Contemporary Mathematics
http://ojs.wiserpub.com/index.php/CM/

Research Article

Airport Flight-Gate Allocation Assignment Using Local Antimagic Vertex Coloring

G. Muthumanickavel, M. Nalliah*

School of Advanced Sciences, Vellore Institute of Technology, Vellore-632 014, India
E-mail: nalliah.moviri@vit.ac.in

Received: 3 July 2024; **Revised:** 10 August 2024; **Accepted:** 21 August 2024

Abstract: The management of aviation gate assignments may significantly affect the overall productivity of an airport. This work presents a novel approach to enhancing gate assignment algorithms. The methodology used in this study incorporates graph theory and other intricate techniques, with specific focus on the concept of local antimagic vertex coloring. This study presents a comprehensive methodology for identifying optimal gate locations, which incorporates several factors such as aircraft characteristics, passenger capacity, and resource availability. Based on the results of this study, the incorporation of local antimagic vertex coloring exhibits the capacity to significantly enhance the operational effectiveness of airports via the reduction of passenger line wait times and alleviation of traffic congestion challenges. The aforementioned result was derived by using comprehensive models of the physical cosmos and conducting a meticulous analysis of empirical facts. This novel concept has the capacity to enhance the efficacy of airport gate allocation, resulting in heightened levels of customer contentment and reduced operational costs. The implementation of this procedure will facilitate the preservation of the efficiency of the operational networks for air transportation.

Keywords: airport, flight-gate allocation, local antimagic vertex coloring, congestion minimization, adaptability, airport operations

MSC: 05C90, 05C78, 05C15

1. Introduction

The study of local antimagic vertex coloring has been the focus of much research, which has established a solid basis for addressing challenging optimization problems, such as scheduling and allocation. The concept of local antimagic vertex coloring was introduced by Arumugam et al. [1]. Utami and Wijaya subsequently adapted this concept to practical scenarios, such as the organization of expatriate assignments [2]. Hongyan Li et al. demonstrated the adaptability of graph coloring techniques in solving real-world scheduling problems, specifically in the context of allocating airplane gates at airports [3]. The main objectives of previous influential studies on airport gate assignments conducted by Babic et al., Mangoubi and Mathaisel, and Yan and Huo were to reduce walking lengths and improve gate assignments [4–6]. Wen et al., Tian and Xiong, and Ju and Xu used genetic algorithms to improve gate scheduling [7–9]. Additional research on enhancing airport operations was conducted by Jiang, Wen, and Zeng et al. Research on the impacts of safety thresholds

DOI: https://doi.org/10.37256/cm.5420245242

and time delays has also been carried out by Bouras et al., Liu et al., and Zhou et al. [10–12]. The studies conducted by Ruan and Zheng, Zhang et al., and Zheng et al. contribute to the existing body of research on airport gate assignment [13–17]. This study focuses on multi-objective optimization and load balancing. Ganguli and Roy conducted research on the use of graph coloring in course timetabling, which has potential implications for scheduling athletic events [18]. Luo et al. conducted research on vertex shader models, Qin and Liao studied athlete tracking, and Li and Wang focused on decision-making models [19–21]. These studies provide methodological support for the idea of complex scheduling situations [22–27]. This study establishes the foundation for future studies that might use local antimagic vertex coloring to schedule and allocate gates in airport operations.

Fair airplane gate allocation improves resource efficiency, customer happiness, and airport operations. Due to flight lengths, delays, and airport infrastructure's geographical limits, allocating gates for incoming and departing aircraft may be difficult and time-consuming. Long-standing gate assignment strategies that maximize gate occupancy may cause congestion and operational inefficiencies. This strategy may overlook the need of distributing one's efforts among all their duties. Graph theory has become a powerful tool for solving optimization problems in several academic fields in recent years. Graph theory best represents the complex relationships between airport gates, flights, and operational limitations. Thus, it optimizes airport gate allocation.

This paper develops a new airport flight-gate allocation method using local antimagic vertex coloring, a graph theory idea with great combinatorial optimization potential. Graphing the airport's architecture is its main idea. To clarify, each gate represents a network vertex, and the connections between gates reflect operational limitations and geographical concerns. This feature ensures aircraft are evenly distributed between gates, minimizing congestion and improving traffic flow. In airport gate allocation, local antimagic vertex coloring might have several benefits. First, it distributes duty evenly across gates, decreasing congestion at any gate. The algorithm's decentralized design allows last-minute plan changes and delay management, improving system flexibility. Due to its flexibility, gate allocations may be changed in real time. This improves gate operating efficiency and the travel experience for all stakeholders.

Local antimagic vertex coloring is proposed as a solution to airport flight-gate allocation in this research. Airport gate allocation needs attention. The theoretical basis and possible benefits of the strategy to reduce congestion, improve resource allocation, and maximize operational efficiency will be discussed. The algorithm's underlying concepts and empirical assessment utilizing actual airport layouts will be examined in the following parts. Comparisons will also be done with traditional gate assignment methods. Our study and its ramifications aim to improve airport efficiency and benefit travelers.

Definition 1 Let us consider a connected graph $G = (V, E)$ with $|V| = n$ and $|E| = m$. In this context, we define a bijection $f : E \to \{1, 2, \ldots, m\}$ as a local antimagic labeling if it satisfies the condition that for any two adjacent vertices u and v, the sum of labels assigned to the edges incident to them, denoted as $w(u)$ and $w(v)$ respectively, must be distinct. Here, $w(u)$ is calculated as the summation of labels assigned to the edges in the set $E(u)$, where $E(u)$ represents the collection of edges incident to vertex u. Consequently, every local antimagic labeling induces a proper coloring of the vertices in G, where each vertex is assigned a color corresponding to its computed weight. The minimum number of colors required to achieve proper colorings through local antimagic labelings of G is referred to as the local antimagic chromatic number, denoted as $\chi_{la}(G)$ [1].

Definition 2 The idea might be conceptualised as the point where intervals cross on the real number line. Within this framework, each interval in the supplied set is represented by a single vertex, and a connection is established between every pair of vertices that correspond to two intervals with an intersection. Within the framework of an interval graph, designated as G, there exists a vertex whose degree is no more than $k-1$. If we establish the notation $w(G) = k$, then we may denote the interval with the earliest right endpoint as v. Any interval that crosses with v must inevitably intersect at its right endpoint; otherwise, v would not be the interval with the earliest right terminus. As a result, all intervals that intersect with interval v also intersect with one another, so creating a complete subgraph. Given that the order of the graph G is denoted as $w(G) = k$, it follows that the entire subgraph of G has a maximum size of k. Consequently, this implies that vertex v has at most $k-1$ neighbouring vertices. Hence, it follows that the degree of vertex v is no more than $k-1$.

