AIAA AVIATION Forum
5-9 June 2017, Denver, Colorado
17th AIAA Aviation Technology, Integration, and Operations Conference

AIAA 2017-4259

Airport Surface Movement Scheduling with Route Assignment Using First-Come First-Served Approach

Bae-Seon Park*, Hyeonwoong Lee†, Seon-Young Kang‡, and Hak-Tae Lee§

Inha University, Incheon, Republic of Korea 22212

This paper describes a first-come first-served scheduler that is being developed for integrated departure and arrival management at airports. Previous first-come first-served scheduler developed for Traffic Flow Management research was enhanced to handle directionality constraints at taxiway links as well as crossing constraints at taxiway junctions. Instead of using predetermined taxi routes, a route assignment function is added. For the given flight, the scheduler finds upto five taxi routes. Scheduling is performed for each route, and the route and schedule that result in the minimum delay for the given flight is selected. The scheduler was tested with a historic surface traffic at Incheon International Airport on April 1st, 2015 that contains about 800 flights in 24 hour period. Two different prioritization strategies were compared. When the priority was based solely on the scheduled departure or arrival times, the average delay was 6.3 minutes. If priority was given to arrival flights, average delay for arrival flights was 1.3 minutes while that of departure flight was increased to 13.4 minutes. It was also discovered that taxiway crossing constraints have little impact on the delays.

I. Introduction

Currently, the Ministry of Land, Infrastructure, and Transportation of Republic of Korea is developing an integrated departure and arrival management system to relieve the congestion at the busiest airports in the country. The Extended First-Come First Served (EFCFS) scheduling algorithm described in this paper is one of the several scheduling schemes that are being investigated for the program.[1] An approach based on generalized dynamic programming was suggested to find the optimal solution for the runway scheduling problems.[2] To solve scheduling problems for entire airport surface, approaches based on Mixed Integer Linear Programming (MILP) and Receding Horizon (RH) were used.[3,4] MILP was also applied to a deterministic departure scheduling problem at runways and used to solve a multiple taxi route scheduling.[5–7] At NASA, a decision support tool, Spot and Runway Departure Advisor (SARDA), was developed to efficiently manage departure and was test at Dallas Fort-Worth International Airport and Charlotte Douglas International Airport. With various human machine interfaces, NASA performed fast-time and real-time Human-in-The-Loop (HITL) simulations.[8–10]

In this paper, use of EFCFS method that has significantly lower computational cost than optimization based approaches is investigated. Theoretically, the basic algorithms used for the FCFS scheduler that was initially developed for Traffic Flow Management (TFM) scheduling[11] can solve any scheduling problem if the problem can be formulated in a node-link structure. Links are characterized by having finite transit times, and link constraint is specified by maximum number of aircraft that can be accommodated in a link. Nodes are characterized by having zero transit times, and node constraint is specified by maximum number of aircraft that can pass through the node in unit time. However, for surface scheduling, a link that represents a taxiway segment has an availability that is dependent not only on the number of aircraft in the link but also on the direction of the existing traffic. This issue is solved by introducing positive and negative aircraft count for each link. In addition, rate constraints are enforced at nodes that represent taxiway junctions so that conflict free schedules can be generated.

*Research Assistant, Dept. of Aerospace Engineering, Bldg. 4, Rm. 232, Student Member AIAA, pearship6842@gmail.com
†Research Assistant, Dept. of Aerospace Engineering, Bldg. 4, Rm. 232, Student Member AIAA, 22161558@inha.edu
‡Research Assistant, Dept. of Aerospace Engineering, Bldg. 4, Rm. 232, Student Member AIAA, seonyoung4@gmail.com
§Assistant Professor, Dept. of Aerospace Engineering, Bldg. 2S, Rm. 321, Member AIAA, haktae.lee@inha.ac.kr

Generally, the current surface operation is based on a set of predetermined routes based on runway configurations and gates. Since, the integrated departure and arrival management operation includes the concept of dynamic routing, a simplified model of route assignment is implemented. For given gate and runway pair, minimum distance route is computed. Based on this minimum distance route, alternate routes are generated by removing a single link from the minimum distance route. Upto five routes including the initial minimum distance route become the candidates. For a given flight with a gate and runway pair, list of transit times for each candidate route is calculated and then scheduling is performed. The route with minimum delay is selected, which is similar to the method describe in Lee et al.[12]

Input for the schedule was generated using the recorded data at the Incheon International Airport on April 1st, 2015 from 0:00 to 24:00. About 800 flights were schedule with two prioritization strategies. In the first strategy, flights were prioritized based solely on original scheduled times providing no preference between arrival and departure. With the given condition, it showed an average delay of 6.3 minutes. In the second strategy, priority was given to arrival flights. It resulted in significantly smaller delays for the arrival flights, average of 1.3 minutes, but increased the average departure delay to 13.4 minutes including some flights being delayed over 60 minutes.

To investigate the impact of enforcing conflict free taxiway junction crossing, two scheduling results were compared, one with nominal junction crossing rates and the other with one quarter of the nominal rate. The two results show little difference suggesting that conflict can be tactically handled so that it is not necessary to include the constraints when generating the schedule.

Following this introduction, Section II describes the scheduling algorithm in detail. Section III presents the scheduling results at the Incheon International Airport with historic flight. Section IV concludes the paper.

II. Extending First-Come First-Served Approach

For airborne aircraft, the flight progression can be represented as a sequence of airspaces such as departure airport, departure terminal area, enroute sectors, arrival terminal area, and arrival airport. This path can be simplified to a node-link structure as shown in Fig. 1. Departure and arrival airports as well as airspace boundaries become nodes, and the transit path in each sector or terminal area become links. Nodes are characterized by instant transit. So a rate constraint can be imposed at nodes such as Aircraft Departure Rate (ADR) or Aircraft Arrival Rate (AAR). Links are characterized by finite transit time, and the link constraint is the maximum number of aircraft that can be accommodated in the link such as the Monitor Alert Parameter (MAP) of a sector. To solve a TFM problem, node constraints at airports are enforced in the form of AAR and ADR but no constraints are enforced at nodes that represent the airspace boundaries.

