# Aircraft Recovery Problem: Implementation of a Time Band Model Using the Python-MIP Library.

João Victor Mendes Rocha.
Universidade Federal de Santa Catarina.

joaorochabr@gmail.com

**PAPER ID: SIT2189**

**ABSTRACT**

This study highlights the impact on costs generated by flight delays and cancellations, which pose a challenge for airlines in the country's economic context. The Aircraft Recovery Problem (ARP) involves the recovery of schedules, aircraft, crews, and passengers. In this context, the objective of this study was to develop a flight planning model aimed at minimizing the costs resulting from aircraft delays and cancellations. To achieve this goal, a systematic literature review was conducted on the ARP, identifying relevant solution methods and adopting a network representation model. The innovative use of the Python-MIP library was employed to implement an algorithm capable of solving the aircraft recovery problem in a generalized manner, covering various scenarios considered by other researchers. The obtained results are illustrated for three different scenarios, taking into account the cancellation cost and passenger load on each flight. The comparison of the results with the study by Arguello et al. (1998) revealed a percentage difference. The discussion emphasizes the importance of considering the computational complexity and limitations of approximation models when dealing with real problems, recognizing that the results are only approximations of reality. The proposed algorithm allows for a generalized and efficient resolution of the ARP.

**Keywords**: Aircraft Recovery Problem. Optimization. Scheduling.

.

## 2. BRIEF INTRODUCTION

The airline recovery problem, known as the Aircraft Recovery Problem (ARP), is complex and involves sequential stages of schedule recovery, aircraft, crew, and passenger recovery. One of the steps involves determining flight departure and cancellation times, as well as revising routes affected by the aircraft.

Recovery problems are typically formulated for cost minimization rather than profit maximization models since airline revenues are fixed, as tickets are generally sold in advance (VINK, J. et al., 2021). VINK (2021) categorizes ARP solution methods by the type of methodology used: exact optimization methods, metaheuristic methods, hybrid methods, multi-agent systems, among others.

It is important to resolve recovery-related problems as quickly as possible because after a certain time, solutions may become unviable due to time constraints and their impact on subsequent stages. The first step in the recovery plan is to recover the aircraft, the second stage is crew recovery, and finally, the re-accommodation of passengers affected by disruptions (BELOBABA et al., 2009, p. 259).

The costs of delays have a direct impact on the financial results of airlines, harming their profits. In the first quarter of 2019, the Brazilian airline Gol recorded costs of services provided totaling R$ -2,083,594,685. Additionally, the additional costs associated with delays were approximately R$ -12,844,198.07, representing 1% of the service costs. In the last quarter of 2019, TAM recorded costs of services provided amounting to R$ -14,183,903,843. As for the costs related to delays, they were specified as R$ -1,088,687.18.

There are various solution methods for the ARP, including exact optimization methods, metaheuristics, hybrids, and multi-agent systems. It is crucial to solve recovery problems as quickly as possible since delayed solutions can become infeasible and impact subsequent stages. The objective was Developing a new flight planning using the Python programming language and the MIP library to minimize the costs generated by aircraft delays and cancellations.

## 3. PROBLEM DEFINITION

This study is grounded in the methodology established by Arguello, Bard, and Yu (1998), which is designed to minimize costs associated with the repositioning of irregular aircraft. Irregular aircraft pertains to those unable to adhere to the originally scheduled flight plan due to mechanical or meteorological factors. Arguello, Bard, and Yu (1998) devised a time band model to address this issue. The primary objective of this method is to formulate a revised aircraft allocation plan that converges with the original schedule while accounting for incurred delays, all while striving to achieve this reallocation at the lowest possible cost by the end of the day.

To facilitate a comprehensive grasp of the problem, we will commence with a general definition of Aircraft Repositioning Problem (ARP) and subsequently delve into an in-depth elucidation of the network model pioneered by Arguello, Bard, and Yu (1998). The input data essential for this problem encompasses a flight schedule serving as the foundational framework, costs linked to flight delays and cancellations, aircraft assignments, expenses incurred for aircraft replacement, and a roster of available aircraft. The entirety of the flight network is considered in the quest for viable solutions. The desired outcome involves the generation of a novel, cost-optimized scheme for reallocating aircraft to their original schedules.

