Ramanujan Edge-Popup: Finding Strong Lottery Tickets with Ramanujan Graph Properties for Efficient DNN Inference Execution

Hikari Otsuka
Ángel López García-Arias
Thiem Van Chu
Yasuyuki Okoshi
Kazushi Kawamura
Masato Motomura

Department of Information and Communications Engineering
Tokyo Institute of Technology
4259 Nagatsuta-cho, Midori-ku, Yokohama, Kanagawa 226-8501 Japan
otsuka.hikari@artic.iir.titech.ac.jp

Abstract— As Strong Lottery Tickets (SLTs) can build highly accurate neural networks from random weights and binary masks, specialized SLT hardware enables efficient inference. Its performance, however, depends on the inference accuracy of an SLT found in its training process. We propose Ramanujan Edge-Popup, which explores SLTs through the lens of spectral graph theory and obtains sparse and accurate SLTs. The experiment with VGG-11 using CIFAR-10 shows that Ramanujan Edge-Popup achieves 5.78% better accuracy than Edge-Popup with 97.02% sparsity.

I. Introduction

Strong Lottery Ticket (SLT) is a sparse subnetwork that achieves high accuracy comparable to trained dense Deep Neural Networks (DNN) within an over-parameterized randomly weighted DNN. The accurate subnetwork can be obtained by masking the random weights.

SLT has attracted attention as one of the methods for reducing the amount of DNN operations. Recently, Hiddenite [4], the specialized hardware for SLTs, enabled efficient inference. It reconstructs the SLT with only a binary mask—*supermask* [13]—and a random seed because a random number generator can generate random weights from the random seed.

However, The performance of architectures for SLTs such as Hiddenite depends on the accuracy and sparsity of the SLT itself. Therefore, it is necessary to obtain SLTs that are sparser and more accurate to make such architectures more useful.

In DNN pruning, it is empirically considered that the quality of pruned networks is related to spectral graph theory. Recently, Pal *et al.* [8] showed that trained pruned networks further improve accuracy by pruning so that each layer is *Ramanujan graph*: a sparse graph that has robust connectivity in spectral graph theory. Now, in the case of not optimizing weights, i.e., searching SLTs, does knowledge based on spectral graph theory improve the performance of SLTs?

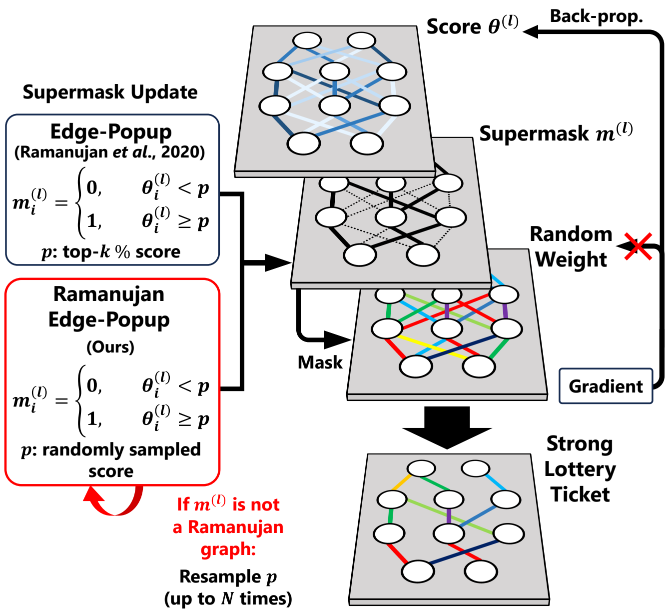

Fig. 1. The learning procedure comparison of EP and R-EP at the l-th layer.

This paper proposes the Ramanujan Edge-Popup (R-EP) algorithm, which searches sparser and highly accurate SLTs through the lens of spectral graph theory. R-EP finds the supermasks such that each layer is a Ramanujan graph. It is the extension of the Edge-Popup (EP) [9], the first algorithm to search successfully for SLTs. In experiments on CIFAR-10 using VGG-11, R-EP achieves 5.78% higher accuracy (77.88%) than EP (72.10%) for the same sparsity. Interestingly, applying the layer-wise sparsity of R-EP to EP also improves the accuracy.