2. Methods

Graph modeling:

• The problem of airport gate allocation is first addressed by using a graph. The diagram illustrates each gate represented as a node, with connections built based on pragmatic considerations and proximity. The local antimagic vertex coloring method may efficiently represent the intricate interconnections among gates, using this network as a fundamental framework.

Local antimagic vertex coloring:

• The suggested methodology requires the use of the local antimagic vertex coloring method as an essential element. The objective of this technique is to provide a systematic approach for assigning colors to the vertices, which represent gates, in a way that guarantees that the discrepancy between the total of colors of each vertex's neighboring vertices and its prior value does not surpass one. To accomplish this goal, it is necessary to ensure that the colors of the vertices' nearby neighbors remain uniform. The key characteristic of this system is in its decentralized structure, which allows for the flexibility of gate assignments to meet changes in flight schedules and unexpected circumstances. Because to the aforesaid quality of flexibility, people can effectively use resources, resulting in a significant reduction in the amount of traffic that has to be controlled.

Balanced Workload distribution:

• The primary objective of the algorithm is to provide a balanced and even allocation of flights throughout many airport terminals. The solution employs a color-coding scheme to distinguish neighboring gates that have similarities. The initiative seeks to accomplish two objectives: reducing waiting times and optimizing gate use. The algorithm's equitable allocation of tasks raises the level of service provided to clients and boosts the overall efficiency of organizations.

Adaptability and real-time updates:

• The programme has sufficient adaptability to accommodate changes in flight schedules and routes due to its unregulated architecture. Fluctuations in airline schedules may impact the level of demand at airports. Fortunately, advancements in technology enable airports to promptly modify gate assignments in response to fluctuations in airport passenger traffic. Adopting adaptable approaches is necessary to optimize gate assignment, enhance operational efficiency, reduce reliance on human intervention, and increase the probability of error-free outcomes.

Experimental evaluation:

• Empirical studies are conducted to evaluate the efficacy of the local antimagic vertex coloring method, employing real airport floor layouts and historical flight data. A comparative analysis is undertaken to evaluate the effectiveness of the newly created algorithm in reducing pressure and guaranteeing fair allocation of work, in comparison to the standard gate assignment techniques.

Performance metrics:

• When evaluating the effectiveness of the algorithm, many aspects are considered, such as gate utilization, congestion, and equitable workload allocation. Another method for assessing the program's flexibility and capacity to withstand changes includes examining real-life events.

Implementation and complexity analysis:

• The effectiveness of the program is usually evaluated using many indicators, including congestion levels, fairness in employment distribution, and gate utilization. After the program is put into action, its ability to adjust to new events and handle stress efficiently is assessed.

The suggested methodology employs a local antimagic vertex coloring technique to address the issue of gate assignment for aircraft at airports. The primary objective is to minimize the duration of delays, alleviate traffic congestion, and accommodate more aircraft that were previously not taken into account. A comprehensive investigation has been conducted to examine potential enhancements in airport efficiency and customer satisfaction. This may be ascribed to the use of rigorous methodology, theoretical frameworks, and empirical facts that are connected to the system that is being investigated.

3. Algorithm

Algorithm for Airport Flight-Gate Allocation using Local Antimagic Vertex Coloring

1. Initialize data structures

• Construct a graph G that adheres to realistic constraints. The vertices should be represented as gates, while the connections between them should be shown as adjacency links.

• Incorporate the colors of the gates into the recently instantiated dictionary.

• Developing a comprehensive compendium for documenting and allocating flight schedules.

2. Function

• The assignment of a weight to a vertex v in a graph is contingent upon its color.

• The vertex weight of a given vertex v is determined by the sum of the labels of all edges that are connected to v.

• a. Choose a vertex (Flight) that has not been assigned a color.

b. The process of assigning colors to vertices should be carried out in a way that minimizes the disparity in the number of edges connected to each vertex.

c. In order to accurately calculate the weights of vertices, it is essential to ensure that the weight values assigned to each pair of adjacent vertices, denoted as u and v, remain different. The weight of a vertex u is computed by summing the labels of the edges in its incident edge set, represented as $E(u)$.

3. The procedure involves the assignment of gate colors

• In the event that a vertex v is present inside the graph:

• To ascertain the color of a vertex, the "Calculate color" command may be used, where v denotes the vertex.

• Incorporate the designated color into the collection of gate colorations.

4. Gates assignment

• For each flight indicated in the flight schedule:

• Identify the gate vertex that corresponds to the specified gate for the flight.

• The assignment of the flight shall be allotted to the specifically selected gate, with its color being determined in accordance with the gate colors dictionary.

• The allocation of the flight task is to be kept inside the flight assignments dictionary.

5. Provide an updated flight schedule

• In the case of changes to the flight schedule, including the inclusion, alteration, or removal of flights:

• Provide a current iteration of the flying itinerary.

• Perform the Assign Flights to Gates procedure once again with the intention of revising the flight assignments.

6. Handle disruptions

• In instances of interruptions, such as gate closures and flight delays:

• Provide details on the flights and gates that have experienced disruptions.

• Modify the flight schedule and gate colors accordingly.

• It is recommended that the Assign Flights to Gates procedure be re-executed in order to implement any required modifications to the existing flight-gate assignments.

7. Output: The allotment of gate allocations for each particular aircraft

This overview offers a robust basis for comprehending the algorithm. Python, Java, and C++ are all exemplary instances of programming languages that may be used for this purpose, in conjunction with diverse data structures such as trees, dictionaries, and queues. In order to ensure the efficacy of the algorithm in practical situations, it is important to consider exceptional occurrences, error handling, and optimization techniques.

3.1 *Main results*

By implementing local antimagic vertex coloring to the Airport flight-gate allocation issue, the main outcome is a fair and efficient assignment of flights to gates. This method guarantees that no two neighboring flights (vertices) will be assigned the same gates (color). This approach applies separate labels to the edges of the graph that represents the airport layout. These weights ensure that the total of weights connected egdes to each vertex produces values. This unique allocation of weights allows for a conflict-free assignment of flights to gates. When using the intervals of the graph approach, flights are assigned to gates according to predetermined intervals that match the availability and needs of both flights and gates. This technique takes into account the temporal features of flight schedules, guaranteeing that flights are allocated to gates within their operating time frames without any overlaps or conflicts. The graph technique for intervals improves allocation by reducing the total waiting time for aircraft and maximizing gate usage. Both approaches contribute to a resilient flight-gate assignment system by assuring optimal usage of resources, minimizing delays, and ensuring seamless airport operations. The local antimagic vertex coloring ensures that gate assignments are both unique and conflict-free, while the graph interval approach optimizes the allocation of flights to gates in a way that maximizes efficiency.

