

The Air Traffic Flow Management Problem: An Integer Optimization Approach

Dimitris Bertsimas¹, Guglielmo Lulli², and Amedeo Odoni³

¹ Sloan School of Management and Operations Research Center, M.I.T.
dbertsim@mit.edu

² Dept. of Informatics, Systems and Communication, University of Milano “Bicocca”
guglielmo.lulli@disco.unimib.it

³ Dept. of Aeronautics and Astronautics and Operations Research Center, M.I.T.
arodoni@mit.edu

Abstract. In this paper, we present a new Integer Program (IP) for the Air Traffic Flow Management (ATFM) problem. The model we propose provides a complete representation of all the phases of each flights, i.e., the phase of taking-off, of cruising and of landing; suggesting all the actions to be implemented to achieve the goal of safe, efficient, and expeditious aircraft movement. The distinctive feature of the model is that it allows rerouting decisions. These decisions are formulated by means of “local” conditions, which allow us to represent such decisions in a very compact way by only introducing new constraints. Moreover, to strengthen the polyhedral structure of the underlying relaxation, we also present three classes of valid inequalities.

We report short computational times (less than 15 minutes) on instances of the size of the US air traffic control system that make it realistic that our approach can be used as the main engine of managing air traffic in the US.

1 Introduction

The continuous growth of the air transportation industry have put an enormous strain on the aviation system. Congestion phenomena are persistent and arise almost on a daily basis as a consequence of bad weather conditions which cause sudden capacity reductions. In the year 2000, approximately one out of every four flights in the United States was delayed or canceled, [6]. The resulting delays have a significant economic impact. The Air Transport Association has estimated that system delays drove an estimated \$5.9 billion in direct operating costs for United States airlines in 2005. Similar figures have been shown by European airlines.

As a result, air traffic flow management (ATFM) has become increasingly crucial. ATFM attempts to prevent local demand-capacity imbalances by adjusting the flows of aircraft on a national or regional basis. Until now, the ATFM have been mainly focusing on airports’ congestion. On this subject, the most popular approach, by far, has been the allocation of ground delays to departing flights, i.e., postponing their departure time. From the paper by Odoni [9], who was the first to formalize this problem, a plethora of models and algorithms have been developed to detect optimal strategies to assign ground delays to flights (see [1] and [6]).

However, it has become increasingly evident that very significant delays and system throughput degradations have arisen from en-route airspace problems and limitations. The problem posed by the en-route sector capacity constraints is persistent and may take at least one more decade to resolve [4]. One of the implications of the simultaneous presence of airport and en-route airspace constraints is that devising good strategies is a much more complicated task. Any mathematical model developed for this purpose has to consider a true network of capacitated elements, en-route sectors and airports [8]. Moreover, a larger set of options to resolve congestion is available: ground holding, airborne holding, miles-in-tails and rerouting, i.e., the possibility of reroute a flight on a different flight path if the current route passes through a region that unexpectedly becomes congested.

As opposed to the airport congestion case, the research literature dealing with en-route congestion is quite sparse. One of the first attempts to include in the ATFM problem en-route capacity restrictions was by Helme [5], who proposed a multi-commodity minimum-cost flow on a time-space network to assign airborne and ground delay to aggregate flow of flights, commodities of the network flow model. While the formulation of this model is straightforward and easy to understand, its computational performance was rather weak. Lindsay et al.[7] formulated a disaggregate deterministic 0-1 integer programming models for deciding ground and airborne holding of individual flights in presence of both airport and airspace capacity constraints. Bertsimas and Stock [2] presented a deterministic 0-1 IP model to solve a similar problem. The model decides on the departure time and sector occupancy time of each aircraft. The model enables very efficient computation of optimal solutions, since several of the constraints provide facets of the convex hull of solutions. However this model, as well as those cited above, does not consider rerouting as an option. It assumes that the flight path is known in advance and is fixed.

To the best of our knowledge the only work which considers rerouting, at a least at a macroscopic level, is the work by Bertsimas and Stock Patterson [3]. They presented a dynamic, multi-commodity, integer network-flow model. The model addressed routing as well as scheduling decisions, but it did not provide computational performances aligned with the dimensions of real instances.