A prevalent airline policy involves upholding the original schedule for the subsequent day by imposing an aircraft balance constraint. This constraint necessitates the fulfillment of aircraft reassignment following an operational disruption. The time horizon for contemplating aircraft reassignment aligns with the conclusion of the current day.

An optimal solution is one that reinstates the airline's initial schedule, thereby ensuring the ability to fulfill all initially planned flights for the ensuing day.

When an aircraft becomes inoperative, the locations of all aircraft and their designated

scheduled flights are known. It is assumed that there is a limited number of spare aircraft, unallocated and available for use. These aircraft can serve as substitutes for other inoperative and/or designated aircraft as necessary.

The flight schedule and assigned routes determine the expected aircraft locations at the end of the day to fulfill the scheduled flights for the following day. Aircraft balancing is a fundamental process in airport management, involving the assurance that the number of aircraft arriving and departing from a station is balanced and aligns with the values previously programmed by the airlines.

This is crucial to ensure airport operations' efficiency and to prevent delays and overloads at specific times or locations. Balancing can be conducted at different levels, such as at a single station or throughout the airport system, considering factors such as aircraft characteristics, runway and terminal capacity, and air traffic demands.

Other constraints may limit aircraft reassignment. The first of these is the minimum return time, which is the minimum time interval an aircraft must wait after landing (arrival) before it can take off again (departure). This minimum time constraint can be applied system-wide or to specific fleets or stations.

The second constraint pertains to the airline's flexibility in permitting substitutions between fleets. In certain circumstances, such substitutions can prove advantageous.

Station requirements are factored into aircraft allocation planning. These requirements encompass operating hours and closing times, which can be either general or specific to each fleet type.

Operating hours determine permissible departure times from a station, while closing times indicate when departures are not allowed (curfew). An aircraft route is considered feasible if all departures on the route can be completed before the respective station's closing time.

Another critical viability condition for aircraft routing is route connectivity, which necessitates that the destination of the first flight on a route aligns with the origin of the subsequent flight. Consequently, an aircraft routing will only be deemed viable if it adheres to correct connectivity, adheres to minimum return times, and does not violate the curfew of the corresponding station.

A solution (routing) to the Aircraft Repositioning Problem (ARP) comprises a set of aircraft routes, with one designated for each available aircraft. To attain viability, each routing must satisfy the aforementioned viability conditions. Furthermore, each scheduled flight must be allocated to a maximum of one route, and any unassigned flight is treated as canceled. Lastly, the solution must ensure aircraft balance. By the day's end, the aircraft allocated to a terminal station must indeed be present at the specified station after the implementation of the new route, thereby ensuring that all stations possess an equal number of aircraft at the conclusion of the planning period.

To fulfill the objective of discovering the most cost-effective solution for the prevailing problem, cost functions must be provided for each alternative considered for each flight. This implies that, for each flight, cost functions for delay, cancellation, and aircraft replacement must be supplied.

Drawing from the costs associated with each potential response to aircraft delays, it becomes feasible to evaluate aircraft reassignment for regular flights. Implicit within each aircraft route are the departure and arrival times that the designated aircraft can adhere to for each flight on the route.

These times govern the potential delay costs linked to the route, along with any replacement costs if different aircraft were originally assigned to certain flights. Flights that do not align with the routes assigned to aircraft are categorized as canceled, and the ensuing costs are incorporated into the total cost of each assigned route. These costs play a pivotal role in determining the cost-effectiveness of feasible solutions for the problem under consideration, with the overarching aim of identifying a lower-cost solution.

## 4. TIME BAND MODEL

A time band network is constructed based on the assumption that the event horizon is partitioned into fixed-size time intervals, and the following input data is provided:

1. Flight data, which includes flight identification, origin, destination, scheduled departure and arrival times, and the function used to calculate delay costs.
2. A list of available aircraft, including their time and location.
3. A list of station data, specifying operating hours.

With these foundational assumptions in place, it becomes possible to transform the Aircraft Repositioning Problem (ARP) into a time band network.