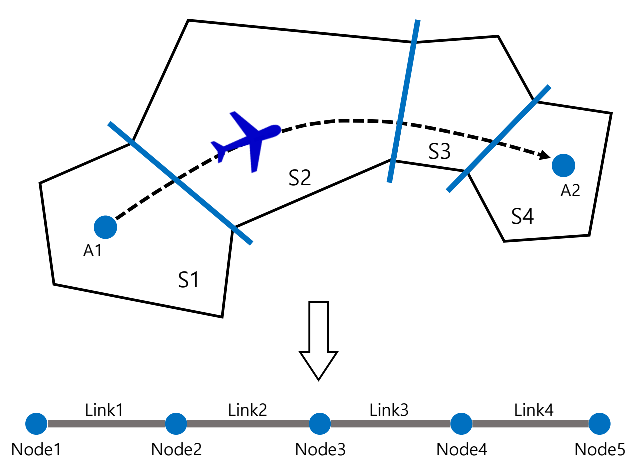


Figure 1. Typical flight path and node-link structure for enroute operation.

Once a scheduling problem is formulated in this node-link structure, and the scheduling priority is determined, earliest arrival time and corresponding earliest departure time of each flight can be computed using the FCFS scheduler described in Park and Lee.[11,13]

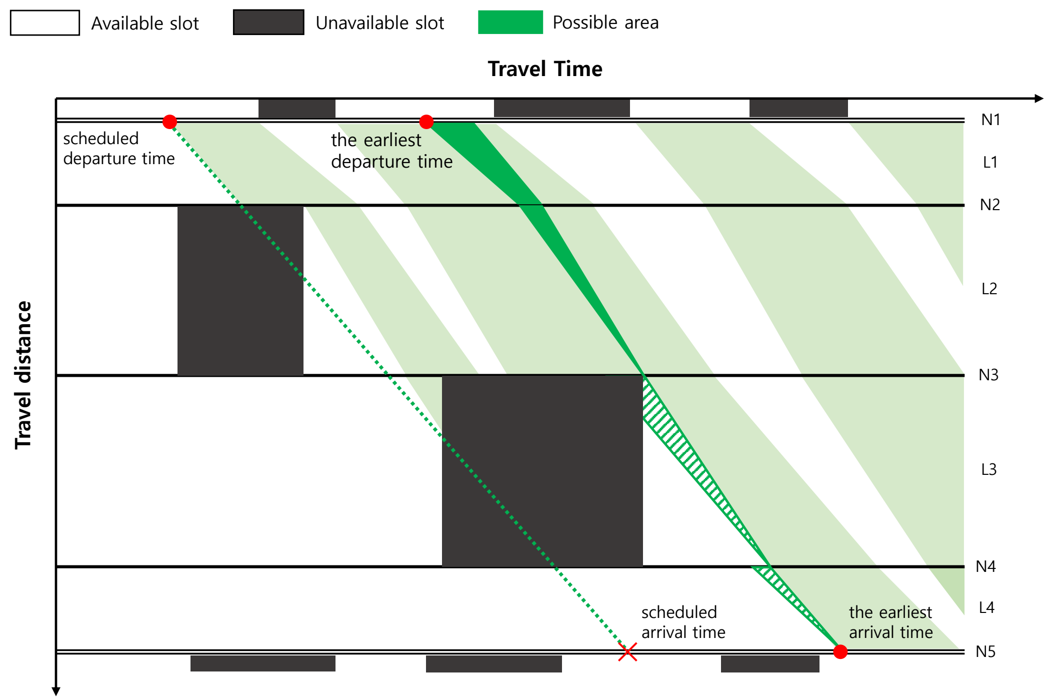


Figure 2. Solution process of a FCFS departure scheduler

FCFS scheduling algorithm shown in Fig. 2 uses a two-step process. During the initial forward propagation step, earliest arrival time is found that satisfies all the constraints at nodes and links that the flight passes through. In this example, ADR constraint is applied at node N1, maximum aircraft count constraints are used at links L2 and L3, and AAR constraint is applied at node N5. Once the earliest arrival time is determined, the earliest departure time is computed through the backward propagation. When the complete schedule is determined for the given flight, all the node and link counts are updated and the process is repeated with the next flight. As shown in Fig. 2 as different slopes in the propagation lines, small variation in transit time that represents change of flight speed or lengthening or shortening of fight route is allowed.

Table 1 shows an example input that has unimpeded entry, exit, and transit times for all the links that the flight passes through. The range of acceptable transit time change is also specified for each link. Scheduler will determine the actual transit time to achieve minimum delay.

Table 1. Example nominal flight schedule. (Time is measured in seconds from the midnight.)

Flight ID	Entry Time	Exit Time	Transit Time	Previous Link	Current Link	Next Link	Speed-up	Slow-down
F01	29100	29127	27	XXXX	N01	L01	1	1
F01	29127	29140	13	L01	L02	L03	0	2
F01	29140	29181	41	L02	L03	L04	0	2
F01	29181	29195	14	L03	L04	L05	1	1
F01	29195	29218	23	L05	N02	XXXX	0	1

A. Airport Surface Movement

1. Directions of Airport Taxiway Link

As shown in Fig. 3, the surface movement paths can naturally be modeled as a node-link structure.[14] However, it is necessary to identify the properties of this geometric node-link structure if the FCFS scheduling algorithm is to be applied.