To our knowledge, R-EP is the first attempt to use

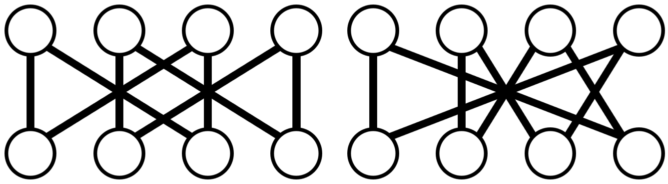

Fig. 2. Examples of bipartite graphs with the same number of edges and different connectivity. The left graph is an ordinary bipartite graph. In contrast, the right is a Ramanujan graph.

the perspective of spectral graph theory in the search for SLTs. If the graphical properties of SLTs are revealed, we can reduce its search space. It will be feasible to obtain accurate SLTs without using back-propagation by solving discrete optimization problems if the search space is sufficiently reduced.

II. Strong Lottery Ticket

To date, it has been theoretically proven that SLTs exist in DNNs under various conditions. Cunha *et al.* [1] have proven they exist in CNN architectures. Diffenderfer and Kailkhura [2] have shown that they also exist in binarized DNN. Recently, the existence of SLTs within Equivariant Neural Networks has been revealed by Ferbach *et al.* [3].

SLTs exist in the architecture of various tasks, such as generative models [12] and GNNs [3]. Therefore, the hardware for SLTs has the potential to solve various tasks efficiently.

Edge-Popup (EP) is a notable algorithm for finding SLTs. As shown in Fig. 1, EP has a score for each weight and masks weights whose scores are below a top-k % score value. Note that $(k/100) \in [0, 1]$ is a value predefined as the density of the supermask in each layer, and the sparsity means $(1 - k/100)$. It searches for more accurate supermasks by updating scores instead of weights through back-propagation.

III. Ramanujan Graph

In spectral graph theory, the Ramanujan graph is a highly connected sparse graph. For the general graph, Hoory *et al.* [6] define a Ramanujan graph as Def. 1.

Definition 1. *Let $\mathcal{G}$ be an unweighted and connected graph. Let $\tilde{\mathcal{G}}$ be the universal cover of $\mathcal{G}$. Then, the graph $\mathcal{G}$ is a Ramanujan graph if $\lambda(\mathcal{G}) \leq \rho(\tilde{\mathcal{G}})$. Here $\lambda(\mathcal{G})$ is the second-largest eigenvalue of $\mathcal{G}$ and $\rho(\tilde{\mathcal{G}})$ is the spectral radius of $\tilde{\mathcal{G}}$.*

Although it is challenging to calculate $\rho(\tilde{\mathcal{G}})$ numerically, we can determine that a graph $\mathcal{G}$ satisfying $\lambda(\mathcal{G}) \leq \sqrt{d_{avgL}(\mathcal{G}) - 1} + \sqrt{d_{avgR}(\mathcal{G}) - 1} \leq \rho(\tilde{\mathcal{G}})$ is a Ramanujan graph by using the Cor. 1 given by Hoory [5] since

Algorithm 1: Ramanujan Edge-Popup

input: Dataset $\boldsymbol{D} = \{(\boldsymbol{x}_i, y_i)\}$, number of layers L, number of samples N,
scores $\boldsymbol{\Theta} = \{\boldsymbol{\theta}^{(1)}, \boldsymbol{\theta}^{(2)}, ..., \boldsymbol{\theta}^{(L)}\}$,
sparsities $\boldsymbol{S} = \{s^{(1)}, s^{(2)}, ..., s^{(L)}\}$,
score thresholds $\boldsymbol{P} = \{p^{(1)}, p^{(2)}, ..., p^{(L)}\}$

