EPiC Series in Computing
Volume 104, 2024, Pages 268–294
Proceedings of 3rd International Workshop on
Mathematical Modeling and Scientific Computing

Aircraft Flows Merging: Velocity Control and Interactions between Air Traffic Controller and Pilot

Arseniy A. Spiridonov

Krasovskii Institute of Mathematics and Mechanics,
Ural Branch of the Russian Academy of Sciences,
Yekaterinburg, Russia
spiridonov.arseniy@gmail.com

Abstract

A problem of safe aircraft flows merging in an airport approach zone is studied. In the situation under consideration, the air routes of aircraft flows have a branched structure, that is, there can be multiple paths leading an aircraft from the entry point of the flow to the final point of the scheme. Also, there may be several merge points, at which an aircraft flow joins other ones, original and/or merged earlier. At each point of the air route scheme, a safe passage of vessels must be provided, that is, the presence of a safe time interval between the instants of aircraft passage must be guaranteed. Regulation of the arrival instants to points of a scheme is carried out by changing aircraft velocities, routes, or usage of special scheme elements: holding areas, point-merge schemes, path alignments, etc. In the problem, some model is considered taking into account directions from an air traffic controller to a pilot connected with changing the aircraft's velocity and/or route. The resultant schedule of the aircraft arrivals to the scheme points is optimized from the point of view of a criterion minimizing the deviations of aircraft arrivals to the final point of the scheme from the nominal ones and the number of directions from air traffic controllers to pilots. The methodology for constructing the model for such a problem is proposed within the mixed integer linear programming framework. The case of several runways is not included, but the model can be easily extended to cover this case. Results of numerical modeling are given.

1 Introduction

1.1 Background

The main objective of air traffic management is to guide an aircraft to its destination and to do it safely. The safety is understood in the conventional sense: each pair of neighbor aircraft must be separated by a safe distance (or time interval). When aircraft move along magistral airways, the safety problem is not too critical since all vessels move with approximately the same speed and almost do not change their mutual distances. On the contrary, in approach

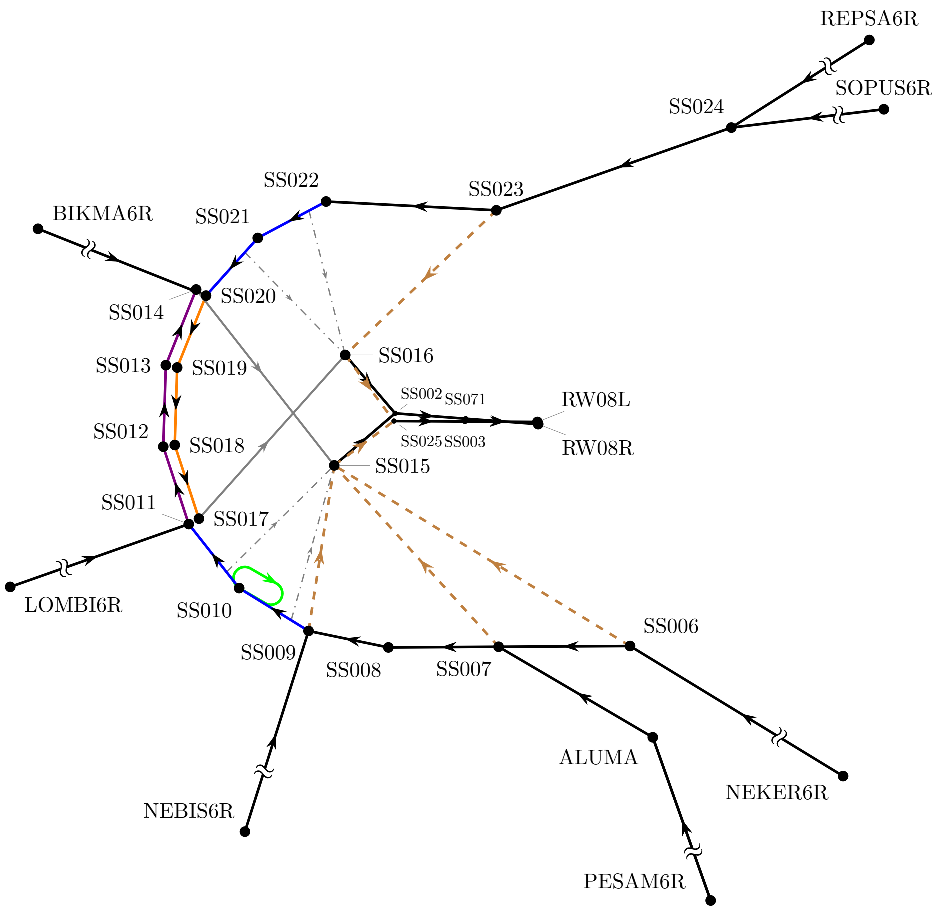

Figure 1: A schematic view of the approach zone of Koltsovo airport (Yekaterinburg, Russia), easterly heading runways approach. The scheme is drawn to scale except legs from outer points **SOPUS6R**, **REPSAR6R**, **BIKMA6R**, **LOMBI6R**, **NEBIS6R**, **PESAM6R**, and **NEKER6R** are reduced. The green oval stands for a holding area. The blue, purple, and bright brown arcs are merge-point schemes. The brown dashed lines denote segments of direct passage to points. The grey dash-dotted lines stand for possible path alignments.

zones near airports, air traffic controllers (ATCs) intensively manage aircraft motion. The main reason to this is that aircraft from flows coming from different directions appear out of step, and an ATC needs to make an effort to join them safely into one or several landing queues. At the same time, the ATC should support safety at intermediate points of the scheme.

The main problem in an approach zone is that the scheme of possible aircraft motion routes can be quite complicated. For example, in Fig. 1, one can see the approach zone scheme of Koltsovo airport (Yekaterinburg, Russia) gathering seven flows and directing them to (possibly) two runways. The flows merge at some points, further the resultant flows also merge with each other. The safety must be provided at all control points of the scheme.

However, in the modern literature, a situation is considered mostly when several aircraft flows merge at one prescribed point and aircraft have only one path reaching the final point.

Such a situation can be classified as *one-stage single-route* without any reflection of the air route structure and possibilities it makes available to regulate the air traffic. From the practical point of view, it is more important to study situations of *multi-stage* (or *cascade*) *multi-route* aircraft flows merging. Here, "*multi-stage*" assumes several sequential control points where the flows merge. "*Multi-route*" means (possible) presence of several feasible paths connecting the entry point of a flow and the final destination point (or points) of the scheme. The author has found only several papers dealing with similar situations (see the literature review below).

There is only one way to provide a safe time interval between aircraft at a control point of the scheme (assuming they both pass this point). This way is to change somehow the instants when these aircraft arrive at this point. Such a change can be made either by varying aircraft velocity, and/or by changing their routes. The latter demands to involve some elements of the scheme: path alignments, holding areas, point-merge schemes, or alternative routes. With that, one should take into account that usage of alternative routes changes the set of control points the aircraft passes and, therefore, as well the safety problems an ATC solves at those points.

Earlier, the author has considered [21, 23, 20] similar problems with a branched structure of the route scheme. In works [21, 23], a situation without multiplicity of routes is studied when for each flow, there is only one route from the entry point to the final one. In [20], a formalization is suggested for a multi-route situation. These works have followed the modern mainstream approach. The practical situation is formulated as a mixed integer linear programming (MILP) problem. The main object to be found as a solution of the obtained optimizational problem is the set of arrival instants of each aircraft to each control point along the route assigned to it.

From the practical point of view, the main flaw of this approach based on managing arrival instants is that in real life ATCs and pilots operates not in terms of times, but of velocity of an aircraft. One of the demands to the control of an aircraft is a reasonable velocity regime of motion. However, independent assignment of instants for arrival at contiguous control points even within given constraints on them implies independent switches of the aircraft velocity, what often is not reasonable.

Such switches can not appear in one-stage formulation, since it does not consider motion of aircraft within the scheme. Each flow route consists of only one segment, which is characterized by some nominal passage time θ^{nom} and interval $[\theta^{\min}, \theta^{\max}]$ of a possible variation of the arrival instant due to possibilities provided by the scheme. All maneuvers and velocity switches performed by an aircraft are hidden in the certain value chosen for the arrival to the final point.

Straightforward extension of this approach to a multi-stage merging implies another cause for "chaotic" velocity switches. MILP-solvers being linear produce the solution at the boundary of given constraints $[\theta^{\min}, \theta^{\max}]$ ignoring the nominal duration θ^{nom}, which is reasonable from the technological point of view.

Consider the following small example (see Fig. 2). Let there be two consequent segments with the equal intervals $[\theta^{\min} = 1000, \theta^{\max} = 2000]$ sec of possible passage times. The passage times of the first and final points are fixed as $t^q = 0$ sec and $t^s = 3000$ sec. Let both segments

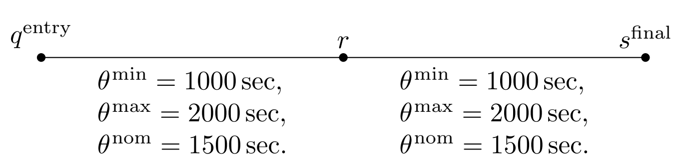

Figure 2: A simple example of disadvantage of constructing MILP-formalization in terms of arrival instants.

nominal passage duration be $\theta^{\text{nom}} = 1500$ sec if there is no need to vary the instant of arrival at the point r. In a regular situation, there is no such a need, and the arrival instant at the point r should be reasonably set to $t^r = 1500$ sec. However, as a result of application of an (MI)LP-solver, one gets either $t^r = 1000$ sec, or $t^r = 2000$ sec according to the given constraints. Such a solution leads to unnecessary velocity switches: in the first case, acceleration in the segment (q, r) and deceleration in the segment (r, s). In the second case, the change is opposite: first deceleration and then acceleration.

Thus, it is reasonable to manage the velocity of aircraft motion, not the arrival instants, and to punish velocity switches. Due to this, only those velocity switches remain, which are necessary indeed for safety of the motion. However, the arrival instants should stay in the model since the overall optimization is expressed in terms of the deviations between the assigned and nominal arrival instants at the final point of the scheme.

So, the paper discusses a method for constructing an optimizational model of aircraft merging problem, which takes into account

- managing instants of aircraft arrival to control points via variation of aircraft velocity;
- taking into account interactions between an ATC and a pilot, both for choice of a route and assigning motion velocity (to exclude unnecessary velocity switches, in particular);
- providing safety passage of an aircraft at all control points it passes according the choice of its route and routes of all other aircraft under consideration.

With that, the developed formalization does not consider the dynamics of aircraft, its real control, and/or actual motion.

1.2 Literature review

Importance of the problem of constructing a safe schedule at the final flight stages began to increase significantly since the beginning of the second half of the 20th century. At this time, the first works and studies on this problem appeared (see, for example, [18, 17]). Work [18] presents some overview of the models already available at that time for aircraft scheduling and routing problems.

Over the past few decades, a large number of different models and methods for solving the corresponding problems were proposed. Most of them are devoted to the case of one-stage single-route merging (see, for example, some reviews [3, 8, 2, 24, 25, 26, 28] and references within), when only one merge point is assumed in the problem. Due to the simplicity of the air route scheme in the case of one-stage merging, it is possible to process significant ensembles of vessels in a short time. However, such a simplified scheme does not reflect the actual airspace arrangement.

Nowadays, a very popular approach to formalize the aircraft flows merging problem is by means of mixed integer linear programming. As far as the author knows, work [1] is the first where the MILP approach is used.

Recently, works have begun to appear that consider multi-stage multi-route problem statements. For example, in work [27], some heuristic algorithm is proposed to solve a multi-route aircraft flows merging with a scheme having two point-merge schemes whose resultant queues are merged further. Some safety formalizations are considered only at the point-merge schemes, but at the final merge point, the safety is provided by a heuristic algorithm.

