

저자 (Authors)	강선영, 이학태
출처 (Source)	한국항공우주학회 학술발표회 초록집 , 2015.11, 315-318(4 pages)
발행처 (Publisher)	한국항공우주학회 The Korean Society for Aeronautical & Space Sciences
URL	http://www.dbpia.co.kr/journal/articleDetail?nodeId=NODE06768684
APA Style	강선영, 이학태 (2015). 항적 자료를 기반으로 항공기의 공역 체류시간을 산출하는 데이터 처리 프로그램. 한국항공우주학회 학술 발표회 초록집, 315-318
이용정보 (Accessed)	인하대학교 165.246.75.*** 2020/05/21 13:55 (KST)

저작권 안내

DBpia에서 제공되는 모든 저작물의 저작권은 원저작자에게 있으며, 누리미디어는 각 저작물의 내용을 보증하거나 책임을 지지 않습니다. 그리고 DBpia에서 제공되는 저작물은 DBpia와 구독계약을 체결한 기관소속 이용자 혹은 해당 저작물의 개별 구매자가 비영리적으로만 이용할 수 있습니다. 그러므로 이에 위반하여 DBpia에서 제공되는 저작물을 복제, 전송 등의 방법으로 무단 이용하는 경우 관련 법령에 따라 민, 형사상의 책임을 질 수 있습니다.

Copyright Information

Copyright of all literary works provided by DBpia belongs to the copyright holder(s) and Nurimedia does not guarantee contents of the literary work or assume responsibility for the same. In addition, the literary works provided by DBpia may only be used by the users affiliated to the institutions which executed a subscription agreement with DBpia or the individual purchasers of the literary work(s) for non-commercial purposes. Therefore, any person who illegally uses the literary works provided by DBpia by means of reproduction or transmission shall assume civil and criminal responsibility according to applicable laws and regulations.

항적 자료를 기반으로 항공기의 공역 체류시간을 산출하는 데이터 처리 프로그램

Data Processing Program to Determine Transit Time
for Sectors based on Trajectory Data

강선영^{1*}, 이학태¹
인하대학교¹

초 록

현재 인하대학교에서는 NASA에서 연구한 선입 선처리 (First-Come First-Served, FCFS) 스케줄링 기법을 기초로, 이를 우리나라 공역에 적용하기 위해 연구를 진행하고 있다. 해당 스케줄러는 특정 입력 데이터가 필요하기 때문에, 우리나라 항적 자료로부터 이를 도출해줄 기법이 요구되었다. 본 논문에서는 객체 지향 프로그래밍 도구인 Java를 이용하여, 각 공역에서 항공기의 체류시간을 산출해내는 기법을 제시한다. 본 논문에서 사용한 항적 자료의 양은 대략 250만 개 정도로, 이러한 대용량의 자료를 처리하기 위해 데이터베이스를 사용하였다.

ABSTRACT

Currently, a study is in progress in Inha university to apply the First-Come First-Served (FCFS) scheduling technique studied by NASA to Korean National Airspace. Since a specific type of input data are needed for this scheduler, a technique is required to determine the flights schedules from trajectory data. In this paper, a methodology that calculates entry, exit, and transit times for each sector using an objected-oriented programming tool, java is presented. A database tool, MySQL, is used to efficiently handle large trajectory data, which contain around 2.5 million data points.

Key Words : Data Processing(데이터 처리), FCFS Scheduler(FCFS 스케줄러), Trajectory Data(항적 자료), Air Traffic Management(항공운항관리)

1. 서론

항공운송시장의 규모는 세계화가 진행됨에 따라 오늘날과 같이 크게 성장하였다. 이에 따라 항공 교통량도 함께 증가하면서 항공기의 출도착 지연이 빈번하게 발생하고 있다. 이러한 문제를 해결하기 위해 국제민간항공기구 (International Civil Aviation Organization, ICAO)에서는 새로운 항행 시스템 구축을 위해 Aviation System Block Upgrade (ASBU)를 추진하고 있으며⁽¹⁾, 미국 연방 항공청 (Federal Aviation Administration, FAA)에서는 NextGen 프로그램을 진행하고 있다⁽²⁾. 우리나라도 이러한 대열에 합류하여 ASBU의 국내 적용을 위해 노력하고 있다.⁽³⁾

