Development of a First-Come First-Served Departure Scheduler

Bae-Seon Park and Hak-Tae Lee

Inha University, 100, Inha-ro, Incheon, Republic of Korea

Abstract

This study presents the development of a first-come first served departure scheduler. This algorithm computes departure and arrival times as well as sector entry and exit times. The algorithm can be applied to any system that can be modelled with a node-link structure. In addition to its versatility, the execution is very fast. It can compute more than tens of thousands of schedules in minutes.

Departure scheduling has been performed over flights departing from and arriving to the airports in Republic of Korea. Initial flight schedule was generated using recorded flight data. About 730 flights have been scheduled using two different airport capacities at Incheon, Gimpo, and Jeju airports. With departure and arrival capacities fixed to 50 aircraft per hour, the average delay was eleven minutes. When the capacities were increased to 100, average delay was reduced to 90 seconds per flight. It shows that the delays are sensitive to the capacities of the three busiest airports in Republic of Korea.

Keywords: First-come first-served scheduling, Node-link representation, Traffic flow management.

Introduction

As air travel demand constantly increasing, current capacities at many of the busy airports are close to or exceeding their limits. However, it is very costly to build new airports or expand existing airports to increase capacity. Limited sector capacities can also cause delay, especially in the presence of severe weather. To alleviate this problem, traffic flow management techniques are being researched, led by researchers in USA and Europe. One of the key components of traffic flow management is scheduling optimization.

To reduce delays, efficient scheduling algorithms are being developed. Ref. 1 proposed a closed-form solution based on the first-come first-served (FCFS) principle using constraint algebra. It can efficiently handle multi-point scheduling problem. Ref. 2 expanded the results of Ref. 1 so that the scheduling algorithm can be applied to a general node-link structure and can handle time varying airport and sector capacity. Ref. 3 developed a similar multi-point scheduling algorithm for the integration of dynamic airspace and traffic flow management. Ref. 4 proposed a methodology based on optimization using mixed integer programming.

In this paper, a departure schedule based on the algorithms of Ref. 1 and 2 are developed. A special data type called ‘interval’ is introduced and related operations are developed to efficiently handle the constraints. An example scheduling problem was solved using the actual trajectory data over Incheon Flight Information Region (FIR) with different airport capacity constraints to compute and analyze the delays. Finally, future research items are proposed.

FCFS Departure Scheduler

Interval Operations

A time interval is the basic element that constitutes the FCFS departure scheduling algorithm. An ‘Interval’ can represent a time slot that a given airport or airspace is open for an aircraft to enter or closed. An interval has beginning and ending time where the ending time should be greater or equal to the beginning time. Fig. 1 illustrates the concept of this interval.

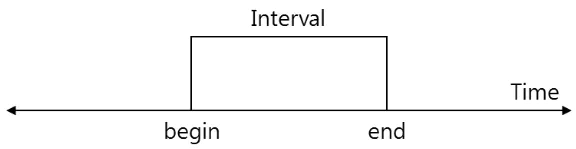


Fig. 1: Interval on a horizontal time line

The scheduling algorithm progresses by updating a set of intervals called the ‘Intervals’. Similar to the set operations, intersection, union and complement are defined. However, depending on the specific use of the interval, it can be a closed interval or an open interval. The openness and closeness of an interval depends on the whether it represents an available or unavailable time slot on a node or a link. Fig. 2, 3 shows the difference between intersection and union operations for two open and closed intervals where the ending time of one interval is equal to the beginning time of the other. In this scheduling algorithm, unavailable slots are represented by open intervals (b, e), and available slots are represented by closed intervals [b, e].

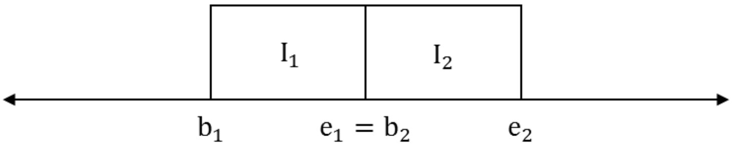


Fig. 2: Interval operation

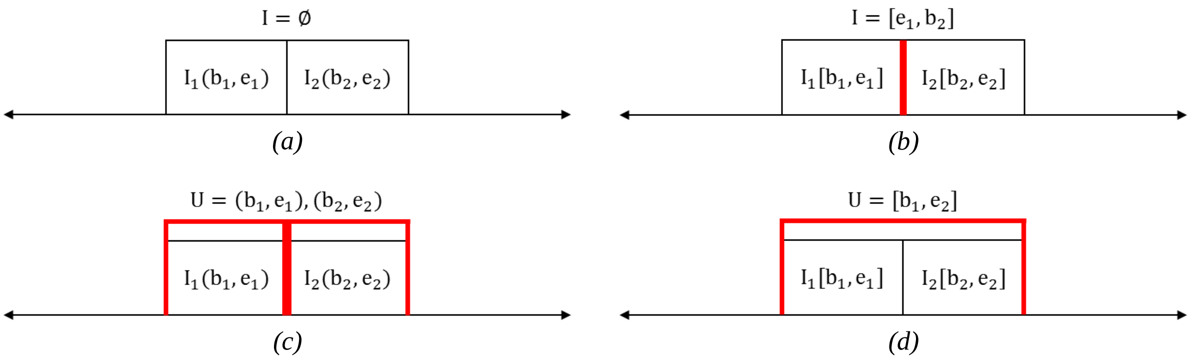


Fig. 3: Two cases of intersection and union set operations

FCFS Algorithm

The FCFS scheduling algorithm first performs forward propagation to find the earliest arrival time, and then from the fixed earliest arrival time, the constraints are propagated backward to find the entry and exit schedules including the earliest departure time. For this, the algorithm requires original flight schedule and shown in Table 1, aircraft departure rates (ADR) and aircraft arrival rates (AAR) as shown in Fig. 4, and sector capacities as shown in Fig. 5. Airport and sector capacities can be time varying to reflect various operational conditions.