In work [10], a machine learning approach is used to process a multi-stage merging in a tree-like approach scheme. The authors do not specify the universality of the solver, that is, the possibility to handle different scheme structures with different characteristics of the incoming flows.

In paper [11], they consider an ensemble consisting both of arriving and departing aircraft. However, the consideration is “macroscopic” (in terms of the paper), that is, one-stage in the terminology of the current text. The only points of conflict are runways. The motion of aircraft over the air route scheme is not taken into account. So, after obtaining some optimal arrival-departure schedule, the problem of safe guiding all aircraft in the ensemble remains unsolved. ATCs must decide by theirselves how to manage an arriving aircraft from entering to the assigned runway and further during taxiing. Similar decisions should be made for a departing aircraft during taxiing and further after take off. In comparison with paper [11], the current text works at the “microscopic” level providing a definition of entire motion of aircraft over the scheme.

In the series of works by a group of Italian authors A. D'Ariano, D. Pacciarelli (see, for example, [12, 4, 6, 5, 16, 15]), a problem of multi-route multi-stage merging of aircraft flows is studied as a job-shop problem. In the formalizations, both subproblems of the original problem are solved: determining the route for each vessel and constructing the optimal (in some sense) schedule of aircraft arrivals to scheme checkpoints. In their earlier papers [12, 4, 6, 5], these subproblems are solved successively: at first, some route is chosen for each aircraft, then an optimal schedule is constructed for this set of routes, and, further, attempts are made iteratively to rebuild the route set and construct optimal schedules for the new variants. In [16, 15], both subproblems are solved simultaneously within the framework of mixed integer linear programming. In [16], strict MILP methods are used. In [15], some heuristic approaches are considered. In the papers of this series, the enumeration of possible routes of aircraft is global: all possible routes are listed in advance and chosen by an integer variable, which is the number of a taken route. On the contrary, in the current text, local selection of routes is exploited: at each control point, one of possible outgoing edges is selected by means of a corresponding indicator binary variable (see Subsection 4.3 below).

In [9], the problem of merging aircraft flows for metroplex airports is considered within the framework of the multi-stage multi-route approach. The problem statement assumes a global search of possible aircraft routes and runway assignment. Formalization is carried out within the MILP approach.

Since 1960-70's, there is an appreciable number of works considering more or less complete aerodynamics of an aircraft motion from the point of view of producing a safe schedule. But excluding the problem of velocity control, such an approach gives a lot of other problems, which make solving the problem extremely difficult.

So, the author could not find any works, in which schedule generation is carried out not in terms of time, but in terms of the velocity of aircraft without consideration of a complicated modelling of an aircraft motion. The works mentioned above and based on MILP-models do not solve the issue of “chaotic” velocity switches (see Fig. 2 and its description). In addition, the author was unsuccessful to find works that would consider some model of interaction between an ATC and a pilot ensuring the choice of the route and changing the velocity of a vessel.

1.3 Contribution of the research

The main purpose of this paper is to suggest a formalization approach giving a MILP-optimizational problem, which solution produces description of all facets of the motion of all aircraft: their routes, the instants of velocity changes, and the new velocity magnitudes expressed in the terms of directions given by an ATC to a pilot. These elements of the solution define some arrival instants to the control points, which, in their turn, provide safe passages for all aircraft at all points they pass and optimality (in some sense) of the suggested motion schedule. Now, the author does not intend to create a model, which can be solved analytically. It is supposed to pass a model obtained by means of the suggested methodology to some MILP-solver. Among all available libraries providing a MILP-solver, the optimization library `Gurobi` [7] has been chosen because it has a quite convenient interface and high performance.

The work is based to a considerable extent on works [20, 13]. From work [20], a graph description of an air route scheme is taken as well as an approach to safe aircraft route search. In [13], a colleague of the author has considered optimization of the velocity regime along a fixed route with given arrival instants to some control points along it and a model of ATC–pilot interactions.

All information regarding the work of air traffic controllers as well as supplementary material for simulations are provided by the colleagues from NITA, LLC (New Information Technologies in Aviation, Saint-Petersburg, Russia) [14].

2 Problem statement

In this work, it is assumed we are given an air route scheme. The scheme is represented as an oriented acyclic graph. Situations when an aircraft is taken out from its planned route and sent to some previous part of the scheme are considered to be exceptional and processed individually. So, in general, the motion of an aircraft is assumed to be one-directional from the entry point to a final one, and the route graph is acyclic.

Some vertices of the graph representing the control points of the scheme are marked as entry points for considered aircraft flows and someone is marked as final. Also, it is known, along which segments aircraft from a flow can move.

As an additional information to the graph, one knows locations of elements of the scheme: holding areas and path alignments. Point-merge schemes popular nowadays for providing small delays are assumed to be represented as a special case of path alignments. For each segment, wherefrom a path alignment can start, a set of points is known, whereto the aircraft can follow directly.

For each segment, a flag is given, which defines whether the order of aircraft passing this segment can be changed. This information implicitly describes presence of several echelons, which allow one aircraft moving along this segment to overtake another vessel. The echelon stack information is not considered explicitly anywhere. If it is important for some certain scheme, then it should be represented by some additional graph vertices and segments.

Each holding area has information on the minimal and maximal delays it can provide. Each control point has information on the minimal and maximal velocity of an aircraft when it passes this point. It is assumed that possible magnitudes of the aircraft velocities are taken from some discrete set representing the fact that ATCs assign some specific velocities, say, multiple to 5 or 10 knots. Also, it is supposed that a pilot can change the aircraft velocity not only at the control points of the scheme, but when moving along the segments too.

As it is said above, it is important to take into account changes of aircraft velocity. Such changes can occur due to work of an aircraft autopilot, ATCs' directions or according to technological velocity constraints for some part of the air route scheme. The problem of accounting autopilot's work is that an ATC does not know what velocity settings a certain autopilot has. So, in the suggested model, velocity is changed only due to ATCs' directions where there is some range of allowed velocities and without directions where there is a point constraint for aircraft velocity.

The choice of an aircraft route also can be performed according settings of the autopilot or according to directions of an ATC. But in contrast to velocity, route settings of autopilot are known since they reflect standard terminal arrival routes (STARs). Thus, to represent the possibility of automatic choice of the route, one segment outgoing from a vertex through aircraft of some flow can pass is marked as nominal to be chosen automatically without an interaction of a pilot with an ATC. Other outgoing segments are marked to be chosen after a respective ATC direction only.

So, the following set of ATC directions to a pilot is considered:

- "after passing the control point q, go to the point r" (motion along a non-nominal segment);
- "at the instant t, go to the point q" (a path alignment or leaving a point-merge scheme);
- "at the instant t, set the velocity v";
- "in the holding area located at the point q, work out t seconds of delay" (usage of a holding area).

It is assumed that directions from an ATC are given right in time before the prescribed operation. For a path alignments, it is possible to calculate the exact time instant, when the aircraft should align to the point.

ATCs are not allowed to combine several directions into one more complex. So, in the functional to be optimized, all ATC–pilot directions are counted individually.

As it is said earlier, the main goal is to construct a safe schedule of motion of aircraft from a given ensemble over the scheme. For each aircraft, one knows its flow number, its type (light, medium, heavy), and its entry time (when it appears at the entry point of its flow). Safety is checked only at control points, not in segments. Passage of two aircraft through a control point is considered safe if there is a certain safe time interval between the instants of their passages. This interval depends on the aircraft types and the order they pass the point. It is assumed that aircraft passage of the starting point of a flow is already safe.

Among all safe schedules, it is desirable to find some optimal. The optimality of a schedule is considered from two points of view. At first, the arrival instants to the final point assigned to aircraft should be close to some nominal values given *a priori* (for example, taken from the passenger schedule). Also, it is necessary to minimize the number of interactions between ATCs and pilots.

From this point of view, the problem is multi-criteria. However, it is reduced to a single-criterion one by means of traditional approach of linear combining the criteria (see Subsection 4.4 below).

Generally, the proposed technique of constructing MILP-formalization is designed to solve an offline form of the aircraft flows merging problem when a set of nominal instants of vessels that pass through the scheme is used as the initial data. In the future, the author also plans to suggest a technique, which allows one to construct a MILP-formalization of an aircraft flows merging problem for some real time situation when some aircraft are already inside the scheme and have passed partially their routes to the final point.

3 Input data

The input data of the problem is the following information.

Description of aircraft. For each aircraft, a tuple $\{\sigma_i, l_i, t^{\mathrm{nom}}_{i,\mathrm{entry}}, \mathtt{v}^{\mathrm{entry}}_i, t^{\mathrm{nom}}_i\}$ is given. The symbol σ_i denotes the type of the aircraft: L — light; M — medium; H — heavy. An integer value l_i is the number of the flow, in which the aircraft is running. The value $t^{\mathrm{nom}}_{i,\mathrm{entry}}$ is the nominal instant of aircraft passage of the entry point of its flow. The symbol $\mathtt{v}_i^{\text{entry}}$ stands for the velocity of the aircraft when it enters the flow. The instant t_i^{nom} is the nominal (desired) instant of arrival of the aircraft to the final point of the scheme. It participates in the functional to be optimized (see (30), (31) below).

Description of the directed acyclic graph of the route scheme.

- a set $V = \{q\}$ of vertices corresponding to points of the air route scheme; for each vertex the following parameters are given:
  - the coordinates (x, y, z) of the point (in some local airport coordinate system);
  - the minimal $\mathtt{vMin}^q$ and maximal $\mathtt{vMax}^q$ velocities allowed for aircraft passing this point;
  - if a holding area is located at the point, then the minimal $\mathtt{tMin}^q$ and maximal $\mathtt{tMax}^q$ delay times are also given, which can be performed using the holding area. In the model, without loss of generality, it is assumed that the values are equal for each aircraft.
- a set $E = \{e\}$ of edges (or segments) connecting the vertices; for each edge the following parameters are given:
  - the starting vertex;
  - the final vertex;
  - an indicator ρ^e reflecting the necessity of the direction from ATCs to pass along the segment ($\rho^e = 0$ if a direction form an ATC is not required, and $\rho^e = 1$, otherwise);
  - an indicator c^e reflecting the possibility of changing the order of vessels passing along this edge ($c^e = 1$, if the order change on segment e is possible and $c^e = 0$, otherwise).

The possibility to change the aircraft order on a segment means the presence of different flight levels on this segment and a wide range of possible velocities allows one aircraft to overtake other aircraft following this segment. Note that the explicit structure of the flight levels is not considered in the paper.

However, flight levels can be considered by introducing multiple copies of control points representing one or another flight level at a control point and edges connecting them. These edges reflect the capability of a vessel to move on one or another echelon and to descend/climb with change of the echelon. Within framework of such a representation of echelons, order changes are allowed by presence of a sufficient distance between points and a sufficient range of permissible velocities and prohibited in the opposite case. Unfortunately, such an idea increases the size of the obtained model, which leads to decreasing the efficiency of computations.

It is assumed that this graph is directed and has no cycles. Multigraphs are allowed, that is, two vertices can be connected by several segments with different properties.

The previous aircraft type, σ_i	The next aircraft type, σ_j		
	Light	Medium	Heavy
Light	60	60	60
Medium	180	60	60
Heavy	180	120	120

Table 1: The values of the safety intervals (in seconds) between aircraft of different types.

Description of aircraft flows. For each aircraft flow its entry point to the scheme is given.

All other segments, which can be passed by aircraft from a flow, are defined accordingly to the scheme graph structure.

Description of safe intervals. For each vertex q, there is a table of safety intervals $\tau^{\text{safe},q}_{\sigma_i,\sigma_j}$ between a leading aircraft of the type σ_i and the following one with the type σ_j. In performed numerical simulations, the same safety intervals $\tau^{\text{safe}}_{\sigma_i,\sigma_j}$ described in Table 1 were used at all points, since it does not violate the construction of the formalization.