To implement the algorithm proposed by Arguello, Bard, and Yu (1998), the following information is required:

- A list of nodes ($L$).
- A minimum turnover time in minutes (MTT).
- A flight delay function $k$ per unit of time ($DLC(t)$).
- A cost function for the delay of flight $k$ from node $i$ to node $j$ ($d(k,i,j)$).

The initialization step primarily involves sorting the station nodes. Since there are at most $r$ of these nodes (where $r$ is the number of available aircraft), this sorting process can be completed in $O(r\ log\ r)$ time complexity.

The transformation process utilizes at most $t \times r$ nodes and $(t-1)\times(n+m)$ arcs. Consequently, the effort required for transformation is polynomial in relation to the number of flights, stations, and time intervals.

The resulting network comprises two distinct node types: station nodes and terminal nodes. Each station node is connected to other station nodes through outgoing arcs, representing the scheduled flights departing from the departure station. The number of these flight arcs for each station node corresponds to the total number of scheduled flights departing from the station during the considered time horizon.

Each station node has an equal number of outgoing arcs leading to specific stations, signifying the availability of each flight for each time segment during which the aircraft is available. Station nodes may lack directed incoming arcs, indicating that the available aircraft are already on the ground, while nodes with these arcs represent the available aircraft due to flight arrivals.

Each station node has a directed arc leading to its respective terminal node. Unlike station nodes, terminal nodes do not need to represent a single time segment but encompass all time segments in which aircraft can land at a station without being able to take off due to airport operational restrictions. Consequently, terminal nodes do not feature outgoing arcs. The terminal node serves as the endpoint for both aircraft in flight and aircraft available during a period before the event horizon with no flight allocation.

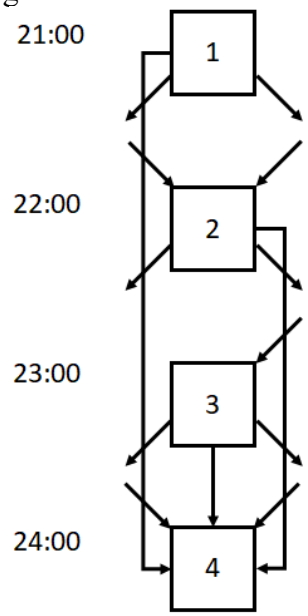



**Figure 1 Time Band Network for a Station.**

The specified time interval is set at one hour, with nodes 1, 2, and 3 representing stations that are separated by one-hour intervals. Node 4 is a terminal node. Here's a breakdown of the nodes and their significance:

Node 1 signifies that one or more aircraft are already positioned at the station and will become available at 21:00 hours. At 22:00 hours, the two outgoing arcs from node 1 indicate that two flights have been scheduled to depart from station 1 at 21:00 hours. The arc from node 1 connecting to node 4 indicates that the available aircraft have no scheduled departure and will remain at the station.

Nodes 2 and 3 each have two and one incoming arcs, respectively, indicating the possibility of two flights arriving between 22:00 and 23:00 hours and one flight arriving between 23:00 and 00:00 hours. Since the airport closes at 00:00 hours, it is expected that the flight from node 3 will arrive before 23:30. Otherwise, no departure arcs can originate from node 3, and nodes 3 and 4 could be merged into a single terminal node.

Similar to node 1, nodes 2 and 3 also have arcs connecting them directly to terminal node 4, indicating that aircraft available after 22:00 hours can terminate at the station. Flight arcs to node 4 indicate that aircraft serving these flights terminate at node 4. It's important to note that a flight can land at any time after 00:00 hours; however, to take off, the flight needs to be ready before 00:00 hours.

The terminal node aggregates all activities after the departure time, encompassing both flight arrivals after the airport's operational hours and serving as a collector for all aircraft remaining at the corresponding station beyond the operating hours.

The process of transforming the time-flow network employs the first arrival convention to determine the feasibility of an aircraft at a station. By marking the station with the earliest available time, it becomes possible to ascertain the first point at which aircraft will be available within the time band. From that point onwards, departure times are determined for all flights outside the time segment, along with the associated delay for each flight. Thus, it is assumed that all aircraft can commence servicing a flight at the same time as the earliest available aircraft within the corresponding time band.

A mathematical representation can be constructed from a time band network. This model can also prove valuable in obtaining precise lower bounds when delay costs are approximated due to extended time bands.