4. Applications

4.1 *Application 1*

The study included a sample size of 10 flights, each of which was assigned a letter code from the set A, B, C, D, E, F, G, H, I, J.

When examining flight timings are given in (Table 1) without constructing a graph G, the following information emerges.

Table 1. Flights time durations

Flights	Time duration
A	01.00-03.00
B	09.00-12.00
C	01.00-06.00
D	14.00-24.00
E	04.00-15.00
F	02.00-08.00
G	08.00-17.00
H	18.00-23.00
I	06.00-09.30
J	19.00-22.00

Represent these time intervals (Table 1) within an interval graph to detect potential time conflicts among the flight schedules.

In this context, we should employ a local antimagic vertex coloring scheme for the graph in a manner that ensures that no two flights with time conflicts share the same color.

The edges, referred to as "flight radar signals", are crucial for ensuring efficient and conflict-free gate allocations. Each edge symbolizes a unique radar signal that corresponds to the specific moments when two airplanes cross paths.

In order to assign unique radar signals to the 20 edges in your airport flight-gate assignment using local antimagic vertex coloring, begin by treating each edge as an individual connection between two planes with overlapping schedules. Allocate a unique numerical identifier ranging from 1 to 20 to each edge, ensuring that no two edges share the same identifier. The labeling is achieved by the use of a bijection approach, where each edge is allocated a distinct label from the set 1, 2, ..., 20. The process starts by identifying all pairs of flights that intersect, which subsequently serve as the edges in the graph. Upon identification, systematically give a radar signal (label) to each edge in a manner that guarantees each edge is allocated a distinct number. This uniqueness is crucial as it ensures that when calculating the vertex weight (the sum of the labels of all edges connecting to a vertex), each flight will have a separate total. This particular sum corresponds to a unique gate assignment, guaranteeing that no two overlapping flights are allocated the same gate.

This issue can be resolved by assigning gates to flights through a scheduling process that ensures flights with overlapping time intervals do not experience time conflicts with one another.

We can assign distinct flight timings to different sections or tracks within the airport using the principles of local antimagic vertex coloring.

Flights which come on same time should assign different gates in terminals this avoid delays for other flights.

We can provide an effective gate allocation solution by representing flights as vertices and their respective times as edges in a set in Figure 1.

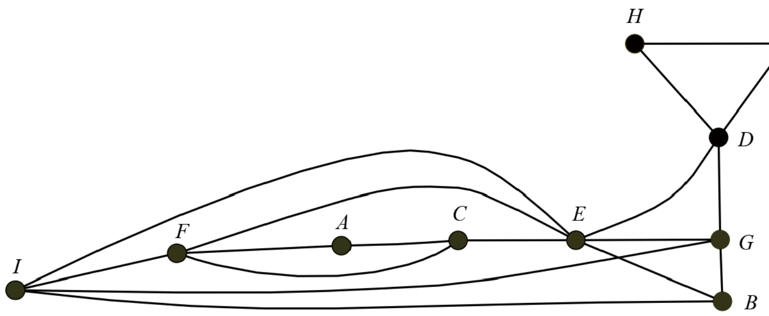

Figure 1. Graph G

The application of the local antimagic chromatic number concept allows us to determine the minimum number of gates required in an airport, resulting in cost savings and optimal utilization of airport space.

In this approach, we prioritize flight timings as the primary factor, using them to determine the allocation of flights to specific gates.

We do not employ a "first come, first serve" approach since flights have varying departure timings, necessitating adjustments and waiting periods for certain flights.

The graph representing the assigned flight gates, which we'll refer to as G, underwent a labeling process using the local antimagic vertex coloring technique to determine G's local antimagic chromatic number. As depicted in Figure 1, $\chi_{la}(G) \geq \chi(G) \geq 4$. However, as Figure 2 demonstrates, $\chi_{la}(G_1) \leq 4$, indicating that the local antimagic complete vertex coloration of the given graph G_1 indeed results in the utilization of 4 distinct colors, specifically $\chi_{la}(G_1) = 4$ [2].

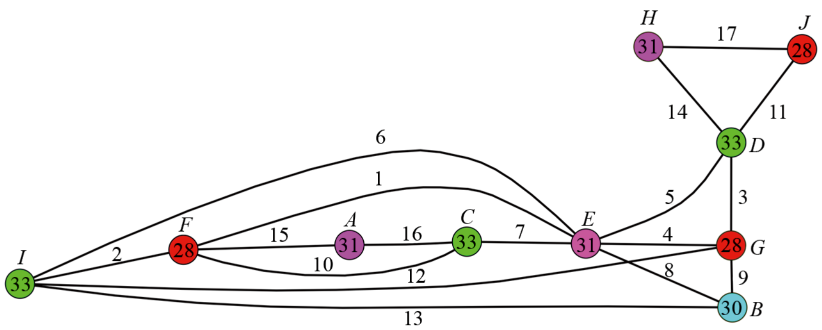

Figure 2. G_1-4 (gates) colors (Blue, Green, Pink, Red)

The outcome of the local antimagic vertex coloring applied to the presented graph is highly efficient. This result serves as the foundation for the creation of a gate allocation system for assigning flights to gates, as outlined in Table 1. Consequently, the optimal gate allocation system for accommodating 10 flights across 4 gates requires a total of 4 allocation times, as indicated in Table 2. Remarkably, this allocation duration aligns with the local antimagic chromatic number of the represented graph, denoted as G_1.

To model the assignment of flight gates for the ten flights, a graph was constructed. In this graph, each flight corresponds to a vertex, and whenever two flights share the same gate, an edge connects their respective vertices. The resulting graph, representing the flight gate allocation assignment, is visually depicted in Figure 1.

The chart displays the timings for a total of 10 flights (Figure 3).

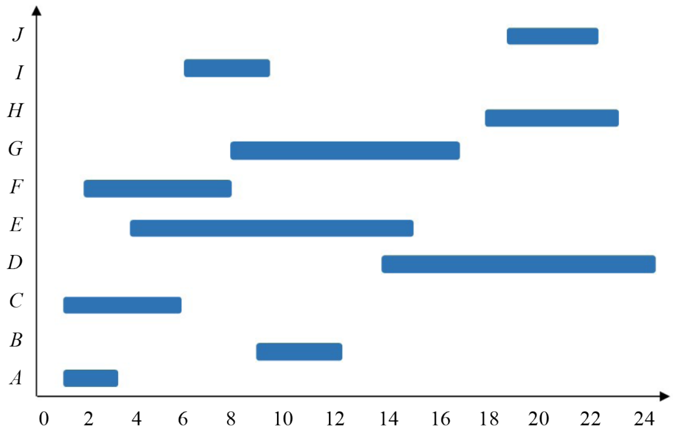

Figure 3. Time intervals in the chart

The green line on the chart indicates that there are three flights scheduled to land at the airport simultaneously (Figure 4).

