Nusantara EduLearn Journal
Vol. 4 No. 2 (2026), p. 1-13
Available online at https://cendikianusantara.com/index.php/EduLearn

From Graph Theory to Computational Thinking: Performance Trade-Offs in Graph Coloring Algorithms

Ajeng Zulfaida, Olief Ilmandira Ratu Farisi*
Universitas Nurul Jadid, Indonesia

DOI: https:// 10.61987/edulearn.v1i1.56

Article History:
Received: 07 January 2026
Revised: 13 March 2026
Accepted: 19 April 2026

Keywords:
Graph Coloring, Algorithm Performance, Computational Efficiency

Abstract:
Graph coloring is an important topic in graph theory with applications in scheduling, traffic management, network optimization, and other conflict-avoidance problems. This study aims to compare the performance of the Greedy, Welch–Powell, and Sequential algorithms in terms of computational time and the number of colors used across graphs with different sizes and structures. A quantitative comparative approach was employed using graph datasets obtained from open sources. The datasets were represented as adjacency matrices and processed using the three graph coloring algorithms under a consistent computational procedure. The results show that the Greedy and Welch–Powell algorithms produced the same number of colors for most datasets, while the Greedy algorithm generally required the least computational time. For larger graphs, the Greedy and Welch–Powell algorithms produced fewer colors than the Sequential algorithm in some cases. The Sequential algorithm demonstrated relatively fast execution but tended to require more colors, whereas the Welch–Powell algorithm achieved competitive coloring results with greater computational time. These findings imply that algorithm selection should consider both coloring quality and computational efficiency. The comparative framework also has educational implications by supporting algorithmic thinking, computational thinking, and evidence-based problem-solving in graph theory learning.

INTRODUCTION

The increasing complexity of modern society requires effective methods for organizing, scheduling, and optimizing interconnected activities. Problems involving transportation, communication networks, educational scheduling, geographic information systems, and power distribution often require decisions about allocating limited resources without creating conflicts (Asgarli, Krehbiel, & Levinson, 2025; Meng et al., 2024). Graph theory provides an important mathematical framework for representing such problems through vertices (V) and edges (E) (Farisi, 2023). A graph can be represented as (G=(V,E)), where (V) is a non-empty set of points called vertices and (E) is a set of lines connecting pairs of vertices, known as edges (Tao, Aihara, & Chen, 2026). Among the important applications of graph theory is graph coloring (Bai, Li, Xu, & Zhang, 2023). This concept enables conflicting objects or activities to receive different labels or colors, thereby supporting efficient decision-making (Feghali, Le, & Le, 2025). Therefore,

***Email Correspondence:** farisi133@unuja.ac.id*
This is an open access article under the CC BY 4.0 license. © 2026, the author(s)
https://creativecommons.org/licenses/by/4.0/

investigating effective graph coloring methods is important because their performance can influence the efficiency of problem-solving processes across social, technological, and educational contexts.

Graph coloring is theoretically defined as the process of assigning labels or colors to graph elements under the condition that adjacent elements must not receive the same color (D'ascenzo, D'emidio, Flammini, & Monaco, 2024). The historical development of graph coloring began with the map-coloring problem, which later became one of the fundamental problems in graph theory (Haviv & Rabinovich, 2025). Graph coloring can be classified into vertex coloring, edge coloring, and map coloring, depending on the graph elements being colored (Crombez et al., 2023). In this study, the concepts of computational time and chromatic number are used to evaluate algorithmic effectiveness. The chromatic number represents the minimum number of colors required to color a graph without violating adjacency constraints (Rahman, Dufresne, & Medvedev, 2024). Computational time, meanwhile, reflects algorithmic efficiency (Yakut & Karagöz, 2025). These theoretical concepts provide a basis for systematically examining and comparing the performance of several graph coloring algorithms under graphs with different sizes and orders.