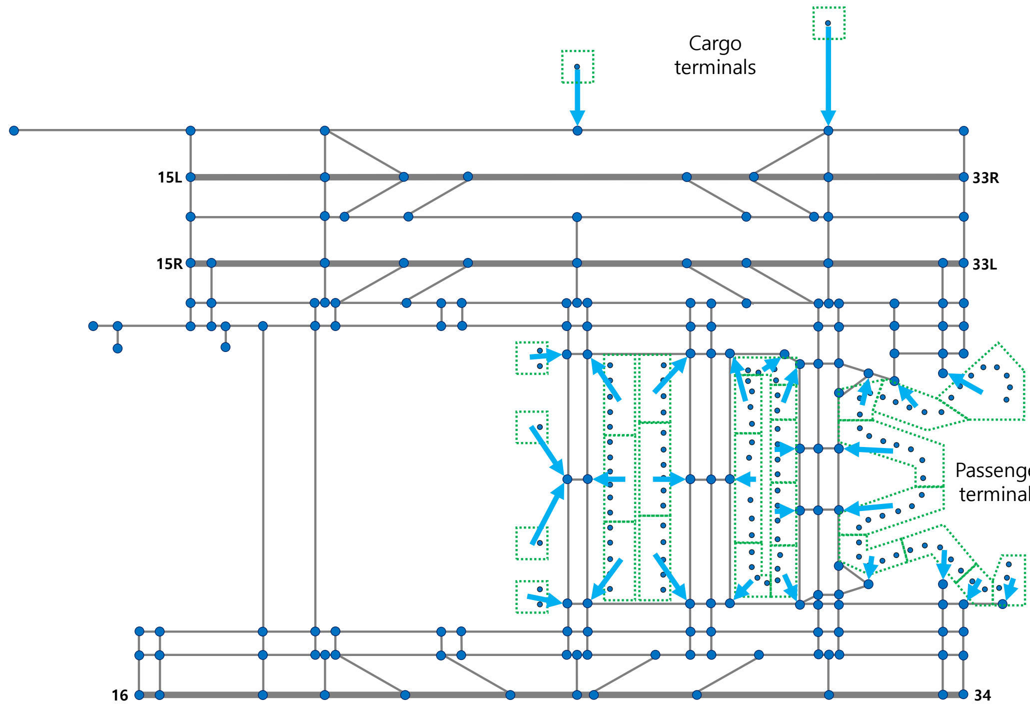


Figure 3. Node-link model of Incheon International Airport.

One of the fundamental differences is that the links for taxiway or taxilane segments have directionality unlike the links representing transit through a sector. As shown in Fig. 4, if an aircraft is already traversing on a taxiway segment, another aircraft moving in the same direction can enter the segment as far as a proper distance is maintained between the two aircraft. However, no aircraft can enter the taxiway segment in the opposite direction until the segment becomes empty.

Previous FCFS scheduler determines the availability of link for any other aircraft to enter based only on the current count of aircraft. To handle the directionality, negative aircraft count is devised. Each taxiway segment is assigned with an arbitrary direction. Number of aircraft in the assigned direction is considered positive count and the number of aircraft in the opposite direction is considered negative count. Figure 5 shows an example of different availability based on direction of flight. For the positive direction in Fig. 5(a), the link is available when the count is positive and is smaller than the maximum positive count. For the negative direction in Fig. 5(b), the link is available when the count is negative and is larger than the minimum negative count. With the negative count concept, it is possible to determine the link availability based on the current count, maximum count, and direction.

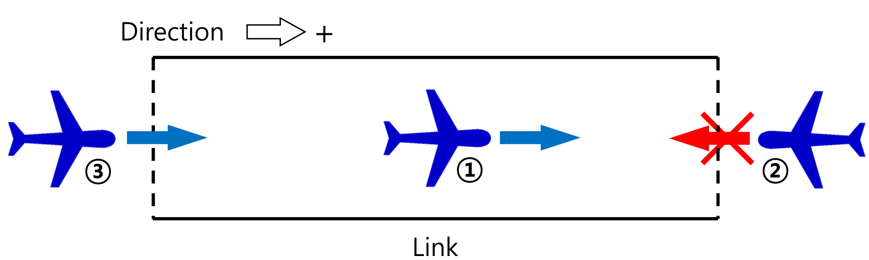


Figure 4. Link availability based on direction.

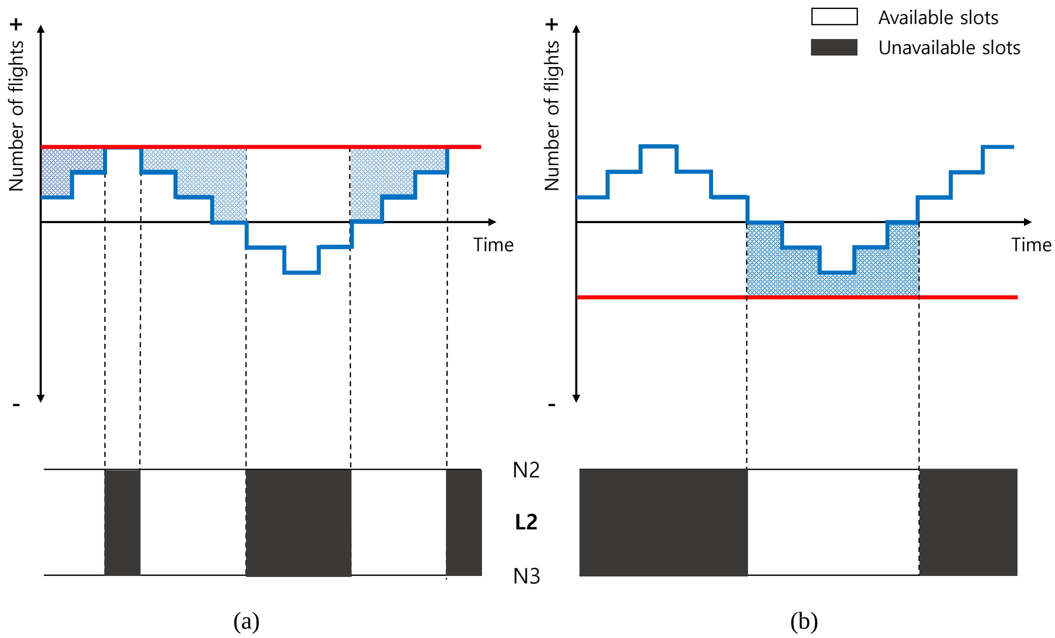


Figure 5. Slot availability depends on the direction: (a) positive(+) (b) negative(-) .

2. Constraints of crossing taxiway node

Another characteristics of surface movement is the role of nodes. In the original FCFS algorithm, only the beginning and end nodes have the role of enforcing rate constraints as airport capacity constraints. Nodes of airport surface represent actual taxiway junctions. As shown in Fig. 6, the transit through a junction is not instant and has a finite transit time.

Initially, an attempt to model this junction as a short link with a capacity of one was made. It is possible to eliminate conflicts at junctions by using this idea. However, if more than two links are connected to a node, a separate link is required for every entry and exit links pair. For example, if four links are connected to a node, there are six entry and exit pair, which will require six links for one junction.

To solve this problem, modeling junctions using a node is investigated. If the dimension of the junction is l and the taxing speed is V, the node is unavailable for a time period l/V. So, if the availability slot is blocked l/V before and after from the time when the aircraft is expected to cross the center of the junction, then the conflict at this junction can be effectively eliminated. This is equivalent to imposing a maximum rate of $1/(l/V)$ constraint at the node as shown in Fig. 7.

