A study on constructing an efficient examination scheduling system

Linh Nguyen Mai Vu[1], Hieu Chi Tran[1], Anh Thi Tram Nguyen[1*]

[1]Ho Chi Minh City Open University, Ho Chi Minh City, Vietnam
[*]Corresponding author: tramanh.nguyen@ou.edu.vn

ARTICLE INFO

DOI:10.46223/HCMCOUJS.tech.en.14.1.2920.2024

Received: August 21st, 2023

Revised: February 03rd, 2024

Accepted: February 07th, 2024

Keywords:
coloring; graph; scheduling problem; welsh-powell

ABSTRACT

The objective of this study is to investigate the final exam scheduling process at the Ho Chi Minh City Open University and develop an automated exam scheduling application. Our primary objectives are to prevent students from having conflicting exam schedules and to ensure that no student has to take more than two exams on the same day. This research focuses on applying graph coloring algorithms to the problem of automatic exam scheduling. Our research findings indicate that the graph coloring algorithm is highly effective for automated exam scheduling. This study has the potential to expand and support the development of an automatic exam scheduling and management system, in line with our overall goals. We conduct the experiments on the practical data at HCMCOU and obtain promising results.

1. Introduction

Most universities worldwide are now using the credit-based training method to offer more flexibility to students and institutions. However, this credit-based approach could potentially lead to conflicts between courses when it comes to the final exam scheduling problems (Chengjun, 2016). The most significant drawback is the difficulty of arranging exam schedules for students every semester. There are several issues that require attention when scheduling final exams, such as conflicting schedules for students with overlapping courses, too many exams for one student on the same day, a shortage of available examination facilities caused by uneven student enrollment, and a shortage of resources to supervise exams resulting from the diverse workloads of faculty members. Students and the university can be negatively affected if these issues don’t get resolved carefully.

Regarding the problem of exam schedules, two most commonly used approaches are often studied: the graph coloring algorithm (Runa & Siddhartha, 2017) and the genetic algorithm (Murat & Hanefi, 2018). The graph coloring algorithm assigns vertice for each course, colors them based on whether that course has overlapping students, and then arranges those courses with different colors into non-overlapping time slots. This algorithm allows taking the number of students participating in each exam and the exam duration into account when solving the scheduling problem and it is considered as the most straightforward approach to the exam scheduling problem. Alternatively, the genetic algorithm implements the principles of evolution and takes a different approach from the graph coloring algorithm. It operates through crossover, mutation, and selection rules to find the most optimal schedule in many possible exam schedules available. Both approaches have their strength and can be applied to the exam scheduling problem. Choosing between the two algorithms depends deeply on the specific requirements of the application and the scenario of the problem (Murat & Hanefi, 2018; Runa & Siddhartha, 2017).

By analyzing the final exam scheduling process at Ho Chi Minh City Open University, our study aims to develop a completely automated exam scheduling application to improve the overall fairness and effectiveness of the final exam schedule. The automated exam scheduling application is designed to take students' course registration results for the semester, encompassing Student ID, Course Code, Course Title, Student Class, and Class Group, as input. The primary objective is to generate an optimized and conflict-free exam schedule as output, with a focus on ensuring that no student is assigned more than two exams on the same day and addressing constraints related to courses with overlapping enrollments.

In the context of the automated exam scheduling problem at Ho Chi Minh City Open University, there are several factors should be considered. One of the main concerns is the large number of courses and student enrolments every semester. Additionally, it's important to not only make sure that various learning facilities are utilized effectively but also ensure that the distribution of exams between learning facilities will not be a burden to students. Considering these factors, the Genetic Algorithm appears to be an excellent choice due to its suitability for complex exam scheduling problems with a large number of courses and constraints. However, we have ultimately opted for the Welsh-Powell graph coloring algorithm because it has a simple structure, appropriate for tackling a new and complex problem. Another significant advantage of the graph coloring algorithm is its ability to support the learning process. By experimenting and customizing the algorithm to meet our specific research requirements, we can gain a deeper understanding of the problem and accumulate knowledge in this field.