4 MILP formalization

4.1 Auxiliary information about the graph of the scheme

Let a scheme graph be read from the input data, which is an acyclic directed graph $G = \langle V, E\rangle$. Here, V is the set of the graph vertices, E is the set of the edges connecting the vertices from the set V.

The original graph $G = \langle V, E\rangle$ is modified as follows. If the edge $e = (p, q) \in E$ is a segment, from which a path alignment is allowed, then a special non-nominal edge e' is added to the set E. For this edge e', the vertex p is the starting point, the final vertex r is the point, whereto the path alignment is allowed (see Fig. 3). Such an edge requires a direction from ATCs, that is, $\rho^{(p,r)} = 1$ for this edge.

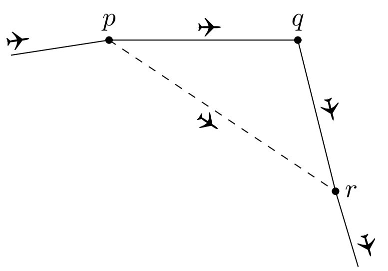

Figure 3: The case of path alignment from the segment $e = (p, q)$ to one point r. The edge $e' = (p, r)$ is special and added to process path alignment.

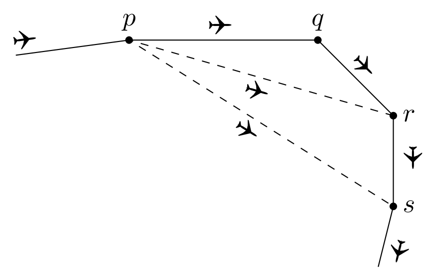

Figure 4: The case of path alignment from the segment $e = (p, q)$ to two points: r and s. The segments $e' = (p, r)$ and $e'' = (p, s)$ are special and added to process path alignment.

If it is possible to align from the segment $e = (p, q) \in E$ to several points of the scheme (see Fig. 4), then for each such a point a special non-nominal edge is introduced (edges are added to the set E) with the beginning at the point p and the end at the point whereto the path alignment is possible. For each such an edge, a direction from an ATC to pass this edge is required.

When constructing a scheme graph in the case of point-merge schemes (see Fig. 5), the edges of the waiting arc (that is, between the points $q, \ldots, r$ in Fig. 5) can be considered as edges, from which a path alignment to the merge point of the point-merge scheme is possible. For these edges, similar constructions are performed as described earlier for the processing of path alignments.

In addition, for point-merge schemes, in practice, a direct pass without using the waiting arc of the point-merge scheme is allowed. For this, a non-nominal edge from the point q of the

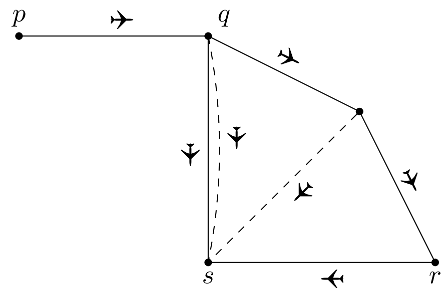

Figure 5: The case of the point-merge scheme starting from the point q with the final point s. The edge (q, s), drawn with a solid line, is introduced to process direct path. The edge (r, s) is introduced to process passage the whole waiting arc and following to the final point of the point-merge scheme.

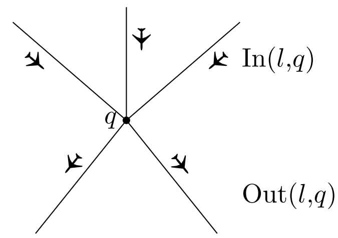

Figure 6: The sets $\mathrm{In}(l, q)$ of the edges of some flow l coming to the vertex q, and $\mathrm{Out}(l, q)$ of the edges outgoing from the vertex q.

beginning of the waiting arc to the point s is introduced to the set E (see Fig. 5, the edge (q, s), drawn with a solid line). Such an edge also requires a direction from ATCs, that is, $\rho^{(q,s)} = 1$ for this edge. Thus, there are two edges (q, s) with different properties: the edge of direct path (solid line) and the edge of path alignment (dashed line).

Finally, a non-nominal edge is introduced for the case when an aircraft passes the entire waiting arc of the point-merge scheme and goes to its final point (see the segment (r, s) drawn with a solid line in Fig. 5). Passage along this edge also requires a direction from an ATC.

In real point-merge schemes, there is an emergency path from the final point of the waiting arc. This path is used when an ATC is unable to put some aircraft into the landing queue and directs it back to some part of the scheme. From the point of view of the suggested model, such a situation is exceptional and is not considered at all. This emergency path is not considered as a part of the constructed graph of the air route scheme.

For each segment e of the graph G, two constants $\texttt{distMin}^e$ and $\texttt{distMax}^e$ related to its length are introduced. Here, $\texttt{distMin}^e$ and $\texttt{distMax}^e$ are the minimal and maximal values of the segment length, respectively. For each special segment introduced to process path alignment (see, for example, the edge (p, r) in Fig. 3), $\texttt{distMin}^e$ is equal to the distance between the starting point of the segment, from which the path is aligned, and the point, whereto the path is aligned, that is, $\texttt{distMin}^e = \big|(p, r)\big|$ in terms of Fig. 3. Here and below, the denotations $|e|$ and $\big|(p, q)\big|$ mean the length of the corresponding segment. In other words, it is the length of the shortest path to the point to align. And $\texttt{distMax}^e$ is equal to the sum of the length of segment, from which alignment is possible, and the distance between its final point and the point, whereto the path is aligned, that is, $\texttt{distMax}^e = \big|(p, q)\big| + \big|(q, r)\big|$ in terms of Fig. 3. In other words, it is the length of the longest path to the point to align. Note, that for the segments introduced to process a path alignment, $\texttt{distMin}^e < \texttt{distMax}^e$. For other segments of the graph G, $\texttt{distMin}^e = \texttt{distMax}^e$ and they are equal to the length of the segment e computed of the basis of the coordinates of its end points.

Possible routes of aircraft motion within the lth flow form a subgraph $G_l = \langle V_l, E_l \rangle$. Here, V_l is the set of vertices, through which the routes of aircraft from the flow l can pass, E_l is the set of edges between vertices from the set V_l.

The subgraph G_l is obtained by a depth-first search in the modified graph G starting from the entry point of the lth flow. Also, during this search, for each vertex $q \in V_l$, sets $\mathrm{In}(l, q)$ and $\mathrm{Out}(l, q)$ of edges are formed (see Fig. 6). Let $\mathrm{In}(l, q)$ denote the set of edges, along which aircraft from the flow l can get to the vertex $q \in V_l$. Similarly, $\mathrm{Out}(l, q)$ denotes the set of edges, along which vessels from the flow l can leave the vertex $q \in V_l$. For the vertices q corresponding to the entry points to the flows, it is supposed $\mathrm{In}(l, q) = \emptyset$. Similarly, for the vertex q corresponding to the end point of the route, one considers $\mathrm{Out}(l, q) = \emptyset$.

It is natural to assume that if aircraft from different flows come to one check point, then they can move along all routes leaving this point. Within the framework of the introduced notation, this assumption can be written as follows: if $q \in V_{l'} \cap V_{l''}$, then $\mathrm{Out}(l', q) = \mathrm{Out}(l'', q)$. However, this assumption is not essential for the formalization proposed below and these sets may differ: aircraft from different flows may leave some vertices common for their flows along some different sets of legs.

4.2 Velocity grid

In [13], they work with a non-linear model connected with the formula of uniform motion $S = v \cdot t$ where both the velocity v and time t are variables. Such quadratic problems can be solved by `Gurobi`, but performance of computations in this case is very low in comparison with the linear case. When such an approach was explicitly applied to the aircraft flows merge problem, even model examples with ensembles of 5 aircraft have not been computed. So, some additional efforts are necessary to take into account aircraft velocities.

One of the ways to overcome the velocity non-linearity is to consider a discrete set of possible velocity values. Of course, in this case, the non-linearity does not disappear, but it becomes discrete, connected with integer variables. Discrete non-linearities can be linearized.

Another reason to pass to discrete velocities is that in real life ATCs assign not some arbitrary values of velocity, but some multiples to a convenient number, for example, to 5 or 10 knots.

Thus, in the text, some discrete set $\mathcal{V} = \{v_j\}$, $j = 0, \ldots, J$, $v_0 < v_1 < \ldots < v_J$, of possible values of aircraft velocity is considered. The range $[v_0, v_J]$ covers all permissible values of velocity of aircraft over the scheme. For example, one can use a uniform grid $v_j = v_0 + j \cdot \mathtt{dV}$, where v_0 is the minimal permissible velocity value, `dV` is some reasonable step value of the grid.

4.3 Variables and constants

Below, there is a list of variables used in the model and domains of their definitions.

Hereinafter, the point $q \in V$ is called *regular*, if there is no holding area; the points with holding areas are called *irregular*. If regularity of a point is not focused in the description of variables and constraints, then such constraints are common for both regular points and points with holding areas.

Continuous variables.

For each aircraft i from the flow l and for each point $q \in V_l$, the following continuous variables are created:

- $\mathtt{tIn}_i^q$ is the instant of coming of the ith aircraft to the point q.
- $\mathtt{tOut}_i^q$ is the instant of leaving of the ith aircraft of the point q.

  For the final point q of the scheme, the corresponding variable $\mathtt{tOut}_i^q$ is not created and considered. Splitting the times of coming and leaving are necessary to process holding areas.

Binary variables.

For each aircraft i from the flow l and for each segment $e = (q, r) \in E_l$, the following binary variable is created:

- $\xi_i^e = \xi_i^{(q,r)}$ reflects the fact of passage of the segment e by the aircraft i: $\xi_i^e = 1$ if the aircraft i passes the segment e, and $\xi_i^e = 0$ otherwise.

For each aircraft i from the flow l and for each point $q \in V_l$, the following binary variables are created:

- ζ_i^q reflects the fact of passage of the point q by the aircraft i: $\zeta_i^q = 1$ if the aircraft i passes the point q, and $\zeta_i^q = 0$ otherwise.

  For the entry point of the flow l and for the final point of the scheme, the corresponding variables ζ_i^q are equal to 1, since the aircraft i certainly passes these points.

- $b_{i,j}^q$ is an indicator variable reflecting the chosen value of the velocity of the ith aircraft at the point q corresponding to the jth node of the velocity grid $\mathcal{V}$: $b_{i,j}^q = 1$ if for the ith aircraft the velocity corresponding to the jth node of the grid at the point q is set and $b_{i,j}^q = 0$, otherwise.

  The values of the variables $b_{i,j}^q$ corresponding to the nodes of the grid $\mathcal{V}$ outside the range $[\mathtt{vMin}^q, \mathtt{vMax}^q]$ of the permissible velocity values at the point q are identically equal to 0. The values of the variables $b_{i,j}^q$ corresponding to the nodes of the grid $\mathcal{V}$ inside this range are determined in the process of searching for a solution to the problem. If not a single node falls into the range $[\mathtt{vMin}^q, \mathtt{vMax}^q]$ of the point q, then the node closest to the range is chosen and the corresponding variable $b_{i,j}^q$ is allowed to be equal to 1.

  For the entry point of the flow l, the variable $b_{i,j^{\text{entry}}}^q$ corresponding to the node closest to the entry velocity $\mathtt{v}_i^{\text{entry}}$ is set to 1. Other variables $b_{i,j}^q$, $j \neq j^{\text{entry}}$, are set to 0.

  It is assumed that for points with a holding area, the velocity, at which an aircraft enters the holding area at such a point, is kept up to the instant of leaving the area and the point.

