A Two-Step Hybrid Metaheuristic for Flight Level Allocation in 4D Trajectory Generation

Jérémy Omer, Alexandre Gondran

A Two-Step Hybrid Metaheuristic for Flight Level Allocation in 4D Trajectory Generation

J. Omer[1] and A. Gondran[2]

[1] Group for Research in Decision Analysis (GERAD)
École Polytechnique de Montréal, Montréal (QC), Canada
jeremy.omer@gerad.ca
[2] MAIAA, ENAC
French Civil Aviation University, Toulouse, France
alexandre.gondran@enac.fr

1 Context

With an already congested European airspace, the predicted increase of air traffic is an important challenge for the decades to come. The current air traffic management (ATM) is based on a geographic structure of the airspace, but an alternative trajectory-based management is considered for more efficiency [1]. In this context, a 4D *business trajectory* (BT) is negotiated between airlines and ATM authorities. A large part of air traffic control is then delegated to the pilots who are responsible to follow their BTs. The role of the controllers is to monitor the traffic, assist the pilots, and propose new BTs in case of unpredicted events.

During the negotiation procedure, airlines send their demands regarding their preferred take-off time and cruising altitude. ATM then computes BTs accommodating these demands while meeting safety requirements, and other structural constraints. The main safety constraint is that no conflict occurs, i.e. every pair of aircraft remains separated with either one of two reference horizontal and vertical separation distances. One key issue is thus to compute a set of trajectories without conflict. Omer and Chaboud [2] suggest a simulation-based approach to modify conflicting forecast trajectories using an automated conflict resolution. The conflict resolution can be handled with mathematical programming [3], but it might result in irregular and complex trajectories if a lot of conflicts have to be dealt with. The purpose of this work is to develop a flight level (FL) allocation algorithm to avoid most losses of separation occurring between cruising flights before running the automated conflict resolution.

2 Model: target graph vertex coloring

Our model is a modified version of the *list coloring problem* of Allignol et al.[4]. Assuming that the airspace is layered into FLs separated with the vertical reference separation distance, the separation can be guaranteed by allocating different FLs to each pair of aircraft that doesn't meet the horizontal separation. Given a set of aircraft V, a graph of conflict $G(V,E)$ is built by connecting a pair of aircraft with an edge when a loss of horizontal separation is detected between their trajectories. If each FL is matched to one color, the FL allocation problem is equivalent to searching for a coloring of $G(V,E)$ that minimizes the gap with the preferred flight levels (PFLs). This problem is called *target coloring*: each vertex (each flight) has one target color (PFL). The particularity of the context is that we do not intend to compute separated trajectories with the FL allocation alone. Remaining conflicts are allowed as long as their numbers does not overcome a given threshold C. It means that colorings with less than C conflicting edges are allowed (one edge is in conflict if its vertices have the same color). Denoting δ_{FL_i,FL_j} the Kronecker function returning 1 if $FL_i = FL_j$ and 0 otherwise, the model to solve is:

$$\min \left\{ \sum_{i\in V} (FL_i - PFL_i)^2 : \sum_{(i,j)\in E} \delta_{FL_i,FL_j} \leq C \right\} \tag{1}$$

where $\sum_{(i,j)\in E} \delta_{FL_i,FL_j}$ corresponds to the number of conflicting edges and $\sum_{i\in V} (FL_i - PFL_i)^2$ is a distance between the solution and the target coloring. A drawback of defining a threshold C is that a solution of (1) is likely to leave exactly C conflicts. Let Z^1 be the best known value of (1), a second step model is then solved to trade a small extra violation of demands with more conflict resolutions:

$$\min \left\{ \sum_{(i,j)\in E} \delta_{FL_i,FL_j} : \sum_{i\in V} (FL_i - PFL_i)^2 \leq (1+\rho) Z^1 \right\} \tag{2}$$

3 Algorithm: H$_2$col-t, variant of H$_2$col for target coloring

H$_2$col [5] is a hybrid evolutionary algorithm based on the HEA memetic algorithm [6]. Like HEA, H$_2$col blends a greedy partition crossover and the Tabucol [7] tabu search procedure, but its main originality is that the population of solutions is reduced to only two individuals. With this modification, H$_2$col obtains very good results on the DIMACS benchmark for standard vertex coloring. For this reason we base the solution of the FL allocation problem on H$_2$col.

In order to tackle the target coloring problem, some modifications are made on the crossover operator and on the Tabucol algorithm, and (1) and (2) are solved sequentially with the same method. First, the threshold constraint is treated as a second objective. Tabucol then considers the two objectives in a lexicographic order giving priority to the constraint satisfaction. For instance, while solving (1), Tabucol first focuses on getting less than C conflicts before lowering the violation of demands. As regards the crossover, the operator of HEA is designed to deal with the symmetry that exists between the colors in standard vertex coloring. In contrast, the targets partially break this symmetry, so classical crossover operators are combined with the greedy partition crossover of HEA. This new algorithm is denoted H$_2$col-t.

4 Experimental tests and perspectives

The algorithm is tested on European air traffic data. The historical flight plans of several days of traffic are used to get the demands. Two sets of initial trajectories are then computed by respecting the current network of routes or joining the arrival and destination airports with direct routes. To evaluate the benefit H$_2$col-t, a comparison is made with the tabu search algorithm alone and with a version of H$_2$col-t using more than two solutions in its population. Preliminary results show the efficiency of H$_2$col-t on large instances involving more than 30 000 flights.

In the future, it is necessary to focus on conflicts involving two aircraft in evolution. Modifying their cruise altitude is generally irrelevant, but small delays in the take-off times could succeed in avoiding a large number of losses of separation. Using an appropriate model, H$_2$col-t should also be a promising method to solve this problem.

Acknowledgements The second author gratefully acknowledges financial support under grant ANR 12-JS02-009-01 “ATOMIC“.

References

1. SESAR Joint Undertaking: European ATM master plan, edition 2. Technical report (2012)
2. Omer, J., Chaboud, T.: Automated conflict free planning. experiments on real air traffic data. In: Congress of International Council of Aeronautical Sciences (ICAS). (2012)
3. Omer, J., Farges, J.L.: Hybridization of nonlinear and mixed-integer linear programming for aircraft separation with trajectory recovery. IEEE Transactions on Intelligent Transportation Systems **14** (2013) 1218–1230
4. Allignol, C., Barnier, N., Gondran, A.: Optimized flight level allocation at the continental scale. In: International Conference on Research in Air Transportation (ICRAT). (2012)
5. Moalic, L., Gondran, A.: Variations on Memetic Algorithms for Graph Coloring Problems. arXiv **1401.2184** (2014)
6. Galinier, P., Hao, J.K.: Hybrid evolutionary algorithms for graph coloring. Journal of Combinatorial Optimization **3** (1999) 379–397
7. Hertz, A., de Werra, D.: Using Tabu Search Techniques for Graph Coloring. Computing **39** (1987) 345–351