Efficient Aircraft Traffic Scheduling using Population Heuristics with CART Algorithm

R. S. Priyaa[1], S. Saravana Kumar[2]

[1]M. Phil Scholar, Department of Computer Science,
Sri Jeyendra Saraswathi College of Arts and Science, Coimbatore, India

[2]Assistant Professor, Department of Information Technology,
Sri Jeyendra Saraswathi College of Arts and Science, Coimbatore, India

Abstract: ***A system was developed to efficiently schedule aircraft into congested resources over long ranges and present that schedule as a decision support system. The scheduling system consists of a distributed network of independent schedulers, loosely coupled by sharing capacity information. This loose coupling insulates the schedules from uncertainty in long distance estimations of arrival times, while allowing precise short-term schedules to be constructed. This paper also proposes a covered flight scheduling by analysis accident dataset. The CART algorithm is implemented to the scheduling Air traffic by identified patterns of factors which are significantly associated with incidents or accident. The achieved result shows the proposed Population Heuristics with CART Algorithm provide much improved result in Speed and Accuracy than existing algorithm***

Keywords: Data mining, decision support system, CART, Population Heuristics, uncertainty

1. Introduction

The Air Traffic Control (ATC) system has the responsibility of maintaining a safe and orderly flow of aircraft in the National Airspace System (NAS) of the United States. An important part of this responsibility is the planning of airport operations, such as the arrival and departure of aircraft. In this paper, we present a procedure for computing safe and efficient arrival schedules while taking into account several key operational consflights. Statistical analysis of aircraft arrivals at several major airports in the United States has shown that the distribution of times between estimated arrival times of successive aircraft (estimated when the aircraft are 100 miles from their final destinations) is nearly-exponential.

2. Problem Definition

Traditionally, scheduling problems have been confined to the airport and sometimes the arrival fixes into the airspace surrounding the airport (A1–A4) [6-7]. The arrival fixes mark the entry into the TRACON, where the aircraft are under the control of specialized arrival controllers. Outside of this area, the aircraft are managed by air traffic controllers in one or more centers. Such uncertainties mean that schedules produced over long ranges may be inaccurate or infeasible. Methods based on releasing aircraft from ground holding typically assume that the unimpeded flight times are deterministic. What seems to be needed instead is an adaptive approach, where schedules can be adjusted as the uncertainties are resolved. An alternative is to use a stochastic approach, where the uncertainties are modeled into a distribution and decisions made to control the probability of particular undesirable conditions from occurring.

3. Related Work

Andreatta [1] proposed ground holding to free flight: An exact approach. In which observes the congestion in air traffic networks is a serious problem and has received a lot of attention both from the aviation authorities and from the scientific research community. J Atkin [2] proposed On-line decision support methodology to tackle the complex problem faced every day by runway controllers at London Heathrow Airport. Aircraft taxi from stands to holding areas at the end of the take-off runway where they wait in queues for permission to take off. J Beasley [3] proposed on displacement problem arises when we have to make a sequence of decisions and each new decision that must be made has an explicit link back to the previous decision that was made. This link is quantised by means of the displacement function. D Bertsimas [4] proposed 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 consflights provide facets of the convex hull of solutions. H Idris [5] adapted a refined benefits assessment of multicenter traffic management advisor for philadelphia and new York. One of the major goals of the Federal Aviation Administration (FAA) is to ensure the safe, efficient, and orderly movement of air transportation through the air traffic management (ATM) system, including both domestic and oceanic airspace.

4. Proposed Architecture

The Proposed System will consider more input parameters for rate profiles and for long resource scheduling. The proposed method is applied to the maximization of aircraft arrivals and minimization of delays in the arrival airspace due to exogenous capacity reductions. The objective function measurements [8] to iteratively update system control parameters until parameter values are reached which locally optimize the objective function. In proposed system CART algorithm were used to prevent the accident and incident vectors and identified patterns of factors which are significantly associated with accidents or with incidents. The findings were ranked using Cluster similarity measure.

Results of the analyses conducted on multiple databases were compared at the end

4.1 Data Pre-Processing

Our study analyzed the underlying factors of accidents and incidents. The historical data on incidents is large enough to represent these factors qualitatively. Also, we consider all factors that are present in the events, primary or contributory. This minimizes the impact of the bias in reporting a factor as contributory versus primary. We first developed a common taxonomy across the accident and incident databases to identify common fields (factors) between the two classes of events. We then transformed each report into a vector consisting of the common fields populated with their corresponding values for each report. We applied the CART algorithm to the accident and incident vectors and identified patterns of factors which are significantly associated with accidents or with incidents. The findings were ranked using Cluster similarity measure. Results of the analyses conducted on multiple databases were compared at the end.

