



저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.



변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공학석사학위 논문

배속 시뮬레이션을 이용한 무인항공기 충돌 회피
알고리즘 연구

Separation Management of Unmanned Aircraft Using
Fast-Time Simulation



2019년 2월

인하대학교 대학원

항공우주공학과

류희경

공학석사학위 논문

배속 시뮬레이션을 이용한 무인항공기 충돌 회피
알고리즘 연구

Separation Management of Unmanned Aircraft Using
Fast-Time Simulation

2019년 2월

지도교수 이 학 태

이 논문을 석사학위 논문으로 제출함

이 논문을 류희경의 석사학위논문으로 인정함.

2019년 2 월



주심 최 기 영 (인)

부심 유 창 경 (인)

위원 이 학 태 (인)

초 록

최근 유인항공기 공역에서의 대형 무인항공기 통합 운용에 대한 요구가 증가하고 있는데, 이를 가능하게 하는 핵심 기술인 탐지 회피(Detect and Avoid)에 대한 연구가 활발하게 진행 중이다.

본 논문에서는 무인항공기의 주변의 위협이 되는 다른 항공기의 상태 정보가 주어진다 가정 하에, 현재 표준화가 진행 중인 Well Clear 기준에 따라 위험도를 평가하고, 이에 따라 필요한 회피 기동의 시작 시점, 회피 기동의 종류, 그리고 본 궤적으로 복귀하는 시점에 대한 연구를 수행하였다.

이를 위하여 기존의 배속 시뮬레이션 시스템을 개선하여 수직 수평 기동과 경로점 재설정을 가능하게 하였고, NASA에서 개발한 회피 기동 생성 엔진과 연동하여 다수의 후보 기동을 생성하도록 하였다. 최종적으로 비용함수를 이용하여 최적의 회피 기동과 본 궤적 복귀 시점의 선택이 어떻게 달라지는지 살펴보고 선택의 기준을 제시하였다.

본 연구에서 제시한 5 가지의 조우 시나리오의 경우 수평면에서의 회피 기동이 가장 효율적인 것으로 나타났다.

Keyword: DAA(Detect And Avoid), DAIDALUS(Detect And Avoid Alerting Logic for Unmanned System), DWC(DAA Well Clear), 충돌 감지 및 회피 (Conflict Detection and Resolution), 궤적 회복(Trajectory Recovery)

ABSTRACT

The demand for the integration of large UAVs in regular airspaces that have been exclusively used by manned aircraft has been increasing. Detect and Avoid, a key technology that enables the integrated operation, is actively being researched.

In this thesis, the risk is evaluated using the Well Clear metrics, which is currently in the process of becoming an international standard, assuming that state information of other intruder aircraft is given. We investigated the starting points of the avoidance maneuver, types of the maneuvers, and the point of return to the original trajectory.

Existing fast-time simulation system has been improved to enable vertical and horizontal maneuvers with given requirements and modification of waypoints. Several candidate maneuvers are generated by integrating the NASA developed avoidance maneuver generation tool to the fast-time simulation system. And then, cost functions are used to investigate how the optimal avoidance maneuver and the selection of the trajectory return point change.

In the case of the five encounter scenarios presented in this thesis, horizontal avoidance maneuver is found to be the most efficient.

Keyword: DAA(Detect And Avoid), DAIDALUS(Detect And Avoid Alerting Logic for Unmanned System), DWC(DAA Well Clear), Conflict Detection and Resolution, Trajectory Recovery

목 차

1. 서론	1
1.1. 연구개요	1
1.1.1. 연구 목표	3
1.1.2. 연구 방향	3
2. 충돌 감지 알고리즘	5
2.1. Divide And Conquer (DAC)	5
2.1.1. 연산량 감소를 위한 Divide and Conquer 기법 적용	5
2.1.2. 거리 기준 적용한 위험 알람 로직 개선	7
2.2. Wellclear 알고리즘	9
2.2.1. Wellclear 정의	9
3. 충돌 회피 알고리즘	12
3.1. Traffic Collision Avoidance System (TCAS)	12
3.1.1. TCAS의 정의	12
3.1.2. TCAS 알고리즘	13
3.2. DAIDALUS 알고리즘	22
3.3. 배속 시뮬레이션 (Fast-Time-Simulation)	26
4. 비용 함수의 설정	31
4.1. 기동 종류 비용	32
4.2. 기동 체류 시간 비용	32
4.3. 충돌 위험 비용	33

5. 충돌 회피 검증	35
5.1. Case 1: 마주보는 순항 시나리오	35
5.2. Case 2: 예각 횡단 순항 시나리오	42
5.3. Case 3: 둔각 횡단 순항 시나리오	48
5.4. Case 4: 순항 상승 시나리오	55
5.5. Case 5: 순항 하강 시나리오	61
6. 결론	67
7. 참고문헌	68



표 목 차

표 1 위험 단계별 Well Clear 기준과 Score	11
표 2 Sensitivity Level에 따른 기준	16
표 3 RA의 종류	17
표 4 조우하는 순항 시나리오	36
표 5 Case 1 시나리오에 대한 비용 함수	40
표 6 예각 횡단 순항 시나리오	42
표 7 Case 2 시나리오에 대한 비용 함수	47
표 8 둔각 횡단 순항 시나리오	48
표 9 Case 3 시나리오에 대한 비용 함수	53
표 10 순항 상승 시나리오	55
표 11 Case 4 시나리오에 대한 비용 함수	60
표 12 순항 하강 시나리오	61
표 13 Case 5 시나리오에 대한 비용 함수	66



그림 목 차

그림 1 무인기 자율 비행 제어 기술등급	2
그림 2 연구방향 및 순서도	3
그림 3 DAC 이전 알고리즘 계산도	5
그림 4 DAC 이후 알고리즘 계산도	5
그림 5 Protected Zone 설정 구역	6
그림 6 DAC 적용한 시간별 궤적	6
그림 7 단순 거리 기준 충돌 감지	7
그림 8 CA 알람이 울린 HiTL의 CWP화면	8
그림 9 Well clear 정의	10
그림 10 TCAS TA화면 예시	12
그림 11 TCAS 알고리즘 순서도	14
그림 12. TCAS의 TA, RA 영역	15
그림 13 Positive RA	18
그림 14 Negative RA	19
그림 15 Altitude Crossing 경우	19
그림 16 Strengthening RA	20
그림 17 Reversed RA	21
그림 18 DAIDALUS 알고리즘 구조	22
그림 19 Constant Velocity Aircraft Projection	23
그림 20 DAIDALUS의 Bands 예시	23
그림 21 DAIDALUS Band의 충돌 감지 구역	24
그림 22 DAIDALUS 의 Resolution 예시	25
그림 23 항공기 운동 모델	26
그림 24 항공기 가로 운동과 세로 운동 모델 과정	26
그림 25 변경된 항공기 운동 모델 구조 형태	27
그림 26 재구성된 궤적 생성 모델	28

그림 27 DAIDALUS의 충돌 구역과 회피 가능 기동	29
그림 28 최소 비용 함수 기동을 선택 시 기동 상황	30
그림 29 전체 알고리즘 순서도	34
그림 30 마주보는 순항 시나리오 궤적	36
그림 31 4가지 회피 기동	37
그림 32 Altitude Resolution	38
그림 33 Track Resolution	38
그림 34 Ground Speed Resolution	39
그림 35 Vertical Speed Resolution	39
그림 36 기동에 따른 WCS	40
그림 37 CPA 지점 5초 전 회복 기동한 WCS	41
그림 38 CPA 지점 10초 전 회복 기동한 WCS	41
그림 39 예각 횡단 순항 시나리오 본 궤적	42
그림 40 Altitude Resolution	43
그림 41 Altitude Resolution WCS	43
그림 42 Track Resolution	44
그림 43 Track Resolution WCS	44
그림 44 Ground Speed Resolution	45
그림 45 Ground Speed Resolution WCS	45
그림 46 Vertical Speed Resolution	46
그림 47 Vertical Speed Resolution WCS	46
그림 48 최적 기동 선택 후 궤적	47
그림 49 둔각 횡단 순항 시나리오 본 궤적	48
그림 50 Altitude Resolution	49
그림 51 Altitude Resolution WCS	49
그림 52 Track Resolution	50
그림 53 Track Resolution WCS	50
그림 54 Ground Speed Resolution	51
그림 55 Ground Speed Resolution WCS	51

그림 56 Vertical Speed Resolution	52
그림 57 Vertical Speed Resolution WCS	52
그림 58 회피 기동 중 선택 후 궤적	53
그림 59 회피 기동 중 선택 후 궤적	54
그림 60 순항 상승 시나리오 본 궤적	55
그림 61 Altitude Resolution	56
그림 62 Altitude Resolution WCS	56
그림 63 Track Resolution	57
그림 64 Track Resolution WCS	57
그림 65 Ground Speed Resolution	58
그림 66 Ground Speed Resolution WCS	58
그림 67 Vertical Speed Resolution	59
그림 68 Vertical Speed Resolution WCS	59
그림 69 회피 기동 중 선택 후 궤적	60
그림 70 순항 하강 시나리오 궤적	61
그림 71 Altitude Resolution	62
그림 72 Altitude Resolution WCS	62
그림 73 Track Resolution	63
그림 74 Track Resolution WCS	63
그림 75 Ground Speed Resolution	64
그림 76 Ground Speed Resolution WCS	64
그림 77 Vertical Speed Resolution	65
그림 78 Vertical Speed Resolution WCS	65
그림 79 회피 기동 중 선택 후 궤적	66

Glossary

ACL	Autonomous Control Levels
ADS-B	Automatic Dependent Surveillance-Broadcast
ALIM	Altitude Limitation
BADA	Base of Aircraft Data
CA	Collision Avoidance Conflict Alert
CPA	Closest Point to Approach
CWP	Controller Working Position
DAA	Detect And Avoid
DAC	Divide And Conquer
DAIDALUS	Detect And Avoid Alerting Logic for Unmanned System
DMOD	Distance Minimum
DWC	DAA Well Clear
ECEF	Earth Centered Earth Fixed
FAA	Federal Aviation Administration
FIR	Flight Information Region
FTS	Fast Time Simulation
GRACE	Generic Resolution Advisor and Conflict Evaluator
HITL	Human In The Loop
HMD	Horizontal Miss Distance
LoWC	Loss of Well Clear
MOPS	Minimum Operational Performance Standards
RA	Resolution Advisory
RTCA	Radio Technical Commission for Aeronautics
SAA	See And Avoid SenseAndAvoid
SL	Sensitivity Level
SST	Self-Separation Threshold
TA	Traffic Advisory
TAS	True Air Speed
TCAS	Traffic Collision Avoidance System
WCS	Well Clear Score
WCV	Well Clear Volume
ZTHR	Altitude Threshold

1. 서론

1.1. 연구개요

무인항공기는 군수와 민간에서 활발히 개발되어 있고, 기술은 꾸준히 발전해 종류와 개수도 기하급수적으로 증가하였다. 최근 높은 공역에서 무인항공기 이용 요구가 증가하고 있는 한편, 무인항공기의 안전기능 미비로 인하여 공중충돌 및 비행안전 문제가 심각하게 부각되고 있다. 유인 항공기의 경우 조종사가 관제 및 TCAS에 의한 충돌 방지 지시를 인지하여 직접 보고 회피하는 See-And-Avoid를 수행한다. 무인항공기의 경우 현재 유인항공기와 동등한 충돌 회피 능력을 확보해야 하며 주변을 탐지하고 회피하는 기능을 필요로 한다. 이에 많은 국가에서 Sense And Avoid (SAA)와 Detect And Avoid (DAA)와 관련된 연구를 수행하고 있다. [1]

무인항공기의 충돌회피 기술은 ADS-B와 트랜스폰더 등의 탐지 장비와 연결하여 SAA를 구현하는 연구가 진행 중이며, 무인항공기도 유인항공기와 동등한 수준의 충돌 감지 및 회피 능력이 필요하며, 이에 대한 명확한 기준 수립이 필요하다.[1]

DAA는 무인 항공기에서 주변 항공기 또는 주변 기타 장애물로부터 위험을 감지하고 이를 해소하기 위한 시스템으로 위험 해소를 위한 시스템 요구성능과 기준 수립에 대해 현재까지 활발히 논의되고 있으며, 2017년 무인 항공기의 DAA를 위한 절차와 성능 조건이 RTCA DO-365으로 정의되어 있다.[5]

충돌 가능성을 감지하는 방법은 일반적으로 침입기의 궤적과 자신 항공기의 거리 추정 방법을 사용하고, 또 다른 확률 추정방법도 있다. 확률 추정방법은 항공기와 침입기의 모델을 설계하여 위치, 속도, 방위각, 고도변화를 미지변수로 몬테카를로 시뮬레이션을 통한 충돌 발생 가능성을 계산한다. 또한 현대에는 ADS-B항공기를 이용하여 기하학적 방법으로 충돌 감지하는 방법이 성행하고 있다.[15]

현재는 법규에 의해 무인항공기는 원격조종사 아래에 비행을 실시하도록 되어있다. 기존의 자동비행(Autopilot)의 경우 미리 입력한 경로에 따라 비행을 하거나, 조종사의 조작에 의해 주어진 조건의 비행 상태를 유지하는 기능으로, 공중충돌, 항공기 고장, 기상조건 변화 등에 항공기 능동적으로 대처할 수

없다. 이에 차세대 항공 교통 관제 시스템과 결합하여 능동적으로 대처할 수 있는 자율 비행(Autonomous Flight)에 대해 관심이 높아지고 있다. 자율 비행은 항공기의 예기치 않은 이 상황발생에 대해서도 항공기가 스스로 인지하고, 판단한 후 대처하는 것을 의미한다.[2]

미국의 경우 향후 25년간 무인 항공기를 비롯한 대부분의 무인 시스템에 자율능력과 관련된 기술을 개발하고 적용하는 것을 목표로 삼고 있으며[3], 그림 1처럼 무인기 자율 비행 제어 수준을 10단계로 구분하고 있다.[4]

- 1단계: 지상의 조종사가 조종하여 무인기를 운용하는 기술 수준
- 2단계: 실시간으로 항공기 상태를 진단하는 단계
- 3단계: 항공기 고장과 비행조건에 능동적으로 실시간 대처하는 기능을 구현하는 수준
- 4단계: 주변상황이나 임무변화에 대처해 항공기 자체적으로 경로를 재구성하는 단계
- 5단계: 편대비행 기술
- 6단계: 편대운용 시 전술적인 임무재고성하는 기능시현단계를 의미
- 7단계: 편대비행을 통해 전술적 목적에 능동적으로 대처하는 기술
- 8단계: 항공기간 분산연산을 통해 지상관제 없이 이러한 기능들을 구현
- 9단계: 전술적 단계를 넘은 전략적 그룹운영 수준
- 10단계: 자율비행에 의한 군집비행을 실현하는 것으로 구성

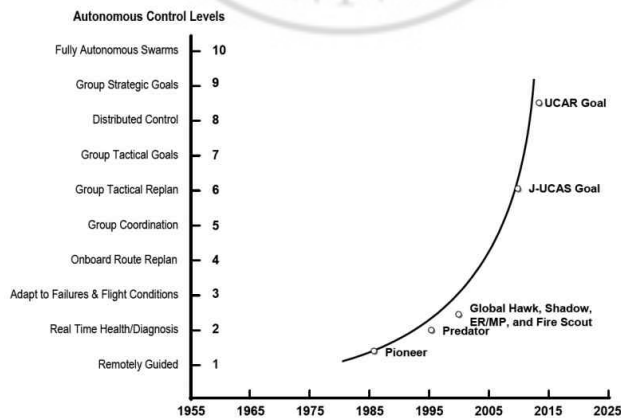


그림 1 무인기 자율 비행 제어 기술등급[4]

1.1.1. 연구 목표

본 논문에서는 4단계에 준하는 자율비행 제어에 필요한 충돌 감지와 회피 알고리즘을 통해 항공기 스스로 경로점을 재설정하고, 최적경로를 배속시물레이션으로 연산하는 알고리즘에 대해 제시한다.

ADS-B를 통해 무인 항공기의 상태변수로 충돌을 감지하고, 배속 시물레이션을 통해 충돌 회피 기동을 계산한다. 회피할 수 있는 기동의 범위 안에서 선택하는 기준과 충돌 회피가 종료된 이후 본 궤적으로 복귀하는 시점에 대한 연구로, 비용함수에 따라 선택의 기준이 어떻게 달라지는지 살펴보고 선택의 기준을 제시한다.

1.1.2. 연구 방향

첫 번째 단계에서 충돌을 감지 알고리즘에서 DAC기법을 사용하여 인접한 구역에서 충돌을 감지하고, Well clear 알고리즘을 이용하여 충돌 감지의 위험도를 판정한다. 두 번째 단계의 충돌 회피 알고리즘에서는 BADA 기반의 궤적생성 모델을 통해 무인항공기가 충돌을 감지하였을 때 TCAS 기반의 DAIDALUS 알고리즘으로 충돌 감지 구역을 계산하고, 이를 회피 할 수 있는 기동을 배속 시물레이션을 수행할 수 있도록 한다. 또한 충돌 회피가 종료된 이후에 본 궤적으로 돌아오는 기준을 제안한다. 세 번째 단계에서는 그러한 기동의 범위 안에서 비용 함수(Cost Function)에 따라 기동의 선택 기준을 제시한다. 이러한 충돌 회피 궤적에서 위험도가 일어나는 부분이 있는지 Well Clear를 통해 다시 한 번 더 검증하였다.

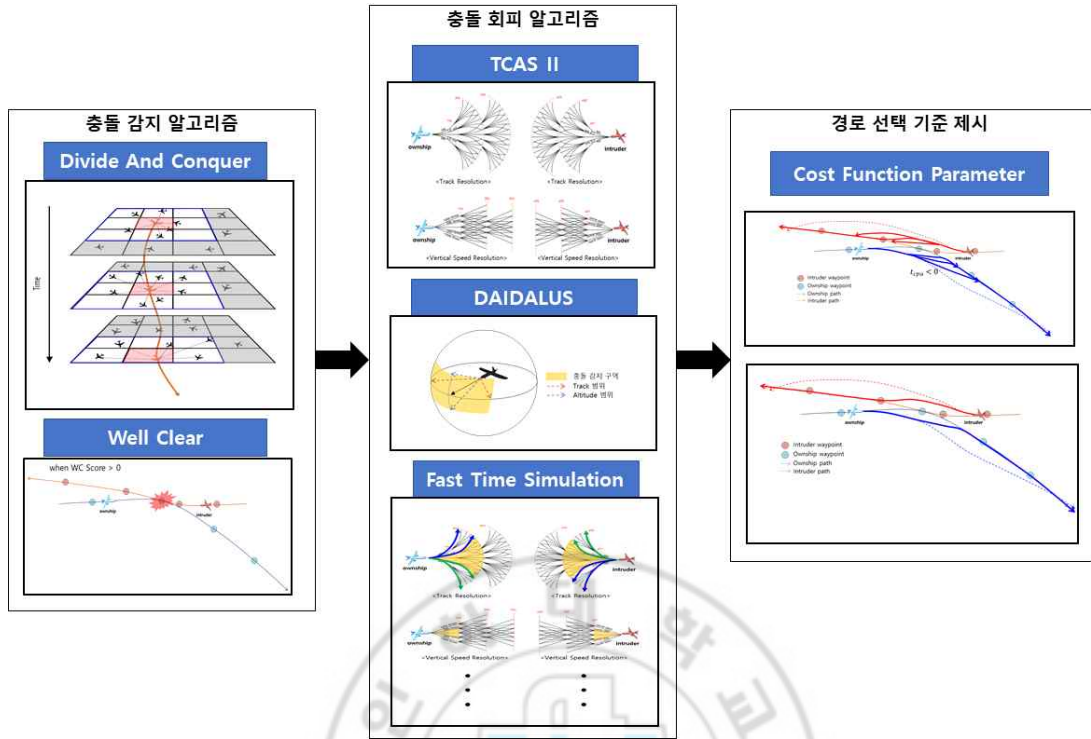


그림 2 연구방향 및 순서도

2. 충돌 감지 알고리즘

2.1. Divide And Conquer (DAC)

2.1.1. 연산량 감소를 위한 DAC 기법 적용

항공기간의 충돌을 방지하기 위해 위험알람 기능을 구현하기 위해 항공기들 간의 거리를 계산하는 과정이 있다. 만약 한 두 대가 있다면 계산은 한번으로 끝나지만 수 만 대의 항공기가 국소적인 시간에 함께 있다면 각 한 대의 비행기와 다른 항공기들 간의 거리를 전부 계산을 하게 된다. 이는 필요 없는 연산이 많아지게 되며, 시간적으로, 메모리 적으로 낭비이다. 이를 방지하기 위해 특정 거리를 설정하여 각 축마다 구역을 만드는 Divide and Conquer(DAC)기법을 적용시켰다.

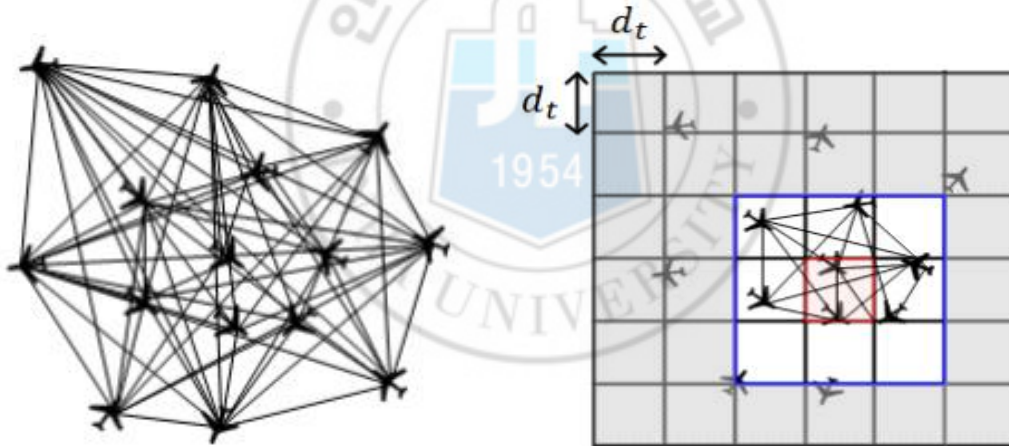


그림 3 DAC 이전 알고리즘 계산도

그림 4 DAC 이후 알고리즘 계산도

그림 3 은 DAC기법을 적용시키기 이전의 알고리즘 계산도를 시각적으로 보여준 것이다. 그림 4 은 DAC기법을 적용시켜 일정한 간격인 d_t 로 나눈 블록도를 나타내었다.

각 항공기마다 모든 항공기들을 계산하는 것이 아니라 DAC기법을 적용하여 인접해 있는 구역에 있는 비행기들만 계산할 수 있도록 코드에 적용을 시켰다.