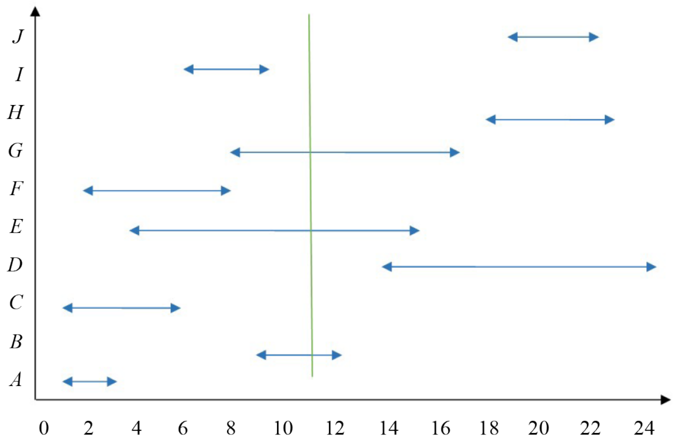

Figure 4. Time intervals relation

In line with the concepts of interval graph theory, it is necessary to design a graph in which the vertices and edges are suitably labeled, using the idea of local antimagic vertex coloring.

The color schemes shown in this illustration are the result of applying the local antimagic vertex coloring method to a specified group of vertices.

The allocation of gates using graph coloring yields the ensuing outcome (Table 2):

Table 2. Flights gate allocation using local antimagic vertex coloring

Gates	Weights (Colors)	Flight accomdation
1	Blue (30)	B
2	Green (33)	C, D, I
3	Pink (31)	A, E, H
4	Red (28)	F, G, J

If there is an unexpected arrival at the airport, it is possible to allocate an aircraft to any available gate. An analysis is conducted to evaluate the capacity of individual gates to handle arriving aircraft by examining the arrival times of the aircraft.

4.2 *Application 2*

When examining the flight times shown in (Table 3) without constructing a graph G_2, the following information is revealed.

Represent these time intervals (Table 3) within an interval graph to detect potential time conflicts among the flight schedules.

To address this situation, it is advisable to provide a local antimagic vertex coloring approach for the graph, guaranteeing that no two aircraft with conflicting schedules are assigned the same color.

Where the edges called as "flight radar signals", are essential for providing efficient and conflict-free gate assignments. Every edge represents a distinct radar signal that corresponds to the time periods when two aircraft intersect.

In order to assign unique radar signals to the 79 edges in your airport flight-gate assignment using local antimagic vertex coloring, begin by treating each edge as an individual connection between two planes with overlapping schedules. Allocate a unique numerical identifier ranging from 1 to 79 to each edge, ensuring that no two edges share the same identifier. The labeling is achieved by the use of a bijection approach, where each edge is allocated a distinct label from the set 1, 2, ..., 79. The process starts by identifying all pairs of flights that intersect, which subsequently serve as the edges in the graph. Upon identification, systematically give a radar signal (label) to each edge in a manner that guarantees each edge is allocated a distinct number. This uniqueness is crucial as it ensures that when calculating the vertex weight (the sum of the labels of all edges connecting to a vertex), each flight will have a separate total. This particular sum corresponds to a unique gate assignment, guaranteeing that no two overlapping flights are allocated the same gate.

To resolve this issue, gates can be assigned to flights using a scheduling process that guarantees flights with overlapping time intervals do not encounter time conflicts.

By applying the principles of local antimagic vertex coloring, we can allocate unique flight schedules to various sections or tracks within the airport.

In Table 3, Indhira Gandhi international airport flight arrival and departure data taken over by [28].

Table 3. Indhira Gandhi international airport flight arrival and departure list

No	Date	Flight No.	Dep	Arr	STD	STA	ATD	ATA
1.	2024/03/15	6E2279	BHO	DEL	2024-03-15 21:20	2024-03-15 22:40	2024-03-15 21:29	2024-03-15 22:45
2.	2024/03/15	UK813	DEL	BLR	2024-03-15 17:40	2024-03-15 20:20	2024-03-15 18:11	2024-03-15 20:48
3.	2024/03/15	6E2285	PNQ	DEL	2024-03-15 18:15	2024-03-15 20:30	2024-03-15 18:17	2024-03-15 20:15
4.	2024/03/15	6E615	BOM	DEL	2024-03-15 13:15	2024-03-15 15:25	2024-03-15 13:49	2024-03-15 15:32
5.	2024/03/15	AI770	CCU	DEL	2024-03-15 20:00	2024-03-15 22:35	2024-03-15 21:20	2024-03-15 23:30
6.	2024/03/15	6E2193	IXC	DEL	2024-03-15 18:45	2024-03-15 19:35	2024-03-15 19:12	2024-03-15 19:55
7.	2024/03/15	I5710	DEL	IXR	2024-03-15 11:00	2024-03-15 12:50	2024-03-15 11:16	2024-03-15 12:45
8.	2024/03/15	6E2482	DEL	PAT	2024-03-15 19:20	2024-03-15 20:45	2024-03-15 19:46	2024-03-15 21:00
9.	2024/03/15	6E64	JED	DEL	2024-03-15 03:00	2024-03-15 10:40	2024-03-15 03:01	2024-03-15 09:58
10.	2024/03/15	UK940	BOM	DEL	2024-03-15 19:45	2024-03-15 21:55	2024-03-15 20:07	2024-03-15 21:51
11.	2024/03/15	6E6762	IDR	DEL	2024-03-15 19:05	2024-03-15 20:30	2024-03-15 19:13	2024-03-15 20:34
12.	2024/03/15	AI503	BLR	DEL	2024-03-15 17:15	2024-03-15 20:10	2024-03-15 17:29	2024-03-15 20:01
13.	2024/03/15	6E6637	DEL	BDQ	2024-03-15 18:35	2024-03-15 20:10	2024-03-15 18:53	2024-03-15 20:09
14.	2024/03/15	IX1590	BLR	DEL	2024-03-15 05:10	2024-03-15 09:20	2024-03-15 05:30	2024-03-15 09:32
15.	2024/03/15	6E6016	DEL	BOM	2024-03-15 20:00	2024-03-15 22:20	2024-03-15 20:25	2024-03-15 22:23
16.	2024/03/15	6E6003	IXJ	DEL	2024-03-15 11:30	2024-03-15 14:30	2024-03-15 11:50	2024-03-15 14:34
17.	2024/03/15	6E2221	ATQ	DEL	2024-03-15 07:30	2024-03-15 08:35	2024-03-15 07:51	2024-03-15 08:42
18.	2024/03/15	6E698	MAA	DEL	2024-03-15 14:35	2024-03-15 17:20	2024-03-15 14:56	2024-03-15 17:17
19.	2024/03/15	UK910	BOM	DEL	2024-03-15 17:25	2024-03-15 19:45	2024-03-15 17:59	2024-03-15 19:54
20.	2024/03/15	AI635	IDR	DEL	2024-03-15 09:40	2024-03-15 11:20	2024-03-15 09:45	2024-03-15 10:46
21.	2024/03/15	6E369	GOX	DEL	2024-03-15 21:05	2024-03-15 23:35	2024-03-15 21:20	2024-03-15 23:36
22.	2024/03/15	6E2087	DEL	GOP	2024-03-15 14:15	2024-03-15 15:40	2024-03-15 14:58	2024-03-15 16:02
23.	2024/03/15	6E2769	DEL	PAT	2024-03-15 08:40	2024-03-15 10:10	2024-03-15 08:47	2024-03-15 10:03
24.	2024/03/15	6E379	HYD	DEL	2024-03-15 06:30	2024-03-15 08:50	2024-03-15 06:50	2024-03-15 08:32
25.	2024/03/15	AI677	BOM	DEL	2024-03-15 13:00	2024-03-15 15:15	2024-03-15 13:19	2024-03-15 15:02