2. Related works

In modern educational management, exam scheduling is a critical issue and often be quite difficult to resolve. This process involves creating reasonable exam schedules for students during the final exam period and ensuring that the schedule meets all the requirements. Such as avoiding scheduling conflicts between courses with overlapping student registration and limiting the number of exams on any given day for any given student. Previous research (Jacek, Jan, & AHG, 1983; Michael, Gilbert, & Sau, 1996) has shown that this scheduling problem is classified as an NP-hard problem. With a considerable amount of students, courses, and conditions involved, the search space becomes extensive, making it challenging to obtain the most reasonable exam schedule. At the time that this study was made, there was no polynomial-time algorithm to solve all instances of this problem with an optimal solution.

In 2004, representatives conducted a comprehensive survey to analyze the various approaches adopted for solving the automated scheduling problem from 1990 to 2003 (Edmund, Patrick, Greet, & Hendrik, 2004), which evaluated the efficacy of optimization techniques, including graph coloring algorithms and genetic algorithms. The research discusses various approaches to the scheduling problem but emphasizes the importance of adopting and combining advanced algorithms to enhance the quality of scheduling outcomes.

Automated scheduling problem has been the main object of this field for a long time, the general aim is to develop an efficient algorithm to solve this problem. One of the prominent approaches is the graph coloring algorithm known as Welsh-Powell, first introduced in a study by Welsh and Powell in 1967 (Dominic & Martin, 1967). This algorithm involves assigning colors to the vertices of a graph, with each vertice representing a course, and courses with overlapping students will have different colors. Those courses with the same color are scheduled into non-overlapping time slots, thus generating an automated exam schedule. The Welsh-Powell algorithm is considered quite effective and often produces results close to the minimum number of colors needed for the graph. It is essential to be mindful that although this approach may prove useful, it cannot guarantee to identify the smallest possible quantity of colors necessary in every scenario. The proficiency of this algorithm depends on the original vertex order and the order of adding colors to the graph.

The more optimized approach to automated scheduling is metaheuristics, they are high-level problem-solving strategies used to find approximate solutions to as difficult problems as the scheduling problem due to their ability to handle large solution spaces and complex constraints. It is designed to navigate through vast search space to find the near-optimal solutions within a reasonable amount of time, some common metaheuristics applied to scheduling include Genetic Algorithms, Simulated Annealing, and Tabu Search. Research in 2020 discusses how different metaheuristics are combined or modified to create a hybrid approach to scheduling problems by present case studies where these hybrid metaheuristics have been applied to real-world Resource-Constrained Project Scheduling Problems (Pellerin, Nathalie, & François, 2020). This paper offers to provide an understanding of the strengths and weaknesses of different hybridization methods in the context of scheduling problems. Another paper in 2023 also proved the preeminence of metaheuristics in the scheduling field by building an automated timetable generator using Genetic Algorithms (Adedayo, Esan, & Ajayi, 2023). They implement Genetic Algorithms into solving automated scheduling problems by dividing the problem into six important modules: Data encoding, Generating initial population, Computing fitness, Crossover evolution, Mutation, and Generating new population.

In the context of applying different algorithms to scheduling problems, a study conducted in 2011 (Elmahdi & Wessam, 2011) put together four different algorithms, namely Welsh-Powell, DSatur, Tabu Search, and Genetic Algorithms to research, evaluate, and compare their performance. To ensure an optimized testing environment, both simulated and real-world datasets were taken into advance to evaluate all aspects including computational speed, accuracy, and computational complexity. In their conclusion, Tabu Search and Genetic Algorithm's performance in solving the scheduling problem is better than the other algorithms. The study briefly discusses the strengths and weaknesses of those algorithms in the context of automated scheduling and also suggests several improvements to enhance and optimize them.