Modeling rerouting decisions has posed one of the greatest challenges in this field of research. Our goal is to combine the model “flexibility” in terms of range of decisions of model presented in [3] with the shown mathematical properties of the model presented in [2], so that we are able to solve efficiently sized problems. Herein, we present a mathematical model for the ATFM which includes all the possible options to resolve air congestion, including rerouting. The scope of the model is to suggest the time of departure, the route, the time required to cross each sector and the time of arrivals taking into account the capacity of all sectors and airports. The main feature of the model is the formulation of rerouting decisions in a very compact way. With respect to previous models, the methodology we presented does not require any additional variables, but it only introduces new constraints. These constraints implement local routing conditions that are sufficient for the purpose of the model. To strengthen the polyhedral structure of the underlying relaxation, we also present three classes of valid inequalities.

The paper is organized as follows: In Section 2, we present the mathematical model for the ATFM problem with rerouting, and three classes of valid inequalities as well. The computational experience is reported in Section 3. Finally, Section 4 contains conclusions and indications for future research.

2 The Mathematical Model

The mathematical model we present here, is intended to determine how to adjust the release time of each flight into the system (time of departure), how to control its flight speed once in the air and how to reroute it in case of sectors' congestion along the preferred path. As an underlying model we consider the model proposed by Bertsimas and Stock [2].

Any origin-destination route is represented as a sequence of sectors flown by an aircraft. In ATFM models which do not include rerouting as an option, the sequence of sectors to be flown is pre-determined. To contemplate rerouting in the mathematical model the set of possible sectors that might be flown has to be enlarged.

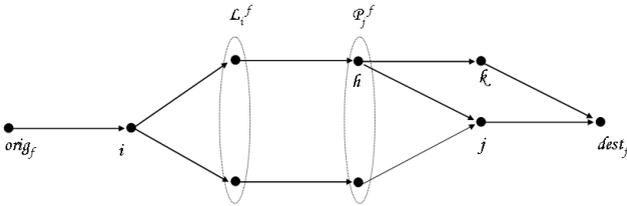


Fig. 1. Given a flight f , the set \mathcal{L}_i^f of sectors that follow sector i , and the set of sectors \mathcal{P}_i^f that precede sector j

A key element of the proposed model is the definition of routes. The origin-destination routes can be represented by digraphs. The set of nodes of the digraph (\mathcal{S}_f) represents the set of capacitated elements of the airspace, e.g., airports and sectors. The set of arcs defines the sequence relations. There is an arc from a node i to node j if i and j are contiguous sectors and sector j can be flown soon after sector i . In Figure 1, three different routes between the airport of origin and of destination are reported. Within the ATFM framework, we may suppose, without loss of generality, that the digraph of $o-d$ routes is acyclic. This allows us to equip the set of sectors with a binary relation, and hence envision the set of possible routes within the framework of so-called partially ordered set (poset). The airport of departure and arrival are the minimum and the maximum elements of the poset respectively. The set of possible routes between the $o-d$ pair corresponds to the set of maximal chains of the poset. To impose that each flight follows exactly one route we use local conditions, that can be simply stated as follows:

- to fly a sector any aircraft has first to fly one of the previous sectors for at least a number of time periods equal to their sector flight time;

or equivalently,

- if an aircraft is flying a sector, for at least a number of time periods equal to its flight time, then immediately after it will fly only one of the subsequent sectors.

To formally describe these routing conditions we introduce the following additional notation. For each sector i ($\in \mathcal{S}_f$) the subset of sectors which follow i is denoted by $\mathcal{L}_i^f \subset \mathcal{S}_f$. Analogously the subset of sectors that precede i is denoted by $\mathcal{P}_i^f \subset \mathcal{S}_f$ (see Figure 1).

In what follows, we call *forks* all the sectors followed by more than one sector, e.g., Sector i and Sector h in Figure 1, while those sectors preceded by more than one sector are called *joints*, Sector j in the same figure.

