



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](#)

석사학위논문

SAF의 행위 자동 계획 기법에
관한 연구

2015년



HANSUNG
UNIVERSITY

한성대학교 국방과학대학원

국방 M & S 학과

국방 M & S 학전공

조 준 호

석사학위논문
지도교수 김종만

SAF의 행위 자동 계획 기법에 관한 연구

A Study of Automated Behavior Planning Technique for
Semi Automated Forces



2014년 12월 일

HANSUNG
UNIVERSITY

한성대학교 국방과학대학원

국방 M & S 학과

국방 M & S 학전공

조 준 호

석사학위논문
지도교수 김종만

SAF의 행위 자동 계획 기법에 관한 연구

A Study of Automated Behavior Planning Technique for
Semi Automated Forces

위 논문을 국방M&S학 석사학위 논문으로 제출함

2014년 12월 일

한성대학교 국방과학대학원

국방 M & S 학과

국방M&S학전공

조 준 호

조준호의 국방M&S학 석사학위논문을 인준함

2014년 12월 일



심사위원장 _____ 인

심 사 위 원 _____ 인

심 사 위 원 _____ 인

국 문 초 록

SAF의 행위 자동 계획 기법에 관한 연구

한성대학교 국방과학대학원

국방 M&S 학과

국방 M&S학 전공

조 준 호

군에서는 오랫동안 군사 시뮬레이션 시스템을 다양한 분야에 활용해 오고 있다. 분석, 훈련, 획득 분야, 그리고 최근에는 시험평가 분야 및 전투효과 분석을 위해서 모델링 및 시뮬레이션을 활용하고 있다. 또한, 컴퓨터 상에서 인간행위와 부대의 집단논리를 표현하는 자율적 가상 개체, 즉 가상군(Computer Generated Force, CGF)에 대한 기술개발이 주목받고 있다. 가상군은 훈련 참가자의 개입 여부 및 자율 지능의 정도에 따라 SAF(Semi Automated Forces)와 FAF(Fully Automated Forces)로 구분한다.

민간 게임이나 기존 위게임 모의 모델에서는 개체의 지능 처리를 위하여 대부분 유한 상태 기계(finite state machine)나 규칙기반 시스템(rule-based system)을 적용하고 있다. 이러한 기술은 군사 전문가의 전문 지식을 쉽게 구축할 수 있는 장점이 있지만, 발생할 수 있는 모든 상황을 예측하여 그 상황에 적절한 대처방법을 모델링 단계에서 미리 규칙으로 구축해야 하는 단점이 있다.

이러한 문제점을 해결하기 위하여, 본 연구는 모의 개체로 하여금 주어진 목적을 달성하기 위해 실시간으로 계획을 수립하는 인공지능 계획(AI planning) 기법을 사용한다. 인공지능 계획 기법은 개체가 특정 목적에 도달하기 위해 어떤 행동을 수행해야 될지 결정하는 과정이다. 인공지능 계획 기

법을 적용함으로써, 기존 모의논리 또는 규칙에 의한 모델링 방식보다 높은 추상화 수준에서 행위를 모델링할 수 있고 동적으로 행위를 구성하기 때문에, 소스코드나 스크립트에 의한 기존 동작 방식보다 더 유연하다고 볼 수 있다.

본 논문에서는 SAF 개체가 주어진 임무를 달성하기 위하여 스스로 행동을 계획하여 상황에 능동적으로 반응할 수 있는 행위 자동 계획 기법을 제시하였다.

SAF 개체가 수행하는 행위를 계층적으로 구성하기 위한 행위 구조는 HTN 기반 계획(HTN-based planning) 기법에서의 과업 구조를 활용하여 제시하였다. 행위는 계층적인 구조를 가지며, 행위를 수행하기 위한 필수 조건인 시작조건과, 행위를 종료하기 위한 조건인 종료 조건을 퍼지 표현식으로 표현하였다.

행위의 최종 목적을 달성하기 위하여 필요한 행위 목록들을 자동으로 생성하는 행위 계획 수립 기법은 STRIPS 기반 계획(STRIPS-based planning) 기법을 활용하였다. 행위 계획 수립 시 그 탐색 영역을 줄이기 위하여 인공지능의 계획 방법 중의 하나인 목표 상태로부터 현재 상태를 역으로 찾는 목표지향 행위 계획(Goal Oriented Action Planning, GOAP) 방법을 활용함으로써, 자동으로 행위 계획을 수립하는 방법을 제시하였다.

SAF 개체의 행위 모델링 및 모의 실험을 통해서도 행위 자동 계획 기법의 효과와 설계를 검증하고, SAF 개체의 행위 자동 계획 기법의 활용 방안을 제시하였다.

【주요어】 가상군, SAF, 인공지능, 자동계획, 목표지향행위계획, 퍼지, 모델링, 시뮬레이션

목 차

제 1 장 서론	1
제 1 절 연구 배경 및 목적	1
제 2 절 연구 범위 및 구성	3
제 2 장 이론적 배경	5
제 1 절 SAF(Semi Automated Forces)	5
제 2 절 기존 SAF 운용 도구	5
1. OneSAF	6
2. VR-Forces	9
3. STAGE	11
제 3 절 인공지능 기술	13
1. 유한 상태 기계(Finite State Machine)	13
2. 퍼지 추론(Fuzzy Inference)	15
3. STRIPS 기반 계획(STRIPS based planning)	17
4. HTN 기반 계획(Hierarchical Task Network based planning)	19
제 3 장 행위 자동 계획 기법 설계	21
제 1 절 행위 자동 계획 기법 개념	21
제 2 절 계층적 행위 설계	23
1. 계층적 행위 정의	23
2. 행위 구조	25
3. 클래스(Class) 다이어그램	29

제 3 절	행위 계획 기능 설계	35
1.	가용 행위 검색	36
2.	연결 행위 탐색	37
3.	시작/종료 조건 매칭	39
4.	최적의 연결 행위 선택	43
제 4 장	행위 자동 계획 기법 모의 실험	44
제 1 절	모의 환경 및 조건	44
제 2 절	모의 준비	45
1.	모의 시나리오 정의	45
2.	과업 및 행위 모델링	47
제 3 절	모의 실시 및 결과 분석	52
제 5 장	결론	59
【참고문헌】		61
ABSTRACT		64

【 표 목 차 】

[표 2-1] VR-Forces가 제공하는 조건 목록	10
[표 3-1] 계층적 행위 관계의 예	24
[표 3-2] XML 형태의 하위 행위의 구성의 예	27
[표 3-3] 퍼지 표현의 예	27
[표 3-4] 기본 행위 및 복합 행위 클래스 속성과 오퍼레이션	31
[표 3-5] 퍼지 논리 표현식 클래스 속성과 오퍼레이션	33
[표 3-6] 퍼지 논리 표현식 평가 방법	36
[표 4-1] ‘전술적 이동’ 과업 모델링	48
[표 4-2] ‘행정적 이동’ 과업 모델링	48
[표 4-3] ‘집결지 행동’ 과업 모델링	49
[표 4-4] ‘진지 점령’ 과업 모델링	49
[표 4-5] ‘우회 이동’ 과업 모델링	49
[표 4-6] ‘대응사격’ 과업 모델링	50
[표 4-7] ‘포탄 낙하시 조치’ 과업 모델링	50
[표 4-8] 행위 모델링	51
[표 4-9] 퍼지 팩트 정의	52
[표 4-10] 기존 위게임 모델, 상용 SAF 운용도구와의 차이점 비교	57

【 그림 목 차 】

<그림 2-1> OneSAF 모델링 개념	7
<그림 2-2> VR-Forces GUI 화면	9
<그림 2-3> STAGE 화면	11
<그림 2-4> STAGE 의 행위 스크립트의 예	12
<그림 2-5> 유한 상태 기계(FSM)의 예	14
<그림 2-6> 퍼지 추론 유형	16
<그림 2-7> 퍼지 추론 시스템의 과정	16
<그림 2-8> STRIPS 기반 계획의 예	18
<그림 2-9> HTN 기반 계획의 예	20
<그림 3-1> SAF의 행위 자동 계획 기법 개념도	21
<그림 3-2> 임무-과업-행위 구성의 예	23
<그림 3-3> 조합패턴의 클래스 다이어그램	25
<그림 3-4> 행위 구조도	26
<그림 3-5> 퍼지 변수의 정의	28
<그림 3-6> 행위 클래스 다이어그램	30
<그림 3-7> 퍼지 논리 표현식 클래스 다이어그램	32
<그림 3-8> 행위 계획 절차	35
<그림 3-9> 연결 행위 탐색의 예	38
<그림 3-10> 두 퍼지 팩트간 비교 연산 그림	40
<그림 3-11> 연결 행위의 전체 비용 산출의 예	43
<그림 4-1> 모의 시나리오 상황도	45
<그림 4-2> mergeCGF 도구를 사용한 모의 시나리오 작성	46
<그림 4-3> mergeCGF 도구를 사용한 임무 계획 작성	46
<그림 4-4> 청군 전차 1소대의 예상되는 행위 계획 수립 개념도	47
<그림 4-5> mergeCGF 소프트웨어를 사용한 모의 실행	53
<그림 4-6> 임무 작성 창 의 과업 진행 상황	53
<그림 4-7> ‘행정적 이동’ 중인 전차 1소대	54

<그림 4-8> 적 전차 사격에 대한 행위 계획 수립 로그	54
<그림 4-9> 3차원 상황도에서의 포탄 낙하 상황	55
<그림 4-10> 적 포탄낙하 상황에 대한 행위 계획 수립 로그	56



제 1 장 서 론

제 1 절 연구 배경 및 목적

군에서는 오랫동안 군사 시뮬레이션 시스템을 다양한 분야에 활용해 오고 있다. 분석 분야에서는 작계 및 방책 분석, 군 구조분석, 전력평가 등을 위해 활용하고, 훈련분야에서는 장비조작 숙달훈련, 합동 전쟁연습, 지휘관 및 참모의 지휘훈련, 작계 및 전술훈련을 위해 이용하며, 획득 분야에서는 체계설계, 시험평가, 소요검증 등을 위해 활용하고 있다, 최근에는 체계 시험평가지 실제 환경에서 실시하기 어려운 시험을 모델링 및 시뮬레이션으로 대체하기도 하고, 개발된 체계의 실 전투효과를 분석하는 데에도 활용하고 있으며, 향후 효율적인 국방경영을 위한 과학적 분석수단으로써 정책 및 의사결정 분야에서의 활용성도 점차 높아질 것으로 기대된다.¹⁾

이와 같이 군사 시뮬레이션의 활용성이 높아짐에 따라 보다 신뢰성 있는 데이터를 획득하기 위하여 적은 인원으로 단일 체계 또는 소규모 부대의 운용보다 여러 체계 또는 대규모 부대의 상황을 모의하는 것이 효과적이며, 최근에는 컴퓨터 상에서 인간행위와 부대의 집단논리를 표현하는 자율적 가상 개체, 즉 가상군(Computer Generated Forces, CGF)에 대한 기술 개발이 주목받고 있다. 가상군을 워게임 모델에 사용함으로써, 훈련용 가상 모의 모델에서는 적군 또는 아군의 지원부대를 가상군으로 운용하여 게임어 및 훈련대상 모의 개체 수를 줄일 수가 있으며, 분석 모델에서는 시나리오 작성 단계에서 모의 개체가 수행하는 임무를 단순화 할 수 있기 때문에 시나리오 작성시 발생하는 번거로움을 줄일 수 있다

CGF는 훈련 참가자의 개입 여부 및 자율 지능의 정도에 따라 SAF (Semi Automated Forces)와 FAF(Fully Automated Forces)로 구분한다. 현재의 CGF 기술은 아직 FAF를 모의하기에는 부족한 것이 현실이므로 CGF와 SAF를 거의 같은 의미로 사용하기도 한다.²⁾

1) 이성기외 2명, 『가상군 기술의 국방 적용방안』, 한국인터넷정보학회, 제13권 제4호, 2012, p.17~23

민간 게임에서 운용되는 NPC(Non Player Character) 개체는 주로 유한 상태 기계(Finite State Machine)를 인공지능 기술로 사용하고 있으며, 미 육군 PEO-STRI³⁾의 OneSAF 또는 VT-MAK의 VR-Forces와 같은 군사 모의 목적의 모델은 개체의 지능 처리를 위해 규칙기반 시스템(rule-based system)을 인공지능 기술로 적용하고 있다. 이러한 기술은 군사 전문가의 전문 지식을 쉽게 구축할 수 있는 장점이 있지만, 이것은 발생할 수 있는 모든 상황을 예측하여 그 상황에 적절한 대처방법을 모델링 단계에서 미리 규칙으로 구축해야 하는 단점이 있다.

SAF 모의 모델에서 새로운 행위를 생성하여 모의하거나, SAF 모의 모델이 다른 모델과 상호 연동할 경우, SAF 개체는 새로운 상황에 대처해야 하며, 이는 모의 환경 및 피/아 상태 변화에 따라 전혀 예측하지 못한 상황이 발생하여 더 이상 주어진 과업을 수행할 수 없는 문제가 발생할 수 있다.

이러한 문제점을 해결하기 위하여 모의 개체로 하여금 주어진 목적을 달성하기 위해 계획을 수립하는 인공지능 계획(AI planning) 기법을 사용한다. 인공지능 계획 기법은 개체가 특정 목적에 도달하기 위해 어떤 행동을 수행해야 될지 결정하는 과정이다. 기존의 위게임 모델이나 규칙기반 시스템에서는 발생 가능한 예외 상황에 대해 모의논리 또는 규칙으로 구현하였다. 하지만 인공지능 계획 기법과 기존 방식의 차이점은 행위를 모델링하는 추상화 수준이 서로 다르다는 것이다. 인공지능 계획 기법은 기존의 모의논리 또는 규칙에 의한 방식보다 높은 추상화 수준에서 행위를 모델링할 수 있다. 또한 소스코드나 스크립트에 의한 동작방식이 아닌, 동적으로 행위를 구성하기 때문에 기존 방식보다 더 유연하다고 볼 수 있다. 예를 들어, ‘소화기에 의한 피해 발생시 대응 사격’이나 ‘탄약이 부족하면 보급을 실시’하는 것과 같은 행동들은 모의 논리나 규칙으로 충분히 구현 가능하다. 하지만, 모의논리나 규칙에서 다루던 것들을 모델링 수준을 높여서 ‘대응 사격’ 행위나 ‘보급’ 행위를 모델링할 때 관련 조건들을 고려한다면, ‘중화기에 의한 피해 발생시’ 또는 ‘탄약이 파손되면’과 같이 고려하지 않은 우발 상황에 대해서도 동적으로 상황

2) 이성기외 4명, 『자율지능형 컴퓨터 가상개체 기술의 국방 M&S 적용현황 및 활용방안』, 정보과학회지 제26권 제11호, 2008, p.27-32

3) PEO-STRI(Program Executive Office for Simulation, TRaining & Instrumentation) : 미 육군 모의훈련사업국

대처가 가능할 것이다. 이러한 방법은 실 세계에서 사람의 계획 수립 절차와 유사하다. 일반적으로 계획 기법은 상태기반 계획 기법인 STRIPS 기반 계획 (STRIPS-based planning)과 과업기반 계획 기법인 HTN 기반 계획 (HTN-based planning)이 있다.⁴⁾

본 논문에서는 위 두 가지 계획 기법을 모두 적용하여 SAF 개체가 주어진 임무를 달성하기 위하여 스스로 행동을 계획하여 상황에 능동적으로 반응할 수 있는 기법을 제안한다.

제 2 절 연구 범위 및 구성

본 논문에서는 SAF 개체의 행위 자동 계획 기법을 위해 크게 두 가지 방법을 제안한다. SAF 개체가 수행하는 임무를 행위 별로 계층적으로 구성하기 위한 행위 구조와, 행위의 최종 목적을 달성하기 위하여 필요한 행위 목록들을 자동으로 생성하는 행위 계획 수립 기법에 대해 설계하였다.

SAF 개체가 수행하는 임무(mission)는 필요한 여러 과업(task)으로 기술되고, 과업은 여러 개의 복합적인 행위(behavior)로 구성할 수 있다. 또한 복합적인 행위도 여러 개의 행위를 계층적으로 구성할 수 있다. 즉, 하나의 임무에 대해서 필요한 여러 과업이 기술되고, 과업을 달성하기 위해 필요한 여러 행위들이 계층적으로 구성된다. 예를 들어 방어 작전 시나리오에 필요한 ‘진지 전투’ 임무 중 ‘진지 점령(전환)’이라는 과업은 ‘전환 진지로 이동’이라는 행위와 ‘진지 보강’라는 행위로 구성하며, ‘진지 보강’ 행위는 다시 ‘진지 작업’ 행위와 ‘사주경계’ 행위로 구성할 수 있다.