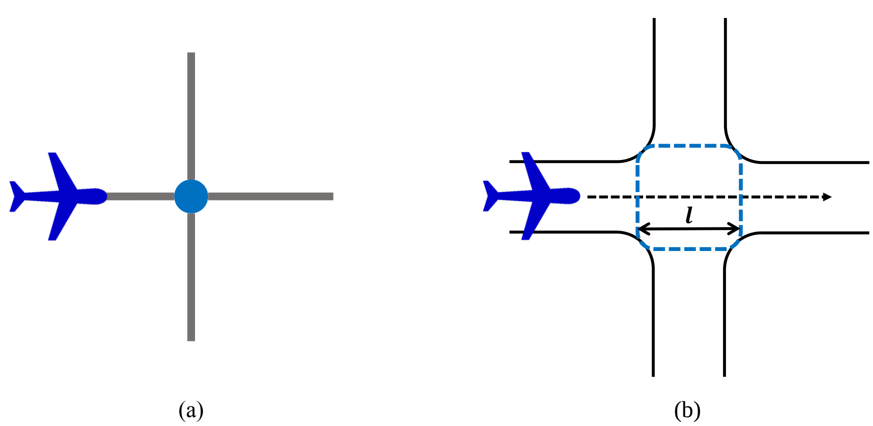


Figure 6. Taxiway junctions: (a) node-link model (b) actual taxiway .

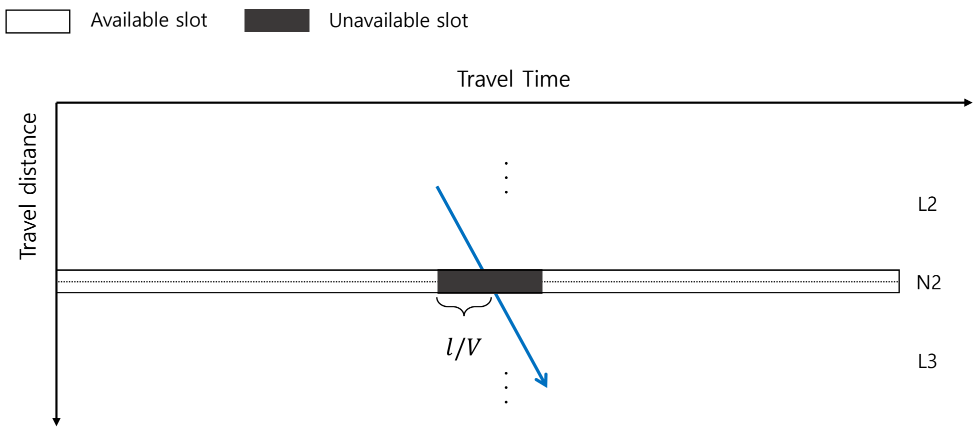


Figure 7. Updating available slots in crossing taxiway node N2.

3. *Verification using a simple airport model*

To test the EFCFS scheduler on an airport surface, a simple airport model with one runway and two gates is created as shown in Fig. 8. Departure route is shown in Fig. 8(a) and arrival route is shown in Fig. 8(b). To verify the scheduler, rate constraints at gates and runway are set unrealistically high at a turnaround time of ten seconds. Capacity of all the links is fixed to five aircraft. Junction crossing time at all the junction nodes is fixed to ten seconds, which is equivalent to imposing a rate constraint of 360 aircraft per hour. Table 2 shows the original schedule.

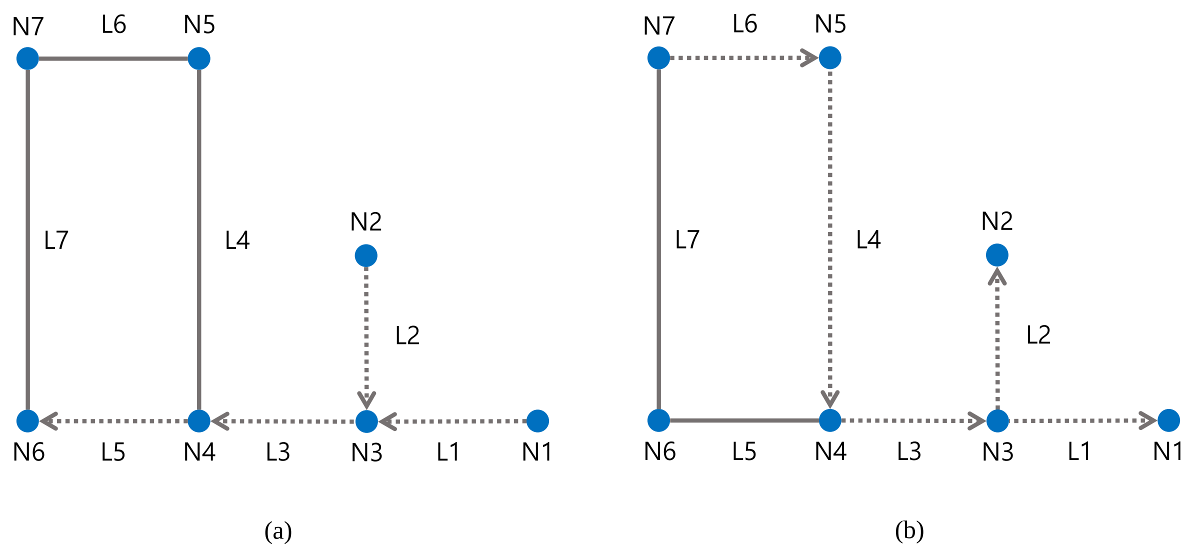


Figure 8. Simple airport surface node-link model. (a) Departure route. (b) Arrival route.

Table 3 shows the scheduling results with link capacity constraints only. As can be expected, the delay is zero for the first four or five flights because all the links are initially empty, but the delay starts to build up thereafter. Figure 9 shows the number of aircraft in links L3 and L4 as functions of time. As can be seen from Fig. 8, link L3 has to accommodate traffic in both directions and it can be observed in Fig. 9(a) with alternating positive and negative counts. Link L4 is used only for the arrival route and shows only negative counts, which indicates an operation in only one direction.

Table 4 shows the scheduling results with both the link constraints and node rate constraints. Due to unrealistic departure schedule that is spaced ten seconds apart, it can be seen that the node constraints significantly increase the delay.

Table 2. Nominal schedule for the simple airport model.

