A Pre-tactical Generalised Air Traffic Flow Management Problem

Rainer Kaufhold, Rainer.Kaufhold@dfs.de
Deutsche Flugsicherung GmbH, Department Research and Development,
Research Campus Langen, Germany

Steffen Marx, Steffen.Marx@tu-dresden.de
Carla Müller-Berthel, Carla.Mueller-Berthel@tu-dresden.de
Karl Nachtigall, Karl.Nachtigall@tu-dresden.de
Technical University of Dresden, Faculty of Transport and Traffic Sciences,
Chair Traffic Flow Science, Germany

Abstract

In this paper we present a new model for solving a Generalised Air Traffic Flow Management Problem *(GATFM), which combines a ground-holding problem with en-route air traffic flow management and, moreover, contains a runway assignment problem.*

This approach shall help to close the existing planning gap between the Europe-wide air pre-tactical traffic flow managment process of CFMU and the fine tuned, short term and airport focussed planning tools used in tactical planning..

The main contributions of this paper is a novel, network based capacity management model. Based on this approach an integer linear programming problem formulation for the GATFM problem and a dedicated new solution approach, based on column generation, is deduced and discussed. The theoretical concept is generic and can be applied to a lot of similar problems in ATFM, especially to combine slot allocation and optimal re-routing First results for real world data of the airport Frankfurt/Main are presented and look very promising.

Introduction

The European ATFM is currently performed by two instances, the CFMU covers flow management of en-route air traffic in the European upper airspace, whereas the regulation of in- and outbound traffic of an airport is done by ATC and airport planners.

Tactical planning tools (4D-trajectory planner, arrival- and/or departure management systems) are typically short term acting, approximately 30 minutes before time of operation, and have a very detailed view to the airspace infrastructure, but therfore only focus to a small area, especially the runway system of airports and its surrounding pickup and deliver sectors. Compared to the short term tactical planning tools, the slot allocation process of CFMU matches airspace capacity and demand in a rough manner. CFMU applies a pure ground-holding strategy, which only decides on the departure time slot. En-route flight plans are considered to be fixed given. Airport capacity is only indirectly treated in terms of arrival sector capacity; a refinement to the runway system to differ arriving and departing traffic streams is not captured. In daily practice one observes a gap between this Europe-wide, but more rough planning process of CFMU and the very detailed, but only locally acting tactical tools.

In this paper we present a model, which shall help to close this gap by linking the CFMU slot allocation process with the fine tuned short term tactical process.

This paper is organised as follows: The next section classifies this contribution within the context of Air Traffic Flow Management. Basic aspects of the mathematical model are described in the following section. Finally we give some details for modelling the airport Frankfurt/Main and discuss computational results for some selected traffic scenarios.

Background

During the 3^{rd} Aviation Research Programme of the German Federal Government a new pre-tactical flow management tool has been designed by the German Air Traffic Control (DFS), the German Aerospace Center (DLR) and the Technical University of Dresden.

A prototyped demonstrator has been realized and is called **C**ooperative **L**ocal Res**ou**rce Planer (CLOU). It is a decision supporting pre-tactical flow management tool, which generates for each runway arrival/departure balanced traffic streams within a time horizon of 2-3 hours before time of operation. All stakeholders, flow management staff, aircraft operators, ATC and airport planners, shall have access to the CLOU and may use it in a what-if-probing mode to support collaborated decision making for intended traffic flow regulations.