For each aircraft i from the flow l and for each irregular point $q \in V_l$, the following variable is created:

- $\mathtt{toHA}_i^q$ reflects the fact of entering to the holding area at the point q by the aircraft i: $\mathtt{toHA}_i^q = 1$ if the aircraft i delays in the holding area at the point q, and $\mathtt{toHA}_i^q = 0$ otherwise.

  For the point q, which does not have a holding area, the corresponding variable $\mathtt{toHA}_i^q$ is not created and considered.

  It is assumed that the final point q does not have a holding area, so the corresponding variable $\mathtt{toHA}_i^q$ also is not considered. (One can abandon this requirement by introducing in the model the variables $\mathtt{tOut}_i^q$ and related constraints for the final point q.)

For each aircraft i from the flow l', for each aircraft j from the flow l'', and for each regular point $q \in V_{l'} \cap V_{l''}$, the following variable is created:

- $\pi_{i,j}^q$ reflects the aircraft order in the pair (i, j) when passing the point q: $\pi_{i,j}^q = 1$ if holds $\mathtt{tIn}_i^q < \mathtt{tIn}_j^q$, and $\pi_{i,j}^q = 0$ otherwise.

For each aircraft i from the flow l', for each aircraft j from the flow l'', and for each irregular point $q \in V_{l'} \cap V_{l''}$, the following variables are created:

- $\pi^{\text{ii},q}_{i,j}$ reflects the order of reaching the point q by the aircraft i and j: $\pi^{\text{ii},q}_{i,j} = 1$ if holds $\mathtt{tIn}^q_i < \mathtt{tIn}^q_j$, and $\pi^{\text{ii},q}_{i,j} = 0$ otherwise.
- $\pi^{\text{io},q}_{i,j}$ reflects the order of instants when the ith aircraft enters to the point q and the jth aircraft leaves it: $\pi^{\text{io},q}_{i,j} = 1$ if holds $\mathtt{tIn}^q_i < \mathtt{tOut}^q_j$, and $\pi^{\text{io},q}_{i,j} = 0$ otherwise.
- $\pi^{\text{oi},q}_{i,j}$ reflects the order of instants when the ith aircraft leaves the point q and the jth aircraft enters to it: $\pi^{\text{oi},q}_{i,j} = 1$ if holds $\mathtt{tOut}^q_i < \mathtt{tIn}^q_j$, and $\pi^{\text{oi},q}_{i,j} = 0$ otherwise.
- $\pi^{\text{oo},q}_{i,j}$ reflects the order of instants when the aircraft i and j leave the point q: $\pi^{\text{oo},q}_{i,j} = 1$ if holds $\mathtt{tOut}^q_i < \mathtt{tOut}^q_j$, and $\pi^{\text{oo},q}_{i,j} = 0$ otherwise.

The necessity for these four variables π at such points is stipulated by the fact that in the case of entering to a holding area, an aircraft passes the point twice: when entering the holding area and leaving it. Thus, it is necessary to check the safety at both these instants. Note that if any aircraft from a pair does not enter to a holding area, then the values of the corresponding variables π should be equal (see below (23)–(25)) since the corresponding instants tIn and tOut should be equal (see below (21)).

Constants.

Some global constants used in the formalization:

- M denotes "practical" infinity in computations; it is selected equal to 10^7;
- $\mathtt{T}^{\text{day}}$ is the maximal value for the variables indicating the time; it is selected equal to 10^5 seconds. It is used to separate all time variables from the practical infinity M.

4.3.1 Constraints connected with regular points

This section describes the process of creating constraints for regular points of the scheme. The constraints related to holding areas are described in Subsubsection 4.3.3.

Constraints connected with instants of passage of checkpoints. For each aircraft i from the flow l and for its entry point q, the following equality holds, since the instant of aircraft entry to the scheme is known from the input data of the problem:

$$\mathtt{tIn}^q_i = t^{\text{nom}}_{i,\text{entry}}. \tag{1}$$

For each aircraft i from the flow l and for each point $q \in V_l$, the following constraints are imposed on values $\mathtt{tIn}^q_i$ and $\mathtt{tOut}^q_i$ determining the instants coming to the point and leaving it:

$$\mathtt{tIn}^q_i \in [t^{\text{nom}}_{i,\text{entry}}, t^{\text{nom}}_{i,\text{entry}} + \mathtt{T}^{\text{day}}], \tag{2}$$

$$\mathtt{tOut}^q_i \in [t^{\text{nom}}_{i,\text{entry}}, t^{\text{nom}}_{i,\text{entry}} + \mathtt{T}^{\text{day}}]. \tag{3}$$

For the entry point q of the flow l, constraint (2) is not imposed since the variable $\mathtt{tIn}^q_i$ is determined by constraint (1) and (2) is satisfied by default.

For the final point q of the scheme, constraint (3) is not imposed since the variable $\mathtt{tOut}^q_i$ is not considered.

For each aircraft i from the flow l and for each regular point $q \in V_l$, except the final point of the scheme, it is assumed that

$$\mathtt{tOut}^q_i = \mathtt{tIn}^q_i. \tag{4}$$

The case of inequality of the instants $\texttt{tOut}_i^q$ and $\texttt{tIn}_i^q$ is possible if the point q has a holding area (that is, the point is irregular). The creation of constraints for the points with holding areas is described in Subsubsection 4.3.3.

Constraint (4) is not imposed for the final point of the scheme, since the variable $\texttt{tOut}_i^q$ is not created and considered for the final point of the scheme.

Constraints connected with aircraft velocity. For each aircraft i from the flow l and for each point $q \in V_l$, the constraint is imposed that determines the unique choice of a velocity value at this point in the case of aircraft passage through the point:

$$\sum_{j=0}^{J} b_{i,j}^q = \zeta_i^q. \tag{5}$$

If the ith aircraft does not pass the point q ($\zeta_i^q = 0$), then the sum in the left-hand side of constraint (5) is equal to 0, what means that all the variables $b_{i,j}^q$ in the sum are equal to 0. This means that if the aircraft i does not pass the point q, then no velocity of its passage is chosen for this point.

On the contrary, if the aircraft i passes the point q ($\zeta_i^q = 1$), then the sum in the left-hand side of constraint (5) is equal to 1. Due to the binary nature of the variables $b_{i,j}^q$, one and only one variable under the sign of the sum is equal to 1, others equal 0. This means that if the ith aircraft passes the point q, then for this point, some one velocity value is chosen.

Constraints connected with aircraft order. For each ordered pair of an aircraft i from the flow l' and an aircraft j from the flow l'', the variables that determine the order of their passage of a regular point $q \in V_{l'} \cap V_{l''}$ are connected as follows:

$$\pi_{i,j}^q + \pi_{j,i}^q = 1. \tag{6}$$

Due to the binary nature of the variables and this constraint, only one of them equals 1, the other is equal to 0. Their values determine the order of the aircraft i and j at the point q.

If the point q is the entry point of the flow l' (and l''), then the values of the variables $\pi_{i,j}^q$ and $\pi_{j,i}^q$ are specified on the basis of the aircraft entries on the scheme:

$$\begin{aligned} &\pi_{i,j}^q = 1, \quad \pi_{j,i}^q = 0, \ \text{if } \texttt{tIn}_i^q = t_{i,\text{entry}}^{\text{nom}} \le \texttt{tIn}_j^q = t_{j,\text{entry}}^{\text{nom}}, \\ &\pi_{i,j}^q = 0, \quad \pi_{j,i}^q = 1, \ \text{otherwise.} \end{aligned} \tag{7}$$

Therefore, in the case of the entry point, constraint (6) is not imposed.

Constraints connected with safety. For each pair of the aircraft i from the flow l' and the aircraft j from the flow l'', $i \neq j$, passing the regular point $q \in V_{l'} \cap V_{l''}$, the following constraints provide their safety passage of this point:

$$\texttt{tIn}_i^q - \texttt{tIn}_j^q + \tau_{\sigma_i,\sigma_j}^{\text{safe},q} \cdot \pi_{i,j}^q - \pi_{j,i}^q \cdot \texttt{M} - (2 - \zeta_i^q - \zeta_i^q) \cdot \texttt{M} \le 0. \tag{8}$$

Due to the regularity of the point q and constraint (4), any of the instants tIn and tOut related to the aircraft i and j can be considered in constraint (8).

Constraint (8) becomes insubstantial due to the value M in the case if at least one of the aircraft does not pass through the point q, that is, if at least one of variables ζ_i^q or ζ_j^q equal 0.

If both vessels pass the point q, that is, if $\zeta_i^q = \zeta_j^q = 1$, then the last term in the braces is equal to 0 and constraint (8) becomes standard for describing the order and safety of the aircraft i and j at the point q. If the ith aircraft arrives at the point q earlier than the jth one ($\pi_{i,j}^q = 1$, $\pi_{j,i}^q = 0$), constraint (8) has the form $\mathtt{tIn}_i^q - \mathtt{tIn}_j^q + \tau_{\sigma_i,\sigma_j}^{\mathrm{safe},q} \le 0$ and guaranties the presence of the safety interval $\tau_{\sigma_i,\sigma_j}^{\mathrm{safe},q}$ between the aircraft arrivals to the point q. Otherwise, if the jth aircraft passes the point q earlier than the ith one ($\pi_{i,j}^q = 0$, $\pi_{j,i}^q = 1$), then, due to the constant $\mathtt{M}$, constraint (8) becomes insubstantial. At the same time, the symmetric constraint works with the exchanged indices i and j.

If the regular point q is the entry point of the flow l' (and l''), then it is not necessary to impose constraint (8) (and paired to it), since the safety of the vessels of one flow at its entry point is provided *a priori*.

4.3.2 Constraints connected with segments

Constraints connected with passage of segments and points. For each vessel i from the flow l and all vertices $q \in V_l$, the following constraints are imposed:

$$\zeta_i^q = \sum_{e \in \mathrm{In}(l,q)} \xi_i^e, \quad \zeta_i^q = \sum_{e \in \mathrm{Out}(l,q)} \xi_i^e. \tag{9}$$

Constraints (9) reflect the fact that if the ith aircraft of the flow l passes through the vertex $q \in V_l$ (that is, if $\zeta_i^q = 1$), then it comes to this vertex along some one edge $e' \in \mathrm{In}(l,q)$ and then leaves this vertex along some one edge $e'' \in \mathrm{Out}(l,q)$. Conversely, if the ith aircraft of the flow l does not pass through the vertex $q \in V_l$ (that is, if $\zeta_i^q = 0$), then it could not pass any edge e from $\mathrm{In}(l,q)$ and $\mathrm{Out}(l,q)$. If $\mathrm{In}(l,q) = \emptyset$ or $\mathrm{Out}(l,q) = \emptyset$, then the corresponding restrictions are not imposed. In particular, the first of restrictions (9) is not imposed for the entry points of aircraft flows, the second one is not imposed for the final point of the scheme.

Constraints connected with possible order change. For each ordered pair of aircraft i from the flow l' and j from the flow l'' and for each segment $e = (q,r) \in E_{l'} \cap E_{l''}$, the connection between the variables $\pi_{i,j}^q$ and $\pi_{i,j}^r$ is described by the following constraint:

$$-(2 - \xi_i^e - \xi_j^e) - c^e + \pi_{i,j}^q \le \pi_{i,j}^r \le \pi_{i,j}^q + c^e + (2 - \xi_i^e - \xi_j^e). \tag{10}$$

If at least one vessel does not pass along the segment e ($\xi_i^e = 0$ and/or $\xi_j^e = 0$), then the values of the variables $\pi_{i,j}^q$ and $\pi_{i,j}^r$ are not connected, since constraint (10) does not impose significant restrictions.