List of flight schedules shown in Table 1 includes the identification generally denoted by the call sign of the flight, scheduled departure time at the departure airport, entry, exit, and transit times at each sector that the flight passes through, and the scheduled arrival time at the arrival airport. The first and the last enroute airspaces are considered to be terminal areas. Because the separation within the terminal area is ensured by arrival and departure sequencing and spacing, maximum number of aircraft is not a proper metric for the capacity. For this study, the capacity constraints inside the terminal airspace are not enforced. The flight path for the flight F01 shown in Table 1 is A00 – TRACONTA00 – Z01 – TRACONTA02 – A02. For better scheduling results, small changes in the nominal transit times are allowed. This is represented by the terms, speed-up and slow-down. They do not necessarily means changing the flight speed, but can be achieved by straightening or stretching the flight paths as well as changing the flight speeds. Since the paths tend to be close to great circle and aircraft generally cannot significantly increase the flight speed due to wave drag, the limit for the speed-up is set to one percent. The limit for the slow-down is set to five percent, because it can be achieved through path stretch.

Table 1: Flight schedule input

* Time unit : millisec

flightID	entryTime	exitTime	transitTime	upperStreamSector	currentSector	downStreamSector	speed-Up(%)	slow-Down(%)
F01	46860000	47168570	308570	----	A00	TRACONTA00	0	4
F01	47168570	47991430	822860	TRACONTA00	Z01	TRACONTA02	1	3
F01	47991430	48300000	308570	TRACONTA02	A02	----	0	3
F02	45720000	46028570	308570	----	A30	TRACONTA30	0	4
F02	46028570	46748570	720000	TRACONTA30	Z20	Z10	1	5
F02	46748570	47571430	822860	Z20	Z10	TRACONTA00	1	5
F02	47571430	47880000	308570	TRACONTA00	A00	----	1	3

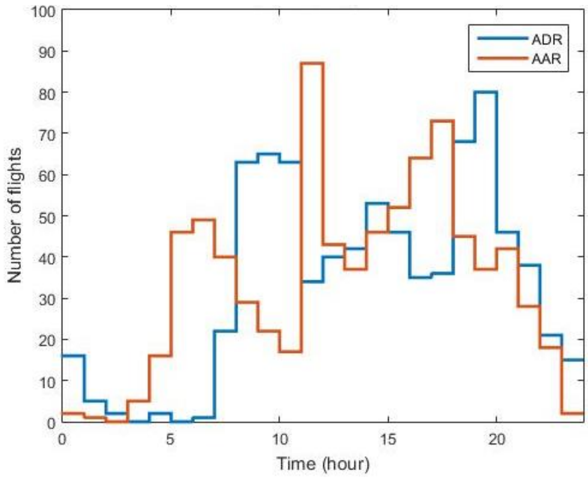


Fig. 4: The example of airport capacity (AAR and ADR) with respect to time

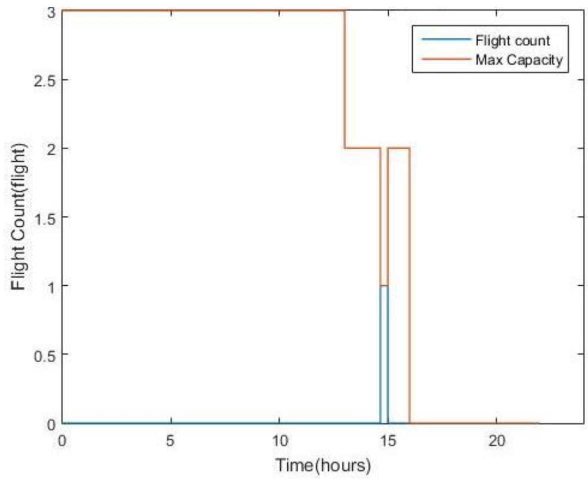


Fig. 5: The example of sector capacity

FCFS scheduling computes the schedule of a flight based on its priority. For this study, flights are prioritized based on its scheduled departure time. Fig. 6 shows the progression of an example flight F02. If F02 departs airport A30 at its scheduled departure time and flies according to the original schedule, it will arrive at A00 at its scheduled arrival time along the green straight line. Departure and arrival airports as well as the sector boundaries can be considered nodes denoted by N. Nodes are characterized by having zero transit time. Terminal airspaces and sectors can be considered to be links denoted by L. They have finite transit times. For this study, node constraints are enforced only at the departure and arrival airports. Within the enroute airspace, node constraints can be enforced at metering fixes if necessary.