Despite the widespread usefulness of graph coloring, determining an appropriate coloring method remains a significant problem because different algorithms may produce different results when applied to graphs with varying characteristics (Angkhanawin, Deger, Pritchard, & Adams, 2025). Society increasingly depends on efficient scheduling, resource allocation, and conflict-management systems, including transportation services, computer networks, geographic information systems, electrical engineering, and power systems. Graph coloring has also been implemented in reservation systems and university course scheduling (Deeds, Sabale, Kayali, & Suciu, 2024). However, selecting an algorithm solely based on its application domain may not guarantee optimal performance. This issue is also relevant to education, particularly in teaching graph theory and algorithmic problem-solving, because students need to understand how algorithm selection affects efficiency and solution quality in practical mathematical problems (Vanderbush & Weber, 2026).

Previous studies have investigated the application and comparison of graph coloring algorithms in several specific contexts (Mobarsa, Shukla, Maheta, Sata, & Jarboui, 2025). Various algorithms, such as the Greedy algorithm, Welch–Powell algorithm, Tabu Search algorithm, and Sequential algorithm, have been applied to optimize classroom scheduling, seminar scheduling, and map coloring (Gao & Hu, 2024). These studies demonstrate that graph coloring algorithms can effectively address practical problems involving scheduling and allocation (Kattemölle, 2024). They also provide valuable evidence regarding differences in algorithmic performance. However, most previous research has concentrated on comparisons between only two algorithms and has been limited to particular case studies (Maus, 2023). Consequently, the findings cannot yet provide a sufficiently comprehensive understanding of how multiple graph coloring algorithms perform under different graph conditions.

The limitations of previous studies indicate an important research gap concerning comprehensive comparisons among several graph coloring algorithms. Although earlier research has successfully demonstrated the usefulness of

individual algorithms in specific applications, the comparison of algorithmic performance across graphs with systematically varying sizes and orders remains limited (Antoniadis, Broersma, & Meng, 2026). Furthermore, previous studies have frequently emphasized application outcomes or chromatic numbers without simultaneously providing sufficient attention to computational time (Wang, Yan, & Jin, 2024). This limitation is important because an algorithm that produces a satisfactory coloring result may require considerably more computational resources than another method (Shen, Li, Yuan, & Chen, 2025; Terci & Boz, 2025). In educational contexts, this gap also affects the learning of graph theory because students and researchers need evidence-based comparisons to understand the advantages and limitations of different algorithms. Therefore, this study contributes by evaluating multiple graph coloring algorithms using common performance indicators, thereby providing a broader basis for selecting appropriate algorithms.

The novelty of this study lies in its comparative approach to evaluating several graph coloring algorithms simultaneously under graphs with different sizes and orders. Rather than limiting the analysis to two algorithms within a single practical case study, this research examines algorithmic effectiveness through two fundamental indicators: computational time and chromatic number. This approach provides a more systematic understanding of the relationship between graph characteristics and algorithm performance. The study also contributes to the development of graph theory research by integrating computational and educational perspectives. From a computational perspective, the results can support the selection of efficient algorithms for solving graph coloring problems. From an educational perspective, the comparison can provide meaningful learning material for graph theory, algorithm design, and mathematical problem-solving. Understanding algorithmic strengths and limitations can also help learners develop analytical thinking when solving complex problems.

Based on the identified research gap, the main problem addressed in this study is determining which graph coloring algorithm is the most effective when evaluated across graphs of varying sizes and orders. Specifically, this research investigates differences in computational time and chromatic number among several graph coloring algorithms. The underlying argument of this study is that no single algorithm necessarily provides the best performance under every graph condition. The effectiveness of an algorithm is expected to depend on graph characteristics, particularly the size and order of the graph, as well as the performance criteria used for evaluation. Therefore, a comparative analysis is necessary to identify the relative strengths and limitations of each method. The preliminary assumption is that algorithms with lower computational time may not always produce the smallest chromatic number, whereas methods producing better coloring results may require greater computational effort. Through this investigation, the study aims to provide evidence supporting appropriate algorithm selection for practical applications and graph theory learning.