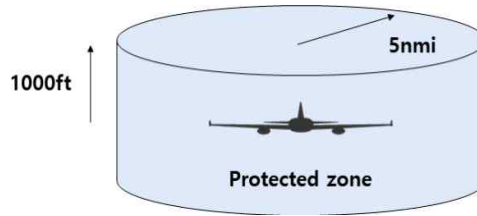


그림 5 Protected Zone 설정 구역

d_4 는 그림 5의 위험 알람 기준으로 수평 길이인 5nmi로 설정을 하였다. DAC에서 1 구역을 나눌 때는 인천의 FIR의 최소와 최대 위경도를 참고하여 초기값으로 설정하였다. 최소 위도는 30.0, 최고는 38.633333, 최소경도는 124.0, 최고는 133.65를 참조하였다. 초기값으로 Lambert Conformal의 과정으로 ECEF 좌표계로 변환하여 그림 4과 같이 구역을 나눠 각 구역마다 좌표를 매겼다. 항공기가 위치해있는 좌표를 x좌표에서 몇 구역인지, y좌표에서 몇 구역인지 구한 다음 이 구역 좌표의 차이가 1이면 인접하다고 판단하였다. 이후 이 조건을 만족하면 각자의 거리를 구하도록 알고리즘을 개선하였다.

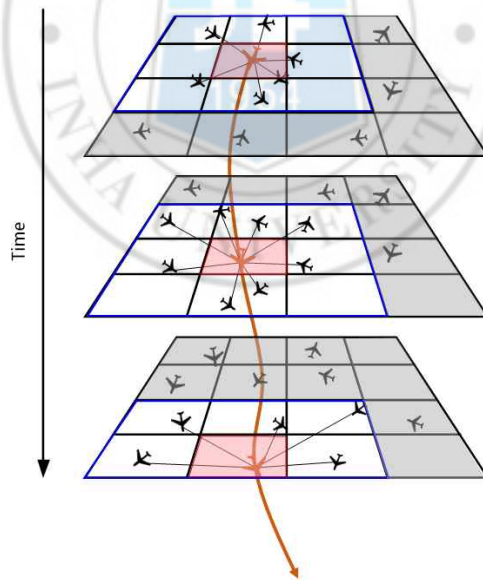


그림 6 DAC 적용한 시간별 궤적

2.1.2. 거리 기준 적용한 위험 알람 로직 개선

항공기의 충돌 방지를 위해 위험 알람(CA)이 울리는 시스템을 HiTL 시뮬레이션에 적용시켰다. 이의 기준은 그림 5의 거리와 시간 기준으로 설정하고 항공기의 현재 위치 정보를 받아 이들을 계산하여 위의 기준을 충족시키면 알람이 울린다. 미래의 위치를 구하는 방법은 아래의 수식으로 표현하였다. 알고리즘이 검증이 된 후에 미래 위치에 대한 거리기준을 만족하는지에 대해 살펴보았다.

$$s_{t+1} = s_t + \frac{ds}{dt} * dt \quad (\text{수식 1})$$

1초마다 항공기의 위치가 받아들여지므로 dt 의 값을 1초로 지정했다.

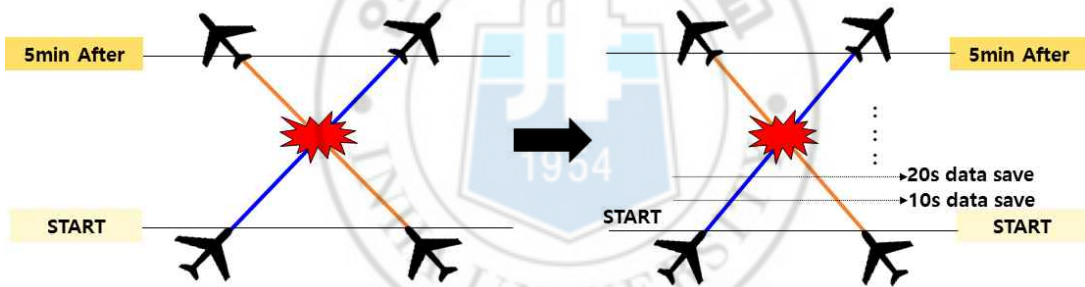


그림 7 단순 거리 기준 충돌 감지

이 방법은 위의 그림 7 에서의 상황처럼 현재 위치와 5분 후 위치에서는 충돌을 하지 않지만 그 시간 사이에서 충돌이 발생할 모순이 발생한다. 이를 방지하기 위해 10초 간격으로 거리를 계산 하여 CA 거리 기준에 부합하는지 저장을 하는 방법을 채택하였다. 10초 간격으로 데이터가 쌓인 것에서 하나의 값이라도 만족을 하면 CA 알람이 울려 경고 표시가 나타나도록 하였다.

2.2. Well Clear 알고리즘

2.2.1. Well Clear 정의

Well Clear 라는 용어는 FAA의 규약인 Part 91 “General Operating and Flight Rules” 의 91.113 “Right of Way Rules” 와 91.181 “ Course to be flown” [8] 에 정의 되어 있다. Well Clear는 항공기의 안전을 보장하는 구역을 의미하며, 유인항공기는 조종사가 주관적으로 결정하였다. 이에 무인항공기의 충돌 방지하는 기준으로 Well Clear를 정의하기 위해 조종사의 주관적 분리 기준을 정량화하는 연구가 진행 되었으며 최근에 RTCA “MOPS for Detect and Avoid (DAA) Systems” DO-365에서 DWC를 정의 하였다.[5] Well Clear는 Hazard Zone에서 유지되며, DO-365에서 Hazard Zone을 아래의 식 2의 조건으로 정의를 하였다.

- $[r \leq S^*] \text{ AND } [HMD_p \leq HMD^*] \text{ AND } [d_h \leq h^*] \equiv \text{TRUE}$ 수식 2
- $S^* = \max(DMOD, \frac{1}{2}(\sqrt{\dot{r}\tau_{mod}})^2 + 4DMOD^2 - \dot{r}\tau_{mod}^*)$
- $HMD = \begin{cases} \sqrt{(d_x + v_{rx}t_{CPA})^2 + (d_y + v_{ry}t_{CPA})^2} \\ -\infty \end{cases}$
- $d_h = |h_2 - h_1|$

where

$$d_x = x_2 - x_1$$

$$d_y = y_2 - y_1$$

$$v_{rx} = \dot{x}_2 - \dot{x}_1$$

$$v_{ry} = \dot{y}_2 - \dot{y}_1$$

$$t_{CPA} = \begin{cases} -\frac{d_x v_{rx} + d_y v_{ry}}{v_{rx}^2 + v_{ry}^2} & \text{f or } r \geq DMOD \\ 0 & \text{f or } r < DMOD \end{cases}$$

$$\tau_{mod} = \begin{cases} 0 & \text{f or } r_{xy} \leq DMOD \\ -\frac{r_{xy}^2 - DMOD^2}{r_{xy}r_{xy}} & \text{f or } r_{xy} > DMOD \\ \infty & \text{f or } r_{xy} \leq DMOD \end{cases}$$

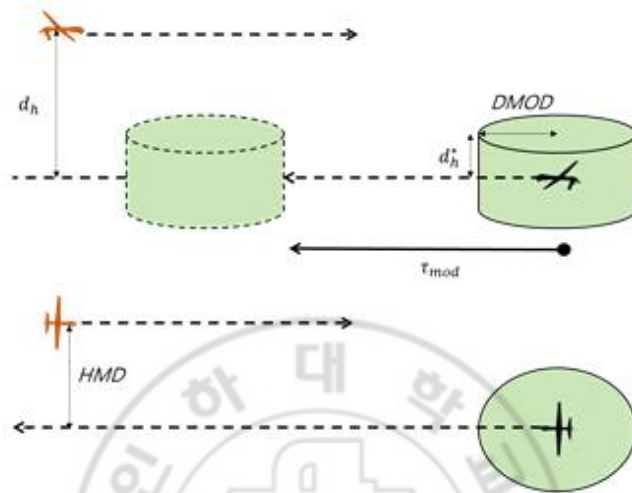


그림 9 Well clear 정의

Hazard Zone은 그림 9과 같은 녹색 경계를 말한다. 또한 각 파라미터의 Threshold는 다음과 같이 표 1로 정의할 수 있다. Well clear의 기준은 Modified Tau가 35초 이하 일 때, HMD가 4000ft 이하일 때, 항공기간의 수직거리가 450ft 이하 일 때를 나타내며, 이를 어길 시 Loss of Well Clear (LoWC)라 칭하고 충돌 위험 상태임을 인지한다. 각 Threshold에 따라 충돌 위험 단계 분류가 가능하다. 각 분류별로 Well Clear Score (WCS)를 점수화 하여 진행한 연구[16]을 적용하여 충돌 위험 정도를 식별하였다.

표 1 위험 단계별 Well Clear 기준과 Score[5]

Alert Type		Preventive Alert	Corrective Alert	Warning Alert	LoWC
Score		1	2	3	4
Alert Level		Caution	Caution	Warning	Warning
Hazard Zone	τ_{mod}^* (sec)	35	35	35	35
	HMD* (ft)	4000	4000	4000	4000
	d_h^* (ft)	700	450	450	450
Hazard Zone Alert Times	Minimum Average Time of Alert(sec)	55	55	25	-
	Late Threshold (sec)	20	20	15	-
	Early Threshold (sec)	75	75	35	-

3. 충돌 회피 알고리즘

3.1. Traffic Collision Avoidance System (TCAS)

3.1.1. TCAS의 정의 [6] [7]

TCAS는 유인항공기에 탑재되는 항공기의 공중충돌을 방지하기 위한 시스템이다. 관제시스템에 독립적으로 구현하여 자신의 항공기 주변에서 약 40마일까지 정보를 받아들여 상대 항공기에 대한 위치와 속도, 고도 정보를 이용하여 충돌을 방지하고 이동 경로를 예측한다.

TCAS I은 공중충돌 방지의 기초적인 기능을 수행하는 시스템으로, 단지 충돌 경보만 알릴뿐 조종사에게 회피 방법을 제시하지 않는다. TCAS II는 TCAS I의 모든 기능을 구현하며 추가로 조종사에게 회피 방법 또한 화면과 음성으로 지시할 수 있고, 대다수 상용 항공기에 탑재되어 있다. 하지만 회피 방법은 수평방향은 전혀 고려하지 않고, 수직방향으로 한정 되어 고도 변경만으로 항공기들 간의 간격을 넓힌다.

주변 항공기 중 다른 항공기의 충돌 가능성을 감지하면 먼저 TCAS는 예측된 충돌시간 20~48초 전에 Traffic Advisory (TA)를 조종사에게 알린다. 조종사의 화면은 다음 그림과 같이 침입항공기에 강조를 해주며 “Traffic! Traffic!” 라고 오디오 알람으로 알려준다. 이러한 경고는 조종사가 다른 항공기를 시각적으로 획득하고 잠재적인 회피 기동을 준비할 수 있게 한다.

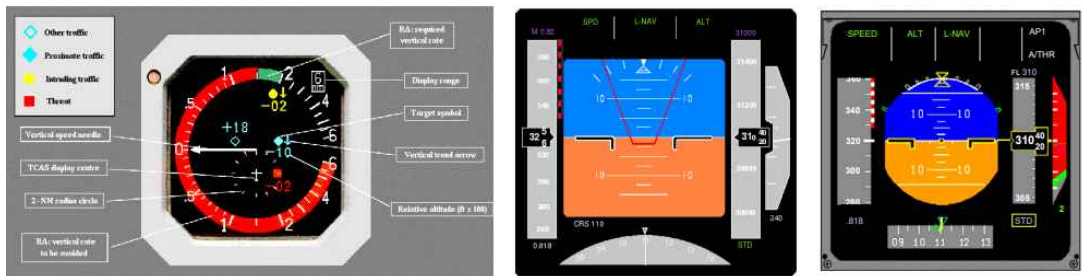


그림 10 TCAS TA화면 예시

TA알람이 울리고 나서도 계속해서 충돌 위험이 감지되고 심화된다면 Resolution Advisory (RA)단계에 접어들어 충돌을 회피할 수 있는 수직기동을 알려준다. RA는 예측된 시간 15~25초전에 알려주며, 수직 권고사항으로는 “Climb! Climb!” 또는 “Descend! Descend!” 등 다양하게 존재하며, 이는 TCAS 자체의 알고리즘과 경험식에 의해 결정된다.

차세대 TCAS III는 수직 회피 뿐 아니라 수평 회피까지 포함하여 개발할 예정이었으나 항공기 간의 거리가 멀어 질수록 오차가 커지는 현상으로 회피 기동을 사용할 만큼의 성능을 제공할 수 없어 개발이 중지되었다.[8]

3.1.2. TCAS 알고리즘

TCAS의 알고리즘은 트랜스폰더를 통해 받은 침입 항공기의 고도, 속도, 위치 정보를 통해 직선 선형 보간법을 통해 충돌지점인 Closest of Point Approach(CPA)의 위치를 계산하여 충돌을 감지한다. 그 이후 각 상황에 맞는 자신항공기(Ownship)의 회피 기동을 선택하여 권고사항(Advisory)을 표현한다. 자세한 알고리즘의 순서도는 그림 12와 같다.[17]

TCAS의 충돌을 감지하는 조건은 다음 식 과 같다. 이 두 조건을 동시에 만족 시 TA와 RA의 영역으로 판단하며 각 영역의 알고리즘을 따라 반응이 결정된다.

- $d_h \leq ZTHR \text{ AND } dr \leq DMOD$
- $t_{CPA} \leq \tau$

where

$$t_{CPA} = \frac{DMOD^2 - r_{xy}^2}{r_{xy}r_{xy}}, \quad \tau = -\frac{r}{r}$$

TA와 RA의 영역은 다음 그림과 같이 표현 할 수 있다. TA는 CPA 지점에 도달하기 20~40초 전 영역을 의미 하며, RA는 15~35초 전 영역을 의미한다. 이 영역을 나누는 기준은 고도의 Sensitivity Level (SL)에 따라 나뉘지며, 그 Level에 따라 고도 분리 기준(ALIM)과 변수의 기준은 다음 표로 알 수 있다.[6]

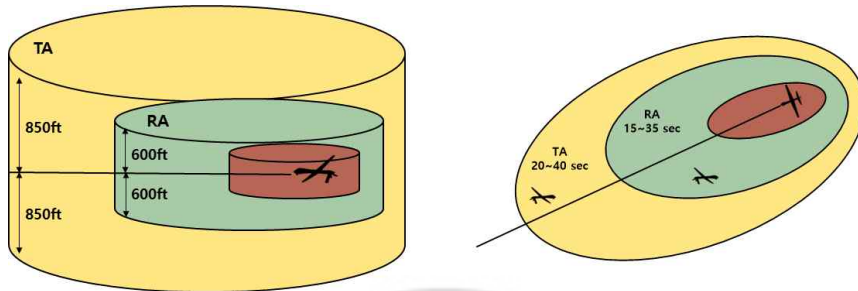


그림 11. TCAS의 TA, RA 영역

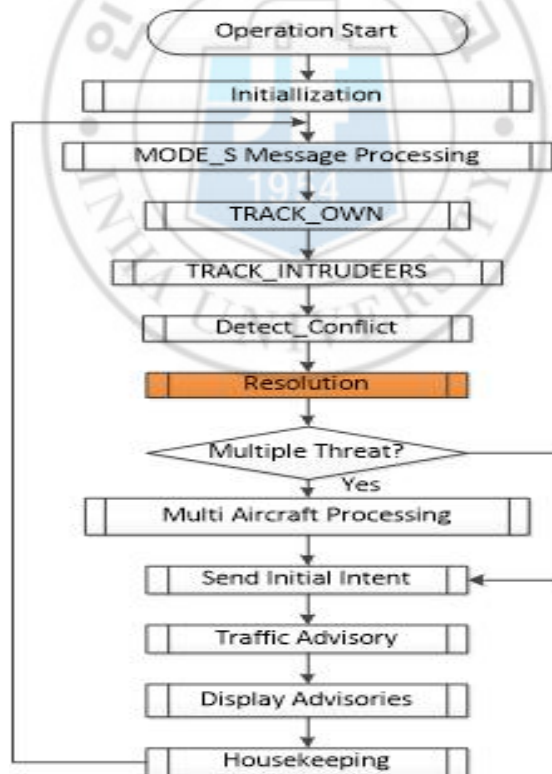


그림 12 TCAS 알고리즘 순서도[17]

표 2 SL에 따른 기준

Own Altitude (ft)	SL	Tau (seconds)		DMOD (nmi)		ZTHR (ft) Altitude Threshold		ALIM (ft)
		TA	RA	TA	RA	TA	RA	RA
< 1000(AGL)	2	20	N/A	0.3	N/A	850	N/A	N/A
1000-2350	3	25	15	0.33	0.20	850	600	300
2350-5000	4	30	20	0.48	0.35	850	600	300
5000-10000	5	40	25	0.75	0.55	850	600	350
10000-20000	6	45	30	1.00	0.80	850	600	400
20000-42000	7	48	35	1.30	1.10	850	700	600
>42000	7	48	35	1.30	1.10	1200	800	700

RA의 영역에 들어오면 항공기가 충돌을 회피 할 수 있는 Resolution을 안내한다. Resolution의 선택기준은 TCAS의 복잡한 알고리즘을 통해 제시된다. 조종사에게 특정 고도변화율로 항공기의 기동을 제시하고 첫 RA에 대해 5초 이내에 반응해야 하며, 충돌 감지가 해소 되지 않고 계속 발생한다면 2.5초 이내로 반응해야 한다.

충돌 회피 종류는 Positive와 Negative로 두 가지로 나뉘며, Positive는 상승 또는 하강 명령을 하는 것을 말하고, Negative는 수직속도 제한하는 명령의 방법으로 특정 고도 변화율을 제시한다. RA에서 발생할 수 있는 모든 종류는 다음의 표와 같다.

표 3 RA의 종류

RA Type	Upward Sense		Downward Sense	
	RA	Required Vertical Rate (fpm)	RA	Required Vertical Rate (fpm)
Positive (Corrective)	Climb	1500 to 2000	Descend	-1500 to -2000
Positive (Corrective)	Crossing Climb	1500 to 2000	Crossing Descend	-1500 to -2000
Positive (Corrective)	Crossing Maintain Climb	1500 to 4400	Crossing Maintain Descend	-1500 to -4400
Positive (Corrective)	Maintain Climb	1500 to 4400	Maintain Descend	-1500 to -4400
Negative (Corrective)	Reduce Descent	0	Reduce Climb	0
*Negative (Corrective)	Reduce Descent	>-500	Reduce Climb	<500
*Negative (Corrective)	Reduce Descent	>-1000	Reduce Climb	<1000
*Negative (Corrective)	Reduce Descent	>-2000	Reduce Climb	<2000
Negative (Preventive)	Do Not Descend	>0	Do Not Climb	<0
Negative (Preventive)	Do Not Descend > 500 fpm	>-500	Do Not Climb > 500 fpm	<500
Negative (Preventive)	Do Not Descend > 1000 fpm	>-1000	Do Not Climb > 1000 fpm	<1000
Negative (Preventive)	Do Not Descend > 2000 fpm	>-2000	Do Not Climb > 2000 fpm	<2000

TCAS의 RA를 결정하는 알고리즘을 자세히 살펴보면, 가장 먼저 Positive 명령의 방향을 결정한다. 자신 항공기의 고도 변화율을 기본으로 1500fpm으로 설정하여 상승 또는 하강을 각각 진행했을 때, CPA지점에서의 예상 위치를 계산한다. 예상한 위치에서 침입항공기와 상대고도를 구한다. 이때, 상대 고도가 고도 분리 기준(ALIM)을 만족하는 방향을 선택하게 된다. 다음 그림의 예시를 보면 상승과 하강의 CPA지점에서의 예상 위치를 보았을 때, 상대고도 A와 B를 비교하여 고도 분리 기준(ALIM)을 만족하는 하강을 선택하게 된다.

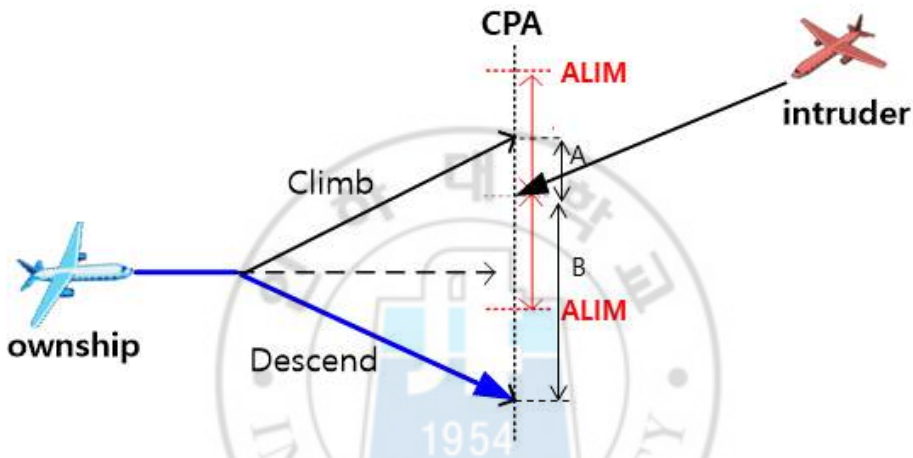


그림 13 Positive RA

두 번째로는 Negative RA를 선택하는 단계로 넘어간다. 이전 단계에서 정해진 수직 방향으로 고도 변화율을 0, 500, 1000, 1500fpm으로 변경하여 CPA지점의 예상 위치를 계산한다. 그 중 고도 분리 기준을 넘을 수 있고, 현재 기동과 비교하여 변화율이 가장 적은 기동을 선택하게 된다. 예시 그림에서는 최소 고도 분리 기준(ALIM)을 유지 할 수 있으며, 현재 기동에서 변화가 제일 작은 기동인 “Limit Descent 500fpm”을 선택하게 된다.

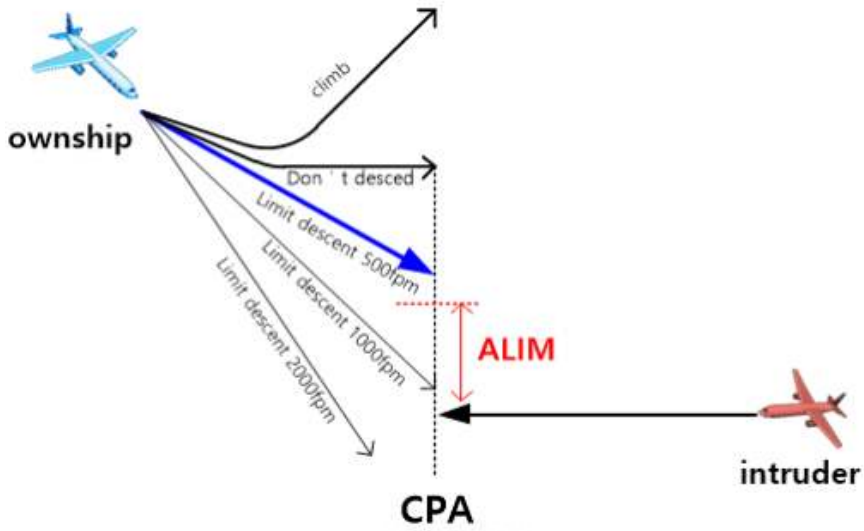


그림 14 Negative RA

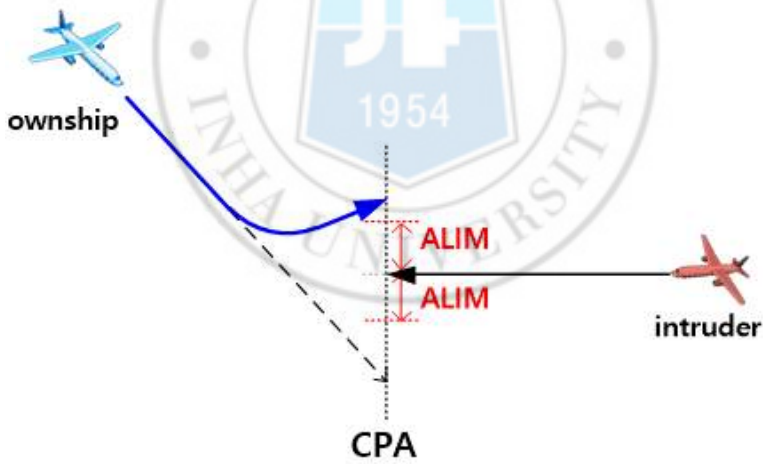


그림 15 Altitude Crossing 경우

그리고 이후 다양한 고려사항들을 살펴보고, 최종 기동을 선택하게 된다. 고려 사항 중 하나는 자신 항공기와 상대 침입기가 CPA이전 지점에서 고도가 같아지는 여부이다. 고도 분리 기준을 지켜도 고도가 동일한 지점이 된다면 충돌위험이 크기 때문에 다음 그림과 같이 CPA지점 이전에서 고도가 같아지는 기동은 피하도록 한다.