NASA Ames Research Center에서는 위와 같은 항공기의 출도착 지연을 줄이기 위해, 선입 선처

리 (First-Come First-Served, FCFS) 알고리즘을 기초로 한 스케줄링 기법에 대해 연구를 진행해왔다.⁽⁴⁾ 이 선입 선처리 스케줄링에서 필요로 하는 입력 데이터는 Table 1과 같다. 인하대학교에서도 NASA에서 진행한 연구를 기초로 항공기의 출도

Table 1. Input data for FCFS scheduling

	Data	Description
1	Callsign	항공기의 ID
2	Departure Airport	출발 공항
3	Arrival Airport	도착 공항
4	Entry Time	현재 섹터로 들어온 시간
5	Exit Time	현재 섹터로부터 나간 시간
6	Transit Time	항공기가 현재 섹터에 머무른 시간 (Exit - Entry)
7	Upstream 섹터	현재 섹터로 들어오기 이전의 섹터
8	Current 섹터	현재 항공기가 머물고 있는 섹터
9	Downstream 섹터	현재 섹터 이후에 통과할 섹터

작 스케줄링 기법을 연구 및 개발 중에 있다. 본 논문에서는 이러한 스케줄링 연구를 위해 Table 1과 같은 형태의 데이터를 도출하는 프로그램을 Java를 이용하여 제작하였다.

2. 본론

본 논문에서 사용한 데이터는 항공기의 항적 자료와 우리나라 비행 정보 구역 (Flight Information Region, FIR) 내의 섹터 및 Terminal Maneuvering Area (TMA)의 좌표이다. 항적 자료는 2015년 4월 1일부터 4월 2일까지의 데이터로, 인천공항에서 출도착하는 항공기들만을 고려한 것이다. 이 데이터는 인천공항으로부터 제공받았으며, 모든 데이터는 인천 비행 정보 구역 주변의 데이터만을 포함하고 있다. 구역 좌표 (섹터 및 TMA; 앞으로는 섹터로 통일)는 우리나라의 AIP 문서를 참고하였다. 이를 동안 수집된 항적 자료는 약 250만 개 정도로, 대용량의 데이터를 처리하기 위한 데이터베이스 프로그램으로 Oracle사의 MySQL을 활용하였다.

2.1 요구사항 분석

Fig. 1은 항적 자료의 일부를 도시한 그림이다. Fig. 1과 같이, 항적 자료는 시간에 따라 정렬되어있고, 항공기의 콜사인, 위도, 경도, 고도, 대지 속도 등의 상태 정보를 포함하고 있다. 하지만, 일부 데이터의 경우 콜사인이 존재하지 않는 등의 오류가 있어 이를 제외시켜줄 수 있는 필터가 요구된다.

Fig. 1을 Table 1의 스케줄링 연구에 필요한 입력 데이터와 비교했을 때, 그 형태가 매우 다르다. 항적 자료는 섹터와 관련된 내용을 전혀 가지고 있지 않기 때문에, 항적 자료를 토대로 각