RESEARCH METHODS

This study employed a comparative research design with a quantitative approach to systematically evaluate and compare the performance of graph coloring algorithms. The comparative design was selected because the primary

objective of the study was to identify differences in algorithm performance based on measurable indicators, namely computational time and chromatic number (Brighen, Slimani, Rezgui, & Kheddouci, 2023; Schidler & Szeider, 2023). The research was conducted in a computational environment using Google Colab and the NetworkX library; therefore, no specific geographical research location was required. Google Colab was selected as the computational platform because it provides an accessible environment for implementing algorithms and processing graph datasets (Saad, 2024). Three graph coloring algorithmsthe Greedy algorithm, the Welch–Powell algorithm, and the Sequential algorithmwere implemented using the Python programming language (Yusak et al., 2024). The use of the same computational environment for all algorithms was intended to maintain consistency and fairness in the performance comparison. The research instruments used to support the implementation and evaluation process are presented in Table 1.

Table 1. Description of Research Instruments

Instrument	Function/Description
Graph Datasets	Used as test data for the implementation and comparison of graph coloring algorithms. The datasets consist of graphs with different numbers of vertices and edges to represent varying levels of graph complexity.
Google Colab Platform with the NetworkX Library	Used as the computational platform for implementing the graph coloring algorithms, processing graph datasets, and evaluating computational performance.

Table 1 shows the instruments used to support the implementation and evaluation of the graph coloring algorithms. The graph datasets served as the primary test data, allowing the algorithms to be evaluated on graphs with different numbers of vertices and edges. Meanwhile, Google Colab and the NetworkX library provided the computational environment for implementing the algorithms and measuring their performance (Zhu et al., 2025). The use of the same computational platform and library for all algorithms was intended to maintain consistency during the experimental process. These instruments were selected to ensure that the comparison of the Greedy, Welch–Powell, and Sequential algorithms could be conducted systematically based on computational time and chromatic number.

The population of this study consisted of graph datasets with different structural characteristics, particularly variations in the numbers of vertices and edges. The samples were selected purposively from open-access datasets obtained from Kaggle and previous studies. The use of purposive sampling was based on the need to select datasets that met specific criteria related to the research objectives. The selected graphs represented different sizes and levels of complexity, ranging from small-order graphs to large-order graphs, allowing the performance of the algorithms to be examined under different conditions. Data collection was conducted by gathering the selected graph datasets and preparing them for computational analysis. Each graph was represented using an appropriate data structure, including an adjacency matrix, to facilitate the identification of relationships among vertices and the implementation of graph coloring algorithms (Faiqoh and Farisi, 2024). The datasets were then processed using the same procedures to maintain consistency throughout the experiment.

The collected data were analyzed quantitatively by implementing each graph coloring algorithm on the same graph datasets and recording the resulting chromatic number and computational time. The analysis was conducted by comparing the performance of the Greedy, Welch–Powell, and Sequential algorithms across graphs with different sizes and levels of complexity. Computational time was used to evaluate the efficiency of each algorithm, while the chromatic number was used to evaluate the effectiveness of the coloring results (Meng et al., 2024). The results obtained from each algorithm were subsequently compared to identify their respective strengths and limitations under different graph conditions. The findings were organized and presented in tables to facilitate comparison and interpretation. Through this comparative analysis, the study aimed to determine which algorithm demonstrated the most effective overall performance based on the balance between computational efficiency and the ability to produce a minimum or relatively low chromatic number.

RESULTS AND DISCUSSION

Results

This study compares the performance of three graph coloring algorithms, namely the Greedy, Welch–Powell, and Sequential algorithms. The evaluation focuses on two main indicators: the chromatic number produced by each algorithm and the computational time required to process each graph. To provide a comprehensive comparison, the algorithms were tested on graph datasets with different numbers of vertices and edges, ranging from small graphs to relatively large and dense graphs. The variation in graph size and structure allows the performance of each algorithm to be observed under different levels of computational complexity. The results are also relevant from an educational perspective because the comparison provides concrete examples that can be used to support the learning of graph coloring, algorithmic efficiency, and problem-solving in graph theory. The characteristics of the graph datasets used in the experiment are presented in Table 2.

