Position: Future Directions in the Theory of Graph Machine Learning

Christopher Morris [1] **Fabrizio Frasca** [2] **Nadav Dym** [2] **Haggai Maron** [2 3] **İsmail İlkan Ceylan** [4] **Ron Levie** [2] **Derek Lim** [5] **Michael Bronstein** [4] **Martin Grohe** [1] **Stefanie Jegelka** [5 6]

Abstract

Machine learning on graphs, especially using graph neural networks (GNNs), has seen a surge in interest due to the wide availability of graph data across a broad spectrum of disciplines, from life to social and engineering sciences. Despite their practical success, our theoretical understanding of the properties of GNNs remains highly incomplete. Recent theoretical advancements primarily focus on elucidating the coarse-grained expressive power of GNNs, predominantly employing combinatorial techniques. However, these studies do not perfectly align with practice, particularly in understanding the generalization behavior of GNNs when trained with stochastic first-order optimization techniques. In this position paper, we argue that the graph machine learning community needs to shift its attention to developing a balanced theory of graph machine learning, focusing on a more thorough understanding of the interplay of expressive power, generalization, and optimization.

1. Introduction

Graphs serve as powerful mathematical representations, adept at capturing intricate interactions among entities across a spectrum of disciplines, spanning from life (Wong et al., 2023) to social sciences (Easley & Kleinberg, 2010) and optimization (Cappart et al., 2021). This diversity underscores the critical demand for specialized machine-learning methods that extract valuable patterns from complex graph data.

Hence, in recent years, neural networks capable of handling graph-structured data received a lot of attention in the machine learning community, especially *messsage-passing graph neural networks* (MPNNs) (Gilmer et al., 2017; Scarselli et al., 2009).[1] Nowadays, MPNNs, or, more generally, GNNs, are among the most prominent topics at top-tier machine learning conferences,[2] and have showcased promising outcomes across diverse domains, including breakthroughs in discovering new antibiotics (Stokes et al., 2020; Wong et al., 2023).

While GNNs are successful in practice and are making real-world impact, their theoretical properties are understood to a lesser extent. That is, only GNNs' expressive power, i.e., their ability to separate graphs and express functions over graphs is understood to some extent (Azizian & Lelarge, 2021; Geerts & Reutter, 2022; Morris et al., 2019; 2021; Xu et al., 2019). However, most current analyses heavily rely on combinatorial techniques, such as the 1-*dimensional Weisfeiler–Leman algorithm* (1-WL), a well-studied heuristic for the graph isomorphism problem (Grohe, 2017; Weisfeiler & Leman, 1968; Weisfeiler, 1976). While the graph isomorphism perspective has helped the community understand GNNs' ultimate limitations in capturing graph structure, it is inherently binary. For example, it does not give insights into the degree of similarity between two given graphs, prohibiting a more fine-grained analysis. While some recent works (Böker et al., 2023; Chen et al., 2022) aim at a more fine-grained analysis, they still have strong limitations, such as not considering continuous node and edge features. A second limitation of current GNN expressivity results is that they are fairly specific. They are tailored to particular classes of GNNs, overlooking practically relevant architectural choices.

While understanding MPNNs' and related architectures, e.g., graph transformers (Müller et al., 2023), expressive power is vital, understanding when such architectures generalize to unseen graphs and how to find parameter assignments that allow so is equally important. However, in MPNNs and GNNs, the essential aspects of *generalization* and *optimization* are severely understudied. The few existing works that study MPNNs' generalization properties, e.g., Garg et al.

[1]RWTH Aachen University [2]Technion - Israel Institute of Technology [3]NVIDIA Research [4]University of Oxford [5]MIT [6]TU Munich. Correspondence to: Christopher Morris <morris@cs.rwth-aachen.de>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

[1]We use the term MPNNs to refer to graph-machine learning architectures that fit into the framework of Gilmer et al. (2017) and use the term GNNs in a broader sense, i.e., all neural network architectures capable of handling graph-structured inputs.

[2]http://tinyurl.com/mpn89vju

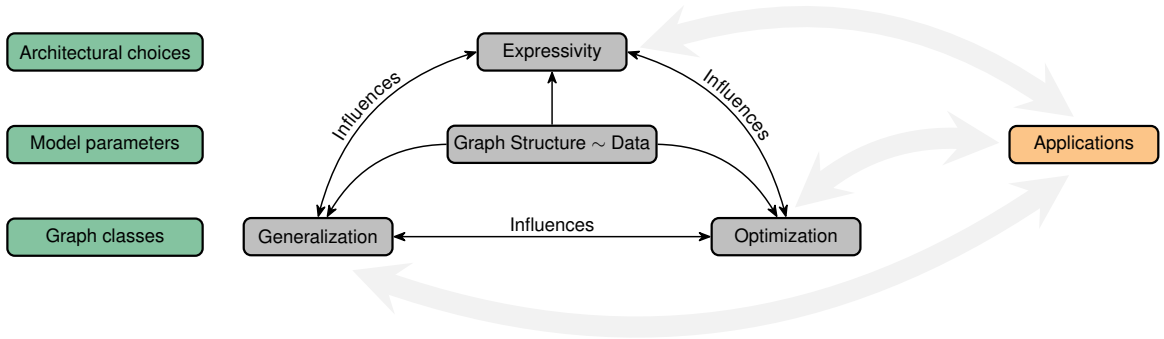

Figure 1. Interactions of the four challenges within graph machine learning: Fine-grained *expressivity*, *generalization*, *optimization*, *applications*, and their interactions. The green boxes *architectural choices* (hyperparameter and other design choices like normalization layers), *model parameters*, and *graph classes* (different types of graphs) represent aspects of all four challenges.

(2020); Liao et al. (2021); Maskey et al. (2022); Scarselli et al. (2018), study MPNNs' generalization properties via VC (Vapnik-Chervonenkis) dimension theory or related formalisms and derive generalization bounds that depend on high-level graph parameters such as the maximum degree or number of nodes. In addition, all current works are based on variants of classical uniform generalization bounds. This entails large constants in the bound and a description of a classical bias-variance curve that does not describe the typical reality of deep learning on graphs, in which higher complexity and more expressive models often generalize better.

While some initial works make progress towards studying GNNs' optimization aspects, i.e., the dynamics of *stochastic gradient descent* (SGD) to adapt their parameters, they still make strong assumptions, such as the use of linear activation functions (Xu et al., 2021a), unrealistic learning scenarios (Du et al., 2019), or neglecting the influence of the graph structure (Tang & Liu, 2023); see also (Bechler-Speicher et al., 2023).

Overall, we argue that the following challenges persist in graph machine learning.

Expressivity The vast majority of existing results on GNNs' expressive power are coarse-grained and focus on specific architectures. Additionally, guidelines for choosing between highly-expressive GNNs are needed.

Generalization Besides expressivity, it is vital to choose models based on a training dataset so that it generalizes to unseen data. Current works studying GNNs' generalization properties often rely on many simplifying assumptions, e.g., not considering graph structure or optimization.

Optimization We must ensure that GNNs trained with SGD converge to assignments leading to expressive models that generalize well and understand the role of the architecture and graph structure.

Applications Current theoretical results are often loosely aligned with practical assumptions and needs of application domains. Hence, developing graph machine learning theory aligned with practical requirements is crucial.

We argue that graph machine learning needs a nuanced theory to develop an in-depth understanding of the various architectural choices that govern GNNs' expressive power (challenge 1), generalization properties (challenge 2), and optimization dynamics (challenge 3), as well as the *interplay* of these aspects. Moreover, theoretical research within the graph machine learning communities must be aligned with domain experts' practical needs (challenge 4). Thereto, we propose concrete challenges to address these requirements; see Figure 1 for a high-level overview of the interactions of the four challenges in this position paper.