On the timelines of nodes and links, available and unavailable slots alternates, which are represented by intervals. As shown in Fig. 7, the scheduling process starts with the first interval from the intervals. If the forward propagation becomes blocked by any of the downstream constraints, propagation restarts from the next interval of the upstream component.

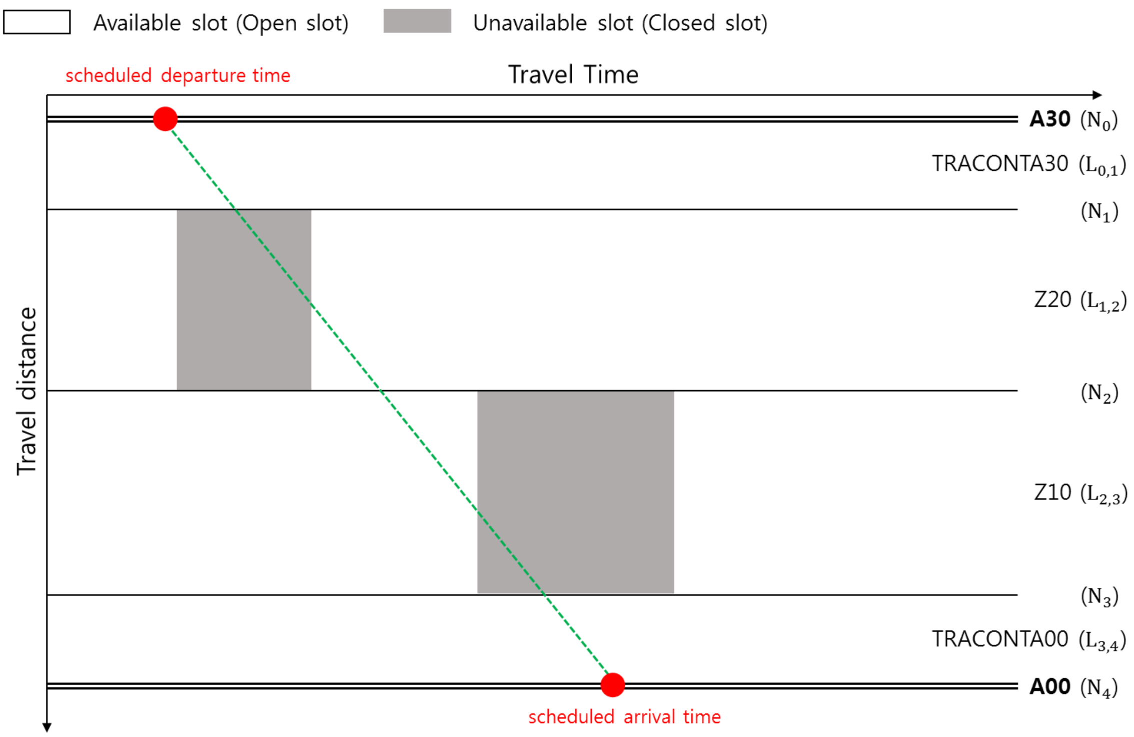


Fig. 6: The schedule of F02 in node-link structure

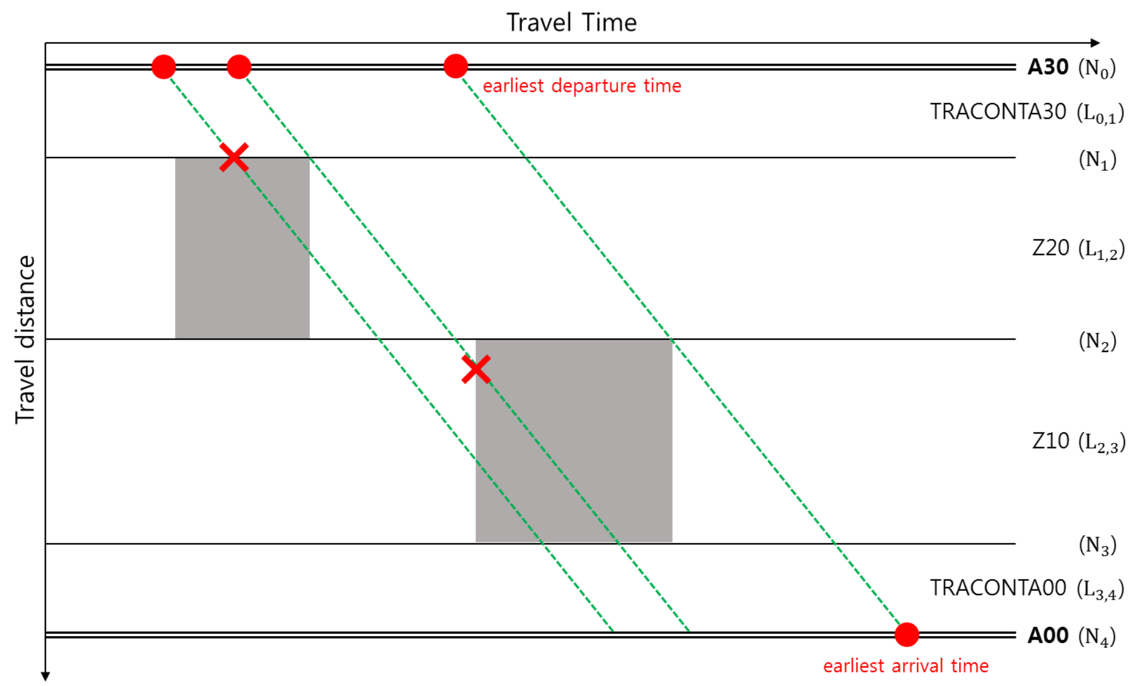


Fig. 7: The order of FCFS scheduler process

Forward Propagation

Node N_0 denotes the starting node and the node N_N denotes the ending node. For the first flight F02 that is being scheduled, there is no constraint at N_0 . For this study, it is assumed that aircraft cannot depart earlier than its scheduled departure time. So, the possible entry slot for F02 at N_0 can be computed by Eqn 1, also shown in Fig. 8. $P_{entry_{N_n}}$ is possible entry slots, O_{N_n} is open slots, and $S_{entry_{N_n}}$ is entry slots at the node N_n , where n equals to 0 or N.

$$P_{entry_{N_n}} = O_{N_n} \cap S_{entry_{N_n}} \quad (1)$$

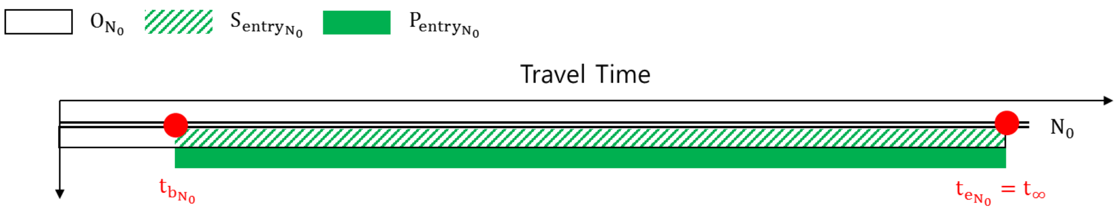


Fig. 8: The possible entry slots at the start node N_0