두 번째 고려사항은 TCAS II의 Version 7.0부터 적용이 되었다. 2002년 독일 Uberlingen 상공에서 발생한 항공기 충돌사건을 필두로 TCAS의 강건성이 문제가 대두되었다. 한 항공기는 상승 명령을, 다른 항공기는 하강 명령을 TCAS에서 올렸지만, 관제사의 명령은 상승이 아닌 하강명령으로 전해져 충돌로 이어졌다. 이와 같은 사고를 방지하고자 TCAS는 반전 명령(Reversed RA)와 지시 강화(Strengthening RA) 또한 고려하여 고도 변화율을 제시한다. 아래의 그림과 같이 침입항공기가 정해진 명령이 아닌 비행경로를 진행한다면 방향을 바꾸는 반전 명령 또는 정해진 방향을 더 강화하는 고도 분리를 충분히 할 수 있는 기동을 선택하게 된다.

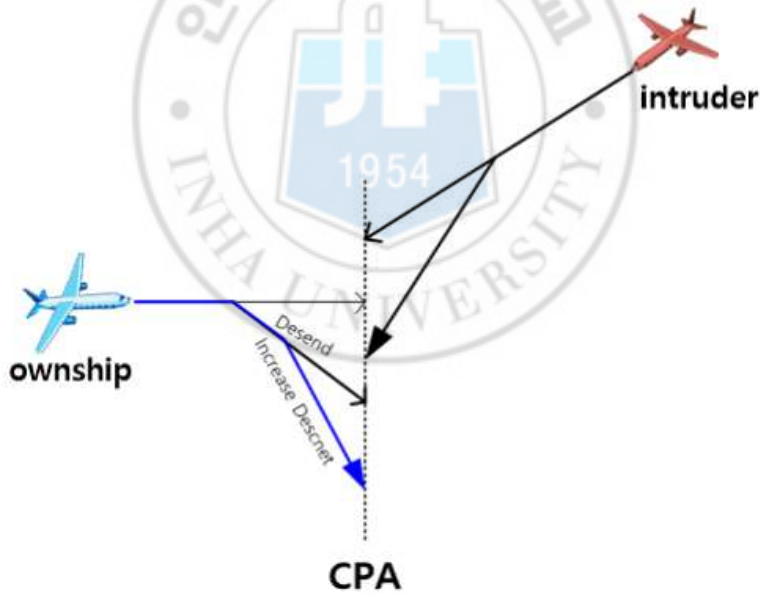


그림 16 Strengthening RA

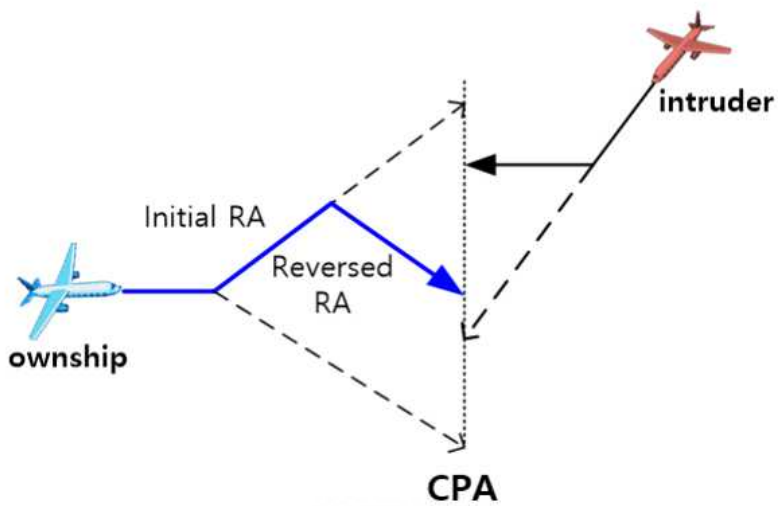


그림 17 Reversed RA



3.2. DAIDALUS 알고리즘

Detect And Avoid Alerting Logic for Unmanned System (DAIDALUS)는 NASA에서 개발한 무인항공기 충돌 회피 기능을 제공하는 시스템이다. 이 시스템은 자체 분리(Self-Separation)와 충돌 경고 알고리즘으로 구성되어 무인항공기의 원격 조종사에게 상황을 인식 할 수 있게 도와준다. [9]

DAIDALUS에서 Well Clear 영역을 Well Clear Volume (WCV)으로 정의하고 이는 항공기의 상태인 거리와 시간의 변수로 판단된다. 이러한 변수들은 TCAS II의 RA 로직에 연관되며, 이는 DAIDALUS가 TCAS II의 수학적 접근을 가능하게 만든다. [10]

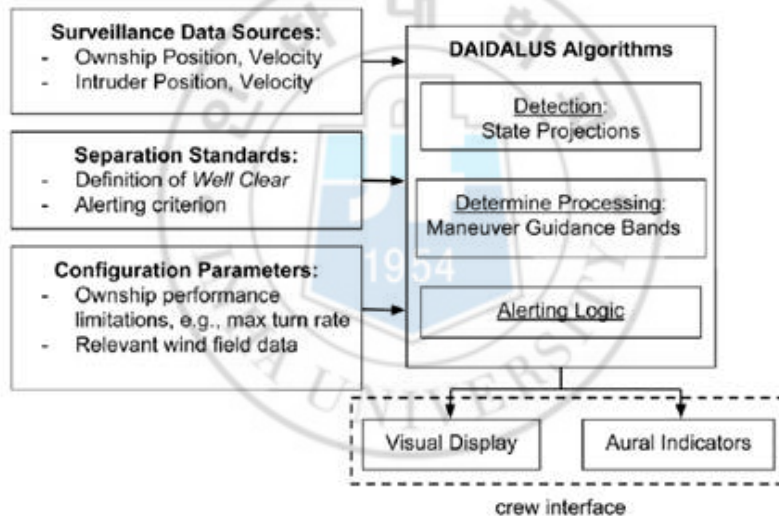


그림 18 DAIDALUS 알고리즘 구조

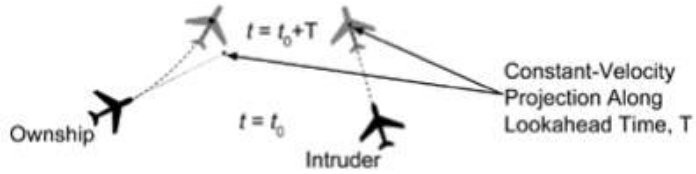


그림 19 Constant Velocity Aircraft Projection

그림 18을 보면 자신항공기(Ownship)의 위치와 속도, 침입항공기(Intruder)의 위치와 속도의 데이터, Well Clear 분리 기준과 각 항공기의 제한 기동의 입력 데이터를 주입하면 DAIDALUS 알고리즘에서 Ownship과 Intruder는 그림 19 처럼 속도가 일정하다고 가정 하에 2차원과 3차원의 벡터 모델을 이용하여 예상 위치를 계산 한다. 예상위치에서 충돌을 감지하고 충돌 회피 기동의 Bands는 TCAS RA 로직의 수학적 기법은 통해 추출한다.[10] 이 알고리즘의 최종적인 결과물은 Bands의 회피기동을 표현하는 것과 음성 알람으로 구성되어 있다.

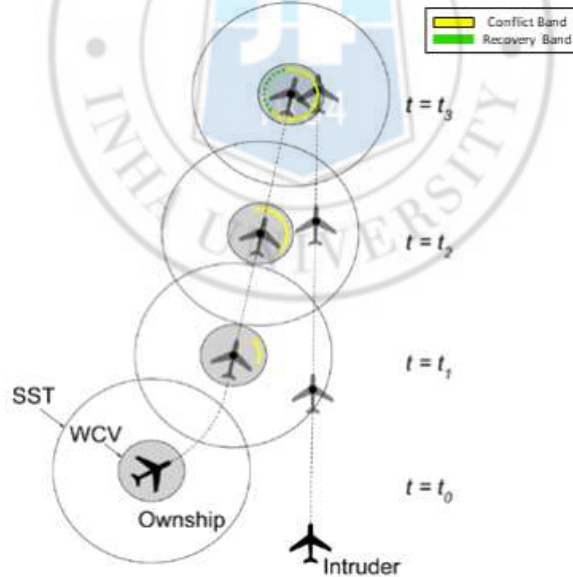


그림 20 DAIDALUS의 Bands 예시

DAIDALUS에서의 Bands는 Conflict Band와 Recovery Band로 구성되어 있으며, 그림 20과 같다. Ownship의 Self-Separation Threshold (SST)의 영역 안으로 Intruder가 진입할 때를 Preventive Bands로 칭하며 Ownship은 Conflict Bands를 계산하게 된다. 계속 충돌이 해소가 되지 않고 진행이 되면 Corrective Bands로 단계가 넘어간다. 또한 WCV의 영역 안으로 진입하게 되면 Well Clear Violation 단계로 Recovery Band가 구성되어 충돌을 회피하는 기동을 시작하게 된다.

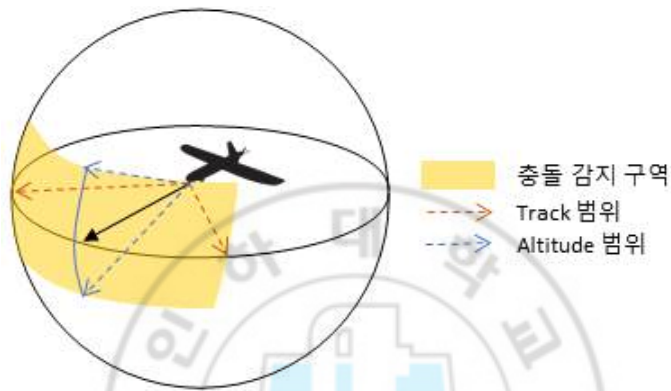


그림 21 DAIDALUS Band의 충돌 감지 구역

Bands는 고도(Altitude), 수직속도(Vertical Speed Resolution), 수평속도(Ground Speed Resolution), 방위각(Track)의 4가지 Band Resolution으로 구성된다. 그림 21,22과 같이 매 시간마다 충돌 감지 구역을 계산하고 이 구역을 벗어 날 수 있는 방법으로 4가지 종류의 기동 범위를 제공한다. 또한 범위를 제공하면서 선호 방향은 현재 항공기의 상태에 따라 정해진다. 항공기의 상태에서 가장 변화가 적은 방향을 제시하게 된다.

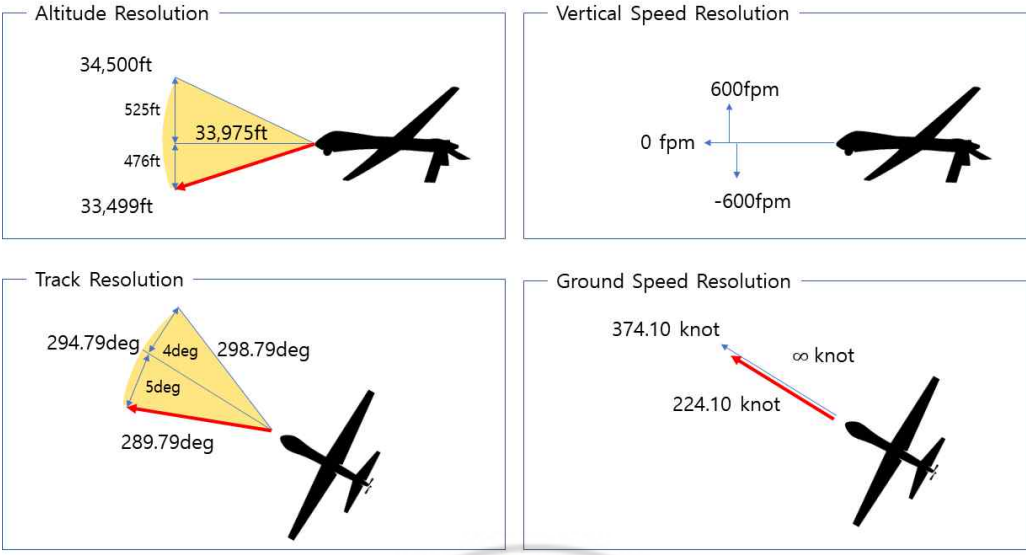
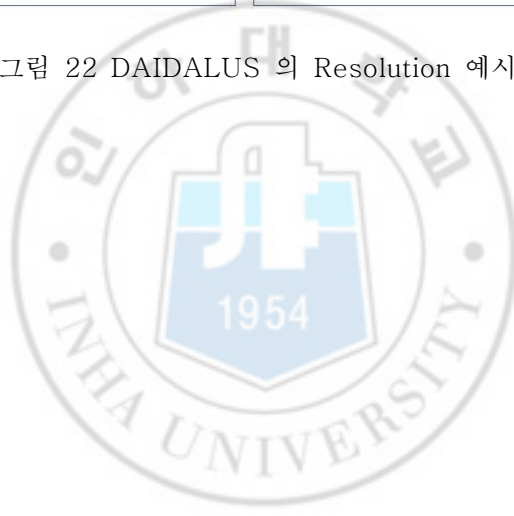


그림 22 DAIDALUS 의 Resolution 예시



3.3. 배속 시뮬레이션 (Fast-Time-Simulation)

배속 시뮬레이션(Fast Time Simulation, FTS)은 Eurocontrol에서 제공하는 BADA(Base of Aircraft Data)를 이용하여 주어진 경로점을 추종하는 단순 항공기 궤적 생성 모델로 구성되어 있다.[11]

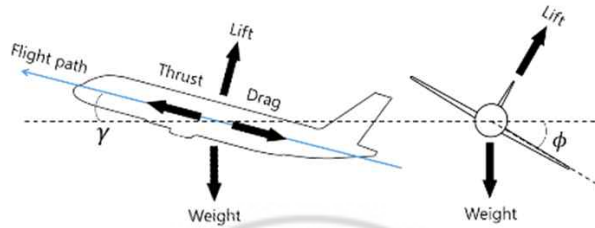


그림 23 항공기 운동 모델

그림 23과 같이 가로 운동과 세로 운동으로 나누어 구성하고, 서로 연동되어 있는 비행 경로각(γ)과 뱅크 각(ϕ)을 고려하였다. 운동 모델에 관련된 과정은 다음 그림 24 과정과 같다.

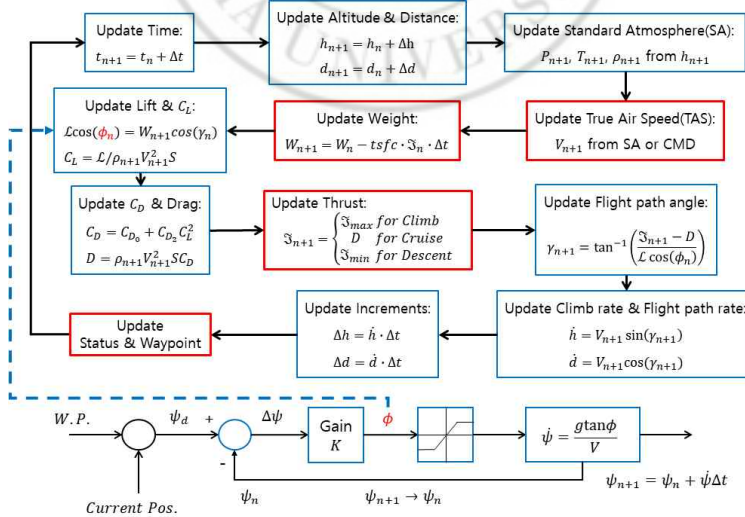


그림 24 항공기 가로 운동과 세로 운동 모델 과정

단순 운동 모델은 항공기가 주어진 경로점의 고도와 순항 속도에 도달하게 되면 그 상태를 유지하도록 구성되어 있으며, 한 항공기마다 배속 시뮬레이션을 수행하도록 구성되어 있다. 항공기 운동 모델에 대한 매 시간 마다 다른 항공기와의 충돌 감지 계산을 운용할 수 있어야 하며, 충돌을 감지하게 되면 운용항공기의 기동 변화를 주어질 수 있도록 재구성이 필요하였다.

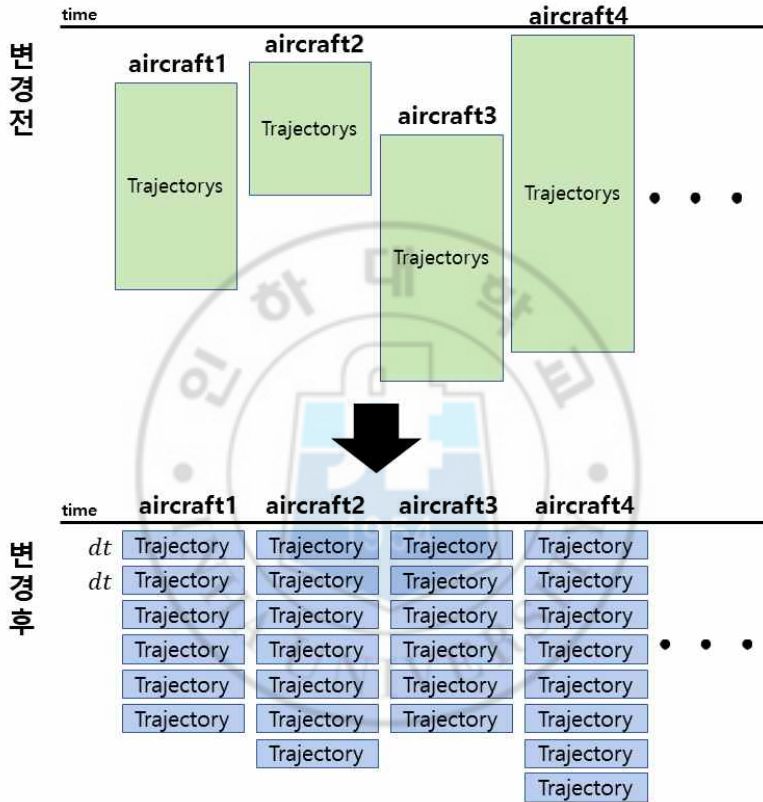


그림 25 변경된 항공기 운동 모델 구조 형태

먼저 그림 25와 같이 항공기 모델의 구조 형태를 변경하였다. 변경 전의 운동 모델에서는 한 항공기에 대한 모든 궤적을 수행한 후 다음 항공기로 넘어가는 구조로 구성되어 있으며, 동시간대에 맞춰있지 않고 항공기 별로 독립적으로 수행되어 있는 것을 볼 수 있다. 이에 모든 항공기의 시간을 동일하게 맞추어 시간 단위 동안 항공기별로 궤적을 생성하도록 재구성 하였다.

이 과정에서 매 시간 단위마다 항공기의 상태와 위치, 속도를 구하여 Well Clear를 계산하는 충돌 감지 할 수 있는 알고리즘을 적용시킬 수 있었다. 또한 항공기의 상태 변수로 DAIDALUS 알고리즘을 수행하여 충돌 회피 기동을 연산하도록 운동 모델 과정을 재구성하였다.

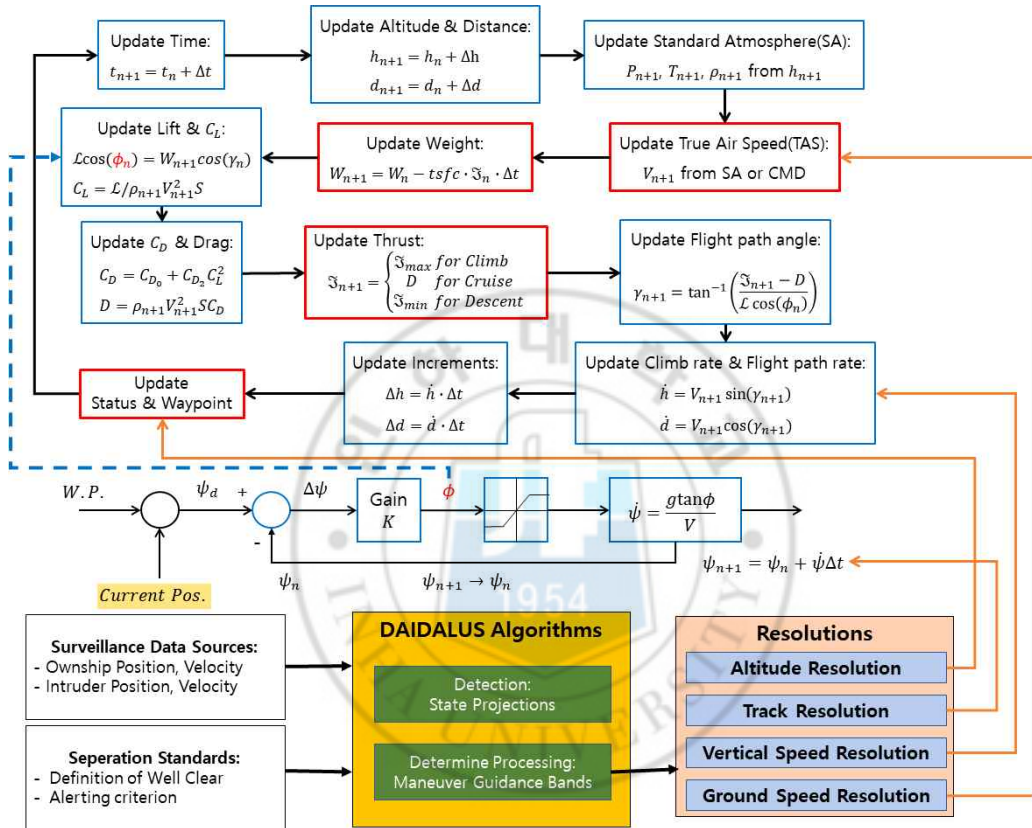


그림 26 재구성된 궤적 생성 모델

그림 26과 같이 DAIDALUS 알고리즘으로 계산된 회피 Resolution은 고도 기동, 방위각 기동, 수직 속도 기동, 수평 속도 기동으로 4가지 종류를 구할 수 있으며, 각각은 궤적 생성 모델에서 기동을 변경할 수 있게 변경하였다.