Past studies Michael et al. (1996); Jacek et al. (1983); Edmund et al. (2004) have played a significant role in addressing the automated exam scheduling problem; they have laid the groundwork for understanding the complexity of this problem and helped us acknowledge future challenges. By combining experience from those studies with modern technologies, our study aims to optimize the graph coloring algorithm for the exam scheduling problem at Ho Chi Minh City Open University.

3. Our procedures

3.1. Addressed issues and research goals

This section outlines the primary issues addressed in this scientific research paper, with a primary focus on achieving fairness and effectiveness in the final exam scheduling process for students, faculty, and the university as a whole. The central research goal is to ensure that no student takes more than two exams on the same day while considering essential data constraints during the exam scheduling process.

- No scheduling conflicts for courses with overlapping student enrollment: This crucial constraint is fundamental to guaranteeing a conflict-free exam schedule, where no student faces overlapping exam times between different courses. Addressing this issue is paramount to providing an equitable examination experience for all students.

- Limitation on the number of exams on any given day: Recognizing the impact of multiple exams on the same day on students' stress levels and academic performance, this constraint aims to minimize the number of exams scheduled for a student on any particular day. By implementing this constraint, the research endeavors to enhance students' well-being and academic outcomes during the exam period.

In alignment with these research goals, the subsequent sections of this paper will detail the methodologies employed and the optimization techniques utilized to address the identified issues effectively. The proposed approach aims to streamline the exam scheduling process, optimize the allocation of exams, and contribute to an improved academic environment at the Nguyen Kiem branch of Ho Chi Minh City Open University.

3.2. Data flow and approach

This study focuses on developing an automated exam scheduling system to ensure effectiveness and accuracy in creating exam schedules for the university. The system relies on the Welsh-Powell graph coloring algorithm to address our issues. The aim is to optimize the exam scheduling process and ensure fairness in the schedule for both students and the university.

Input data: a table containing information about students' course registration results for the semester, including fields such as Student ID, Course Code, Course Title, Student Class, and Class Group for each row of data. For reference, we constructed a simulated dataset for students' course registration results in one semester, as shown in Table 1.

Table 1

Simulated dataset

Student ID	**Course code**	**Course title**	**Student class**	**Class group**
1030001	ACCO1329	Audit 1	DH20KT01	AC2001
1030002	ACCO1329	Audit 2	DH20KT02	AC2002
1030003	ACCO1329	Audit 3	DH20KT02	AC2002
1030004	ACCO1329	Audit 4	DH20KT02	AC2002
1030005	ACCO1329	Audit 5	DH20KT02	AC2002
1030006	ACCO1329	Audit 6	DH20KT02	AC2002
1030007	ACCO1329	Audit 7	DH20KT01	AC2001

Source: Ho Chi Minh City Open University (n.d.)

Output data: A table presenting the exam schedule sorted by date for the courses to be held is shown in Table 2.

Table 2

Exam schedule arranged by date

Exam date	Course code	Course title	Student class
30/07/2023	BADM1390	Entrepreneurship	DH20IT02
30/07/2023	BADM1390	Entrepreneurship	DH20ITJ1, DH20IT01
30/07/2023	BADM2346	Cellular Technology	DH20SH01
30/07/2023	BADM2346	Cellular Technology	DH20SH02
30/07/2023	MATH1315	Probabiloty and Statistics	DH20AU02
31/07/2023	ACCO1329	Audit 1	DH20KT01
31/07/2023	ACCO1329	Audit 1	DH20KT02
31/07/2023	BIOT2403	Biotechnology	DH20SH02
31/07/2023	BIOT2403	Biotechnology	DH20SH01

Source: Ho Chi Minh City Open University (n.d.)

Our procedures: This section introduces three stages approach to the automated exam scheduling application for the Nguyen Kiem branch of Ho Chi Minh City Open University.

- Stage 1. Preprocessing Data: This stage preprocesses the input data into a graph representation with each node corresponding to a specific course, and establishes relationships between nodes that have overlapping student enrollment. This graph serves as the foundation for efficiently organizing courses and the relationship between them in the exam scheduling process.