2. Expressive Power of GNNs

Most of the expressiveness studies examine the ability of GNNs to assign distinct values to non-isomorphic graphs, or in other words, the ability of GNNs to separate different graphs. Moreover, the ability to separate graphs is directly connected to MPNNs' ability to approximate (continuous, permutation-invariant) functions over graphs (Chen et al., 2019). The seminal works of Morris et al. (2019); Xu et al. (2019) showed that MPNNs' ability to separate graphs is equivalent to the *limited* separation power of the 1-WL algorithm for the graph isomorphism problem (Weisfeiler & Leman, 1968). These works inspired the study of *expressive GNNs*, whose expressive power surpasses that of the 1-WL test, typically at the cost of more computational resources. For example, Morris et al. (2019); Maron et al. (2019) derived architectures equivalent to the more powerful k-dimensional Weisfeiler–Leman algorithm (k-WL). Many other types of expressive GNNs were proposed in the literature, e.g., by utilizing random features (Abboud et al., 2021), subgraph counts (Bouritsas et al., 2020), or by employing sets of sub-

graphs (Bevilacqua et al., 2022; Cotta et al., 2021; Qian et al., 2022; Frasca et al., 2022). See the following surveys for a more thorough review: (Jegelka, 2022; Li & Leskovec, 2022; Morris et al., 2021; Sato, 2020; Zhang et al., 2023a). Several recent works developed tools to analyze the expressive power of GNN architectures or construct hierarchies thereof (Geerts & Reutter, 2022; Zhang et al., 2023b). Alternative expressive power measures have been proposed in the past several years, for example, measuring the degree of invariant or equivariant polynomials an architecture can represent (Puny et al., 2023), or measures based on the ability to compute the count of substructures (Chen et al., 2020) or other graph properties (Zhang et al., 2023c) and to mix information from different nodes (Di Giovanni et al., 2023b).

2.1. Challenges

Based on the above, we identify the following challenges.

Challenge II.1: From combinatorial to geometric expressiveness results. The common results on the expressive power of GNNs are *coarse-grained.* Since these tools are rooted in graph isomorphism testing formalism, and the graph isomorphism problem inherently involves binary outcomes, they fall short of offering insights into the *degree of similarity* between graphs. This omission potentially limits the scope of a nuanced analysis of GNNs. For example, Maron et al. (2019); Morris et al. (2019); Xu et al. (2019) have independently devised 1-WL-expressive GNNs, distinguishing the same set of graphs. However, their empirical performance in practical situations is not identical, and which one is better varies across tasks.

To guide practitioners in selecting these maximally expressive MPNNs, we need to consider the *geometry of the space of graphs* induced by the geometry of the feature space produced by the MPNNs. Note that an MPNN is a function that maps graphs to features in some Euclidean space, and the Euclidean metric in the feature space can be pulled-back to a (pseudo) metric of graphs. The choice of architecture determines this graph metric and which graphs can be separated by the architecture and how easily. Understanding this *fine-grained expressivity* could lead to a more systematic approach to designing GNN architectures. In addition, using such a "continuous" notion of graph similarity will also lead to results akin to *geometric stability* in geometric deep learning (Bruna & Mallat, 2013), potentially related to generalization; see Section 3.

While Böker et al. (2023) took a first step towards a non-binary class expressivity and identified several well-known metrics in the graph space, which are topologically equivalent to the Euclidean metric in the feature space; the results only consider graphs without continuous node or edge features, use specific aggregation functions, and do not offer an explicit way to bound graph distances by feature distances and vice-versa. As an initial step, (Chuang & Jegelka, 2022) derived a pseudometric on graphs with continuous attributes based on the computation trees of MPNNs. Additional related works are on transferability and convergence of GNNs, which only give one side of a topological equivalency, namely, convergence of graphs lead to convergence of features (Levie et al., 2021; Keriven et al., 2020; Ruiz et al., 2020).

We propose developing *fine-grained expressivity* results, namely metric equivalencies between explicit graph metrics and feature metrics for GNNs on graphs with features. An ideal result would derive a bi-Lipschitz correspondence between such metrics. Note that Levie (2023) gave one side of the Lipschitz inequality for graphs with features, namely, the Euclidean feature distance is bounded by the graph cut-distance for any MPNN. Lastly, fine-grained expressivity can lead to *optimal universal approximation theorems*. Universal approximation results based on the Stone–Weierstrass theorem require a better understanding of the topology of the space of graphs. The finer the topology we consider on the input space, the more points are "far apart" in this space, and therefore, the harder it is to separate far-apart points using functions. Hence, optimal universal approximation theorems should find the finest topology in which MPNNs separate points.

Challenge II.2: Towards understanding expressiveness for all practical architectures. A second limitation of current GNN expressivity results is that they are very *specific*. They are tailored to specific classes of GNNs, overlooking practical and relevant architectural choices. More specifically, the equivalence of MPNNs and the 1-WL is obtained for particular choices of MPNN architectures (Morris et al., 2019; Xu et al., 2019). Therefore, there is a need to understand the effect of various architectural decisions made in practice, such as different activations, aggregation, normalization, and pooling, on the expressive power of MPNN architectures. Examples of results in this vein are the expressive advantage of sum pooling over mean and max pooling (Xu et al., 2019) and the expressive advantage of analytic activations over piecewise linear activations (Amir et al., 2023).

Apart from MPNNs, a comprehensive understanding of expressivity is especially lacking for graph transformers (GTs). The effectiveness of GTs heavily relies on incorporating structural and positional encodings (Müller et al., 2023). These encodings introduce information about the underlying graph structure into the transformer architecture, which is inherently designed without an awareness of graph structures. However, it is still largely unclear how these encodings influence an architecture's ability to capture graph structure; see some preliminary results in Lim et al. (2023b;a); Zhu et al. (2023); Zhang et al. (2023c), and also Cai et al.

(2023); Rosenbluth et al. (2024) for some connections between MPNNs and GTs. However, it remains unclear how to abstract the various architectural choices and engineering tricks via a mathematical model, allowing for a detailed mathematical analysis of GT's potential benefits over MPNNs.

Challenge II.3: Towards uniform expressiveness results. Most GNN expressive power results in the literature either do not quantify or only give loose bounds on the size of the GNN required (in terms of number of parameters, width, or depth) to compute different functions on graphs, e.g., Aamand et al. (2022); Amir et al. (2023); Morris et al. (2019). Moreover, most expressivity results are *non-uniform*, i.e., they depend on the graph size. Further study on this could have several benefits. First, the size of a GNN required for different tasks is related to its generalization ability. Secondly, the size of GNNs required for expressing certain functions could help compare different GNN architectures that have similar or the same expressive power; for instance, for two MPNN architectures that are 1-WL equivalent, one may prefer the one that requires fewer parameters for distinguishing 1-WL-distinguishable graphs. Moreover, deriving uniform expressivity results could shed light on GNNs' ability to generalize to larger graphs, as seen during training. Barceló et al. (2020) achieves a result that is uniform, i.e., one model parametrization can express the target function on *all* graphs; see also Grohe (2021) for discussions. However, the result hinges on the use of logical classifiers. In addition, Grohe (2023) proves that all functions computable by MPNNs are expressible in the logic $FO^2 + C$, leading to an exact characterization of the functions that are computable by MPNNs of polynomial size and bounded depth in terms of logic and also standard computational complexity classes.

Challenge II.4: Towards expressiveness on relevant classes of graphs. Most expressive GNNs have been designed to attain high separation power on the general family of all possible graphs. However, practical applications typically target specific classes of graphs for which tailored expressiveness results may be obtained or are otherwise known. For example, most molecules are in the family of *planar* graphs (Simmons & Maggio, 1981), on which the graph isomorphism problem can be solved more efficiently by specialized algorithms (Hopcroft & Wong, 1974). Because of this, we advocate addressing the challenge of developing the study of expressive GNN architectures targeting relevant graph classes of interest, following pioneering works by Dimitrov et al. (2023) on planar graphs, and by Bause et al. (2023) on outer-planar graphs. Besides atomistic systems, other examples include bipartite or tripartite graphs stemming from optimization problems (Cappart et al., 2021; Qian et al., 2024a) or recommender systems (Wu et al., 2023).