Table 2. Graph Datasets

Graph	Vertices	Edges
Graph 1	6	8
Graph 2	10	18
Graph 3	20	30
Graph 4	22	87
Graph 5	138	493
Graph 6	500	62,624
Graph 7	900	307,350
Graph 8	1,000	246,708

Table 2 presents the characteristics of the graph datasets used to evaluate the three graph coloring algorithms. The datasets vary substantially in both the number of vertices and the number of edges. Graph 1 contains the smallest number of vertices, with 6 vertices and 8 edges, whereas Graph 8 contains the largest number of vertices, with 1,000 vertices and 246,708 edges. Graph 7 has 900 vertices and 307,350 edges, making it the graph with the largest number of edges among the datasets. These

variations provide different levels of graph complexity for evaluating algorithm performance. From a learning perspective, the datasets can also be used to demonstrate how changes in graph size and density influence the computational behavior of graph coloring algorithms. The chromatic numbers obtained from the three algorithms are presented in Table 3.

Table 3. Chromatic Numbers Produced by Each Algorithm

Graph	Greedy Algorithm	Welch–Powell Algorithm	Sequential Algorithm
Graph 1	3	3	3
Graph 2	4	4	4
Graph 3	4	4	4
Graph 4	8	8	8
Graph 5	11	11	11
Graph 6	69	69	73
Graph 7	204	204	204
Graph 8	121	121	122

Table 3 presents the chromatic numbers generated by the Greedy, Welch–Powell, and Sequential algorithms for each graph dataset. The results show that all three algorithms produced identical chromatic numbers for Graphs 1 through 5. For Graph 6, the Greedy and Welch–Powell algorithms produced a chromatic number of 69, while the Sequential algorithm produced 73. For Graph 7, all three algorithms produced the same chromatic number of 204. For Graph 8, the Greedy and Welch–Powell algorithms produced 121, whereas the Sequential algorithm produced 122. Thus, the Greedy and Welch–Powell algorithms generated lower chromatic numbers than the Sequential algorithm for Graphs 6 and 8, while all algorithms produced the same result for the other datasets. These findings indicate that the Greedy and Welch–Powell algorithms were more effective than the Sequential algorithm in minimizing the chromatic number for the datasets where differences occurred. In graph theory learning, these results illustrate that different coloring strategies may produce identical solutions for some graphs but different solutions as graph complexity increases.

The computational time required by each algorithm was also measured to evaluate their computational efficiency. Each algorithm was executed using the same graph datasets and computational environment, allowing the execution times to be compared under consistent conditions. The results provide an additional perspective because an algorithm may produce a similar or lower chromatic number while requiring more computational time. The computational time recorded for each algorithm is presented in Table 4.

Table 4. Computational Time Required by Each Algorithm (seconds)

Graph	Greedy Algorithm	Welch–Powell Algorithm	Sequential Algorithm
Graph 1	0.0001	0.0002	0.0001
Graph 2	0.0002	0.0004	0.0002
Graph 3	0.0004	0.0006	0.0004
Graph 4	0.0009	0.0014	0.0009
Graph 5	0.0029	0.0038	0.0029
Graph 6	0.1089	0.5685	0.1344
Graph 7	0.5474	3.8481	0.6641
Graph 8	0.4766	1.9658	0.5662

Table 4 shows the computational time required by each algorithm to process the eight graph datasets. The results indicate that the Greedy and Sequential algorithms required nearly identical computational times for Graphs 1 through 5. The Greedy algorithm required 0.0001, 0.0002, 0.0004, 0.0009, and 0.0029 seconds for Graphs 1 through 5, respectively, while the Sequential algorithm required the same computational times for these graphs. In contrast, the Welch–Powell algorithm consistently required more computational time, with values of 0.0002, 0.0004, 0.0006, 0.0014, and 0.0038 seconds for Graphs 1 through 5. The difference became more apparent for the larger graphs. For Graph 6, the Greedy algorithm required 0.1089 seconds, compared with 0.5685 seconds for Welch–Powell and 0.1344 seconds for Sequential. For Graph 8, the respective times were 0.4766, 1.9658, and 0.5662 seconds. These results indicate that the Greedy algorithm generally demonstrated the lowest computational time, followed by the Sequential algorithm, whereas the Welch–Powell algorithm required the longest processing time.