for $(\boldsymbol{x}_i, y_i) \in \boldsymbol{D}$ **do**
  for l **in** $1, 2, ..., L$ **do**
    Sampling the sparsities
    $\boldsymbol{S} = \{s^{(l)}\} \cup \{s_1, s_2, ..., s_N\}$ s.t. $s_i \in [0, 1]$;
    Sort $\boldsymbol{S}$ in descending order;
    for s **in** $\boldsymbol{S}$ **do**
      $p = \text{Percentile}(\boldsymbol{\theta}^{(l)}, s)$;
      Get a mask $\boldsymbol{m}^{(l)}$ using $\boldsymbol{\theta}^{(l)}$ and p;
      Calculate Δ_R for $\boldsymbol{m}^{(l)}$;
      if $\Delta_R \geq 0$ **then**
        /* Satisfy the property of Ramanujan graph */
        $s^{(l)} = s$;
        $p^{(l)} = p$;
        break;
      end
    end
  end
  Edge-Popup$(\boldsymbol{\Theta}, \boldsymbol{P})$;
end

each layer of NN can be regarded as an irregular bipartite graph. Here $d_{avgL}(\mathcal{G})$ and $d_{avgR}(\mathcal{G})$ are left and right average degree of the bipartite graph $\mathcal{G}$, respectively.

Corollary 1. *For irregular bipartite graph $\mathcal{G}$ with minimal degree at least two, the spectral radius of $\tilde{\mathcal{G}}$ satisfies $\rho(\tilde{\mathcal{G}}) \geq \sqrt{d_{avgL}(\mathcal{G}) - 1} + \sqrt{d_{avgR}(\mathcal{G}) - 1}$.*

Recent experimental results have shown a relationship between the robust connectivity of a Ramanujan graph and the accuracy of the trained subnetwork. Pal *et al.* [8] found that pruning the dense DNN so that the pruned network preserves Ramanujan graph properties improved the accuracy of the trained subnetwork. We are motivated by the work of Pal *et al.* to extend the idea to SLT search. Fig. 2 shows the example of an ordinary bipartite graph and Ramanujan graph.

IV. Ramanujan Edge-Popup

This section proposes R-EP for finding the SLTs where each layer is a Ramanujan graph. As shown in Fig. 1, R-EP, like EP, explores the supermasks by updating the score corresponding to each weight. However, R-EP uses a randomly sampled value p within the score $\boldsymbol{\theta}^{(l)}$ as a score threshold for masking weights instead of the top-k % score, unlike EP. R-EP samples N thresholds based

TABLE I
Comparison of EP, R-EP, and R-EP* with VGG-11 using CIFAR-10. EP* means EP using the same layer-wise sparsity of the supermask searched by R-EP.

Algorithm	Top-1 Test Acc. [%]	Sparsity [%]
EP	72.10	97.02
R-EP	**77.88**	97.02
EP*	**78.85**	97.02

on the $(N-1)$ random sparsities and the sparsity of the supermask in the previous iteration. Here, a threshold for sparsity s is obtained as the $(100 \times s)$-th percentile of $\boldsymbol{\theta}^{(l)}$. Supermask update is completed by finding the supermask, which is a Ramanujan graph, or trying out all the thresholds. If the supermask $\boldsymbol{m}^{(l)}$ obtained by using a threshold is a Ramanujan graph, i.e., $\Delta_R := \sqrt{d_{avgL}(\boldsymbol{m}^{(l)}) - 1} + \sqrt{d_{avgR}(\boldsymbol{m}^{(l)}) - 1} - \lambda(\boldsymbol{m}^{(l)}) \geq 0$, the SLT using the supermask is also a Ramanujan graph. Hence, R-EP uses the threshold. It facilitates the search for sparser supermasks by trying the threshold in order from largest to smallest. Algo. 1 provides the pseudocode of R-EP in one epoch.

V. Experiments

This section evaluates the performance of R-EP on image classification using the experimental setups as shown in Sec. A. Sec. B show that the SLT, where each layer is a Ramanujan graph, significantly improves performance over the one obtained by EP.