We suggest identifying a taxonomy of families of graphs that hold particular importance in practical applications to re-evaluate the expressive power of known architectures about these and to derive optimal lower-bound intricacy figures for maximally expressive approaches on these classes; see Challenge II.5. These analyses would support the design of architectures, which, on specific graph families, can attain better complexity-expressivity tradeoffs than more general GNNs.

Challenge II.5: Towards a formal trade-off between expressive power and computational cost. Another challenge concerning expressivity is exploring the realm of *expressive GNNs*. While there are many works proposing GNNs whose expressivity surpasses the 1-WL test, the GNN community needs principled methods of navigating this vast collection of architectures by providing guidelines for how expressive a GNN architecture needs to be to address a given task (Di Giovanni et al., 2023b). In addition, in practice, we observe a trade-off between computational complexity and expressive power, where more expressive architectures typically come with a higher computational complexity. There is a need to quantify this trade-off formally. For example, there is a lack of theoretical knowledge to address questions such as: given specific expressive power requirements (e.g., 1-WL, or k-WL, counting specific substructures), what are the lower bounds of the time and space complexity for a GNN model with the requisite representational capacity? Tahmasebi et al. (2023) recently obtained preliminary results in this direction. The answer to this question could guarantee that the GNN architectures we currently have are optimal in terms of complexity or, conversely, point towards possible improvements that can lead to GNNs with identical expressivity properties and improved complexity.

Challenge II.6: Towards linking architecture, task, and graph structure. In GNNs, the graph is part of the input and the computational device. At times, the input graph might not be ideally suited for message passing, leading to phenomena such as over-squashing (Alon & Yahav, 2021). Characterizing the graph properties that lead to such phenomena, e.g., through curvature (Topping et al., 2022) or diffusion distances (Di Giovanni et al., 2023a), can lead to more principled approaches for decoupling the computational and input graphs in MPNNs through *rewiring* (Barbero et al., 2024; Qian et al., 2024b), or more efficient sparse GTs. A better characterization of the tasks that a specific MPNN architecture can implement on a particular graph, such as in the recent work of (Di Giovanni et al., 2023b), can also lead to a novel type of expressiveness results.

3. Generalization Properties of GNNs

The few existing works that study MPNNs' generalization properties, e.g., Garg et al. (2020); Liao et al. (2021); Maskey

et al. (2022); Scarselli et al. (2018), study their generalization properties via VC dimension theory or related formalisms. However, the bounds derived in such works usually depend only on relatively generic graph parameters, such as the maximum degree or number of nodes, ignoring more expressive graph parameters. While Morris et al. (2023) recently established a tight connection between the expressivity of the 1-WL and MPNNs' VC dimension, the analysis is restricted to a particular MPNN architecture using sum aggregation; see also Section 2. Hence, it does not allow for fine-grained analysis, e.g., considering different architectural choices and more quantitative analysis in terms of numbers of parameters or depth. An additional important factor is that the above works assume that the train and test sets are sampled from the same distribution. Hence, they reveal little about MPNNs' ability to generalize out-of-distribution, e.g., generalizing to larger graphs than found in the training set. Furthermore, no theoretical understanding exists of the generalization abilities of more expressive GNN architectures and how they are influenced by graph structure.

3.1. Challenges

To advance the study of GNNs' generalization ability, we suggest to address the following challenges.

Challenge III.1: Understanding the influence of expressiveness and architectural choices on generalization. Although classical learning theory (Morris et al., 2023) would suggest worse generalization results for higher expressiveness, more expressive GNN architectures often achieve better generalization performance. While the underlying reasons for these improvements are not fully clear, this phenomenon hints at the important role played by particular inductive biases and their interplay with the data distribution at hand (Bouritsas et al., 2020). We thus propose to establish precise conditions under which increased expressive power, jointly with architectural choices, leads to enhanced generalization. Franks et al. (2024) made recent progress in that direction, focusing on linearly separable data.

We propose to analyze how expressivity transforms the feature space to facilitate better generalization, for example, by deriving conditions on the underlying data distribution such that more expressive power leads to better predictive performance. At the same time, it is pivotal to unravel the influence of different design choices on GNNs' generalization abilities, including, e.g., aggregation functions, skip connections, or normalization layers. For example, an essential aspect would be understanding if certain aggregation functions exhibit superior generalization and offer advantages in making predictions on larger graphs beyond the training set and for which tasks. A first step in this direction was taken by Xu et al. (2020; 2021b). Related to this, we deem it essential to investigate architectural paradigms beyond message-passing, such as graph transformers (Müller et al., 2023). In this sense, a relevant endeavor would be to decipher how the attention mechanism contributes to improved generalization compared to MPNNs, the influence of various structural and positional encodings on this aspect, and how they influence generalization bounds. Building on advancements in understanding the expressive power of structural encodings, see Section 2, we propose to understand if certain encodings offer better generalization properties and how different encodings influence the sample complexity.

Challenge III.2: Understanding the impact of graph structure on generalization and its interplay geometry. This challenge asks how graph structure influences an architecture's generalization properties and how this is linked to an architecture's expressive power and the feature space's geometry. For example, it is essential to understand if graph properties such as sparsity influence generalization and how to integrate them into generalization bounds. Moreover, considering practical, relevant graph classes such as planar or bipartite graphs, it is important to precisely understand whether GNNs operating on certain graph classes offer better generalization than others. Further, leveraging results on the fine-grained geometry of the feature space outlined in Section 2 possibly allows to derive tighter generalization bounds. While Levie (2023) took a first step in this direction, they did not consider the coarsest topology in the space of graphs equivalent to the geometry of the feature space, and hence their results are suboptimal.

Challenge III.3: Develop a theory of data augmentation. Given the cost and scarcity of labeled training data in many practical graph machine learning scenarios, several recent works (e.g., as surveyed in Ding et al. (2022)) suggested leveraging *data augmentation* techniques (i.e., enhancing the training dataset with additional labeled data samples) to increase the training data size effectively. Importantly, there is little theoretical understanding, existing work, e.g., uses graphons (Han et al., 2022), and relatively little practical evidence regarding which data augmentation schemes are helpful for specific graph distributions and learning tasks. Recently, graph data augmentation strategies have seen application in mitigating covariate distribution shifts (Sui et al., 2023) and in rationalizing GNN predictions (Liu et al., 2024; Wu et al., 2022). Unfortunately, however, compared to the widespread beneficial effect data augmentation has on learning with other popular data modalities—such as images—the effectiveness of data augmentation in graph machine learning still lags behind. Hence, a critical challenge is the development of a general theory of data augmentation for graph-structured data. Hence, we suggest deriving a better understanding of how much we can perturb graph data without distorting the underlying data-generating distribution, e.g., leveraging recent advancements in principled graph

similarity (Gervens & Grohe, 2022).

Challenge III.4: Understanding and improving extrapolation, especially to larger graphs. Extrapolation to graphs sampled from a different distribution than the training distribution is crucial in many applications. For instance, extrapolation to larger graphs is especially important in tasks where labels are much more expensive for larger graphs, such as combinatorial optimization applications, where solvers are slow on large instances (Cappart et al., 2021). It has been shown both theoretically and empirically that GNNs can fail on graphs that are larger than, or in general have a different structure than, the graphs that they are trained on (Yehudai et al., 2021; Veličković et al., 2022; Xu et al., 2021b; Zhou et al., 2022). Hence, we need to better understand under which conditions on data distributions, models, and optimization out-of-distribution generalization is possible. From the perspective of model and optimization, Xu et al. (2021b) take a first step in showing how restrictions on the architecture can enable extrapolation to different graph structures. In addition, initial results (Adam-Day et al., 2023) within the Erdős–Rényi model, leveraging results from finite model theory, show that MPNNs eventually become independent of their inputs as the graph grows. Another perspective is to characterize theoretical conditions on the structure of the graphs in training and test sets. In transferability and convergence analysis, MPNNs are shown to be transferable between different graphs that are sampled from the same random graph model (Keriven et al., 2020; Le & Jegelka, 2023; Levie et al., 2021; Ruiz et al., 2020), but we still need to better understand the scope of practical situations in which these conditions hold. More generally, we still need to analyze the entire landscape of conditions for extrapolation in practical situations, taking into account data, task, architecture, and other inductive biases, e.g. from optimization, for a more general and actionable picture of when GNNs and graph transformers extrapolate to different kinds of data distributions, and develop new methods for improving GNN extrapolation ability. A related question is to estimate the reliability of a model on new distributions in a way consistent with empirical behavior. This question relates to understanding the geometry of graph representations (see also Challenges II.1 and III.2) and relevant divergences of distributions that reflect GNN stability. A first work in this direction is Chuang & Jegelka (2022). Answering these questions would also be a basis to build a better understanding of transfer learning for graphs.