고도 기동의 경우, 추종 경로점의 고도를 변경함으로써 기동 변경을 할 수 있도록 구성하였고, 다른 종류의 기동은 각 부분에서 기동을 변경할 수 있도록 구성하였다.

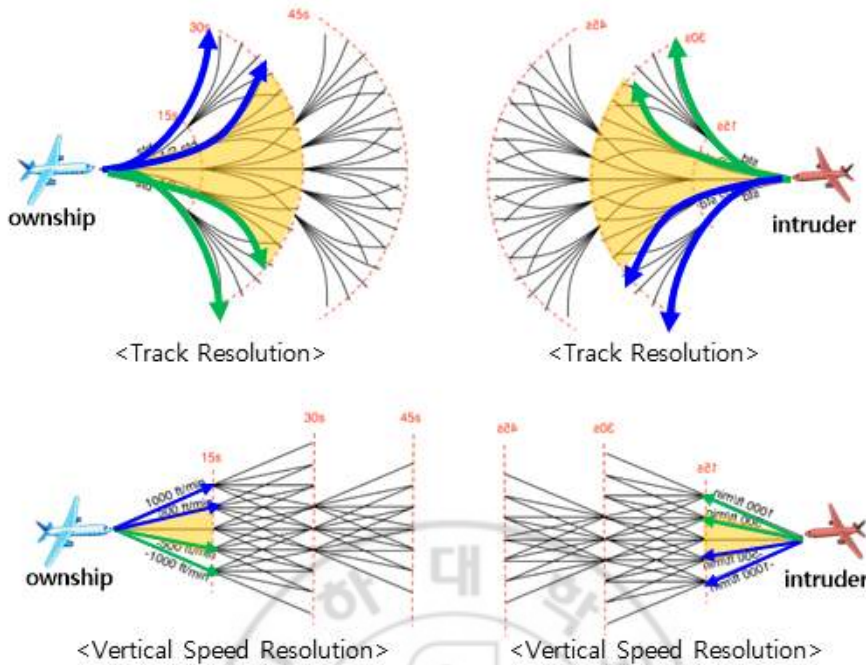


그림 27 DAIDALUS의 충돌 구역과 회피 가능 기동

침입항공기에 대해 LoWC가 발생하게 된다면 DAIDALUS 알고리즘을 통해 항공기의 충돌 범위를 계산하고, 충돌 범위를 벗어나는 Resolution에서 기동 범위를 변경하며 배속 시뮬레이션을 수행한다. 고도 기동은 100ft 간격으로, 방위각은 1deg 간격, 수평속도는 5knots 간격, 수직 속도는 100fpm 간격으로 시뮬레이션을 재수행하여 충돌이 일어나지 않는 기동을 찾게 된다. 회피 기동을 산출하는 방법은 최적화 방법인 비용 함수(Cost Function)를 이용하였다. 비용 함수가 가장 작은 값을 산정하는 기동을 선택하게 된다.

회피 기동에는 단순 충돌을 회피 하는 기동에서부터 본 궤적으로 돌아오는 시점에 대한 기동 모두 시뮬레이션을 수행한다. 회피 기동을 지속해야 하는 시간과 어떤 경로점

으로 돌아올 수 있는지에 대해서도 수행해야 한다.

현 무인항공기는 법규에 의해 원격조종으로 비행이 가능하며, 이는 다음 그림 28와 같다. 충돌을 회피 할 수 있는 기동을 원격 조종사의 개입으로 수행하고 본 궤적으로 돌아오는 시점 또한 조종사에 의존되어 있다. 이에 본 논문에서 제시하는 알고리즘은 CPA 지점 이후 본 궤적으로 돌아오는 기동을 배속시물레이션을 모두 수행한 이후 비용 함수가 최소가 되는 기동을 선택하여 회피 기동을 할 수 있도록 구성하였다.

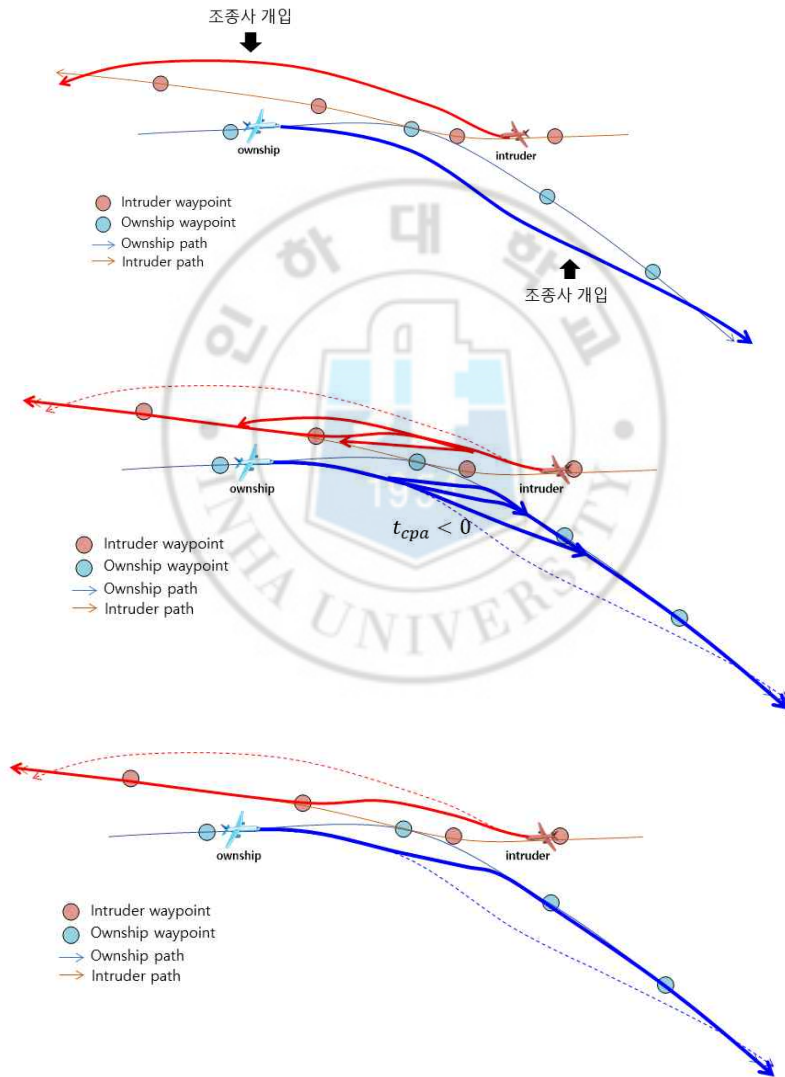


그림 28 최소 비용 함수 기동을 선택 시 기동 상황

4. 비용 함수의 설정

비용 함수(Cost Function)는 최적 충돌 회피 기동을 선택을 할 때, 선택의 기준이 되며 중요한 인자가 된다. 하지만 비용함수의 인자를 결정하는 것은 전체 시뮬레이션의 성능을 결정하는 것으로 신중하고 정확해야 한다. 비용 함수는 가중치와 인자의 곱으로 나타내며, 비용함수가 낮을수록 최적 기동으로 판단한다. 최적기동을 판단하는 규칙은 다음과 같다.[12]

- 최적 기동이 없을 때는 첫 번째 기동 후보가 최적 기동이 된다.
- 기존 최적 기동이 충돌을 발생하고, 새로운 기동이 충돌을 발생하지 않는다면, 새로운 기동이 최적 기동이 된다.
- 기존 최적 기동이 충돌을 발생하지 않고 새로운 기동도 충돌을 발생하지 않는다면, 둘 중 비용함수가 최소가 되는 기동을 최적 기동으로 선택한다.

본 논문에서 사용한 비용 함수는 다음 식으로 나타낸다.

$$\begin{aligned} \text{Cost} &= \text{cost}_{rank} + \text{cost}_{holding} + \text{cost}_{wc} \\ &= \sum_i C_i \\ &= \sum_i W_i F_i \end{aligned}$$

- W_i = 비용 함수의 가중치
- F_i = 비용 함수의 인자

개별적인 비용 함수의 인자와 가중치의 곱을 전체 합한 것을 전체 비용 함수라고 정하며, 가중치는 비용 함수의 영향력에 따라 값이 정해진다. 본 논문에서 가중치와 인자의 종류는 다음과 같이 정하였다.

4.1. 기동 종류 비용

기동의 종류에 대한 비용은 우 선회, 좌 선회, 수직 속도 증가와 감소, 수평 속도 증가와 감소에 따라 비용 함수의 인자로 정하며, 가중치(W_{rank})는 0.1로 설정하였다.

- $\text{cost}_{rank} = W_{rank} \cdot F_{rank}$
- $W_{rank} = 0.1$
- $F_{rank} = k$
- $k = \begin{cases} 0 & \text{if right turn} \\ 1 & \text{if left turn} \\ 2 & \text{if increased vertical speed} \\ 3 & \text{if reduced vertical speed} \\ 4 & \text{if slow down} \\ 5 & \text{if speed up} \end{cases}$

기동 종류 비용에 대한 인자(F_{rank})가 작은 기동종류가 적은 비용 함수를 발생하도록 설정하였으며, 이는 NASA에서 개발한 GRACE 시스템의 비용함수를 따랐다.[12]

4.2. 기동 체류 시간 비용

기동 체류 시간은 기동이 시작 시간부터 기동이 끝난 시간까지를 의미한다. 기동 체류 시간이 길수록 비용은 증가하게 되고, 체류 시간이 짧을수록 최적 기동이 된다. 체류시간에 대한 가중치(W_{ht})는 0.5로 설정하였다.

- $\text{cost}_{holding\ time} = W_{ht} \cdot F_{ht}$
- $W_{ht} = 0.5$
- $F_{ht} = \frac{T_{end} - T_{start}}{T_{lookahead}}$

4.3. 충돌 위험 비용

충돌 위험 비용은 비용 함수에서 가장 중요한 부분을 차지하도록 구성하였으며, WCS에서 LoWC가 감지되는 기동은 절대 선택할 수 없도록 구성하였으며, 다음과 같다.

- $cost_{wcs} = W_{wcs} \cdot F_{wcs}$
- $W_{wcs} = 1$
- $F_{wcs} = \sum_i num_{WCS_i} \cdot WCS_i \cdot w_i$
- $num_{WCS_i} = WCS_i$ 가 발생하는 횟수
- $WCS_i = \begin{cases} 0 & \text{if } otherwise \\ 1 & \text{if } preventive \\ 2 & \text{if } corrective \\ 3 & \text{if } warning \\ 4 & \text{if } lowc \end{cases}$
- $w_i = WCS_i$ 에 대한 가중치

$$w_i = \begin{cases} 1 & \text{if } otherwise \\ T_{end} - T_{start} & \text{if } lowc \end{cases}$$

WCS는 앞서 Well Clear의 기준을 말하는 표에서 보여준다. WCS에 대한 가중치는 LoWC가 발생할 때는 체류시간으로 설정한다.

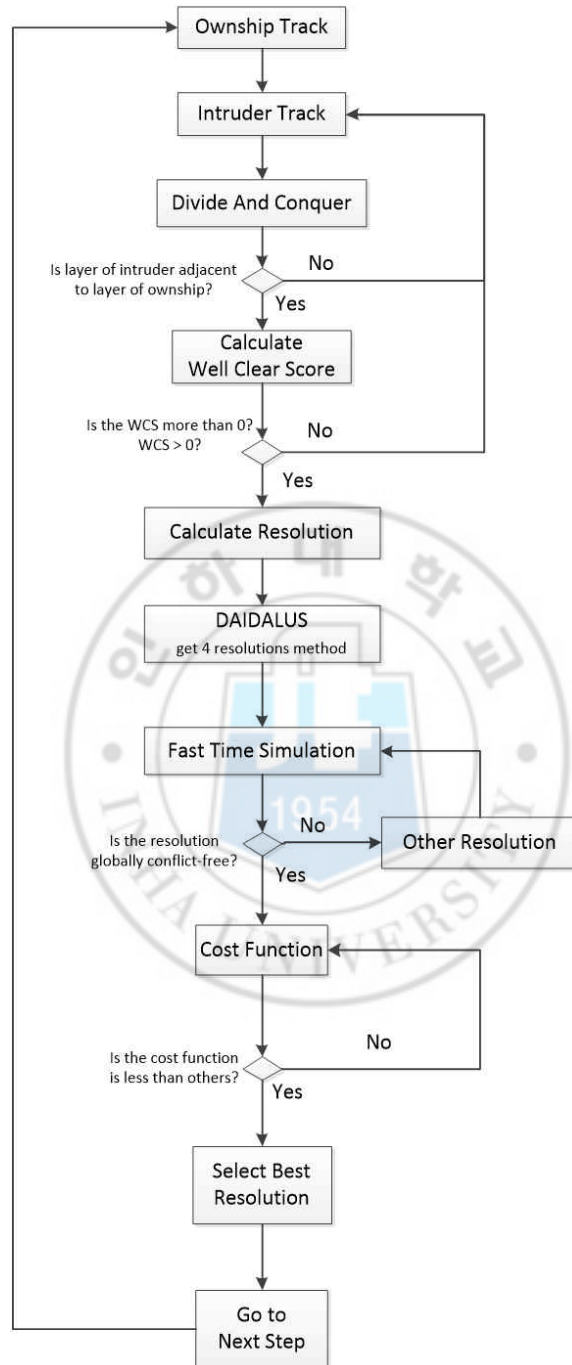


그림 29 전체 알고리즘 순서도

5. 충돌 회피 검증

본 알고리즘의 순서도는 다음 그림 29과 같다. WCS가 0보다 큰 Preventive 상태에 접근하게 되면 DAIDALUS 알고리즘에서 충돌 범위를 알려주고 그 이후 4가지의 회피 기동이 주어진다. 후보 회피 기동 중에 충돌이 재 발생하지 않고, 비용 함수가 가장 작은 값의 기동을 선택하게 된다. 하지만, 회피 기동을 배속 시뮬레이션을 수행하였을 때 충돌이 일어날 것이라고 판단이 된다면 고도 기동은 100ft 간격으로, 방위각은 1deg 간격, 수평속도는 5knots 간격, 수직 속도는 100fpm 간격으로 시뮬레이션을 재수행하여 충돌이 일어나지 않는 기동을 찾게 된다.

또한 충돌 회피 기동을 진행하는 시간과 본 궤적으로 회복 하는 시점 또한 배속 시뮬레이션을 통해 최적 시간을 찾을 수 있다. 가장 첫 번째 기준은 CPA에 도달한 시간으로, 두 항공기가 가장 가깝게 되는 시점에서 회복을 시작한다. 이 기준으로 기동을 진행하였을 때, 충돌이 일어나지 않는다면 복귀 시간을 5초 간격으로 줄이면서 비용함수를 살펴보고, 충돌이 일어난다면 CPA 시간보다 5초 간격씩 늘어가며 최적 경로를 찾게 된다.

검증을 위해 1:1 항공기의 조우 상황에 대해 진행을 하였다. 첫 번째 시나리오는 두 항공기가 모두 순항 조건일 때 마주보는 상황을 모사하였다. 두 항공기 모두 경로점을 입력 받아 배속 시뮬레이션을 진행하였다.

5.1. Case 1: 마주보는 순항 시나리오

Ownship과 Intruder는 표 4의 시나리오 같이 4개의 경로점으로 구성되어 있다. 경로점의 변수는 위치 정보인 위도, 경도, 고도와 속도, 방위각으로 구성되어 있다. 이러한 경로점은 실제 기종 B788 항공기의 ADS-B로 받은 비행 항로의 일부분이다. 시나리오의 본 궤적은 그림 31과 같이 마주보면서 순항하는 시나리오임을 알 수 있다.

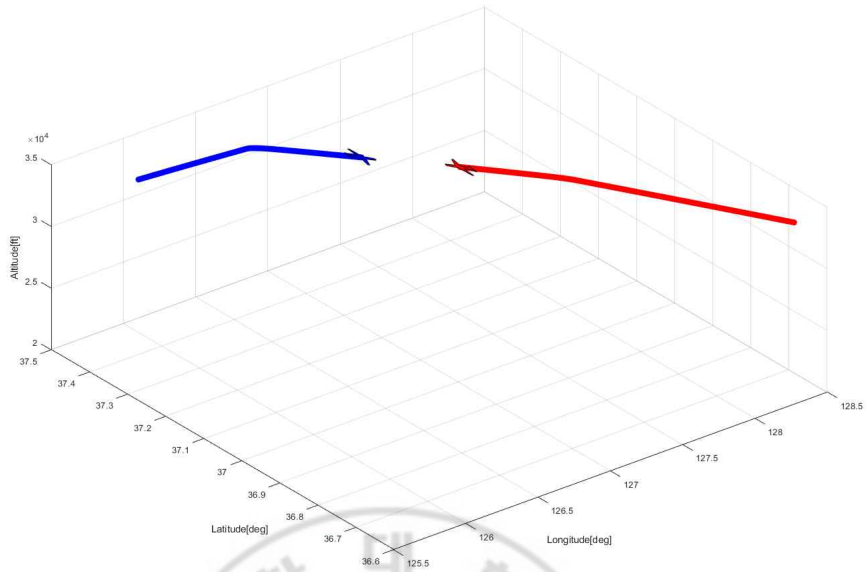


그림 30 마주보는 순항 시나리오 궤적

표 4 마주보는 순항 시나리오

Ownship						
State	Direction	Waypoint				
Cruise	West	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.62521	128.3375	33975	306	392
		36.98726	127.6705	33975	296	376
		37.38015	126.5976	33975	295	371
		37.40588	125.8568	33975	270	349

Intruder						
State	Direction	Waypoint				
2Cruise	East	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		37.40588	125.8568	33975	90	349
		37.38015	126.5976	33975	115	371
		36.98726	127.6705	33975	116	376
		36.62521	128.3375	33975	126	392

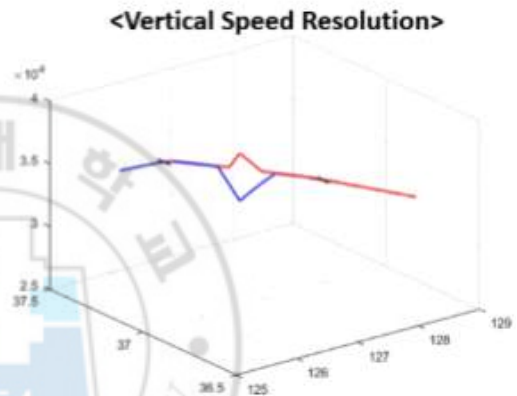
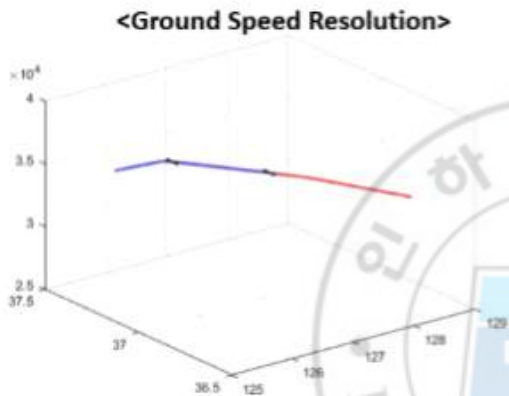
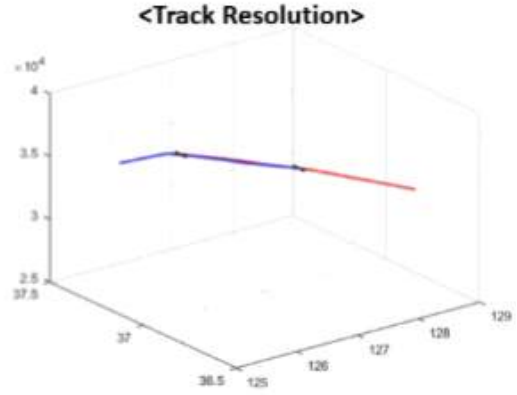
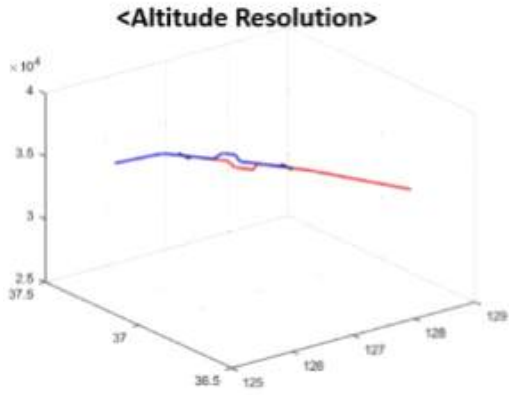