Once the possible entry slots at a node is obtained, possible exit slots are computed by adding minimum and maximum transit times Δt_u and Δt_d as shown in Eqn 2. $t_{1_{entry}}$, $t_{2_{entry}}$ are beginning and ending time of each possible entry slot, $t_{1_{exit}}$, $t_{2_{exit}}$ are those of each possible exits slot.

$$\begin{aligned} t_{1_{exit}\,N_n} &= t_{1_{entry}N_n} + \Delta t_{u_{n,n+1}} \\ t_{2_{exit}\,N_n} &= t_{2_{entry}N_n} + \Delta t_{d_{n,n+1}} \end{aligned} \quad (2)$$

Fig. 9 shows the all the possible locations of beginning and ending time slots for entry exit with respect to a link constraint. Among the nine possible combinations, five cases lead to unavailable exit slots. Four possible solutions are listed in Table 2.

Table 2: Possible exit slot of Fig.9

Link constraint		End time ①	End time ②	End time ③
Begin time	①	-	$[t_{1_{exit}}, t_e]$	$[t_{1_{exit}}, t_{2_{exit}}]$
Begin time	②	-	$[t_b + \Delta t_u, t_e]$	$[t_b + \Delta t_u, t_{2_{exit}}]$
Begin time	③	-	-	-

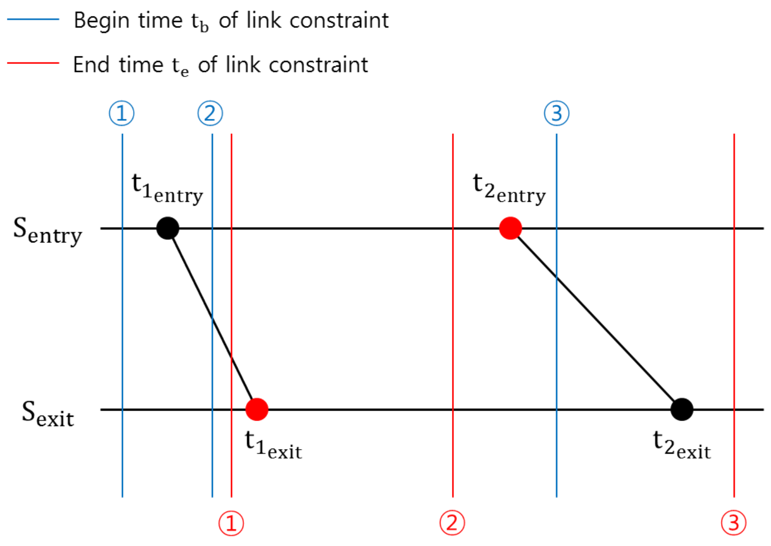


Fig. 9: Several cases of link constraints

The union of each possible exit slot $p_{exit_{N_n}}$ is the possible exit slots $P_{exit_{N_n}}$, and this is becomes the possible entry slots $P_{entry_{N_{n+1}}}$ at the next node N_n ($P_{exit_{N_n}} = P_{entry_{N_{n+1}}}$).