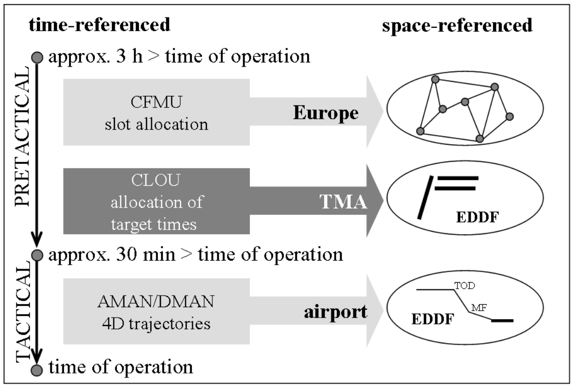

Figure 1 Air Traffic Flow Management

Flow regulations of CLOU shall affect departing and arriving flights of the considered airport. The presented model is especially dedicated to airports operating departures and arrivals in a mixed mode. Arriving flights, which are still on ground at their origin airport, may be imposed by a ground delay; already airborne arrivals are regulated by aircraft performance feasible en-route manoeuvres. For departing flights, originated from the considered airport, a ground holding strategy can be applied. Clearly, an overall optimum requires a clever ratio of arriving and departing flights. Moreover, the complexity of this optimisation problem is increased by the task to decide which of the available *spacial resources*, mainly defined by the runway system, shall be used by each of the flights. Runway allocation for a flight depends on its destination and is mainly determined by the operational mode, which are either fixed due to certain weather conditions, or, can be manually determined by the ATC staff according to the mix of the traffic flow demand forecast. Clearly, efficient decisions on those demand-depending operation rules will strongly interact with the time-slot calculation of the flight movements. A two-step serial method, which at first assigns available runway resources to each flight and then, by the second steps, allocates appropriative time slots for its usage of the pre-assigned space, will be not efficient enough to resolve this interdependencies between the runway assignment and time slot calculation.

For each of the flights considered within a time horizon of 2-3 hours, CLOU calculates a target time, which should fall into the so-called control window of the flight. The control window is a time interval, which contains all feasible target times. For airborne flights, this range is determined by the aircraft performance parameters. In case of a CFMU regulation, the control window of departing flights is defined by the CFMU slot window; otherwise the target time for departure is not limited from above. Those flight plan related data are calculated by preprocessing the demand describing input data 'schedule, estimate and CFMU slot'. Optimisation criteria were designed as a combination of different objectives, which reflect the interests of all stakeholders.

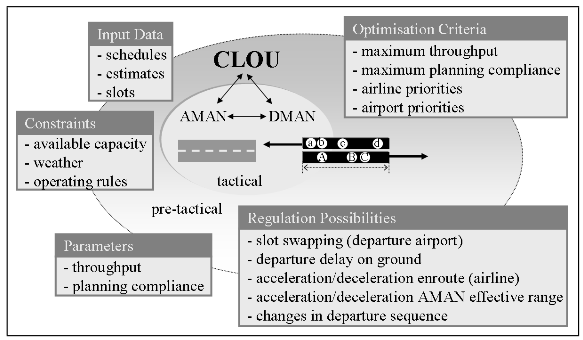

Figure 2 CLOU Basic Concept (source: K-ATM AP2210)

In summary, we are given a complex *Generalised Traffic Flow Management Problem*, which includes a runway assignment and a ground-holding plus en-route flow regulation problem.

Mathematical Model

Capacity Resource Counter Network Model

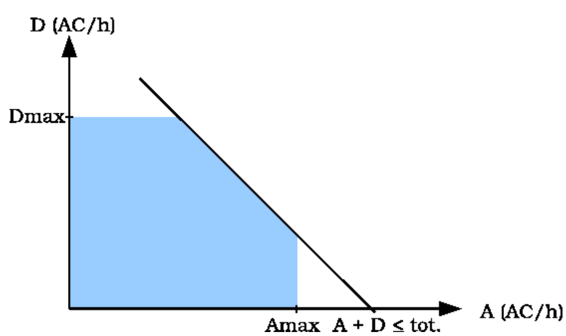

Figure 3 Capacity Limits of an Airport.

Overall airport capacity is usually characterised by three limiting bounds, the maximum number A_{max} of arrivals, the maximum number D_{max} for departures and a bound for the number of total movements with $A + D \leq tot$. (see Figure **3**).

Airspace capacity must be carefully distributed among the runway system and between arriving and departing streams. To do this, we use a network, which will be called the *capacity resource counter network*. Each node represents a partial amount of capacity which is allowed to be used by a specified part of the runway system for either only departures or arrivals or both types of flights. The task of each such node is twofold; at first it has to control, that only flights of the right type (arrival and/or departure) are allowed to use this capacity resource and on the other hand it has a counter function for watching the limited capacity, which is distributed over time.

Each flight allocates capacity by using a certain route or path through this network. This path starts at the source node, which is the counter for the total airport movements and terminates at one of the possible sink nodes in the network. For departing flights, the definition of the sinks is done by significant waypoints of the different departing routes within the airport sourrounding area. By differentiating the permeability of the counter nodes by the departure destination directions, this network model is flexible enough to cover a lot of operational rules used in practice.

