G = (V, E)$ be a graph and let P^{r_1,r_2} be a two-rooted graph pattern. Suppose that a ϕ-type morphism is a type of graph morphisms such that (homomomorphisms and subgraph isomorphisms are both examples of such types) for all pairs of vertices v, w in G the following sets have the same cardinality:

$$\#\{f : V(P) \to V(G) \mid f(r_1) = v, f(r_2) = w \text{ and } f \in \Phi\}$$
$$=$$
$$\#\{f : V(P) \to V(G) \mid f(r_1) = w, f(r_2) = v \text{ and } f \in \Phi\}$$

where we denoted the set of all graph morphisms satisfying the conditions imposed by a ϕ-type by Φ. We denote the cardinality of these sets by $\#\phi(P^{r_1 r_2}, G^{vw})$ and the sets themselves with $\Phi(P^{r_1 r_2}, G^{vw})$

For every $v \in V$ we can construct a tree-shaped graph $G_{v,d,P}$ of depth d in the following way:

1. The root vertex of $G_{v,d,P}$ is a vertex with the same label l_v as v.
2. For all $w \in V$ with label l_w for which $\#\phi(P^{r_1 r_2}, G^{vw}) = k$, an edge from the root vertex to a new vertex with label l_w is added k times.
3. Repeat the previous step for all vertices on the second level. This means computing all numbers $\#\phi(P^{r_1,r_2}, G^{w,z}) = k$ and adding the corresponding number of edges from level 2 to level 3. An important remark here is that if a vertex already has an edge to a vertex with label l_z (i.e. the root vertex from step 1 has label l_z), only $k - 1$ vertices with label l_z are added on level 3.
4. Complete this tree in this way until the depth is d.
5. We define a mapping $\alpha : V(G_{v,d,P}) \to V(G)$ mapping a vertex in this tree to a vertex in G with the same label.

The previous construction and the presence of identyfying labels in G ensures us that for all vertices with label l_u in any $G_{v,d,P}$ (independent of the choice of v) $\alpha_v(u)$ will always be the same vertex in $V(G)$. This is why we will denote this mapping by α in the proofs.

This can be seen as an unfolding of this graph according to the neighborhood structure implied by the common appearance of vertices in patterns P inside the graph G. The introduction of the morphism type ϕ allows us to prove a general result, independent of the type of pattern counts used to define the neighborhoods.

Example 6. *Consider the following couple G and P^{r_1,r_2} in figure 3. Applying the steps described above results in the tree $G_{v,2,P}$*

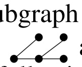
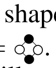
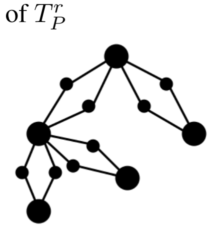
Let now T^r be some rooted, directed tree and P^{r_1,r_2} a two-rooted symmetric graph pattern as before. The associated graph T^r_P is constructed by replacing every edge with a subgraph of the same shape as P_{r_1,r_2}. Consider the following couple $T^r =$  and $P^{r_1,r_2} =$ . Applying the steps described above results in the following graph T^r_P illustrated in figure 4. We call this kind of graph a **P-tree** and define a notion corresponding type of graph mapping:

Figure 4: Example of T^r_P



Definition 7. A ϕ-homomorphims from a rooted P-tree T^r_P to a rooted graph G^v is a mapping $f : V(T_P) \to V(G)$ such that $f(r) = v$ and that the

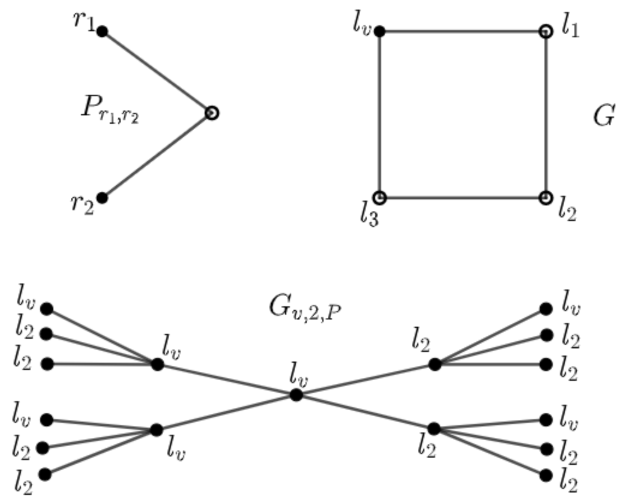


Figure 3: An example of $G_{v,d,P}$

restriction $f\big|_P$ is a ϕ-type morphism on every subgraph of shape P connecting nodes on the "virtual backbone" of T_P^r. We denote the number of such ϕ-homomorphisms with $\phi - \mathsf{hom}(T_P^r, G^v)$ and remark that it bears the same properties as the (ϕ, ψ)-homomorphisms from the previous sections of this paper.