그림 31 4가지 회피 기동

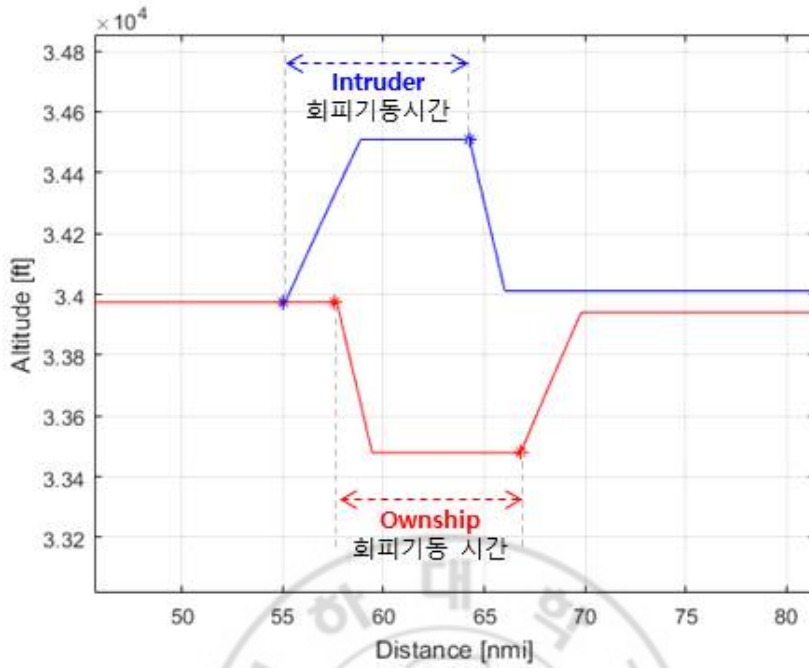


그림 32 Altitude Resolution

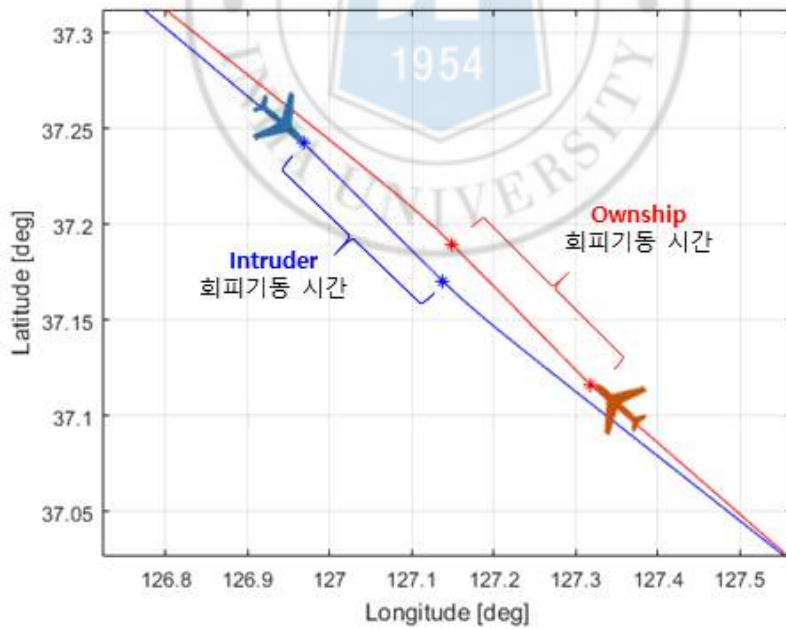


그림 33 Track Resolution

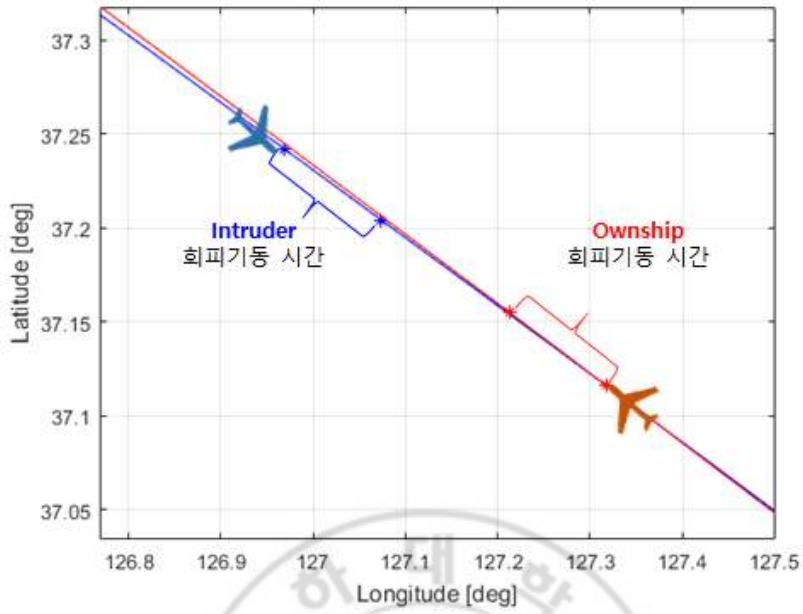


그림 34 Ground Speed Resolution

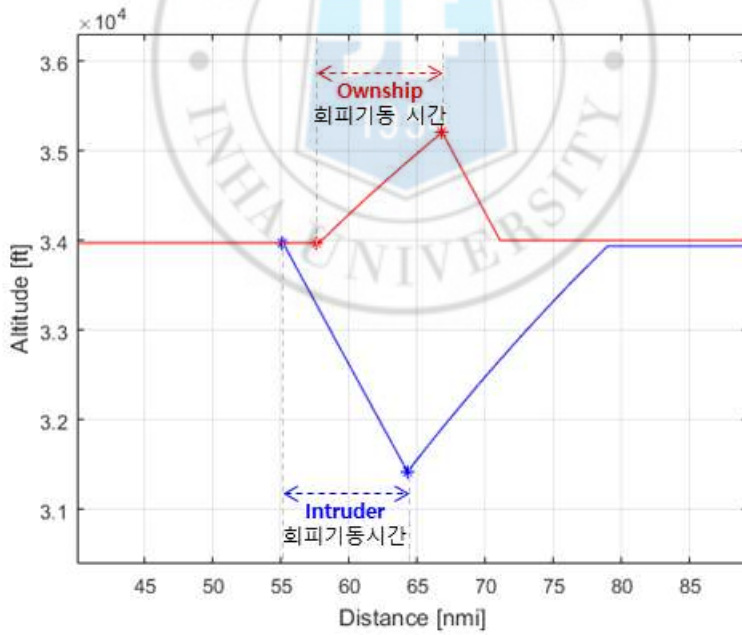


그림 35 Vertical Speed Resolution

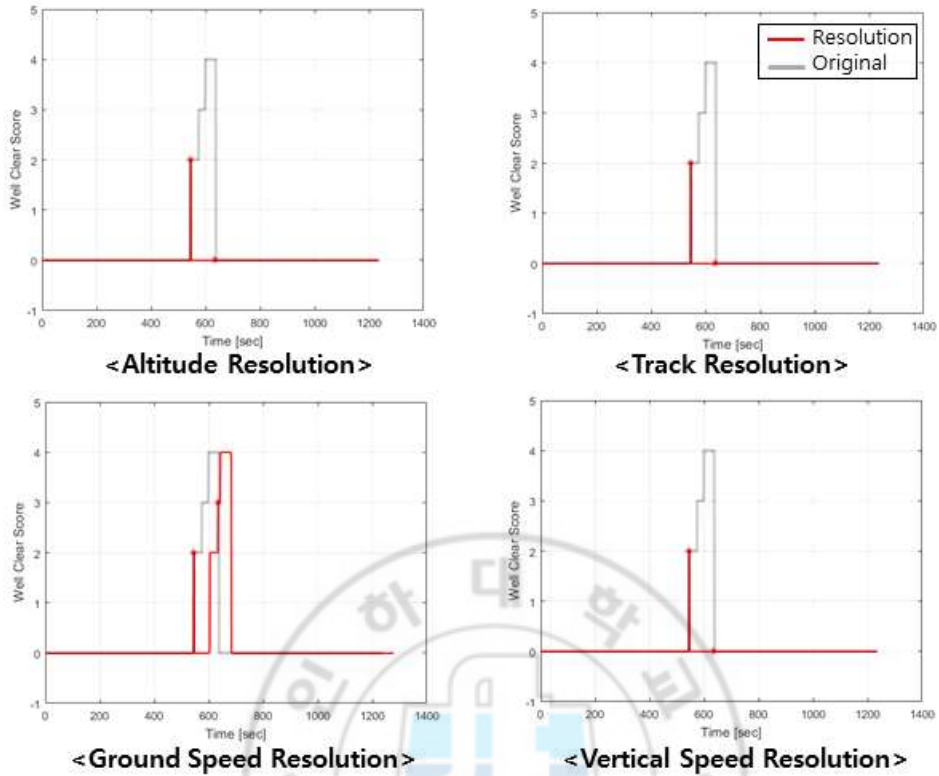


그림 36 기동에 따른 WCS

그림 32과 그림 33은 회피 기동을 한 후, CPA 시점에서 회복 기동을 한 궤적이다. 그림 34은 4가지 기동에 따른 WCS를 비교한 것이다. 본 궤적으로 진행하였다면, 회색 그래프처럼 LoWC 상태로 진행되지만 회피 기동에 성공하고, 회복 기동을 진행한 결과 WCS를 해소한 것을 볼 수 있다. 다만, 수평 속도 기동을 살펴보면, WCS가 해소되었다 다시 진행되는 것을 볼 수 있다. 이는 비용 함수를 통해 선택되지 않도록 설계하였다.

표 5 Case 1 시나리오에 대한 비용 함수

Resolution	Cost
Altitude	8.70976
Track	8.41112
Vertical Speed	8.61112
Ground Speed	8.56538e5

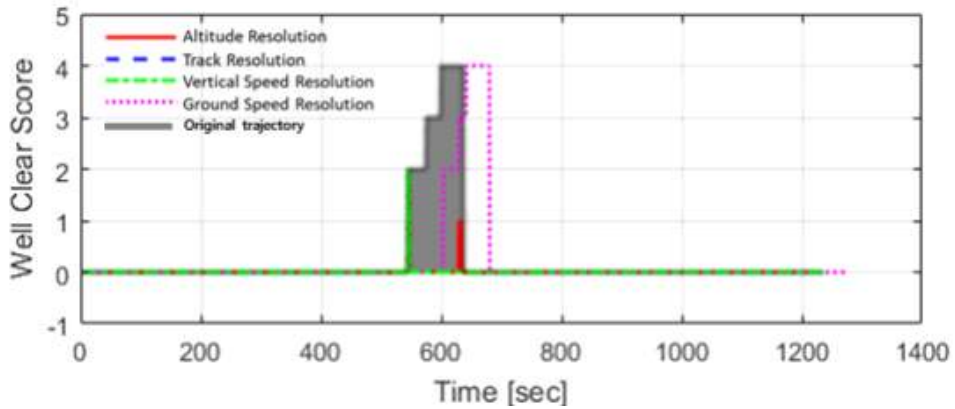


그림 37 CPA 지점 5초 전 회복 기동한 WCS

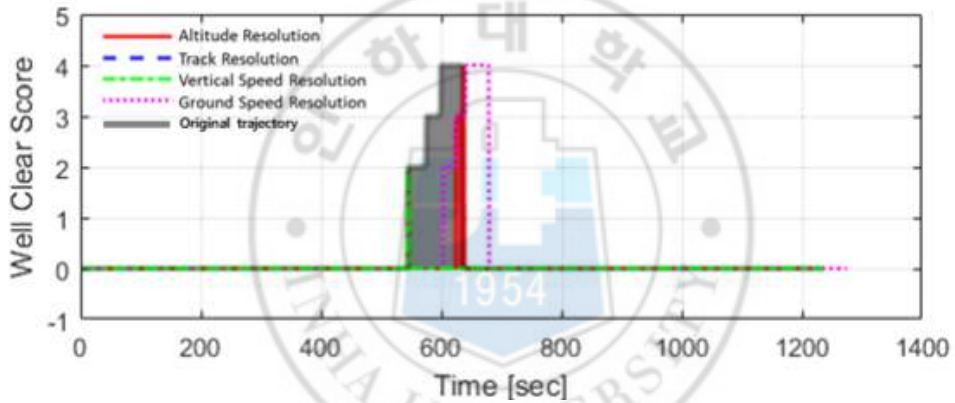


그림 38 CPA 지점 10초 전 회복 기동한 WCS

CPA지점에서 본 궤적으로 회복 기동을 진행하였을 때, 충돌이 발생하지 않는 것을 확인할 수 있었다. 그 이후에 본 궤적과 가장 가깝게 진행 되는 지점을 찾는 것을 우선시 하므로 CPA 지점보다 5초전에 회복 기동을 한 기동과 10초 전에 회복 기동을 하였을 때, 그림 37, 38 을 보는 바와 같이 충돌위험이 일어나는 것을 볼 수 있고, 이는 CPA지점이 가장 안전하면서, 본 궤적과 가장 가까운 기동임을 알 수 있다.

5.2. Case2: 예각 횡단 순항 시나리오

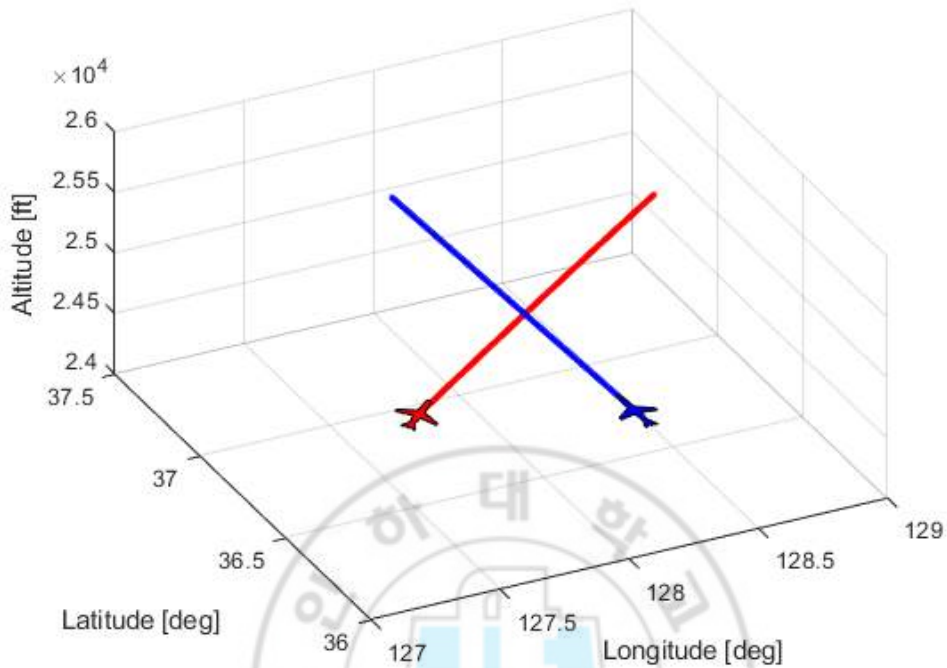


그림 39 예각 횡단 순항 시나리오 본 궤적

표 6 예각 횡단 순항 시나리오

Ownship						
State	Direction	Waypoint				
Cruise	West	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.35899	127.4211	25000	45	255
		37.25893	129.0416	25000	45	255

Intruder						
State	Direction	Waypoint				
Cruise	East	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.12763	128.1057	25000	0	240
		37.47199	128.0404	25000	0	240