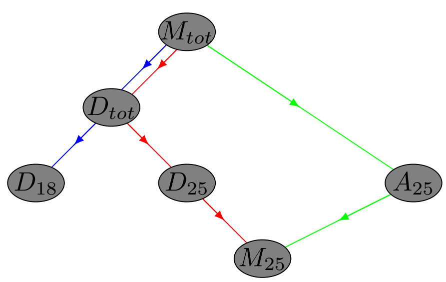

Figure 4 Simplified Capacity Resource Counter Network

A simplified model for Frankfurt/Main (for more details, see section Computational Results), is given by Figure **4**. The symbols M,A,D indicate that either all types of **M**(ovements) or only **A**(rrivals), **D**(epartures) respectively, are operated by this node. The indices tot, 25 or 18 describe the used part of the runway system, which could be either the **tot**al or the parallel system RW**25** or the runway RW**18**. In this way, the blue and red path may be used by departing flights with take-off from runway RW18 (blue) or runway RW25 (red). Arriving flights are always landing on RW25 and must therefore use the green path.

For each flight we have to calculate a so-called *target time* T, which is defined to be that point of time at which the flight is registrated by the capacity resource counters. The counters may be referenced to to different locations. Landing and take-off events for one runway will be counted at the runway itself. When looking to the overall airport capacity, it will be better to count arriving and departing flights at the gate in terms of on- and off-block events. In order to cover this, each arc $a : C_1 \longrightarrow C_2$ of the resource counter network are assigned by *transition* times, which is the time difference for counting the same flight by the initial (C_1) and terminal node (C_2) of the arc. The target time T for all flights is referenced to one common location, which is the associated location of the network source node. For a route R used by flight f with target time T, we define by $cto^R_{r,f}(T)$ to be the *calculated time over* the resource counter r.

An integer linear model for GATFMP

This section gives an overview of the mathematical model and the implemented solution method.

In a more system theoretical sense, the discussed optimisation task can be understood as a *Generalised Air Traffic Flow Management Problem* , in which optimal ground-holding as well as airborne delay and moreover a route (= runway) assignment for each flight has to be computed. In order to combine those optimisation problems to one common model, we use time- and route-indexed binary decision variables

$$x^R_{f,T} := \begin{cases} 1, & \text{if flight } f \text{ takes route } R \text{ with target time } T \\ 0, & else \end{cases} \qquad (1)$$

The use of time-indexed decision variables is a well known approach for formulations of the ground-holding or slot allocation problem (see e.g. [2, 4, 7, 5, 3]) Binary decision variables are known to be very **powerful** for modelling complex practical requirements. On the other hand, those models often suffer from the extreme large number of decision variables, which leads to **large computation times**.

For the following we assume, that the route choice set $\mathcal{R}(f)$ of possible routes for flight f keeps small and can therefore be explicitly enumerated. In 'Frankfurt/Main' only a departing flight is allowed to use either runway RW18 or the runway system RW25. Arriving flights will always use the runway system RW25. This means, that the set $\mathcal{R}(f)$ will contain at most 2 different routes. In contrast to the small number of route indices R the set of possible time indices T (= set of all potential target times) is extremely large.

To define 'counter' capacity appropriately we apply time disretisation, i.e. time is divided into a partition of periods $I_1, ..., I_n$, for each of which an integer valued capacity $c(I_j) \in \mathbb{Z}$ is defined. Using the previous explained time transformation 'calculated time until', the left hand of the sum

$$\sum_{r \in R; cto^R_{f,r}(T) \in I_j} x^R_{f,T} \leq c_j$$

counts all that flights, which are planned to use capacity resource r during the time period I_j. This load number is limited by the capacity c_j. The collection of all flow constraints (including all different counters) is denoted by the set $\mathcal{FC}$ and indexed by $j \in \mathcal{FC}$.

The cost coefficients $\omega^R_f(T)$ define the cost for scheduling flight f with target time T on route R. Note, that this general model allows arbitrary complex cost function $\omega^R_f(T)$. This advantage could be intensively used to design an objective of the optimisation problem, which covers the different interests of all stakeholders. Some of the main indicators are:

CFMU slot violation is the difference of between the CFMU calculated take-off time (CTOT) and the calculated target time.