Fig. 10 shows that the flight has reached to the end node N_4, by operating the process above repeatedly. And the forward propagation is completed by obtaining the possible entry slots P_{entryN_4} at N_4, using Eqn 1. The earliest time of P_{entryN_4} is determined to be the earliest arrival time, t_{EA}.

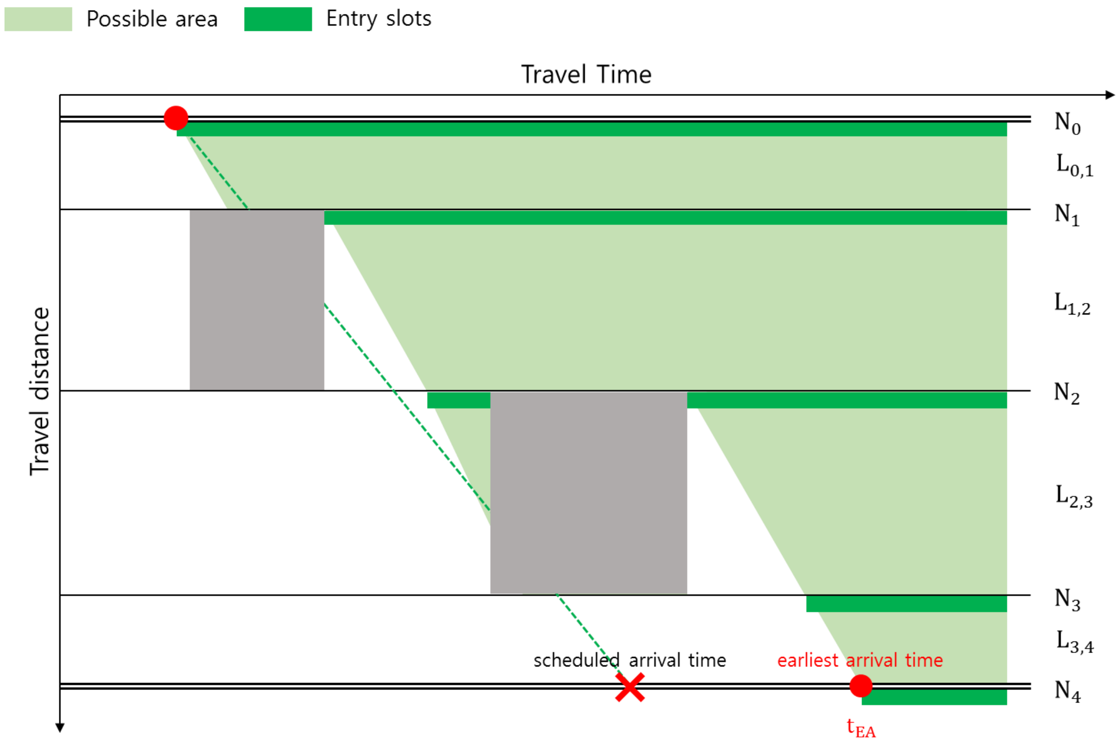


Fig. 10: The result of forward propagation

Backward Propagation

Once the earliest arrival time is determined, backward propagation is started from the end node N_4 to the start node N_0. The process from N_4 to N_3 is shown in Fig. 11 and Eqn 3. $S_{F_{N_n}}$ is entry slots from forward propagation, and $S_{B_{N_n}}$ is entry slots $[t_{EA} - \Delta t_{d_{n,n+1}}, t_{EA} - \Delta t_{u_{n,n+1}}]$ from backward propagation at node N_n.

$$S_{entryN_n} = S_{F_{N_n}} \cap S_{B_{N_n}} \tag{3}$$

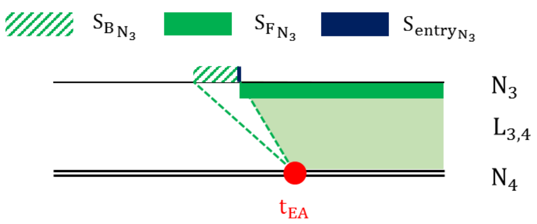


Fig. 11: Backward propagation from N_4 to N_3

The earliest departure time t_{ED} is determined by operating the process above repeatedly to the start node N_0, as shown in Fig. 12. A possible area in the Fig. 12 is an area overlapping the possible area determined by forward propagation, and the flight can be operated without violating the constraints in this area.