4. Optimization Dynamics of GNNs

Only some works directly focus on the theoretical understanding of gradient descent-based learning for GNNs. These works typically make substantial simplifying assumptions. For example, Xu et al. (2021a) showed global convergence of GNNs with *linear activations*. In contrast, Du et al. (2019) showed global convergence for the neural tangent kernel obtained as the infinite width limit of GNNs, simplifying practical scenarios. In addition, the analysis in Tang & Liu (2023) does not consider graph structure.

In contrast, experimental and theoretical evidence suggests that the GNN optimization procedure may only be optimal in some settings. In several papers (Böker et al., 2023; Huang et al., 2022), GNNs with random parameters or graph kernels with handcrafted features outperform trained GNNs. Furthermore, GNNs' performance often deteriorates as depth increases, a problem frequently attributed to phenomena such as over-smoothing, over-squashing, and graph bottlenecks. While many normalization techniques have been suggested to avoid these issues successfully, they often are not accompanied by a substantial performance gain (Rusch et al., 2023). Additional evidence of the limitation of the GNN optimization procedure is the recent work of Bechler-Speicher et al. (2023), which shows that GNNs perform suboptimally on learning tasks that can be solved only using node features, despite the theoretical ability of GNNs to exploit these features only and disregard the graph structure.

4.1. Challenges

Based on the above, we identify the following challenges.

Challenge IV.1: Towards guarantees for convergence quality and rate. One important step towards better understanding the learning of graph representation models is to improve existing analysis of convergence by considerably weakening the strong linearity assumptions used in existing work (Du et al., 2019; Tang & Liu, 2023; Xu et al., 2021a). The convergence guarantees we are concerned with are both guarantees on the rate of convergence and guarantees on the quality of convergence, as well as a characterization of what the dynamics converge to, e.g., a global or low-loss local minimum or basin of attraction and other properties, including relations to generalization, see Section 3, e.g., via certain biases for simplicity. We may also ask whether GNN training exhibits phenomena that have been observed in other neural networks, such as the edge of stability or convergence not to a point but an invariant measure (Cohen et al., 2021; Chandramoorthy et al., 2022; Lobacheva et al., 2021; Zhang et al., 2022).

Challenge IV.2: Towards understanding the influence of architectural choice on GNNs' optimization. Here, the primary objective is to unravel the impact of various architectural choices, such as aggregation functions or normalization layers, on the properties of GNNs trained with SGD. The goal is to discern how these architectural choices shape the convergence properties of GNNs, explicitly investigating whether SGD can lead to parameter assignments

that possess favorable attributes for generalization. This can be accomplished by adapting recent advancements in understanding the convergence properties of SGD on standard feed-forward neural networks , e.g., Du & Lee (2018); Arora et al. (2019); Xu & Du (2023) and applying these insights to MPNNs. This adaptation may involve interpreting MPNN computation through the recursive unrolling of neighborhood structures and comprehending how graph structure can be incorporated into these results. This exploration should include GTs; insights from MPNNs can potentially be transferred, with modifications, to GTs. We need to investigate whether GTs exhibit distinct properties when trained with SGD. For instance, we must explore the potential significant influence of attention mechanisms within GTs.

Challenge IV.3: Towards understanding the influence of the graph structure on GNNs' optimization. In other realms of deep learning, neural networks typically adhere to a consistent structure, resulting in the limited impact of their structure on the convergence properties of SGD. However, for GNNs, the neural networks' architecture is directly shaped by the underlying data distribution, i.e., dictated by the graphs' connectivity. Consequently, this challenge focuses on unraveling the intricacies of how graph structure, encompassing factors such as sparsity or specific graph classes, affects the convergence properties of SGD. In addition, we seek to understand how parameters controlling graph structures, e.g., the number of symmetries, can be incorporated into convergence guarantees. Given that GTs integrate graph structure through structural and positional encodings, it is also interesting to understand the interplay between these encodings and the convergence behavior of SGD, as well as the impact of choosing different encodings on the convergence dynamics.

Challenge IV.4: Towards harnessing the power of depth. For convolutional neural networks, successful training of deeper neural networks via normalization techniques, such as batch normalization or skip connections, have led to significant performance improvements (He et al., 2016). Several works, e.g., Zhao & Akoglu (2020), have developed analogous normalizations for GNNs, and in some cases, complex combinations of these techniques do lead to improved performance (Li et al., 2021). Nevertheless, these solutions have not consistently demonstrated gains significant enough to persuade the community of their fundamental architectural role, as is arguably the case for their counterparts in other modalities. We suggest approaching this challenge by developing toy models of graph machine-learning problems that can only be solved by deep GNNs and identifying training mechanisms that will be guaranteed to lead to successful learning. As a second step, the obtained training techniques can be applied to real-world tasks, emphasizing tasks where deeper GNNs are expected to be beneficial, such as the "long-range graph benchmark" (Dwivedi et al., 2022).

Challenge IV.5: Defeating randomness. The existence of instances where graph features produced by GNNs with random weights perform on par with trained GNNs (Böker et al., 2023; Huang et al., 2022) suggests that optimization of GNNs sometimes does not lead to significant improvement over the initial solution. This finding is coherent with that MPNNs of moderate size, with random weights, attain their maximal separation power (Amir et al., 2023; Aamand et al., 2022). Nonetheless, it seems plausible that optimizing the GNNs' weights should lead to better graph features and more successful learning by SGD. This challenge aims to find mathematically precise explanations and models where the optimal GNN parameters are significantly better for learning than the average GNN parameters and suggest ways to successfully learn these optimal GNN parameters; see also Challenge IV.1.

5. Connecting Theory with Practice

While addressing the above-outlined challenges regarding the GNNs' expressive power, generalization abilities, and optimization dynamics is essential to push GNN theory forward, we also want to stress the need to align such theory with practical needs.

For example, currently, proposed expressive architectures, such as higher-order GNNs aligned with the k-WL (Morris et al., 2021), are rarely employed by domain experts, e.g., in molecular property prediction (Duval et al., 2023). In addition, theoretical results that have been derived so far often make unrealistic assumptions, ignoring practical needs such as continuous node and edge features. Moreover, results investigating the generalization properties of GNNs based on VC dimension theory and related concepts, e.g., Garg et al. (2020); Morris et al. (2023), result in large sample complexities, providing no practical guidelines.

Hence, it is essential to adapt new theoretical results to domains where GNNs are frequently used, e.g., the molecular domain (Duval et al., 2023) or combinatorial optimization (Cappart et al., 2021). Moreover, in practice, using certain engineering tricks or folklore knowledge is common, e.g., using a specific normalization layer in an application domain. Therefore, it is essential to incorporate these choices into theory, understanding why they work in practice and how to improve them potentially.