Estimate Delay is the difference between estimated landing time and calculated target time. Low estimate delay will avoid airborne holdings.

Schedule Delay is the difference between schedule time and target time and used to optimise punctuality.

Now, the complete model is given by

$$\text{total cost} = \sum_{(f,T,R)} \omega^R_f(T) x^R_{f,T} \rightarrow min$$

subject to

$$(a) \quad \forall\, f \in F \sum_{T \in \Delta(f), R \in \mathcal{R}(f)} x^R_{f,T} = 1$$

$$(b) \quad \forall\, \mathbf{j} \in \mathcal{FC} \sum_{r \in R; cto^R_{f,r}(T) \in I_j} x^R_{f,T} \leq c_j$$

$$x^R_{f,T} \in \{0,1\}$$

Constraints of type (a) guarantee, that each flight will be assigned with exactly one resource route R and one target time T. The inequalities (b) define the capacity constraints, which allow for each pair $j := (r_j, I_j)$ of a resource counter r_j and time period I_j a limiting number c_j for the allowed movements. The left hand sum of (b) counts the number of flights, which will consume capacity during the considered time period.

Since the early 1990's, huge integer linear problems, even with natural and concise formulations were challenging to solve in practice. The most significant advance in general methodologies occurred in 1991 when Padberg and Rinaldi [6] merged the enumeration approach of branch and bound algorithms with the polyhedral approach of cutting planes to create the technique usually call branch (and bound), cut and price or simply BCP. Integrating the contributions of many in the field , their paper launched a new area in discrete optimisation techniques. Nowadays the BCP framework seems to be the modern state-of-the art to handle huge integer linear programs:

Branch and bound is the broad class of algorithms from which branch, cut and price is descended. A branch and bound algorithm uses a divide and conquer strategy; it partitions the solution space into subproblems and then optimises over each subproblem individually. If the number of decision variables is very large (which in fact is the case for our model), the variables (= columns of the constraint matrix) are generated dynamically. If a column i is not present in the current model matrix, then the associated variable x_i is implicitly taken to have value zero. The process of dynamically generating variables is called *column generation* (see [1]) and done by computing the reduced cost of the non-active column, which will be added to the model if it has **negative** *reduced cost*. The term 'price' in

BCP originates from the linear programming jargon to 'price out' variables with negative reduced cost by the use of the dual prices associated with each constraint of the LP.

Branch And Bound algorithms are usually organised by processing the resulting decision or search tree, where the childs of a parent node represent the problem partition. Fully enumerated, this tree is of exponential size. In order to keep the investigated area of the search tree small, efficient branch and bound implementations make use of lower and upper bounds for the subproblems. Upper bounds may either be calculated directly from a feasible integral solution, or, for the case that only a fractional solution of the LP-Relaxation is available, by applying a problem specific rounding heuristic. We use a rounding method by calculating a FCFS solution, which uses for each flight an earliest possible target time defined by the mean target time of all non-zero, fractional decision variables associated with this flight.

If all variables are contained in the model, a lower bound is easily given by the value of the LP-relaxation (= minimal objective with fractional decision variables $0 \leq x_{f,T}^{\tilde{R}} \leq 1$), which is calculated during exploring a search tree node. When using column generation, this simple lower bound definition does not hold anymore, because the (typically huge amount of) non-known columns must be incorporated into the computation of the true lower bound. Finding good lower bounds can therefore only done specifically to the problem and requires apart from a deep theoretical insight sometimes a lucky hand.

Within the context of our model, generating new columns means to look for each flight f for an alternative route R through the counter network and an alternative target time T, which leads to new (i.e. not yet) considered decision variables $x_{f,T}^{R}$. Adding the most promising candidates to the model, will improve the actual best solution. Finding those variables is called the *pricing problem* and done by identifying those variables with minimum negative reduced cost. Reduced cost are calculated by using the dual prices, which are associated with each linear constraint of the underlying linear model. In our case, we have to deal with two types of constraints or equivalently, dual prices. This are

- Constraints of type (a) are assigned with dual prices ξ_f, which may be interpreted as that amount of cost, for which the solution could be potentially improved by cancelling this flight.
- Constraints of type (b) impose dual prices μ_j, which measure the saturation of the associated resource counter r_j during the time period I_j. Large values for μ_j, indicate high traffic congestion on r_j during I_j. The values μ_j may be understood to be that amount of cost for which the objective value will be increased, if one more additional flight will make use of the resource r during the time period I_j.