- Stage 2. Implement Welsh-Powell algorithm: This stage applies the Welsh-Powell graph coloring algorithm to the graph from stage 1. This algorithm ensures that no student is assigned more than two exams on the same day while considering constraints related to courses with overlapping student enrollment.

- Stage 3. Output and Results: Following the application of the Welsh-Powell graph coloring algorithm to obtain the final exam schedule. The schedule provides the exam dates for each subject, ensuring that all constraints and preferences are satisfied. This stage presents the culmination of our efforts in achieving an automated, fair, and efficient exam schedule.

3.3. Welsh-Powell graph coloring algorithm

We utilized the Welsh-Powell graph coloring algorithm to examine the graph G(v, e) with the following definitions:

- v (vertex): represents the vertices of the graph, corresponding to the courses to be scheduled for exams.

- e (edge): represents the edges connecting two vertices (courses), v1 and v2, with overlapping student enrollment, indicating that these courses have students in common who will take the exams.

In the Welsh-Powell graph coloring algorithm, based on the graph G(v, e), the vertices (courses) will be colored in such a way that no two adjacent vertices (two courses with overlapping student enrollment) have the same color.

Input: The list of vertices v (courses) and the edges E connecting two vertices representing two courses with common students taking the exams.

Output: The list of vertices v (courses) after being colored.

Algorithm 1 Welsh-Powell Graph Coloring Algorithm

```
procedure WelshPowell(G)                                         ▷ G is the input graph
    Sort the vertices of G in descending order based on their degrees
    Initialize an empty array of colors to store the colors of vertices
    colors[vertex] ←−1 for all vertices in G
    currentColor ← 0                                             ▷ The current color index
    for each vertex v in the sorted order do
        if colors[v] = −1 then                                   ▷ Vertex is not colored
            colors[v] ← currentColor                             ▷ Color the current vertex
            for each uncolored neighbor u of v do
                if colors[u] = −1 then                           ▷ Neighbor is not colored
                    colors[u] ← currentColor       ▷ Color the neighbor with the same color
            currentColor ← currentColor + 1                      ▷ Move to the next color
```

4. Results and discussions

Through extensive research, we explored the scheduling problem in different fields, delving into graph theory and graph coloring algorithms for a deeper understanding. With the help of Elmahdi Driouch and Wessam Ajib research in 2011 (Elmahdi & Wessam, 2011), we have assessed the strengths and weaknesses of Welsh-Powell, DSatur, Tabu Search, and Genetic Algorithms, with the ultimate goal of effectively applying them to automated exam scheduling.

The main focus of the research is on the application of the Welsh-Powell graph coloring algorithm to the automated exam scheduling problem for the Ho Chi Minh City Open University. Based on the research results, this study developed and tested a model applying the Welsh-Powell algorithm to the automated exam scheduling problem, intending to ensure no student does not have to take more than two exams on the same day. The study aimed to alleviate the pressure on students during the exam period and expedite the scheduling process, which can be complex and time-consuming for the university staff.

To validate the reliability and accuracy of the exam scheduling system using the Welsh-Powell graph coloring algorithm in real-world scenarios, a comprehensive testing procedure was carried out. We meticulously designed and employed four distinct sets of test cases representing four different scenarios of input data from the real-world scenario. These test cases ensure that our program meets its requirement, to serve as a reliable and conflict-free automated exam scheduling program. Through these test cases, they showed the program's robustness, reliability, and efficiency in scheduling by analyzing its output and comparing it with the requirements of each case. The results of these tests provide insight into the program's strengths and weaknesses and serve as valuable feedback for potential improvements.

4.1. Results

4.1.1. Standard data

In this first test case, we use a standard data set, consisting of diverse subjects with a reasonable number of student enrollments and no overlapping courses (Table 3). The main objective of this test case was to evaluate the program's performance when dealing with standard exam scheduling scenarios with no overlapping or overscheduling. The results (Table 4) show the program efficiently scheduled exams without conflicts and created a well-distributed exam timetable for students. Because the given data set in this test case has no overlapping subject, every subject final exam was scheduled in one single day, ensuring that no student would face the risk of a conflicting exam schedule.