STD-Scheduled Time of Departure; STA-Scheduled Time of Arrival; ATD-Actual Time of Departure; ATA-Actual Time of Arrival

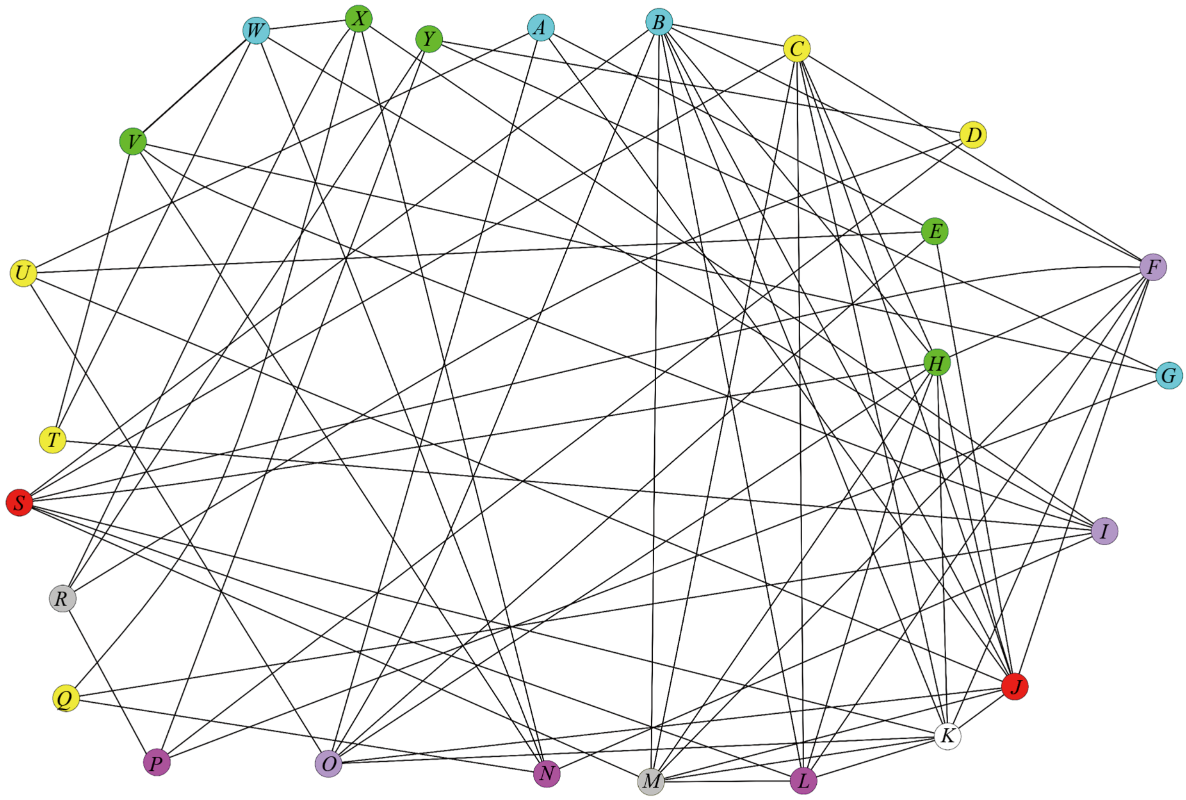

Figure 5. Graph G_2

Aircraft that arrive simultaneously should be assigned separate gates at terminals to prevent delays for other aircraft. In Figure 5, we may provide a proficient gate allocation solution by expressing flights as vertices and their corresponding times as edges in a set.

The application of the local antimagic chromatic number concept allows us to determine the minimum number of gates required in an airport, resulting in cost savings and optimal utilization of airport space.

In this approach, we prioritize flight timings as the primary factor, using them to determine the allocation of flights to specific gates.

Our approach to scheduling is not based on a "first come, first serve" basis due to the fact that flights have different departure times, which requires us to make adjustments and wait for certain flights.

To adhere to the principles of interval graphs, it is necessary to create a graph that has appropriately labeled the edges, using the concept of local antimagic vertex coloring.