Now, the idea of the pricing mechanism can be described as follow: Cancelling flight f will **decrease** the total objective for the amount ξ_f. Adding f with an alternative route R and alternative target time T again will **increase** the total cost again for the amount

$$\omega_f^R(T) + \sum_{r \in R; cto_{f,r}^R(T) \in I_j},$$

which consists of the local cost for the flight f itself and the additional cost for the other flights due to congested counter time periods. Hence, the exitance of an alternative route R and and target time T with non balanced prices in the sense

$$\omega_f^R(T) + \sum_{r \in R; cto_{f,r}^R(T) \in I_j} \mu_j < \xi_f,$$

will potentially improve the overall solution. Exactly this is reflected by **negative** *reduced cost*

$$\hat{\omega}_f^T(R) = \omega_f^R(T) + \xi_f - \sum_{r \in R; cto_{f,r}^R(T) \in I_j} \mu_j < 0,$$

for which we have to look during the pricing problem. Searching for new columns with minimum negative cost can be formulated as a time-depending shortest path problem in the underlying resource counter network.

For GATFM Problem we obtain a lower bound by using the well known concept of *Lagrange relaxation*. Out-pricing of the capacity constraints, Lagrange relaxation leads to the lower bound lb.

$$\begin{aligned} lb &= \sum_j c_j \mu_j \\ &+ \sum_f \min_{R,T} \left(\omega_f^R(T) - \sum_{r \in R; cto_{f,r}^R(T) \in I_j} \mu_j \right) \end{aligned}$$

For each flight we have to find the minimum priced route and target time, which is already done during the solution of the aboved discussed pricing problem. Hence, we need no more computation time to find this lower bound. Computer runs show, that this lower bound is reasonable increasing during the iteration and is tight enough to cut off a lot of useless search in the branch-and-bound tree.

Computational Results

In this section we develop exemplarily the load counter network for Frankfurt Airport and execute experiments in order to evaluate the performance and the benefit of this approach.

Modelling Airport Frankfurt/Main

The airport Frankfurt/Main consists currently of three runways, a closely spaced parallel runway system (heading 070/250) and a single runway (heading 180). The parallel runway system serves arrivals and departures whereas the single runway is solely dedicated to departures (cp. table 4). The runway system is mainly operated in Western direction (operation direction 25).

Traffic regulation measures are enabled by flexible shifting of departures between the parallel runway system and the single runway. This shift has to be done in departure streams: via TABUM[1], via BIBOS[2] and via one of the other waypoints (REST).

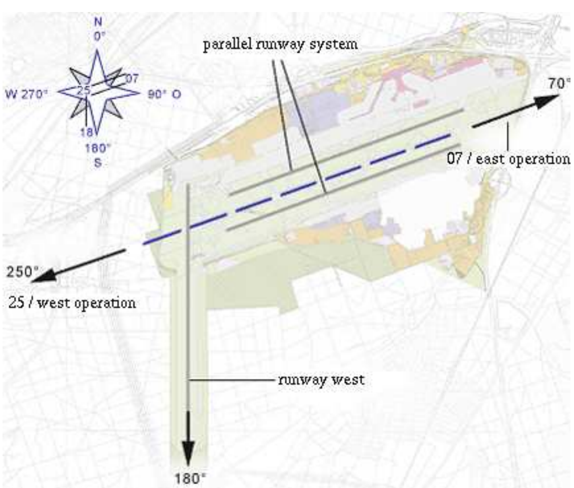

Figure 5 Runway System of Frankfurt Airport (Source: Fraport AG)

Because of these characteristics the runway system is modeled as two-runway system in operation direction 25/18. The load counter network of the airport Frankfurt/Main consists of three load counters representing the runway system (MOV, ARR, DEP), two counters representing the departure capacity (DEP_{25}) and the total capacity (MOV_{25}) of the parallel runway system and one counter representing the departure capacity (DEP_{18}) of the single runway. Furthermore, three counters serve as sinks for departing flights according to their SID[3] and one additional counter for (all) arrivals.