If both vessels pass along the edge e ($\xi_i^e = \xi_j^e = 1$), then two cases are possible. If the order change on the segment e is prohibited ($c^e = 0$), then constraint (10) transforms into the form $\pi_{i,j}^q \le \pi_{i,j}^r \le \pi_{i,j}^q$, that is, $\pi_{i,j}^r = \pi_{i,j}^q$. It means that the orders of passage of the points q and r by the aircraft i and j are the same.

On the contrary, if the order change on the segment e is allowed ($c^e = 1$), then constraints (10) does not define any connection on the variables $\pi_{i,j}^r$ and $\pi_{i,j}^q$, that is, the order of passage of the points q and r by the aircraft i and j may differ.

If the initial point q of the segment e contains a holding area, then in constraint (10), one needs to use the variable $\pi_{i,j}^{\mathrm{oo},q}$ instead of the variable $\pi_{i,j}^q$.

If the last point r of the segment e contains a holding area, then in constraint (10), one needs to use the variable $\pi_{i,j}^{\mathrm{ii},r}$ instead of the variable $\pi_{i,j}^r$.

Additionally, it is possible to use a constraint on changing the order of vessels of the same type: for each aircraft i from the flow l' of a given type, for each aircraft j from the flow l'' of the same type as the vessel i, for each $e = (q, r) \in E_{l'} \cap E_{l''}$ when generating constraint (10), the value of the flag c^e is assumed to be equal to 0.

Constraints connected with time of motion along segments. As it is said earlier, one of the goals is to take into account switches of velocity of an aircraft because this is a tool to control unnecessary velocity switches and to assign passage times for contiguous segments according presence of absence of the velocity switches. One of the reasons to pass to a discrete set of velocity values is just the desire to control the switches because it is possible to implement such a control for discrete set of allowed values. The variables $b^q_{i,j}$ have been entered just to index the current choice of a velocity.

But now, it is necessary to pass from the velocity index variables $b^q_{i,j}$ to passage times since namely the passage times allow one to connect the instants when an aircraft is at neighbor control points of its route.

If a velocity is switched during motion along some segment there are two degrees of freedoms: the w value of the velocity and the instant when the switch occurs. Both of them affect the passage time of the segment where the switch takes place.

The chain linking a velocity at some segment and possible passage times is the following:

1. knowing the variables b, the corresponding velocities, initial and final, are known;
2. knowing the velocities, the possible ranges of passage times (for both of them) of the segment are known; here, it is also should be taken into account if the segment corresponds to the situation of a path alignment;
3. taking minium and maximum of all these times, one gets the entire interval of all possible passage times;
4. finally, these extreme values are used to connect the instant of passage the starting and final points of the segment.

Namely, these ideas are fulfilled below in terms of variables and constraints. Some of the constraints are non-linear, but since binary variables are involved, the constraints are linearized (unfortunately, by introducing new binary variables).

The introduction of a discrete set $\mathcal{V}$ of possible velocity values of movement leads to discrete sets

$$\left\{\Delta t^{\min,e}_j = \mathtt{distMin}^e / v_j\right\}, \quad \left\{\Delta t^{\max,e}_j = \mathtt{distMax}^e / v_j\right\} \tag{11}$$

of minimal and maximal durations of passage of each segment e of the scheme depending on the minimal $\mathtt{distMin}^e$ and maximal $\mathtt{distMax}^e$ lengths of the segment. For special segments introduced to process path alignments, one has $\mathtt{distMin}^e < \mathtt{distMax}^e$ and the noncoincidence of these sets. For the other segments, one has $\mathtt{distMin}^e = \mathtt{distMax}^e$ and, as a sequence, coincidence of these sets.

In this model, it is supposed that the velocity with which an aircraft arrives at the point is maintained upon leaving it, but can be instantly changed when moving along the segment outgoing from this point.

For each aircraft i from the flow l and for each segment $e = (q, r) \in E_l$, four auxiliary variables $\delta t^{\mathrm{Out},\min,e}_i$, $\delta t^{\mathrm{Out},\max,e}_i$, $\delta t^{\mathrm{In},\min,e}_i$, and $\delta t^{\mathrm{In},\max,e}_i$ are created. The first two describe the minimal and maximal times of passage of the edge e with the velocity chosen when leaving the point q; the next pair of the variables describes the minimal and maximal times of passage of the edge e with the velocity with which the ith aircraft arrives at the point r:

$$\begin{aligned} \delta t_i^{\mathrm{Out},\min,e} &= \sum_{j=0}^{J} b_{i,j}^q \cdot \Delta t_j^{\min,e}, \quad \delta t_i^{\mathrm{Out},\max,e} = \sum_{j=0}^{J} b_{i,j}^q \cdot \Delta t_j^{\max,e}, \\ \delta t_i^{\mathrm{In},\min,e} &= \sum_{j=0}^{J} b_{i,j}^r \cdot \Delta t_j^{\min,e}, \quad \delta t_i^{\mathrm{In},\max,e} = \sum_{j=0}^{J} b_{i,j}^r \cdot \Delta t_j^{\max,e}. \end{aligned} \tag{12}$$

For the ordinary edges, one has $\delta t_i^{\mathrm{Out},\min,e} = \delta t_i^{\mathrm{Out},\max,e}$ and $\delta t_i^{\mathrm{In},\min,e} = \delta t_i^{\mathrm{In},\max,e}$; for the special segments introduced to process path alignments, $\delta t_i^{\mathrm{Out},\min,e} < \delta t_i^{\mathrm{Out},\max,e}$ and $\delta t_i^{\mathrm{In},\min,e} < \delta t_i^{\mathrm{In},\max,e}$.

For each aircraft i from the flow l and for each segment $e = (q, r) \in E_l$, constraints on the passage time of the segment e have the following form:

$$\delta t_i^{\min,e} = \min(\delta t_i^{\mathrm{Out},\min,e}, \delta t_i^{\mathrm{In},\min,e}), \tag{13}$$

$$\delta t_i^{\max,e} = \max(\delta t_i^{\mathrm{Out},\max,e}, \delta t_i^{\mathrm{In},\max,e}), \tag{14}$$

$$-(1 - \xi_i^e) \cdot \mathtt{M} + \delta t_i^{\min,e} \le \mathtt{tIn}_i^r - \mathtt{tOut}_i^q \le \delta t_i^{\max,e} + (1 - \xi_i^e) \cdot \mathtt{M}. \tag{15}$$

If the ith aircraft does not pass the segment e ($\xi_i^e = 0$), then constraint (15) becomes insubstantial. Otherwise, the ith aircraft passes the segment e ($\xi_i^e = 1$), then constraint (15) takes the form

$$\delta t_i^{\min,e} \le \mathtt{tIn}_i^r - \mathtt{tOut}_i^q \le \delta t_i^{\max,e}.$$

In the definitions of the values $\delta t_i^{\min,e}$ and $\delta t_i^{\max,e}$, the non-linear operations (max and min) are used. Their linearization is performed by introducing additional binary variables

$$\theta_i^{\min,e} = \begin{cases} 1, & \text{if } \delta t_i^{\min,e} = \delta t_i^{\mathrm{Out},\min,e} \text{ (that is, if } \delta t_i^{\mathrm{Out},\min,e} \le \delta t_i^{\mathrm{In},\min,e}), \\ 0, & \text{otherwise (that is, if } \delta t_i^{\mathrm{Out},\min,e} \ge \delta t_i^{\mathrm{In},\min,e}), \end{cases}$$

$$\theta_i^{\max,e} = \begin{cases} 1, & \text{if } \delta t_i^{\max,e} = \delta t_i^{\mathrm{Out},\max,e} \text{ (that is, if } \delta t_i^{\mathrm{Out},\max,e} \ge \delta t_i^{\mathrm{In},\max,e}), \\ 0, & \text{otherwise (that is, if } \delta t_i^{\mathrm{Out},\max,e} \le \delta t_i^{\mathrm{In},\max,e}). \end{cases}$$

Note that the equality $\delta t_i^{\mathrm{Out},\min,e} = \delta t_i^{\mathrm{In},\min,e}$ ($\delta t_i^{\mathrm{Out},\max,e} = \delta t_i^{\mathrm{In},\max,e}$) is reflected in both variants of the values $\theta_i^{\min,e}$ ($\theta_i^{\max,e}$), which corresponds to the fact that in the case of equality of these values, their minimum (maximum) is equal to any of them.

Using the variables $\theta_i^{\min,e}$ and $\theta_i^{\max,e}$, relations (13) and (14) are linearized as

$$-(1 - \theta_i^{\min,e}) \cdot \mathtt{M} + \delta t_i^{\mathrm{Out},\min,e} \le \delta t_i^{\min,e} \le \delta t_i^{\mathrm{Out},\min,e}, \tag{16}$$

$$-\theta_i^{\min,e} \cdot \mathtt{M} + \delta t_i^{\mathrm{In},\min,e} \le \delta t_i^{\min,e} \le \delta t_i^{\mathrm{In},\min,e}, \tag{17}$$

$$\delta t_i^{\mathrm{Out},\max,e} \le \delta t_i^{\max,e} \le \delta t_i^{\mathrm{Out},\max,e} + (1 - \theta_i^{\max,e}) \cdot \mathtt{M}, \tag{18}$$

$$\delta t_i^{\mathrm{In},\max,e} \le \delta t_i^{\max,e} \le \delta t_i^{\mathrm{In},\max,e} + \theta_i^{\max,e} \cdot \mathtt{M}. \tag{19}$$

If $\delta t_i^{\mathrm{Out},\min,e} \le \delta t_i^{\mathrm{In},\min,e}$ ($\theta_i^{e,\min} = 1$), then constraint (16) takes the form $\delta t_i^{\min,e} = \delta t_i^{\mathrm{Out},\min,e}$, and constraint (17) imposes weaker conditions on the variable $\delta t_i^{\min,e}$.

Conversely, if $\delta t_i^{\mathrm{Out},\min,e} > \delta t_i^{\mathrm{In},\min,e}$ ($\theta_i^{e,\min} = 0$), then constraint (17) takes the form $\delta t_i^{\min,e} = \delta t_i^{\mathrm{In},\min,e}$, and constraint (16) imposes weaker conditions on the variable $\delta t_i^{\min,e}$.

Similar reasoning applies to relations (18) and (19).

4.3.3 Constraints connected with irregular points

Constraints connected with usage of holding areas and passing the points where the areas are located. For each aircraft i from the flow l and for each irregular point $q \in V_l$, the following constraint is imposed:

$$\mathtt{toHA}_i^q \le \zeta_i^q. \tag{20}$$

If the ith aircraft does not pass the point q ($\zeta_i^q = 0$), then it cannot enter the holding area ($\mathtt{toHA}_i^q = 0$). Conversely, if this vessel enters the holding area at the point q ($\mathtt{toHA}_i^q = 1$), then it must pass this point ($\zeta_i^q = 1$).

Constraint connected with time spent in holding area. For each aircraft i from the flow l and for each irregular point $q \in V_l$, a constraint is imposed on the relationship between the instants $\mathtt{tIn}_i^q$ and $\mathtt{tOut}_i^q$:

$$-(1-\zeta_i^q)\cdot \mathtt{M} + \mathtt{tMin}^q \cdot \mathtt{toHA}_i^q + \mathtt{tIn}_i^q \le \mathtt{tOut}_i^q \le \mathtt{tIn}_i^q + \mathtt{tMax}^q \cdot \mathtt{toHA}_i^q + (1-\zeta_i^q)\cdot \mathtt{M}. \tag{21}$$

If the ith aircraft does not pass the point q ($\zeta_i^q = 0$), then constraint (21) becomes insubstantial. Otherwise, if the vessel passes the point q ($\zeta_i^q = 1$), then the following logic is applied.