각각의 행위들은 시작조건과 종료조건을 포함한다. 시작조건은 해당 행위를 수행하기 위한 필수 조건을 의미하고, 종료조건은 그 행위가 종료될 수 있는 조건들로 구성한다. 그리고 이 조건들의 만족 여부를 판단하기 위한 조건 매칭 방법은 조건을 퍼지화하여 퍼지 언어항을 비교하는 방법을 사용하였다. 행위의 종료 상태를 달성하기 위하여 필요한 행위 계획 수립시 그 탐색 영역을

4) E. Long, 『Enhanced NPC Behaviour using Goal Oriented Action Planning,』 Master thesis, School of Computing and Advanced Technologies, University of Abertay Dundee, 2007.

줄이는 방법으로, 인공지능의 계획 방법 중의 하나인 목표 상태에서부터 현재 상태를 역으로 찾는 목표 지향 행위 계획(Goal Oriented Action Planning, GOAP) 방법을 사용하였다.

본 연구의 논문 구성은 다음과 같다.

1장은 서론으로 연구의 배경 및 목적, 연구 범위와 구성에 대하여 기술하였다.

2장에서는 SAF의 행위 자동 계획 기법에 대한 이론적 배경으로 SAF의 정의 및 기존 SAF 운용도구, 그리고 유한상태기계, 퍼지 추론, 계획 기법 등의 인공지능 이론에 대하여 설명하였다.

3장에서는 SAF의 행위 자동 계획 기법에 대한 설계로, 행위 자동 계획 기법의 개념, 계층적 행위의 정의 및 구조 설계, 그리고 행위 계획 기능에 대해 설계하였다.

4장에서는 행위 자동 계획 기법을 실험하기 위한 모의 환경과, 모의 준비를 위하여 모의 시나리오 작성 및 사용되는 행위 모델링을 하였다. 그리고 모의 실험을 실시하여 행위 자동 계획 기법의 효과와 설계를 검증하였다.

5장에서는 본 논문의 활용 방안을 제시하고 결론을 맺었다.

제 2 장 이론적 배경

제 1 절 SAF(Semi Automated Forces)

앞 장에서 살펴본 바와 같이 군사 시뮬레이션 환경에서 원활한 전술훈련을 수행하기 위해 필요한 적군 및 지원군 역할을 하기 위해서 많은 게임어 동원이 불가피하고, 이에 따라 이들 요원이 동원된 부대나 부서에서는 해당 조직의 운용에 많은 어려움을 겪게 된다. 이러한 게임어 동원 문제를 근본적으로 해결하기 위하여, 적이나 아군의 부대 및 무기체계를 가상군(Computer Generated Forces, CGF)으로 모의하여 다양한 전술훈련을 경제적이고 효율적으로 수행하는 것이 가능하다. 가상군은 인간 행위를 모델링하여 전투개체가 자동적으로 행동을 수행하도록 하는 전투개체의 컴퓨터 표현을 의미한다.

가상군은 사람의 개입 여부에 따라 완전-자율(fully-automated)과 반-자율(semi-automated)로 구분된다. SAF(Semi Automated Forces)는 사람의 개입이 존재하는, 일부 자율성을 가지고 있는 가상군이다. 그렇기 때문에 SAF 운영자는 개체의 전체적인 계획을 수립하거나, 달성해야 하는 임무 또는 과업을 직접 지정해야 한다. 이것은 스스로 목표를 세우고, 스스로 임무를 결정하여 행동하는 완전-자율과는 차이가 있다. 시나리오에 작전 및 임무가 주어진 상황에서, 지휘관의 명령에 따라 행동해야 하는 전투 모의에서의 가상군 개체는 완전-자율 보다는 반-자율이 더 적합할 수 있다.

제 2 절 기존 SAF 운용 도구

SAF 모의 기술을 핵심기술로 인식한 미국은 90년대 중반부터 SAF 기술을 개발하여 왔다. 초기 Modular SAF 개발을 시작으로 CCTT(Close Combat Tactical Trainer)용 SAF 등을 개발하였으며, 90년대 후반부터 현재까지 SAF 기반 모의를 위한 프레임워크인 OneSAF를 개발하여 활용하는 단계에 이르렀다. 또한, VT MAK, PRESAGIS와 같은 민간 업체에서도 전술 모의를 위한 SAF 모의 도구인 VR-Forces, STAGE 등의 제품을 출시하여 다양한 곳에서

사용하고 있다. 국내 M&S에서도 위와 같은 상용 제품을 주로 사용하여 활용하고 있으며, 현재 국방과학연구소를 주축으로 관련 핵심 기술을 연구 개발하고 있는 실정이다.

아래 절에서는 국내·외에서 주로 사용하고 있는 OneSAF, VR-Forces, STAGE 도구에 대한 특징과, 특히 개체의 지능적 행위 수행 방법에 대해 기술하였다.

1. OneSAF

미 육군 모의훈련사업국(PEO-STRI)에서 개발한 OneSAF는 미 육군의 차세대, 조립형, 개체기반 시뮬레이션 모델로써, 첨단개념연구(ACR⁵⁾), 훈련, 연습 및 군사작전(TEMO⁶⁾), 연구개발 및 획득(RDA⁷⁾)분야와 같은 다양한 M&S 영역에 대한 통합 시뮬레이션 서비스를 제공하기 위해 개발되었다. 조립형이라는 말은 하나의 실행파일이나 구성요소로 이루어진 것이 아니라 다양한 구성요소가 활용목적에 따라 체계적으로 통합되어 다양한 형상을 구성할 수 있다는 의미이다. 조립형 개념을 통해 다양한 확장이 가능하고 HLA를 지원할 수 있도록 설계되어 타 모델이나 체계와 상호운용성을 보장한다. 또한 OneSAF에서 운용되는 개체는 가상군의 특성을 가진다.⁸⁾

OneSAF는 조립형이라는 특성을 갖고 있기 때문에, 타 모델이나 시뮬레이션 체계와는 상이한 모델링 개념을 가지고 있다. 특히 기존 모의 모델은 모의논리에 개체를 적용시키는데 반해 OneSAF는 개체에 모의논리를 적용시키는 개념을 사용한다. OneSAF에서 활용되는 모델은 물리적 및 행위적 신뢰성을 시뮬레이션을 위한 구성요소로 표현한 것으로 개체모델, 부대모델, 물리모델, 행위모델, 환경모델로 분류된다.

개체모델은 병사, 전차, 트럭, 항공기 등과 같은 단일의 전투 플랫폼을 의미하며, 시뮬레이션 기본 단위이다. 이것은 구성요소(chassis), 물리모델, 행위의 조합으로 구성된다. 부대모델은 소대, 중대, 대대와 같은 군대 조직을 의미

5) ACR(Advanced Concepts and Requirements) : 첨단 개념 연구

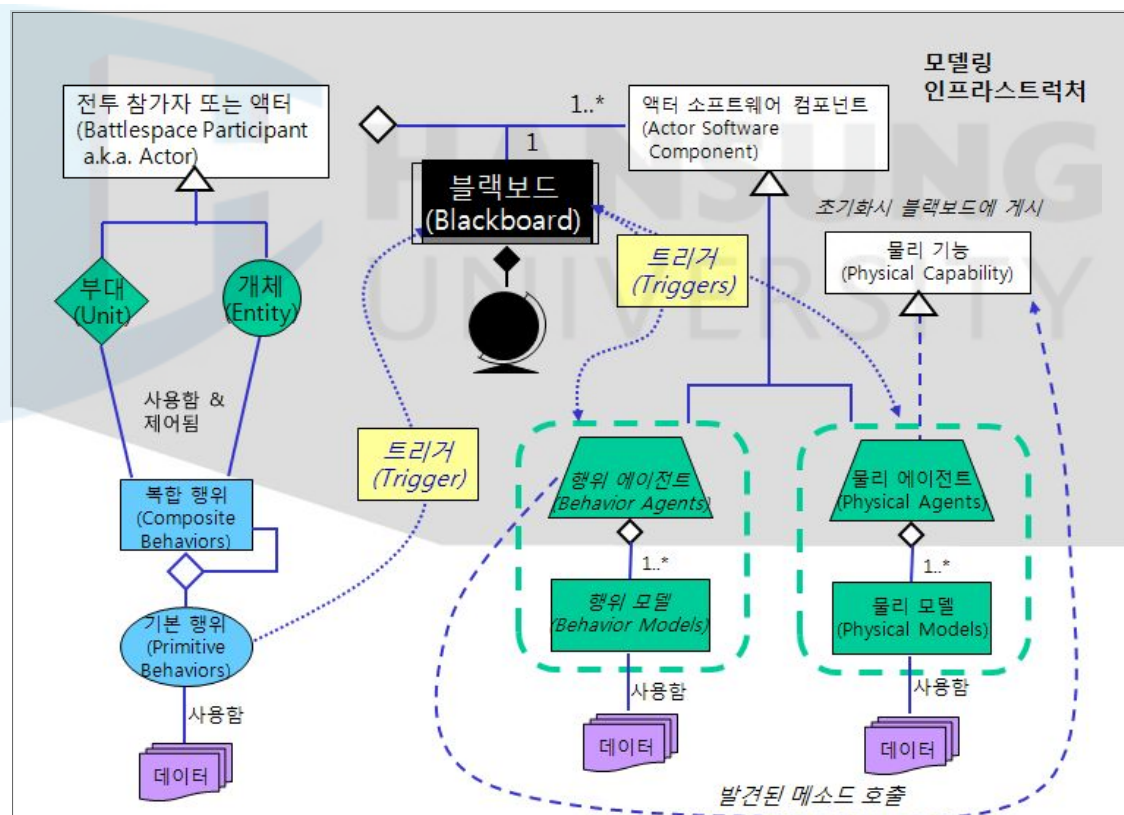
6) TEMO(Training, Exercises, and Military Operations) : 훈련, 연습 및 군사작전

7) RCA(Research, Development, and Acquisition) : 연구, 개발, 획득

8) 문형곤외 2명, 『OneSAF 모형 도입/실용화』, 한국국방연구원 연구보고서 모07-2474, 2008, p.44~63

하며, 1개 이상의 개체와 부대모델 및 지휘통제를 위한 행위로 구성된다. 물리모델은 취약성, 기동, 통신, 무기 등과 같은 무기 및 장비의 기본적인 특성 및 물리적 기능을 표현하였다. 물리모델은 전장 속에서의 개체가 가진 독립된 기능이기 때문에 개체는 전장상황과 같은 정보를 인식하는 기능이 요구된다. 이러한 정보 인식은 물리 에이전트라는 객체가 담당하며, 이 물리 에이전트에 물리모델이 포함되어 물리모델의 기능을 호출하는 역할을 수행한다. 행위모델은 계획수립, 실행, 평가 등과 같이 지휘통제 및 전술적 기능을 수행하는 개체 및 부대의 활동을 의미한다. 행위모델은 개체 및 부대의 기본적인 활동인 기본행위(primitive behavior)와 하위 행위들의 조합인 복합행위(composite behavior)로 분류된다. 환경모델은 시뮬레이션과 관련된 지형 및 기상과 같은 환경자료를 표현한 것이다.⁹⁾

아래 <그림 2-1>은 OneSAF의 모델링 개념을 도식화 한 것이다.¹⁰⁾



<그림 2-1> OneSAF 모델링 개념

9) 김태섭, 『무기체계 분석을 위한 모의엔진 아키텍처 연구』, 한국시뮬레이션학회 논문지 제 19권 제 2호, p51~62, 2010

10) Dr Nash, 『Overview of the OneSAF Models Infrastructure』, OneSAF Users Conference, 2009

위 그림에서 좌측 상단의 전투 참가자(battlespace participant)는 액터(actor)라고도 불리며 개체와 부대가 된다. 개체와 부대는 행위를 가질 수 있는데, 행위는 기본행위 또는 복합행위이고 파라미터 자료와 같은 데이터를 활용할 수 있다. 액터 소프트웨어 컴포넌트(actor software component)는 액터가 시뮬레이션에서 자신의 역할을 수행하기 위해 구현된 능력이다. 이것은 행위모델을 통제하는 행위 에이전트와 물리모델을 통제하는 물리 에이전트를 포함한다. 각 모델은 기능 구현을 위해 데이터를 사용한다. 물리 에이전트는 물리 기능(physical capability)에 포함될 수도 있는데, 행위 에이전트는 행위 구현을 위해 물리 기능을 활용할 수 있다. 액터는 타 액터와의 상호작용을 위해 블랙보드(blackboard)를 활용한다. 이것은 다른 액터에게 보낼 메시지나 상태와 같은 데이터를 기록하는 역할을 한다. 이 블랙보드의 내용은 다른 액터들이 모델링 인프라스트럭처(modeling infrastructure)를 통해 읽을 수 있다.

위 그림에서 주목할 점은 물리 에이전트와 행위 에이전트는 모델링 인프라스트럭처의 범위에 포함되는데 반해, 물리모델과 행위모델은 모델링 인프라스트럭처에 포함되지 않는다. 이는 모델링 인프라스트럭처에서는 역할 및 기능과 관련된 부분은 에이전트가 전담하는 것으로 간주하고 실제 기능은 에이전트가 모델을 호출하여 처리하도록 하여 모델의 조립성 및 확장성을 제공한다.

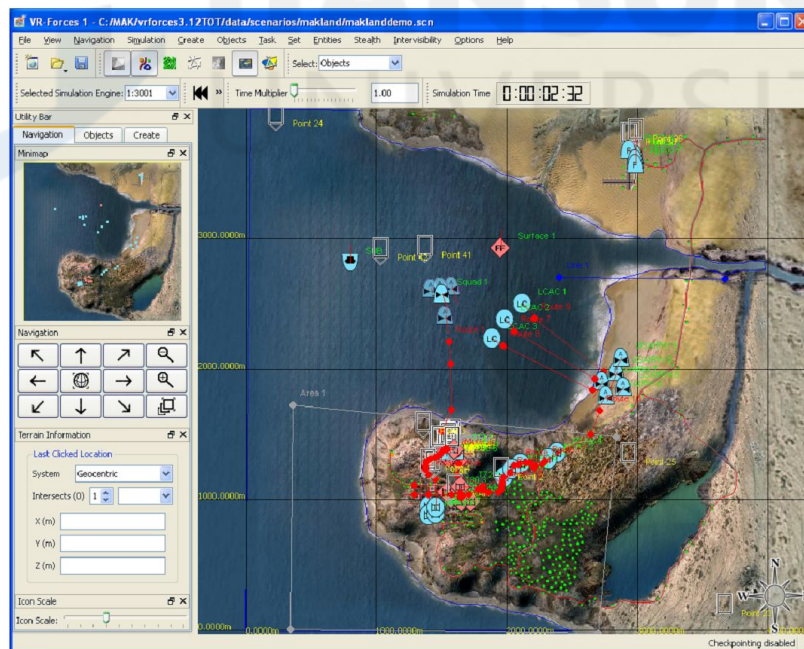
OneSAF의 모델링 인프라스트럭처는 블랙보드(blackboard)를 통하여 퍼지(fuzzy) 형태의 팩트(fact)를 사용하는 퍼지 추론 엔진을 제공한다. OneSAF에서는 대부분 '반응 행위(reactive behavior)'에 대하여 팩트를 사용하는데, 복합행위에서 반응 규칙(reaction rule)을 적용한다. 예를 들어 '전술적 이동'이라는 복합행위에는 '직접사격 반응 행위'를 적용할 수 있다. 인터럽트라고 부르는 반응 행위는 '반응 팩트(reaction fact)'가 참일 때 발생한다. 행위 수행 중 반응 팩트가 참이 되면 현재 행위를 중단하고 반응 행위가 수행된다. 그러나 반응 행위가 종료되면 인터럽트된 행위는 재수행하지 않는다.

2. VR-Forces

미국 VT-MAK에서 개발한 VR-Forces는 전술 지휘 훈련, 위협 또는 가상 군 모의를 위한 시뮬레이션 도구이다. VR-Forces를 사용하여 모의 개체를 배치하고, 계획을 수립하는 등의 시나리오를 작성하며, 시뮬레이션을 수행한다. 시뮬레이션이 수행되면 차량 및 병사 개체는 지형과의 충돌 처리, 도로 이동, 장애물 회피, 통신장비 모의, 적군 탐지 및 교전을 수행할 수 있다.

VR-Forces의 특징은 다음과 같다. 첫째, VR-Forces는 시뮬레이션 엔진과 통제용 사용자 인터페이스를 여러 개로 분리하여 운영할 수 있는 분산 아키텍처를 지원하여 협업 시나리오 생성 및 훈련을 할 수 있다. 둘째, VR-Forces Toolkit을 사용하여 어플리케이션을 커스터마이징 하거나 새로운 어플리케이션에 VR-Forces의 기능을 통합할 수 있다. 셋째, VR-Forces는 조립 아키텍처를 지원하여 개체 이동 동역학, 행위, 전술, 피해평가 모델, 센서, 방호, 무장 등을 추가/변경할 수 있다.