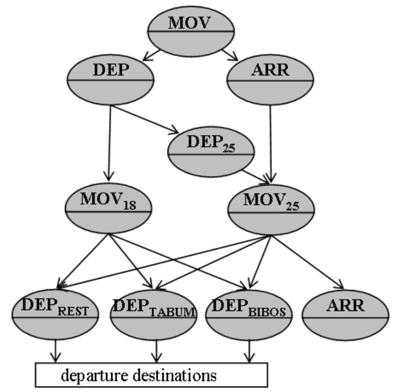

Figure 6 Load Counter Network of Frankfurt Airport

Optimisation Runs

The computer runs have been performed on a Novell/Suse Linux 10.0 workstation with 8 GB main memory an two intel Xeon 3.2 GHz processors.

Computer runs are performed for real world data of Monday, April 18 from the German airspace whose orign or destination is the airport Frankfurt/Main. In order to cover typical, but different traffic situations, a lot of different scenarios were generated by DFS and DLR. We will present two selected instances:

MS3200 Runway 18 has to be closed for two hours (10am to 12am) due to strong headwinds. The overall capacity decreases from 84 to 66 flights per hour.

MS5200 Normal operation of the runway system offering full capacity with 87 flights per hour.

Running times and performence are illustrated by the following figure.

[1] waypoint being passed by flights in Northern direction
[2] waypoint being passed by departing flights in North-Western direction
[3] standard instrument departure route

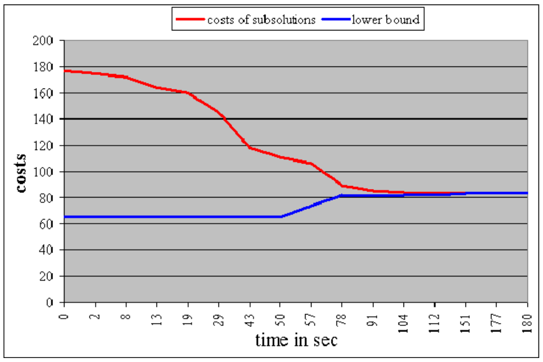

Figure 7 Computational run of scenario MS3200

The column-generation (CG) approach initially starts with a First-Come-First-Serve solution. The calculated times by this method are called natural times, which principally coincide with that times one observes by the currently applied planning procedures. This solution is immediately available and is uniformly improved during the CG iteration. During each pricing iteration step, a lower bound is calculated. This frontier (red data) monitors the achieved quality of the actual solution (blue data). Half of the optimisation time (ca. 100 seconds) is used for improving the solution, the remaining time is used to 'prove' optimality.

For each of the above described scenarios, Table 1 shows the results of the computer runs within a mixed objective, which contains a reasonable balance of all discussed indicators. For each of the solutions two different delay statistics are reported; the difference between calculated target time and schedule, and target time and estimate, respectively. Those statistics contain always the mean (μ) and maximum delay, as well as the standard deviation (σ) of the delay distribution. Moreover, the table gives for each solution the value of punctuality, which is the percentage of non-schedule-delayed flights[4].

Due to the runway closure MS3200 has an extreme imbalance of capacity exceeding demand, which explains the high values of 30 minutes average delay per flight. The percentage of punctuality of the natural solution can be improved from 8% up to 40%. The improvement of the other indicators, contained in the objective function, can be indirectly seen from the improvement of the total cost (176 down to 83).

The results show the enormous potential of the simultaneous optimisation being realised with this model. Regulation measures are possible by several defined parameters allowing the user adjusting the optimisation and thus adjusting the flow.

[4] A flight is defined to be punctual, if its delay is less than 15 minutes.

Summary and Conclusions

The presented model for solving a ground-holding problem in combination with a runway assignment problem has been sucessfully implemented for the special situation at the airport Frankfurt/Main. The model is flexible enough to cover almost all practical requirements. The first results look quite promising with respect to the achieved improvements in matching demand and capacity. The solution method by a Branch-Cut-And-Price approach is fast enough to be used by a pre-tactical planning tool.