4.2 CART Algorithm

CART stands for classification and regression trees, a non-parametric statistical algorithm developed by Leo Breiman et al. The non-parametric approach of CART does not depend on any function to describe the relation between data sets. CART can be used to predict or analyze both categorical (classification) and continuous or numerical (regression) data. A unique feature of CART is that it illustrates the data in the form of a decision tree, unlike other statistical analysis procedures. The tree structure allows CART to handle complex data, presenting the output in the form of diagrams that are easy to understand. CART does not require variables to be selected in advance. CART is a binary recursive partitioning technique. The term binary implies that each data set is represented by a node in a decision tree, which can only be split into two subsets; the tree starts from the root node containing the data objects, which is split into two child nodes. Thus each resulting node can be split into another two child nodes, depending on the splitting criterion for the variable selected from the group of independent or explanatory variables. CART is recursive because binary partitioning can be repeated to split the data into additional children nodes. The result of the split can be a terminal node, which implies that it cannot be split further, or a parent node, which consists of objects to be divided again into two child nodes. This process of splitting is repeated until resulting child nodes are homogeneous; the condition that requires the objects in the node to be similar or the child node should contain a predefined number of objects.

The following is a pseudo procedure:
Step 1:
Start with root node (t = 1)
Step 2:
Search for a split s* among the set if all possible candidates s that gives the purest decrease in impurity.
Step 3:
Split node 1 (t = 1) into two nodes (t = 2, t = 3) using the split s*.
Step 4:
Repeat the split search process (t = 2, t = 3) as indicated in steps 1-3 until the tree growing the tree growing rules are met.

4.3 Ranking Using Clustering Algorithm

Once significant factor-sets were identified by the algorithm [9] m, we ranked them based on the fuzzy cluster measure. We calculate the Factor Support Ratio for each factor-set as the ratio of the factor-set's support in accident dataset over its support in the incident dataset. The information conveyed by this measure about the factor-set is different than that of the deviation (the difference between the factor-set's accident and incident supports) that is used in the algorithm. The Support Ratio is the probability of a factor-set being involved in an accident divided by its probability of being involved in an incident. Consider factor sets A and B and their corresponding measures.

4.3.1 Algorithm

The process of growing the tree by CART is summarized as follows:

- Assign the data objects to a root node.
- Select splitting criterion and explanatory variable that reduces impurity.
- Split the root node into two child nodes by dividing the data objects according to splitting criterion and independent or explanatory variable selected from the group of data objects.
- Repeat the above two steps considering each resulting node as a parent node until the maximum size tree is obtained.
- Prune the tree by eliminating a group of nodes using cross validation and cost complexity.

The process of tree building begins by splitting the root node into two child nodes. CART computes the best split by considering all probable splits for each independent or explanatory variable. The best split is obtained when the impurity function, which exists between the parent node and two child nodes, is minimized. For a classification tree, the impurity measure i(t) is computed using different criteria such as the Gini Index, Entropy Index ,and Towing rule, which determine the best split. For a regression tree, where the response variable is numerical, Least-Square where Nw(t) is the measure of the weighted number of objects in node t, wi is the value of the weighting variable for object i, ft is frequency variable, yi is the value of dependent or response variable, and y(t) is the mean of the values of objects in node t . Using best split, which reduces impurity as a splitting criterion, an over large or complex tree is grown following recursive partitioning of the nodes. This process of tree building continues until the values of all the objects in all of the terminal nodes are homogeneous or similar or when the objects in the terminal nodes reach a predefined number, in the case of a regression tree. The tree building process in case of a classification tree stops when all the objects in the terminal node belong to same class.