TimeSec	Index	Hour	Min	Sec	Latitude	Longitude	Alt_H	F_L	GroundSpd	ThrustEng	WindDir	CallSign	TimeOfSec	TrackNo	S_P	S_P	SpeedKmph	SpeedMph
2015-03-21-02:38:308	59496	-	-	-	33.22949	123.70706	340	421	227.1	3700	648.018	---	618	391.9205	60.99705	-181.3294	114.9172	
2015-03-21-02:38:318	59497	-	-	-	37.212294	126.40868	12	148.9	251.9	3265	604.828	---	17	-52.46075	80.5	-45.95346	43.41432	
2015-03-21-02:38:328	59498	-	-	-	37.322201	126.50949	104	286.1	295.5	7270	310.301	---	42	-46.46075	87.025	-45.44923	-115.52406	
2015-03-21-02:38:341	59493	-	-	-	37.440412	121.50787	0	14.6	264.8	6403	---	---	67	385.125	97.84075	-7.489836	4.677812	
2015-03-21-02:38:351	59491	-	-	-	37.092176	126.62828	29	193.6	128.3	3792	AR620	---	57	-43.122	71.21875	-55.27472	42.89628	
2015-03-21-02:38:378	59490	-	-	-	33.22608	123.00208	290	525.1	76.4	6167	PA212	---	1382	21.375	-213.125	262.81764	42.50974	
2015-03-21-02:38:403	59489	-	-	-	37.22962	126.77668	31	229	292.2	3613	PA206	---	1371	-46.0875	88.8075	-111.94924	40.89799	
2015-03-21-02:38:518	59488	-	-	-	33.47918	126.57242	130	404.9	11.2	3853	KA210	---	343	-51.3875	87.1225	54.60384	32.82664	
2015-03-21-02:38:544	59484	-	-	-	34.12676	126.21428	900	8.4	15.4	7777	---	---	387	31.9075	-104.1	2.48728	1.50384	
2015-03-21-02:38:564	59485	-	-	-	34.36441	126.65478	300	33.7	343.3	7776	---	---	1377	53.375	-90.4075	-4.24978	46.62494	
2015-03-21-02:37:705	59482	-	-	-	37.463205	126.56426	27	162.7	136.9	301	---	---	1120	-46.875	77.5	-32.99372	76.97828	
2015-03-21-02:37:684	59481	-	-	-	36.02897	126.40181	0	8	298	5230	---	---	75	108.125	16.46875	-5.448956	1.922276	
2015-03-21-02:37:289	59478	-	-	-	35.69319	123.32861	360	408.8	277.1	3665	AR203	---	386	134.46875	9	-205.21152	15.89104	
2015-03-21-02:37:288	59479	-	-	-	37.25764	126.70302	110	227.6	144.5	3783	TP9410	---	498	-38.875	83.25	43.5346	42.56824	
2015-03-21-02:37:260	59476	-	-	-	36.36406	123.6463	110	512.4	81.1	2406	---	---	1467	224.8825	246.5125	246.50472	40.51912	
2015-03-21-02:36:306	59473	-	-	-	37.42719	126.60778	53	260.7	243.3	5164	AR682	---	387	-43.8875	83.025	-119.83844	40.24864	
2015-03-21-02:36:306	59472	-	-	-	36.26918	126.18148	0	11.9	171.7	5382	---	---	1004	126.925	27.126	4.6782	4.24192	
2015-03-21-02:36:141	59470	-	-	-	35.998152	126.64325	280	436.5	81.1	5151	TH8751	---	578	-45.84075	5.24875	-4.24978	124.40232	
2015-03-21-02:36:001	59468	-	-	-	31.076205	123.46797	247	514.4	71.3	5013	---	---	12	-213.25	-384.5	246.24776	84.12292	
2015-03-21-02:36:001	59468	-	-	-	31.46770	124.25182	400	416.9	257.1	5022	---	---	253	-173.25	-386.5	246.00569	40.4388	
2015-03-21-02:35:863	59467	-	-	-	34.62243	126.57852	92	311.5	69.3	6503	AR.712	---	1274	46.125	-74.78125	146.88736	56.74928	

Fig. 1. Trajectory data

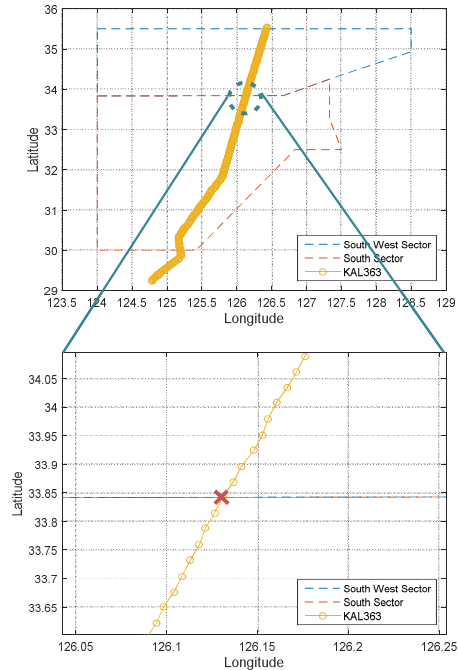


Fig 2 Trajectory of KAL363

지점에서 항공기가 어떤 섹터에 존재하는지, 그리고 어느 시점에 섹터를 통과하였는지에 대한 계산이 필요하다.

Fig. 2는 항적 자료 중, KAL363 항공기의 데이터를 도시한 것이다. Fig. 2의 상단 그래프의 굵은 실선은 항공기의 항적을, 점선은 섹터를 나타내고, 하단의 그래프는 항공기가 섹터를 통과하는 부분을 확대한 것이다. 하단의 그림 중, 빨간색으로 표시된 위치 즉, 항공기가 섹터를 통과할 때의 시간을 산출하여 스케줄링에 필요한 입력 데이터를 도출하는 것이 필요하다.