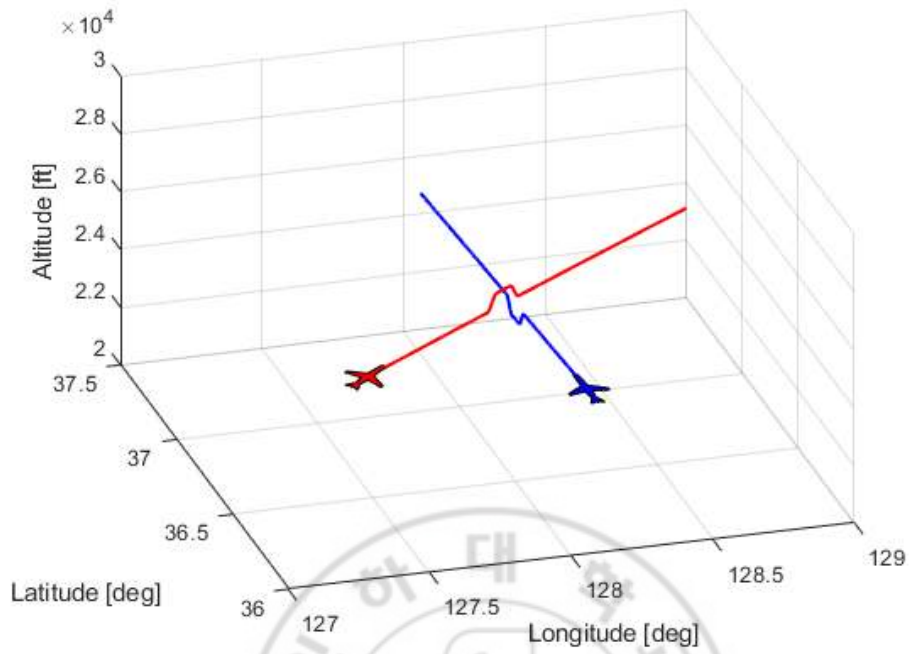


그림 40 Altitude Resolution

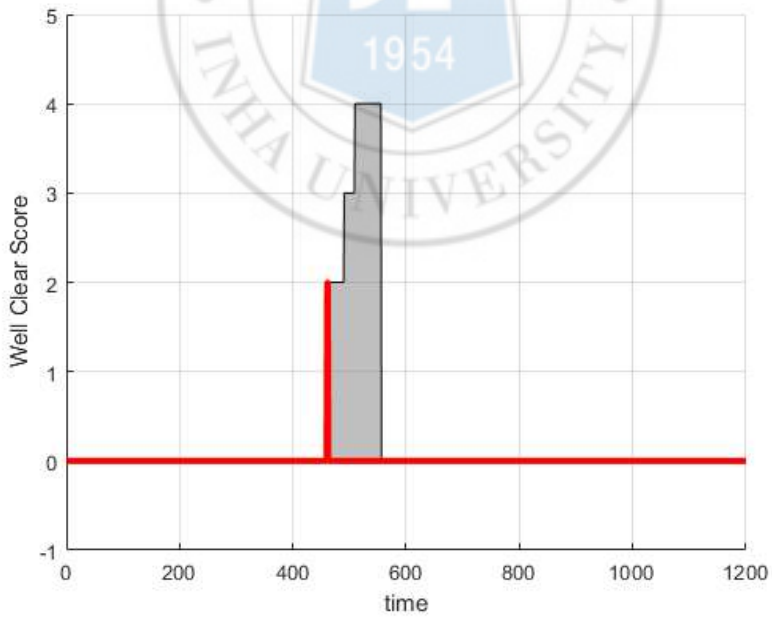


그림 41 Altitude Resolution WCS

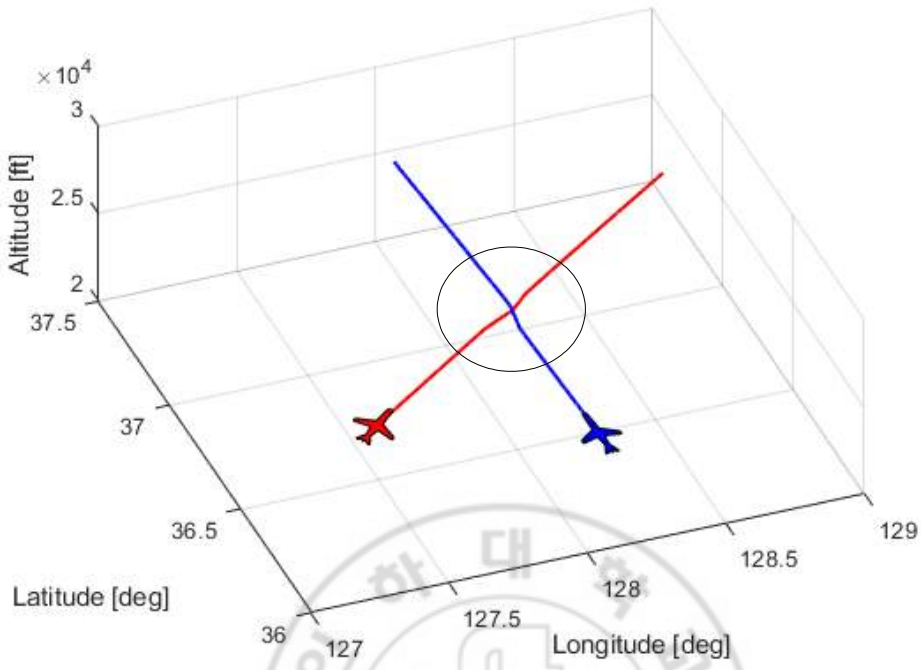


그림 42 Track Resolution

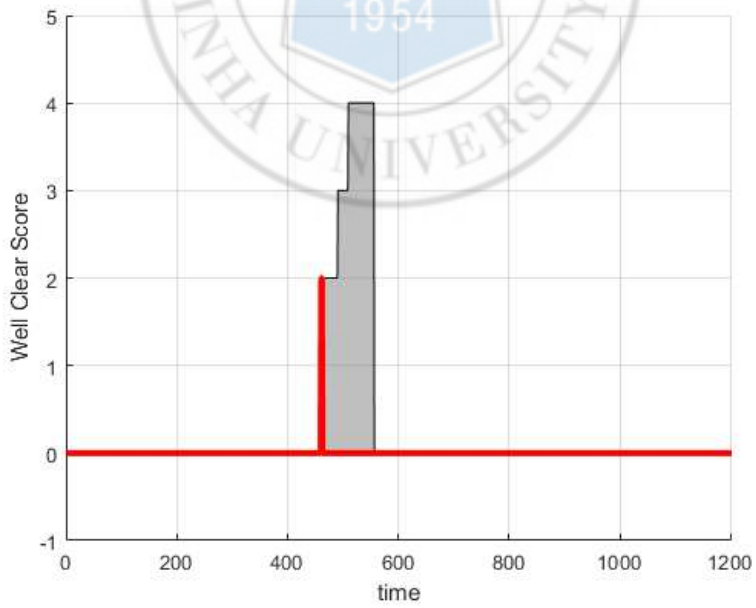


그림 43 Track Resolution WCS

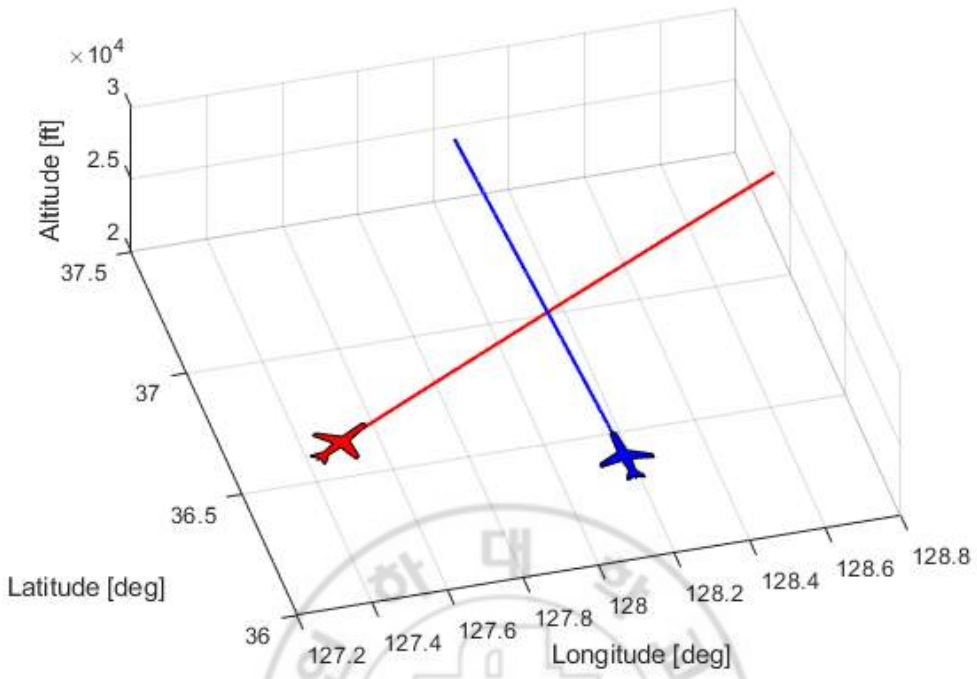


그림 44 Ground Speed Resolution

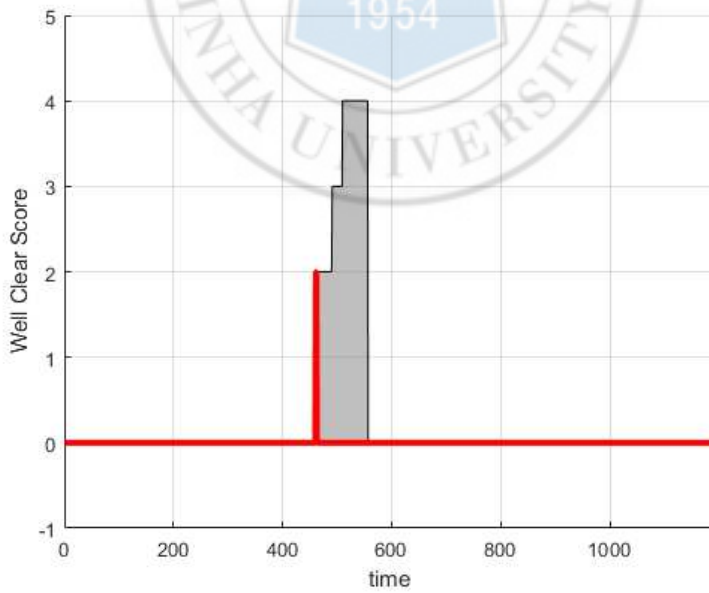


그림 45 Ground Speed Resolution WCS

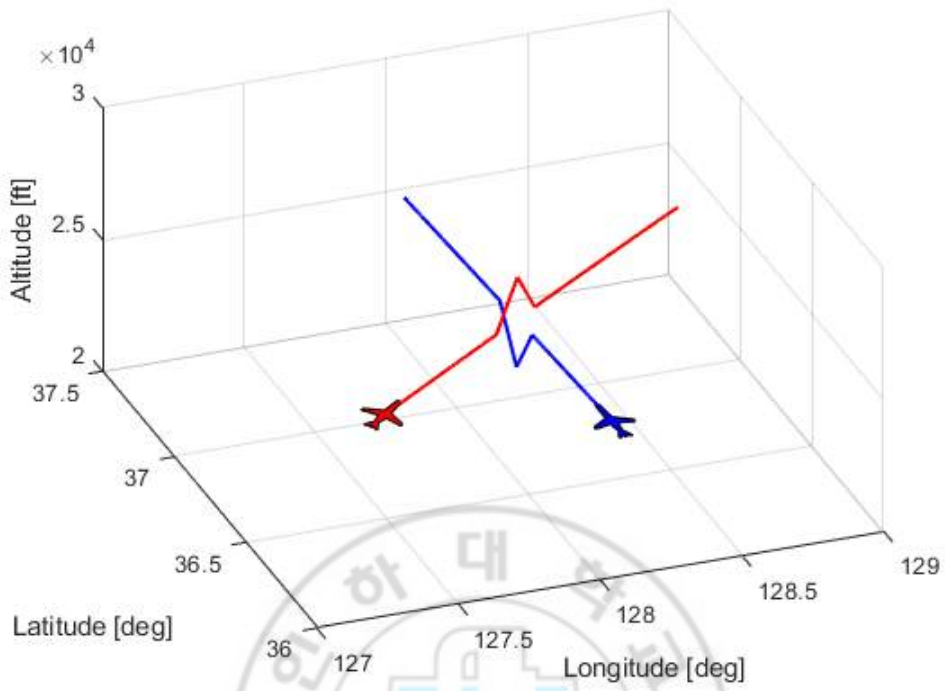


그림 46 Vertical Speed Resolution

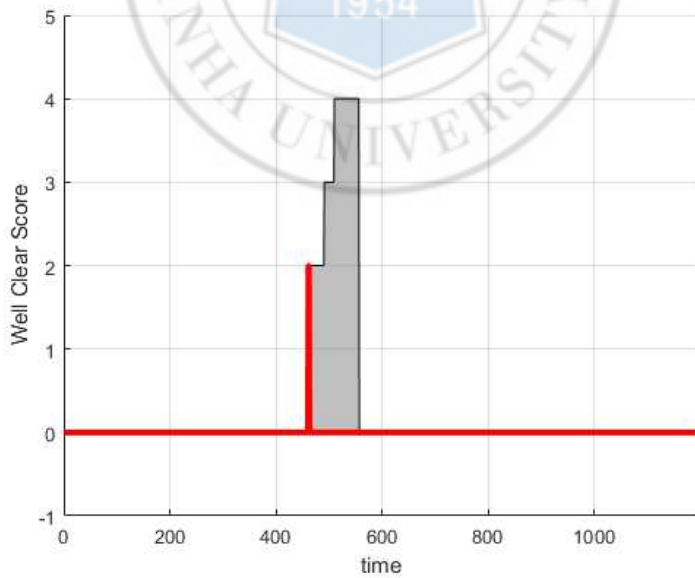


그림 47 Vertical Speed Resolution WCS

표 7 Case 2 시나리오에 대한 비용 함수

Resolution	Cost
Altitude	8.6223
Track	8.4250
Vertical Speed	8.6112
Ground Speed	8.6223

이 시나리오는 Ownship과 Intruder가 서로 횡단하며, 서로 다가오는 각도를 예각으로 지정하였다. 이 시나리오에서 회피 기동은 성공적으로 진행하였으며, WCS가 4가지 Resolution 모두 회피 기동이후 발생하지 않을 것을 확인 할 수 있다. 또한, 본 궤적으로 회복기동을 진행하였을 때는 CPA지점에서 회복 기동을 시작하며, 돌아오는 경로점을 기동 체류시간 후에 새로 생성하여 가장 안전하되, 본 궤적과 얼마나 떨어졌는지의 척도를 비용함수로 선택을 진행하였다.

이 시나리오에 대한 4가지 기동의 비용함수는 표 7과 같다. 충돌이 발생한 기동이 없어 그 차이는 작았지만, 그 중에서도 방위각(Track) 기동이 비용함수가 가장 낮음을 확인 할 수 있었다. 마지막으로 회피기동과 회복기동은 방위각 기동을 선택하였다.

선택 이후, 회피 기동을 진행하여 다음 그림 49과 같은 궤적을 수행하게 된다.

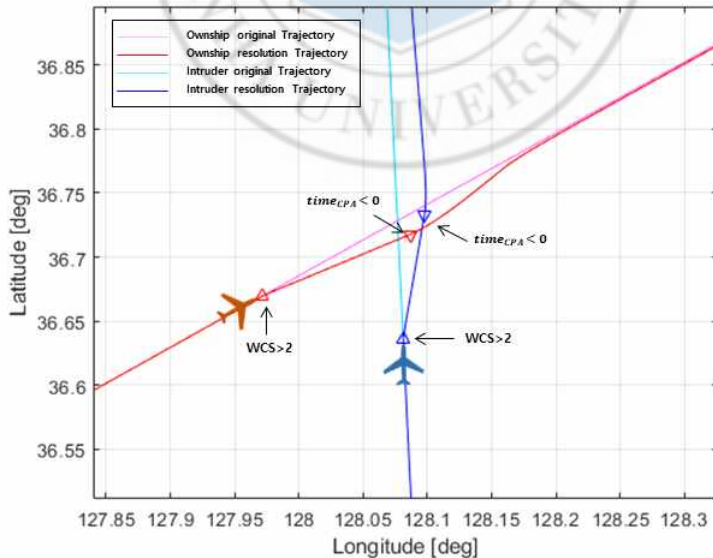


그림 48 최적 기동 선택 후 궤적

5.3. Case 3: 둔각 횡단 순항 시나리오

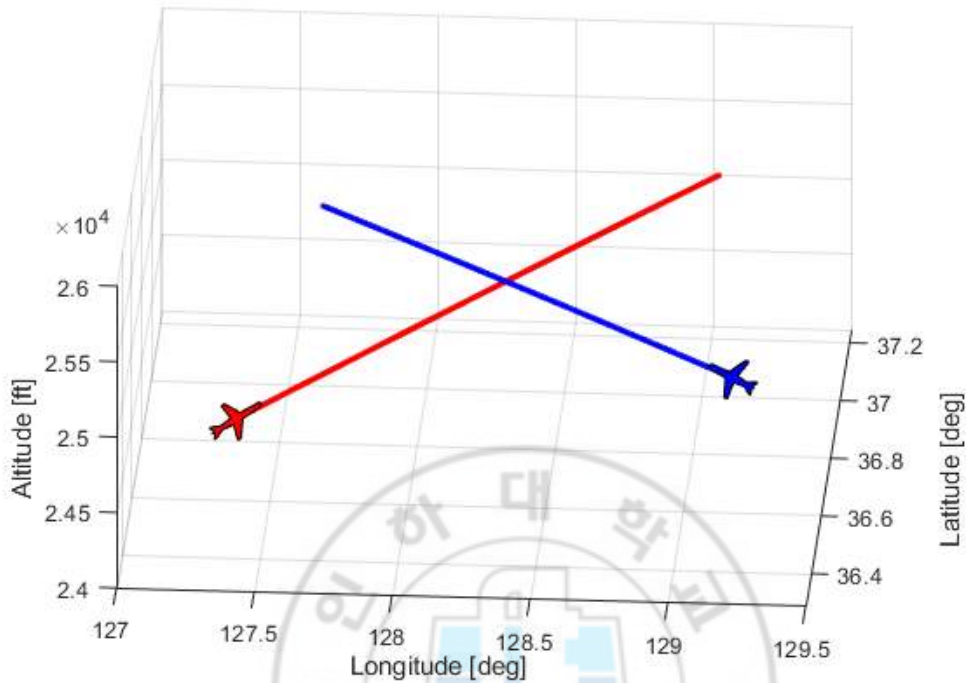


그림 49 둔각 횡단 순항 시나리오 본 궤적

표 8 둔각 횡단 순항 시나리오

Ownship						
State	Direction	Waypoint				
Cruise	West	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.358989	127.4211	25000	45	270
		37.25893	129.0416	25000	45	270

Intruder						
State	Direction	Waypoint				
Cruise	East	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.548891	129.1865	25000	315	240
		37.20762	127.292	25000	315	240

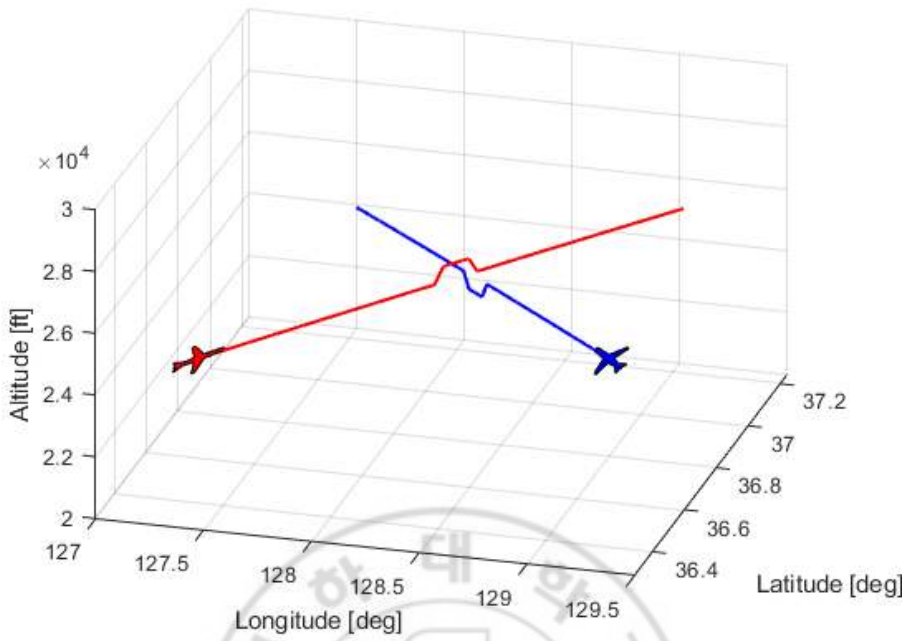


그림 50 Altitude Resolution

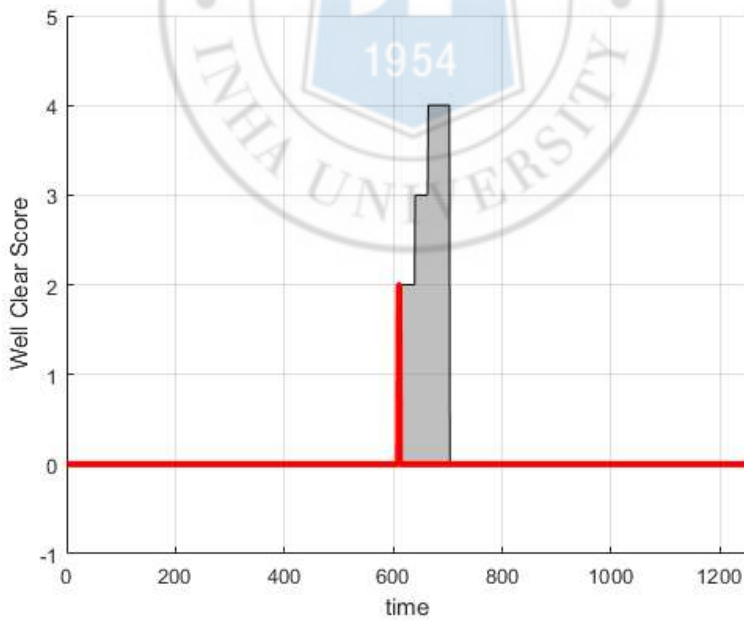


그림 51 Altitude Resolution WCS

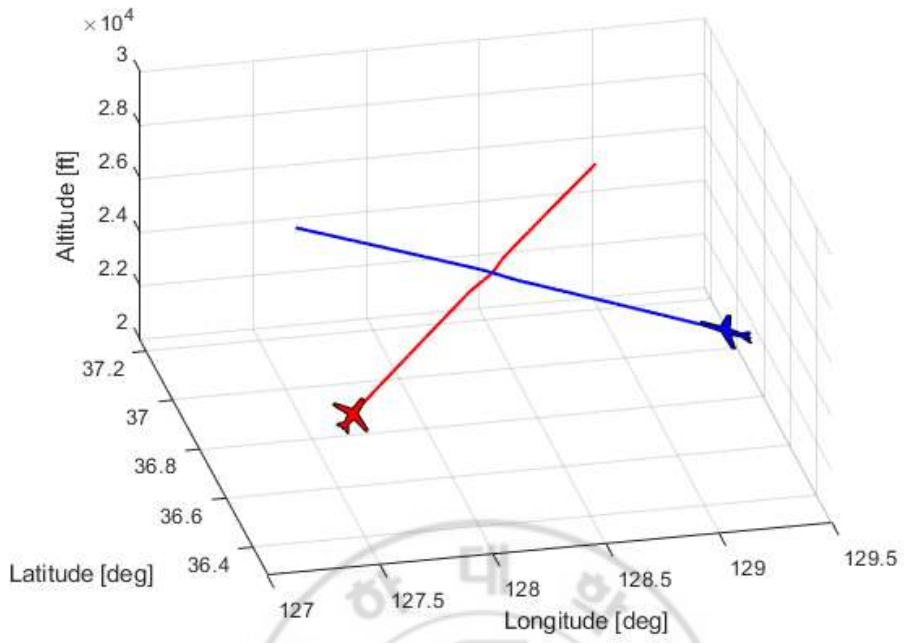


그림 52 Track Resolution

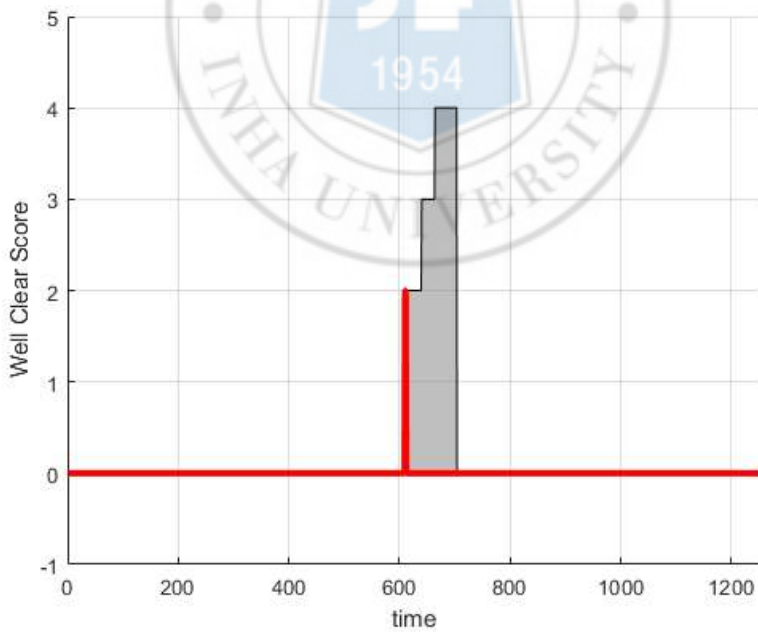


그림 53 Track Resolution WCS

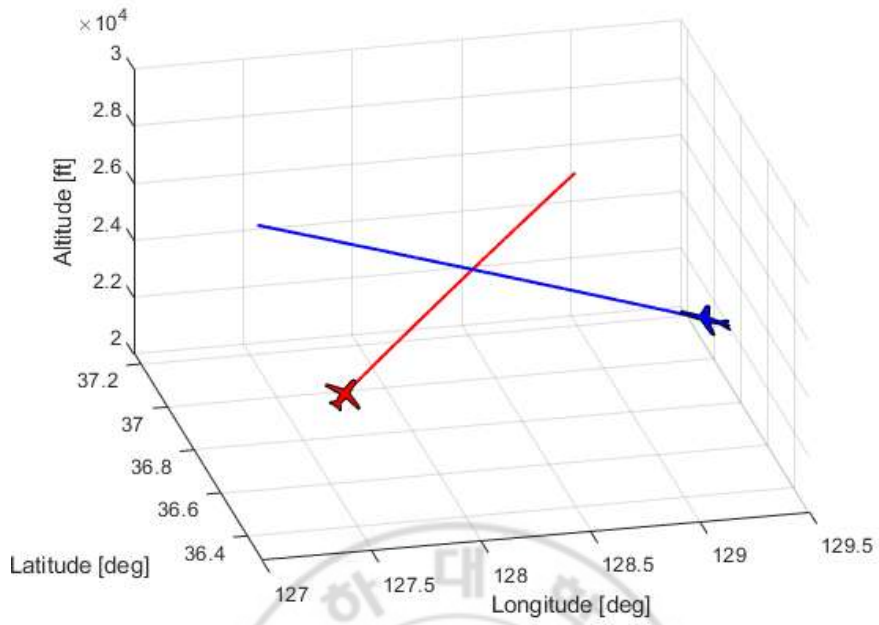


그림 54 Ground Speed Resolution

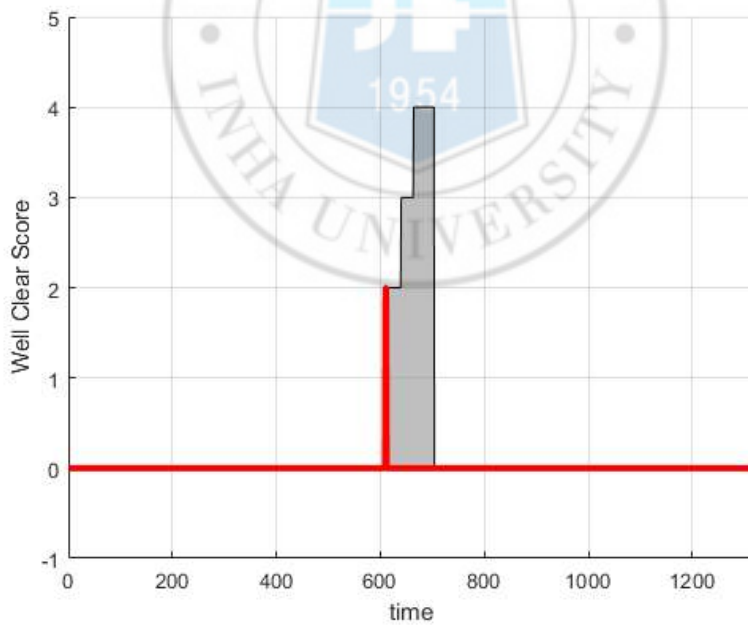


그림 55 Ground Speed Resolution WCS

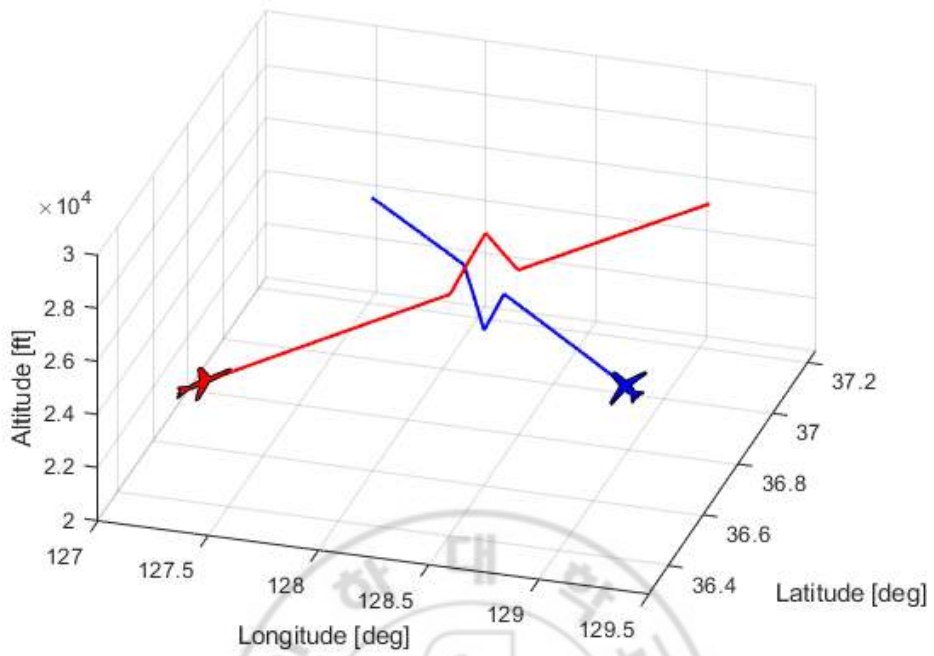


그림 56 Vertical Speed Resolution

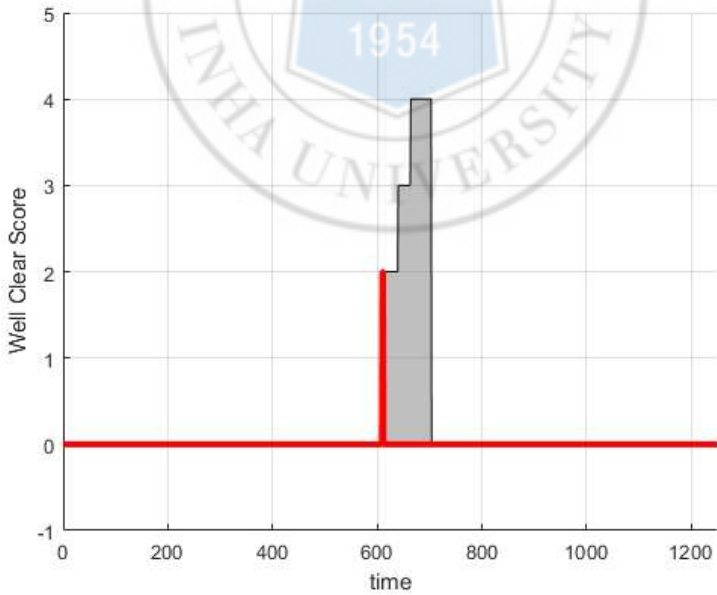


그림 57 Vertical Speed Resolution WCS

표 9 Case 3 시나리오에 대한 비용 함수

Resolution	Cost
Altitude	8.6333
Track	8.4334
Vertical Speed	8.6333
Ground Speed	9.0250

이 시나리오는 Ownship과 Intruder가 서로 순항하면서 조우하며, 다가오는 각도를 둔각으로 지정하였다. 이 시나리오에서 예비 회피 기동은 모두 성공적으로 진행하였으며, WCS가 4가지 Resolution 모두 회피 기동이후 발생하지 않을 것을 확인 할 수 있다. 그림 59를 보면 본 궤적으로 회복기동을 진행하였을 때는 CPA지점에서 회복 기동을 시작하며, 돌아오는 경로점은 검은색 점으로 기동 체류시간 후에 새로 생성하여 가장 안전하되, 본 궤적과 얼마나 떨어졌는지의 척도를 비용함수로 선택을 진행하였다.