4.3.2 Algorithm

If D contains only leaf node training examples of the same class Cj then make T a labelled with class Cj CART

Algorithm After getting the accident dataset and candidate set, we will build the decision ladder. First, we resort to the association rules to sort the category, for example with the rule {ti}→{ci}, if we get {ti} in one document then we can make ti a decision node and make it left child labelled with ci , and for every ti , if we have the rule ,we name it effective rule. And then we start with the most frequent category to find the decision tree node. If we cannot find any effective rule, we compute the entropy of each attribute and find the most proper attribute as the decision node. Below give a detail description about how to make a the decision ladder

4.4 Proposed population heuristic for Aircraft Landing for Scheduling

The Population Heurisic that we have developed for the aircraft landing problem. For reasons relating to commercial confidentiality however, we have had to omit from our discussion a number of the computational devices we adopted to speed the convergence of our population heuristic. In order to explain our PH we introduce the following notation:

P be the number of aircraft
Ei be the earliest landing time for aircraft i
(i=1,. , P)
Li be the latest landing time for aircraft i
(i=1. . . P)
Ti be the target (preferred) landing time for aircraft I (i=1. . . P)
Sij be the required separation time (>=0) between aircraft I landing and aircraft j landing (where aircraft i lands before aircraft j), i=1, . . . ,P; j=1, . . . , P;i != j.

All of the above have known values in any given situation and typically all times are expressed in seconds. Our decision variables (what we are trying to decide) are: xi the scheduled landing time for aircraft i. The problem therefore is to decide values for the xi which lie in the time windows [Ei, Lj] and satisfy the separation criteria, whilst attempting to ensure that aircraft land at (or before) their target time. Below we discuss the elements of our PH relating to:

- Representation, how we represent a solution to the problem in our PH
- Fitness, assessing the value of a solution;
- Parent selection, choosing who shall have a child;
- Crossover, having a child from parents;
- Dealing, with the separation constraints;
- Population replacement, placing the child in the population.

5. Experimental Results

The dataset of “national” size consider 30 airports, 10 of which are hubs, 145 sectors and 22 time periods. All the instances involve 6,475 flights with 5,180 connecting flights (80%). For these instances the nominal capacity of the sector is set to 130 per period. The capacity of sectors affected by the weather front is reported in the first column of Tables 2 and 3 as a percent of this nominal capacity of 130.

In order to compare the results achieved by our PH and Cart algorithm with CSP and DS sequencing decisions the case where aircraft land in the same order as they did on the day in question, but respecting time windows and with the minimum possible separation time, we refer to as the ACTUAL case. By using the minimum possible separation time here we are able to compare the controller sequencing decisions against the PH sequencing decisions with any effects due to aircraft not landing with the minimum possible separation time removed from the comparison. In order to get a numeric evaluation of the results of our PH with CART as compared with the CSP and DS case we consider:

Time span D the time to land all the aircraft, max[xi jiˆ1;...;P]

Average delay = $-\sum_{i=1}^{P} D_i / P$

Maximum delay - Dmax[Di ji ˆ 1;...;P]

Timespan was felt to be of importance as it gives some insight into the possibility of gaining extra capacity in the long term. Average delay and maximum delay give insight into the immediate effects of improved sequencing decisions.

Comparing the PH solution shown in Figure with the ACTUAL case timespan decreased from 1894 to 1853 s (ie 41 s were saved), average delay decreased by 40 s, but the maximum delay increased by 187 s. The significance of these results depends upon your viewpoint. From the viewpoint of an aircraft passenger a reduction in average delay of 40 s is hardly significant=noticeable. However, from the viewpoint of London Heathrow, and of NATS, reducing the timespan by 41 s equates to a percentage time saving of 100(41=1894) 2.16 %, since in the ACTUAL case the last aircraft landed 1894 s after the start of the time period. Were this to be repeated across time such a saving would have the potential for Heathrow to cope with (approximately) one extra landing per h. This would be a significant improvement.

The results quoted above (reduced time span and reduced average delay, but increased maximum delay) are indicative of the tradeoffs that occur in deciding a landing schedule. Indeed, across the data sets we considered in our investigation, we commonly found that time span an average delay could be reduced, provided we were prepared to see an increase in the maximum delay. The proposed Cart algorithm mines the data very fast to give alert for pilots within fraction on seconds. We had tested the algorithm with different no of records and output timing is provided as in Table 1.

For the instance plotted in Table1 when the capacity is 40% of the nominal value, the reduction in assigned ground and airborne holding delay is more than 15%: the amount of ground delay assigned by the model drops from 699 time units without rerouting to 600 when rerouting is an option (scale on the diagram). Model with and without rerouting. In problem instances where congestion is limited, i.e., when the reduction of capacity is small, the effect of rerouting is null, as expected, because all aircraft can fly their preferred route.