VR-Forces의 사용자 인터페이스는 아래 <그림 2-2>와 같다.¹¹⁾



<그림 2-2> VR-Forces GUI 화면

11) 『VR-Forces Users Guide』, VT MAK, 2009

VR-Forces에서 모의되는 개체는 지상 개체, 항공기 개체, 해상/수중 개체, 사람 등의 독립적인 플랫폼 단위이며, 이 개체들을 통합(aggregate)하여 부대 단위로도 모의할 수 있다. 또한 개체의 속도, 위치, 연료, 탄약 등의 상태를 변경할 수 있으며, 물리모델을 통하여 다른 개체 또는 탐지 표적에 대하여 회피, 표적 선택, 무장/탄약 선택 등 반응 행위를 수행할 수 있다.

모의 개체의 임무 계획은 과업(tasks), 요청데이터, 조건문으로 구성된다. 개체의 과업은 ‘특정 물체로 이동’, ‘경로 순찰’, ‘두 물체 간 순찰’, ‘개체 팔로잉’, ‘유효 사격’, ‘이륙 및 착륙’ 등을 포함한다. 조건문은 특정 상태가 참 또는 거짓일 때 수행되는 과업을 명시하는 것으로, 이 문장에는 조건을 한 번 평가하여 참일 때 수행하는 ‘If’문, 주기적으로 조건을 평가하여 참이 되면 수행하는 ‘When’문, 조건이 참일 동안 반복적으로 수행하는 ‘When’ 문이 있다. VR-Forces에서 제공하는 조건들은 아래 [표 2-1]과 같다. 또한 조건문은 TRUE, FALSE와 같은 부울(boolean) 연산자와 AND, OR, NOT과 같은 논리(logical) 연산자를 사용할 수 있다.

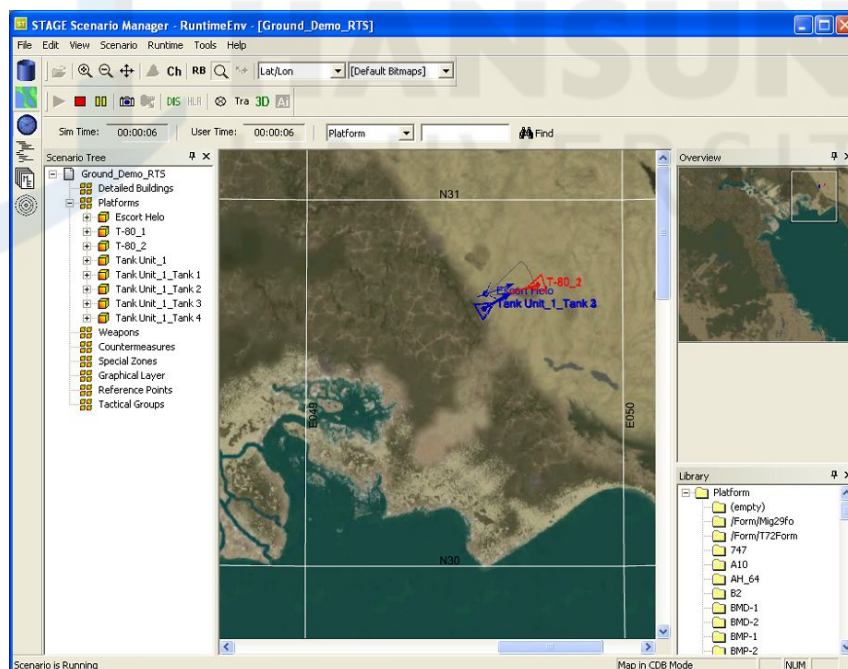
[표 2-1] VR-Forces가 제공하는 조건 목록

조건(Condition)	내 용 (Content)
EndInArea	개체가 주어진 영역 내에 있는가?
EntLeftOfLine	개체가 단계선(Phase Line) 왼쪽에 있는가?
EntDestroyed	개체가 파괴되었는가?
EntUnderFire	개체가 사격 중인가?
EntHasTarget	개체가 위치한 표적을 가지고 있는가?
EntEmbarked	개체의 탑승 상태 변화가 있는가?
EntAltitude	개체가 특정 고도 아래 또는 위에 있는가?
EntDetected	개체가 다른 개체를 탐지하였는가?
ReceiveTextMessage	개체가 무선기 메시지를 수신하였는가?
SimTime	시뮬레이션 시간이 특정 시간 이전인가?(이후인가?)

3. STAGE

미국 PRESAGIS에서 개발한 STAGE는 전술 DB를 구축하고 실시간 전술 운영 환경을 모의하는 소프트웨어이다. 모의 시나리오에는 탐지, 통신, 교전 등을 사용하여 모의 개체간 상호작용을 하는 독립적인 플랫폼(항공기, 배, 트럭, 전차/장갑차 등)을 포함한다. 플랫폼에는 총, 포, 미사일 등의 무장을 탑재할 수 있다.

STAGE는 기본적으로 제공하는 라이브러리를 확장할 수 있는 유연성을 제공한다. 사용되는 컴포넌트를 생성하고, 컴포넌트의 파라미터를 정의하고 값을 추가할 수 있으며, STAGE의 편집기, 툴 등의 기능을 확장할 수 있다. 또한, STAGE를 독립적으로도 운용할 수 있지만, DIS¹²⁾, HLA¹³⁾ 등의 표준 연동 프로토콜을 통하여 다른 소프트웨어와 통합하여 사용할 수 있다. 아래 <그림 2-3>은 STAGE의 실행 화면을 나타낸 것이다.¹⁴⁾



<그림 2-3> STAGE 화면

12) DIS(Distributed Interactive Simulation) : 다양한 유형의 시뮬레이션 모형과 시뮬레이터를 하나의 동일한 가상전장 환경에 연결하기 위한 대화형 분산 시뮬레이션 구조

13) HLA(High Level Architecture) : 시뮬레이션 간의 상호 연동을 위한 공통의 프로토콜과 명세를 규정하는 기술

14) 『STAGE 6.0 User's Guide』, PRESAGIS, 2009, p44

STAGE는 기 정의된 행위 스크립트 언어를 사용하여 실시간으로 개체(플랫폼 또는 무기)를 제어한다. 행위 스크립트 언어는 전술적인 환경의 특성을 평가하는 조건(condition)문과 플랫폼의 행위에서 원하는 변화를 기술하는 행동(action)문, 그리고 조건문과 행동문을 연결하는 제어(control)문으로 구성된다. 제어문은 트리거문, 선택문, 반복문, 변수 등을 사용할 수 있다.

예를 들어, 항공기 플랫폼의 행위를 스크립트로 작성하면 아래 <그림 2-4>와 같다. 아래 스크립트의 내용은, 레이더를 사용하여 충분히 가까운 위협을 탐지하고, 탐지된 위협이 항공기(고정익 및 회전익 항공기)이면 공대공 미사일을 발사하고, 항공기가 아니면 공대지 미사일을 발사하는 행위 스크립트이다.¹⁵⁾

```
INITIALIZATION_SECTION
    int flag = 0;
    system.activate(radar_sensors);
REACTION_CONTROL_SECTION
    # 플랫폼 탐지
    track.cycle_on(detected_platforms);
    while (track.next() > 0) do
        # 15km 이내의 표적 선택
        if ((track.range <= 15000.0) and (track.ident = hostile)) then
            opponent.assign(track.index);
            if (flag == 0) then
                # 표적이 항공기이면 공대공 미사일 발사, 아니면 공대지 미사일 발사
                if ((track.type = fixed_wing) or (track.type = helicopter)) then
                    weapon.launch(tracking_missile, "air_to_air");
                else
                    weapon.launch(tracking_missile, "air_to_ground");
                endif;
            endif;
            flag = 1;
        endif;
    endwhile;
END_SCRIPT
```

<그림 2-4> STAGE 의 행위 스크립트의 예

STAGE는 전술 환경에서 개별무기체계 또는 팀 운영을 훈련하고 평가하기 위하여, 또한 시뮬레이션 환경에서 전술/운영 시나리오 및 시스템 분석을 위하여, 그리고 대규모 개체 모의를 통한 실 체계의 프로토타입을 목적으로 사용할 수 있다.

15) 전계서, p465

제 3 절 인공지능 기술

SAF에서 사용되는 인공지능 분야는 크게 추론(reasoning), 계획(planning), 협업(cooperation), 학습(learning) 등이 있다. 주어진 사실이나 규칙으로부터 인지된 입력에 대하여 결론을 얻는 과정을 추론이라고 하며, 계획은 의사결정의 한 형태로써, 목적을 달성하기 위하여 행동의 순서를 결정하는 과정을 의미한다. 두 개 이상의 개체들이 하나의 목적을 달성하기 위해 역할을 분담하는 과정을 협업이라고 하고, 학습은 사실과 규칙을 계속적인 과정에 의해 습득하는 일련의 과정을 의미한다.

본 논문에서는 SAF 개체의 계획 분야에서 사용되는 전통적인 기법인 유한 상태 기계, 최근에 계획 분야에서 주로 사용되는 STRIPS 기반 계획, 그리고 HTN 기반 계획 기법에 대해 소개하고, 추론 분야에서 주로 사용되지만, 행위의 시작 및 종료 조건 표현 방식으로 사용되는 퍼지 추론에 대해 설명하였다.

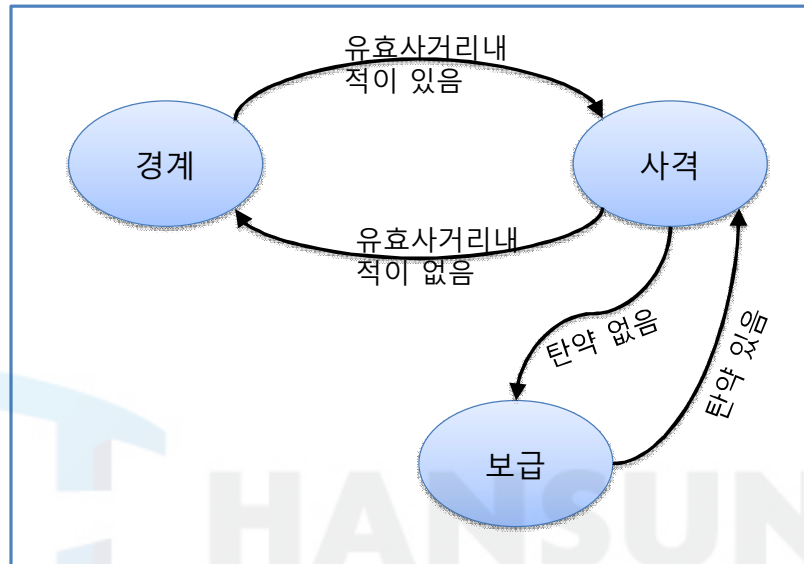
1. 유한 상태 기계(Finite State Machine)

일반 상용 게임에서 주로 사용되는 인공지능 기술인 유한 상태 기계(Finite State Machine, FSM)는 유한한 개수의 상태(state)들로 구성된 기계를 말한다. 이 FSM의 알고리즘 자체는 간단하지만, 대부분의 시스템들이 유한한 상태들로 표현 가능하기 때문에 시스템을 제어하는 알고리즘으로 가장 많이 사용된다.¹⁶⁾

FSM은 하나의 입력을 받아 현재 상태에서 다른 상태로 전이(transition)한다. 모의 개체가 몇 개의 상태로 주어질 때 개체의 현재 상태에 따라 외부에 대처하는 방식이 결정된다. 즉, 외부의 상황이 변화하게 되면 현재 상태도 FSM에 의해 다른 상태로 전이하게 된다. 또한, 개체는 특정한 상태에 있을 경우 항상 같은 방식으로 행동하게 된다. 예를 들어 아래 <그림 2-5>에서 보는 바와 같이, 모의 개체가 ‘경계’ 상태에서 유효사거리 내 적을 발견하면 ‘사

16) 이상용, 『인공지능의 세계』, 서울 : 21세기사, 2008, p438~441

격' 상태로 이동하며, 발견된 적이 있는 동안에는 계속 '사격' 상태로 남아 있게 된다. 만약 이 상태에서 발견된 적이 없어진다면 다시 '경계' 상태로 바뀌며, 적이 없는 동안 계속 '경계' 상태로 남아 있게 된다. 또한 '사격' 상태에서 탄약이 없으면 '보급' 상태로 바뀌고, 탄약이 보급되면 다시 '사격' 상태로 돌아간다.



<그림 2-5> 유한 상태 기계(FSM)의 예

결과적으로 FSM의 전반적인 아이디어는 단순하고, 그 구현 방법도 단순히 switch 구문 혹은 if 구문을 사용하여 프로그래밍 되는 것이 전부이지만, 그 활용 범위와 용도는 상당히 다양하다. 그래서 FSM은 특별한 인공지능 기능을 요구하지 않는 게임에서 많이 사용한다. FSM을 사용한 게임 내의 개체는 그 행동의 예측이 쉬워 게임의 재미가 떨어지기 때문에, 게임의 난이도를 높이기 위해 FSM을 확장하거나 계층적 FSM으로 설계하기도 한다. 하지만 FSM은 복잡하고 많은 수의 상태를 표현할 경우, 상태 다이어그램을 정리하기가 어렵고, 상태 변화를 가능하게 하는 외부 자극 처리가 복잡하게 된다.¹⁷⁾

17) 장장훈, 『유전자 알고리즘과 신경망을 이용한 MMORPG의 지능캐릭터 구현에 관한 연구』, 동명정보대학교 컴퓨터공학과 석사학위논문, 2008

2. 퍼지 추론(Fuzzy Inference)

퍼지 이론(fuzzy theory)은 1965년 Lotfi. A. Zadeh 교수가 학술전문지 'Information and Control'에 발표한 'Fuzzy sets' 논문에서 처음으로 소개되었고, 이 논문에서 Zadeh 교수는 '아름다운 여성의 집합', '키가 큰 사람의 집합' 등 경계가 명확하지 않은 집합을 퍼지집합이라 하였으며 인간사고의 의미정보의 전달 중 추상화라는 부분에 중요한 역할을 부여하여 그것을 수학적 이론으로써 전개하였다.¹⁸⁾

퍼지 이론은 부정확하고, 불확실하고, 신뢰할 수 없는 지식을 표현하고 추론하기 위한 다양한 개념과 기법들로 구성되어 있다. 이것은 대략적 혹은 주관적인 가치와 모호하고 불완전한 데이터를 사용하는 규칙을 만들어 낼 수 있다. 또한 논리를 부정확하게 표현함으로써, 기존의 IF-THEN 규칙보다 실제로 인간이 생각하는 방식에 더 가깝다.¹⁹⁾

행위의 규칙으로 사용될 수 있는 것 중에 '적이 많으면 신속히 이동하라'라는 규칙에서 '적이 많다'라는 용어가 정확히 몇 명 이상을 의미하는 것인지, '신속히'라는 용어가 어느 정도의 속도인지 불분명하다. 퍼지 집합에서는 이를 '소속도(degree of membership)'라고 표현한다. 예를 들어, '적이 많다'에 대한 소속도는 0명의 경우 '0', 5명의 경우 '0.2', 10명의 경우 '0.5', 20명의 경우 '0.8' 등으로 나타낼 수 있다. 이러한 소속도를 함수로 나타낸 것이 멤버십 함수(membership function)이다. 멤버십 함수는 어떠한 퍼지 변수(χ)가 어떠한 집합 A에 속하는 정도인 소속도($\mu_A(\chi)$)를 나타내는 함수으로써, 소속도가 1에 가까우면 χ 가 A에 속하는 정도가 높다는 것이며, 반대로 0에 가까우면 낮다는 것을 나타낸다.

이미 습득한 지식이나 경험을 바탕으로 새로운 지식을 얻는 과정을 '추론(inference)'이라 한다. 퍼지 추론은 몇 개의 퍼지 명제로부터 하나의 다른 근사적인 퍼지 명제를 유도하는 근사 추론(approximate reasoning)방식으로, 여러 조건들에 따라 행동을 결정짓는 전진 추론 방식인 GMP (Generalized Modus Ponens)와, 이미 결론을 안 상태에서 조건을 찾아내는 후진 추론 방

18) 김평원, 『퍼지 논리를 활용한 논증 텍스트 분석 모형 연구』, 국어교육학연구 41권 0호, p339-378, 2011

19) 조영임, 『인공지능시스템』, 서울 : 홍릉과학출판사, 2012, p.270

식인 GMT(Generalized Modus Tolens)로 구분된다. 아래 <그림 2-6>은 퍼지 추론 유형인 GMP와 GMT에 대한 일반적인 형태와 그에 따른 예를 나타낸다.