2.2 데이터 처리 프로그램

Fig. 3은 입력 데이터를 산출하는 과정을 나타낸다. 프로그램에서 가장 먼저 이루어지는 것은 데이터베이스로부터 데이터를 읽어오는 것이며, 이 때

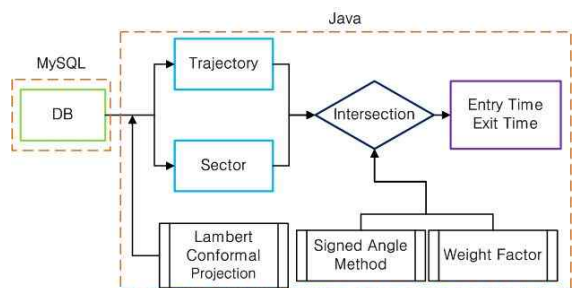


Fig. 3. Data Processing Procedure

오류데이터를 제외하고 읽어올 수 있도록 한다. 읽어온 항적 자료는 콜사인에 따라 Trajectory라는 형태로, 공역 자료는 각 공역의 ID에 따라 Sector라는 형태로 저장되고, 이 데이터는 Intersection이라는 처리과정을 통해 스케줄링의 입력 데이터로 산출된다.

2.2.1 Lambert Conformal Conic Projection

본 논문에서 항공기의 궤적은 점들을 잇는 선분으로, 섹터는 하나의 다각형으로 고려하였다. 이러한 기하학적 표현함을 통해 궤적과 섹터사이의 관계 분석이 용이해진다. 따라서 항공기 궤적 선분과 섹터 다각형을 이루는 선분 사이의 교점을 찾음으로써, 항공기가 섹터를 통과하는 시간을 산출할 수 있다. 하지만, 이러한 기하학적 계산은 2차원 평면상에서 수행하는 것이 훨씬 수월하다. 따라서 위경도로 기록되어 있는 항적 및 공역 자료를 2차원 평면으로 투영시키기 위해 람베르트 정각원추도법 (Lambert Conformal Conic Projection) 을 사용하였다. 원추도법이란 삿갓 모양의 원추를 지구 표면에 씌워 그 원추면에 지구의 경도, 위도선을 투영하여 전개한 도법이다.⁽⁵⁾

2.2.2 Signed Angle Method⁽⁶⁾

항공기의 궤적과 섹터사이의 교점을 찾기 전에 항공기가 각 위치에서 어떤 섹터 내부에 있는지를 먼저 판별해야한다. 이를 위해 Signed Angle Method를 적용하여 현재 위치에서의 섹터를 찾아내었다. 이 판별법은 Fig. 4와 같이 임의의 점 P를 기준으로 다각형의 꼭짓점(P1~P5) 한 쌍이 이루는 사잇각들의 총합이 360°가 될 경우, 점 P가 다각형 내부에 있다는 것을 의미하고, 총합이 0°일 경우 이는 다각형 외부(P')에 있다는 것을 의미한다. 이 때, P'-P4, P'-P5와 같이 사잇각이 180° 보다 큰 경우에는 사잇각에서 360°를 빼줘야 한다.

2.2.3 Weight factor

마지막으로 항공기의 궤적과 섹터 사이의 교점을 찾기 위한 방법이다. 본 논문에서는 교점을

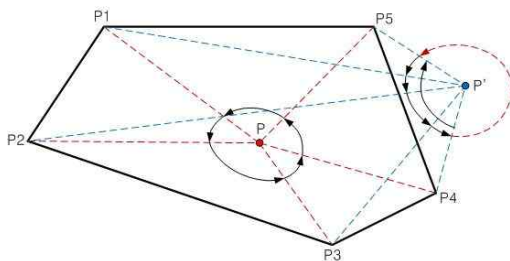


Fig. 4. Signed Angle Method

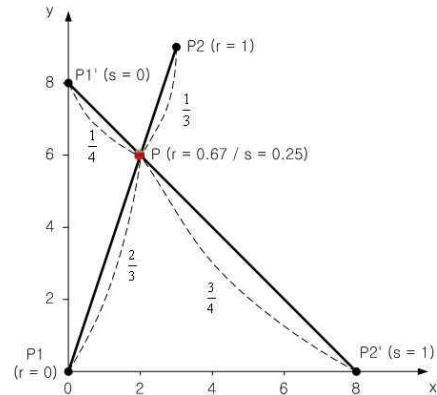


Fig. 5. Weight Factor

찾기 위해 weight factor를 도입하였다. 이는 아래의 식 (1)에서 r 과 s 을 의미하며, 0에서 1사이의 값을 가진다. weight factor에 대한 이해를 돕기 위해 Fig. 5를 첨부하였다.