Moreover, to quickly disseminate state-of-the-art, theoretically principled GNN architectures to real-world applications, providing efficient, easy-to-use implementations of such architectures is essential. Wang & Zhang (2023a) took a first step in this direction by providing open-source implementations of recently proposed expressive GNN archi-

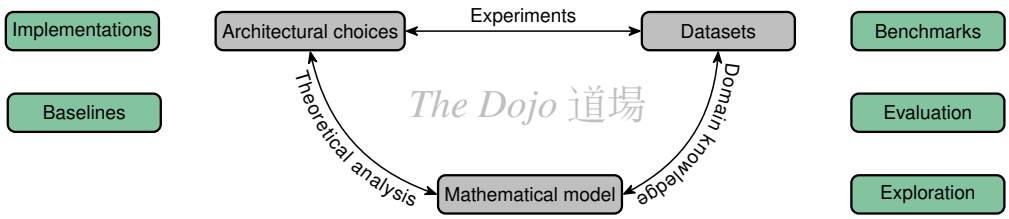

Figure 2. Proposal for a better alignment of theoretical and practical research within the graph machine learning community. We propose the tight interaction and iterative refinement of mathematical models and architectural choices via rigorous experimental evaluations, supported by state-of-the-art baseline implementations, benchmarks, evaluation pipelines, and visual exploration tools.

tectures. Hence, it is crucial to push such initiative further by establishing a library used by the community, possibly extending PyTorch Geometric (Fey & Lenssen, 2019) or DGL (Wang et al., 2019), making such architectures readily available and easily benchmarked for practitioners.

Similarly, theoretical papers often need more thorough experimental evaluations. They are mostly evaluated on small, out-of-date benchmark datasets, and often, their hyperparameters are not tuned sufficiently. Therefore, it is often unclear if they perform better than tuned state-of-the-art GNNs. Hence, it is essential to establish proper experimental pipelines and evaluation protocols for newly proposed theoretically-principled architectures. Alongside establishing solid evaluation protocols, it is important to establish (synthetic) benchmark datasets to investigate the effect of expressivity and its connection to generalization and optimization in detail. A first step in this direction was recently made by introducing the Brec dataset (Wang & Zhang, 2023b), which still lacks diversity in graph structure.

In addition, with the emergence of large-language models (LLMs), several recent works tried using them for tasks such as node/graph classification or graph generation (Chen et al., 2024; Fatemi et al., 2023). However, their application remains mostly ad-hoc, and it is unclear when they outperform GNNs or help GNNs make better predictions.

5.1. Challenges

We derive the following challenges from the above to better align theory with practical needs.

Challenge V.1: Unifying practical studies of theoretically principled GNN architectures. We have argued that expressiveness, generalization, and optimization are interrelated aspects necessitating a more holistic treatment. We believe this should also be the case when studying these aspects from a practical, experimental perspective and suggest establishing a "Theo-practical Dojo" inspired by the work of (Joshi et al., 2023) on geometric graphs. The primary objective is to guarantee a unified and standardized experimental comparison of GNN architectures designed to guide theoretically grounded considerations and foster controlled, comparative studies. Accordingly, the dojo would consist of pivotal benchmarks extending beyond graph discrimination and encompassing tasks, including predicting relevant graph properties. Other than exposing standard protocols for training and evaluation in the spirit of (Hu et al., 2020), to jointly elicit aspects related to expressiveness, generalization, and optimization, proposed tasks would cover families of graphs with diverse structural characteristics, various types of training-test distribution shifts and training datasets of different scales. The dojo should allow controlling for and contrasting the complexity of methods in comparison, regarding the number of learnable parameters and their empirical running time.

Finally, the dojo could be extended with a tool to visually and interactively explore the hidden representation space of models in comparison. We envision the tool would enhance comprehension of the relationship between separation power and generalization. Overall, from the analyses supported by the dojo, researchers would obtain insights on their approaches, such as architectural "blind spots," performance gaps, and scaling difficulties, in a way to informatively guide follow-up research refining their mathematical models, see Figure 2.

Challenge V.2: A library of state-of-the-art, theoretically-guided GNN implementations. To support the development of the above-outlined dojo, it is crucial to have an ideally large set of well-maintained and documented implementations of state-of-the-art theoretically-guided GNN architectures available, possibly building on existing GNN implementation libraries such as PyTorch Geometric (Fey & Lenssen, 2019) or DGL (Wang et al., 2019). Moreover, the dojo's evaluation pipeline and available datasets should be integrated into such a library. In addition, we propose to provide one-click reproducible baseline results to compare newly proposed architectures to existing ones easily and ensure fair comparisons.

Challenge V.3: Adapting theoretically-guided GNN architectures to domain knowledge. The challenges outlined in Sections 2 to 4 mainly deal with the derivation of a general theory of graph machine learning. However, to quickly

disseminate the theory to practice and leverage it by domain experts, it needs to be adapted to the specific needs of specific domains. Hence, we propose to identify key application domains for graph machine learning, e.g., molecular property prediction or combinatorial optimization, and, together with domain experts, work out their specific requirements. For example, it is essential to understand which kind of graph structures arise in particular domains or what engineering tricks, such as normalization layers, are currently leveraged. Building on such a requirements list, we propose to adapt and refine the mathematical theory and theoretical results. In addition, we also suggest investigating the potential of Neural Architecture Search (White et al., 2023; Elsken et al., 2019) for GNNs to automatically adapt existing architectures to meet specific requirements. It would also be interesting to comparatively study architectural solutions found by NAS and those emerging by the theoretical studies proposed in this paper.

Challenge V.4: Improving graph machine learning in different learning paradigms. Most of the graph machine learning works we have cited here study supervised graph machine learning, theoretically and empirically. Nonetheless, many other learning paradigms impact graph machine learning and other areas, e.g., self-supervised learning, generative modeling, transfer learning, foundation models, few-shot learning, meta-learning, reinforcement learning, and planning. Advances in graph machine learning within these different learning paradigms have found substantial empirical impact in several cases. Graph learning research has, for instance, fostered interesting advances in the area of AI agents and planning (Ståhlberg et al., 2021; Deac et al., 2020) or has, otherwise, achieved noteworthy results when in combination with the aforementioned learning approaches, see, e.g., chip design using GNNs and reinforcement learning (Mirhoseini et al., 2021), self-supervised learning for 3D molecular property prediction (Godwin et al., 2022), and transfer learning from supervised pre-trained molecular models (Ying et al., 2021). However, the success of some of these different learning paradigms in graph machine learning has lagged significantly behind their success in other areas, such as natural language processing, computer vision, and audio processing. For instance, self-supervised learning on pure graph data (without, e.g., the task-specific molecular 3D information as in Godwin et al. (2022)) has had much less empirical impact than in NLP and CV. Also, generative models for graphs are much weaker than those of other domains; sequence models that generate molecules via string representations are easier to use and often outperform graph generative models (Flam-Shepherd et al., 2022), so sequence models are frequently used in practice for drug-design applications. Outside of graph machine learning, each learning paradigm has rich theory and specific methods that enable their success. Further theoretical and empirical studies into connecting graph machine learning with these different learning paradigms could be very impactful.

Challenge V.5: The principled application of LLMs for GNNs. Nowadays, there is a large set of work on mixing text and molecule/protein/crystals modalities, e.g., performing protein retrieval based on their text descriptions (Xu et al., 2023), editing molecular structures with text instructions (Liu et al., 2022), or generating geometric properties of crystal structures solely from fine-tuned LLMs (Gruver et al., 2024). Pioneering works also propose using LLMs for general interactive reasoning and mining on graphs (Fatemi et al., 2023; Zhao et al., 2023a). Hence, a future challenge regarding the foundations of graph learning would be to understand these model capabilities precisely, with a specific focus on their limitations and failure cases. Another pressing challenge is applying LLM/GPT-style training on graphs in a principled manner, e.g., for autoregressive graph generation. Here, we need to traverse the graph canonically to represent the graph as a sequence of tokens. However, theoretically, this is challenging as it entails determining the orbit of each node (the structural role of each node in the graph). Hence, a challenge here is to devise "approximate" node traversal strategies (Zhao et al., 2023b) that work well with LLMs-like training objectives or devise exact ones for practical, relevant graph classes, e.g., molecular graphs.