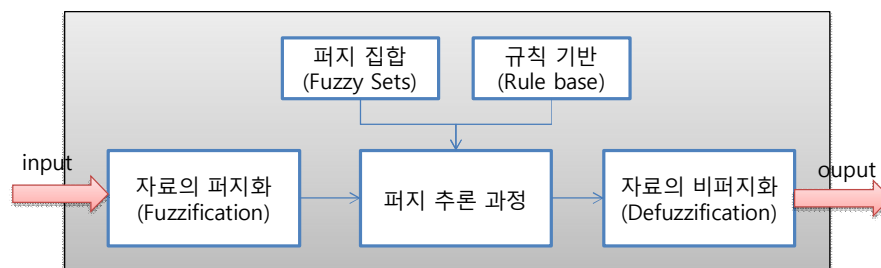
GMP(Generalized Modus Ponens)		
[전제]	x is A'	적이 매우 많다
[조건]	IF x is A THEN y is B	적이 많으면 신속히 이동한다.
[결론]	y is B'	매우 신속히 이동한다.

GMT(Generalized Modus Tolens)		
[전제]	y is B'	신속히 이동하지 않는다.
[조건]	IF x is A THEN y is B	적이 많으면 신속히 이동한다.
[결론]	x is A'	적이 많지 않다.

<그림 2-6> 퍼지 추론 유형

퍼지 추론의 'If A Then B'에서 A항은 조건항(antecedent part), B항은 결과항(consequent part)라 하며, 이는 결국 조건항의 멤버십 함수($\mu_A(x)$)와 결과항의 멤버십 함수($\mu_B(y)$)를 연결시켜 주는 역할을 하게 된다. 이렇게 언어로 표현된 A항은 멤버십 함수를 통해 퍼지화(수치화, fuzzification)되고, 퍼지 추론을 통해 B값으로 도출되며, 도출된 결과는 비퍼지화(defuzzification)과정을 거쳐 수치화 되어 결과값으로 나오게 된다.

퍼지 추론 시스템의 과정은 아래 <그림 2-7>과 같다. 그림 우측 상단의 '규칙 기반(Rule base)'은 퍼지 규칙을 포함하며, 좌측 상단의 '퍼지 집합(Fuzzy Sets)'은 퍼지 집합을 멤버십 함수로 표현해 놓은 것을 말한다.²⁰⁾



<그림 2-7> 퍼지 추론 시스템의 과정

20) 정영근, 『퍼지 추론을 이용한 최단 경로 탐색 알고리즘의 개발』, 서울대학교 대학원 지구환경시스템공학부 석사학위 논문, 2006, p.4-7

3. STRIPS 기반 계획(STRIPS based planning)

인공지능에서의 계획(planning)은 의사결정의 한 형태로써, 목적을 달성하기 위하여 행동의 순서를 결정하는 과정을 의미한다. 또한 계획은 인간 지능(human intelligent)의 필수 요소이며, 가상군의 자율성과 유연성을 증대시키는 필수 능력이다. 계획 기법(planning technique)은 주어진 목표에 도달하기 위해 가능한 계획 공간을 효율적으로 탐색하기 위한 기법을 의미한다. 잘 알려진 계획 접근법은 상태-기반(state-based) 계획 기법인 STRIPS²¹⁾ 기반 계획(STRIPS based planning)과 과업-기반(task-based) 계획 기법인 HTN 기반 계획 기법(Hierarchical Task Network based planning)이다.

STRIPS 기반 계획 기법은 Nils J.Nilsson 과 Richard Fikes 가 개발한 최초의 계획시스템이다. STRIPS의 기본 원칙은 주어진 최초의 상태를 변형시켜 목표로 하는 상태를 만드는 과정이 ‘참’이 되도록 오퍼레이터(operator)의 순서를 찾아내려고 시도하는 문제 해결방법이다. 이것은 초기 상태에서 최종 상태까지 올바른 상태 변화를 생성하기 위해 점진적으로 행동을 추가한다.

STRIPS 기반 계획의 예를 들면, 한 공간에서 지면의 A위치에 원숭이가 있고, C위치에 박스가 있다. 원숭이는 B위치의 천장에 매달려 있는 바나나를 가지고 싶지만, 박스를 B위치로 옮기고 박스 위로 올라가야 한다. 이 예제를 위한 최초 상태, 목표 상태, 행동에 대한 절차를 표시하면, 아래 <그림 2-8>과 같다. 결국 STRIPS 기반 계획에 따라, 최초 상태에서부터 목표 상태에 이르기까지 행동의 사전조건, 사후조건을 비교하여 최종적으로 Move(A, C), MoveBox(C, B), ClimbUp(B), TakeBananas(B) 순으로 행동 순서를 찾게 된다.²²⁾

21) STRIPS : STanford Research Institute Problem Solver

22) <http://en.wikipedia.org/wiki/STRIPS>

최초 상태 : At(A), Level(low), BoxAt(C), BananasAt(B)
목적 상태 : Have(Bananas)

액션 :

```
// X에서 Y로 이동
_Move(X, Y)_
시작조건 : At(X), Level(low)
종료조건 : not At(X), At(Y)

// 박스 위로 올라감
_ClimbUp(Location)_
시작조건 : At(Location), BoxAt(Location), Level(low)
종료조건 : Level(high), not Level(low)

// 박스 아래로 내려감
_ClimbDown(Location)_
시작조건 : At(Location), BoxAt(Location), Level(high)
종료조건 : Level(low), not Level(high)

// X에서 Y로 원숭이와 박스를 이동
_MoveBox(X, Y)_
시작조건 : At(X), BoxAt(X), Level(low)
종료조건 : BoxAt(Y), not BoxAt(X), At(Y), not At(X)

// 바나나 획득
_TakeBananas(Location)
시작조건 : At(Location), BananasAt(Location), Level(high)
종료조건 : Have(Bananas)
```

<그림 2-8> STRIPS 기반 계획의 예

목표 지향 행위 계획(Goal Oriented Action Planning, GOAP)은 주어진 계획을 수행하는 개체가 현재 상태에서 목표 상태로 전이하기 위해 필요한 행동들을 스스로 찾게 해 주는 방법으로써, 탐색영역을 줄이기 위한 목표 상태에서 부터 현재 상태로 역으로 찾는 방법이다. 일반적으로 이 계획방법은 개체의 행동을 계획하기 위하여 개체와 주변 상황을 상태 기반으로 모델링할 때 많이 사용된다. 예를 들어, ‘적을 사살하라’라는 목표가 주어진 경우, ‘적 사살’ 상태를 만들기 위한 이전상태 ‘사격’을 찾고 ‘사격’ 상태를 만들기 위한 이전상태 ‘무장탄약장전’ 상태를 찾고, ‘무장탄약장전’ 상태를 만들기 위한 이전상태 ‘무장 선택’을 찾는 방법이다. 각 상태 전이에 필요한 행위가 같이 기술되어 있기 때문에, 이렇게 찾은 행동을 역으로 계획하면 해당 목표를 달성하는데 필요한 행동들을 찾을 수 있다.²³⁾

23) J.Orkin, 『Applying Goal-Oriented Action Planning to Games in AI Game Programming Wisdom 2』, Charles River Media, 2003, p.217~228

이러한 목표 지향 행위 계획의 장점은 사람의 행동과 유사하여 모델링 하기 쉬우며, 계획 모듈을 분리하여 유지보수 및 재사용이 가능하다. 또한 개체의 행위를 하드 코딩하는 것이 아니라 단지 목표를 할당하고 SAF 개체가 스스로 찾아가게 하는 것이기 때문에 상황과 환경에 동적으로 적응할 수 있고, 모델 작성자가 기대하지 않았던 최적의 행동도 찾을 수 있다. 그러나 이 방법은 코드가 복잡하여 디버깅이 어렵고, SAF 개체가 많은 행동과 자유도를 갖는 경우, 특별한 계획을 수행하거나 계획을 반복하게 하는 등의 정교한 제어가 어렵다. 또한 계획을 수행하는데 많은 시간이 소요되어 짧은 반응 시간을 요하는 상용 게임에서는 문제가 될 수 있다.

4. HTN 기반 계획(Hierarchical Task Network based planning)

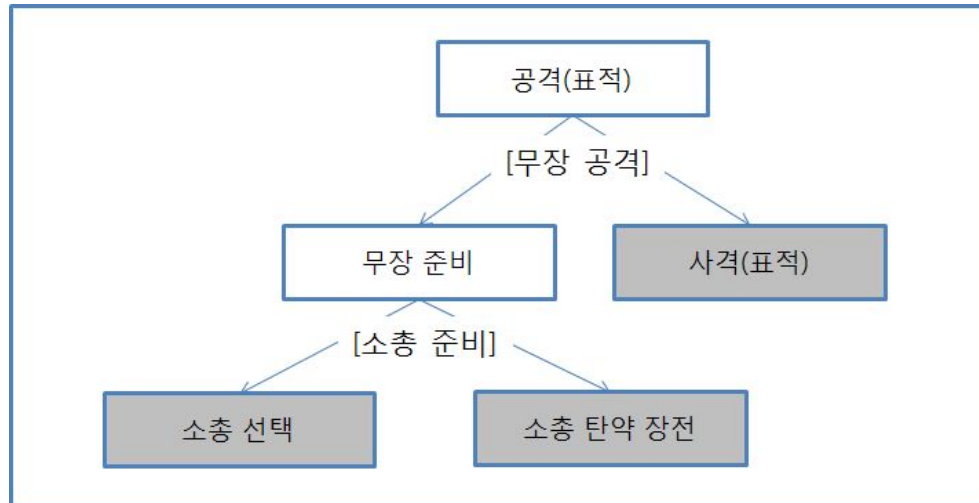
HTN(Hierarchical Task Network)는 STRIPS에서의 초기 상태, 최종 상태, 그리고 오퍼레이터 등의 개념들은 유사하지만, 계층적인 과업을 사용한다는 점이 다르다. 과업들을 계층적으로 구성하면서 계획을 수립하는 이런 개념들이 HTN의 핵심이며, 과업들을 세분화(decomposition)하는 과정이 규칙에 의하여ダイナ믹하게 진행된다. STRIPS와의 가장 큰 차이점은, 확실한 목적 상태에 도달하기 위한 것이 아니라, 과업 집합을 형성하기 위한 계획을 수립하는 것이다.

HTN에서의 단위 과업(primitive task)은 더 이상 세분화가 필요 없는 실행이 가능한 행동을 말하며, 비 단위 과업(non-primitive task)은 세분화가 필요한 과업을 말한다. 비 단위 과업을 세분화하는 과정을 HTN에서는 메소드(method)라고 하며, 이것은 주어진 문제를 축소(reduction)하는 과정이라고 할 수 있다. HTN에서의 계층화는 결국 문제를 분해하여 축소하고자 하는 방법이다. STRIPS와 같은 계획 방법은 상태와 목표 상태, 그리고 상태 전이를 위한 오퍼레이터만이 주어졌다면, HTN에서는 상태 전이의 과정을 과업으로 연계하고, 과업을 세분화할 수 있는 방법을 제공하는 점이 특징이다.²⁴⁾

예를 들어, 아래 <그림 2-9>와 같이 ‘공격(표적)’ 과업은 ‘무장공격’이라는

24) Glenn Wissing, 『Multi-agent planning using HTN and GOAP』, Luica University of Technology, Bachelor's Thesis, 2007, p.3~4

메소드에 따라서 ‘무장 준비’와 ‘사격(표적)’으로 세분화 할 수 있고, ‘무장 준비’ 과업은 ‘소총 준비’ 메소드에 따라 ‘소총 선택’과 ‘소총 탄약 장전’ 과업으로 세분화 할 수 있다.



<그림 2-9> HTN 기반 계획의 예

STRIPS 기반 계획과 HTN 기반 계획의 결정적 차이점은, HTN 기반 계획은 직접적으로 어떤 행동을 실행할 지를 결정하기 보다, 어떤 전략을 선택하고 어떻게 달성하기 위한 결정에 초점을 맞추고 있다. STRIPS 기반 계획 방법으로도 전략을 구현할 수는 있으나, 과업간 상호 관계의 조합으로 HTN 기반 계획 방법이 훨씬 더 자연스럽다.

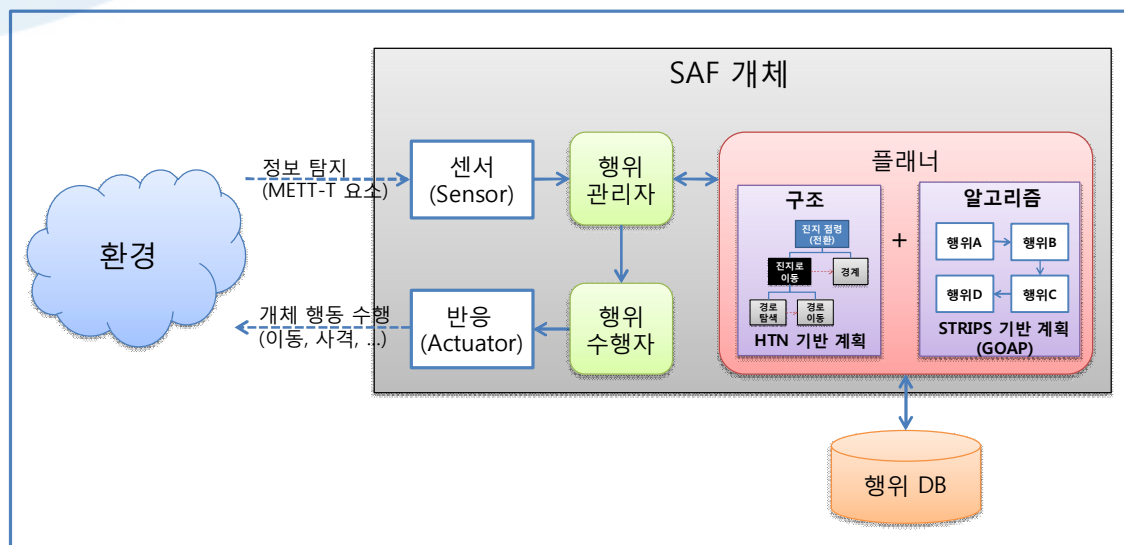
제 3 장 행위 자동 계획 기법 설계

제 1 절 행위 자동 계획 기법 개념

OneSAF 또는 VR-Forces와 같은 기존 SAF 모의 도구에서 사용되는 규칙 기반 추론(rule-based inference) 기법은 이미 구축된 규칙에 대해서만 반응하기 때문에, 새로운 환경에 적응적으로 대처가 가능한 계획(planning) 기법을 사용한다.

본 논문에서 제안하는 행위의 계획 기법은 앞 장에서 기술한 STRIPS 기반 계획과 HTN 기반 계획 기법을 혼합하여 적용하였다. SAF 개체가 수행하는 임무를 행위 별로 계층적으로 구성하기 위한 행위 구조는 HTN과 유사한 방법으로 과업 데이터베이스를 구성하였고, 행위의 최종 목적을 달성하기 위하여 필요한 행위 목록들을 자동으로 생성하는 행위 계획 수립 기능은 STRIPS 기반 계획의 일종인 목표 지향 행위 계획(Goal Oriented Action Planning, GOAP) 기법을 사용하였다.

SAF의 행위 자동 계획 기법에 대한 개념도는 아래 <그림 3-1>과 같다.



<그림 3-1> SAF의 행위 자동 계획 기법 개념도

모의 개체는 센서 모델에서 계획 판단을 위한 정보를 획득한다. 정보는 전술적인 상황조치 고려 요소인 METT-T를 고려한다. 임무(Mission), 적상황(Enemy), 가용시간(Time), 지형 및 기상(Terrain) 그리고 가용병력(Troops) 요소에 대한 정보를 수집한다. 예를 들어, 현재 임무가 무엇인지, 탐지된 적이 많은지, 가용시간이 충분한지, 현재 위치한 곳이 산악지형인지, 가용무기 및 탄약이 충분한지 등에 대한 정보를 획득한다.

행위 관리자는 현재 수집된 정보를 바탕으로 행위의 계획 여부를 판단한다. 계획이 필요하다고 판단되면, 플래너에게 계획 수립 요청을 하고, 그 결과를 행위 수행자에게 전달한다.

행위 수행자는 행위 관리자가 생성한 새로운 계획을 수행한다. 이것은 계획에 있는 행위들을 순차적으로 실행함으로서, 이동, 사격 등의 개체 행동을 수행한다.