The following are the indices, sets, and variable parameters used in the formulation of the Time Band Model for the Aircraft Repositioning Problem (ARP):

Indices and Sets:

- $i, j$: Node indices.
- $k$: Flight index.

Sets:

- $F$: Set of flights.
- $G(i)$: Set of flights originating from station-flow node $i$.
- $H(k, i)$: Set of destination nodes for flight $k$ originating from station-flow node $i$.
- $I$: Set of station-flow nodes.
- $J$: Set of station-terminal nodes.
- $L(i)$: Set of flights terminating at node $i$.
- $M(k, i)$: Set originating from station-flow node for flights $k$ terminating at node $i$.
- $P(k)$: Set of station-flow nodes from which flight $k$ originates.
- $Q(k)$: Set of station-flow nodes within the station containing a terminal station node $i$.

Parameters:

- $a_i$: Number of aircraft available at node $i$.
- $c_k$: Cost of flight $k$ cancellation.
- $d_{ij}^k$: Cost of flight $k$ delay from station-flow node $i$ to node $j$.
- $h_i$: Number of aircraft required at the conclusion of the event horizon at terminal station $i$.

Variables:

- •$x_{ij}^k$: Aircraft flow for flight $k$ departing from station $i$ to node $j$.
- $y_k$: Flight cancellation indicator.
- $z_i$: Aircraft flow from station node $i$ to terminal node within the same station.

The variables $x_{ij}^k$ and $z_i$ indicate the utilization of aircraft resources along the network arcs. Variable $z_i$ signifies the number of aircraft available in a specific network segment that have not been allocated to flights for the remainder of the day. This is analogous to aircraft remaining available at node $i$ and proceeding directly to a corresponding terminal node.

On the other hand, variable $x_{ij}^k$ indicates the assignment of an available aircraft within time segment $i$ for flight $k$. This is analogous to an aircraft flying from station node $i$ on flight $k$ with destination node $j$.

The $y_k$ variables are auxiliary artificial variables, as they assist in capturing the cancellation cost for flights to which aircraft cannot be efficiently assigned in terms of costs.

The mathematical modeling employed here represents a network flow model but incorporates flight cancellation indicators, represented by variable $y_k$, following the original problem definition.

The Time Band Model for ARP is then defined by Equations (1)-(7).

Arguello et al. (1998) primarily adapted the model directly from the time flow network, constructing the sets based on node and arc incidences. The parameter $d_{ij}^k$ is the only one that needs to be computed explicitly, while the other parameters are directly inputted for the problem.

The objective function defined in Equation (1) minimizes the sum of total delay costs for assigned flights and total cancellation costs for unassigned flights.

$$Minimize \sum_{k\in F}\sum_{i\in P(k)}\sum_{j\in H(k,i)} d_{ij}^k\, x_{ij}^k + \sum_{k\in F} c_k y_k \quad (1)$$

Sujeito à:

$$\sum_{i\in P(k)}\sum_{i\in H(k,i)} x_{ij}^k + y_k = 1, \quad \forall k \in F \quad (2)$$

$$\sum_{k\in G(i)}\sum_{j\in H(k,i)} x_{ij}^k + z_i - \sum_{k\in L(i)}\sum_{j\in M(k,i)} x_{ji}^k = a_i, \quad \forall i \in I \quad (3)$$

$$\sum_{k\in L(i)}\sum_{j\in M(k,i)} x_{ij}^k + \sum_{j\in Q(i)} z_j = h_i, \quad \forall i \in J \quad (4)$$

$$x_{ij}^k \in \{0,1\}, \quad \forall k \in F, i \in I, j \in H(k,i) \quad (5)$$

$$y_k \in \{0,1\}, \quad \forall k \in F \quad (6)$$

$$z_i \in Z_+, \quad \forall i \in I \quad (7)$$

Equations (3) and (4) ensure the conservation of restricted flow in the network. The first equation implies aircraft flow along flight arcs and, after subtracting the flow to the time-station node, must be equal to the number of aircraft already present in the station but available during the associated time segment. The constraint (4) enforces aircraft balance by requiring that the aircraft flow at each terminal station node is equal to the number of aircraft to be terminated at the corresponding station.