$$P(x, y) = (1-r)P_1 + rP_2 \quad \dots (1)$$

$$= (1-s)P_1' + sP_2'$$

$$\begin{bmatrix} x \\ y \end{bmatrix} = (1-r) \begin{bmatrix} x_1 \\ y_1 \end{bmatrix} + r \begin{bmatrix} x_2 \\ y_2 \end{bmatrix} \quad \dots (2)$$

$$= (1-s) \begin{bmatrix} x_1' \\ y_1' \end{bmatrix} + s \begin{bmatrix} x_2' \\ y_2' \end{bmatrix}$$

$$\begin{bmatrix} x_2 - x_1 & x_1' - x_2' \\ y_2 - y_1 & y_1' - y_2' \end{bmatrix} \begin{bmatrix} r \\ s \end{bmatrix} = \begin{bmatrix} x_1' - x_1 \\ y_1' - y_1 \end{bmatrix} \quad \dots (3)$$

식 (1)을 좀 더 분별하기 쉽도록 식 (2)와 같이 x, y 의 행렬로 나타내었고, 식 (2)를 식 (3)과 같이 r 과 s 에 대한 행렬식($AX=d$)로 정리할 수 있다. 이를 Table 2와 같은 판별법을 사용하여 교점이 존재하는지를 판단하고, r 과 s 가 존재할 경우 이 weight factor를 시간에 적용하여 섹터 통과 시간을 산출할 수 있다.

Table 2의 판별법은 2차원 평면에서 완벽한 해답이 될 것이다. 하지만 실제 항공기는 2차원이 아닌 3차원에서 비행하기 때문에 위경도 외에 고도 정보가 존재한다. 따라서 Fig. 6과 같은 상황에서는 Table 2의 판별법을 통해 교점을 찾을 수 없는 문제가 있다.

이 경우에는, P1이 위치한 섹터2의 고도 정보

Table 2. Intersection Point Criterion Standard

Case	Description
$\det A \neq 0$	해(r, s)가 존재함; 한 개의 교점
$\det A = 0$ & $d = 0$	무수히 많은 해가 존재함; 두 선분이 일치 ($r = 0, s = 0$)
$\det A = 0$ & $d \neq 0$	해가 존재하지 않음; 두 선분이 평행 ($r, s = \text{null}$)

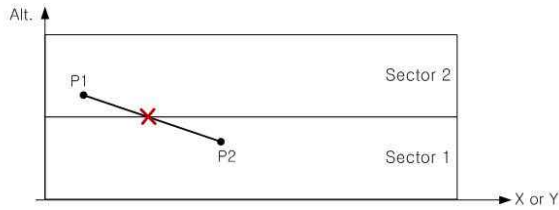


Fig. 6. Intersection caused by height change
 (Min & Max Alt.)와 P2의 고도를 비교한다. 만약 P2가 섹터2의 제한 고도 밖 ($P2 > \text{Max Alt.}$ & $P2 < \text{Min Alt.}$)에 존재한다면 이는 항공기가 섹터를 통과했다는 것이 되고, 이때의 고도 비율을 이용하여 통과 시간을 구할 수 있다.

2.3 데이터 처리 결과

프로그램을 실행하여 얻은 결과의 일부를 Fig. 7에 나타내었다. 결과로부터 AAR8981 항공기는 서울에서 출발하여 제주로 향하는 항공기임을 추정할 수 있다. Fig. 7의 사각형으로 표시한 부분을 살펴보면 이어지는 세 데이터의 Downstream, Current, Upstream 섹터가 대각선 방향으로 일정하게 이어지는 것을 확인할 수 있다.

또한, 섹터의 ID가 비어있는 부분이 있는데 이는 항공기가 어느 섹터에도 포함되어 있지 않은 경우이다. 본 논문에서는 인천 비행 정보 구역 중에서 섹터와 TMA에 대해서만 고려해두었기 때문에 Class G와 같은 그 외의 공역에서는 섹터 데이터가 비어있게 된다.