6. Conclusion

Here, we stressed the importance of a broader theory of graph machine learning. Concretely, we highlighted the importance of developing a more fine-grained theory of expressivity, relying less on the simple perspective of graph isomorphism testing and considering more practically relevant architectural parameters such as normalization layers and skip connections. In addition, we underlined the need for more work investigating the generalization and optimization aspects of graph machine learning, focusing on more realistic assumptions. Of course, the aspects of expressivity, optimization and generalization are closely linked, and this *interplay* warrants further understanding. Finally, we stressed the importance of aligning this more balanced theory of graph machine learning with practical needs, e.g., by considering expert knowledge. By investigating the challenges we have introduced, we believe that our research community will be able to rethink the pillars of expressiveness, generalization, and optimization in a more holistic and cohesive manner that is more directly informed by practical and domain considerations. We hope that this paper presents a valuable handbook of directions for developing a more realistic and balanced theory of graph machine learning and that its insights will help spur novel research results and avenues in the future.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgements

Christopher Morris is partially funded by a DFG Emmy Noether grant (468502433) and RWTH Junior Principal Investigator Fellowship under Germany's Excellence Strategy. Nadav Dym is supported by Israeli Science Foundation grant No. 272/23. Haggai Maron is the Robert J. Shillman Fellow and is supported by the Israel Science Foundation through a personal grant (ISF 264/23) and an equipment grant (ISF 532/23). Fabrizio Frasca is funded by the Andrew and Erna Finci Viterbi Post-Doctoral Fellowship. Ron Levie is supported by the Israel Science Foundation (grant No. 1937/23). Derek Lim is funded by an NSF Graduate Fellowship. Michael Bronstein is supported by the EPSRC Turing AI World-Leading Research Fellowship No. EP/X040062/1 and the EPSRC AI Hub for Mathematical Foundations of Intelligence No. EP/Y028872/1. Martin Grohe acknowledges funding from the European Union (ERC AdG SymSim, 101054974). Stefanie Jegelka acknowledges funding from NSF award CCF-2112665 (TILOS AI Institute) and the Alexander von Humboldt Foundation.

References

Aamand, A., Chen, J., Indyk, P., Narayanan, S., Rubinfeld, R., Schiefer, N., Silwal, S., and Wagner, T. Exponentially improving the complexity of simulating the Weisfeiler-Lehman test with graph neural networks. *NeurIPS*, 2022.

Abboud, R., Ceylan, İ. İ., Grohe, M., and Lukasiewicz, T. The surprising power of graph neural networks with random node initialization. In *IJCAI*, 2021.

Adam-Day, S., Iliant, T. M., and İsmail İlkan Ceylan. Zero-one laws of graph neural networks. In *NeurIPS*, 2023.

Alon, U. and Yahav, E. On the bottleneck of graph neural networks and its practical implications. In *ICLR*, 2021.

Amir, T., Gortler, S. J., Avni, I., Ravina, R., and Dym, N. Neural injective functions for multisets, measures and graphs via a finite witness theorem. In *NeurIPS*, 2023.

Arora, S., Du, S. S., Hu, W., Li, Z., and Wang, R. Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. In *ICLR*, 2019.

Azizian, W. and Lelarge, M. Characterizing the expressive power of invariant and equivariant graph neural networks. In *ICLR*, 2021.

Barbero, F., Velingker, A., Saberi, A., Bronstein, M., and Di Giovanni, F. Locality-aware graph-rewiring in gnns. *ICLR*, 2024.

Barceló, P., Kostylev, E. V., Monet, M., Pérez, J., Reutter, J. L., and Silva, J. P. The logical expressiveness of graph neural networks. In *ICLR*, 2020.

Bause, F., Jogl, F., Indri, P., Drucks, T., Penz, D., Kriege, N. M., Gärtner, T., Welke, P., and Thiessen, M. Maximally expressive GNNs for outerplanar graphs. In *NeurIPS, Workshop on New Frontiers in Graph Learning*, 2023.

Bechler-Speicher, M., Amos, I., Gilad-Bachrach, R., and Globerson, A. Graph neural networks use graphs when they shouldn't. *arXiv preprint*, 2023.

Bevilacqua, B., Frasca, F., Lim, D., Srinivasan, B., Cai, C., Balamurugan, G., Bronstein, M. M., and Maron, H. Equivariant subgraph aggregation networks. In *ICLR*, 2022.

Bouritsas, G., Frasca, F., Zafeiriou, S., and Bronstein, M. M. Improving graph neural network expressivity via subgraph isomorphism counting. *arXiv preprint*, 2020.

Bruna, J. and Mallat, S. Invariant scattering convolution networks. *PAMI*, 35(8):1872–1886, 2013.

Böker, J., Levie, R., Huang, N., Villar, S., and Morris, C. Fine-grained expressivity of graph neural networks. In *NeurIPS*, 2023.

Cai, C., Hy, T. S., Yu, R., and Wang, Y. On the connection between mpnn and graph transformer. In *ICML*, 2023.

Cappart, Q., Chételat, D., Khalil, E. B., Lodi, A., Morris, C., and Veličković, P. Combinatorial optimization and reasoning with graph neural networks. In *IJCAI*, 2021.

Chandramoorthy, N., Loukas, A., Gatmiry, K., and Jegelka, S. On the generalization of learning algorithms that do not converge. In *NeurIPS*, 2022.

Chen, S., Lim, S., Mémoli, F., Wan, Z., and Wang, Y. Weisfeiler-Lehman meets Gromov-Wasserstein. In *ICML*, 2022.

Chen, Z., Villar, S., Chen, L., and Bruna, J. On the equivalence between graph isomorphism testing and function approximation with GNNs. In *NeurIPS*, 2019.

Chen, Z., Chen, L., Villar, S., and Bruna, J. Can graph neural networks count substructures? *NeurIPS*, 2020.

Chen, Z., Mao, H., Li, H., Jin, W., Wen, H., Wei, X., Wang, S., Yin, D., Fan, W., Liu, H., and Tang, J. Exploring the potential of large language models (LLMs) in learning on graphs. *SIGKDD Explorations Newsletter*, 25(2):42–61, 2024.

Chuang, C.-Y. and Jegelka, S. Tree mover's distance: Bridging graph metrics and stability of graph neural networks. In *NeurIPS*, 2022.

Cohen, J. M., Kaur, S., Li, Y., Kolter, J. Z., and Talwalkar, A. Gradient descent on neural networks typically occurs at the edge of stability. In *ICLR*, 2021.

Cotta, L., Morris, C., and Ribeiro, B. Reconstruction for powerful graph representations. In *NeurIPS*, 2021.

Deac, A., Bacon, P.-L., and Tang, J. Graph neural induction of value iteration. *ICML, Workshop on Graph Representation Learning and Beyond (GRL+)*, 2020.

Di Giovanni, F., Giusti, L., Barbero, F., Luise, G., Lio, P., and Bronstein, M. M. On over-squashing in message passing neural networks: The impact of width, depth, and topology. In *ICML*, 2023a.

Di Giovanni, F., Rusch, T. K., Bronstein, M. M., Deac, A., Lackenby, M., Mishra, S., and Veličković, P. How does over-squashing affect the power of gnns? *arXiv preprint*, 2023b.

Dimitrov, R., Zhao, Z., Abboud, R., and İsmail İlkan Ceylan. PlanE: representation learning over planar graphs. In *NeurIPS*, 2023.

Ding, K., Xu, Z., Tong, H., and Liu, H. Data augmentation for deep graph learning: A survey. *ACM SIGKDD Explorations Newsletter*, 24(2):61–77, 2022.

Du, S. S. and Lee, J. D. On the power of over-parametrization in neural networks with quadratic activation. In *ICLR*, 2018.