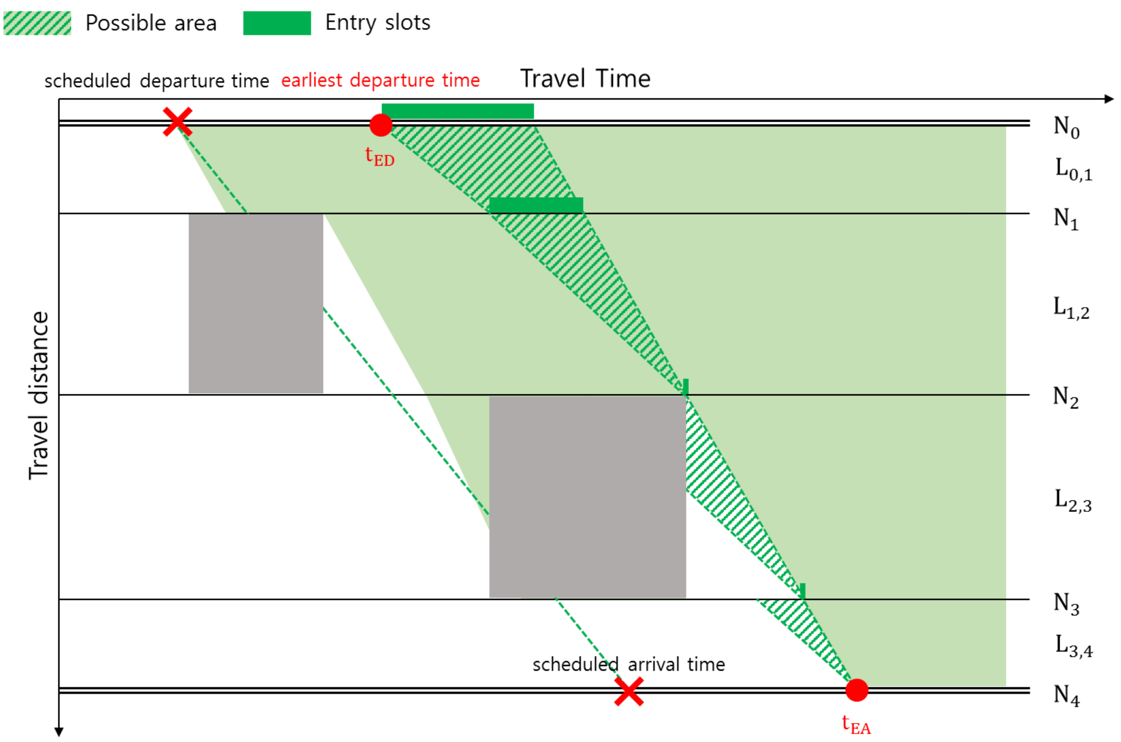


Fig. 12: The result of backward propagation

At the start node N_0 and the end node N_4, which are the departure and the arrival airports, the unavailable slots are updated by blocking time intervals before and after the determined departure and arrival times as shown in Fig. 13. dt is a time period associated with the corresponding ADR or AAR and computed by dividing the unit time by the ADR or AAR as shown in Eqn 4. If the ADR is 60, dt is 1 minute. This ensures that ADR or AAR do not exceed the maximum given value at any time.

$$dt = \frac{unit\, time\, of\;\; ADR\, or\; AAR}{ADR\, or\; AAR} \tag{4}$$

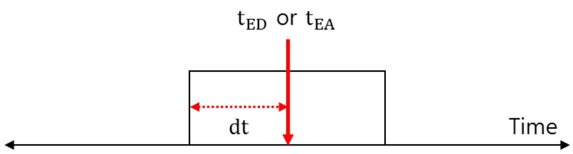


Fig. 13: Size of unavailable slots at the airports

Sector status should also be updated. For the sectors that a flight passes through, using the scheduled entry and exit time, the number of aircraft count during this time will be increased by 1 at the given sector. This updated sector count will be compared to the sector constraints and new available and unavailable slots will be updated as shown in Fig. 14.

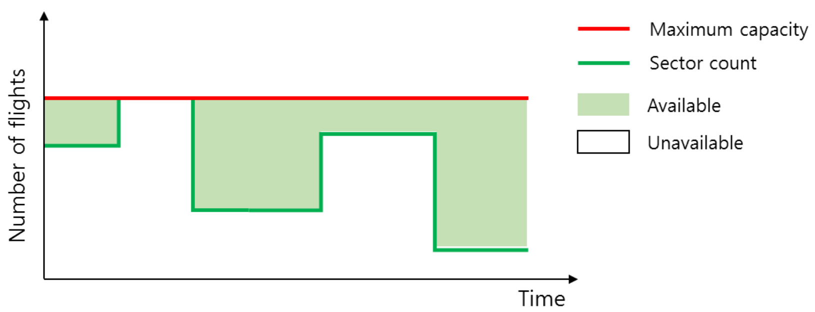


Fig. 14: Available or unavailable slot update in sector

Simulations

Trajectory data from April 1st 2015 to April 2nd 2015 that spans 48 hours were processed to obtain the initial flight schedule. Since most of the data ends outside of the Incheon FIR boundary, four virtual TMAs were placed outside Incheon FIR (ICN FIR). For inbound international flights to ICN FIR, the first trajectory point was considered to be departure airport with unlimited capacity. For outbound flights, the last point was considered to be arrival airport with unlimited capacity. With these assumptions, it is possible to schedule flights without the full trajectory information about the international flights and examine the impact of capacity constraints within Incheon FIR. Fig. 15 shows the first land last points of the trajectory data and the airspace boundaries around ICN FIR.