Table 3

Case 1 processed data

No.	**Course code**	**Course title**	**Overlapping courses**
1	ENGL1248	Phonetics and phonology	
2	FINA2331	Tax and business operations	
3	ITEC4415	Software testing	
4	BIOT2346	Cell biotechnology	
5	BLAW1229	Legal writing and research	

Table 4

Case 1 automated schedule

Exam date	**Course code**	**Course title**	**Student class**
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV02
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV01
29/07/2023	FINA2331	Tax and business operations	DH20CF02
29/07/2023	FINA2331	Tax and business operations	DH20CF01
29/07/2023	ITEC4415	Software testing	DH20IM01
29/07/2023	BIOT2346	Cell biotechnology	DH20SH01
29/07/2023	BIOT2346	Cell biotechnology	DH20SH02
29/07/2023	BLAW1229	Legal writing and research	DH20LA02
29/07/2023	BLAW1229	Legal writing and research	DH20LA01

4.1.2. One overlapping enrollments

The second test case was designed to demonstrate how the program processes in a scenario where two subjects have overlapping enrollments, “Phonetics and phonology” and “Academic English 3” (Table 5). By utilizing the strengths of the Welsh-Powell coloring algorithm, the program effectively schedules those two conflicted subjects “Phonetics and phonology” and “Academic English 3” on two different days to avoid conflict, and every other subject that has no conflict will be placed on the first day of the exam week (Table 6). This second test cast highlights the program’s capability to avoid conflicts for students with multiple courses, ensuring a conflict-free exam schedule.

Table 5

Case 2 processed data

No.	Course code	Course title	Overlapping courses
1	ENGL1248	Phonetics and phonology	GENG1341
2	FINA2331	Tax and business operations	
3	ITEC4415	Software testing	
4	BIOT2346	Cell biotechnology	
5	BLAW1229	Legal writing and research	
6	GENG1341	Academic English 3	ENGL1248

Table 6

Case 2 automated schedule

Exam date	Course code	Course title	Student class
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV02
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV01
29/07/2023	FINA2331	Tax and business operations	DH20CF02
29/07/2023	FINA2331	Tax and business operations	DH20CF01
29/07/2023	ITEC4415	Software testing	DH20IM01
29/07/2023	BIOT2346	Cell biotechnology	DH20SH01
29/07/2023	BIOT2346	Cell biotechnology	DH20SH02
29/07/2023	BLAW1229	Legal writing and research	DH20LA02
29/07/2023	BLAW1229	Legal writing and research	DH20LA01
30/07/2023	GENG1341	Academic English 3	NKAVNC3

4.1.3. Multiple overlapping enrollments

In the third test case, to assess the program's capability to handle complex scheduling situations, it created the most complicated scenarios with significant overlapping enrollments between multiple subjects (Table 7). Despite the increment in overlapping courses, the program generated exam schedules without any overlap (Table 8), ensuring fairness and avoiding conflicts for students with overlapping exams. With multiple overlapping subjects, the program scheduled those subjects to different days, spanning from the 29th of July to the 2nd of August, spanning from 29th July the 2nd of August, to avoid conflicting schedules for students. For example, the subject "Phonetics and phonology" is conflicting with 4 other subjects: "British-American culture", "Academic English 5", "Translation practice 2", and "Probability and Statistics." So, the program ensures that the student enrolled in those 5 subjects will not have conflicts by scheduling them on 5 different days of the exam week.