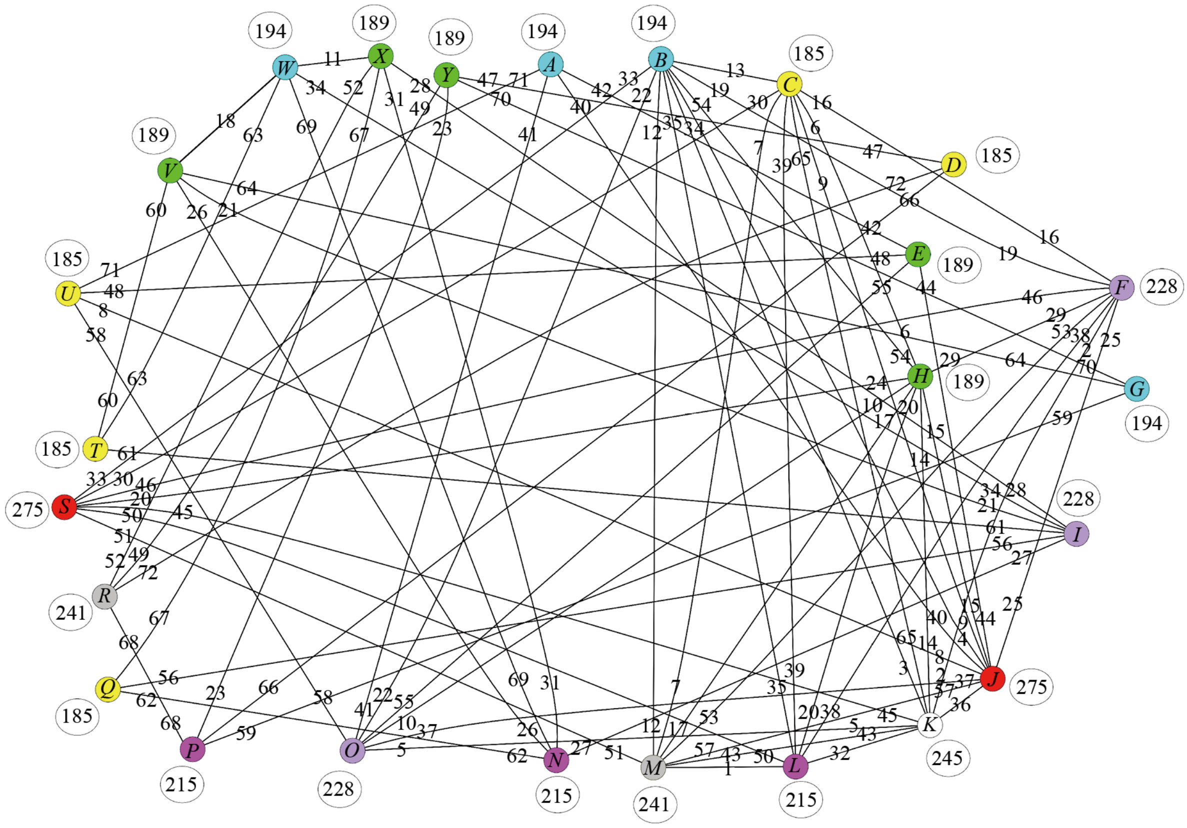

Figure 6. G_3-8 (gates) colors (Blue, Green, Grey, Red, Rose, violet, Yellow, White)

The color schemes shown in this illustration are the outcome of implementing the local antimagic vertex coloring method on a specific group of vertices.

The graph representing the assigned flight gates, denoted as G_2, underwent a labeling process using the local antimagic vertex coloring method to determine the local antimagic chromatic number of G_2. As shown in Figure 5, $\chi_{la}(G_2) \geq \chi(G_2) \geq 8$. However, as depicted in Figure 6, $\chi_{la}(G_3) \leq 8$, indicating that the local antimagic vertex coloring of the given graph G_3 indeed uses 8 distinct colors, specifically $\chi_{la}(G_3) = 8$.

The result of applying the local antimagic vertex coloring on the given graph is highly efficient. This outcome is the basis for developing a gate allocation system that assigns flights to gates, as described in (Table 4). Hence, the most efficient gate allocation system to accommodate 25 flights across 8 gates necessitates a total of 8 allocation intervals, as specified in (Table 4). The duration of this allocation coincides with the local antimagic chromatic number of the represented graph, denoted as G_3.

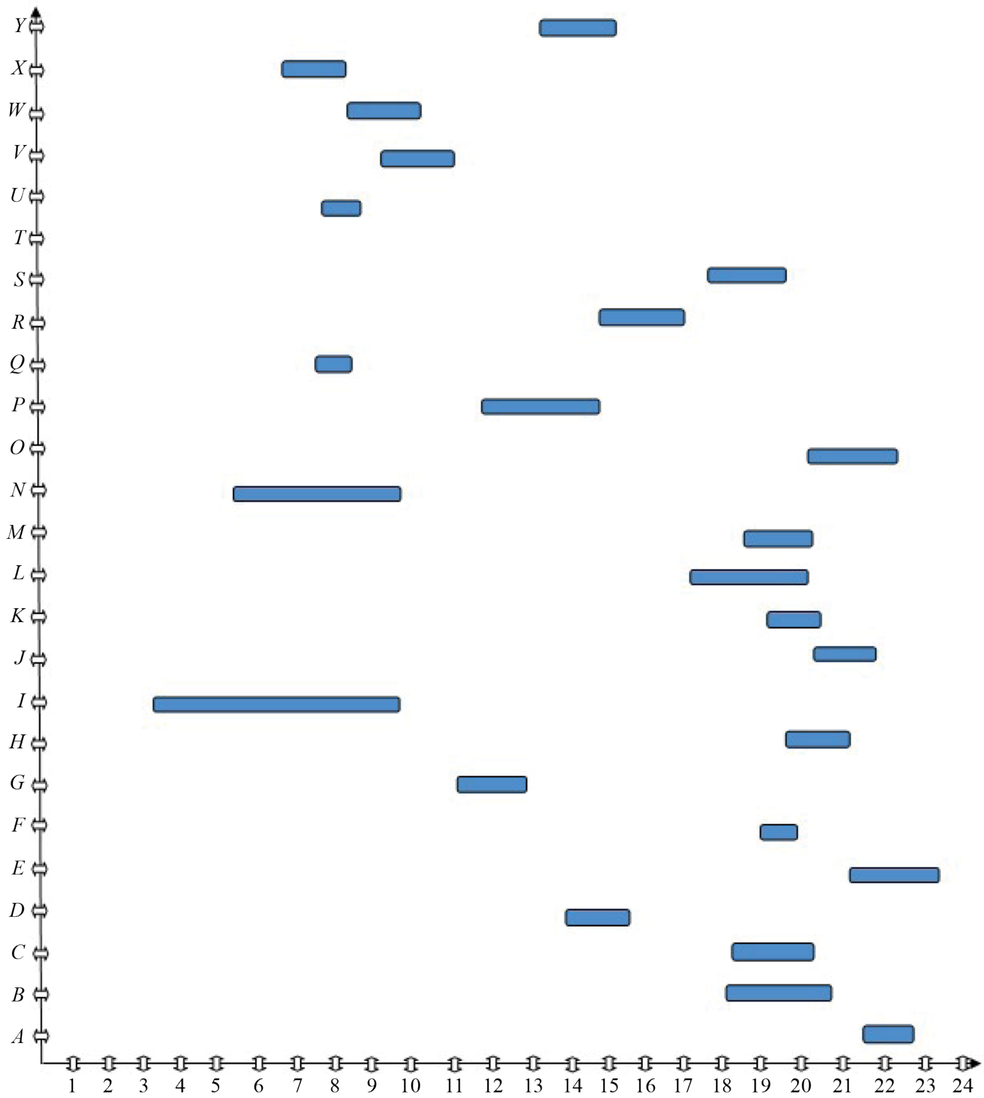

Figure 7. Flight arriving and departure time intervals in the chart

A graph was developed to represent the assignment of flight gates for the twenty five flights. The graph represents flights as vertices, and an edge is created between two vertices whenever the corresponding flights share the same gate. The graph illustrating the flight gate allocation assignment is graphically shown in Figure 5.