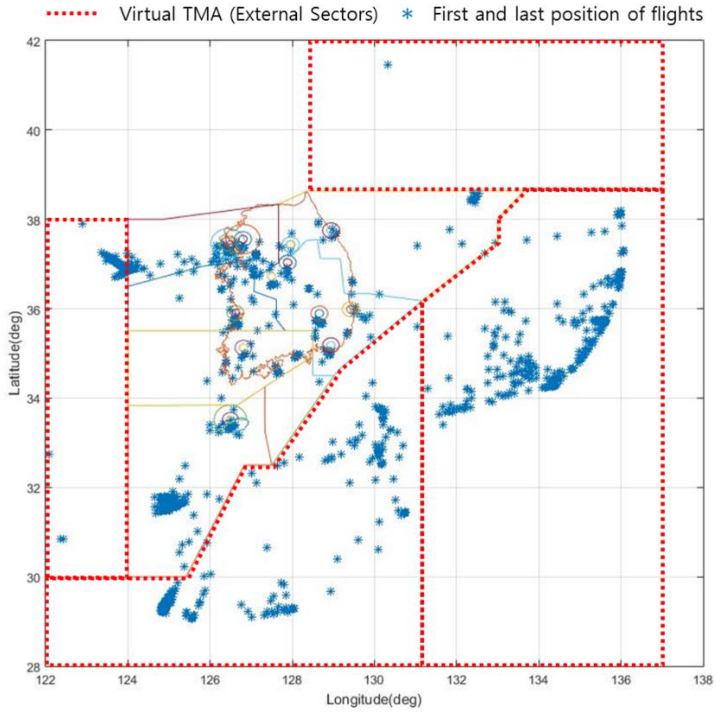
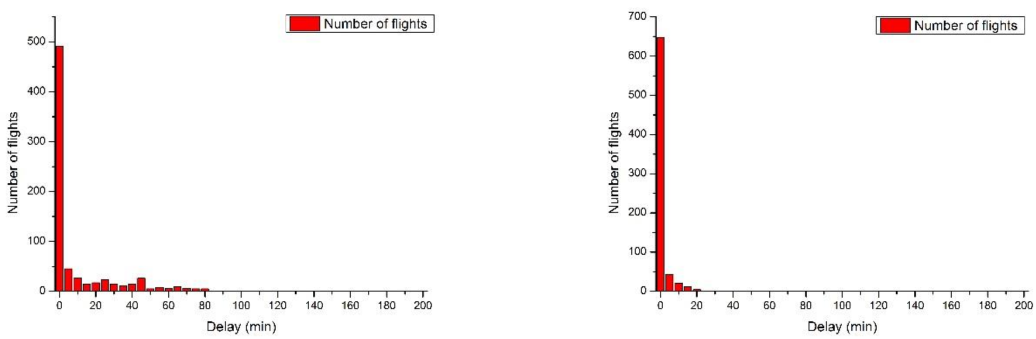


Fig. 15: The range of flight data and sectors in Republic of Korea

Among the recorded trajectory, about 730 flights were used for the simulations. For the virtual and international airports, the ADR and AAR were set to 3600 to represent unlimited capacity conditions. ADR and Aar for all other airports in republic of Korea were fixed to 100. To study the impact of the busiest airports, another case where the ADR and AAR of the three busiest airport, Incheon, Gimpo, and Jeju, were reduced to 50 was computed. Sector capacities were fixed at ten for all the sectors within Incheon FIR.

Fig. 16 (a) shows the delay distribution for the reduced capacities at the three airports. Average delay was eleven minutes. Fig. 16 (b) shows the delay distribution for the higher capacity case. The average delay was significantly reduced to 90 seconds. Moreover, maximum delay was about 20 minutes compared to about 80 minutes for the reduced capacity case.



(a) *(b)*

Fig. 16: Delay Distributions

Fig. 17 shows an example at the South West Sector that the schedule creates a schedule that satisfies the given capacity constrains at the sectors. Fig. 18 shows the capacity constraints and actual ADR and AAR at Incheon Airport. Fig. 18 (a) and (b) show the computed ADR and AAR for the constraint of 50. Fig. 18 (c) and (d) show the computed ADR and AAR for the constraint of 100.

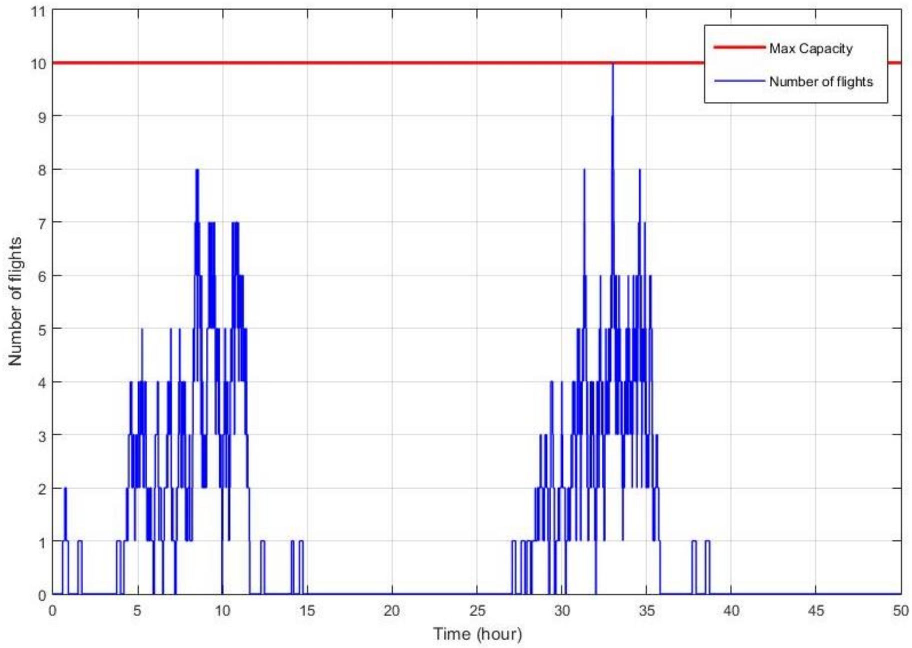


Fig. 17: Sector capacity constraints (South West Sector)