If the aircraft i does not enter the holding area at point q ($\mathtt{toHA}_i^q = 0$), then the constraint (21) takes the form $\mathtt{tIn}_i^q \le \mathtt{tOut}_i^q \le \mathtt{tIn}_i^q$, that is, $\mathtt{tOut}_i^q = \mathtt{tIn}_i^q$. Conversely, if the vessel i enters the holding area at the point q ($\mathtt{toHA}_i^q = 1$), then the instant $\mathtt{tOut}_i^q$ of leaving the holding area is chosen from the range $[\mathtt{tIn}_i^q + \mathtt{tMin}^q, \mathtt{tIn}_i^q + \mathtt{tMax}^q]$, which corresponds to the delay time provided by this holding area.

Generally speaking, constraint (21) works only in the case of usage of the holding area by the aircraft i. In other cases (no passage of the point q or passage without usage the holding area), the constraint becomes insubstantial. Indeed, this logic can be provided only with the variables $\mathtt{toHA}_i^q$, since they are subject together with the variables ζ_i^q of constraint (20). Therefore, the terms $\pm(1-\zeta_i^q)\cdot \mathtt{M}$ can be omitted.

Constraints connected with aircraft order. When considering a pair of aircraft passing a point with a holding area, it is necessary to take into account the potential entry of one or both vessels into the holding area and a possible change in the order of passage in this pair. A change in the order may occur due to one vessel not entering the holding area and overtaking the one, having entered there. Another situation of order change is when both vessels enter the holding area, but the one that was going ahead works out a greater delay in the holding area.

To take into account the order in a pair (i, j) of aircraft when they enter and exit the holding area, one needs 4 sets of binary variables responsible for the order in which these aircraft pass a given point: $\{\pi_{i,j}^{\mathrm{ii},q}\}$, $\{\pi_{i,j}^{\mathrm{io},q}\}$, $\{\pi_{i,j}^{\mathrm{oi},q}\}$, $\{\pi_{i,j}^{\mathrm{oo},q}\}$.

For each ordered pair of aircraft i from the flow l' and j from the flow l'', the variables determining the order in which vessels pass the irregular point $q \in V_{l'} \cap V_{l''}$, are related as follows:

$$\pi_{i,j}^{\mathrm{ii},q} + \pi_{j,i}^{\mathrm{ii},q} = 1, \quad \pi_{i,j}^{\mathrm{io},q} + \pi_{j,i}^{\mathrm{io},q} = 1, \quad \pi_{i,j}^{\mathrm{oi},q} + \pi_{j,i}^{\mathrm{oi},q} = 1, \quad \pi_{i,j}^{\mathrm{oo},q} + \pi_{j,i}^{\mathrm{oo},q} = 1. \tag{22}$$

Similarly to constraint (6), each of the constraints in this group determines the order of passage of vessels i and j of the point q in the case of entering to/passing by the holding area by one or both aircraft.

If the point q is the entry point of the flow and it has a holding area, then the values of the variables $\pi_{i,j}^{\mathrm{ii},q}$ and $\pi_{j,i}^{\mathrm{ii},q}$ are specified similarly to (7) based on the order of the entry instants of these aircraft:

$$\pi_{i,j}^{\mathrm{ii},q} = 1, \quad \pi_{j,i}^{\mathrm{ii},q} = 0, \text{ if } \mathtt{tIn}_i^q = t_{i,\mathrm{entry}}^{\mathrm{nom}} \le \mathtt{tIn}_j^q = t_{j,\mathrm{entry}}^{\mathrm{nom}},$$
$$\pi_{i,j}^{\mathrm{ii},q} = 0, \quad \pi_{j,i}^{\mathrm{ii},q} = 1, \text{ otherwise.}$$

In this case, the corresponding constraint among (22) on these variables may not be imposed. The values of the remaining variables are determined during the decision process depending on the fact of entering to/passing by the holding area by one or both aircraft.

For each ordered pair of aircraft i from the flow l' and j from the flow l'', and for each irregular point $q \in V_{l'} \cap V_{l''}$, constraints are imposed on the connection of the variables that determine the order of passing such a point:

$$-\mathtt{toHA}_i^q + \pi_{i,j}^{\mathrm{ii},q} \le \pi_{i,j}^{\mathrm{oi},q} \le \pi_{i,j}^{\mathrm{ii},q} + \mathtt{toHA}_i^q, \tag{23}$$
$$-\mathtt{toHA}_j^q + \pi_{i,j}^{\mathrm{ii},q} \le \pi_{i,j}^{\mathrm{io},q} \le \pi_{i,j}^{\mathrm{ii},q} + \mathtt{toHA}_j^q, \tag{24}$$
$$-(\mathtt{toHA}_i^q + \mathtt{toHA}_j^q) + \pi_{i,j}^{\mathrm{ii},q} \le \pi_{i,j}^{\mathrm{oo},q} \le \pi_{i,j}^{\mathrm{ii},q} + (\mathtt{toHA}_i^q + \mathtt{toHA}_j^q). \tag{25}$$

The meaning of these constraints is as follows. If both vessels enter the holding area at the point q ($\mathtt{toHA}_i^q = \mathtt{toHA}_j^q = 1$), then the variables $\pi_{i,j}^{\mathrm{ii},q}$, $\pi_{i,j}^{\mathrm{io},q}$, $\pi_{i,j}^{\mathrm{oi},q}$, $\pi_{i,j}^{\mathrm{oo},q}$ are not interconnected, since inequalities (23)–(25) become insubstantial.

If the ith aircraft does not enter the holding area at the point q ($\mathtt{toHA}_i^q = 0$), then constraint (23) takes the form $\pi_{i,j}^{\mathrm{ii},q} \le \pi_{i,j}^{\mathrm{oi},q} \le \pi_{i,j}^{\mathrm{ii},q}$, that is, $\pi_{i,j}^{\mathrm{oi},q} = \pi_{i,j}^{\mathrm{ii},q}$. Otherwise, ($\mathtt{toHA}_i^q = 1$), constraint (23) becomes insubstantial and does not connect the variables $\pi_{i,j}^{\mathrm{ii},q}$ and $\pi_{i,j}^{\mathrm{oi},q}$.

The same logic works for the variables $\pi_{i,j}^{\mathrm{io},q}$ and $\pi_{i,j}^{\mathrm{ii},q}$ if the jth aircraft enters or does not enter the holding area at the point q (see constraint (24)).

If both vessels do not go to the holding area at the point q ($\mathtt{toHA}_i^q = \mathtt{toHA}_j^q = 0$), then constraint (25) takes the form $\pi_{i,j}^{\mathrm{ii},q} \le \pi_{i,j}^{\mathrm{oo},q} \le \pi_{i,j}^{\mathrm{ii},q}$, that is, $\pi_{i,j}^{\mathrm{oo},q} = \pi_{i,j}^{\mathrm{ii},q}$. And due to constraints (23) and (24), one gets that the variables $\pi_{i,j}^{\mathrm{ii},q}$, $\pi_{i,j}^{\mathrm{oi},q}$, $\pi_{i,j}^{\mathrm{io},q}$, and $\pi_{i,j}^{\mathrm{oo},q}$ are equal. It corresponds the fact that for each aircraft, the instants $\mathtt{tIn}^q$ of entering the point q and $\mathtt{tOut}^q$ of leaving it coincide.

Constraints connected with safety. For each pair of aircraft i from the flow l' and j from the flow l'', $i \ne j$, constraints providing the safe passage of the irregular point $q \in V_{l'} \cap V_{l''}$ are imposed:

$$\mathtt{tIn}_i^q - \mathtt{tIn}_j^q + \tau_{\sigma_i,\sigma_j}^{\mathrm{safe},q} \cdot \pi_{i,j}^{\mathrm{ii},q} - \pi_{j,i}^{\mathrm{ii},q} \cdot \mathtt{M} - (2 - \zeta_i^q - \zeta_j^q) \cdot \mathtt{M} \le 0, \tag{26}$$
$$\mathtt{tIn}_i^q - \mathtt{tOut}_j^q + \tau_{\sigma_i,\sigma_j}^{\mathrm{safe},q} \cdot \pi_{i,j}^{\mathrm{io},q} - \pi_{j,i}^{\mathrm{io},q} \cdot \mathtt{M} - (2 - \zeta_i^q - \zeta_j^q) \cdot \mathtt{M} \le 0, \tag{27}$$
$$\mathtt{tOut}_i^q - \mathtt{tIn}_j^q + \tau_{\sigma_i,\sigma_j}^{\mathrm{safe},q} \cdot \pi_{i,j}^{\mathrm{oi},q} - \pi_{j,i}^{\mathrm{oi},q} \cdot \mathtt{M} - (2 - \zeta_i^q - \zeta_j^q) \cdot \mathtt{M} \le 0, \tag{28}$$
$$\mathtt{tOut}_i^q - \mathtt{tOut}_j^q + \tau_{\sigma_i,\sigma_j}^{\mathrm{safe},q} \cdot \pi_{i,j}^{\mathrm{oo},q} - \pi_{j,i}^{\mathrm{oo},q} \cdot \mathtt{M} - (2 - \zeta_i^q - \zeta_j^q) \cdot \mathtt{M} \le 0. \tag{29}$$

In these constraints, the same logic works as in constraints (8), however, the safety is checked for each case of entering to or passing by the holding area at the point q by one or both aircraft in the pair.

Due to constraints (21)–(25), some of the relations above may coincide, since the instants $\mathtt{tIn}^q$ and $\mathtt{tOut}^q$ and some of the variables $\pi_{i,j}^{\mathrm{ii},q}$, $\pi_{i,j}^{\mathrm{oi},q}$, $\pi_{i,j}^{\mathrm{io},q}$, and $\pi_{i,j}^{\mathrm{oo},q}$ may coincide.

4.4 Functional to be optimized

The following type of the individual penalty function is used to evaluate the optimality of the assignment of the instant $t_i = t_i^q$ of arrival of the ith aircraft from the flow l to the final point q of the scheme:

$$f_i(t_i, t_i^{\text{nom}}, \sigma_i) = \beta_{\sigma_i} \cdot \begin{cases} -K^- \cdot (t_i - t_i^{\text{nom}}), & \text{if } t_i < t_i^{\text{nom}}, \\ 0, & \text{if } t_i = t_i^{\text{nom}}, \\ K^+ \cdot (t_i - t_i^{\text{nom}}), & \text{if } t_i > t_i^{\text{nom}}. \end{cases} \tag{30}$$

Here, t_i^{nom} is the planned instant of arrival of the aircraft i at the final point q. The coefficient β_{σ_i} describes the preference for maneuvering of aircraft of one or another type. For example, usually ATCs prefer to change motion of light vessels rather than heavy ones.

By choosing the positive values of the coefficients K^- and K^+, one can regulate what penalize more: early or late arrivals.

The functional to be optimized has the form of the sum of the individual penalties for each aircraft:

$$F\big(\{t_i\}, \{t_i^{\text{nom}}, \sigma_i\}\big) = \sum_{i=1}^{N} \big(f_i(t_i, t_i^{\text{nom}}, \sigma_i) + \alpha \cdot \omega(i) \cdot g_i\big) \to \min, \tag{31}$$

where $\{t_i^{\text{nom}}\}$ is a set of the nominal instants of arrival of aircraft at the final point of the scheme, $\{t_i\}$ is a set of assigned instants of arrival of aircraft. For the formalization of the problem, it is important that the penalty functions f_i are convex piecewise linear. Despite of non-linearity of f_i, one can construct a linear programming problem having (possibly) more constraints and be equivalent to the original problem (see, for example, [19, 22]).

The value g_i is the number of interactions between the pilot of the ith aircraft and ATCs when passing non-nominal segments of the scheme, entering holding areas or changing velocity. The multiplier α is a weight coefficient of g_i scaling it to the value of f_i. The multiplier $\omega(i)$ is a weight coefficient determining the cost of interactions with a particular aircraft.