Lemma 8. *For any vertex t on the "virtual backbone" of a P-tree T_P the number of $\phi-\mathsf{hom}(T_P, G)$ can be factores into a product as:*

$$\phi - \mathsf{hom}(T_P^t, G^w) = \phi - \mathsf{hom}(T_{1_P}^t, G^w) \cdot \phi - \mathsf{hom}(T_{2_P}^t, G^w)$$

where T_{1_P} and T_{2_P} are the two P-trees obtained by cutting T_P at t.

Proof. This holds because the definition of a ϕ-homomorphism of a P-tree imposes no conditions on the images of different P-shaped parts of the tree: i.e. the number of mappings $\phi - \mathsf{hom}(T_{1_P}^t, G^w)$ and $\phi - \mathsf{hom}(T_{2_P}^t, G^w)$ do not depend on each other. □

These constructions bear the following usefull property. Suppose that we use this new notion of neighborhood to define a color refinement procedure, called the P-WL-procedure, in the following way:

$$\chi_{P,G,v}^{(0)} := \big(\chi_G(v)\big)$$
$$\chi_{P,G,v}^{(d)} := \textsc{Hash}\big(\chi_{\mathcal{F},G,v}^{(d-1)}, \{\{\chi_{\mathcal{F},G,u}^{(d-1)} \mid u = h(r_2) \text{ for some } h \in \Phi(P^{r_1,r_2}, Gv, w)\}\}\big), \text{ for } d > 0.$$

We will denote equivalence according to this color refinement procedure after d steps by $\sim_P^d$ and equivalence after d steps of the "classical" WL (A.1) by $\equiv^d$. The following proposition shows that performing this color refinement procedure on the original graph is the same as the "classical" WL on the tree graph $G_{v,d,P}$.

Proposition 9. *For any two graphs G, H and vertices $v \in G$ and $w \in H$ the following equivalence holds:*

$$v \in G \sim_P^d w \in H \Leftrightarrow v \in G_{v,d,P} \equiv^d w \in H_{w,d,P}$$

Proof. The essence of the proof comes down to the fact that in both color refinement schemes, WL (see equation A.1) in the original graphs and P-WL in the unfoldings, the (multi)sets of neighbors have a one-to-one correspondence and have the same multisets of labels. To make this work technically, we introduce some definitions and assume that G and H have some unique node identifiers.

The construction of $G_{v,d,P}$ establishes an important correspondence between two "neighborhood sets". On one hand we have a multiset of vertices of G, the so-called P-neighborhood of a vertex v:

$$\mathcal{N}_{G,P}(v) = \{\{h(r_2) \in V(G) \mid h \in \Phi(P^{r_1,r_2}, G^{v,\cdot})\}\}$$

On the other, the neighbors of any non-leaf vertex with label l_v in the tree $G_{v,d,P}$.

To use these correspondences in out proofs we define a mapping α_v from the vertices of $G_{v,d,P}$ to G. By using the unique node identifiers (labels) in G, the following is a well defined function:

$$\begin{aligned} V(G_{v,d,P}) &\longrightarrow V(G) \\ x \text{ with label } l_v &\longmapsto v \text{ with label } l_v \end{aligned}$$

Moreover, the remark in the third step of the constructions ensures that this correspondence holds no matter which vertex was chosen to be the root of the "unfolding" of G. In other words a non-leaf vertex with some label l_v has exactly the same neighbors in $G_{u,d,P}$ for all $u \in V(G)$. We also note that α_v can be denoted as α as it only depends on the vertex label and hence well-defined and independent of v . We can observe the following things about any vertex $u \in G$:

$$\left|\mathcal{N}_{G_{v,d,P}}(u)\right| = \|\{\{\alpha(\mathcal{N}_{G_{v,d,P}}(u))\}\}\| = \|\{\{N_{G,P}(\alpha(u))\}\}\|$$

Moreover: $\{\{\alpha(\mathcal{N}_{G_{v,d,P}}(u))\}\} = \{\{N_{G,P}(\alpha(u))\}\}$

Exactly the same constructions are defined for H and its vertices and we define the mappings by α for readibility.

We proof the proposition by induction on the depth d.