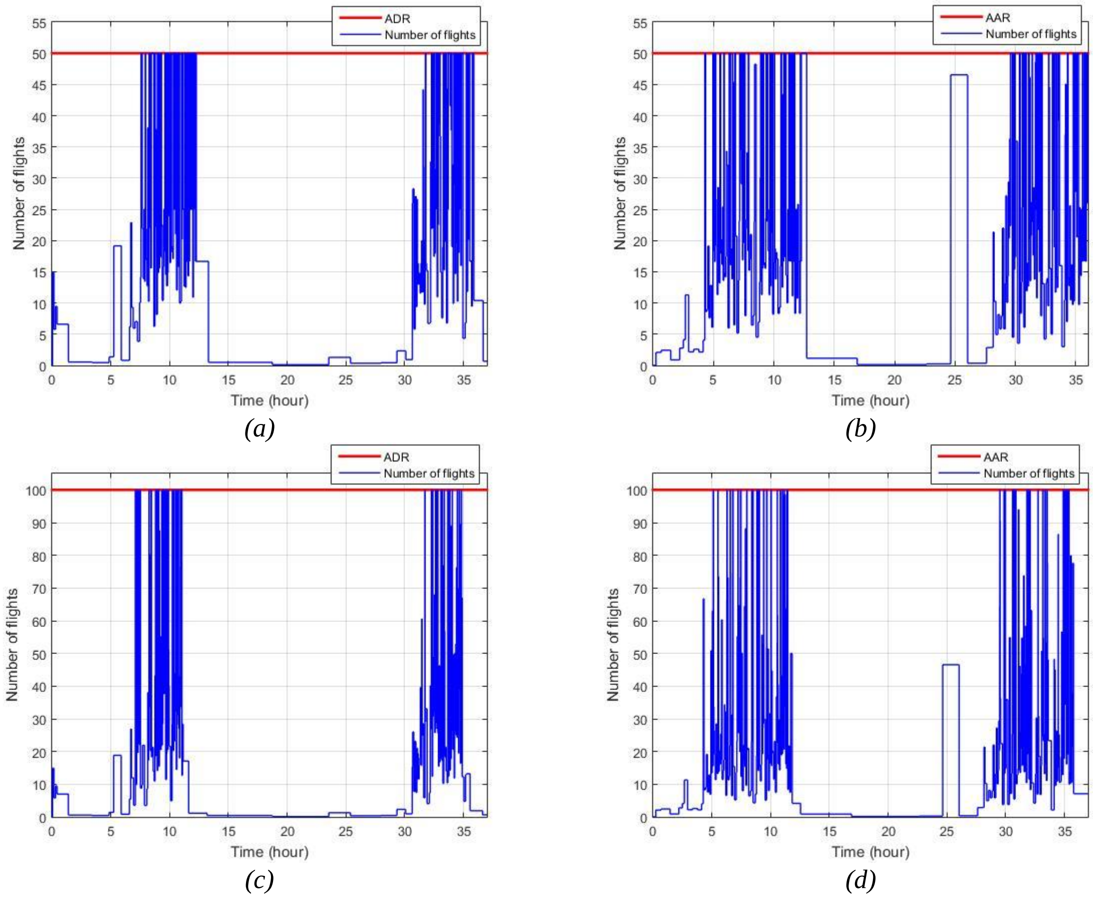


Fig. 18: Airport ADR and AAR constraints (ICN)

Conclusions

In this paper, an efficient FCFS scheduling algorithm that can handle multiple constrains along the path were developed. Using this scheduler, flights originating and terminating at Republic of Korea was scheduled. For the sample capacity constrains used for this study, average delays ranging from 90 seconds to 11 minutes were achieved. In the future, scheduling studies using more realistic capacity constrains will be performed, to analyze the choke point and sensitivity of the delay with respect to airport and airspace capacities. In addition, the same scheduling algorithm will be applied to a node link structure on airport surface.

References

1. Meyn, L., "A Closed-Form Solution to Multi- Point Scheduling Problems," *AIAA Modeling and Simulation Technologies Conference*, Toronto, Ontario, Aug. 2-5, 2010.

2. Park, C., Lee, H.T., and Meyn, L., "Computing Flight Departure Times Using an Advanced First-Come First-Served Scheduler." *12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference* and *14th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*. 2012.

3. Palopo, K., Chatterji, G., and Lee, H.-T., "Interaction of Airspace Partitions and Traffic Flow Management Delay," *10th AIAA Aviation Technology, Integration and Operations (ATIO) Conference*, Fort Worth, Texas, Sep. 13-15, 2010.

4. Bertsimas, D. and Patterson, S. S., "The Air Traffic Flow Management Problem with Enroute Capacities", *Operations Research*, Vol. 46, No. 3, May-June 1998, pp. 406-422.