2.1 The Mathematical Formulation

The model's formulation requires definition of the following notation:

- $\mathcal{K} \equiv$ set of airports,
- $\mathcal{S} \equiv$ set of sectors,
- $\mathcal{S}^f \subseteq \mathcal{S} \equiv$ set of sectors that can be flown by flight f ,
- $\mathcal{F} \equiv$ set of flights,
- $\mathcal{T} \equiv$ set of time periods,
- $\mathcal{C} \equiv$ set of pairs of flights that are continued,
- $\mathcal{P}_i^f \equiv$ set of sector i 's subsequent sectors,
- $\mathcal{L}_i^f \equiv$ set of sector i 's previous sectors,
- $D_k(t) \equiv$ departure capacity of airport k at time t ,
- $A_k(t) \equiv$ arrival capacity of airport k at time t ,
- $S_j(t) \equiv$ capacity of sector j at time t ,
- $d_f \equiv$ scheduled departure time of flight f ,
- $a_f \equiv$ scheduled arrival time of flight f ,
- $s_f \equiv$ turnaround time of an airplane after flight f ,
- $orig_f \equiv$ airport of departure of flight f ,
- $dest_f \equiv$ airport of arrival of flight f ,
- $l_{fj} \equiv$ number of time units that flight f must spend in sector j ,
- $T_j^f = [\underline{T}_j^f, \bar{T}_j^f] \equiv$ set of feasible time periods for flight f to arrive in sector j ,
- $\underline{T}_j^f \equiv$ first time period in the set T_j^f ,
- $\bar{T}_j^f \equiv$ last time period in the set T_j^f .

The Decision Variables. As mentioned above, the model herein presented is based on the Bertsimas-Stock model [2] and we use the same decision variables.

$$w_{j,t}^f = \begin{cases} 1, & \text{if flight } f \text{ arrives at sector } j \text{ by time } t, \\ 0, & \text{otherwise.} \end{cases}$$

This definition of the decision variables ($w_{j,t}^f$) using “by” instead of “at” is critical to the understanding of the formulation. If flight f arrives at time t at sector j then both variable of time period t and subsequent ones will be set to 1 (i.e., $w_{j,\tau}^f = 1 \ \forall \tau \geq t$).

The Objective Function. As in most other ATFM models in the literature, the model we propose minimizes a cost function which is a combination of both airborne-holding delay (AH) and ground holding delay (GH), of the form $\alpha \cdot AH + GH$ with $\alpha > 1$. For convenience, we re-write the objective function as $\alpha \cdot TD - (\alpha - 1) \cdot GH$, being $TD (= AH + GH)$ the total delay.

To ensure equity among flights, we include in the objective function cost coefficients that are a super-linear function of the tardiness of a flight of the form $(t - a_f)^{1+\epsilon}$, with ϵ close to zero. This will favour the assignment of a moderate amount of total delay to each of two flights rather than the assignment of a small amount to one and a large amount to the other.

For each flight f and for each time period t , we define the following cost coefficients:

$$\begin{aligned} c_{td}^f(t) &= (t - a_f)^{1+\epsilon} \equiv \text{total cost of delaying flight } f \text{ for } (t - a_f) \text{ unit of time,} \\ c_g^f(t) &= (\alpha - 1)(t - d_f)^{1+\epsilon} \equiv \text{cost reduction for holding flight } f \text{ on the ground} \\ &\quad \text{for } (t - d_f) \text{ unit of time.} \end{aligned}$$

In view of the description above, the objective function is as follows:

$$\text{Min} \sum_{f \in \mathcal{F}} \left(\sum_{t \in T_{dest_f}^f} c_{td}^f(t) \cdot (w_{dest_f,t}^f - w_{dest_f,t-1}^f) - \sum_{t \in T_{orig_f}^f} c_g^f(t) \cdot (w_{orig_f,t}^f - w_{orig_f,t-1}^f) \right)$$