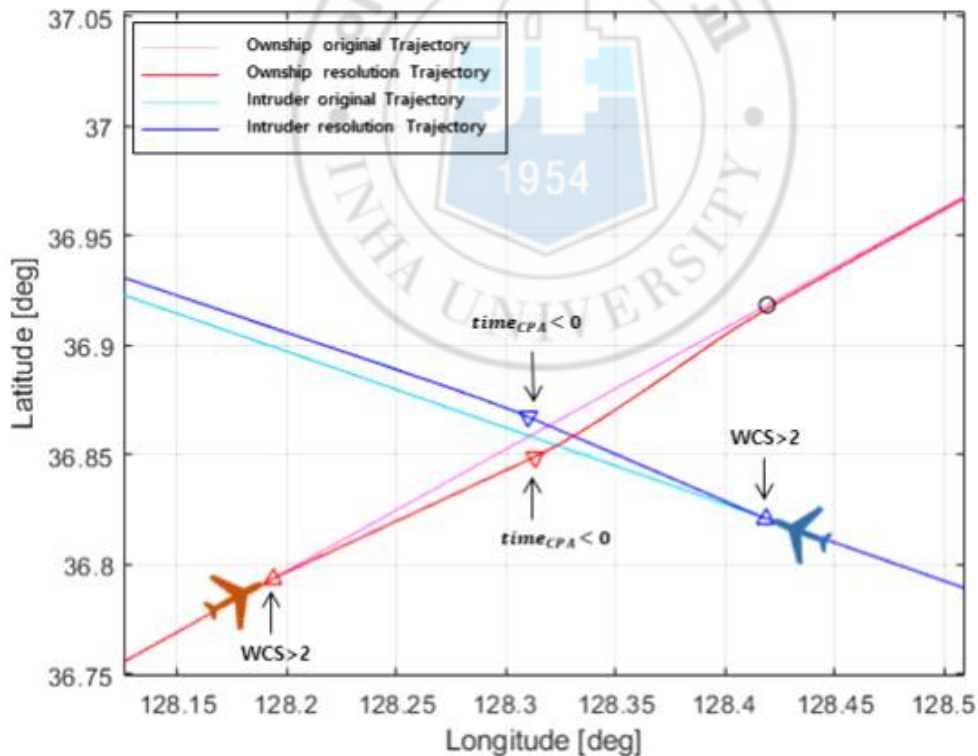


그림 58 회피 기동 중 선택 후 궤적

5.4. Case 4: 순항 상승 시나리오

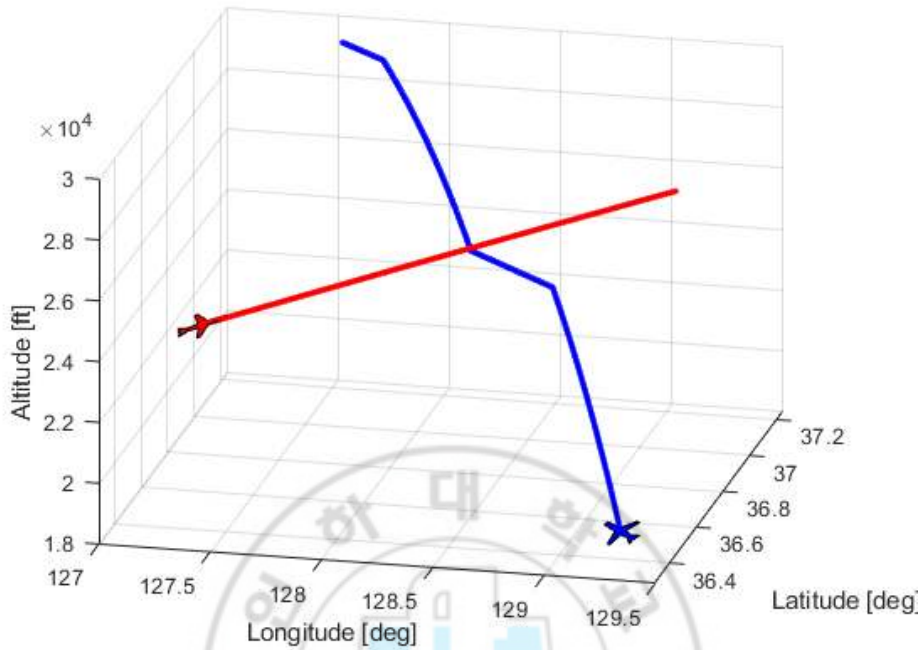


그림 59 순항 상승 시나리오 본 궤적

표 10 순항 상승 시나리오

Ownship						
State	Direction	Waypoint				
Cruise	West	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.35899	127.42112	25000	45	270
		37.25893	129.0416	25000	45	270
Intruder						
State	Direction	Waypoint				
Climb	East	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.54889	129.1865	18000	315	240
		36.85774	128.3114	25000	315	240
		37.20762	127.292	30000	315	240

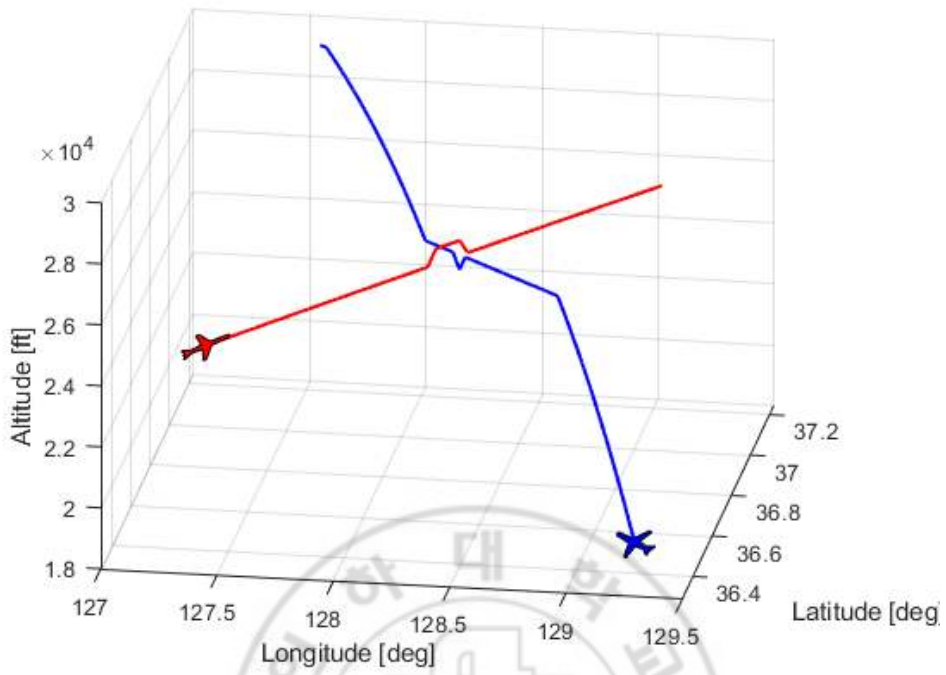


그림 60 Altitude Resolution

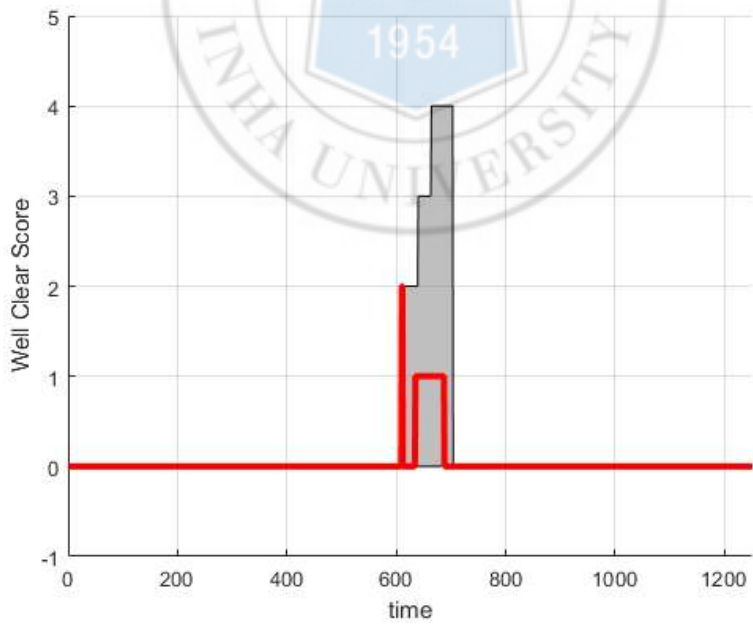


그림 61 Altitude Resolution WCS

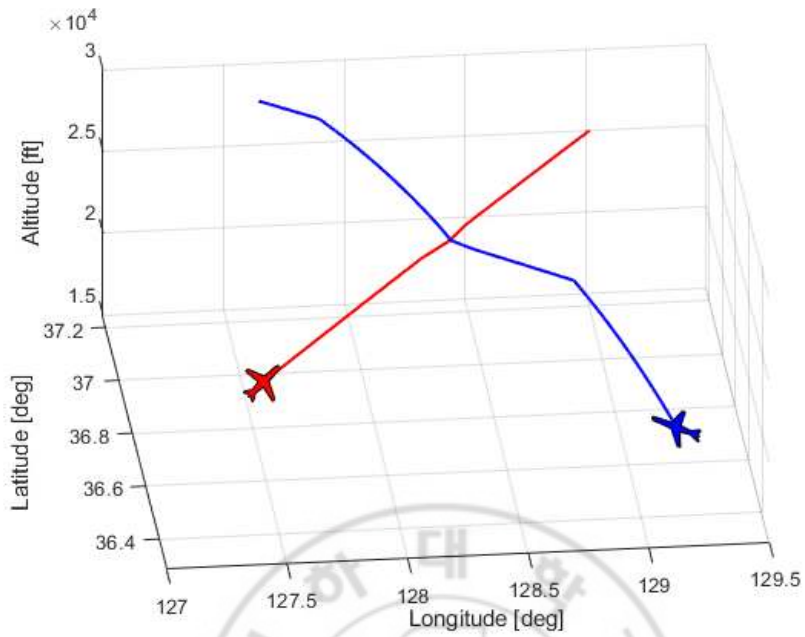


그림 62 Track Resolution

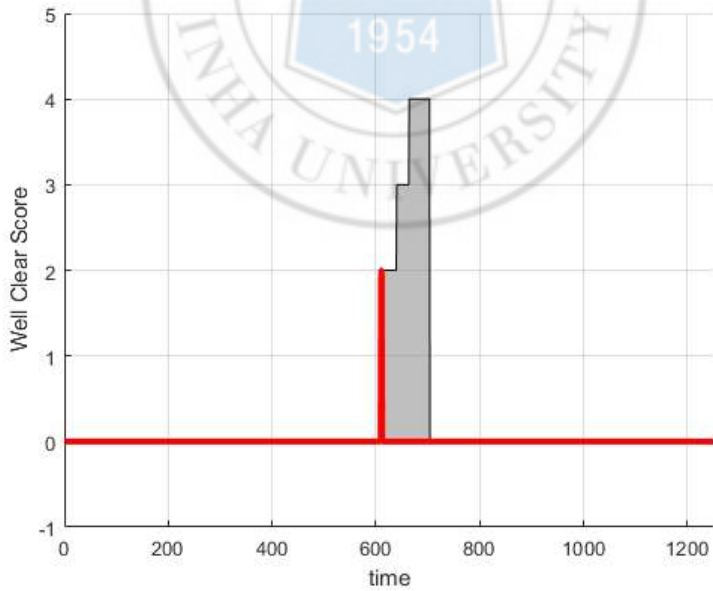


그림 63 Track Resolution WCS

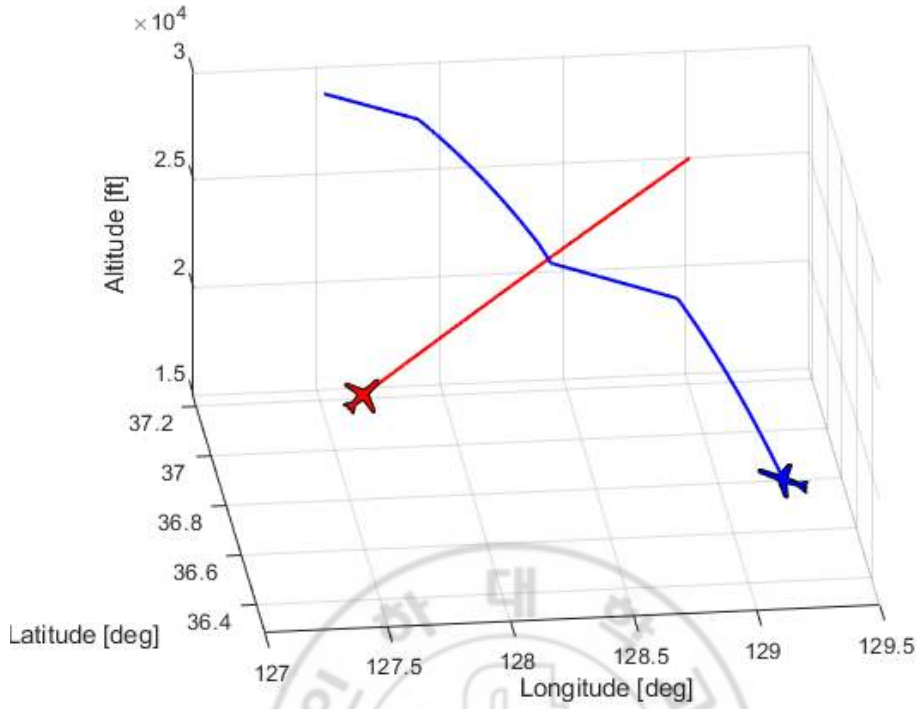


그림 64 Ground Speed Resolution

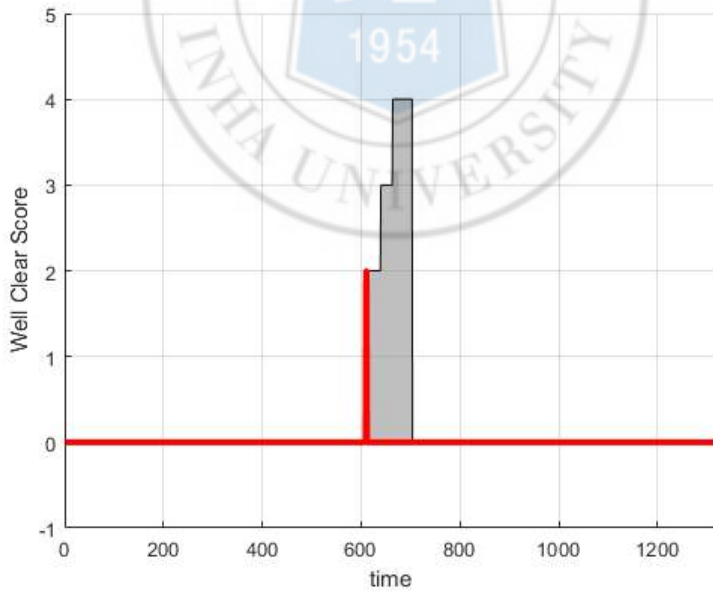


그림 65 Ground Speed Resolution WCS

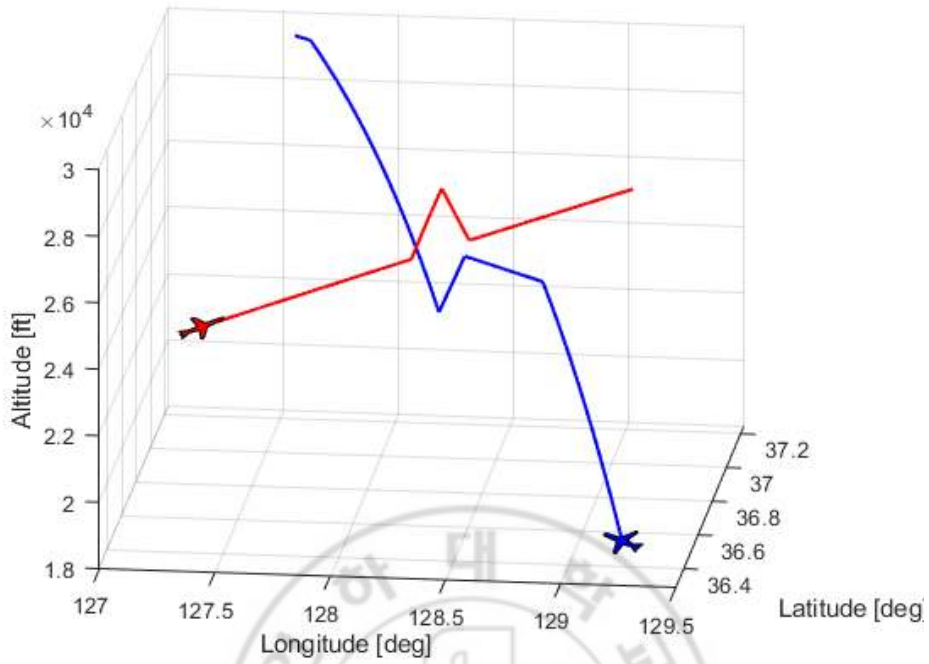


그림 66 Vertical Speed Resolution

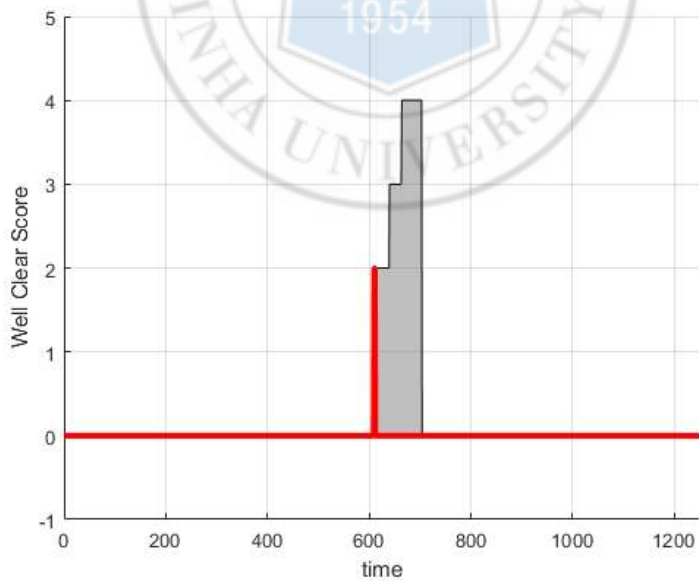


그림 67 Vertical Speed Resolution WCS

표 11 Case 4 시나리오에 대한 비용 함수

Resolution	Cost
Altitude	61.6333
Track	8.4362
Vertical Speed	9.0277
Ground Speed	8.6333

이 시나리오는 Ownship은 순항하며, Intruder는 상승 기동으로 조우한다. 이 시나리오에서 예비 회피 기동은 모두 성공적으로 진행하였으며, WCS가 4가지 Resolution 중 Altitude Resolution에서 WCS가 1이 되어 Preventive Alert가 회피 기동이후 발생한다.

LoWC 만큼 충돌에 직접적인 영향을 주지는 않지만, 더 안전한 Resolution이 있다면 최적경로로 선택하게 된다. 이에, 고도 변경 기동은 비용함수가 다른 기동에 비해 비용함수가 큰 것을 확인할 수 있다.