플래너는 최종 목적 상태를 달성하기 위한 최적의 행위 목록을 생성한다. 가용한 행위를 행위 DB에서 검색하고, 현재의 상태와 최종적으로 달성해야 하는 상태를 이용하여 필요한 행위들을 자동으로 생성한다. 즉, SAF의 행위 자동 계획 기법은 이 플래너에 구현된다.

플래너를 구현하기 위한 설계 방법으로 먼저 계층적인 행위 구조를 설계하였다. 행위를 계층적으로 표현하기 위하여 디자인 패턴 중 조합 패턴(composite pattern)을 사용하였고, 행위의 시작/종료 조건을 정의하는 방법으로 퍼지 표현(fuzzy expression)방법을 적용하였다. 다음으로, 행위 계획 기능에 대해 설계하였다. 목표 상태를 만족하기 위한 행위를 역순으로 찾기 위하여, 현재 행위의 시작 조건과 대안 행위의 종료 조건을 매칭하는 방법을 제안하였고, 최적의 행위 목록을 탐색하는 방법으로 길찾기에서 주로 사용되는 에이스타(A*) 알고리즘²⁵⁾을 사용하였다.

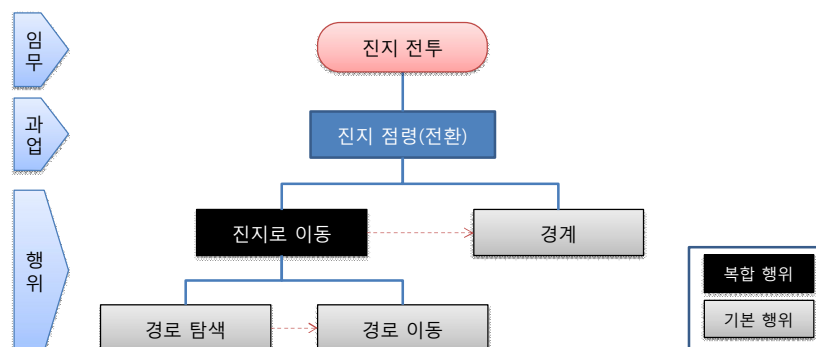
25) 에이스타(A*) 알고리즘 : 주어진 출발지 노드에서 목적지 노드까지 가는 최단경로를 찾아내는 그래프/트리 탐색 알고리즘

제 2 절 계층적 행위 설계

1. 계층적 행위 정의

SAF 개체가 수행하는 임무(mission)는 전술적인 목적에 따라 여러 개의 과업(task)으로 분류할 수 있다. 과업은 전술적 목적 달성을 위한 개체 또는 부대의 행위로 정의할 수 있으며, 행위(behavior)는 부대 또는 개인의 행동들이 조합된 일련의 활동을 말한다. 특히 행위는 여러 개의 복합적인 행위로 구성할 수 있다. 이것은 기존의 자동 계획 기법에서 주로 사용되는 계층적 과업 네트워크(Hierarchical Task Network, HTN)와 유사한 구조를 갖는다.

행위는 계층적으로 구성할 수 있는 구조를 가진다. ‘경로 이동’, ‘경계’, ‘사격’ 등과 같이 더 이상 하위 행위로 분리할 수 없는 기본 행위(primitive behavior)와, 하위 행위들을 순차적으로 구성한 복합 행위(composite behavior)로 분류할 수 있다. 즉, 주어진 임무를 수행하기 위하여 필요한 여러 과업이 기술되고, 과업을 수행하는 데 필요한 행위들을 계층적으로 구성한다. 아래 <그림 3-2>는 임무-과업-행위 구성의 예를 나타낸 그림이다, 그림에서 방어 작전 시나리오에 필요한 ‘진지 전투’ 임무 중 ‘진지 점령(전환)’ 과업은 ‘진지로 이동’ 행위와 ‘경계’ 행위로 구성되며, ‘진지로 이동’ 행위는 다시 ‘경로 탐색’ 행위와 ‘경로 이동’ 행위로 구성된다. 여기에서, 하위 행위로 분리할 수 있는 ‘진지로 이동’ 행위는 복합 행위이며, 하위 행위로 분리할 수 없는 ‘경로 탐색’, ‘경로 이동’, ‘경계’ 행위는 기본 행위가 된다.



<그림 3-2> 임무-과업-행위 구성의 예

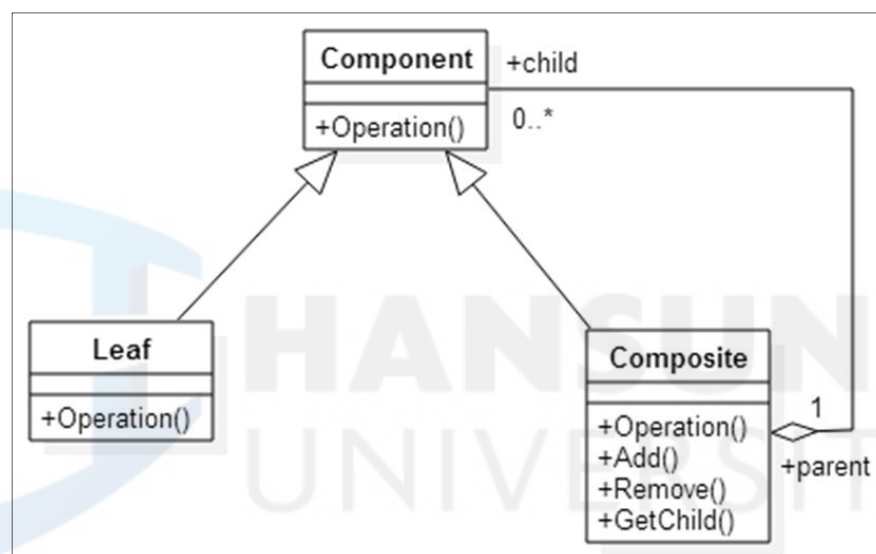
일반적으로 개체가 수행하는 임무를 과업, 복합 행위, 기본 행위로 세분화할 때, 전술적인 목적 및 모델링 상세도에 따라 세분화 기준이 달라질 수 있다. 하나의 개체의 활동을 정의하더라도 어떤 이는 그것을 과업으로 정의할 수 있고, 또 다른 이는 복합 행위 또는 기본 행위로 정의할 수도 있다. 즉, 임무의 세분화 기준은 모델링 전문가의 주관적인 견해에 따라 다르게 정의된다. 그러나 본 논문에서 다루는 행위의 자동 계획 기법은 이러한 임무의 세분화 기준에 따른 영향이 크지 않다. 단지 SAF 개체의 임무를 정의할 때, 좀더 복잡하고 다양하게 행위들을 조합하여 표현할 수 있는 계층적인 구조를 사용한다는 것이 중요하다. 아래 [표 3-1]은 계층적 행위 관계의 예를 나타낸 것이다.

[표 3-1] 계층적 행위 관계의 예

임무(Mission)	과업(Task)	행위(Behavior)
기동	행정적 이동	전투대형 결정 → 경로 탐색 → 행군 이동
	전술적 이동	전투대형 결정 → 경로 탐색 → 경로 이동
	공격간 포탄낙하시 조치	우회 이동 (전투대형 결정 → 우회경로 탐색 → 경로 이동)
		경계
	지뢰지대 개척	POMINS 발사
돌파 및 돌격	돌격 및 소탕	통로 통과 → 돌격 → 경계
진지 전투	진지 점령	진지로 이동 (경로 탐색 → 경로 이동) → 경계
집결	집결지 행동	경계
사격에 의한 공격	대응사격	사격
	엄호/지원 사격	급편진지 점령 (진지로 이동 → 경계) → 고착 사격
경계 활동	순찰	경로 탐색 → 정찰
	진지 경계	경계
철수	계획된 철수	전투대형 결정 → 경로 탐색 → 철수 이동
	퇴각	퇴각 경로 탐색 → 퇴각 이동

2. 행위 구조

작은 행위들을 조합하여 큰 행위로 구성하기 위한 행위의 큰 구조는 소프트웨어의 디자인 패턴 중에서 조합패턴(composite pattern)을 사용한다. 조합패턴은 객체들의 관계를 트리 구조로 구성하여 부분-전체 계층을 표현하는 소프트웨어 디자인 패턴으로, 사용자가 단일 객체와 복합 객체 모두 동일하게 다루도록 하는 설계 형태이다. 조합패턴의 클래스 다이어그램은 아래 <그림 3-3>과 같다.²⁶⁾



<그림 3-3> 조합패턴의 클래스 다이어그램

위 그림에서 Leaf 클래스와 Composite 클래스는 모두 Component 클래스를 상속받으며, Composite 클래스는 Component, 즉 Leaf 또는 하위 Composite 클래스를 멤버로 가질 수 있다. 즉, 기본 행위는 Leaf 클래스로, 복합 행위는 Composite 클래스로 설계하였다.

기본 행위는 더 이상 분리될 수 없는 단위이며, 모의 개체에게 행동을 수행하게 하는 최하위 명령의 단위이기 때문에, C++과 같은 프로그래밍 언어의 소스코드 형태로 구현하며, 이것들을 조합 가능한 형태로 만들기 위하여

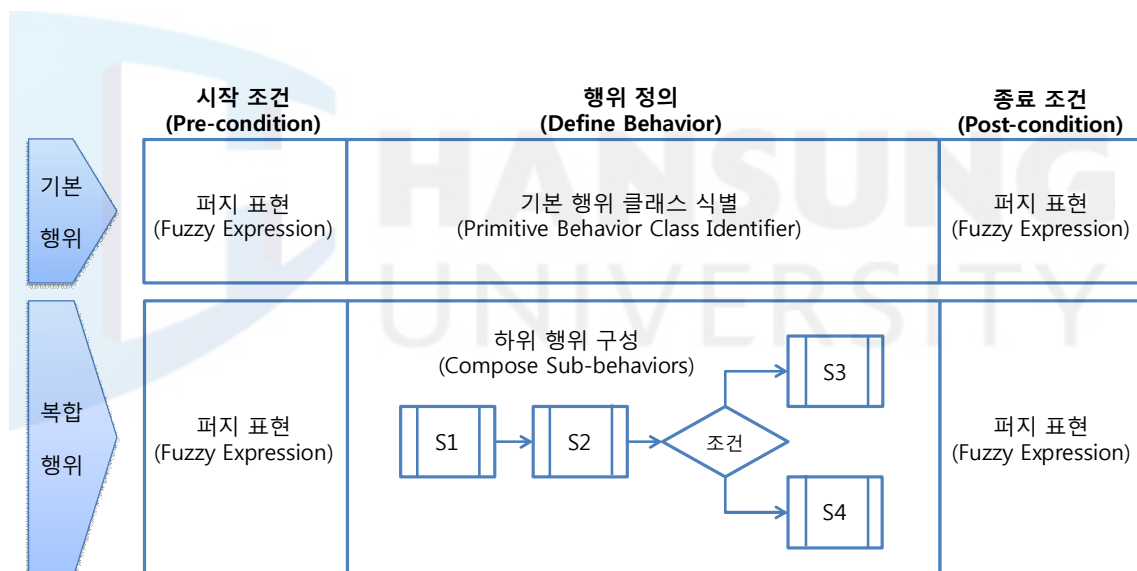
26) http://ko.wikipedia.org/wiki/컴포지트_패턴

XML 형태의 메타 데이터를 생성한다. 기본 행위의 메타 데이터에는 소스코드로 구현한 기본 행위 클래스를 식별할 수 있는 ID를 가지고 있다.

복합 행위는 기본 행위 또는 하위 복합 행위의 조합 정보를 XML 형태의 메타 데이터로 표현한다. 이것은 순차적인 하위 행위 정보(하위 행위의 메타 데이터 목록)와 행위 간 연결 정보를 가지고 있다.

각각의 행위들은 시작 조건과 종료 조건을 포함한다. 시작 조건은 해당 행위를 수행하기 위한 필수 조건을 의미하고, 종료 조건은 그 행위가 종료될 수 있는 조건들로 구성한다. 이러한 행위의 조건들은 ‘적이 많다’ 또는 ‘위험하다’ 등과 같이 모호하게 표현된 교범의 내용을 그대로 반영하기 위하여, 추론 기술에서 주로 사용되고 있는 퍼지 표현(fuzzy expression)을 사용한다.

아래 <그림 3-4>는 행위 구조를 표현한 그림이다.



<그림 3-4> 행위 구조도

복합 행위에서 하위 행위들의 구성은 XML 형태로 표현한다. 하위 행위 목록 들은 <behavior> 엘리먼트로 조건 분기문은 <branch> 엘리먼트로 표현한다. 그리고 각각의 하위 행위들 간의 연결 관계는 <link> 엘리먼트로 표현한다. 위 <그림 3-4>의 복합 행위의 하위 행위 구성을 XML 형식으로 표현하면 아래 [표 3-2]와 같다.

[표 3-2] XML 형태의 하위 행위의 구성의 예

하위 행위 구성	행위간 링크 구성
<pre> <sub-behavior> <behavior id="1" name="S1" configuration=S1.xml" /> <behavior id="2" name="S2" configuration=S2.xml" /> <branch id="3"> <condition> </condition> </branch> <behavior id="4" name="S3" configuration=S3.xml" /> <behavior id="5" name="S4" configuration=S4.xml" /> </sub-behavior> </pre>	<pre> <links> <link prev="Start" next="1" /> <link prev="1" next="2" /> <link prev="2" next="3" /> <link prev="3" next="4" condition="true" /> <link prev="3" next="5" condition="false" /> </links> </pre>

시작 조건 및 종료 조건에서 사용되는 퍼지 표현은 퍼지 로직(fuzzy logic) 연산자와 퍼지 팩트(fuzzy fact)를 사용하는 표현 방식이다. 퍼지 로직 연산자는 ‘AND’, ‘OR’과 같은 이항 연산자와 ‘NOT’의 일항 연산자를 사용한다. 퍼지 팩트는 ‘x is A’로 표현하며, 여기서 x는 퍼지 변수(fuzzy variables), A는 퍼지 집합(fuzzy sets)을 나타낸다. 여기에서는 퍼지 표현을 XML 노드 형태로 표현하였다. ‘적이 많다’, ‘적이 많고, 적과의 거리가 가깝다’, ‘직사화기를 보유하고, 탄약이 충분하다’ 등과 같은 조건은 아래 [표 3-3]과 같이 XML 형태의 퍼지 표현을 사용한다.

[표 3-3] 퍼지 표현의 예

조건 예	XML 노드 형태의 퍼지 표현
적이 많다	<code><fuzzy-fact variable="Enemy" value="Many"/></code>
적이 많고 적과의 거리가 가깝다	<pre> <and> <fuzzy-fact variable="Enemy" value="Many"/> <fuzzy-fact variable="EnemyDistance" value="Close"/> </and> </pre>
직사화기를 보유하고 탄약이 충분하다	<pre> <and> <fuzzy-fact variable="DirectWeapon" value="Have"/> <fuzzy-fact variable="Ammo" value="Enough"/> </and> </pre>
연료가 부족하거나 탄약이 부족하다	<pre> <or> <fuzzy-fact variable="Fuel" value="Insufficient"/> <fuzzy-fact variable="Ammo" value="Enough"/> </or> </pre>
목적지까지의 거리가 멀지 않다	<pre> <not> <fuzzy-fact variable="DistanceToDestination" value="Far"/> </not> </pre>

위 [표 3-2]에서 사용된 퍼지 변수인 ‘적의 수(Enemy)’, ‘적과의 거리(EnemyDistance)’ 등은 아래 <그림 3-5>와 같이 멤버십 함수의 집합으로 정의한다.