The Constraints

$$\sum_{f \in \mathcal{F}: orig_f = k} (w_{k,t}^f - w_{k,t-1}^f) \leq D_k(t) \quad \forall k \in \mathcal{K}, \ t \in \mathcal{T}. \quad (1)$$

$$\sum_{f \in \mathcal{F}: dest_f = k} (w_{k,t}^f - w_{k,t-1}^f) \leq A_k(t) \quad \forall k \in \mathcal{K}, \ t \in \mathcal{T}. \quad (2)$$

$$\sum_{f \in \mathcal{F}: j \in \mathcal{S}_f} (w_{j,t}^f - \sum_{j' \in \mathcal{L}_i^f} w_{j',t}^f) \leq S_j(t) \quad \forall j \in \mathcal{S}, \ t \in \mathcal{T}. \quad (3)$$

$$w_{j,t}^f \leq \sum_{j' \in \mathcal{P}_j^f} w_{j',t-l_{fj'}}^f \quad \forall f \in \mathcal{F}, \ t \in T_j^f, \ j \in \mathcal{S}^f : j \neq orig_f. \quad (4)$$

$$w_{j,\bar{T}_j^f}^f \leq \sum_{j' \in \mathcal{L}_j^f} w_{j',\bar{T}_{j'}^f} \quad \forall f \in \mathcal{F}, j \in \mathcal{S}^f : j \neq \text{dest}_f. \quad (5)$$

$$\sum_{j' \in \mathcal{L}_j^f} w_{j',\bar{T}_{j'}^f} \leq 1 \quad \forall f \in \mathcal{F}, j \in \mathcal{S}^f : j \neq \text{dest}_f. \quad (6)$$

$$w_{\text{orig}_f,t}^f - w_{\text{dest}_{f'},t-s_f}^{f'} \leq 0 \quad \forall (f, f') \in \mathcal{C}, \forall t \in T_k^f. \quad (7)$$

$$w_{j,t-1}^f - w_{j,t}^f \leq 0 \quad \forall f \in \mathcal{F}, j \in \mathcal{S}^f, t \in T_j^f. \quad (8)$$

$$w_{j,t}^f \in \{0, 1\} \quad \forall f \in \mathcal{F}, j \in \mathcal{S}^f, t \in T_j^f. \quad (9)$$

The first three sets of constraints take into account the capacities of various aspects of the system. Constraints (1) ensure that the number of flights which may take off from airport k at time t , will not exceed the departure capacity of airport k at time t . Likewise, Constraints (2) ensure that the number of flights which may arrive at airport k at time t , will not exceed the arrival capacity of airport k at time t . Finally, Constraints (3) ensure that the sum of all flights which may feasibly be in Sector j at time t will not exceed the capacity of Sector j at time t . This difference gives the flights that are in Sector j at time t , since the first term will be 1 if Flight f has arrived in sector j by time t and the second term will be 1 if flight f has arrived at one of the next sectors by time t . So, the only flights that will contribute a value of 1 to this sum are those flights that have arrived at j and have not yet departed from j by time t . Constraints (4), (5) and (6) represent connectivity between sectors. They stipulate that a flight can not arrive at Sector j by time t if it has not arrived to one of the previous sectors by time $t - l_{fj'}$. In other words, a flight cannot enter the next sector on its path until it has spent $l_{fj'}$ time units (the minimum possible) traveling through one of the previous sectors in its current path. Moreover, Constraints (5) and (6) state that a flight will certainly arrive to one of the subsequent sectors. Constraints (7) represent connectivity between airports. They handle the cases in which a flight is continued, i.e., the flight's aircraft is scheduled to perform a later flight within some time interval. We will call the first flight f' and the following flight f . Constraints (8) represent connectivity in time. Thus, if a flight has arrived by time \tilde{t} , then $w_{j,t}^f$ has to have a value of 1 for all later time periods ($t \geq \tilde{t}$).

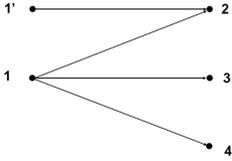
In what follows, we present three classes of valid inequalities with the scope of strengthening the formulation.

Proposition 1: *If Sector j is a fork, then constraints*

$$w_{j,t}^f \geq \sum_{j' \in \mathcal{L}_j^f : |\mathcal{P}_{j'}^f| = 1} w_{j',t+l_{fj'}}^f \quad \forall f \in \mathcal{F}, t \in T_j^f.$$

are valid inequalities for the set of feasible solutions of ATFM.