Our induction hypothesis is twofold:

1. $v \in G \sim_P^{d-1} w \in H \Leftrightarrow v \in G_{v,d,P} \equiv^{d-1} w \in H_{w,d,P}$
2. $\alpha(x) \in G \sim_P^{d-1} \alpha(y) \Leftrightarrow x \equiv^{d-1} y$. Where $x \in G_{u_1,d,P}$ and $y \in H_{u_2,d,P}$ and the statement is independent of the chosen $u_1 \in G, u_2 \in H$

By definition of $G_{v,d,P}$ and $H_{v,d,P}$ and the remarks about α, it is clear that the base case $d = 1$ is trivial.

Suppose now that $v \in G \sim_P^d w \in H$, this means that:

$$v \in G \sim_{\mathsf{PWL}}^{d-1} w \in H \text{ and } \exists \text{ bijection } f : \{\{\mathcal{N}_{G,P}(v)\}\} \to \{\{\mathcal{N}_{H,P}(w)\}\} \text{ such that } u \sim_P^{d-1} f(u) (\forall u \in \mathcal{N}_{G,P})$$

$$\Updownarrow$$

$$v \in G_{v,d,P} \equiv^{d-1} w \in H_{w,d,P} \text{ and } \exists \text{ bijection } g : \{\{\mathcal{N}_{G_{v,d,P}}(v)\}\} \to \{\{\mathcal{N}_{H_{w,d,P}}(w)\}\} \text{ such that } \alpha(x) \sim_P^{d-1} \alpha(g(x)) (\forall x \in \mathcal{N}_{G_{v,d,P}}(v))$$

$$\Updownarrow$$

$$v \in G_{v,d,P} \equiv^{d-1} w \in H_{w,d,P} \text{ and } \exists \text{ bijection } g : \{\{\mathcal{N}_{G_{v,d,P}}(v)\}\} \to \{\{\mathcal{N}_{H_{w,d,P}}(w)\}\} \text{ such that } x \equiv^{d-1} g(x)$$

□

From the definition of $G_{v,d,P}$, it is clear that performing this iterative color refinement procedure on some vertex v in the original graph G is the same as performing the classical Weisfeiler-Leman on $G_{v,d,P}$. The neighbors of v in this graph are exactly those whose labels would be hashed when performing the P-WL procedure on G.

Proposition 10. *Given a tree T^r of depth d, a graph $G = (V, E)$, a pattern P^{r_1,r_2} and a vertex $v \in V$ and using the constructions of $G_{v,d,P}$ and T_P^r described above. The following homomorphism numbers are equal:*

$$\phi - \mathsf{hom}(T_P^r, G^v) = \mathsf{hom}(T^r, G_{v,d,P}^v)$$

Proof. Suppose that the root r of T^r has children $c_1, \dots, c_m$. Every one of these is the root of a tree $T_i^{c_i}$ of depth $\leq d-1$. We denote the tree that is obtained by attaching r to $T_i^{c_i}$ by $T_i^{r_i,c_i}$. All of these trees have maximal depth d. These trees give raise to the P-trees $T_{i,P}^{r_i,c_i}$. Now we have:

$$\begin{aligned}
\mathsf{hom}(T^r, G_{v,d,P}^v) &= \prod_{i=1}^m \mathsf{hom}(T_i^{r_i,c_i}, G_{v,d,P}^v) \\
&= \prod_{i=1}^m \sum_{v' \in \mathcal{N}_{G_{v,d,P}}(v)} \mathsf{hom}(T_i^{c_i}, G_{v,d,P}^{v'}) \\
&= \prod_{i=1}^m \sum_{v' \in \mathcal{N}_{G_{v,d,P}}(v)} \phi - \mathsf{hom}(T_{i,P}^{c_i}, G^{\alpha(v')}) \\
&= \prod_{i=1}^m \sum_{h \in \phi(P^{r_1,r_2}, G^{v,\cdot})} \phi - \mathsf{hom}(T_{i,P}^{c_i}, G^{h(r_2)}) \\
&= \prod_{i=1}^m \sum_{w \in V(G)} \#\phi(P^{r_1,r_2}, G^{v,w}) \phi - \mathsf{hom}(T_{i,P}^{c_i}, G^w) \\
&= \prod_{i=1}^m \phi - \mathsf{hom}(T_{i,P}^{c_i,r_i}, G^v) \\
&= \phi - \mathsf{hom}(T_P^r, G^v)
\end{aligned}$$

Where we used the lemma about factorizing the number of ϕ-homomorphisms of P-trees and the bijection between the neighborhoods in G and $G_{v,d,P}$ established in the proof of the previous proposition. □

Another proposition reformulates results of Grohe (2020a) and Krebs & Verbitsky (2015) for our setting.