퍼지 변수(Fuzzy Variable)	퍼지 변수의 멤버십 함수(Membership Functions)
적의 수(Enemy)	The graph shows three membership functions for the variable '적의 수(Enemy)'. The x-axis ranges from 0 to 30. '적다(Few)' starts at 1.0 at x=0 and decreases linearly to 0.0 at x=10. '보통(Middle)' starts at 0.0 at x=5, increases linearly to 1.0 at x=15, and then decreases linearly to 0.0 at x=25. '많다(Many)' starts at 0.0 at x=20 and increases linearly to 1.0 at x=30.
적과의 거리 (EnemyDistance)	The graph shows three membership functions for the variable '적과의 거리 (EnemyDistance)'. The x-axis ranges from 0 to 300. '가깝다(Close)' starts at 1.0 at x=0 and decreases linearly to 0.0 at x=50. '보통(Middle)' starts at 0.0 at x=50, increases linearly to 1.0 at x=150, and then decreases linearly to 0.0 at x=250. '멀다(Far)' starts at 0.0 at x=200 and increases linearly to 1.0 at x=300.
보유 직사화기 (DirectWeapon)	The graph shows two membership functions for the variable '보유 직사화기 (DirectWeapon)'. The x-axis ranges from 0 to 5. '없음(None)' starts at 1.0 at x=0 and decreases linearly to 0.0 at x=1. '보유(Have)' starts at 0.0 at x=1 and increases linearly to 1.0 at x=5.
보유탄약 (Ammo)	The graph shows two membership functions for the variable '보유탄약 (Ammo)'. The x-axis ranges from 0 to 300. '부족(Insufficient)' starts at 1.0 at x=0 and decreases linearly to 0.0 at x=100. '충분(Enough)' starts at 0.0 at x=100 and increases linearly to 1.0 at x=150, remaining at 1.0 until x=300.
연료 (Fuel)	The graph shows two membership functions for the variable '연료 (Fuel)'. The x-axis ranges from 0 to 100. '부족(Insufficient)' starts at 1.0 at x=0 and decreases linearly to 0.0 at x=40. '충분(Enough)' starts at 0.0 at x=10 and increases linearly to 1.0 at x=30, remaining at 1.0 until x=100.
목적지까지 거리 (DistanceToDestination)	The graph shows five membership functions for the variable '목적지까지 거리 (DistanceToDestination)'. The x-axis ranges from 0 to 500. '매우가깝다(VeryClose)' starts at 1.0 at x=0 and decreases linearly to 0.0 at x=50. '가깝다(Close)' starts at 0.0 at x=50, increases linearly to 1.0 at x=100, and then decreases linearly to 0.0 at x=150. '보통(Middle)' starts at 0.0 at x=100, increases linearly to 1.0 at x=200, and then decreases linearly to 0.0 at x=300. '멀다(Far)' starts at 0.0 at x=250 and increases linearly to 1.0 at x=350, then decreases linearly to 0.0 at x=450. '매우멀다(VeryFar)' starts at 0.0 at x=400 and increases linearly to 1.0 at x=500.

<그림 3-5> 퍼지 변수의 정의

위 그림에서 ‘적의 수’는 멤버십 함수 ‘적다’(0~11), ‘보통’(4~26), ‘많다’(19~30)로 구성되며, 각각의 멤버십 함수는, 위 그림의 ‘적의 수’ 함수 그래프와 같이 정의할 수 있다. 다른 퍼지 변수들도 위 그림에서 제시한 그래프와 같이 정의하며, 이렇게 정의한 퍼지 변수들은 시작 및 종료 조건의 퍼지 논리 표현식에 사용된다.

시작 및 종료 조건에 사용되는 논리 표현식은 일반적으로 도메인 전문가가 교리의 내용을 바탕으로 직접 작성한다. 이는 컴퓨터가 표현식을 평가하거나 다른 표현식과 비교하는 데 사용하기에는 그 식이 중복되거나 논리적으로 복잡하게 작성되어 지는 경우가 많다. 전문가가 작성한 표현식을 평가 또는 비교 연산에 그대로 사용한다면 부정확하거나 불필요한 연산을 할 수 있다. 그렇기 때문에, 논리 표현식을 평가하거나 비교 연산을 수행하기 전에 컴퓨터가 연산하기 쉬운 형태로 정규화 하고, 불필요한 표현식을 제거하는 최소화 과정이 필요하다. 논리 표현식의 정규화를 위하여 논리합 정규형(Disjunctive Normal Form, DNF)²⁷⁾ 알고리즘을 사용하여 논리곱의 논리합 형태(AND의 OR 형태)로 정규화 하여 연산의 형태를 일반화 하였다. 또한, 논리식의 복잡도를 효율적으로 감소시키기 위하여 에스프레소 휴리스틱 논리 최소화(Espresso heuristic logic minimizer)²⁸⁾ 알고리즘을 사용하여 중복되거나 불필요한 표현식을 제거하였다. 논리합 정규형 알고리즘과 에스프레소 알고리즘은 본 논문의 범위를 벗어나기 때문에 알고리즘의 세부 내용을 생략하였다.

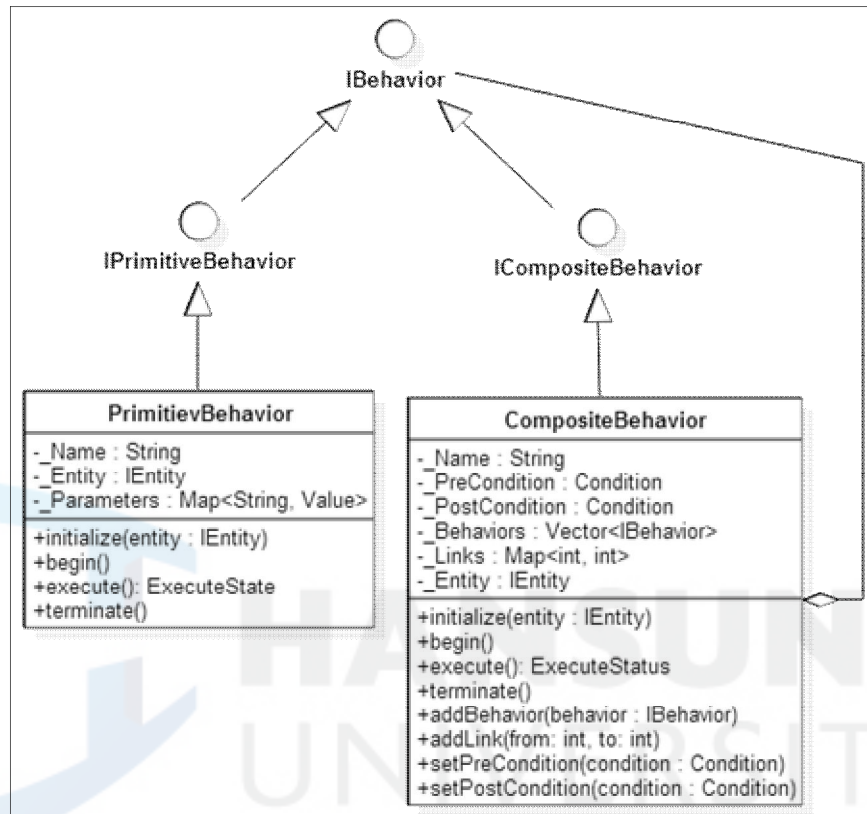
3. 클래스(Class) 다이어그램

행위는 계층적인 구조를 갖기 위해서 소프트웨어 디자인 패턴 중에서 조합 패턴을 사용하여 클래스를 설계하였다. 아래 <그림 3-6>에서 보는 바와 같이, 인터페이스 클래스는 상위 인터페이스인 'IBehavior'와 상위 인터페이스를 상속받은 'IPrimitiveBehavior'와 'ICompositeBehavior' 인터페이스가 있다. 그리고 두 인터페이스의 구현 클래스는 'PrimitiveBehavior'와 'CompositeBehavior'가 있으며, CompositeBehavior는 IBehavior를 자식 노드로 가질 수

27) http://en.wikipedia.org/wiki/Disjunctive_normal_form

28) http://en.wikipedia.org/wiki/Esspresso_heuristic_logic_minimizer

있다. 기본 행위인 PrimitiveBehavior 클래스는 행위 이름, 파라미터, 행위수행상태 등의 속성이 있으며, 복합 행위인 'CompositeBehavior'는 행위 이름, 시작 조건, 종료 조건, 하위행위목록 등의 속성을 가지고 있다.



<그림 3-6> 행위 클래스 다이어그램

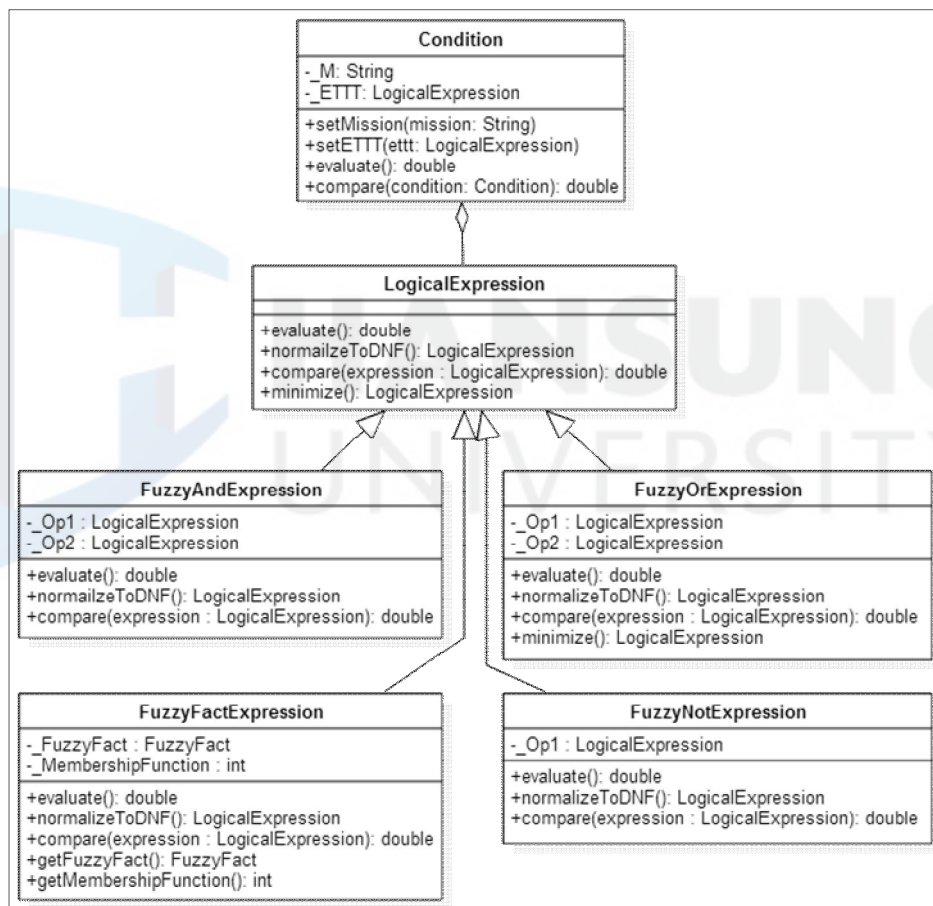
기본 행위는 행위 수행 시작 및 종료 시, 그리고 행위 수행 동안 개체가 할 일을 처리할 수 있도록 구조를 가져야 한다. 대부분의 기본 행위들은 개체가 동작을 수행할 수 있도록 개체에게 트리거를 발생시키는 역할을 한다. 복합 행위는 기본 행위와 유사한 구조를 갖지만, 복합 행위가 포함하고 있는 하위 행위들 설정, 하위 행위들 간의 순서 관계 설정, 그리고 시작 및 종료 조건을 설정하고 관리하는 기능이 추가된다. 기본 행위와 복합 행위 클래스의 속성 및 오퍼레이션 설명은 아래 [표 3-4]와 같다.

[표 3-4] 기본 행위 및 복합 행위 클래스 속성과 오퍼레이션

클래스	속성 또는 오퍼레이션	설명
Primitive Behavior	_Name: String	행위 이름
	_Entity: IEntity	할당된 행위 수행 개체
	_Parameters: Map<String, Value>	행위의 파라미터
	initialize(entity: IEntity)	초기화 함수
	begin()	행위 시작시 호출되는 함수
	execute(): ExecuteState	행위 수행동안 호출되는 함수 (행위 수행 상태를 리턴)
	terminate()	행위 종료시 호출되는 함수
Composite Behavior	_Name: String	행위 이름
	_Entity: IEntity	할당된 행위 수행 개체
	_PreCondition: Condition	시작 조건
	_PostCondition: Condition	종료 조건
	_Behaviors: Vector<IBehavior>	하위 행위 목록
	_Links: Map<int, int>	하위 행위들간의 연결관계
	initialize(entity:IEntity)	초기화 함수
	begin()	행위 시작시 호출되는 함수
	execute(): ExecuteState	행위 수행동안 호출되는 함수 (행위 수행 상태를 리턴)
	terminate()	행위 종료시 호출되는 함수
	addBehavior(behavior:IBehavior)	하위 행위를 추가하는 함수
	addLink(from:int, to:int)	하위 행위들간의 연결관계를 추가하는 함수
	setPreCondition(condition:Condition)	행위의 시작 조건을 설정하는 함수
	setPostCondition(condition:Condition)	행위의 종료 조건을 설정하는 함수

시작 조건 및 종료 조건의 형태인 Condition 클래스는 전술적인 상황조치 고려 요소인 METT-T²⁹⁾에 대한 조건을 입력할 수 있어야 한다. 특히, 임무를 제외한 적 상황, 가용시간, 지형 및 기상, 그리고 가용병력에 대한 요소는 퍼지 논리 표현식으로 정의한다. 퍼지 논리 표현식은 AND, OR, NOT과 같은 논리 연산자와 퍼지 팩트를 사용하여 논리 표현식을 구성하며, 퍼지 논리 표현식을 평가하고, 표현식끼리 비교할 수 있는 연산자를 제공한다.

아래 <그림 3-7>은 행위의 시작 조건 및 종료 조건에서 사용되는 퍼지 논리 표현식에 대한 클래스 다이어그램이다.



<그림 3-7> 퍼지 논리 표현식 클래스 다이어그램

위 그림에서, 시작 및 종료조건 형태인 Condition 클래스는 문자열 형태의 M(임무)과 퍼지 논리 표현식 형태의 ETTT(적 상황, 가용시간, 지형 및 기상,

29) METT-T : 임무(Mission), 적상황(Enemy), 가용시간(Time), 지형 및 기상(Terrain), 가용병력(Troops)

가용병력)를 갖는다. 또한, 퍼지 논리 표현식을 평가하고, 다른 Condition과의 비교를 수행할 수 있는 함수를 정의한다. 논리 표현식인 LogicalExpression 클래스는 AND, OR, NOT, 그리고 퍼지 팩트의 상위 클래스로써, 표현식을 평가하는 함수, 표현식을 정규화 하는 함수, 표현식끼리 비교하는 함수를 정의한다. FuzzyAndExpression, FuzzyOrExpression, FuzzyNotExpression, FuzzyFact Expression 클래스는 논리 표현식으로 사용할 수 있는 퍼지 팩트와 논리 연산자인 AND, OR, NOT을 정의한 클래스이다.

퍼지 논리 표현식 클래스의 속성 및 오퍼레이션 설명은 아래 [표 3-5]와 같다.

[표 3-5] 퍼지 논리 표현식 클래스 속성과 오퍼레이션

클래스	속성 또는 오퍼레이션	설명
Condition	_M: String	임무
	_ETTT: LogicalExpression	적 상황, 가용시간, 지형 및 기상, 가용병력을 표현한 논리 표현식
	setMission(mission:String)	임무를 설정하는 함수
	setETTT(ett:LogicalExpression)	ETTT 논리 표현식을 설정하는 함수
	evaluate(): double	조건문을 평가하는 함수
	compare(c:Condition):double	조건문간 비교하는 함수
Logical Expression	evaluate(): double	논리 표현식을 평가하는 함수
	normalizeToDNF():LogicalExpression	DNF로 정규화 하는 함수
	compare(le:LogicalExpression):double	논리 표현식간 비교하는 함수
	minimize():LogicalExpression	논리 표현식을 최소화하는 함수
FuzzyAnd Expression	_Op1: LogicalExpression	AND 연산자의 좌항
	_Op2: LogicalExpression	AND 연산자의 우항
	evaluate(): double	AND 표현식을 평가하는 함수
	normalizeToDNF():LogicalExpression	AND 표현식을 DNF로 정규화 하는 함수
	compare(le:LogicalExpression):double	논리 표현식간 비교하는 함수

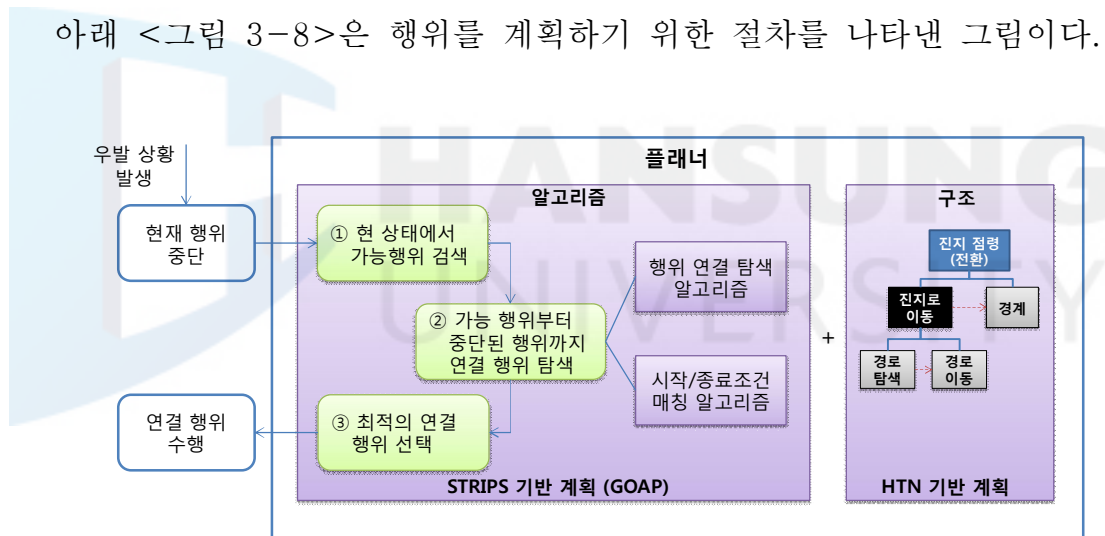
FuzzyOr Expression	_Op1: LogicalExpression	OR 연산자의 좌항
	_Op2: LogicalExpression	OR 연산자의 우항
	evaluate(): double	OR 표현식을 평가하는 함수
	normalizeToDNF():LogicalExpression	OR 표현식을 DNF로 정규화 하는 함수
	compare(le:LogicalExpression):double	논리 표현식간 비교하는 함수
	minimize():LogicalExpression	OR 표현식을 최소화하는 함수
FuzzyNot Expression	_Op1: LogicalExpression	NOT 연산자의 단항
	evaluate(): double	NOT 표현식을 평가하는 함수
	normalizeToDNF():LogicalExpression	NOT 표현식을 DNF로 정규화 하는 함수
	compare(le:LogicalExpression):double	논리 표현식간 비교하는 함수
FuzzyFact Expression	_FuzzyFact: FuzzyFact	퍼지 팩트
	_MembershipFunction: int	멤버십 함수
	evaluate(): double	퍼지 팩트 표현식을 평가하는 함수
	normalizeToDNF():LogicalExpression	퍼지 팩트 표현식을 DNF로 정규화 하는 함수
	compare(le:LogicalExpression):double	논리 표현식간 비교하는 함수
	getFuzzyFact(): FuzzyFact	퍼지 팩트를 리턴하는 함수
	getMembershipFunction(): int	멤버십 함수를 리턴하는 함수

이상과 같이, 계층적인 행위 구조를 설계하기 위하여 조합 패턴을 사용하여 기본 행위와 복합 행위로 구분하여 설계하였고, 각 행위는 퍼지 논리 표현식으로 정의한 시작 조건 및 종료 조건을 사용하였다. 다음 절에서는 이러한 행위 구조를 사용하여 행위 계획을 위한 절차 및 방법에 대해 기술하였다.