Du, S. S., Hou, K., Salakhutdinov, R., Póczos, B., Wang, R., and Xu, K. Graph neural tangent kernel: Fusing graph neural networks with graph kernels. In *NeurIPS*, 2019.

Duval, A., Mathis, S. V., Joshi, C. K., Schmidt, V., Miret, S., Malliaros, F. D., Cohen, T., Lio, P., Bengio, Y., and Bronstein, M. M. A hitchhiker's guide to geometric GNNs for 3D atomic systems. *arXiv preprint*, 2023.

Dwivedi, V. P., Rampášek, L., Galkin, M., Parviz, A., Wolf, G., Luu, A. T., and Beaini, D. Long range graph benchmark. In *NeurIPS*, 2022.

Easley, D. and Kleinberg, J. *Networks, Crowds, and Markets: Reasoning About a Highly Connected World*. Cambridge University Press, 2010.

Elsken, T., Metzen, J. H., and Hutter, F. Neural architecture search: A survey. *Journal of Machine Learning Research*, 20:55:1–55:21, 2019.

Fatemi, B., Halcrow, J., and Perozzi, B. Talk like a graph: Encoding graphs for large language models. *arXiv preprint*, 2023.

Fey, M. and Lenssen, J. E. Fast graph representation learning with PyTorch Geometric. In *ICLR, Workshop on Representation Learning on Graphs and Manifolds*, 2019.

Flam-Shepherd, D., Zhu, K., and Aspuru-Guzik, A. Language models can learn complex molecular distributions. *Nature Communications*, 13(1):3293, 2022.

Franks, B. J., Morris, C., Velingker, A., and Geerts, F. Weisfeiler-leman at the margin: When more expressivity matters. In *ICML*, 2024.

Frasca, F., Bevilacqua, B., Bronstein, M. M., and Maron, H. Understanding and extending subgraph GNNs by rethinking their symmetries. In *NeurIPS*, 2022.

Garg, V. K., Jegelka, S., and Jaakkola, T. S. Generalization and representational limits of graph neural networks. In *ICLR*, 2020.

Geerts, F. and Reutter, J. L. Expressiveness and approximation properties of graph neural networks. In *ICLR*, 2022.

Gervens, T. and Grohe, M. Graph similarity based on matrix norms. In *MFCS*, 2022.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. Neural message passing for quantum chemistry. In *ICLR*, 2017.

Godwin, J., Schaarschmidt, M., Gaunt, A. L., Sanchez-Gonzalez, A., Rubanova, Y., Veličković, P., Kirkpatrick, J., and Battaglia, P. Simple GNN regularisation for 3d molecular property prediction and beyond. In *ICLR*, 2022.

Grohe, M. *Descriptive Complexity, Canonisation, and Definable Graph Structure Theory*. Cambridge University Press, 2017.

Grohe, M. The logic of graph neural networks. In *LICS*, 2021.

Grohe, M. The descriptive complexity of graph neural networks. In *LICS*, 2023.

Gruver, N., Sriram, A., Madotto, A., Wilson, A. G., Zitnick, C. L., and Ulissi, Z. W. Fine-tuned language models generate stable inorganic materials as text. In *ICLR*, 2024.

Han, X., Jiang, Z., Liu, N., and Hu, X. G-mixup: Graph data augmentation for graph classification. In *ICML*, 2022.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *CVPR*, 2016.

Hopcroft, J. E. and Wong, J. K. Linear time algorithm for isomorphism of planar graphs (preliminary report). In *STOC*, 1974.

Hu, W., Fey, M., Zitnik, M., Dong, Y., Ren, H., Liu, B., Catasta, M., and Leskovec, J. Open graph benchmark: Datasets for machine learning on graphs. In *NeurIPS*, 2020.

Huang, T., Chen, T., Fang, M., Menkovski, V., Zhao, J., Yin, L., Pei, Y., Mocanu, D. C., Wang, Z., Pechenizkiy, M., and Liu, S. You can have better graph neural networks by not training weights at all: Finding untrained GNNs tickets. In *LoG*, 2022.

Jegelka, S. Theory of graph neural networks: Representation and learning. *arXiv preprint*, 2022.

Joshi, C. K., Bodnar, C., Mathis, S. V., Cohen, T., and Liò, P. On the expressive power of geometric graph neural networks. In *ICML*, 2023.

Keriven, N., Bietti, A., and Vaiter, S. Convergence and stability of graph convolutional networks on large random graphs. In *NeurIPS*, 2020.

Le, T. and Jegelka, S. Limits, approximation and size transferability for gnns on sparse graphs via graphops. In *NeurIPS*, 2023.

Levie, R. A graphon-signal analysis of graph neural networks. In *NeurIPS*, 2023.

Levie, R., Huang, W., Bucci, L., Bronstein, M., and Kutyniok, G. Transferability of spectral graph convolutional neural networks. *Journal of Machine Learning Research*, 22 (272):1–59, 2021.

Li, G., Müller, M., Ghanem, B., and Koltun, V. Training graph neural networks with 1000 layers. In *ICML*, 2021.

Li, P. and Leskovec, J. The expressive power of graph neural networks. *Graph Neural Networks: Foundations, Frontiers, and Applications*, 2022.

Liao, R., Urtasun, R., and Zemel, R. S. A PAC-Bayesian approach to generalization bounds for graph neural networks. In *ICLR*, 2021.

Lim, D., Robinson, J., Jegelka, S., and Maron, H. Expressive sign equivariant networks for spectral geometric learning. In *NeurIPS*, 2023a.

Lim, D., Robinson, J. D., Zhao, L., Smidt, T. E., Sra, S., Maron, H., and Jegelka, S. Sign and basis invariant networks for spectral graph representation learning. In *ICLR*, 2023b.

Liu, G., Inae, E., Luo, T., and Jiang, M. Rationalizing graph neural networks with data augmentation. *ACM Transactions on Knowledge Discovery from Data*, 18(4), 2024.

Liu, S., Nie, W., Wang, C., Lu, J., Qiao, Z., Liu, L., Tang, J., Xiao, C., and Anandkumar, A. Multi-modal molecule structure-text model for text-based retrieval and editing. *arXiv preprint*, 2022.

Lobacheva, E., Kodryan, M., Chirkova, N., Malinin, A., and Vetrov, D. P. On the periodic behavior of neural network training with batch normalization and weight decay. In *NeurIPS*, 2021.

Maron, H., Ben-Hamu, H., Shamir, N., and Lipman, Y. Invariant and equivariant graph networks. In *ICLR*, 2019.

Maskey, S., Lee, Y., Levie, R., and Kutyniok, G. Generalization analysis of message passing neural networks on large random graphs. In *NeurIPS*, 2022.

Mirhoseini, A., Goldie, A., Yazgan, M., Jiang, J. W., Songhori, E., Wang, S., Lee, Y.-J., Johnson, E., Pathak, O., Nazi, A., et al. A graph placement methodology for fast chip design. *Nature*, 594(7862):207–212, 2021.

Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., and Grohe, M. Weisfeiler and Leman go neural: Higher-order graph neural networks. In *AAAI*, 2019.

Morris, C., L., Y., Maron, H., Rieck, B., Kriege, N. M., Grohe, M., Fey, M., and Borgwardt, K. Weisfeiler and Leman go machine learning: The story so far. *arXiv preprint*, 2021.

Morris, C., Geerts, F., Tönshoff, J., and Grohe, M. WL meet VC. In *ICML*, 2023.

Müller, L., Galkin, M., Morris, C., and Rampásek, L. Attending to graph transformers. *arXiv preprint*, 2023.

Puny, O., Lim, D., Kiani, B., Maron, H., and Lipman, Y. Equivariant polynomials for graph neural networks. In *ICLR*, 2023.

Qian, C., Rattan, G., Geerts, F., Morris, C., and Niepert, M. Ordered subgraph aggregation networks. In *NeurIPS*, 2022.

Qian, C., Chételat, D., and Morris, C. Exploring the power of graph neural networks in solving linear optimization problems. In *AISTATS*, 2024a.