The chart shows the time schedules for a grand total of 25 flights (Figure 7).

The green line on the chart indicates that flights scheduled to landing to departure timing at the airport (Figure 8).

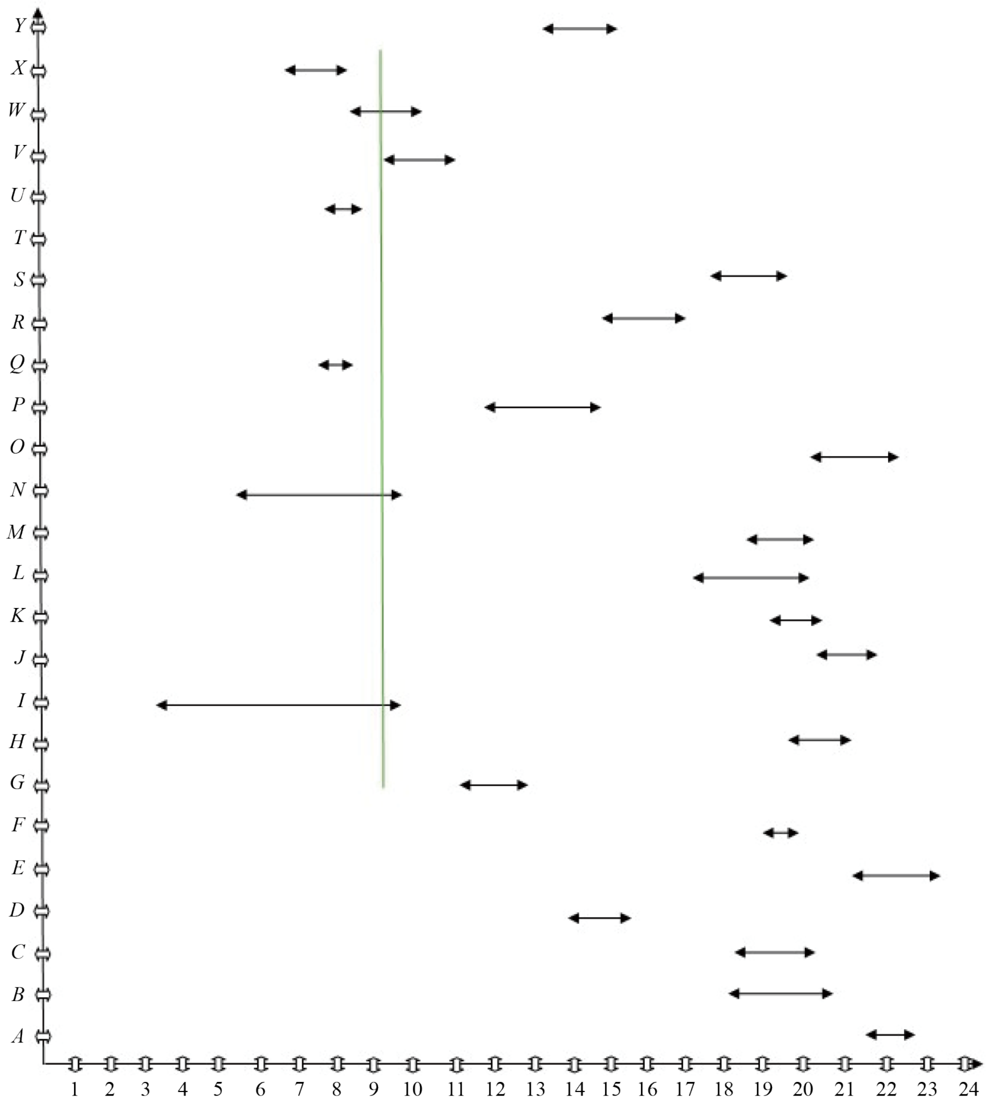

Figure 8. Time intervals relation

To line to the concepts of interval graph, it is necessary to create a graph that has appropriately labeled edges, using the concept of local antimagic vertex coloring.

The color schemes shown in this example are the outcome of implementing the local antimagic vertex coloring technique on a designated set of vertices (Figure 6).

The process of assigning gates using graph coloring produces the following outcome (Table 5).

Table 4. Degrees of each vertices

No.	Vertex	Degree	Edge connections
1.	A (6E2279)	3	E, O, U
2.	B (UK813)	9	$C, F, H, J, K, L, M, O, S$
3.	C (6E2285)	8	B, F, H, J, K, L, M, S
4.	D (6E615)	3	P, R, Y
5.	E (AI770)	4	A, J, O, U
6.	F (6E2193)	7	B, C, H, K, L, M, S
7.	G (I5710)	3	P, V, Y
8.	H (6E2482)	9	$B, C, F, J, K, L, M, O, S$
9.	I (6E64)	6	N, Q, T, V, W, X
10.	J (UK940)	10	$A, B, C, E, F, H, K, M, O, U$
11.	K (6E6762)	9	$B, C, F, H, J, L, M, O, S$
12.	L (AI503)	7	B, C, F, H, K, M, S
13.	M (6E6637)	7	B, C, F, H, K, L, S
14.	N (IX1590)	4	I, Q, V, X
15.	O (6E6016)	7	A, B, E, H, J, K, U
16.	P (6E6003)	3	D, G, Y
17.	Q (6E2221)	3	I, N, X
18.	R (6E698)	3	D, P, X
19.	S (UK910)	7	B, C, F, H, K, L, M
20.	T (AI635)	3	I, V, W
21.	U (6E369)	4	A, E, J, O
22.	V (6E2087)	4	I, N, T, W
23.	W (6E2769)	3	I, N, X
24.	X (6E379)	3	I, N, Q
25.	Y (AI677)	3	D, P, R

Table 5. Flights gate allocation using local antimagic vertex coloring

Gates	Weights (Colors)	Flight accomdation
1	Blue (194)	A, B, G, W
2	Green (189)	E, H, V, X, Y
3	Grey (245)	M, R
4	Red (275)	J, S
5	Rose (215)	L, N, P
6	Violet (228)	F, I, O
7	Yellow (185)	F, I, O
8	White (241)	K

In the case of an unexpected arrival at the airport, it is feasible to assign an aircraft to any unoccupied gate. The analysis of aircraft arrival times at individual gates is performed to assess the gates capacity for handling arriving aircrafts.