제 3 절 행위 계획 기능 설계

앞 장에서 설명하였듯이, 행위를 계획하는 것은 주어진 목적을 달성하기 위하여 행위의 순서를 결정하는 것이고, 계획 기법은 가능한 계획 공간을 효율적으로 탐색하기 위한 기법을 말한다. STRIPS 기반 계획 기법은 주어진 최초의 상태를 변형시켜 목표로 하는 상태를 만드는 과정이 ‘참’이 되도록 오퍼레이터(operator)의 순서를 찾아내려고 시도하는 문제 해결방법이다. 본 논문에서 제시하는 행위 계획 방법은 STRIPS 기반 계획 기법을 사용하여 행위의 순서를 찾아내는 방법을 제시한다. 특히, 현재 상태에서부터 목표 상태로 전이하기 위한 필요한 행동들을 역으로 찾는 목표 지향 행위 계획(Goal Oriented Action Planning, GOAP) 방법을 사용하여 목표 상태를 달성하기 위한 연결 행위를 탐색한다.

아래 <그림 3-8>은 행위를 계획하기 위한 절차를 나타낸 그림이다.



<그림 3-8> 행위 계획 절차

SAF 개체가 시나리오에 입력된 행위 수행 중, 우발 상황이 발생하여 더 이상 행위 수행이 불가능한 경우, 현재 행위를 중단하고 플래너에서 계획 수립 과정이 시작된다. 행위 계획은 다음과 같은 절차로 진행된다. 먼저, 중단된 현재 상태에서 즉시 수행 가능한(가용한) 행위를 검색한다. 다음으로, 검색된 가능 행위로부터 중단된 행위까지 연결 행위를 탐색한다. 이 때, 행위 탐색시 역순으로 행위를 연결하기 위한 탐색 알고리즘과 시작/종료조건 매칭 알고리

즘이 사용된다. 마지막으로, 탐색된 연결 행위들 중 가장 적합한 연결 행위를 최종적으로 선택한다. 선택된 연결 행위가 현 상태에서 가장 적합한 계획이 된다.

1. 가용 행위 검색

행위 계획을 위하여 가장 먼저 해야 할 일은 현 상황에서 가용한 모든 행위를 검색하는 것이다. 즉, 개체가 수행 가능한 모든 행위들 중에서 시작 조건을 만족하는 행위들을 선별하는 과정을 수행한다. 시작 조건을 만족하는지를 판단하기 위해서 퍼지 논리 표현식을 평가해야 한다.

위 절에서 설명한 것처럼, 퍼지 논리 표현식은 AND, OR, NOT과 같은 논리 연산자와 퍼지 팩트를 사용하여 표현했기 때문에, 각 논리 연산자와 퍼지 팩트를 평가하는 방법은 아래 [표 3-6]과 같다.

[표 3-6] 퍼지 논리 표현식 평가 방법

논리 표현식 연산자	평가 방법 (수식)	설명
퍼지 팩트	fuzzify(MF, value)	퍼지 팩트 함수를 적용한 값 (MF : 퍼지 팩트 함수, value : 데이터)
AND 연산자	$A \cap B = \text{Min}(A, B)$	두 개의 평가값 중 최소값 (A, B : 논리 표현식)
OR 연산자	$A \cup B = \text{Max}(A, B)$	두 개의 평가값 중 최대값 (A, B : 논리 표현식)
NOT 연산자	$\bar{A} = 1 - A$	1에서 평가값을 뺀 값 (A : 논리 표현식)

위 표에서 설명한 것과 같이, 퍼지 팩트는 입력 값에 대한 퍼지 멤버십 함수를 적용한 값으로 평가한다. AND 연산자는 두 개의 평가 값 중 최소값이며, OR 연산자는 그 중 최대값이다. 그리고 NOT 연산자는 평가값의 역수 ($1-A$)로 계산한다. 평가 결과는 0에서 1 사이의 실수 값으로 표현되며, 이때, 0은 그 표현식이 완전 거짓을, 1은 완전 참을 의미한다.

조건문의 만족 여부는 퍼지 논리 표현식의 평가 값으로 결정되는데, 만족 여부를 결정하는 기준 값을 얼마로 할지를 판단하는 문턱 값(threshold value)이 주어져야 한다. 이것은 퍼지 팩트 함수의 구간을 정의하는 것과 마찬가지로 문턱 값을 결정하기 위하여 개발된 일반적인 방법은 없으며, 현재는 위게임 모델을 개발하는 전문가의 경험에 의존하여 결정한 후, 실험을 통하여 수정하는 방법을 사용한다.³⁰⁾ 본 논문에서는 문턱 값을 결정하기 위하여 도메인 전문가의 의견을 수집하였고, 전문가들은 논리 표현식의 평가 결과 50% 이상이면 해당 조건문을 만족하는 것으로 판단하였기 때문에 0.5를 문턱 값으로 사용하였다.

2. 연결 행위 탐색

앞 절에서 설명한 가용 행위 검색 방법을 통해 현 상황에서 수행 가능한 행위들을 검색하면, 가용 행위에서부터 우발 상황 발생에 따라 중단된 행위까지의 중간 연결 행위를 탐색한다. 이 때 연결 행위를 탐색하는 데 사용되는 알고리즘은 통상적으로 길 찾기에 주로 사용되는 에이스타(A*) 알고리즘을 사용한다.

에이스타 알고리즘은 주어진 출발 노드에서부터 목표 노드까지 가는 최단 경로를 찾아내는 그래프/트리 탐색 알고리즘 중 하나이다. 이 알고리즘은 각 노드 x 에 대해 그 노드를 통과하는 최상의 경로를 추정하는 순위 값인 "휴리스틱 추정값" $h(x)$ 을 매기는 방법을 쓴다. 이 알고리즘은 이 휴리스틱 추정값의 순서로 노드를 방문한다.³¹⁾

에이스타 알고리즘은 출발 노드로부터 목표 노드까지의 최적경로를 탐색하기 위한 것이다. 이를 위해서는 각각의 노드에 대한 평가함수를 정의해야 한다. 이를 위한 평가함수 $f(n)$ 은 다음과 같다.

$$f(n) = g(n) + h(n)$$

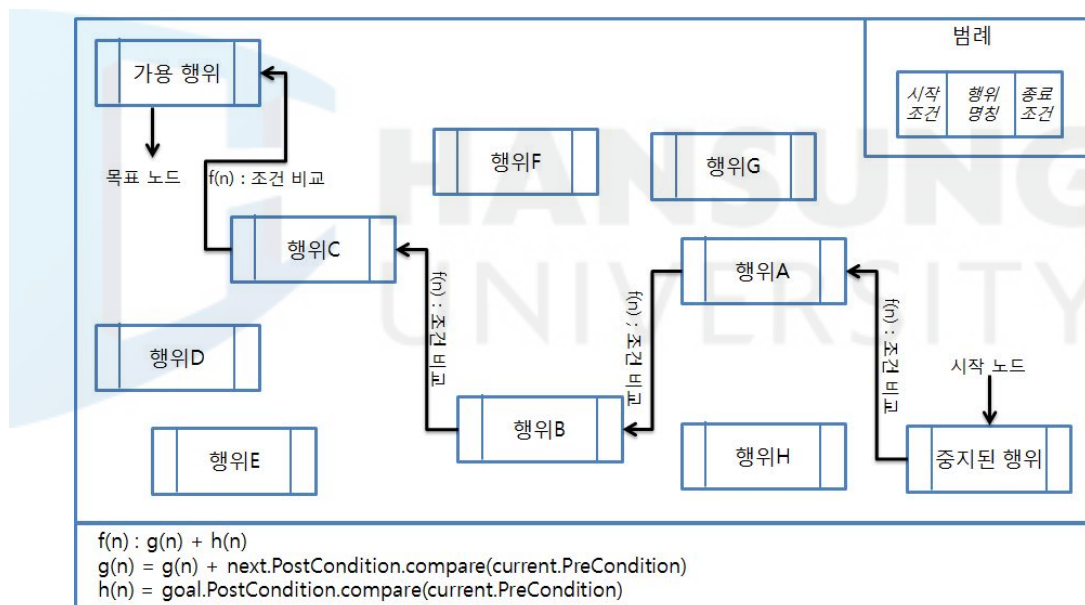
이 때, $\left\{ \begin{array}{l} g(n) = \text{출발노드로부터 노드 } n \text{까지의 경로 비용} \\ h(n) = \text{노드 } n \text{으로부터 목표노드까지의 추정 경로 비용} \end{array} \right\}$

30) 최대회의 2명, 『위게임에서 가상군의 목표지향행위계획 및 재 계획 방법』, KIIS Fall Conference 2012 Vol. 22, No. 2, 2012.

31) http://ko.wikipedia.org/wiki/A*_알고리즘

연결 행위를 탐색하는 데 에이스타 알고리즘을 적용하기 위하여 노드 n 은 행위로, 경로 비용은 행위의 시작조건과 다른 행위의 종료조건 비교값으로 적용하였다. 이 때, 에이스타 알고리즘의 검색 방향은 행위의 탐색 영역을 줄이기 위하여, 목표 지향 행위 계획 방법과 같이 목표 노드에서부터 시작 노드까지 역순 방향으로 하였다.

에이스타 알고리즘의 시작 노드는 우발 상황에 따라 중단된 행위로, 목표 노드는 가용 행위 검색을 통해 검색된 가용 노드로 사용하고, 노드 평가함수 $g(n)$ 은 시작 행위에서부터 현재 행위의 시작 조건과 다음 행위의 종료 조건의 비교값의 누적으로, $h(n)$ 은 현재 행위의 시작 조건과 목표 행위의 종료 조건의 비교값으로 정의하였다. 아래 <그림 3-9>는 연결 행위 탐색의 예를 나타낸 그림이다.



<그림 3-9> 연결 행위 탐색의 예

위 그림과 같이 목표 노드인 중지된 행위에서 부터 시작 노드인 가용 행위까지 역으로 검색한 결과, 중간 행위 노드인 '행위A', '행위B', '행위C' 순으로 검색되며, 결국, '가용 행위', '행위C', '행위B', '행위A', '중지된 행위' 순으로 행위 계획이 수립된다. 이 때 사용되는 조건 매칭 방법은 다음 절의 퍼지 표현식 비교 알고리즘으로 사용하였다.

3. 시작/종료조건 매칭

퍼지 표현식으로 기술된 행위의 시작 조건과 종료 조건을 비교하여 연결 행위를 찾는다. 이 때, 조건 매칭을 위해 퍼지 표현식의 비교 알고리즘을 사용한다. 조건문은 퍼지 팩트와 AND, OR, NOT과 같은 논리 표현식으로 정의하였기 때문에, 조건문간의 비교 방법은 두 개의 퍼지 표현식의 적합 확률(matching rate)을 구하는 과정을 의미한다.

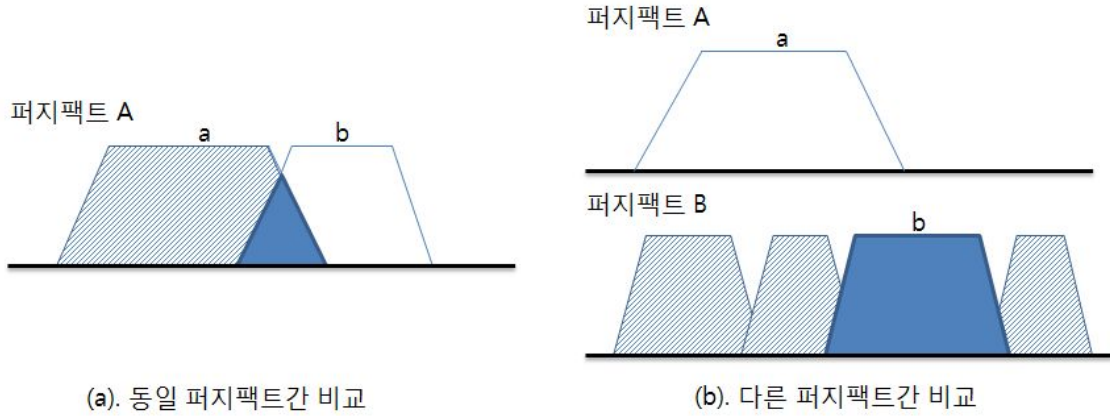
두 개의 퍼지 표현식 E1, E2이 있을 때, 표현식 E1이 표현식 E2와 얼마나 적합한가를 수식으로 표현하면 아래 수식과 같다. 여기에서 E1을 비교 연산자의 좌항, E2를 우항이라고 부르며, 비교 결과는 0에서 1 사이의 값으로, 완전 적합이면 1, 부적합이면 0을 의미한다.

$$0 \leq (E1)compare(E2) \leq 1$$

퍼지 표현식의 비교는 첫째, 두 퍼지 팩트간 비교, 둘째, AND 표현식과의 비교, 셋째, OR 표현식과의 비교, 마지막으로 NOT 표현식과의 비교로 정의한다.

(가) 두 퍼지 팩트간 비교

두 개의 퍼지 팩트간의 비교 방법은 <그림 3-10>과 같이 동일 퍼지 팩트간의 비교 방법(a)과 다른 퍼지 팩트와의 비교 방법(b)이 있다. 동일 퍼지 팩트간의 비교 방법은 좌항의 퍼지 팩트 함수 면적에 대한 좌우항의 퍼지 팩트 함수의 중복면적의 비율로 정의한다. 그러나 서로 다른 퍼지 팩트간의 비교는 직접적인 비교가 불가능하다. 그러므로 좌항의 퍼지 팩트 값을 비교 대상에서 제외하고 우항의 퍼지 팩트 값이 참일 확률을 의미하며, 결국, 우항 퍼지 팩트의 모든 퍼지 팩트 함수 면적에 대한 해당 퍼지 팩트 함수 면적의 비율로 정의한다.