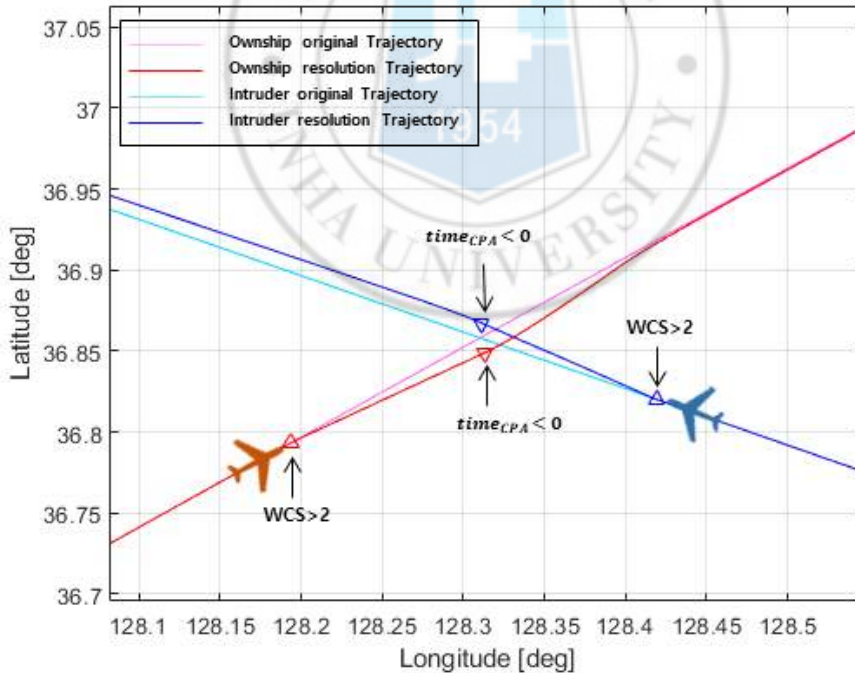


그림 68 회피 기동 중 선택 후 궤적

5.5. Case5: 순항 하강 시나리오

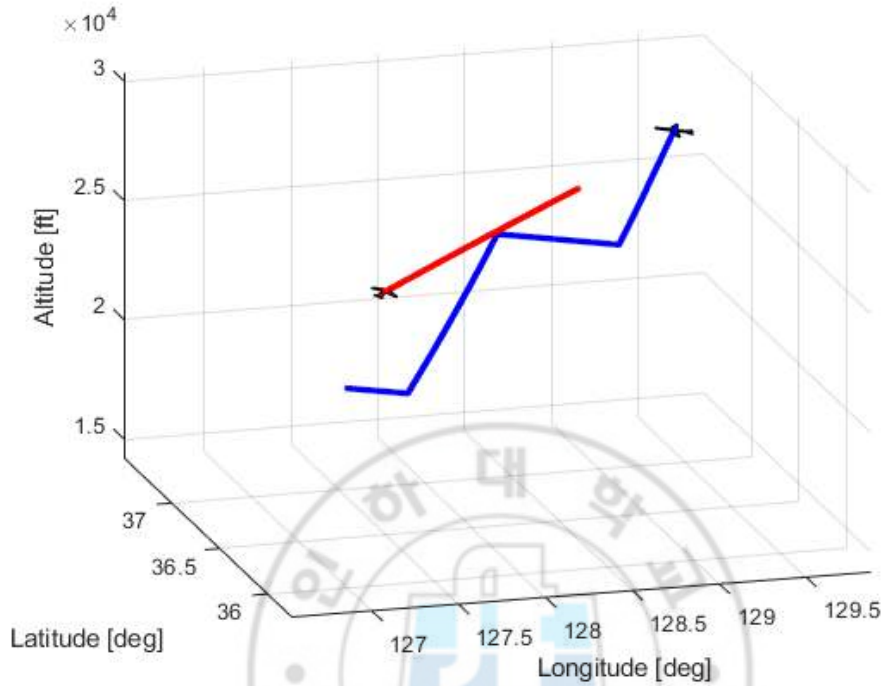


그림 69 순항 하강 시나리오 궤적

표 12 순항 하강 시나리오

Ownship						
State	Direction	Waypoint				
Cruise	West	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.35899	127.4211	25000	45	270
		37.25893	129.0416	25000	45	270
Intruder						
State	Direction	Waypoint				
Descent	East	Latitude [deg]	Longitude [deg]	Altitude [ft]	Track [deg]	TAS [knots]
		36.54889	129.1865	30000	315	240
		36.85774	128.3114	25000	315	240
		37.20762	127.292	18000	315	240

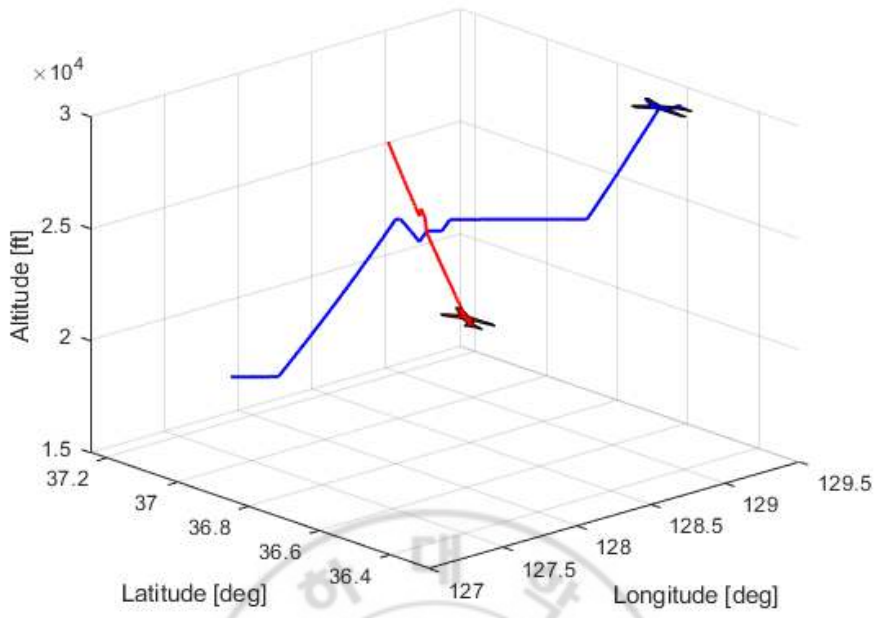


그림 70 Altitude Resolution

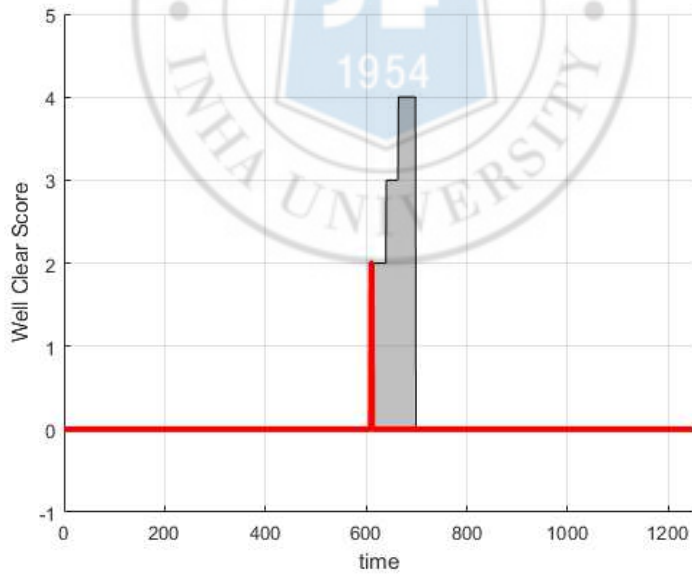


그림 71 Altitude Resolution WCS

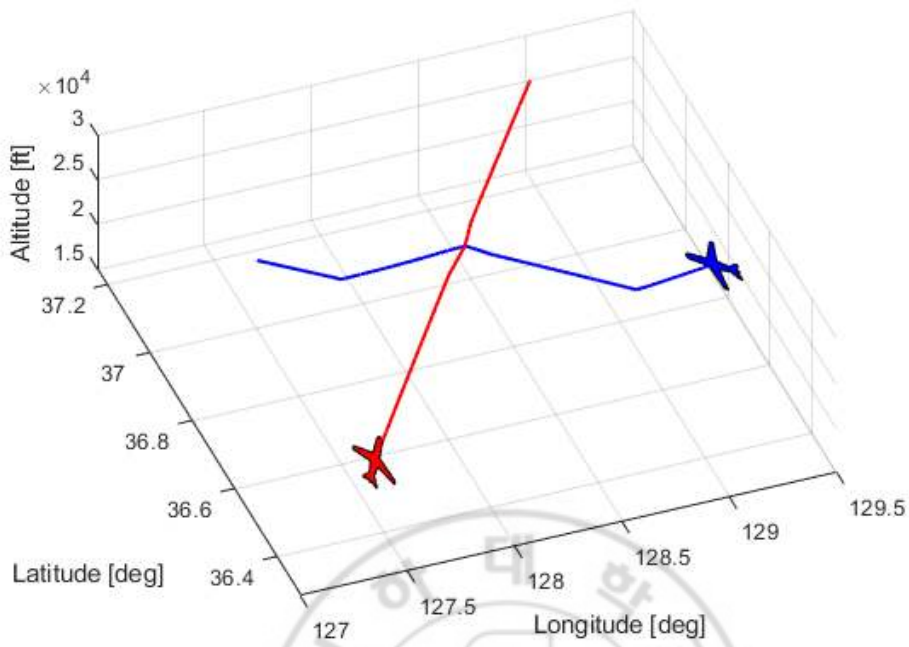


그림 72 Track Resolution

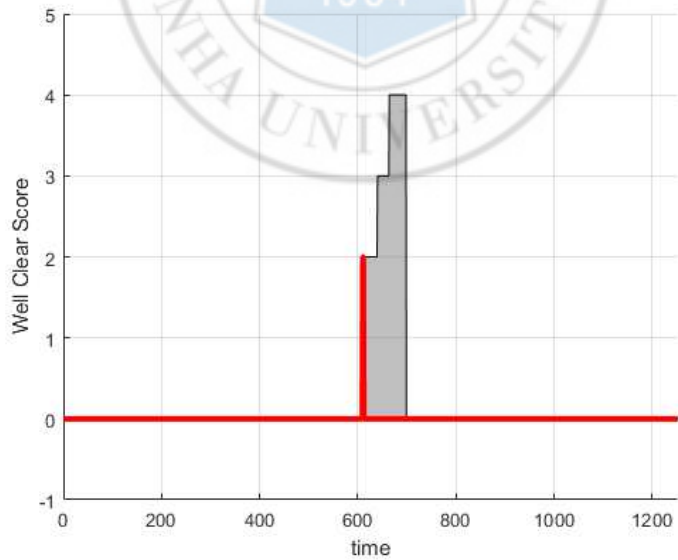


그림 73 Track Resolution WCS

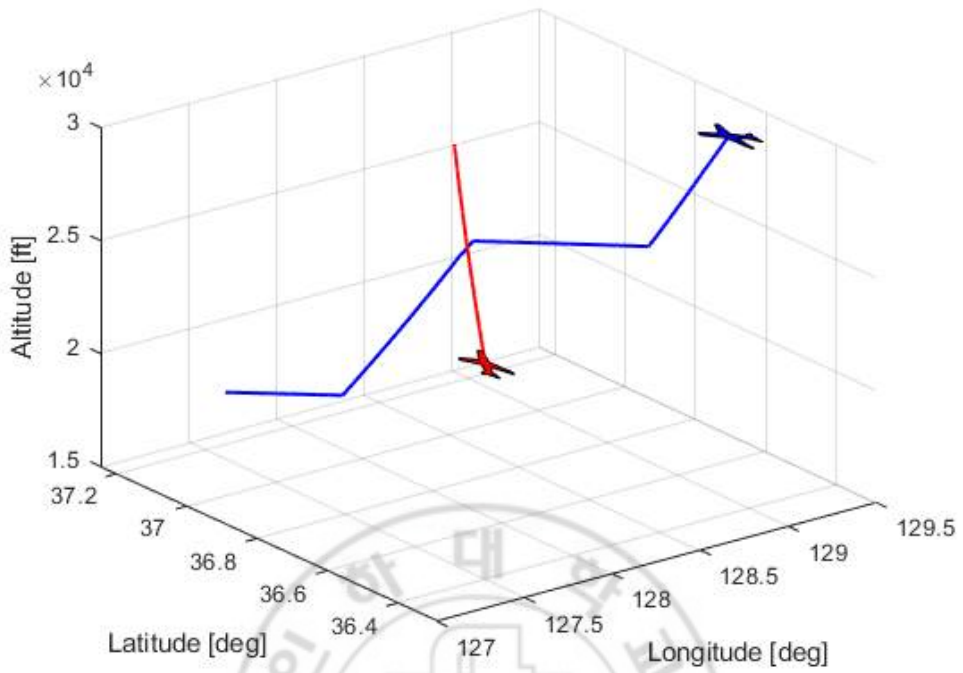


그림 74 Ground Speed Resolution

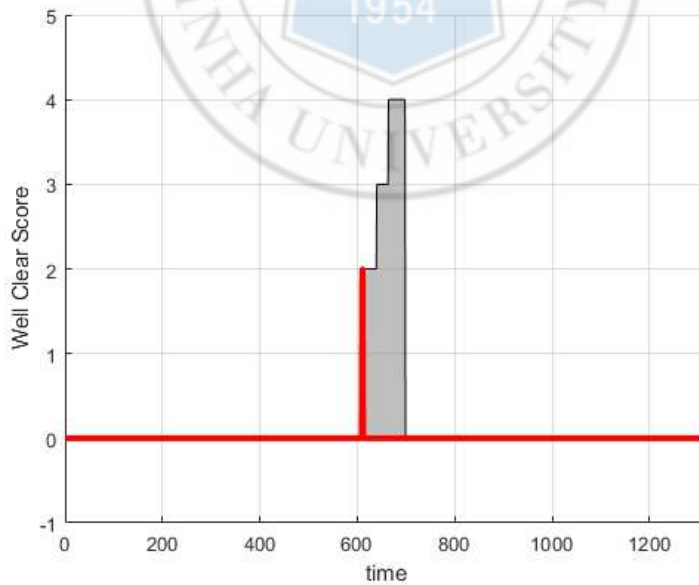


그림 75 Ground Speed Resolution WCS

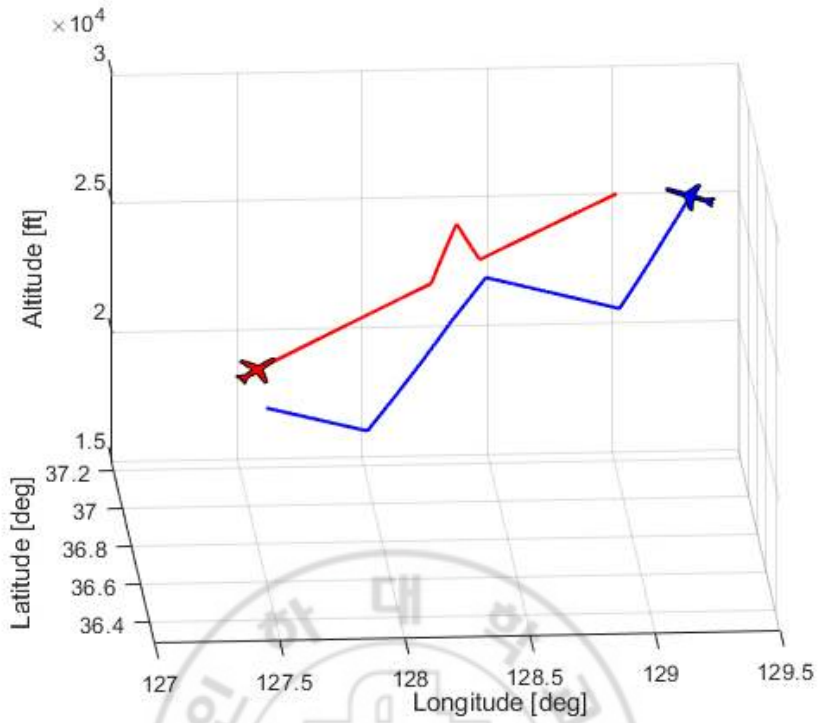


그림 76 Vertical Speed Resolution

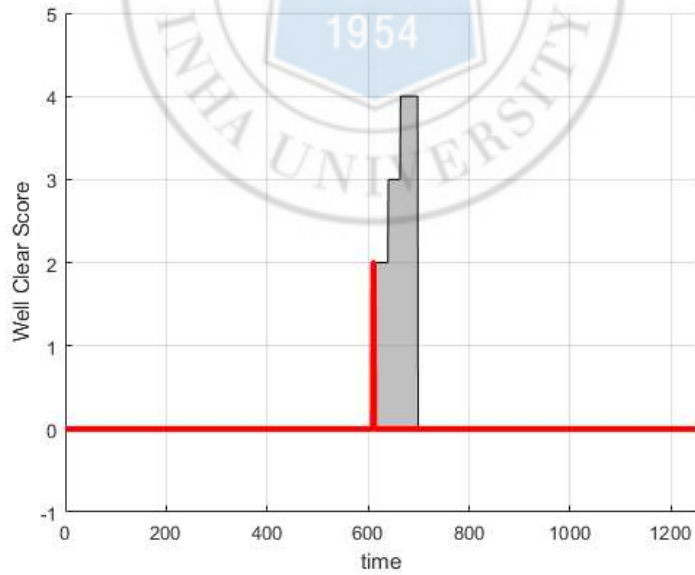


그림 77 Vertical Speed Resolution WCS

표 13 Case 5 시나리오에 대한 비용 함수

Resolution	Cost
Altitude	8.6333
Track	8.4361
Vertical Speed	8.6333
Ground Speed	9.0277

이 시나리오는 Ownship은 순항하며, Intruder는 하강 기동으로 조우한다. 이 시나리오에서 예비 회피 기동은 모두 성공적으로 진행하였으며, 모든 기동이 회피 이후 회복 궤적까지 충돌이 일어나지 않음을 알 수 있었다.

고도 기동은 다음 경로점 고도를 변경함으로써 기동이 회피 고도를 추종하도록 설계하였으며, CPA지점 이후 경로점이 기동 전 고도로 다시 복귀하도록 진행하였다. 수직 속도 기동은 Ownship은 상승 기동을, Intruder는 본래 하강기동을 하는 것을 그대로 유지 하는 방향으로 설계를 하였음을 알 수 있다.

다양한 Resolution 중에 비용함수가 가장 작은 방위각 기동을 선택하였으며, 선택 후 기동의 궤적은 다음과 같다.

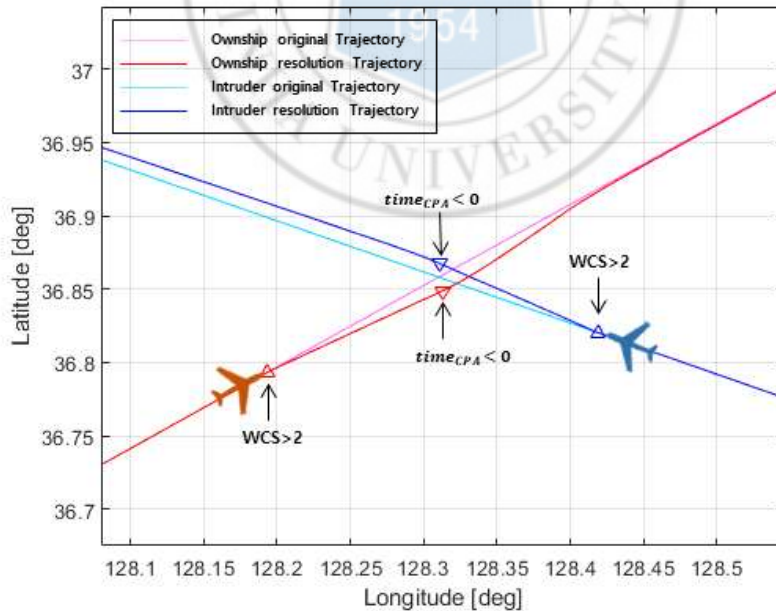


그림 78 회피 기동 중 선택 후 궤적

6. 결론

본 논문은 무인항공기의 자율비행 제어에 필요한 충돌 감지와 회피 알고리즘을 통해 항공기 스스로 경로점을 재설정하고, 최적경로를 배속시물레이션으로 연산하는 알고리즘에 대해 제시하였다. 또한 시물레이션을 통해 최적 경로가 충돌을 발생하지 않는지에 대해 성능을 확인하였다.

항공기의 상태 변수는 ADS-B를 통해 제공받는 가정 하에, 상태 변수 추정 방법은 Nominal 방법으로 경로에 대한 불확실성을 제외하고, 경로점을 추종하는 방향을 반영하였다. 충돌 감지 방법은 기하학적 방법은 Well Clear 기법을 사용하였으며, 충돌 회피 방법은 미리 정의한 Prescribed 방법이 아닌 Optimized 방법을 사용하였다. 비용함수(Cost Function)를 여러 변수를 통해 정의하여 비용을 최소화하는 기동을 선택하였다.

다중 충돌 회피 방법은 NASA에서 개발한 DAIDALUS 알고리즘을 사용하여 Global 방법을 사용하였다. Global 방법은 자신과 상대 항공기만을 고려하는 Pair-wise 방법 뿐 아니라 주변의 모든 항공기를 동시에 고려하여 회피 기동을 산출한다. 주변 항공기의 선별 방법은 DAC 알고리즘을 통해 항공기의 Protected Zone의 기준인 5nmi으로 구역을 나누어 인접한 구역만을 주변 항공기로 식별한다.

회피 기동 방법은 고도 변화, 수직 속도 변화 뿐 아니라 수평속도 변화와 선회 기동을 포함하여 기동을 시물레이션을 진행하였다. 한 가지의 기동만을 선택하도록 하였으며 두 기동을 결합하는 형태의 기동은 향후 연구에 진행할 예정이다.

DAIDALUS에서 계산되는 충돌 감지 구역을 벗어난 기동 범위를 배속 시물레이션으로 경로를 예측 하여, 안전하고 본 궤적으로 돌아가는 시점을 빠른 경로를 찾아내는 알고리즘을 수행하였다. 5가지 조우 시나리오의 경우, 비용함수에 의해 수평면에서의 회피기동이 가장 효율적인 것을 알 수 있었다.

이 알고리즘은 향후 무인항공기의 자율 비행 제어 분야에서 미래 궤적을 예측하는 방향에 적용할 수 있으며, 향후 복잡하고 교통량 고밀도의 공역 상에서 한 Step 앞서 경로를 예측하며, 상시로 바뀌는 상황을 인지하고 상황에 최적의 경로를 찾는 알고리즘에 기여할 것이다.

7. 참고문헌

- [1] 최현진. *무인항공기 충돌회피 기술 동향*. 한국항공우주연구원 e-정책정보 센터. 2016
- [2] 성기정, 김응태, 김성필. *자율비행기술 동향*. 항공우주산업기술 동향 6권2호(2008) pp143-153
- [3] Office of the Secretary of Defense,2005, *Unmanned Aircraft SystemsRoadmap 2005-2030*
- [4] H. Chen, X.-M. Wang, and Y. Li, *A survey of autonomous control forUAV*, in Proc. 2009 Int. Conf. Artif. Intell. Comput. Intell., vol.2, pp. 267-271., 2009,
- [5] RTCA, Inc. *Minimum Operational Performance standards(MOPS) for Detect And Avoid(DAA) Systems*. Radio Technical Commission for Aeronautics (RTCA), Standard Document no. DO-365. 2017.
- [6] Federal Aviation Administration. *TCAS II,Version 7.1. Technical Report HQ 111358*, U.S. Departmentof Transportation. 2011.
- [7] RTCA, Inc. *Minimum Operational Performance Standards (MOPS) forTraffic Alert and Collision Avoidance System II (TCAS II) version 7.1*. DO-185B. 2008.
- [8] D. W.Burgess, S. I. Altman, *TCAS III Bearing Error Evaluation* Lincoln Laboratory,Project Report ATC-231, Apr. 1995
- [9] U.S. Code of Federal Regulations – Title 14 Aeronautics and Space; Part 91 General operating and flight rules; Section 181Course to be Flown. Paragraph 91.181(b) states,
- [10] C. Munoz, A. Narkawicz, G. Hagen, J. Upchurch, A. Dutle, M.Consiglio, and J. Chamberlain, *DAIDALUS: Detect and AvoidAlerting Logic for Unmanned Systems*, Proceedings of the 34th DigitalAvionics Systems Conference (DASC 2015), Prague, Czech Republic, 2015
- [11] C, Munoz, N. Anthony, and J. Chamberlain, *A TCAS-II ResolutionAdvisory Detection Algorithm*. Proceedings of the AIAA Guidance,Navigation and Control Conference, 2013

- [12] 박배선, 이학대. *BADA를 활용한 단순 항공기 궤적 생성 모델*. 한국항공학회 종합학술대회 논문집, 20(1), 190-193. 2016.
- [13] Abramson, M., Refai, M. & Santiago, C., *A Generic Resolution Advisor and Conflict Evaluator (GRACE) in Applications to Detect-And-Avoid (DAA) Systems of Un-manned Aircraft*, Proceedings of 17th AIAA Aviation Technology, Integration, and Operations Conference, Denver, CO, 2017.
- [14] Cone, A., Thippavong, D.P., Lee, S.M., and Santiago, C., *UAS Well Clear Recovery against Non-Cooperative Intruders using Vertical Maneuvers*, 17th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference, (In Press), 2017
- [15] 유창선. *ADS-B 기반의 무인항공기 충돌회피 기술동향*. 항공우주산업기술동향, 13(2), 63-71. 2015.
- [16] Jisoo Kang, Seonyoung Kang, Hyeju Oh, Keeyoung Choi, Hak-Tae Lee, Hyuntae Jung, and Woo-Choon Moon. *Human-in-the-Loop Simulation of Trajectory Based Operation Concept for Remotely Piloted Aircraft System Integration*. AIAA Modeling and Simulation Technologies Conference, AIAA SciTech Forum, (AIAA 2017-0806), 2017
- [17] 조신제, 장대수, 탁민제, 구훤준, 김종성, *TCAS-II를 응용한 무인항공기의 충돌회피기동연구*, 한국항공우주학회지, 제34권 제4호 4, pp. 33~39. 2006.