Table 7

Case 3 processed data

No.	Course code	Course title	Overlapping courses
1	ENGL1248	Phonetics and phonology	ENGL1336, GENG1343, ENGL1345, MATH1315
2	ENGL1336	British-American culture	GENG1343, ENGL1345, ENGL1248, MATH1315
3	ENGL1345	Translation practice 2	ENGL1336, GENG1343, ENGL1248, MATH1315
4	FINA2331	Tax and business operations	GENG1341, FINA2335, FINA2338
5	FINA2335	International finance	GENG1341, FINA2338, FINA2331
6	FINA2338	Corporate finance 2	GENG1341, FINA2335, FINA2331
7	GENG1341	Academic English 3	ITEC4415, ITEC4409, FINA2338, FINA2331, FINA2335
8	GENG1343	Academic English 5	ENGL1336, ENGL1345, ENGL1248, MATH1315
9	ITEC4409	Software engineering	ENGL1336, ENGL1345, ENGL1248, MATH1315
10	ITEC4415	Software testing	GENG1341, ITEC4409
11	MATH1315	Probability and statistics	ENGL1336, ENGL1345, GENG1343, ENGL1248

Table 8

Case 3 automated schedule

Exam date	Course code	Course title	Student class
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV02
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV01
29/07/2023	GENG1341	Academic English 3	NKAVNC3
30/07/2023	ENGL1336	British-American culture	DH20AV02
30/07/2023	ENGL1336	British-American culture	DH20AV01
30/07/2023	FINA2331	Tax and business operations	DH20CF02
30/07/2023	FINA2331	Tax and business operations	DH20CF01
30/07/2023	ITEC4409	Software engineering	DH20IT02
30/07/2023	ITEC4409	Software engineering	DH20IT01
31/07/2023	ENGL1345	Translation practice 2	DH20AV03
31/07/2023	ENGL1345	Translation practice 2	DH20AV01
31/07/2023	FINA2335	International finance	DH20CF01
31/07/2023	FINA2335	International finance	DH20CF01
31/07/2023	ITEC4415	Software testing	DH20IM01
01/08/2023	FINA2338	Corporate finance 2	DH20CF01
01/08/2023	FINA2338	Corporate finance 2	DH20CF02
01/08/2023	GENG1343	Academic English 5	NKAVNC5
02/08/2023	MATH1315	Probability and statistics	DH22AU02

Table 9

Case 4 student with heavy course load data

Student ID	Course code	Course title	Student class	Class group
1200001	BLAW1222	Criminal procedure law	DH20LA02	LA2002
1200001	BLAW1229	Legal writing and research	DH20LA02	LA2002
1200001	BLAW1317	Labour law	DH20LA02	LA2002
1200001	BLAW2330	Tax law	DH20LA02	LA2001

Table 10

Case 4 processed data

No.	Course code	Course title	Overlapping courses
1	BLAW1222	Criminal procedure law	GENG1341, BLAW1317, BLAW1229, BLAW2330
2	BLAW1229	Legal writing and research	GENG1341, BLAW1317, BLAW1222, BLAW2330
3	BLAW1317	Labour law	GENG1341, BLAW1317, BLAW1222, BLAW2330
4	BLAW2330	Tax law	GENG1341, BLAW1317, BLAW1222, BLAW2330
5	GENG1341	Academic English 3	BLAW1317, BLAW1222, BLAW1229, BLAW2330
6	ENGL1248	Phonetics and phonology	
7	FINA2331	Tax and business Operations	
8	ITEC4415	Software testing	
9	BIOT2346	Cell biotechnology	

Table 11

Case 4 automated schedule

Exam date	Course code	Course title	Student class
29/07/2023	BLAW1222	Criminal procedure law	DH20LA02
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV02
29/07/2023	ENGL1248	Phonetics and phonology	DH20AV02
29/07/2023	BIOT2346	Cell biotechnology	DH20SH02
29/07/2023	BIOT2346	Cell biotechnology	DH20SH01
29/07/2023	FINA2331	Tax and business operations	DH20CF01
29/07/2023	FINA2331	Tax and business operations	DH20CF02
29/07/2023	ITEC4415	Software testing	DH20IM01
30/07/2023	BLAW1229	Legal writing and research	DH20LA02
30/07/2023	BLAW1229	Legal writing and research	DH20LA01
31/07/2023	BLAW1317	Labour law	DH20LA02
01/08/2023	BLAW2330	Tax law	DH20LA01
02/08/2023	GENG1341	Academic English 3	NKAVNC3

4.2. Discussions

The biggest advantage of this algorithm is its ability to efficiently and swiftly solve the automated exam scheduling problem. Additionally, it ensures fairness in scheduling exams for every student. However, the algorithm has a few drawbacks, including the increased processing time and memory space required for storage when dealing with a large number of courses and students participating in the exam schedule due to the large number of vertex required.