Qian, C., Manolache, A., Ahmed, K., Zeng, Z., den Broeck, G. V., Niepert, M., and Morris, C. Probabilistically rewired message-passing neural networks. In *ICLR*, 2024b.

Rosenbluth, E., Tönshoff, J., Ritzert, M., Kisin, B., and Grohe, M. Distinguished in uniform: Self attention vs. virtual nodes. In *ICLR*, 2024.

Ruiz, L., Chamon, L. F. O., and Ribeiro, A. Graphon neural networks and the transferability of graph neural networks. In *NeurIPS*, 2020.

Rusch, T. K., Bronstein, M. M., and Mishra, S. A survey on oversmoothing in graph neural networks. *arXiv preprint*, 2023.

Sato, R. A survey on the expressive power of graph neural networks. *arXiv preprint*, 2020.

Scarselli, F., Gori, M., Tsoi, A. C., Hagenbuchner, M., and Monfardini, G. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2009.

Scarselli, F., Tsoi, A. C., and Hagenbuchner, M. The Vapnik-Chervonenkis dimension of graph and recursive neural networks. *Neural Networks*, pp. 248–259, 2018.

Simmons, H. E. and Maggio, J. E. Synthesis of the first topologically non-planar molecule. *Tetrahedron Letters*, 22(4):287–290, 1981.

Ståhlberg, S., Bonet, B., and Geffner, H. Learning general optimal policies with graph neural networks: Expressive power, transparency, and limits. In *ICAPS*, 2021.

Stokes, J., Yang, K., Swanson, K., Jin, W., Cubillos-Ruiz, A., Donghia, N., MacNair, C., French, S., Carfrae, L., Bloom-Ackerman, Z., Tran, V., Chiappino-Pepe, A., Badran, A., Andrews, I., Chory, E., Church, G., Brown, E., Jaakkola, T., Barzilay, R., and Collins, J. A deep learning approach to antibiotic discovery. *Cell*, pp. 688–702.e13, 2020.

Sui, Y., Wu, Q., Wu, J., Cui, Q., Li, L., Zhou, J., Wang, X., and He, X. Unleashing the power of graph data augmentation on covariate distribution shift. In Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., and Levine, S. (eds.), *NeurIPS*, volume 36, 2023.

Tahmasebi, B., Lim, D., and Jegelka, S. The power of recursion in graph neural networks for counting substructures. In *AISTATS*, 2023.

Tang, H. and Liu, Y. Towards understanding generalization of graph neural networks. In *ICML*, 2023.

Topping, J., Giovanni, F. D., Chamberlain, B. P., Dong, X., and Bronstein, M. M. Understanding over-squashing and bottlenecks on graphs via curvature. In *ICLR*, 2022.

Veličković, P., Badia, A. P., Budden, D., Pascanu, R., Banino, A., Dashevskiy, M., Hadsell, R., and Blundell, C. The CLRS algorithmic reasoning benchmark. In *ICML*, 2022.

Wang, M., Yu, L., Zheng, D., Gan, Q., Gai, Y., Ye, Z., Li, M., Zhou, J., Huang, Q., Ma, C., Huang, Z., Guo, Q., Zhang, H., Lin, H., Zhao, J., Li, J., Smola, A. J., and Zhang, Z. Deep graph library: Towards efficient and scalable deep learning on graphs. *arXiv preprint*, 2019.

Wang, X. and Zhang, M. Pytorch geometric high order: A unified library for high order graph neural network. *arXiv preprint*, 2023a.

Wang, Y. and Zhang, M. Towards better evaluation of GNN expressiveness with BREC dataset. *arXiv preprint*, 2023b.

Weisfeiler, B. *On Construction and Identification of Graphs*. Springer, 1976.

Weisfeiler, B. and Leman, A. The reduction of a graph to canonical form and the algebra which appears therein. *Nauchno-Technicheskaya Informatsia*, 2(9):12–16, 1968.

White, C., Safari, M., Sukthanker, R., Ru, B., Elsken, T., Zela, A., Dey, D., and Hutter, F. Neural architecture search: Insights from 1000 papers. *arXiv preprint*, 2023.

Wong, F., Zheng, E. J., Valeri, J. A., Donghia, N. M., Anahtar, M. N., Omori, S., Li, A., Cubillos-Ruiz, A., Krishnan, A., Jin, W., Manson, A. L., Friedrichs, J., Helbig, R., Hajian, B., Fiejtek, D. K., Wagner, F. F., Soutter, H. H., Earl, A. M., Stokes, J. M., Renner, L. D., and Collins, J. J. Discovery of a structural class of antibiotics with explainable deep learning. *Nature*, 2023.

Wu, S., Sun, F., Zhang, W., Xie, X., and Cui, B. Graph neural networks in recommender systems: A survey. *ACM Computing Surveys*, 55(5):97:1–97:37, 2023.

Wu, Y., Wang, X., Zhang, A., He, X., and Chua, T.-S. Discovering invariant rationales for graph neural networks. In *ICLR*, 2022.

Xu, K., Hu, W., Leskovec, J., and Jegelka, S. How powerful are graph neural networks? In *ICLR*, 2019.

Xu, K., Li, J., Zhang, M., Du, S., Kawarabayashi, K., and Jegelka, S. What can neural networks reason about? In *ICLR*, 2020.

Xu, K., Zhang, M., Jegelka, S., and Kawaguchi, K. Optimization of graph neural networks: Implicit acceleration by skip connections and more depth. In *ICLR*, 2021a.

Xu, K., Zhang, M., Li, J., Du, S. S., Kawarabayashi, K.-I., and Jegelka, S. How neural networks extrapolate: From feedforward to graph neural networks. In *ICLR*, 2021b.

Xu, M., Yuan, X., Miret, S., and Tang, J. Protst: Multimodality learning of protein sequences and biomedical texts. In *ICML*, 2023.

Xu, W. and Du, S. S. Over-parameterization exponentially slows down gradient descent for learning a single neuron. In *COLT*, 2023.

Yehudai, G., Fetaya, E., Meirom, E., Chechik, G., and Maron, H. From local structures to size generalization in graph neural networks. In *ICML*, 2021.

Ying, C., Cai, T., Luo, S., Zheng, S., Ke, G., He, D., Shen, Y., and Liu, T.-Y. Do transformers really perform badly for graph representation? *NeurIPS*, 2021.

Zhang, B., Fan, C., Liu, S., Huang, K., Zhao, X., Huang, J., and Liu, Z. The expressive power of graph neural networks: A survey. *arXiv preprint*, 2023a.

Zhang, B., Feng, G., Du, Y., He, D., and Wang, L. A complete expressiveness hierarchy for subgraph GNNs via subgraph Weisfeiler-Lehman tests. In *ICML*, 2023b.

Zhang, B., Luo, S., Wang, L., and He, D. Rethinking the expressive power of GNNs via graph biconnectivity. In *ICLR*, 2023c.

Zhang, J., Li, H., Sra, S., and Jadbabaie, A. Neural network weights do not converge to stationary points: An invariant measure perspective. In *ICML*, 2022.

Zhao, J., Zhuo, L., Shen, Y., Qu, M., Liu, K., Bronstein, M., Zhu, Z., and Tang, J. Graphtext: Graph reasoning in text space. *arXiv preprint*, 2023a.

Zhao, L. and Akoglu, L. Pairnorm: Tackling oversmoothing in gnns. In *ICLR*, 2020.

Zhao, Q., Ren, W., Li, T., Xu, X., and Liu, H. Graphgpt: Graph learning with generative pre-trained transformers. *arXiv preprint arXiv:2401.00529*, 2023b.

Zhou, Y., Kutyniok, G., and Ribeiro, B. OOD link prediction generalization capabilities of message-passing GNNs in larger test graphs. *NeurIPS*, 2022.

Zhu, W., Wen, T., Song, G., Wang, L., and Zheng, B. On structural expressive power of graph transformers. *arXiv preprint*, 2023.