A. Experimental Settings

We evaluate R-EP with VGG-11 [11] using the CIFAR-10 [7] dataset. We use stochastic gradient descent (SGD) with momentum of 0.9 and weight decay of 0.0001. We train VGG-11 for 100 epochs, starting with the learning rate of 0.01 and multiplying by 0.1 after 50 and 75 epochs. We initialize weights to use Signed Kaiming Constant (SKC) [9] with the scaling factor of $\sqrt{1/(1-s)}$. Here s is the sparsity of each layer, and it varies dynamically with each update of the supermask. R-EP samples 101 thresholds.

B. R-EP vs. EP

Tab. I compares EP and R-EP for the same overall sparsity. R-EP improves accuracy by 5.78% over EP. Interestingly, EP*, which is the EP using the same layer-wise sparsity of the supermask searched by R-EP, achieves high accuracy (78.85%) equivalent to R-EP. These results suggest that the concept of spectral graph theory can lead to more accurate supermasks.

VI. Conclusion and Future Works

This paper has shown that searching SLTs through the lens of spectral graph theory significantly improves search performance over EP. As far as we know, this is the first time the concept of Ramanujan graphs has been introduced into the context of SLTs. The finding that utilizing spectral graph theory can lead to highly sparse and accurate SLTs suggests exploring SLTs to use a completely different approach, such as searching ones based on only graph properties from the conventional one.

Future works will extend R-EP to more challenging datasets such as CIFAR-100 and ImageNet [10]. Furthermore, we aim to elucidate the graph properties of SLTs theoretically and to establish a method that can search for SLTs on the HW using their graph properties.

Acknowledgements

This work was supported in part by JSPS KAKENHI Grant Number JP23H05489.

References

[1] Arthur da Cunha, Emanuele Natale, and Laurent Viennot. Proving the lottery ticket hypothesis for convolutional neural networks. In *Proc. Int. Conf. Learn. Repr.*, 2021.

[2] James Diffenderfer and Bhavya Kailkhura. Multi-prize lottery ticket hypothesis: Finding accurate binary neural networks by pruning a randomly weighted network. In *Proc. Int. Conf. Learn. Repr.*, 2021.

[3] Damien Ferbach, Christos Tsirigotis, Gauthier Gidel, and Joey Bose. A general framework for proving the equivariant strong lottery ticket hypothesis. In *Proc. Int. Conf. Learn. Repr.*, 2023.

[4] Kazutoshi Hirose et al. Hiddenite: 4k-pe hidden network inference 4d-tensor engine exploiting on-chip model construction achieving 34.8-to-16.0tops/w for cifar-100 and imagenet. In *Proc. IEEE Int. Solid-State Circuits Conf.*, volume 65, pages 1–3, 2022.

[5] Shlomo Hoory. A lower bound on the spectral radius of the universal cover of a graph. *J. Combin. Theory, Ser. B*, 93(1):33–43, 2005.

[6] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bull. Amer. Math. Soc.*, 43(4):439–561, 2006.

[7] Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

[8] Bithika Pal, Arindam Biswas, Sudeshna Kolay, Pabitra Mitra, and Biswajit Basu. A study on the ramanujan graph property of winning lottery tickets. In *Proc. Int. Conf. Mach. Learn.*, pages 17186–17201. PMLR, 2022.

[9] Vivek Ramanujan, Mitchell Wortsman, Aniruddha Kembhavi, Ali Farhadi, and Mohammad Rastegari. What's hidden in a randomly weighted neural network? In *Proc. IEEE Comput. Soc. Conf. Comput. Vis. and Pattern Recognit.*, pages 11893–11902, 2020.

[10] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115:211–252, 2015.

[11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[12] Sangyeop Yeo, Yoojin Jang, Jy-yong Sohn, Dongyoon Han, and Jaejun Yoo. Can we find strong lottery tickets in generative models? In *Proc. AAAI Conf. on Artif. Intell.*, volume 37, pages 3267–3275, 2023.

[13] Hattie Zhou, Janice Lan, Rosanne Liu, and Jason Yosinski. Deconstructing lottery tickets: Zeros, signs, and the supermask. *Proc. Adv. Neural Inform. Process. Syst.*, 32, 2019.