The most substantial difference in computational time occurred in Graph 7, which contains 900 vertices and 307,350 edges. The Greedy algorithm completed the coloring process in 0.5474 seconds, followed by the Sequential algorithm with a computational time of 0.6641 seconds. In comparison, the Welch–Powell algorithm required 3.8481 seconds. Therefore, the Welch–Powell algorithm required approximately seven times longer than the Greedy algorithm and nearly 5.8 times longer than the Sequential algorithm. This result demonstrates that differences in computational efficiency become more noticeable when the graph contains a larger number of vertices and edges. The finding also provides an important learning implication because students can observe that algorithm selection is not determined solely by the quality of the coloring result. Computational efficiency should also be considered when solving larger graph problems.

The difference in computational time can be attributed to the operational mechanism of the Welch–Powell algorithm. After ordering the vertices according to their degrees, the algorithm repeatedly searches for non-adjacent vertices that can be assigned the same color. The repeated verification of adjacency relationships introduces additional computational overhead, resulting in longer execution times. Consequently, although the Welch–Powell algorithm produced chromatic numbers that were equal to or lower than those of the Sequential algorithm, it required substantially more computational time for the larger datasets. The Greedy algorithm demonstrated the most favorable computational performance, while the Sequential algorithm also maintained relatively low computational times. From an educational perspective, this comparison can support the teaching of algorithm analysis by demonstrating the trade-off between solution quality and computational efficiency. Students can use these results to develop a more critical understanding of how algorithmic strategies influence performance when graph size and complexity increase.

Discussion

The findings of this study indicate that the performance of graph coloring algorithms varies according to graph characteristics and the evaluation criteria used. In terms of chromatic number, the Greedy, Welch–Powell, and Sequential algorithms produced identical results for Graphs 1 through 5. This finding is consistent with previous research showing that different coloring strategies may generate the same coloring results when applied to graphs with relatively simple structures (Asgarli et al., 2025; Feghali et al., 2025). However, differences emerged in Graphs 6 and 8, where the Greedy and Welch–Powell algorithms produced lower chromatic numbers than the Sequential algorithm. This result indicates that algorithmic behavior becomes more distinguishable as graph size and complexity increase. From a theoretical perspective, these findings demonstrate that the effectiveness of a graph coloring algorithm cannot be evaluated using a single graph characteristic. From an educational perspective, such differences provide students with opportunities to understand how graph structure influences algorithmic behavior and to develop analytical thinking through direct comparison of algorithmic solutions.

The computational-time results provide another important perspective on algorithm performance. The Greedy and Sequential algorithms generally required similar processing times for the smaller datasets, whereas the Welch–Powell algorithm consistently required more computational time. The difference became particularly substantial for the larger graphs, with the most notable difference occurring in Graph 7. This finding is consistent with the general observation that algorithms involving additional vertex ordering and repeated adjacency checking may require greater computational effort (Haviv & Rabinovich, 2025; Yakut & Karagöz, 2025). The result also complements previous findings concerning the relatively fast execution of Sequential or First-Fit coloring approaches (D'ascenzo et al., 2024; Rahman et al., 2024). However, the present study shows that computational efficiency alone does not necessarily indicate superior coloring performance. The Sequential algorithm was relatively fast but produced higher chromatic numbers for Graphs 6 and 8. This trade-off is pedagogically valuable because students can learn that algorithm selection involves balancing solution quality and computational efficiency rather than simply choosing the fastest method. Such learning can strengthen computational thinking, particularly the ability to evaluate alternative procedures based on measurable performance criteria.

The results have important theoretical implications for graph coloring and algorithm analysis. The findings demonstrate that chromatic number and computational time represent complementary measures of algorithm performance. The chromatic number represents the minimum number of colors required for a proper vertex coloring, while computational time can be used to evaluate the efficiency of an algorithm in processing a given problem (Deeds et al., 2024; Vanderbush & Weber, 2026). An algorithm may produce a lower number of colors but require substantially more computational resources, while another algorithm may operate faster but produce a less efficient coloring solution. This relationship is clearly illustrated by the comparison between the Sequential and Welch–Powell algorithms. The Sequential algorithm maintained relatively low

computational time but generated higher chromatic numbers for Graphs 6 and 8. Conversely, the Welch–Powell algorithm achieved chromatic numbers comparable to the Greedy algorithm but required considerably longer execution times. In learning contexts, these findings can be transformed into inquiry-based activities in which students formulate hypotheses, implement algorithms, compare outputs, and interpret performance differences. Therefore, graph coloring becomes not only a mathematical topic but also a context for developing computational reasoning and evidence-based decision-making.