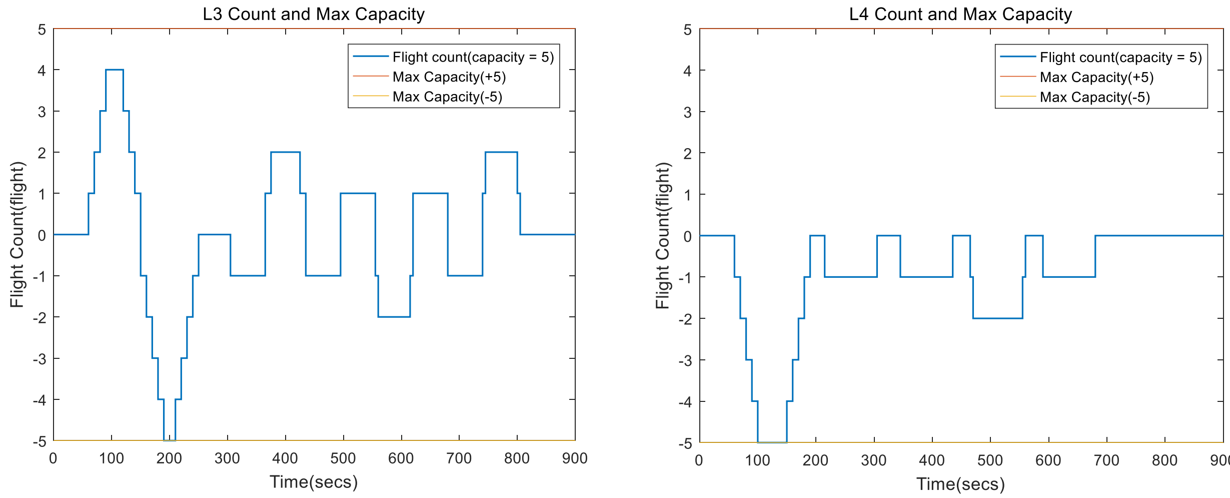
Flight	Path	Time	D/A	Flight	Path	Time	D/A
OZ8900	N1 → N6	0:00:00	Dep.	KE1200	N7 → N2	0:00:00	Arr.
LJ302	N2 → N6	0:00:10	Dep.	OZ8929	N7 → N1	0:00:10	Arr.
KE879	N1 → N6	0:00:20	Dep.	TW752	N7 → N2	0:00:20	Arr.
ZE706	N2 → N6	0:00:30	Dep.	OZ8902	N7 → N1	0:00:30	Arr.
ZE204	N1 → N6	0:00:40	Dep.	TW902	N7 → N2	0:00:40	Arr.
OZ8002	N2 → N6	0:00:50	Dep.	BX8100	N7 → N1	0:00:50	Arr.
LJ304	N1 → N6	0:01:00	Dep.	TW810	N7 → N2	0:01:00	Arr.
KE1902	N2 → N6	0:01:10	Dep.	OZ8906	N7 → N1	0:01:10	Arr.
LJ562	N1 → N6	0:01:20	Dep.	ZE206	N7 → N2	0:01:20	Arr.
BX8136	N2 → N6	0:01:30	Dep.	LJ306	N7 → N1	0:01:30	Arr.

Table 3. Scheduling results without junction constraints.

Flight	Time	Delay (sec)	D/A	Flight	Time	Delay (sec)	D/A
OZ8900	0:00:00	0	Dep.	KE1200	0:00:00	0	Arr.
LJ302	0:00:10	0	Dep.	OZ8929	0:00:10	0	Arr.
KE879	0:00:20	0	Dep.	TW752	0:00:20	0	Arr.
ZE706	0:00:30	0	Dep.	OZ8902	0:00:30	0	Arr.
ZE204	0:05:05	265	Dep.	TW902	0:00:40	0	Arr.
OZ8002	0:05:15	265	Dep.	BX8100	0:02:35	105	Arr.
LJ304	0:07:15	375	Dep.	TW810	0:04:45	225	Arr.
KE1902	0:09:20	490	Dep.	OZ8906	0:06:45	335	Arr.
LJ562	0:11:20	600	Dep.	ZE206	0:06:50	330	Arr.
BX8136	0:11:25	595	Dep.	LJ306	0:08:50	440	Arr.

Table 4. Scheduling results with junction constraints.

Flight	Time	Delay(sec)	D/A	Flight	Time	Delay(sec)	D/A
OZ8900	0:02:40	160	Dep.	KE1200	0:00:00	0	Arr.
LJ302	0:05:00	290	Dep.	OZ8928	0:02:20	130	Arr.
KE879	0:07:20	420	Dep.	TW752	0:04:40	260	Arr.
ZE706	0:09:40	550	Dep.	OZ8902	0:07:00	390	Arr.
ZE204	0:12:00	680	Dep.	TW902	0:09:20	520	Arr.
OZ8002	0:14:20	810	Dep.	BX8100	0:11:40	650	Arr.
LJ304	0:16:40	940	Dep.	TW810	0:14:00	780	Arr.
KE1902	0:19:00	1070	Dep.	OZ8906	0:16:20	910	Arr.
LJ562	0:21:20	1200	Dep.	ZE206	0:18:40	1040	Arr.
BX8136	0:23:40	1330	Dep.	LJ306	0:21:00	1170	Arr.



(a) (b)

Figure 9. Aircraft counts and capacity constraints in links L3 and L4.

B. Route Assignment

One of the fundamental objectives of departure and arrival management is to use efficient taxi routes. In this research, to utilize the low computational cost advantage of the EFCFS scheduler, multiple route options are investigated. Figure 10 summarizes the route generation process. Minimum distance route is first calculated from the given gate and runway pair using a Dijkstra algorithm denoted by Route 0.[15] Alternate routes are searched by removing a link from the original minimum distance route and recalculating the minimum distance route as shown in Routes 2 and 3. Theoretically, there can be as many alternate route as the number of links that consists of the original minimum distance route. However, there are cases that routes cannot be found without a certain link as shown in Route 1. Using this method, upto five candidate routes including the original minimum distance route are selected.

