Deep Learning on Graph Coloring

Haven Cook, Andrew Erickson, Ethan Rapa

Problem Description:

Graph coloring is a widely applicable procedure that aims to classify sets of vertices within a graph such that no two vertices connected by an edge are the same color. While a simple concept, this is an NP-hard problem, and therefore has no efficient solution for large-scale inputs. The best we can do is approximate, and several algorithms exist to do so, all of them attempting to balance the number of colors used to color the graph (lower is better, as it approaches the optimal solution) with computation time. Applications of graph coloring exist in many practical areas, such as job scheduling and compiler register allocation. Initially, our aim is to explore current research on the use of GNNs and other deep learning models as they relate to graph coloring. Overall, our goal is to see whether or not a GNN or fully-attentional graph transformer could extract features of colored graphs as training data that allow for more optimal solutions to be found in less time.

Data Description:

We will focus on curating datasets with small undirected graphs. The current dataset we are focused on is the ZINC[9] graph library, which was used for training the Chromatic Self-Attention mechanism[8]. This dataset contains an average of 23.15 nodes and 49.81 edges per graph. There are ~220011 graphs provided in this dataset.

Timeline:

Deadline	Task
10/6	Project Proposal Due
10/15	Validate dataset
10/20	Have all papers read and make presentation
10/23	Project Intermediate Presentation
11/20	Test final model on validated dataset
12/01	Write up findings
12/04	Project Final Presentation

12/11	Project Report Due
12/18	Report Review Due

Notes:

IEEE Code dataset for reg allocation: https://ieeexplore.ieee.org/abstract/document/10720063
https://ieee-dataport.org/documents/rigset-uel-register-interference-graphs-dataset#files

Labeled graph dataset: https://mat.tepper.cmu.edu/COLOR/instances.html#XXDSJ
https://mat.tepper.cmu.edu/COLOR02/

Model Jupyter Example: https://github.com/amazon-science/gcp-with-gnns-example

https://arxiv.org/pdf/1912.03700

https://arxiv.org/pdf/2109.05948

Relevant Publications:

[1] F. M. Q. Pereira, “A Survey on Register Allocation”.

[2] “Chaitin’s algorithm,” *Wikipedia*. Oct. 12, 2024. Accessed: Oct. 03, 2025. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Chaitin%27s_algorithm&oldid=1250833324

[3] M. Osama, M. Truong, C. Yang, A. Buluc, and J. Owens, “Graph Coloring on the GPU,” in *2019 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, Rio de Janeiro, Brazil: IEEE, May 2019, pp. 231–240. doi: 10.1109/IPDPSW.2019.00046.

[4] “Instruction scheduling,” *Wikipedia*. July 05, 2025. Accessed: Oct. 03, 2025. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Instruction_scheduling&oldid=1298983227

[5] K. Stoffelen, “Instruction Scheduling and Register Allocation on ARM Cortex-M”.

[6] “Register allocation,” *Wikipedia*. Sept. 08, 2025. Accessed: Oct. 03, 2025. [Online]. Available: https://en.wikipedia.org/w/index.php?title=Register_allocation&oldid=1310234205

[7] G. J. Chaitin and P. O. Box, “REGISTER ALLOCATION & SPILLING VIA GRAPH COLORING”.

[8] R. Menegaux, E. Jehanno, M. Selosse, and J. Mairal, “Self-Attention in Colors: Another Take on Encoding Graph Structure in Transformers,” 2023, *arXiv*. doi: 10.48550/ARXIV.2304.10933.

[9] M. J. A. Schuetz, J. K. Brubaker, Z. Zhu, and H. G. Katzgraber, “Graph Coloring with Physics-Inspired Graph Neural Networks,” *Phys. Rev. Research*, vol. 4, no. 4, p. 043131, Nov. 2022, doi: 10.1103/PhysRevResearch.4.043131.

[10]
W. Li, R. Li, Y. Ma, S. O. Chan, D. Pan, and B. Yu, “Rethinking Graph Neural Networks for the Graph Coloring Problem,” Aug. 19, 2022, *arXiv*: arXiv:2208.06975. doi: 10.48550/arXiv.2208.06975.