From a practical perspective, the findings provide useful guidance for selecting graph coloring algorithms according to the characteristics and requirements of a problem. The Greedy algorithm demonstrated the most favorable overall balance because it consistently required relatively low computational time while producing chromatic numbers equal to or lower than those generated by the Sequential algorithm. The Welch–Powell algorithm also produced competitive chromatic numbers, but its higher computational time may become a limitation when processing larger graphs. Meanwhile, the Sequential algorithm can remain useful when computational speed is prioritized, although its coloring results may be less efficient for some larger graphs. These differences have practical relevance in scheduling, network management, resource allocation, and other conflict-avoidance problems (Kattemölle, 2024). In educational practice, these real-world applications can help instructors connect abstract graph theory concepts with authentic problem solving situations. Students can be encouraged to model scheduling or resource allocation problems as graphs and then determine which coloring algorithm provides the most appropriate solution. This approach can support conceptual understanding while simultaneously developing problem-solving, programming, and decision-making skills.

The state of the art of this study lies in its simultaneous comparison of three graph coloring algorithms across graph datasets with substantially different sizes and structures. Previous investigations commonly focused on particular applications or compared a limited number of algorithms (Feghali et al., 2025). In contrast, this study evaluates the Greedy, Welch–Powell, and Sequential algorithms using the same datasets and two consistent performance indicators: chromatic number and computational time. This approach provides a more comprehensive perspective on algorithmic performance. The novelty is particularly evident in identifying situations in which algorithms that produce similar coloring results demonstrate substantially different computational efficiency. The results show that similar coloring quality does not necessarily imply equivalent algorithmic performance. From an educational perspective, this comparative framework also offers a structured learning model for introducing algorithm analysis. Students can observe how the same graph produces different computational behaviors and can use empirical evidence to evaluate algorithmic strengths and limitations. Thus, the study connects graph theory, programming practice, and computational thinking within a single learning context.

The overall contribution of this study is the provision of empirical evidence for understanding the relative strengths and limitations of three commonly used graph coloring algorithms. The findings show that the Greedy and Welch–Powell algorithms can produce lower chromatic numbers than the Sequential algorithm

for certain larger graphs, while the Greedy algorithm demonstrates superior computational efficiency across the tested datasets. These findings provide a basis for selecting algorithms according to specific objectives, whether minimizing the number of colors, reducing computational time, or achieving an appropriate balance between both criteria. Beyond algorithm comparison, the study contributes to education by providing a concrete computational framework through which students can understand graph coloring beyond theoretical definitions. By implementing algorithms and analyzing actual computational results, learners can develop algorithmic thinking, computational thinking, logical reasoning, and problem-solving skills. The comparative results also encourage students to justify algorithm selection using evidence rather than intuition. Therefore, the contribution of this research extends from algorithmic evaluation to the development of meaningful, practice-oriented graph theory learning experiences.

CONCLUSION

This study concludes that the Greedy algorithm provides the most favorable overall balance between computational efficiency and color minimization among the three graph coloring algorithms evaluated, while the Welch–Powell algorithm produces comparable coloring results but requires greater computational time, and the Sequential algorithm offers relatively fast execution at the cost of using more colors for some larger and more complex graphs. The main lesson from these findings is that selecting a graph coloring algorithm should not be based solely on execution speed or the number of colors, but should consider the trade-off between solution quality and computational efficiency in relation to graph characteristics. The practical implication of these findings indicates that algorithm selection in conflict-avoidance systems should be aligned with specific contextual priorities, such as speed or output quality, while the educational implication provides a practical learning framework for understanding algorithmic thinking, computational efficiency, and evidence-based algorithm selection. Scientifically, this study contributes empirical evidence through a systematic comparison of three algorithms using graphs with different sizes and two complementary performance indicators. Nevertheless, this study is limited by the number and characteristics of the graph datasets, the three algorithms evaluated, and the use of a single computational environment. Future research should therefore examine more diverse graph structures, larger datasets, additional coloring algorithms, and alternative performance indicators, as well as investigate how comparative graph coloring activities can be integrated more deeply into mathematics and computer science learning.