For each aircraft i from the flow l, the term g_i in functional (31) associated with the interactions of the pilot with an ATC has the form

$$g_i = \sum_{e \in E_l, \rho^e = 1} \xi_i^e + \sum_{q \in V_l^{\texttt{HA}}} \texttt{toHA}_i^q + \sum_{\substack{e=(q,r) \in E_l, \\ \texttt{vMin}^r < \texttt{vMax}^r}} \left(\xi_i^e \cdot \left(1 - \sum_{j=0}^{J} b_{i,j}^q \cdot b_{i,j}^r \right) \right). \tag{32}$$

Here, the first sum counts the number of ATC–pilot interactions related to the choice of the aircraft route, that is, the number of non-nominal segments of the scheme that the aircraft has passed and to which it must be directed by a dispatcher's direction. This sum considers only non-nominal segments of the flow route (whose $\rho^e = 1$) and those that are actually passed by the ith aircraft (whose $\xi_i^e = 1$).

The second term in (32) takes into account the number of ATC–pilot interactions directing the vessel into holding areas. Here, $V_l^{\texttt{HA}}$ is a subset of irregular points in the vertex set V_l of the flow l, to which the aircraft i belongs. Only those points contribute to the functional, at which the vessel entered the holding area, that is, for which $\texttt{toHA}_i^q = 1$.

Finally, the third term in (32) takes into account the number of ATC–pilot interactions associated with switching the aircraft velocity. More exactly, this term counts the number of velocity switches each aircraft performs in segments whose end point has a non-point constraint for possible velocity values. As it is discussed in Section 2, switches to such a unique velocity can be performed without interactions with ATCs.

Let, at first, consider the multiplier in the inner parentheses.

If the velocity change takes place on the segment $e = (q, r)$, then the indices j corresponding to the velocity values at the points q and r differ. That is, the indices j' and j'' differ, for which $b^q_{i,j'} = 1$ and $b^r_{i,j''} = 1$. Therefore, for all indices $j = 0, \dots, J$, one has $b^q_{i,j} \cdot b^r_{i,j} = 0$, and the sum inside the inner parentheses is equal to 0. The entire multiplier in the inner parentheses equals 1.

On the contrary, if there is no velocity change on the segment e, then there is an index j^* such that $b^q_{i,j^*} = b^r_{i,j^*} = 1$. For this index, the product $b^q_{i,j^*} \cdot b^r_{i,j^*}$ is equal to 1, for the others these products are equal to 0. Therefore, the sum in the inner parentheses is equal to 1, and the entire multiplier is equal to 0. As a sequence, such a segment does not contribute to the interactions counter.

The multiplier ξ^e_i ensures that only those segments e are counted, which are actually passed by the aircraft i.

Thus, the third term in (32) indeed takes into account the number of interactions associated with the velocity change of the aircraft i.

The definition of the third term in (32) contains a non-linearity in the form of a product of three binary variables ξ^e_i, $b^q_{i,j}$, and $b^r_{i,j}$, namely

$$\sum_{e=(q,r)} \left(\xi^e_i \cdot \left(1 - \sum_{j=0}^{J} b^q_{i,j} \cdot b^r_{i,j} \right) \right) = \sum_{e=(q,r)} \left(\xi^e_i - \xi^e_i \cdot \sum_{j=0}^{J} b^q_{i,j} \cdot b^r_{i,j} \right) =$$
$$= \sum_{e=(q,r)} \xi^e_i - \sum_{e=(q,r)} \sum_{j=0}^{J} \xi^e_i \cdot b^q_{i,j} \cdot b^r_{i,j}.$$

Each such a triple product of these variables can be linearized due to their binary nature. For each aircraft i from the flow l, for each segment $e = (q, r) \in E_l$, $\mathtt{vMin}^r < \mathtt{vMax}^r$, and for each index $j = 0 \dots J$, introduce a binary variable $B^e_{i,j} = \xi^e_i \cdot b^q_{i,j} \cdot b^r_{i,j}$. The fact that $B^e_{i,j}$ is a product of three binary variables can be represented by the following linear constraints:

$$B^e_{i,j} \le \xi^e_i, \quad B^e_{i,j} \le b^q_{i,j}, \quad B^e_{i,j} \le b^r_{i,j}, \quad \xi^e_i + b^q_{i,j} + b^r_{i,j} - 2 \le B^e_{i,j}. \tag{33}$$

Thus, with the new notations, the term g_i in (31) has the form

$$g_i = \sum_{e \in E_l, \rho^e = 1} \xi^e_i + \sum_{q \in V^{\mathtt{HA}}_l} \mathtt{toHA}^q_i + \Big(\sum_{\substack{e=(q,r)\in E_l,\\ \mathtt{vMin}^r < \mathtt{vMax}^r}} \xi^e_i - \sum_{\substack{e=(q,r)\in E_l,\\ \mathtt{vMin}^r < \mathtt{vMax}^r}} \sum_{j=0}^{J} B^e_{i,j} \Big). \tag{34}$$

In general, the obtained formalization is linear (linear optimized functional and linear constraints), and the resulting problem can be solved by mixed integer linear programming methods.

The solution obtained for the resulting MILP problem contains information about the assigned route of each aircraft as a set of segments to be passed by it, a set of control points to pass, and a set of instants of arrivals at the control points along the routes. Also, there is information about velocities of each aircraft when it passes each control point. These data together with passage times define instants of velocity switches. The delays of an aircraft in holding areas are defined by the corresponding variables `tIn` and `tOut`. Necessity of path alignments is defined by passing the corresponding segments in the scheme graph; instants to start the alignment can be computed on the basis of the aircraft velocity and passage times of the end points of the alignment segment. Thus, one can obtain a comprehensive set of directions to provide an aircraft's pilot with the optimal safe schedule.

5 Processing strategies

During numerical experiments, several approaches to processing an aircraft ensemble have been applied. Below, a brief description of them is given.

Whole processing. Under this approach, the entire ensemble of aircraft is used to construct a MILP-model to be passed to a solver. Its main problem is that its performance drops significantly as the ensemble size increases since the model size increases.

If there is some information (for example, obtained from an ATC or from some heuristic processing), it can be used as the initial guess for the used MILP-solver.

Sequential processing. Under this approach, ensemble processing begins with creating of a MILP-model for some small subensemble of the original ensemble and whole processing it. This subensemble consists of aircraft having the earliest arrival times. When it is processed, some schedule for it is obtained.

Further, one or several aircraft not processed yet are added into the ensemble, and a MILP-model is created for this wider ensemble. When this new model is sent to the solver, the schedule from the previous iteration is used as an initial guess. Potentially, this can speed up the solving the new problem.

Such a sequential expansion of the ensemble to be handled continues until the entire ensemble is processed.

Rolling window approach. In both previous approaches, eventually, the problem is solved for the entire ensemble with or without some additional information. If this solving finishes successfully, it gives the exact optimal solution of the problem. But often, it takes too long to solve this large MILP-problem.

However, also often, one does not need for the exact solution, but some suboptimal one close to the ideal is sufficient. So, some approach is suggested, which produces a suboptimal solution dealing with small subensembles at each its iteration.

The idea of the approach is quite simple. Take several aircraft with early arrivals. Solve the problem for them. Then, fix schedules for some of them, add some aircraft not processed yet, and solve the problem for the new subensemble considering also some additional constraints generating by the vessels from the fixed part.

When this new subensemble is processed, take some aircraft and fix them too. Repeat the process until all aircraft are considered.

These process has two parameters: k is the size of the rolling window, that is, the number of aircraft to be processed at each stage of the procedure, l is the window shift, that is, the number of aircraft to be fixed after solving the next subproblem and to be added to the subensemble for the next iteration of the procedure.

In some works, a similar approach is considered, but the rolling window takes into account not number of aircraft to be processed, but their arrival times. That is, the number of aircraft to be processed can change at different iterations of the procedure. Such an approach reflects that the merging problem should be solved in real time as some aircraft land or leave the airport zone and some come into it.

As a result of application of this approach, one obtains a suboptimal solution only. But the approach reflects the fact that it is unnecessary to try to change order of aircraft, whose arrivals are quite far form each other.

6 Numerical results

Statistical modeling was carried out to evaluate the performance of the proposed formalization of the model. The modeling has been performed using the rolling window approach. Attempts to use the whole and sequential processions totally failed due to unacceptable computational times.

In numerical experiments, the following values of the coefficients were used: $\beta_{\mathrm{L}} = 1.0$, $\beta_{\mathrm{M}} = 3.0$, $\beta_{\mathrm{H}} = 5.0$, $K^- = 2$, $K^+ = 1$, $\alpha_{\mathrm{L}} = 1$, $\alpha_{\mathrm{M}} = 3$, $\alpha_{\mathrm{H}} = 5$, $\omega = 10$ for all aircraft.

Numerical experiments has been conducted for the following combination of the values of the parameters k and l of the rolling window: $k = 5$, $l = 2, 3$; $k = 10$, $l = 2, 4, 5, 6$; $k = 15$, $l = 5$.

The processing time of each run of the MILP solver (for each MILP model created for each rolling window instance) was limited by 30 seconds. If the constraint is met, then the processing of the ensemble ceases and the ensemble is considered to be processed unsuccessfully. This limitation for each subensemble procession gives some limitation for procession of the entire ensemble. Such an approach is useful from the point of view of possible usage of the procedure in real-time work of an ATC complex to produce repeatedly recommendations to ATCs on the basis of varying actual situation in the air.

In the modeling, the scheme of the approach zone of Koltsovo aircraft was used (see Fig. 1).

Ensembles of 30, 40, and 50 aircraft with a density of 30 aircraft per hour, the uniform and some non-uniform (3% of light, 86% of medium, and 11% of heave aircraft) distributions of the types were considered. The non-uniform distribution of the types reflects some typical situation in Koltsovo airport when almost all aircraft are of the medium type. For application of the rolling windows approach, the aircraft in the ensemble are sorted ascending by their entry times.

The maximum size of a group of aircraft with the same instant of arrival at the final point is 4. The following the probabilities of the appearance of a group of vessels of each size were used: 1 – 90.5%, 2 – 7%, 3 – 2%, 4 – 0.5%. The probabilities of assigning an aircraft to each flow are equal.

Numerical experiments were performed on a node of "Uran" supercomputer at the Krasovskii Institute of Mathematics and Mechanics, Yekaterinburg, Russia. The node is equipped with 18-core processor Intel Xeon Gold 6254 CPU @ 3.10GHZ. The computations was made in multithread regime (6 threads).

To solve a MILP problem, the optimization library `Gurobi` [7] was used.

For statistical modeling on the "Uran" supercomputer, 20 hours of real time were allocated. It was supposed to collect statistics at 300 successfully processed ensembles. The average processing time of the entire ensemble and the number of unsuccessful launches are collected.

Numerical results for case of the uniform distribution of the types is presented in Table 2.

One can see that for the ensembles of 50 aircraft when $k = 5$, $l = 2, 3$, it was not possible to successfully compute 300 ensembles, so the statistical data is not given. For the cases of the ensembles of 30 and 40 aircraft, approximately every second aircraft ensemble could not be processed successfully.

The modelling parameters, $k+l$	30 aircraft		40 aircraft		50 aircraft	
	t^{avg}	The number of unsuccessful runs	t^{avg}	The number of unsuccessful runs	t^{avg}	The number of unsuccessful runs
$5+2$	3.71	3664 (92.43 %)	3.85	20136 (95.53 %)	-	-
$5+3$	2.10	7304 (96.05 %)	2.81	30413 (99.02 %)	-	-
$10+2$	13.56	53 (15.01 %)	20.52	92 (23.47 %)	27.15	137 (31.35 %)
$10+4$	7.19	98 (24.62 %)	11.32	209 (41.06 %)	13.40	368 (55.09 %)
$10+5$	5.73	165 (35.48 %)	8.05	413 (57.92 %)	10.79	620 (67.39 %)
$10+6$	5.85	229 (43.28 %)	6.74	582 (65.99 %)	9.58	909 (75.19 %)
$15+5$	26.71	172 (36.44 %)	40.62	301 (50.08 %)	57.87	413 (57.92 %)

Table 2: The numerical results for the case of the uniform distribution of the aircraft types. The average process time t^{avg} in seconds.