5. Conclusion

One such approach involves the use of a method referred to as local antimagic vertex coloring in order to streamline the assignment of airport gates for aircraft. This technique enables more flexibility in the scheduling of flights, a more equitable allocation of passengers among gates, and a comprehensive reduction of airport congestion. This unique technique has the capability to adjust airport operations and improve resource allocation within the context of a dynamic and constantly developing airway transport industry. This aim may be attained via the optimization of operational efficiency and the enhancement of the customer experience.

6. Open problem

An unresolved challenge is the development of algorithms that can dynamically reassign planes to gates in real-time, taking into account unexpected events like as delays and cancellations. This task requires the development of effective strategies that combine local antimagic vertex coloring and interval graph methods to provide conflict-free and optimum gate use. These techniques must also handle uncertainties and retain computational efficiency. The objective is to achieve a harmonious functioning of airports with minimum interruptions by effectively managing dynamic updates, unpredictability, scalability, and other operational limitations.

Acknowledgement

The first author is thankful to Vellore Institute of Technology, Vellore, for providing a Teaching Cum Research Assistant Fellowship. The authors are thankful to the reviewers for their helpful suggestions leading to substantial improvements in the presentation of the paper.

Conflict of interest

The authors declare no competing financial interest.

References

[1] Arumugam S, Premalatha K, Bacă M, Semaničová-Fecňovčíková A. Local antimagic vertex coloring of a graph. *Graphs and Combinatorics*. 2017; 33: 275-285.

[2] Utami W, Wijaya K, Slamin S. Application of the local antimagic total labeling of graphs to optimize scheduling system for an expatriate assignment. *Journal of Physics Conference Series*. 2020; 1538: 13-20.

[3] Li H, Ding X, Lin J, Zhou J. Study on coloring method of airport flight-gate allocation problem. *Journal of Mathematics in Industry*. 2019; 9: 1-16.

[4] Babic O, Teodorovic D, Tosic V. Air craft stand assignment to minimize walking. *Journal of Transportation Engineering*. 1984; 110(1): 55-66.

[5] Mangoubi RS, Mathaisel DFX. Optimizing gate assignments at airport terminals. *Transportation Science*. 1985; 19(2): 173-188.

[6] Yan S, Huo C. Optimization of multiple objective gate assignments. *Transportation Research Part A: Policy and Practice*. 2001; 35(5): 413-432.

[7] Wen J, Li B, Wang Q, Wen D. Graph coloring model and algorithm for airport parking space allocation. *Systems Engineering-Theory Methodology Application*. 2005; 14(2): 136-140.
[8] Tian C, Xiong G. Airport gate position scheduling strategy based on genetic algorithm. *Computer Engineering*. 2005; 31(3): 186-188.
[9] Ju S, Xu L. Research on optimization problem of gate position assignment based on GSAA. *Journal of Transportation Systems Engineering and Information Technology*. 2008; 8(1): 138-143.
[10] Jiang Y. *Research on Optimization of Airport Parking Space Allocation*. Tianjin: Tianjin University; 2010.
[11] Wen J. Genetic algorithm for airport gate position scheduling. *Science Technology and Engineering*. 2010; 10(1): 135-139.
[12] Zeng L, Jiang Y, Luo Y. Modeling of optimal assignment of gate position based on walking distance of passengers. *Journal of Wuhan University of Technology*. 2014; 38(4): 895-899.
[13] Liu S, Chen W, Liu J. Robust assignment of airport gates with operational safety constraints. *International Journal of Automation and Computing*. 2016; 13(1): 31-41.
[14] Zhou F, Jiang G, Lu Z, Wang Q. Evaluation and analysis of the impact of airport delays. *Scientific Programming*. 2022; 2022: 7102267. Available from: https://doi.org/10.1155/2022/7102267.
[15] Ruan L, Zheng X. Interval multi-objective decision making method based on entropy weight. *Statistics and Decision Making*. 2013; 12: 82-84.
[16] Zhang H, Zhao J, Luo H, Xie X. Objective weight calculation method for multi-objective optimization based on personal preference. *Journal of Control and Decision*. 2014; 29(8): 1471-1476.
[17] Zheng H, Li N, Yang X. Load balancing algorithm for adaptive load index weight. *Computer Engineering and Design*. 2019; 40(3): 623-643.
[18] Ganguli R, Roy S. A study on course timetable scheduling using graph coloring approach. *International Journal of Computational and Applied Mathematics*. 2017; 12(2): 469-485.
[19] Luo R, Xie R, Zhang D. Vertex shader model and algorithm for parking station allocation. *Theory and Practice of System Engineering*. 2007; 5(11): 148-152.
[20] Qin H, Liao Z. Multi-target athlete tracking method based on the characteristics of competition environment. *Computer Engineering and Design*. 2017; 11(38): 3173-3178.
[21] Li Q, Wang X. Multi-stage and multi-type equipment procurement decision-making model and solving algorithm. *Computer Engineering and Design*. 2016; 37(11): 3027-3034.
[22] Borskya S, Unterberger C. Bad weather and flight delays: The impact of sudden and slow onset weather events. *Economics of Transportation*. 2019; 18: 10-26.
[23] Efthymiou E, Njoya ET, Lo PL, Papatheodorou A, Randall D. The impact of delays on customers' satisfaction: an empirical analysis of the British airways on-time performance at Heathrow Airport. *Journal of Aerospace Technology and Management*. 2019; v11: e0219. Available from: https://doi.org/10.5028/jatm.v11.977.
[24] Kong J. *Research on Modeling of Airport Gate Position Scheduling and Genetic Algorithm*. 2008.
[25] Li N. *Simulation and Optimization of Airport Gate Allocation*. Nanjing: Nanjing University of Aeronautics and Astronautics; 2013.
[26] Wang Y. *Research on the Mixed Set Planning Method for Airport Gate Position Scheduling in Busy Airports*. Nanjing: Nanjing University of Aeronautics and Astronautics; 2015.
[27] Bouras A, Ghaleb MA, Suryahatmaja US, Salem AM. The airport gate assignment problem. *The Scientific World Journal*. 2014; 2014(1): 923859. Available from: http://dx.doi.org/10.1155/2014/923859.
[28] Sri Vari Flight. *Flight schedule data for Indira Gandhi International Airport: May 2, 2024*. 2024. Available from: https://www.srivariflight.com [Accessed 28th November 2024].