ACKNOWLEDGEMENT

The authors would like to express their sincere gratitude to all individuals who provided support, guidance, and valuable contributions throughout the completion of this research. The authors also appreciate the availability of open-source graph datasets and computational resources that supported the implementation and evaluation of the graph coloring algorithms. Special appreciation is extended to Lee Haechan for his encouragement, motivation, and

positive support, which served as an important source of inspiration throughout the process of writing and completing this article. Finally, the authors are grateful to everyone who contributed directly or indirectly to the completion of this study.

CONFLICTS OF INTEREST

The authors declare that there are no conflicts of interest related to the publication of this article.

USE OF ARTIFICIAL INTELLIGENCE (AI)-ASSISTED TECHNOLOGY

Artificial intelligence (AI)-assisted technologies were used during the preparation of this manuscript. ChatGPT was used to assist with improving the clarity, grammar, sentence structure, and overall academic writing of the manuscript. Gemini was used to assist in improving and refining the title of the article. The authors reviewed, revised, and verified the AI-assisted outputs and remain fully responsible for the accuracy, originality, interpretation, and final content of the manuscript.

REFERENCES

Angkhanawin, T., Deger, A., Pritchard, J., & Adams, C. (2025). Graph Coloring via Quantum Optimization on a Rydberg-Qudit Atom Array. *Quantum Science and Technology, 11*. https://doi.org/10.1088/2058-9565/ae3b6d

Antoniadis, A., Broersma, H., & Meng, Y. (2026). Incorporating Predictions in Online Graph Coloring Algorithms. *Discrete Applied Mathematics, 379*, 434–445. https://doi.org/10.1016/j.dam.2025.08.063

Asgarli, S., Krehbiel, S., & Levinson, H. (2025). Coloring Graphs as Complete Graph Invariants. *Discrete Mathematics, 349*, Article 115245.

Bai, X., Li, B., Xu, C., & Zhang, X. (2023). Cooperative Coloring of Some Graph Families. *Discrete Mathematics, 347*, Article 113973.

Brighen, A., Slimani, H., Rezgui, A., & Kheddouci, H. (2023). A New Distributed Graph Coloring Algorithm for Large Graphs. *Cluster Computing, 27*, 875–891. https://doi.org/10.1007/s10586-023-03988-x

Crombez, L., Fonseca, G., Fontan, F., Gérard, Y., Gonzalez-Lorenzo, A., Lafourcade, P., … Zheng, D. (2023). Conflict Optimization for Binary CSP Applied to Minimum Partition into Plane Subgraphs and Graph Coloring. *ACM Journal of Experimental Algorithmics, 28*, 1–13. https://doi.org/10.1145/3588869

D'Ascenzo, A., D'Emidio, M., Flammini, M., & Monaco, G. (2024). Digraph k-Coloring Games: New Algorithms and Experiments. *Journal of Artificial Intelligence Research, 81*, 163–202. https://doi.org/10.1613/jair.1.16174

Deeds, K., Sabale, D., Kayali, M., & Suciu, D. (2024). COLOR: A Framework for Applying Graph Coloring to Subgraph Cardinality Estimation. *Proceedings of the VLDB Endowment, 18*, 130–143. https://doi.org/10.14778/3705829.3705834

Faiqoh, N., & Farisi, O. I. R. (2024). Application of Graph Coloring in Learning Schedule Arrangement at MAN 1 Probolinggo. *Nusantara Journal of Computers and Its Application, 9*.

Farisi, O. (2023). *Teori Graf dan Aplikasinya* (Vol. 222). Pustaka Nurja.

Feghali, C., Le, H.-O., & Le, V. B. (2025). The Parameterized Complexity of Strong Conflict-Free Vertex-Connection Colorability. https://doi.org/10.1016/j.dam.2025.12.025

Gao, M., & Hu, J. (2024). Graph Coloring Algorithm Based on Minimal Cost Graph Neural Network. *IEEE Access, 12*, 168000–168009. https://doi.org/10.1109/ACCESS.2024.3439352

Haviv, I., & Rabinovich, D. (2025). A Near-Optimal Kernel for a Coloring Problem. *Discrete Applied Mathematics, 377*, 66–73.