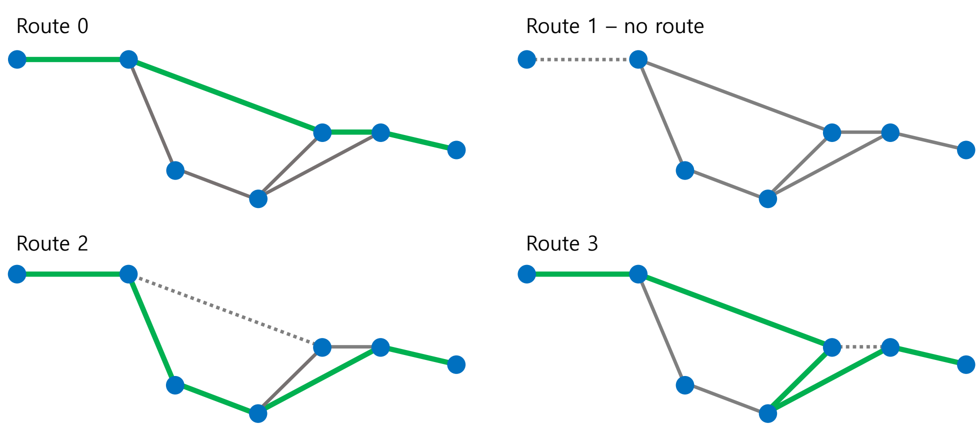


Figure 10. Route generation process.

Figure 11 shows three route options that connect a taxiway node and a runway with the actual node-link data of Incheon International Airport shown in Fig. 3. Transit times through each segment in each route can be computed, and schedules are computed using the transit time information. Route and schedule that result in minimum delay for the given flight are selected, which is similar to the method used for weather rerouting in Lee et al.[12]

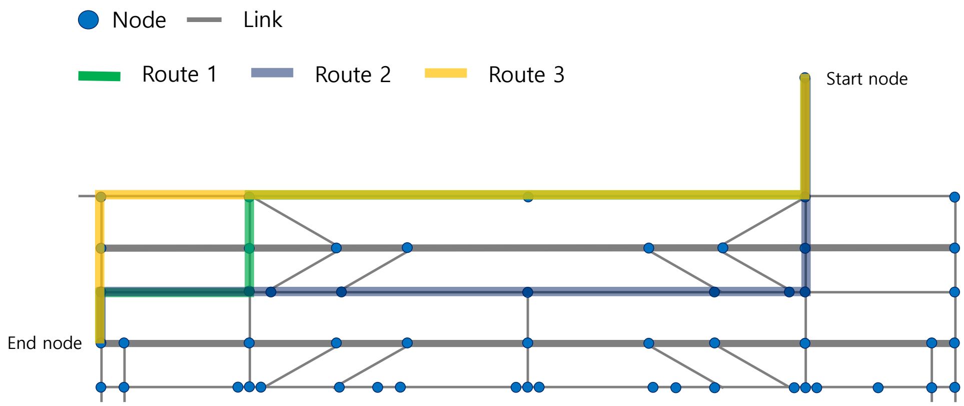


Figure 11. Route assignement example showing three possible routes.

III. Incheon International Airport Scheduling Results

A. Problem Setup

The initial scheduling input is generated using the historic data on April 1st, 2015. Flight Operation Information System (FOIS) data provide each aircraft's gate, runway, scheduled departure and arrival times, and actual departure and arrival times for 784 flights for the given date.

The node-link model of Incheon International Airport shown in Fig. 3 consists of 278 nodes and 377 links.[14] Actual gates are used in the passenger terminals located in between runways 15R/33L and 16/34. Gates in the two cargo terminals are aggregated into two nodes. Ramp area is not modeled in detail for simplicity. Gates are grouped in blocks shown in green dotted polygons and all the gates in the same block are connect to a near by taxiway node denoted by a light blue line with an arrows with straight lines.

Transit times are computed by dividing actual length of the link with the nominal taxi speed of 15 knots.[16] To account for possible congestion and non-straight taxilanes in the ramp area, the transit times between gates and assigned taxiway nodes are computed by dividing the straight line distance by five knots.

Arrival aircraft enters the runway at the runway threshold node and exits the runway at the first node that it encounters after covering the assigned landing field length. Landing field lengths are categorized into three classes based on the wake turbulence separation classes as shown in Table 5.[17] The speed of landing aircraft is assumed to decrease linearly from 150 knots at the runway threshold node to 15 knots at the runway exit node, and the transit times along the runway links are computed accordingly. Departure aircraft are assumed to leave the system once they arrive at the runway threshold nodes.

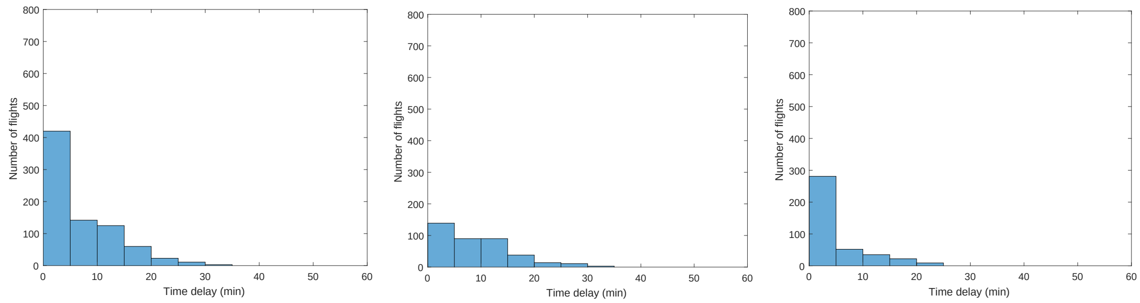
Table 5. Landing field lengths.

Aircraft class	Runway Landing Distance (m)
Heavy	2200
Large	1850
Small	1600

Maximum AAR and ADR of each runway are assigned based on the actual departure and arrival times from the FOIS data with the minimum set to 15 aircraft per hours. These capacity constraints are applied at the six runway threshold nodes. For taxiway junction nodes, passing rate of 926 aircraft per hour is calculated assuming taxiway width of 30 m and taxing speed of 15 knots. Gate nodes are assume to unconstrained. Capacity of the taxiway links is assigned by dividing the length of the taxiway by the sum of aircraft length and the safety distance. For this study, a constant length of 150 m is used for this sum regardless of the actual length of each aircraft. Links that represent the ramp area and runways are considered to be unconstrained.