Lemma 11. *Let $T_1^{r_1}$, $T_2^{r_2}$ be two trees of depth d. The following implication holds*

$$r_1 \equiv^d r_2 \Rightarrow T_1^{r_1} \simeq T_2^{r_2}$$

Proof. We proof this statement by induction on d and note that the base case is trivial. Suppose that $r_1 \equiv^d r_2 \Leftrightarrow T_1^{r_1}$. This is equivalent with $r_1 \equiv^{d-1} r_2 \Leftrightarrow T_1^{r_1}$ and the existence of orderings $\{x_1, \dots, x_k\}$ and $\{y_1, \dots, y_k\}$ respectively such that $x_i \equiv^{d-1} y_i$. This last part implies that $x_i \in T_{1,d-1}^{x_i} \equiv^{d-1} y_i \in T_{2,d-1}^{y_i}$. The induction hypotheses gives us that $T_{1,d-1}^{r_1} \simeq T_{1,d-1}^{r_1}$ and $T_{1,d-1}^{x_i} \simeq T_{1,d-1}^{y_i}$. Lemma 2.5 of Krebs & Verbitsky (2015) implies that we can conclude that $T_{1,d}^{r_1} \simeq T_{2,d}^{r_2}$ □

Proposition 12. *For any graphs G, H, symmetric pattern P and any tree of maximal depth d:*

$$v \in G_{v,d,P} \equiv^d w \in H_{w,d,P} \Leftrightarrow \mathsf{hom}(P^r, G_{v,d,P}^v) = \mathsf{hom}(T^r, H_{v,d,P}^w)$$

Proof. This is shown by the following implications:

$$v \in G_{v,d,P} \equiv^d w \in H_{w,d,P} \Rightarrow G_{v,d,P}^v \simeq H_{w,d,P}^w \Rightarrow$$

$$\mathsf{hom}(T^r, G_{v,d,P}^v) = \mathsf{hom}(T^r, H_{v,d,P}^w) \Rightarrow v \in G_{v,d,P} \equiv^d w \in H_{w,d,P}$$

The first implication is an application of the previous lemma, the second is obvious and the third was shown by Grohe (2020a) as theorem 4.14. □

These 3 propositions and the lemma above allow us to formulate the following theorem:

Theorem 13. *Two vertices v and w in respective graphs G, H are indistinguishable by P-WL for some pattern P^{r_1,r_2} if and only if $\phi - \mathsf{hom}(T_P^r, G^v) = \phi - \mathsf{hom}(T_P^r, H^w)$ for every T_P^r obtained by replacing the edges of a tree T_r by a graph pattern of shape P.*

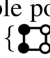
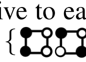
Proof.

$$v \in G \sim_P^d w \in H$$
$$\Updownarrow$$
$$v \in G_{v,d,P} \equiv^d w \in H_{w,d,P}$$
$$\Updownarrow$$
$$\mathsf{hom}(P^r, G_{v,d,P}^v) = \mathsf{hom}(T^r, H_{v,d,P}^w)$$
$$\Updownarrow$$
$$\phi - \mathsf{hom}(T_P^r, G^v) = \phi - \mathsf{hom}(T_P^r, H^w)$$

□

When applying the theory above to characterize the distinguishing power of such models we implicitly assume that a regular edge is also in the pattern set. This way, we always obtain a class of MPNNs at least as strong as the the classic message-passing framework without additional pattern information.

A.3.1 Comparing pattern-induced neighborhoods to extension by additional features

On an intuitive level, this approach bears some similarity with the edge features based on pattern counts. In both cases, the additional information contains relative positions for couples of nodes in a chosen set of patterns. However, in this framework the two roots of the pattern are not required to be on one edge but can have all possible positions relative to each other. We denote this 'completion' of a pattern set by $\bar{\mathcal{P}}$. E.g. if $\mathcal{P} = \{$$\}$ than $\bar{\mathcal{P}} = \{$$\}$. It is an open question whether this results in a stronger class of models.

Proposition 14. *If a pattern set $\mathcal{P}$ has only patterns in which the two roots are joined by an edge, than the distinguishing power of $\mathcal{P}$-MPNNs is limited by $(\emptyset, \mathcal{P}_{\mathsf{hom}})$-MPNNs.*

Proof. This can be demonstrated by induction in exactly the same way as in the proof of 3. □