Constraints (5) and (1) mandate that the aircraft flow along flight arcs and the cancellation indicator are binary. Constraint (7) requires that the $z_i$ variables be non-negative integers. It is possible that in some cases, these $z_i$ variables may assume a value greater than 1 if more than one aircraft becomes available at a specific station node, and no flight assignments are made for these aircraft.

Although it may seem intuitive and arbitrary, this model constitutes a challenging integer programming problem. If cancellation and flight coverage restriction variables are removed from the formulation, a minimum cost flow problem of a single product with an upper limit remains.

## 5. IMPLEMENTATION OF MIP-ARP ALGORITHMIME.

To address the Aircraft Repositioning Problem (ARP), was developed a Python algorithm using the integrated development environments (IDE) Anaconda and Spyder, versions 5.2.2.

The implemented algorithm leverages the MIP library introduced in Section 2.3. This library provides specialized tools for creating programs aimed at solving optimization problems with objective functions and linear constraints. It enables the consideration of continuous and discrete variables, including binary variables, within Python.

Furthermore, we employed a graph creation algorithm, as detailed in Appendix A, to generate a graph that provides the necessary data input for the optimization model. This encompasses defining the indices and sets used in the Model time band ARP, as specified in

Equations (1)-(7). The nodes in the time band network can be categorized into two types:

1. Flow nodes: These represent origins or flight passage points.
2. Terminal nodes: These indicate the locations where aircraft will conclude their actions within the planning horizon.

In the subsequent sections, we present the generated graph to represent the foundational problem, elaborate on the sets and parameters to enhance the understanding of the optimization model's formulation using the MIP library, and explain the data format utilized in constraints and objective functions.

Figure 2 depicts the time band approximation network created for the example problem. Each station's time node represents thirty-minute intervals starting from the associated time. For instance, node 1 represents the time interval from 14:00 to 14:29. Within this network, nodes 5, 11, 19, and 24 correspond to destination stations (terminal nodes), while all other nodes represent nodes within the flow network.

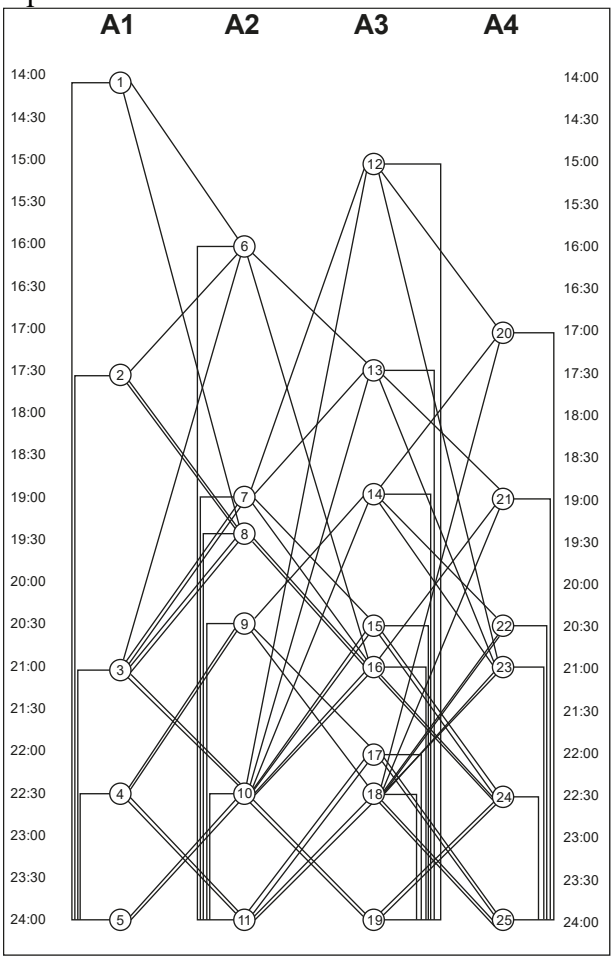



**Figure 2 Graph: Representation of possible flights.**

Accurate definition of the utilized sets is crucial for the algorithm to yield a feasible solution for the Model time band (1)-(7). The elements within these sets determine the index variations in the repetition loops employed in the algorithm and possess a strong correlation with routing possibilities.