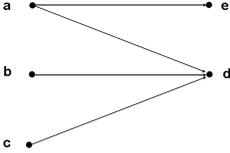
The inequalities of Proposition 1 state that if a flight f has not crossed Sector j by time t ($w_{j,t}^f = 0$), it will not cross any of the subsequent sectors by time $t + l_{fj'}$ unless these subsequent sectors can be reached from elsewhere, as in the



$$\text{cons.}(4) \begin{cases} w_{2,t} \leq w_{1,t-l} + w_{1',t-l} \\ w_{3,t} \leq w_{1,t-l} \\ w_{4,t} \leq w_{1,t-l} \end{cases}$$

$$\text{v.i.1 } w_{3,t} + w_{4,t} \leq w_{1,t-l}$$

Fig. 2. Valid inequality for a fork node



$$\text{cons.}(5) \begin{cases} w_{a,T} \leq w_{d,T} + w_{e,T} \\ w_{b,T} \leq w_{d,T} \\ w_{c,T} \leq w_{d,T} \end{cases}$$

$$\text{v.i.2 } w_{b,T} + w_{c,T} \leq w_{d,T}$$

Fig. 3. Valid inequality for a joint sector

case of *Sector 2* in Figure 2. The fork inequalities hold by Constraints (5) and (8) in case $w_{j,t}^f = 1$ and by Constraints (4) in case $w_{j,t}^f = 0$.

Conditions given above can be extended to the case of a joint sector. If a flight f do not cross Sector j then it have not crossed any of the previous sectors unless these sectors are also adjacent to other sectors. Hence, let us restrict the attention to sectors which are only adjacent to the joint sector among all the previous ones, e.g., sectors b and c of the example in Figure 3.

Proposition 2: *If Sector j is a joint, then constraints*

$$\sum_{j' \in \mathcal{P}_j^f : |\mathcal{L}_{j'}^f| = 1} w_{j', \bar{T}_{j'}^f} \leq w_{j, \bar{T}_j^f} \quad \forall f \in \mathcal{F}.$$

are valid inequalities for the set of feasible solutions of ATFM.

The joint inequalities hold by Constraints (5) if $w_{j,t}^f = 0$ and by Constraints (4) if $w_{j,t}^f = 1$.

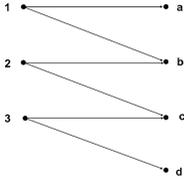
The network of possible routes, represented by an acyclic graph, naturally defines a preorder relation on the set of sectors. Each o - d pair corresponds to a chain of the poset and the Proposition in the sequel immediately follows:

Proposition 3: *If \mathcal{A} is an antichain for the ordered set defined on \mathcal{S}^f , then constraint*

$$\sum_{j \in \mathcal{A}} w_{j, \bar{T}_j^f} \leq 1.$$

is a valid inequality for the set of feasible solutions of ATFM (named, antichain inequality).

These conditions state that each flight follows exactly one route.



$$\text{cons. (6)} \quad \begin{cases} w_{a,T} + w_{b,T} \leq 1 \\ w_{b,T} + w_{c,T} \leq 1 \\ w_{c,T} + w_{d,T} \leq 1 \end{cases}$$

a.i. $w_{a,T} + w_{b,T} + w_{c,T} + w_{d,T} \leq 1$

Fig. 4. Antichain inequality

Proposition 4: *If Sector i is a fork and all the subsequent sectors can be reached only from Sector i , i.e., $\{j \in \mathcal{L}_i^f : |\mathcal{P}_j^f| = 1\} = \mathcal{L}_i^f$, then Constraints (5) are equalities.*

3 Computational Experience

In this section, we present the computational experience on the mathematical model presented in §2.1, including the valid inequalities given in Proposition 1 - Proposition 4. We consider randomly generated instances whose dimension is comparable to realistic ones. In particular, we consider two sets of instances. The first one represents the ATFM problem at a regional level, e.g., east-coast or mid-west US, while the second set, of larger instances, is more representative of the Nation wide problem. The size of the instance depends on the time horizon, the time discretization period, the number of sectors and airports and the demand at each airport. By changing one or all the parameters above, we generate different size instances.