B. Scheduling Results

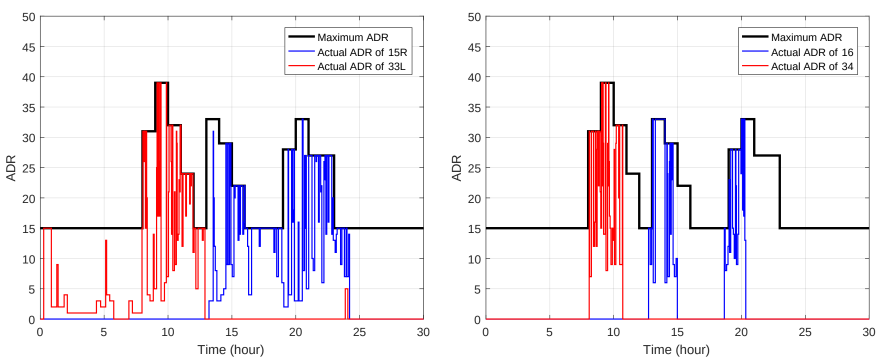
Initial scheduling was performed by assigning priority based solely on the scheduled departure or arrival times. Delays were measured from the original scheduled departure or arrival times to the computed departure or arrival times after the scheduling was performed. For this case, the average delay was 6.3 minutes for all 784 aircraft. Average departure delay was 4.2 minutes for 399 aircraft, and average arrival delay was 8.4 minutes for 385 aircraft. Figure 12 shows the delay distributions. As can be seen from Fig. 12(b), maximum delay is between 30 and 35 minutes for arrival.



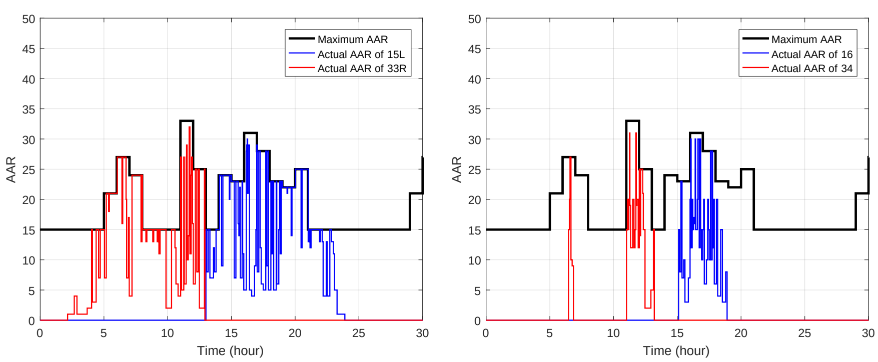
(a) Total. (b) Arrival only. (c) Departure only.

Figure 12. Delay distribution when the priority is based only on the scheduled departure or arrival times.

Figure 13 shows the ADR and AAR for each runway. Runways 15R/33L were used only for departures, and 15L/33R were used only for arrivals. Mixed operations were performed on runway 16/34. It can be noted that the flow direction was changed around noon. All the runway capacity constraints were satisfied. As the capacity constraints are based on actual operation, it shows high runway utilization during the day.



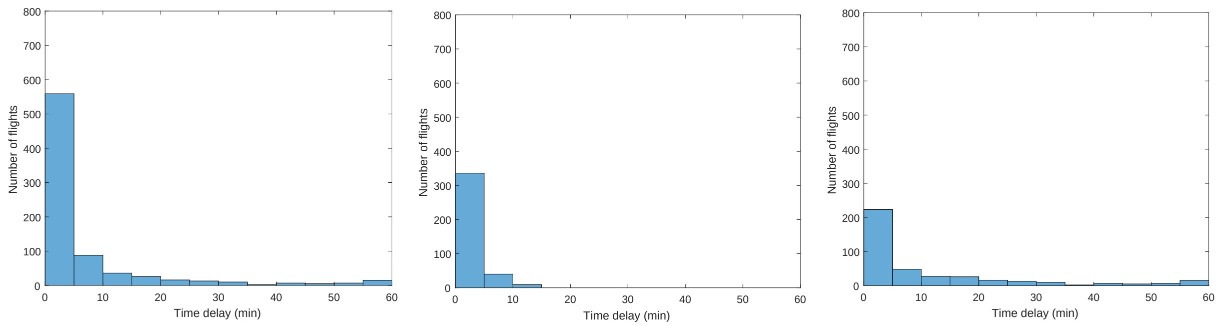
(a) 15R/33L departure. (b) 16/34 departure.



(c) 15L/33R arrival. (d) 16/34 arrival.

Figure 13. Runway utilization when the priority is based only on the scheduled departure or arrival times.

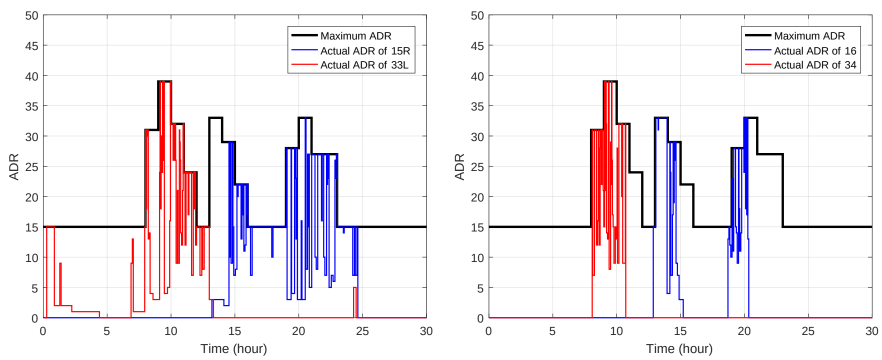
Priorities were given to the arrival flights for the second scheduling experiment to reduce arrival delay. For this case, the average delay was increased from 6.3 minutes to 7.6 minutes for all 784 aircraft. Average departure delay was significantly increased from 4.2 minutes to 13.4 minutes for 399 aircraft while the average arrival delay was significantly reduced from 8.4 minutes to 1.7 minutes for 385 aircraft. Figure 14 shows the delay distributions. Maximum arrival delay was in between 10 and 15 minutes as shown in Fig. 14(b). However, there were a fair number of flights that were delayed over one hour for departure as shown in Fig. 14(c). It suggests that giving priorities to arriving flights can increase the overall delay. It is very likely that priorities should be carefully given to arrival flights with remaining fuel problems and departure flights with TFM initiatives.