For instance, in the network illustrated in Figure 1, flight "flight11" is represented by pairs of nodes [1,6], [1,8], [1,5], [2,8], [3,10], and [4,11]. The arc between nodes [1,5] is only used if "flight11" is canceled, emphasizing that the flight is not associated with a single arc but rather with a set of nodes representing its origin and destination.

Hence, the algorithm developed here commences by defining the sets and parameters used in the formulation of the time band Model (1)-(7):

• $I$: the set of station-flow nodes. • $J$: the set of station-terminal nodes, which, in the graph, are represented by nodes 5, 11, 19, and 25. • $F$: the set of flights. • $F$: the set of flights, with each flight identified by its original itinerary and identifier. If a flight experiences a delay, a different arc will be used, but the original identifier is retained.

The remaining sets are defined using dictionaries and tuples, which are data structures in Python that allow for storing sets of elements, akin to lists. The primary difference is that tuples are immutable, meaning their elements cannot be altered after the tuple is created.

The set $G$, considered in the formulation of the Model time band (6)-(12), is defined as a dictionary that relates to a list of flights. This relation indicates the possible origin node for flights in the set $F$ that are feasible at the indicated node. For example, node 1 can be the origin of 'flight11,' while node 2 can be the origin of 'flight11' and 'flight22.' Similarly, node 7 can be the origin of 'flight12,' 'flight14,' 'flight21,' and 'flight23.'

Set $H$ is defined as a dictionary employing tuples of two elements, representing a feasible flight and a departing node, indicating possible destination nodes for this pair. Some tuples may have multiple destinations departing from the same flight. Due to the nature of the time flow network, the

two mentioned destinations refer to the next aircraft activity and the time it will occur. For instance, from node 1, the arcs can lead to nodes 6 and 8. Arcs connecting the same airport are only allowed between flow nodes and terminal nodes.

Set $L$ is defined as a dictionary that associates all nodes with their arrival possibilities for a flight. For example, node 1 will be empty since there are only outgoing arcs and no arrivals at the node, whereas node 8 will have three possibilities...

Set $M$ is a dictionary of tuples relating a flight to an arrival node, returning the possible origin node for the flight. Set $M$ is similar to set $H$ but employs the inverse view for arcs and nodes.

Set $P$ is a dictionary representing all the origin possibilities for a flight $k$, with each origin except for the original itinerary being empty.

Last set $Q$ is a dictionary relating flights to nodes within the same airport that have an arc to the terminal node of the same airport.

Furthermore, it is worth noting that there was a necessity to create a set of artificial flights that indicate movement between nodes within the same airport. These fictitious flights allow a flight to enter one node and exit the airport through another node. The fictitious flights are denoted as "flightA1," "flightA2," "flightA3," and "flightA4."

## 6. RESULTS

The program was executed for the three scenarios employed by Arguello et al. (1998). The instance data used to illustrate the results are consistent with those considered by Arguello et al. (1998), as depicted in Table 1.

Arguello et al. presented a schedule for this example, wherein the cancellation cost used is a function of the number of passengers on each flight. The passenger load for each flight was determined randomly, assuming that aircraft 1 is at station AAA, aircraft 2 is at station BBB, and aircraft 3 is at station CCC at the beginning of the planning horizon.

The same network is used for all three considered scenarios, with the only difference being the supply and demand of aircraft at each node:

- In Scenario 1, for instance, the aircraft supply $ai$ is 0 for all station time nodes except $a6 = a12 = 1$. Similarly, the demands $h11 = h19 = 1$, and all other $hi = 0$.
- In Scenario 2, the aircraft supply $ai$ is 0 for all station time nodes except $a1 = a12 = 1$. Likewise, the demands $h5 = h19 = 1$, and all other $hi = 0$.
- In Scenario 3, the aircraft supply $ai$ is 0 for all station time nodes except $a1 = a6 = 1$. Correspondingly, the demands $h5 = h11 = 1$, and all other $hi = 0$.