The results can be considered acceptable for those values of the parameters, for which the number of unsuccessful runs does not exceed 10–15% of the total number of runs. Unfortunately, in the case of the uniform distribution of the types, this is true only for ensembles of 30 aircraft and the values $k = 10$, $l = 2$.

Note that almost for all combinations of the parameters (the size of a processed ensemble and values of the parameters k and l), the average process time of an ensemble is less than acceptable 30 seconds. For ensembles of 40 and 50 aircraft and the values of the parameters $k = 15$, $l = 5$ the average process time is greater than 30 seconds.

There are several reasons of such a poor performance. At first, one of the reasons is the uniform distribution of types, which leads to strong heterogeneity of an ensemble. From the point of view of functional (31), rearrangement in the order of two aircraft of the same type does not change the value of functional (31). Since an ensemble is strong heterogenous, the possible number of possible significant permutations is large. So, the methods of solving MILP problems all such permutations of aircraft in an ensemble.

Another reason is that sometimes, fixing an initial part of the ensemble produces some constraints those make the MILP-problem for next subensemble inconsistent. This reason is substantial for large ensembles. In such a situation, the optimal solution is connected with

The modelling parameters, $k+l$	30 aircraft		40 aircraft		50 aircraft	
	t^{avg}	The number of unsuccessful runs	t^{avg}	The number of unsuccessful runs	t^{avg}	The number of unsuccessful runs
$5+2$	3.73	1575 (84.00 %)	4.13	3794 (92.67 %)	5.72	23478 (98.74 %)
$5+3$	2.01	4297 (93.47 %)	2.66	8539 (96.61 %)	-	-
$10+2$	12.33	9 (2.91 %)	17.64	29 (8.81 %)	23.60	36 (10.71 %)
$10+4$	6.28	40 (11.76 %)	9.91	84 (21.88 %)	11.79	131 (30.39 %)
$10+5$	4.80	63 (17.36 %)	7.03	141 (31.97 %)	9.64	214 (41.63 %)
$10+6$	4.96	35 (10.45 %)	6.23	247 (45.16 %)	8.39	376 (55.62 %)
$15+5$	20.37	53 (15.01 %)	31.15	74 (19.79 %)	43.82	119 (28.40 %)

Table 3: The numerical results for the case of the non-uniform distribution of the aircraft types. The average process time t^{avg} in seconds.

making the arrivals of the initial part of the ensemble earlier. But this does not occur when this initial part is considered independently of the following aircraft.

The numerical results in the case of the non-uniform distribution of the types are presented in Table 3. One can see that the number of unsuccessful runs is less for all combinations of the parameters than in the case of the uniform distribution of the types. For ensembles of 30, 40, and 50 aircraft and the values of the parameters $k = 5$ and $l = 2, 3$, the very negative results are saved, however, during the modeling, it was possible to get results for ensembles of 50 aircraft and the values of the parameters $k = 5$, $l = 2$.

For the ensembles of 40 and 50 aircraft and the values of the parameters $k = 15$, $l = 5$, the average process time is still greater than acceptable 30 seconds.

The reason of the improvements of the numerical results in the case of the non-uniform distribution of the types is the small number of possible permutations of aircraft, since an ensemble is quite homogeneous. As it mentioned before, due to specific work of methods in MILP-solver, a significant number of permutations of aircraft that do not improve the value of functional to be minimized are not considered. Thus, it is required to enumerate fewer permutation cases.

7 Conclusion

In the work, a problem of multi-stage multi-route aircraft flows merging in an airport approach zone is considered. The problem uses a detailed description of the structure of the zone airspace. Thus, it is assumed that aircraft might pass sequentially several merge points and possibly have different routes leading from the entry point of an aircraft flow to the final point of the scheme. The presence of holding areas, path alignments, and point-merge schemes, as well as chosen linear segments, on the scheme of an approach zones allows change aircraft order. Also, it is assumed that pilots need to interact with ATCs when using alternative routes, delaying in holding areas or changing aircraft velocity.

The problem of of multi-stage multi-route aircraft flows merging is reduced to a mixed integer linear programming (MILP) problem. In the paper, a methodology of constructing a MILP-problem is considered.

Results of a statistical modelling with the real data presented in the paper show the acceptable performance of the suggested procedure in some cases of the modelling parameters.

In this paper, an offline approach to construct a safe and optimal schedule is presented, that is, *a priori*, there is some planned arrival schedule and it is necessary to optimize it. For real online application of the suggested procedure, one needs to take into account current positions of aircraft inside the scheme when constructing a MILP-model. Such a consideration is one of the directions of the future work.

To increase the computation efficiency of the suggested procedure, other methods (exact or approximate) of solving the problem under consideration can be applied. However, the author hopes that considering the online situation of aircraft flows merging will lead to decreasing the size of the problem and increasing the computation efficiency.

8 Acknowledgments

The author would like to thank his scientific supervisor Sergey S. Kumkov for continuous support, significant comments and important advice during the research.

The author is also grateful to Andrey V. Vasiliev (NITA, LCC [14]) for consultations on work practice of air traffic controllers, the numerical data of aircraft motion, and the suggested models of aircraft flows merging.

Finally, the author would like to acknowledge the assistance of Dmitriy M. Kryukov, who has described aspects and nuances of work of air traffic controllers in the approach zone of Koltsovo airport (Yekaterinburg, Russia).

References

[1] J. E. Beasley, M. Krishnamoorthy, Y. M. Sharaiha, and D. Abramson. Scheduling aircraft landings — the static case. *Transportation Science*, 34(2):180–197, 2000.

[2] M. Ben Messaoud. A thorough review of aircraft landing operation from practical and theoretical standpoints at an airport which may include a single or multiple runways. *Applied Soft Computing*, 98:106853, 2021.

[3] J. A. Bennell, M. Mesgarpour, and C. N. Potts. Airport runway scheduling. *4OR — Q. J. Oper. Res.*, 9(2):115–138, 2011.

[4] A. D'Ariano, P. D'Urgolo, D. Pacciarelli, and M. Pranzo. Optimal sequencing of aircrafts take-off and landing at a busy airport. In *13th International IEEE Conference on Intelligent Transportation Systems*, pages 1569–1574. IEEE, 2010.

[5] A. D'Ariano, D. Pacciarelli, M. Pistelli, and M. Pranzo. Real-time scheduling of aircraft arrivals and departures in a terminal maneuvering area. *Networks*, 65(3):212–227, 2015.

[6] A. D'Ariano, M. Pistelli, and D. Pacciarelli. Aircraft retiming and rerouting in vicinity of airports. *IET Intelligent Transport Systems*, 6(4):433–443, 2012.

[7] Gurobi Optimization, LLC. The offical website. https://www.gurobi.com.

[8] S. Ikli, C. Mancel, M. Mongeau, X. Olive, and E. Rachelson. The aircraft runway scheduling problem: A survey. *Computers and Operations Research*, 132:105336, 2021.

[9] F. Jiang and Z. Zhang. Optimal sequencing of arrival flights at metroplex airports: a study on shared waypoints based on path selection and rolling horizon control. *Aerospace*, 10(10):881, 2023.

[10] A. Kwasiborska and J. Skorupski. Assessment of the method of merging landing aircraft streams in the context of fuel consumption in the airspace. *Sustainability*, 13(22):12859, 2021.

[11] J. Ma, D. Delahaye, M. Sbihi, P. Scala, and M. A. Mujica Mota. Integrated optimization of terminal maneuvering area and airport at the macroscopic level. *Transportation Research Part C: Emerging Technologies*, 98:338–357, 2019.

[12] A. Mascis and D. Pacciarelli. Job-shop scheduling with blocking and no-wait constraints. *European Journal of Operational Research*, 143(3):498–517, 2002.

[13] A. V. Mikhailov and S. S. Kumkov. Air-traffic controller and autopilot joint control of aircraft as mixed integer quadratic problem. *Journal of Physics: Conference Series*, 2514(1):012013, 2023.

[14] NITA, LCC. The official website. https://www.nita.ru/en/main/.

[15] M. Samà, A. D'Ariano, F. Corman, and D. Pacciarelli. Metaheuristics for efficient aircraft scheduling and re-routing at busy terminal control areas. *Transportation Research Part C: Emerging Technologies*, 80:485–511, 2017.

[16] M. Samà, A. D'Ariano, P. D'Ariano, and D. Pacciarelli. Comparing centralized and rolling horizon approaches for optimal aircraft traffic control in terminal areas. *Transportation Research Record: Journal of the Transportation Research Board*, 2449(1):45–52, 2014.

[17] A. H. Sarris and M. Athans. An algorithm for terminal air traffic control. Technical Report ESL-P-466, Massachusetts Institute of Technology, 1971.

[18] R. W. Simpson. Scheduling and routing models for airline systems. Technical Report FTL-R68-3, Flight Transportation Laboratory, Massachusetts Institute of Technology, 1969.

[19] M.J. Soomer and G.J. Franx. Scheduling aircraft landings using airlines' preferences. *European Journal of Operational Research*, 190(1):277–291, 2008.

[20] A. A. Spiridonov. Safe merging of aircraft flows in multi-route schemes. Submitted to *Journal of Computational Science*, 2024.

[21] A. A. Spiridonov and S. S. Kumkov. Cascade merge of aircraft flows as mixed integer linear programming problem. In Y. Kochetov, A. Eremeev, O. Khamisov, and A. Rettieva, editors, *Mathematical Optimization Theory and Operations Research: Recent Trends*, pages 132–146. Springer, Cham, 2022.

[22] A. A. Spiridonov and S. S. Kumkov. Keeping order of vessels in problem of safe merging aircraft flows. *Vestnik Udmurtskogo Universiteta. Matematika. Mekhanika. Komp'yuternye Nauki*, 32(3):433–446, 2022.

[23] A. A. Spiridonov and S. S. Kumkov. Cascade aircraft flows merging with multiple order changes. *Journal of Physics: Conference Series*, 2514(1):012018, 2023.

[24] G. S. Veresnikov, N. A. Egorov, E. L. Kulida, and V. G. Lebedev. Methods for solving of the aircraft landing problem. i. exact solution methods. *Automation and Remote Control*, 80(7):1317–1334, 2019.

[25] G. S. Veresnikov, N. A. Egorov, E. L. Kulida, and V. G. Lebedev. Methods for solving of the aircraft landing problem. ii. approximate solution methods. *Automation and Remote Control*, 80(8):1502–1518, 2019.

[26] M. S. Vié, N. Zufferey, and R. Leus. Aircraft landing planning: past, present and future. In *Proceedings of the 19th annual congress of the french operations research society*, 2018.

[27] L. You, J. Zhang, C. Yang, and R. Hu. Heuristic algorithm for aircraft arrival scheduling based on a point merge system. In H. Wei, H. Wang, L. Zhang, Y. An, and X. Zhao, editors, *CICTP 2020: Transportation Evolution Impacting Future Mobility*, pages 125–138. American Society of Civil Engineers, 2020.

[28] A. Zulkifli, N. A. A. Aziz, N. H. A. Aziz, Z. Ibrahim, and N. Mokhtar. Review on computational techniques in solving aircraft landing problem. In M. Sugisaka, Y. Jia, T. Ito, and J.-J. Lee, editors, *Proceedings of the 2018 International Conference on Artificial Life and Robotics, Oita, Japan*, pages 128–131, 2018.