The airspace is divided into equal size sectors, forming a grid. We also suppose that the minimum amount of time to fly a sector is the same for all the flights and for all the sectors. In order to generate instances which are consistent with the hub-and-spoke operations we cluster airports into hubs and regional airports. There are no flights connecting two regional airports, i.e., regional airports do not have direct connections but they are connected to hubs. For each airport, the demand of flights is randomly generated, drawn from a uniform distribution. Again with the sake of being more adherent to real operations, we generate hubs' demands considering both peak and off-peak periods. The average value of the demand for peak (off-peak) periods is set equal to 15 (8) flights per period. The nominal capacity of sectors and airports (capacity under good weather conditions) is set to values which allow to accommodate all the air traffic without incurring in too much air congestion.

To enforce sectors congestion, we suppose a capacity reduction of some sectors. The reduction of capacity affects 3 sectors at a time, for 5 consecutive time periods. Afterwards, three contiguous sectors experience capacity reduction for other 5 consecutive time periods, and so on. In this way, we “simulate” the effect of a bad weather front which move along a certain direction. We also consider instances with larger weather front and with different speed (number of time periods it stays in the sectors before moving forward), without affecting

the computational performance of the model. Herein we do not report all the computational results for the sake of brevity.

One of the key elements of our model is the set \mathcal{S}^f of sectors that can be flown by flight f . By default, all the sector on the shortest $o-d$ route of flight f are included in \mathcal{S}^f . If one or more of the sectors on the shortest route is congested, additional sectors, those contiguous, are included in the set \mathcal{S}^f . For this purpose, sectors are considered congested if their demand exceeds 80% of the capacity. The number of forks in a $o-d$ pair gives a lower estimate of the number of possible routes between the origin and the destination. On average, the number of forks is about 3, meaning that on average we have at least 3 routes between each $o-d$ pair, even though it can be much larger. In a case with 20 forks, we counted 121 $o-d$ routes.

3.1 Regional Size Instances

The computational results for “regional” instances are reported here. These instances include 20 airports, 10 of which are hubs, and 113 sectors which correspond to about one third of the NAS airspace. We consider a five-hour time horizon subdivided into 20 15-minute time units. All the instances manage roughly 3000 flights. The nominal capacity (capacity under good weather conditions) of sectors is set to 71 flights per period. Five sets of instances have been considered, each with a different percentage of flight connections, as reported in the first column of Table 1 (% of Conn.s). For instance, 50 indicates that half of the flights have a flight connection. In the first column, it is also reported between parenthesis the number of flights considered in the instance. Several scenarios for capacity reduction are tested, from the nominal value to values close to zero (sector closed), reported in the second column of the Table 1. These two parameters, i.e., percentage of connections and capacity, univocally identify each instance.

To compute optimal solutions we use the CPLEX-MIP solver 9.0, implemented using AMPL as modeling language on a PC AMD-Xeon 4 processors 3 GHz, 8 GB RAM with Linux Ubuntu 4.03 OS. With these input data, the mathematical program has the order of 270,000 constraints and 150,000 variables, after pre-processing. In the pre-processing phase about 160,000 constraints and 200,000 variables are eliminated. Given the size of the instances, we accept good solutions within an optimality gap of 1%. The gap of the solution is listed in the fourth column of Table 1. To solve these instances, we also took advantage of the CPLEX’s capabilities of generating constraints (cuts) based on polyhedral considerations. In particular, we enable moderate generation of clique cuts, setting the corresponding parameter to 1. The number of additional cuts of clique, implied bound (Bound) and Gomory type are listed in the fifth, sixth and seventh column of Table 1 respectively. Finally, in the last column, we report the value of the objective function with the intention to provide a clue on the amount of delay assigned.

What immediately appears from the computational results is that CPLEX can compute good solution, if not optimal, in all the cases. The average solution time is 241 seconds. In only one case, the instance with 50% of connections and

