Chinese Journal of Aeronautics, (2014),27(4): 903–912

Chinese Society of Aeronautics and Astronautics
& Beihang University

Chinese Journal of Aeronautics

cja@buaa.edu.cn
www.sciencedirect.com

Dynamic airspace configuration method based on a weighted graph model

Chen Yangzhou *, **Zhang Defu**

School of Electronic Information and Control Engineering, Beijing University of Technology, 100124 Beijing, China

Received 25 March 2013; revised 8 June 2013; accepted 1 April 2014
Available online 5 July 2014

KEYWORDS

Airspace sector;
Air traffic control;
Dynamic airspace configuration;
Graph partitioning;
Graph theory;
Weighted graph;
Workload

Abstract This paper proposes a new method for dynamic airspace configuration based on a weighted graph model. The method begins with the construction of an undirected graph for the given airspace, where the vertices represent those key points such as airports, waypoints, and the edges represent those air routes. Those vertices are used as the sites of Voronoi diagram, which divides the airspace into units called as cells. Then, aircraft counts of both each cell and of each air-route are computed. Thus, by assigning both the vertices and the edges with those aircraft counts, a weighted graph model comes into being. Accordingly the airspace configuration problem is described as a weighted graph partitioning problem. Then, the problem is solved by a graph partitioning algorithm, which is a mixture of general weighted graph cuts algorithm, an optimal dynamic load balancing algorithm and a heuristic algorithm. After the cuts algorithm partitions the model into sub-graphs, the load balancing algorithm together with the heuristic algorithm transfers aircraft counts to balance workload among sub-graphs. Lastly, airspace configuration is completed by determining the sector boundaries. The simulation result shows that the designed sectors satisfy not only workload balancing condition, but also the constraints such as convexity, connectivity, as well as minimum distance constraint.

1. Introduction

In air traffic management, since only several controllers are impossible to put all aircraft simultaneously flying in the whole airspace of a nation under surveillance, the airspace is usually divided into smaller regions referred to as sectors, and each sector is observed by one or more controllers. In this way, the aircraft count of each sector is supposed to be not beyond the controller's ability to monitor. Current sectors are largely determined by historical effects and in an empirical way. And such situation has never changed for a long time. For instance, approximately 600 sectors over USA airspace designed in 1960 have been in use up to now. The configuration of the fixed sectors corresponds to the way that relatively few aircraft fly along the fixed air routes. The airspace characterized by fixed air routes and fixed sectors is referred to as a structured and static one.

* Corresponding author. Tel.: +86 10 67391632.
E-mail addresses: yzchen@bjut.edu.cn (Y. Chen), defuzhang@emails.bjut.edu.cn (D. Zhang).

Peer review under responsibility of Editorial Committee of CJA.

Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.cja.2014.06.009

With the development of air traffic, routes structure and demand profiles have changed a lot over the years. While an increasing number of aircraft fly simultaneously along fixed air routes, more and more air traffic delays may arise for some reasons, such as bad weather and traffic congestion. The situation can be improved by the way the aircraft changes its air routes,[1,2] rather than following certain fixed air routes at all time. Accordingly, sector counts and boundary vary with traffic change. This is a dynamic airspace configuration problem.

Dynamic airspace configuration (DAC)[3] is an encouraging concept proposed to convert airspace sectorization from the structured and static airspace to a dynamic one capable of accommodating dynamically changing traffic demand. A lot of research into DAC has been carried out, and most scholars completed DAC by describing the airspace as a model and then adopting a proper algorithm to partition the airspace into sectors. The airspace models proposed in literature can be summarized as follows: cell model,[4–7] flight trajectory model,[8,9] Voronoi diagram model,[10,11] and graph model.[12–16] In cell model, the airspace is first discretized into cells, i.e., hexagonal grids, and then some algorithms were used to cluster those cells into sectors. For example, Yousefi et al.[4] thought of DAC problem as a standard facility location problem to cluster the cells into sectors, Klein[5] solved the problem via seed growth algorithm, Drew[6] and Tien et al.[7] applied mixed integer programming to cluster cells into sectors. However, the designed sectors may have undesirable shapes for the boundaries which were "jagged". In virtue of flight trajectory model, Briton et al.[8] clustered the flight trajectories into sectors by k-means algorithm, and Basu et al.[9] developed geometric algorithms for DAC. The available literature told us that final sectors based on geometric algorithms still had undesired shapes. By means of Voronoi diagram, Delahaye et al.[10] proposed initial sectors arbitrarily and then optimized them by evolutionary algorithm. Furthermore, Xue[11] improved Delahaye's scenario using iterative deepening algorithm. It should be noted that a common limitation on the three above models is that they have not made use of information on airspace structure. This might lead to the case that the designed sectors might dissatisfy those geometric constraints, such as convex constraint, minimum distance constraint, and so on. At the same time, static airspace structure was taken sufficiently into account in graph model, where vertices represent airports, waypoint and crossing points while edges represent air routes. Applying graph model, Trandac et al.,[12] Martinez et al.[13] Zhang et al.[14,15] and Li et al.[16] implemented airspace sectorization using a constraint algorithm, spectral bisection algorithm, graph partitioning algorithm and spectral clustering respectively. In addition, Klein et al.[17] developed a method that divided a current sector into several dynamic Fix Posting Areas and then reallocated those Areas to achieve DAC.