Table 1: Accuracy Values

No. of DATA	*CSP Algorithm*	*DS Algorithm*	*PH & CART Algorithm*
5000	500 m sec	490 m sec	469 m sec
4000	450 m sec	439 m sec	412 m sec
3000	375 m sec	366 m sec	355 m sec
2000	235 m sec	224 m sec	201 m sec
1000	112 m sec	102 m sec	98 m sec

Table 2: Comparison of Speed and Accuracy

Algorithm	*Speed*	Accuracy
CSP	65.88 %	50.89%
DS	68.3	55.89
PH Cart algorithm	79.86	67.9%

5.1 Cart Misclassification Rate

Misclassification rate to decide on where to split the tree which is used in the pruning. The key term is the relative error (which is normalised to one for the top of the tree). The standard approach is to choose a value of a, and then to choose a tree to minimise

Ra =R+ assize

Where R is the number of misclassified points and the size of the tree is the number of end points. “cp” is a/R(root tree).

Example:

If Tree Size is R = 2000
And no of data slip in Route Tree a = 3000
ER=3000/2000 = 0.6667

Table 3: Misclassification rates of CART

Size of the Tree	*Error rate on Testing*	*Error Rate after training*
0	1.000	1.000
1	0.5210	0.5245
2	0.3120	0.3250
4	0.2960	0.3050
5	0.2920	0.3115
8	0.2810	0.3120
9	0.2785	0.3085
13	0.2675	0.3105
16	0.2615	0.3075
20	0.2545	0.3105
23	0.2495	0.3175
25	0.2470	0.3195

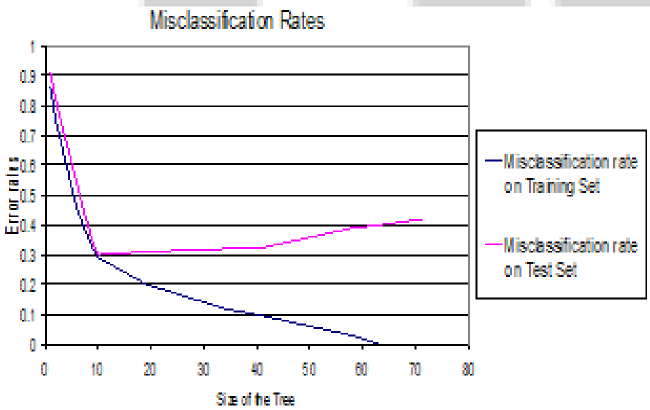

Figure 1: Misclassification rate

6. Conclusion

The proposed distributed scheduler assigns delay such that no sector along the route of flight of a scheduled aircraft is given a problem that exceeds its capability to incur delay. In routine operations, this would translate into the absence of the need for airborne holding. However, there may still be cases of sudden, unplanned drops in capacity that would require delay on aircraft beyond the capability of the remaining sectors. In such cases airborne holding or extended vectors would need to be used, but the use of the system should significantly reduce unplanned airborne holding. This paper also proposes a covered flight scheduling by analysing accident dataset. Depending on the availability of the data, the studies could be extended to other regions. The Population Heuristic that we have developed for the aircraft landing problem, proper scheduling, speed and accuracy it done with the help of the CART algorithm implemented to the scheduling and incident vectors and identified patterns of factors which are significantly associated with scheduling or with incidents. Further this proposed work can be extend to model for multiple aircraft. The problem of routing under convective weather becomes much more complex in a congested airspace because both aircraft conflicts and traffic flow management issues must he resolved at the same time.

7. Future Work

Since the system does not provide any optimization per second, future work should include the incorporation of optimization routines to improve throughput and reduce delay. Such routines will need to operate in real-time, which will be a challenge given the size of the problems and the number of flights involved. However, the system provides a framework within which optimization can be accomplished while still ensuring feasible solutions are presented to controllers and traffic managers. Further this research work can be extend to model for multiple aircraft. The problem of routing under convective weather becomes much more complex in a congested airspace because both aircraft conflicts and traffic flow management issues must he resolved at the same time. To provide a dynamic routing strategy for multiple aircraft that minimizes the expected delay of the overall system while satisfying the consideration of the constraints obtained by the sector capacity, as well as avoidance of conflicts among the aircraft. Moreover, we have used a more general weather dynamic model where it will predict accidents free zone in the future work.

References

[1] Andreatta, G., Brunetta, L., & Guastalla, G. “From ground holding to free flight: An exact approach. Transportation Science,” (2000), 34(4), 394–401.
[2] Atkin, J. A. D., Burke, E. K., Greenwood, J. S., & Reeson, D. “On-line decision support for take-off runway scheduling with uncertain taxi times at London Heathrow airport." Journal of Scheduling, (2008). 11, 323–346.