Table 1. Computational results

% of Conn.s (# of Flights)	Capacity	Solution					
		Time (secs.)	GAP (%)	CUTS			Iter.s
				Clique	Bound	Gomory	
50 (3003)	3	Infeasible					
	4	167.0	0.66	1551	394	46	143528 2972.3
	5	352.2	0.28	2483	486	47	143763 2677.2
	10	225.1	0.45	3340	537	50	142399 1853.8
	20	383.4	0.21	3700	510	57	135670 949.5
	30	724.2	0.49	2046	513	68	130214 409.8
	40	382.0	0.28	852	292	50	124178 164.3
	50	116.1	0.00	1133	297	34	120932 129.0
	60	161.3	0.00	1231	358	36	120613 123.9
	70	213.5	0.00	1276	360	37	120417 123.1
60 (3027)	6	Infeasible					
	7	194.5	0.61	1050	311	37	144194 2883.3
	10	215.7	0.63	2014	406	45	142092 2381.4
	20	207.3	0.72	2420	358	43	134586 1471.9
	30	480.9	0.02	2308	476	38	133793 829.5
	40	390.8	0.00	2381	528	54	128156 455.4
	50	191.3	0.00	1936	448	61	123506 342.5
	60	213.4	0.41	2354	450	59	121359 297.6
	70	204.4	0.40	2404	452	57	120385 280.1
70 (3140)	10	Infeasible					
	11	194.8	0.06	-	-	6	152028 3554.1
	12	139.0	0.17	-	-	-	147699 3357.3
	15	106.4	0.08	-	-	-	141554 2857.6
	20	125.8	0.02	-	-	6	142124 2303.3
	30	76.6	0.00	-	-	-	136489 1487.1
	40	177.3	0.57	-	-	11	135555 949.8
	50	109.4	0.20	-	-	10	127251 688.2
	60	42.7	0.00	-	-	-	124369 612.8
	70	42.8	0.00	-	-	-	123612 567.0
80 (3240)	11	Infeasible					
	12	196.4	0.16	1284	331	48	157695 3080.7
	15	248.3	0.04	1341	320	39	153471 2638.3
	20	271.6	0.00	2169	409	48	153777 2149.6
	30	368.7	0.00	2039	393	45	146618 1398.6
	40	326.5	0.47	1891	486	51	144897 930.0
	50	203.9	0.00	1148	443	46	139034 660.2
	60	237.6	0.98	1458	318	32	136602 564.7
	70	148.9	0.72	1130	296	32	133049 517.5
90 (3196)	14	Infeasible					
	15	235.7	0.99	2320	421	49	155881 2947.0
	20	303.7	0.02	2694	523	43	150234 2325.9
	30	266.5	0.56	2374	440	55	148645 1581.0
	40	327.6	0.75	1500	496	55	143615 1105.8
	50	380.4	0.40	2461	436	62	138366 862.8
	60	220.6	0.00	2512	395	51	134305 745.9
	70	307.7	0.04	982	351	52	134156 695.2

30 flights per period of capacity, the solution time is larger than 10 minutes. This is also the only instance for which the algorithm branched (68 nodes) in order to compute the solution within the optimality tolerance. On average instances with 50% of flight connections require longer computational time. This trend is explained by the larger number of symmetries that this set of instances may have, i.e., between flights flying the same $o-d$ pair at the same time. On the other side, the set of instances that shows better computational performances is the set with 70% of flight connections. This set exhibits by far the smallest computational time.

Moreover, the computational performances of the mathematical model do not degrade as we consider instances close to the “infeasibility border”, as experienced in [10] for instance.

3.2 National Size Instances

The instances of “national” size consider 30 airports, 10 of which are hubs, 145 sectors and 22 time periods. All the instances manage 6475 flights with 5180 connections (80%). For these instances the nominal capacity of the sector is set to 130 flights per period. The capacity of sectors affected by the bad weather front is here reported in percentage of the nominal capacity (first column of Table 2).

To solve these instances we use the same setting for the CPLEX parameters as in the regional case. In addition, a time limit of 3,600 seconds is imposed. With these input data, the mathematical program has the order of 570,000 constraints and 305,000 variables, after pre-processing. In the pre-processing phase about 280,000 constraints and 340,000 variables are eliminated (fixed).