Due to the graph model being embedded with information on underlying topological structure of the airspace, it usually helps to consider the factors such as air routes and key points, i.e., airports, crossing points as well as waypoints for DAC. Therefore, the graph model is preferred in this paper. Furthermore, we also consider traffic flows along air routes which are used to compute the workloads. The workloads can be assigned as the edge weights and the vertex weights. Such topological structure with traffic flows can be described as a weighted graph mathematically. Thus, the weighted graph model is adopted for DAC here. And it is different from the traditional weighted graph that only edges are assigned with weights, but an undirected graph with the weights on both vertices and edges, where traffic information as much as possible is used. This is the key feature of our graph model.

From the above literature on DAC, we know that several constraints should be taken into account when it comes to the design of sectors. The first is workload constraint. The constraint points out that the workload of each sector should be below a threshold and the workloads of those sectors are balanced, and ensures that workload of each sector does not exceed the controller's capacity to control the aircrafts while the workloads are evenly distributed among designed sectors. The second is geometric constraints consisting of convexity constraint, connectivity constraint and minimum distance constraint. The convexity indicates that an aircraft should not enter the same sector twice, the connectivity is that a sector does not be fragmented, and the minimum distance constraint means that the distance between the sector boundaries and the key points as well as the distance between the boundaries and the air routes ought not to be less than a given minimum value. The geometric constraints ensure that the controller have adequate time to control the aircraft and to solve conflicts which may happen. These constraints are critical to ensure the safety of aircrafts. Hence, the above constraints are considered thoroughly in this paper.

Moreover, from literature we can also know there are several metrics for workload, such as traffic mass, aircraft count, dynamic density, and so on. Computing workload metrics other than aircraft count might have taken more factors into accounts. However, there is no evidence that Traffic Mass and dynamic density are more effective than aircraft count for DAC. Workload metric other than aircraft count might be prohibitive in practical application. Thus, aircraft count is adopted as workload metric in this paper.

This paper applies itself to develop a DAC method based on a weighted graph model. Firstly, we set up a weighted graph model for a given airspace which accurately describes the airspace structure information and traffic data. The procedure begins with constructing an undirected graph model for the given airspace, of which the vertices represent the key points such as airports, waypoints, and the edges represent the air routes. Then, those vertices are used as the sites of Voronoi diagram[18] which divides the airspace into units called cells, and aircraft counts of both each cell and each air route are computed. By assigning both the vertices and the edges with those aircraft counts, an accessorial graph model is built up. Furthermore, in order to facilitate the discussion, the accessorial graph model is simplified into a weighted graph model whose vertices have a one-to-one relationship with Voronoi cells. Accordingly the airspace configuration problem is described as a weighted graph partitioning problem. Secondly, the paper develops a graph partitioning algorithm that divides the weighted graph model into sub-graphs. The algorithm mixes general weighted graph cuts (GWGC) algorithm,[19] an optimal dynamic load balancing (ODLB) algorithm,[20] and a heuristic algorithm inspired from K-L algorithm[21] together. After the cuts algorithm partitions graph model into sub-graphs, the load balancing algorithm together with the heuristic algorithm transfers aircraft count to achieve workload balancing among the sub-graphs. Lastly, the cells corresponding to each sub-graph are combined together into a sector. In all, the method attempts to design the sectors with the objective of balancing workload, minimizing