[3] Beasley, J. E., Krishnamoorthy, M., Sharaiha, Y. M., & Abramson, D. "Displacement problem and dynamically scheduling aircraft landings".(2004).
[4] Bertsimas, D. J., & Stock-Patterson, S. "The air traffic flow management problem with enroute capacities. Operations Research, (1998)". 46(3), 406–422.
[5] Idris, H., Evans, A., Evans, S., & Kozarsky, D. "Refined benefits assessment of multi-center traffic management advisor for Philadelphia and New York". Paper presented at the AIAA 4th Aviation Technology, Integration and Operations Forum, Chicago, IL, 20–22 September 2004.
[6] N. Khoussainova, M. Balazinska, and D. Suciu, "Towards Correcting Input Data Errors Probabilistically Using Integrity Constraints," Proc. Fifth ACM Int'l Workshop Data Eng. for Wireless and Mobile Access (MobiDE), 2006.
[7] P. Sistla, O. Wolfson, S. Chamberlain, and S. Dao, "Querying the Uncertain Position of Moving Objects," Temporal Databases: Research and Practice, Springer Verlag, 1998.
[8] A. Deshpande, C. Guestrin, S. Madden, J. Hellerstein, and W. Hong, "Model-Driven Data Acquisition in Sensor Networks," Proc. 13th Int'l Conf. Very Large Data Bases (VLDB), 2004.
[9] C. Aggarwal and P. Yu, "A Survey of Uncertain Data Algorithms and Applications," IEEE Trans Knowledge and Data Eng., vol. 21, no. 5, pp. 609-623, May 2009.
[10] T. Bernecker, H. Kriegel, M. Renz, F. Verhein, and A. Zuefle, "Probabilistic Frequent Itemset Mining in Uncertain Databases," Proc. 15th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2009.
[11] N. Dalvi and D. Suciu, "Efficient Query Evaluation on Probabilistic Databases," Proc. 13th Int'l Conf. Very Large Data Bases (VLDB), 2004
[12] T. Jayram et al., "Avatar Information Extraction System," IEEE Data Eng. Bull., vol. 29, no. 1, pp. 40-48, Mar. 2006.
[13] R. Cheng, D. Kalashnikov, and S. Prabhakar, "Evaluating Probabilistic Queries over Imprecise Data," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2003.
[14] N. Dalvi and D. Suciu, "Efficient Query Evaluation on Probabilistic Databases," Proc. 13th Int'l Conf. Very Large Data Bases (VLDB), 2004.
[15] J. Huang, "MayBMS: A Probabilistic Database Management System," Proc. 35th ACM SIGMOD Int'l Conf. Management of Data, 2009.
[16] R. Jampani, L. Perez, M. Wu, F. Xu, C. Jermaine, and P. Haas, "MCDB: A Monte Carlo Approach to Managing Uncertain Data," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2008
[17] M. Mutsuzaki, "Trio-One: Layering Uncertainty and Lineage on a Conventional DBMS," Proc. Third Biennial Conf. Innovative Data Systems Research (CIDR), 2007.
[18] R. Agrawal, T. Imieli_nski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. ACM SIGMOD Int'l Conf. Management of Data, 1993.
[19] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," Proc. ACM SIGMOD Int'l Conf. Management of Data, 2000.
[20] C. Aggarwal, Y. Li, J. Wang, and J. Wang, "Frequent Pattern Mining with Uncertain Data," Proc. 15th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2009
[21] C.K. Chui, B. Kao, and E. Hung, "Mining Frequent Itemsets from Uncertain Data," Proc. 11th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), 2007.
[22] L. Sun, R. Cheng, D.W. Cheung, and J. Cheng, "Mining Uncertain Data with Probabilistic Guarantees," Proc. 16th ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, 2010
[23] C.J. van Rijsbergen, Information Retrieval. Butterworth, 1979.

Author Profile

R. S. Priyaa Completed her BCA in 2009 and the master degree in computer technology in 2011. She is currently doing M.Phil in Sri Jayendra Saraswathi College of arts and science. Her area of interest is data mining.

S. Saravana Kumar Completed M.Sc., M. Phil., MBA, PGDBA. He is currently working as Assistant professor in Department of Inofrmation Technology, Sri Jayendra Saraswathi College of arts and science. His Teaching Experience is 9 years. Area of specialization is Data Mining and Networking.