References

[1] C. Barnhart, E.L. Johnson, G.L. Nemhauser, M.W.P. Savelbergh, and P.H. Vance. Branch-and-price: Column Generation for Solving Huge Integer Programs. *Operations Research*, to appear.

[2] D. J. Bertsimas and S. Stock. The Air Traffic Flow Management Problem with En-Route Capacities. Technical report, Alfred P. Shool for Management, Massachusetts Institute of Technology, 1994.

[3] M.J.van den Akker and K. Nachtigall. Slot Allocation by Column Generation . DLR Interner Bericht IB 112-97/04, Deutsches Zentrum für Luft- und Raumfahrt (DLR), 1997.

[4] K.S. Lindsay, E.A. Boyd, and R. Burlingame. Traffic Flow Management Modeling with the Time Assignment Model. *Air Traffic Control Quarterly*, 3(1):255–276, 1993.

[5] L. Maugis. Mathematical Programming for the Air Traffic Management Problem with En-Route Capacities. Technical Report CENA/R95-022, CENA Orly Sud 205, 94542 Orly Aerograre Cedex, France, June.

[6] M. Padberg and G. Rinaldi. A branch-and-cut algorithm for the resolution of large- scale traveling salesman problems. *SIAM Review*, (33):60–71, 1991.

scenario	solution	cost	schedule delay			estimate delay			punctuality	run time
			μ	max	σ	μ	max	σ		
			[min]	[min]	$[min]$	[min]	[min]	$[min]$	$[\%]$	[sec]
MS3200	Natural	176.44	34.41	781.00	50.86	29.15	45.00	11.38	8.30	
	Target	83.37	33.88	834.00	62.89	28.61	104.00	31.87	43.72	165
MS5200	Natural	14.17	4.31	115.00	14.27	2.27	8.00	2.14	89.51	
	Target	12.93	3.98	121.00	14.23	1.93	12.00	2.39	90.95	86

Table 1: Computational Results

[7] P.B. Vranas. *The Multi-Airport Ground Holding Problem in Air Traffic Control*. PhD thesis, Operations Research Center, MIT, Cambridge, MA, 1992.

Keywords

FMAN, Flow Manager, CLOU, Cooperative Local Resource Planer, GATFMP, generalised air traffic flow management problem, load counter network, top-down filter model, linear optimisation, integer linear model

Biographies

Rainer Kaufhold worked from 1992 until 1997 as scientific assistant at the Department for Flight Mechanics and Control at the University of Technology, Darmstadt Germany. In 1998 he received his Ph.D. in engineering by a thesis on *Design of ergonomic perspective terrain representations for cockpit We think that these results are posible for a operational use by ATC. The idea is en time slice of 15 min per optimisation run. displays*. In 1998 Rainer Kaufhold joined the research and development department of DFS Deutsche Flugsicherung. Since then he has been focusing his work on improving co-operative planning processes in ATM. He currently is the project manager of the nationally funded project K-ATM (Cooperative Air Traffic Management).

Steffen Marx is scientific staff member of the Chair of Traffic Flow Science of the Technical University of Dresden. He completed his studies at the Technical University as graduate engineer in traffic and transportation sciences majoring in opimisation of strategical flight scheduling for airlines, pre-tactical ATFM processes and airport ground handling processes. Currently he does his Ph.D. in opimisation of strategical flight scheduling.

Carla Müller-Berthel obtained her graduate engineer degree from the Technical University of Dresden and belongs to the scientific staff of the Chair of Traffic Flow Science at the Dresden University of Technology. Her majoring sciences are besides optimisation of pre-tactical ATFM processes taxiway routing on airports. She does her Ph.D. in optimisation of pre-tactical ATFM processes.

Karl Nachtigall is chairholder of the Chair Traffic Flow Science in the Department of Logistics and Aviation at the Technical University of Dresden. He studied mathematics at the University of Hanover and received his Ph.D. in Operations Research from the University of Hildesheim in 1995. Subsequently he conducted research for the DLR (Deutsches Zentrum fr Luft- und Raumfahrt e. V.) in Brunswick as scientific staff in several scientific projects and received his postdoctoral lecture qualification from the University of Hildesheim in 1998.