**Table 1 Base Scenario**

| *Aircraft* | Flight | Origin | Destination | Departure | Arrival | Cancellation Cost. |
|---|---|---|---|---|---|---|
| 1 | 11 | A1 | A2 | 14:10 | 15:20 | 7.350 |
| | 12 | A2 | A3 | 16:05 | 17:00 | 10.231 |
| | 13 | A3 | A2 | 17:40 | 18:40 | 7.434 |
| | 14 | A2 | A1 | 19:20 | 20:35 | 14.191 |
| 2 | 21 | A2 | A1 | 15:45 | 17:00 | 11.198 |
| | 22 | A1 | A2 | 17:40 | 18:50 | 12.985 |
| | 23 | A2 | A3 | 19:30 | 20:30 | 11.491 |
| | 24 | A3 | A2 | 21:15 | 22:15 | 9.581 |
| 3 | 31 | A3 | A4 | 15:15 | 16:20 | 9.996 |
| | 32 | A4 | A3 | 17:30 | 18:30 | 15.180 |
| | 33 | A3 | A4 | 19:10 | 20:20 | 17.375 |
| | 34 | A4 | A3 | 21:00 | 22:05 | 15.624 |

The program was executed for the three scenarios employed by Arguello et al. (1998). The instance data used to illustrate the results are consistent with those considered by Arguello et al. (1998), as depicted in Table 1.

Arguello et al. presented a schedule for this example, wherein the cancellation cost used is a function of the number of passengers on each flight. The passenger load for each flight was determined randomly, assuming that aircraft 1 is at station AAA, aircraft 2 is at station BBB, and aircraft 3 is at station CCC at the beginning of the planning horizon.

The same network is used for all three considered scenarios, with the only difference being the supply and demand at each aircraft.

**Table 2 Objective Function Value**

| | Scenario 1 | Scenario 2 | Scenario 3 |
|---|---|---|---|
| Base Flow | 39.206 | 45.255 | 58.175 |
| MIP Delay Cost + Cancellation Cost Result | 28.050 | 22.176 | 35.285 |

The model's solution enabled the identification of an objective function that led to cost reduction, which was previously primarily associated with the complete cancellation of flights assigned to a specific aircraft's route. The algorithm discovered strategies for utilizing aircraft that had no planning irregularities, thus minimizing these costs. This entailed making decisions regarding which flights to delay, when to enact delays, and whether delaying a flight would be more advantageous than canceling it.

When examining the Cost Reduction Rates discussed in Chapter 2.1 of this study, we can calculate the minimum and maximum values resulting from these reductions.

For the Gol airline in the year 2019, a 28% reduction would equate to a minimum cost reduction of R$ 3,596,375, while a 51% reduction would yield a maximum cost reduction of R$ 6,550,541.

Conversely, for the LATAM airline, the minimum cost reduction would amount to R$ 304,832, with the maximum reduction reaching R$ 555,230. These calculations are pivotal in comprehending the range of potential cost reductions that each company could achieve based on the mentioned percentages.

Although approximation models can provide useful and closely aligned results with reality, it is imperative to acknowledge that they remain approximations and may not encapsulate all aspects and intricacies of the problem at hand.

## 7. REFERENCES


ARGUELLO, M. F. Framework for exact solutions and heuristics for approximate solutions to airlines' irregular operations control aircraft routing problem. 1997. Tese (Doutorado em filosofia) – Departamento de Engenharia Mecanica, Universidade do Texas, Austin, 1997.

ARGUELLO, M.F.; BARD, J.F; YU, G. Models and Methods For Managing Airline Irregular Operations. Operation Research in the Airline Industry, Nova Iorque, v. 9, n. 1, p. 1-45, dez. 1998.

ASSOCIAÇÃO BRASILEIRA DAS EMPRESAS AÉREAS. PANORAMA **2019.** O setor aéreo em dados e análises. São Paulo: ABEAR, 2020.

SANTOS, Haroldo G.; TOFFOLO, T. **Mixed integer linear programming with Python**. From https://media.readthedocs.org/pdf/python-mip/latest/python-mip.pdf.

THENGVALL, B.G; BARD, J.F; YU, G. Balancing user preferences for aircraft schedule recovery during irregular operations. **IIE Transactions.**

VINK, J. ET AL. Dynamic *Aircraft Recovery Problem* – An operational decision support framework. **Computers and Operations Research.** 2020.