3. 결론

우리나라 항적 자료로부터 스케줄링 연구를 위해 필요한 입력 데이터를 도출하기 위해 Java를 이용하여 프로그램을 제작하였다. 이를 통해, 항공기가 비행하며 통과하는 공역 정보와 각 공역 통과시간을 산출할 수 있었다. 하지만, 본 항적 자료는 인천 FIR 주변에서의 정보만 가지고 있고, 출도착하는 공항에 대한 정보가 전무하기 때문에 스케줄링에 필요한 공항 정보와 다른 시간대의 두 항공기의 콜사인이 겹치면서 정상적인 결과 데이터를 도출하는데 문제가 있어 이를 해결할 수 있는 방안이 요구된다. 향후에는 이러한 문제를 해결할 수 있는 방안을 연구하여 개선된 입력 데이터를 얻기 위한 연구를 진행할 예정이다. 이를 통해, 항적 자료 처리에 반복적으로 사

1	Callsign	Entry Time	Exit Time	Transit Time	Upstream Sector	Current Sector	Downstream Sector
2035	AAR8981	110044282	110085502	41220	-----	T01_Seoul	T03_Osan
2036	AAR8981	110085502	110144835	59333	T01_Seoul	T03_Osan	T03_Seoul
2037	AAR8981	110144835	110230743	85908	T03_Osan	T03_Seoul	T04_Seoul
2038	AAR8981	110230743	110328262	97519	T03_Seoul	T04_Seoul	T06_Seoul
2039	AAR8981	110328262	110411870	83608	T04_Seoul	T06_Seoul	T07_Seoul
2040	AAR8981	110411870	110538069	126199	T06_Seoul	T07_Seoul	North West Sector
2041	AAR8981	110538069	111202721	664652	T07_Seoul	North West Sector	South West Sector
2042	AAR8981	111202721	111662937	460216	North West Sector	South West Sector	T29_Gwangju
2043	AAR8981	111662937	111851273	188336	South West Sector	T29_Gwangju	T23_Jeju
2044	AAR8981	111851273	112814201	962928	T29_Gwangju	T23_Jeju	-----
2045	KAL8216	26614258	27832853	1218595	-----	High Central Sector	Low Central Sector
2046	KAL8216	27832853	28128111	295258	High Central Sector	Low Central Sector	North West Sector
2047	KAL8216	28128111	28140549	12438	Low Central Sector	North West Sector	T07_Seoul
2048	KAL8216	28140549	28225149	84600	North West Sector	T07_Seoul	T06_Seoul
2049	KAL8216	28225149	28490720	265571	T07_Seoul	T06_Seoul	North West Sector
2050	KAL8216	28490720	28587268	95548	T06_Seoul	North West Sector	T04_Seoul
2051	KAL8216	28587268	28611297	24029	North West Sector	T04_Seoul	T11_Osan
2052	KAL8216	28611297	28827670	216373	T04_Seoul	T11_Osan	T01_Seoul
2053	KAL8216	28827670	28972278	144608	T11_Osan	T01_Seoul	-----

Fig. 7. Final data for scheduling input
 용될 알고리즘이 포함된 Java library를 개발할 수 있을 것으로 기대된다.

후 기

본 연구는 국토교통부 항공기술연구사업의 연구비 지원 (과제번호 15ATRP-C088159-02-000000) 에 의해 수행되었습니다.

참고문헌

- (1) International Civil Aviation Organization, 2013, *2013-2028 Global Air Navigation Plan*, Doc 9750-AN/963 Forth Edition, Montreal.
- (2) Federal Aviation Administration, 2013, *NextGen Implementation Plan*, Washington DC.
- (3) Kim, J. H. and Park, B. M., 2014, "ICAO ASBU and National Implementation Plans" *KOTI Aviation Brief*, No. 89, pp. 5~8.
- (4) Park, C., Lee, H.-T., and Meyn, L. A., 2012, "Computing Flight Departure Times Using an Advanced First-Come First-Served Scheduler," *Proceedings of the 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, AIAA, pp. 1~8.
- (5) Lambert Conformal Conic Projection, Retrieved Nov. 13, 2015 from <http://terms.naver.com/entry.nhn?docId=636291&cid=42463&categoryId=42463>
- (6) Faisal M. Chowdhury and Kamrul H. Talukder, 2005, "The Unidirectional Edge Method: A New Approach for Solving Point Enclosureproblem for Arbitrary Polygon," *Asian Journal of Information Technology*, Vol. 4, Issue. 5, pp 537~540.