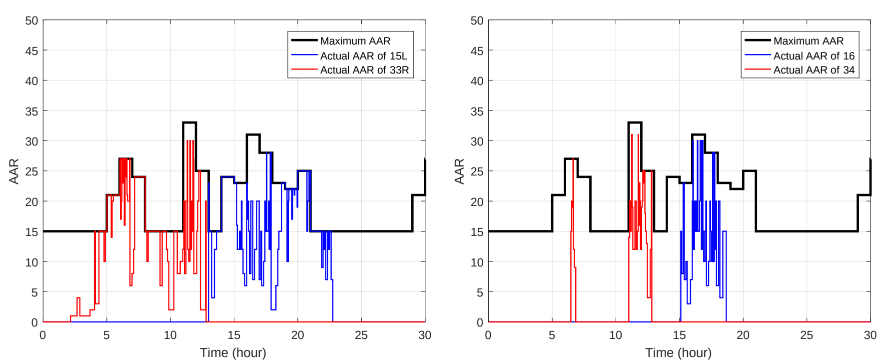
(a) Total. (b) Arrival only. (c) Departure only.

Figure 14. Delay distribution when priorities were given to arrival flights.

Figure 15 shows the ADR and AAR for each runway. All the runway capacity constraints were satisfied.



(a) 15R/33L departure. (b) 16/34 departure.



(c) 15L/33R arrival. (d) 16/34 arrival.

Figure 15. Runway utilization when priorities were given to arrival flights.

Final scheduling experiment was designed to investigate the impact of taxiway junction constraints. From the initial case of no priority, the rate constraints at taxiway junction nodes were reduced to one quarter. Average delays and delay distributions were almost identical to the original case. This result suggests that it might not be necessary to include these constraints at the scheduling phase. It seems that most of the conflicts can be handled tactically without adding any additional delays.

IV. Conclusions

FCFS scheduler originally developed for enroute traffic flow management was enhanced with two extra capabilities to be applied to airport surface movement scheduling. Link directionality problem was solved by introducing negative aircraft count concept and conflicts at junctions were handled using rate constraints at junction nodes. In addition, route assignment capability that evaluates upto five different taxi route was added so that the scheduler can be used with the dynamic taxi routing concept. The EFCFS scheduler was verified with a simple airport model. Several scheduling experiments were performed using a historic surface movement data at Incheon International Airport in Republic of Korea. The results showed giving priorities to arrival flights increased overall system delay. It was also discovered that it might not be necessary to consider junction conflict during the scheduling phase. Further investigations are planned to compared the scheduling results with other scheduling scheme and to study prioritization strategies.

Acknowledgments

This research was funded by the Ministry of Land, Infrastructure, and Transportation (MOLIT) under contract 16ATRP-C088155-03. The authors would like to thank Dr. Daekeun Jeon and his staff at Korea Aerospace Research Institute for providing systematic project management and essential data for the scheduling experiment.

References

[1]Jung, M., Lee, J., Eun, Y. and Jeon, D., *Node-Link Model Generation of Incheon International Airport*, The Korean Society for Aeronautical and Space Sciences(KSAS) Conference, Jeju, Republic of Korea, Nov. 2015.

[2]Montoya, J., Wood, Z. and Rathinam, S., *Runway Scheduling Using Generalized Dynamic Programming*, AIAA Guidance, Navigation, and Control Conference, Portland, OR, USA, 2011.

[3]Keith, G., Richards, A. and Sharma, S., *Optimization of Taxiway Routing and Runway Scheduling*, AIAA Guidance, Navigation, and Control Conference, Honolulu, HI, USA, Aug. 2008.

[4]Clare, G., Richards, A. and Sharma, S., *Receding Horizon, Iterative Optimization of Taxiway Routing and Runway Scheduling*, AIAA Guidance, Navigation, and Control Conference, Chicago, IL, USA, Aug. 2009.

[5]Gupta, G., Malik, W. and Jung, Y., *A Mixed Integer Linear Program for Airport Departure Scheduling*, AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Hilton Head, SC, USA, Sep. 2009.

[6]Montoya, J., Wood, Z., Rathinam, S. and Malik, W., *A Mixed Integer Linear Program for Solving a Multiple Route Taxi Scheduling Problem*, AIAA Guidance, Navigation, and Control Conference, Toronto, Ontario, Canada, Aug. 2010.

[7]Cook, L., Atkins, S. and Jung, Y., *Improved Prediction of Gate Departure Times Using Pre-Departure Events*, AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Anchorage, AK, USA, 14-19 Sep. 2008.

[8]Lee, H., Malik, W., Zhang, B. and Nagarajan, B., *Taxi Time Prediction at Charlotte Airport Using Fast-Time Simulation and Machine Learning Techniques*, AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, Dallas, TX, USA, June 2015.

[9]Hayashi, M., Hoang, T, Jung, Y., Malik, W., Lee, H. and Dulchinos, V. L., *Evaluation of Pushback Decision-Support Tool Concept for Charlotte Douglas International Airport Ramp Operations*, 10th USA/Europe ATM R&D Seminar (ATM2015), Lisbon, Portugal, 23-26 June 2015.

[10]Zhu, Z., Okuniek, N., Gerdes, I., Schier, S., Lee, H. and Jung, Y., *Performance Evaluation of the Approaches and Algorithms Using Hamburg Airport Operations*, 35th Digital Avionics Systems Conference (DASC), Sacramento, CA, USA, 25-29 Sep. 2016.

[11]Park, B. and Lee, H., *Development of a First-Come First-Served Departure Scheduler*, 7th Asia-Pacific International Symposium on Aerospace technology (APISAT), Cairns, Australia, Nov. 2015.

[12]Lee, H., Chatterji, G. and Palopo, K., *Interaction of Airspace Partitions and Traffic Flow Management Delay with Weather*, 30th Digital Avionics Systems Confernece (DASC), Seattle, WA, USA, 16-20 Oct 2011.

[13]Park, C., Lee, H. and Meyn, L., *Computing Flight Departure Times Using an Advanced First-Come First-Served Scheduler*, AIAA Aviation Technology, Integration, and Operation (ATIO) Conference, Indianapolis, Ind, USA, Sep. 2012.

[14]Ministry of Land, Infrastructure and Transport Office of Civil Aviation, *AIRAC Aeronautical Information Publication (AIP) AMENDMENT NR 12/16*, Republic of Korea, 27 Oct. 2016.

[15]Dijkstra, E., *A Note on Two Problems in Connexion with Graphs*, Numerische Mathematik 1 (1959), pp.269-271, 1959.

[16]EUROCONTROL, *User Manual for the Base of Aircraft Data (BADA) Revision 3.9*, No. 11/03/08-08, April 2011.

[17]FAA, *Air Traffic Control*, FAA Order JO 7110.65W, Federal Aviation Administration, 10 Dec. 2015.