For this set of instances the average computational time to provide a solution within 1% is 987 secs. The median value is much smaller, equal to 710 secs. Indeed, in one case, that is the instance with the effective capacity equal to

Table 2. Nation wide instances

Capacity (%)	O.F. value	Solution Time (secs.)	CUTS			Iter.s	GAP (%)
			Clique	Bound	Gomory		
0	Infeasible						
10	2620	891	14879	3664	55	266454	0.09
20	1672	643	11424	2917	32	256845	0.00
30	1108	988	7739	2356	60	244400	0.75
40	693	836	7920	1951	63	238985	0.00
50*	414	3600	5458	1743	67	242378	1.05
60	265	631	6232	1487	53	231241	0.00
70	154	889	5585	1151	77	227936	0.99
80	74	694	3632	1178	74	226179	0.00
90	23	710	5355	1009	69	221224	0.00
100	12	437	1346	707	65	218378	0.00

* 332 Branch-and-Bound nodes have been generated

50% of its nominal value, the algorithm can't compute a good solution with 1% optimality gap within the time limit of 3600 secs. This is also the only case in which the algorithm requires the branching phase, exploring 332 branch-and-bound nodes during its execution. However, accepting a larger optimality tolerance, say 3%, then the algorithm computes a good solution in 569 secs. The other statistics for this solution are as follows: objective function value of the solution is 417 with an optimality gap of 2.58. During the pre-processing phase 5458 cuts of clique type, 1693 of implied bound-type and 67 of Gomory-type are added. For all the instances the statistics are listed in Table 2.

To evaluate the effect of rerouting, we compare the solutions of the ATFM model with and without rerouting respectively. When sectors' capacity is close to the nominal value the difference between the two solutions is rather small, both in terms of ground and airborne holding delay. But, as the sectors' capacity decreases, the benefits of rerouting increase, in terms of smaller amounts of both ground and airborne holding delay assigned. In the most congested case, capacity equal to 10% of its nominal value, the reduction of ground delay is almost 30% (from 949 to 733 time units). The airborne holding delays decrease as well, dropping from 150 time units to 126 with an improvement of 19%. It is important to note that such a benefit is gained with a small amount of rerouting actions. Even in the most congested case, the number of rerouted flights (220) is rather small, which corresponds to 3.4% of the total flights.

4 Conclusions

In this paper, we presented a new mathematical model for the Air Traffic Flow Management problem. The key feature of the model is that it also includes rerouting decisions and they are formulated in a very compact way. In fact, it does not require any additional variable, but it only introduces new constraints, which implements local routing conditions. We also presented three classes of valid inequalities with the scope of strengthening the polyhedral structure of the underlying relaxation.

A wide computational analysis on realistic instances demonstrated the viability of the proposed model. We solved realistic instances of the problem in short computational times, which are consistent with the decision process inside the ATFM Central Unit. Given that our approach includes all the air traffic control decisions (ground holding, air holding, adjusting speed of aircraft and rerouting) combined with the attractive computational times, makes us optimistic that this approach may succeed in becoming the main air traffic control engine.

References

1. Bertsimas, D., Odoni, A.: A critical survey of optimization models for tactical and strategic aspects of air traffic flow management. Technical report, NASA (1997)
2. Bertsimas, D., Stock, S.: The Air Traffic Management Problem with Enroute Capacities. *Operations Research* 46, 406–422 (1998)

3. Bertsimas, D., Stock Patterson, S.: The Traffic Flow Management Rerouting Problem in Air Traffic Control: A Dynamic Network Flow Approach. *Transportation Science* 34, 239–255 (2000)
4. EUROCONTROL Performance Review Commission, Performance Review Report, Brussels (2004)
5. Helme, M.: Reducing air traffic delay in a space-time network. In: *IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, pp. 236–242 (1992)
6. Hoffman, R., Mukherjee, A., Vossen, T.: Air Traffic Flow Management. Working Paper (2007)
7. Lindsay, K., Boyd, E., Burlingame, R.: Traffic flow management modeling with the time assignment model. *Air Traffic Control Quarterly* 1, 255–276 (1993)
8. Lulli, G., Odoni, A.R.: The European Air Traffic Flow Management Problem. *Transportation Science* 41, 1–13 (2007)
9. Odoni, A.R.: The Flow Management Problem in Air Traffic Control. In: Odoni, A.R., Bianco, L., Szego, G. (eds.) *Flow Control of Congested Networks*, pp. 269–288. Springer, Berlin (1987)
10. Vranas, P.B., Bertsimas, D., Odoni, A.R.: The Multi-Airport Ground Holding Problem in Air Traffic Control. *Operations Research* 42, 249–261 (1994)