By simulating various case scenarios, this study ensured that the Welsh-Powell graph coloring algorithm was evaluated and tested to fit with the real-world use case of its application. Through this process, the stability and effectiveness of the algorithm were examined when facing significant scheduling conflicts and complex constraints. This research results demonstrated the capability and efficiency of the Welsh-Powell algorithm in automated exam scheduling problems.

Despite a few drawbacks mentioned above, the automated exam scheduling program is still an advancement in processing time compared to the manual ways of exam scheduling in many universities nowadays. Incorporating the Welsh-Powell graph coloring algorithms into the final exam scheduling process, not only helps reduce the amount of time and effort put into the final exam scheduling process but also increases the reliability and fairness of the generated schedule, ensuring that no student will have conflicting exam schedules.

The research content might not be completely new, but the scope and purpose of the study have been clearly defined, focusing on the development of an effective automated exam scheduling system for the Ho Chi Minh City Open University. The study aims to contribute to the efficient organization of exams at the university by addressing these specific challenges.

5. Conclusions

Based on the results obtained from this study, we recognize the potential of the Welsh-Powell graph coloring algorithm in solving the automated exam scheduling problem. However, for problems with multiple high-dimensional data constraints and requirements, the Welsh-Powell algorithm still falls short of producing the most optimal results. In the future, to further enhance the system and achieve the goal of building an efficient automated exam scheduling and management system, further research and application of additional algorithms, including Genetic Algorithms, Mixed-Integer Programming, and Meta-heuristic algorithms, are necessary. These approaches may offer improved performance and better optimization for complex scheduling scenarios.

References

Adedayo, S., Esan, A., & Ajayi, S. (2023). Development of an automated timetable generator using genetic algorithm. *Lautech Journal of Computing and Informatics, 3*(1), 49-55.

Chengjun, S. (2016). On how to build a modern educational administration management platform in colleges and universities under the credit system. *Proceedings of the 2016 6th International Conference on Management, Education, Information and Control (MEICI 2016)*, 633-636.

Dominic, J. A. W., & Martin, B. P. (1967). An upper bound for the chromatic number of a graph and its application to timetabling problems. *The Computer Journal, 10*(1), 85-86.

Edmund, B. K., Patrick, C. D., Greet, B. V., & Hendrik, L. V. (2004). The state of the art of nurse rostering. *Journal of Scheduling, 7*(November 2004), 441-499.

Elmahdi, D., & Wessam, A. (2011). Efficient scheduling algorithms for multiantenna CDMA systems. *IEEE Transactions on Vehicular Technology, 61*(2), 521-532.

Ho Chi Minh City Open University. (n.d.). Retrieved May 10, 2023, from ou.edu.vn

Jacek, B., Jan, L. K., & AHG, K. R. (1983). Scheduling subject to resource constraints: Classification and complexity. *Discrete Applied Mathematics, 5*(1), 11-24.

Michael, W. C., Gilbert, L., & Sau, L. Y. (1996). Examination timetabling: Algorithmic strategies and applications. *Journal of The Operational Research Society, 47*(3), 373-383.

Murat, D., & Hanefi, C. M. (2018). Solving the exam scheduling problems in central exams with genetic algorithms. *Mugla Journal of Science and Technology, 4*(1), 102-115.

Pellerin, R., Nathalie, P., & François, B. (2020). A survey of hybrid metaheuristics for the resource-constrained project scheduling problem. *European Journal of Operational Research, 280*(2), 395-416.

Runa, G., & Siddhartha, R. (2017). A study on course timetable scheduling using graph coloring approach. *International Journal of Computational and Applied Mathematics, 12*(2), 469-485.