Kattemölle, J. (2024). Edge Coloring Lattice Graphs. *arXiv*. https://doi.org/10.48550/arXiv.2402.08752

Maus, Y. (2023). Distributed Graph Coloring Made Easy. *ACM Transactions on Parallel Computing, 10*, 1–21. https://doi.org/10.1145/3605896

Meng, L., Shao, Y., Yuan, L., Lai, L., Cheng, P., Li, X., … Zhou, J. (2024). A Survey of Distributed Graph Algorithms on Massive Graphs. *ACM Computing Surveys, 57*, 1–39. https://doi.org/10.1145/3694966

Mobarsa, D., Shukla, M., Maheta, N., Sata, A., & Jarboui, S. (2025). Novel Representation of Complexity of Investment Casting Using Injective Coloring and Graph Adaptive Consensus Mechanism. *Scientific Reports, 15*. https://doi.org/10.1038/s41598-025-27113-7

Rahman, A., Dufresne, Y., & Medvedev, P. (2024). Compression Algorithm for Colored de Bruijn Graphs. *Algorithms for Molecular Biology, 19*. https://doi.org/10.1186/s13015-024-00254-6

Saad, R. (2024). Parallel Connectivity in Edge-Colored Complete Graphs: Complexity Results. *Graphs and Combinatorics, 40*. https://doi.org/10.1007/s00373-023-02747-4

Schidler, A., & Szeider, S. (2023). SAT-Boosted Tabu Search for Coloring Massive Graphs. *ACM Journal of Experimental Algorithmics, 28*, 1–19. https://doi.org/10.1145/3603112

Shen, Y., Li, X., Yuan, T., & Chen, S. (2025). Scalable Graph Coloring Optimization Based on Spark GraphX Leveraging Partition Asymmetry. *Symmetry, 17*, Article 1177. https://doi.org/10.3390/sym17081177

Tao, P., Aihara, K., & Chen, L. (2026). Brain-Inspired Chaotic Graph Backpropagation for Combinatorial Optimization. *IEEE Transactions on Neural Networks and Learning Systems, 37*, 3488–3500. https://doi.org/10.1109/TNNLS.2025.3650570

Terci, G. S., & Boz, B. (2025). Coloring Dynamic Graphs with a Similarity and Pool-Based Evolutionary Algorithm. *IEEE Access, 13*, 38054–38075. https://doi.org/10.1109/ACCESS.2025.3546108

Vanderbush, K., & Weber, M. (2026). Neural Algorithmic Reasoning for Approximate k-Coloring with Recursive Warm Starts. *Machine Learning: Science and Technology, 7*. https://doi.org/10.1088/2632-2153/ae6f6e

Wang, X., Yan, X., & Jin, Y. (2024). A Graph Neural Network with Negative Message Passing and Uniformity Maximization for Graph Coloring. *Complex & Intelligent Systems, 10*, 4445–4455. https://doi.org/10.1007/s40747-024-01355-w

Yakut, S., & Karagöz, E. (2025). A Graph-Theoretic Solution to NP-Complete Sudoku Puzzles: Malatya Centrality for Large-Scale Optimization. *IEEE Access, 13*, 182737–182750. https://doi.org/10.1109/ACCESS.2025.3622191

Yusak, M. Y., Putri, D. F., & Syaripuddin, S. (2024). Penerapan Pewarnaan Graf Menggunakan Algoritma Welch-Powell pada Penjadwalan Mata Pelajaran: Studi Kasus Penjadwalan Mata Pelajaran SMP Terpadu Rea Kaltim Kembang Janggut. *Journal of Mathematics Education and Science, 7*(2), 177–183. https://doi.org/10.32665/james.v7i2.2272

Zhu, E., Zhang, Y., Sun, H., Wei, Z., Pedrycz, W., Liu, C., & Xu, J. (2025). HyColor: An Efficient Heuristic Algorithm for Graph Coloring. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 56*, 681–695. https://doi.org/10.1109/TSMC.2025.3629990