<그림 3-10> 두 퍼지 팩트간 비교 연산 그림

아래 수식 (a)는 동일 퍼지 팩트간의 비교 수식을 나타내며, 수식 (b)는 다른 퍼지 팩트간의 비교 수식을 나타낸다. 여기에서 ‘A’, ‘B’는 퍼지집합(예: 적의 수, 목적지까지 거리 등)를 의미하며, ‘a’, ‘b’는 퍼지 팩트 함수(예: 적 다, 많다, 멀다, 등)를 나타낸다. 그리고 ‘ $A \rightarrow a$ ’는 퍼지 팩트(예: 적의 수가 많다)를 의미한다.

$$(a). (A \rightarrow a) \text{ compare } (A \rightarrow b) = \frac{a, b \text{의 중복 면적}}{a \text{의 면적}}$$

$$(b). (A \rightarrow a) \text{ compare } (B \rightarrow b) = \frac{b \text{의 면적}}{B \text{의 전체 면적}}$$

위 수식 (b)와 같이 서로 다른 퍼지 팩트간 비교값은, 결국 우향의 퍼지 팩트($B \rightarrow b$)가 참이 될 확률($p(B \rightarrow b)$)과 동일하며, 아래 수식으로 표현할 수 있다. 이 확률 수식은 아래의 AND 및 OR 표현식과의 비교 수식에서 사용된다.

$$(a). p(A \rightarrow a) = \frac{a \text{의 멤버쉽 함수 면적}}{A \text{의 모든 멤버쉽 함수 면적}}$$

$$(b). p(A \cap B) = p(A) \times p(B)$$

$$(c). p(A \cup B) = p(A) + p(B)$$

위 수식과 같이 퍼지 팩트의 진실 확률(a)은 퍼지집합의 모든 퍼지 팩트 함수 면적에 대한 해당 퍼지 팩트 함수 면적의 비율로 정의하며, AND 표현식에 대한 진실 확률(b)은 각 항에 대한 진실 확률의 곱으로, OR 표현식에 대한 진실 확률(c)은 각 항에 대한 진실 확률의 합으로 정의한다.

(나) AND 표현식과의 비교

아래 수식은 비교 연산자의 좌항이 AND 표현식인 상황에서 우항이 AND 표현식, OR 표현식, 퍼지 팩트인 경우의 비교값을 산출하기 위한 수식을 나타낸 것이다.

$$(a). (E1 \cap E2)compare (E3 \cap E4) = ((E1 \cap E2)compare E3) \times ((E1 \cap E2)compare E4)$$

$$(b). (E1 \cap E2)compare (E3 \cup E4) = ((E1 \cap E2)compare E3) + ((E1 \cap E2)compare E4)$$

$$(c). (E1 \cap E2)compare (A \rightarrow a) = \begin{cases} E1compare (A \rightarrow a) & (\text{if } E1 \text{ has } A) \\ E2compare (A \rightarrow a) & (\text{if } E2 \text{ has } A) \\ p(A \rightarrow a) & (\text{else}) \end{cases}$$

위 수식의 (a)와 같이, 비교 연산자의 피연산자들이 모두 AND 표현식인 경우는 각 표현식 내 퍼지 팩트는 상호 독립적인, 서로 다른 퍼지 집합이며, 이 때는 좌항 표현식과 우항의 각 피연산자와의 비교값의 곱으로 연산한다. 이에 반해, 비교 연산자의 좌항은 AND 표현식이고, 우항은 OR 표현식인 (b)의 경우는 좌항과 우항 표현식 내 퍼지 팩트는 서로 동일한 퍼지 팩트이지만 상호 배반적인 관계인데, 이 경우에는 좌항 표현식과 우항의 각 피연산자와의 비교값의 합을 사용한다. 우항이 퍼지 팩트 표현식인 (c)의 경우는, 좌항 AND 표현식의 피연산자 중 우항 퍼지 팩트와 동일한 퍼지집합이 있는 피연산자와의 비교값만을 적용하고, 우항의 퍼지 팩트와 동일한 퍼지집합이 없는 경우, 우항 퍼지 팩트의 진실 확률($p(A \rightarrow a)$) 값을 적용한다.

(다) OR 표현식과의 비교

아래 수식과 같이, 비교 연산자의 좌항이 OR 표현식인 경우, OR 표현식

의 각 피연산자와 우항의 비교값과, 피연산자별 진실 확률의 비율을 곱한 값을 더하여 비교값을 산출한다. 피연산자별 진실 확률의 비율은 좌항 전체의 진실 확률 대비 좌항의 피연산자별 진실 확률의 비율을 의미한다.

$$(E1 \cup E2)compare E3 = \\ ((E1compare E3) * \frac{p_s(E1)}{p_s(E1) + p_s(E2)}) + \\ ((E2compare E3) * \frac{p_s(E2)}{p_s(E1) + p_s(E2)})$$

$$\text{이때, } p_s(E) = \sum p(E \text{의 피연산자})$$

(라) NOT 표현식과의 비교

비교 연산자의 좌항이 NOT 표현식일 때의 비교 값을 산출하기 위해서는 아래 수식과 같이 연산자의 우항이 일반 표현식일 때와, NOT 표현식일 때의 두 가지 경우가 있다.

아래 수식의 (a)와 같이, 비교 연산자의 우항이 일반 표현식인 경우에는 좌·우항이 모두 동일한 퍼지 팩트인 경우에만 적용한다. 그것은 좌·우항이 서로 다른 퍼지 팩트인 경우에는 직접적인 비교가 불가능하므로 앞에서 설명한 두 퍼지 팩트간의 비교 방법에서의 그것과 동일하다. 비교 연산자의 좌항이 NOT 표현식이고, 우항이 일반 표현식일 때의 비교 값은, 좌항의 퍼지 팩트 함수를 제외한 면적이 r이라고 하면, 이 r에 대하여 r과 우항의 퍼지 팩트 함수의 중복면적의 비율로 정의할 수 있다. 아래 수식 (b)와 같이, 비교 연산자의 좌·우항이 모두 NOT 표현식일 때의 비교 값은 우항이 일반 표현식일 때의 비교 값의 역수가 된다.

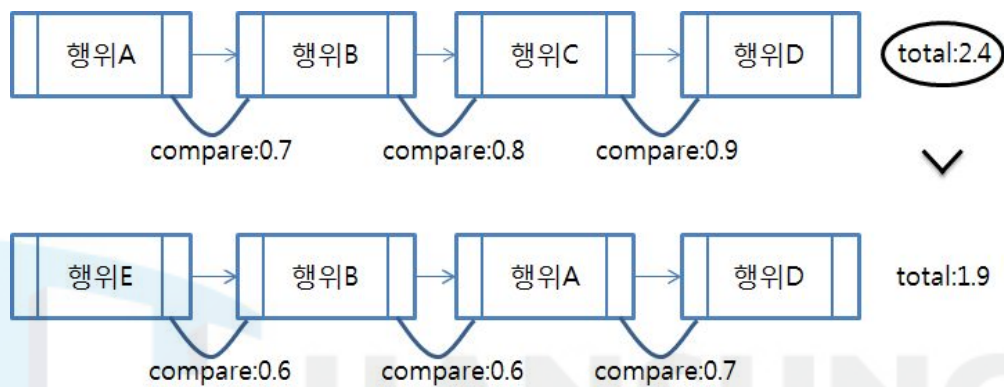
$$(a) \neg E1compare E2 = \\ \neg(A \rightarrow a)compare (A \rightarrow b) = \frac{r, b \text{의 중복면적}}{r \text{의 면적}}$$

이때, $r = A \text{의 전체면적} - a \text{의 면적}$

$$(b) \neg E1compare \neg E2 = 1 - (\neg E1compare E2)$$

4. 최적의 연결 행위 선택

행위의 시작 조건과 종료 조건의 매칭 알고리즘을 적용하여 1개 이상의 연결 행위를 탐색하였다면, 그 중 가장 최적의 연결 행위를 선택해야 한다. 아래 <그림 3-11>과 같이, 연결된 행위 간 시작 조건과 종료 조건을 매칭 알고리즘을 통해 산출된 비교 값을 모두 더한 값이 연결 행위의 전체 비용이 되며, 이 중 전체 비용이 가장 큰 연결 행위를 최종 계획으로 결정한다.



<그림 3-11> 연결 행위의 전체 비용 산출의 예

제 4 장 행위 자동 계획 기법 모의 실험

제 1 절 모의 환경 및 조건

이 장에서는 앞에서 설계한 SAF 개체의 행위 자동 계획 기법의 설계 검증 및 그 효과를 확인하기 위하여 행위 모델링 및 모의 실험을 실시하였다. 본 논문에서 사용하는 SAF 모의 구조는 개체의 기능과 행위를 독립적으로 모델링하고 그것이 실시간으로 결합되어 실행되며, 개체는 운영자에 의해 주어지는 행위를 실행하는 구조이다. 본 실험은 주어진 행위가 사전에 고려하지 않은 우발 상황이 발생했을 때, 최적의 행위를 대신 선택하여 실행하는지 로그를 검사하여 확인한다. 적 직사화기 사격 및 적 포탄 낙하 상황에 대해 대처하지 못하는 과업 수행 중에 이러한 상황을 부여하여 계획에 없는 대응 사격이나 우회 과업을 자동으로 수행하는지 검사하여 SAF의 행위 자동 계획 기법의 효과를 확인한다. 본 실험시 필요한 SAF 개체 모의용 모의 엔진, 모의 시나리오 작성 및 통제 환경, 모의 결과 전시 환경 등의 모의 실험 소프트웨어는 개발시 많은 자원과 기술이 소요되기 때문에, 이것은 ‘리얼타임비주얼’ 업체에서 개발한 모의 소프트웨어(mergeCGF Tool)를 활용하였다. 이 모의 소프트웨어는 모의 개체와 행위를 개별적으로 모델링하고, 개체의 기능을 확장할 수 있는 개방형, 조립형 구조를 가지며, 모의가 시작되면 그것들이 동적으로 결합되는 OneSAF의 인프라 스트럭처와 유사한 모의 엔진이기 때문에, 행위 자동 계획 기법을 실험하기에 최적의 환경을 제공한다. 결국, 이 실험은 위와 같이 분리된 개체와 행위가 동적으로 결합하여 모의를 수행하는 실행 구조를 갖는 모의 엔진에서만 실험이 가능하다. 본 실험을 위하여 행위 구조 설계에 사용된 클래스들과 행위 계획 알고리즘들은 추가적으로 구현하였다.

모의 실험은 다음과 같은 절차로 진행되었다. 먼저 모의 시나리오를 작성하고 행위를 모델링하는 모의 준비 과정을 수행하였다. 모의 준비가 완료되면, 모의 실험 소프트웨어를 사용하여 모의를 실행하고, 그 결과를 분석하였다.

본 실험을 통하여 정확한 효과를 검증하려면 기존 위게임 모델에 있는 모의 시나리오와 동일한 시나리오를 사용하여 그 결과를 비교하는 것이 가장

효과적이지만, 기존 시나리오를 모의하기 위해 필요한 SAF 모의 개체와 수행할 모든 행위를 모델링하고, 필요한 기능을 구현하는데 상당한 자원과 노력이 소요되기 때문에, 본 실험은 행위의 자동 계획 기능만을 실험할 수 있는 정도의 규모로 모의하는 것으로 그 범위를 제한하며, 이에 필요한 모의 개체 및 행위만 모델링하여 실험하였다.

제 2 절 모의 준비

1. 모의 시나리오 정의

SAF 개체의 행위 자동 계획 기법을 실험하기 위한 모의 시나리오는 전차소대의 방어 전투 시나리오로써, 모의 규모는 청군 1개 전차소대(K1A1 전차 3대)와 홍군 1개 탱크소대(T80 전차 3대)이다. 아래 <그림 4-1>은 모의 시나리오에 대한 상황도를 나타낸 그림이다.

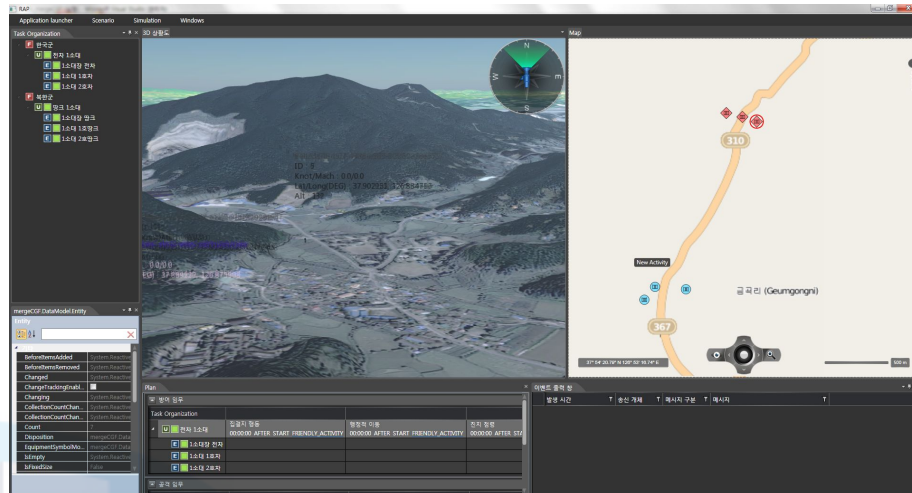


<그림 4-1> 모의 시나리오 상황도

자세한 모의 시나리오의 내용은 다음과 같다. 청군 1개 전차 소대가 집결지(①)에 대기 중 방어 진지(②)로 전술적 이동하여 방어 진지를 점령한다. 이때, 적이 사격 개시선을 통과하면 방어 전투를 실시한다. 홍군 1개 탱크 소대는 최초 집결지(a) 대기 중 공격 출발 진지(b)로 이동 후 공격 기동(전술적

이동)을 실시(㉑→㉒→㉓)하여 목표 지점을 점령(㉔)한다. 이 때, 홍군은 기동 중 우회 불가시 적 진지에 대한 사격을 실시한다.

모의 소프트웨어의 시나리오 작성 기능을 사용하여 아래 <그림 4-2>와 같이 모의 시나리오를 작성하였다.



<그림 4-2> mergeCGF 도구를 사용한 모의 시나리오 작성

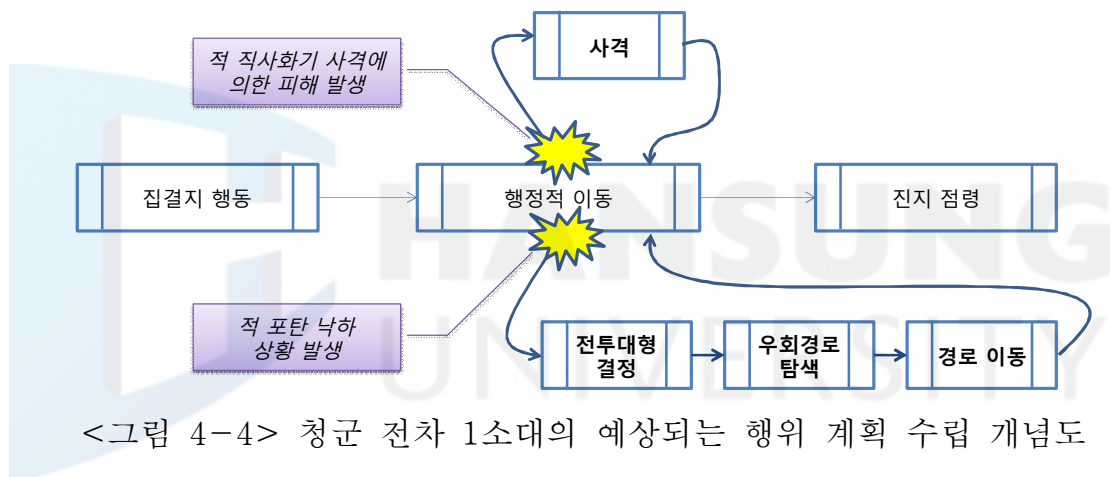
위 그림 좌측 상단의 창은 모의 개체의 편성 정보를 전시한 화면이다. 청군 전차 1소대는 1소대장 전차, 1호, 2호차를, 홍군 탱크 1소대는 1소대장 탱크와 1호, 2호 탱크를 편성하였다. 화면 중앙의 2차원 및 3차원 상황도 창으로 각각의 모의 개체를 배치하고, 그 결과를 확인하였다. 화면 아래 왼쪽 임무 작성 창을 사용하여 과업을 추가하여 임무 계획을 작성하였다. 작성된 임무 계획은 아래 <그림 4-3>과 같다.

방어 임무			
Task Organization	집결지 행동	행진적 이동	진지 점령
U 1소대	00:00:00 AFTER START FRIENDLY_ACTIVITY	00:00:00 AFTER START FRIENDLY_ACTIVITY	00:00:00 AFTER START FRIENDLY_ACTIVITY
E 1소대장 전차			
E 1소대 1호차			
E 1소대 2호차			
공격 임무			
Task Organization	집결지 행동	행진적 이동	전술적 이동
U 1소대	00:00:00 AFTER START FRIENDLY_ACTIVITY	00:00:00 AFTER START FRIENDLY_ACTIVITY	00:00:00 AFTER START FRIENDLY_ACTIVITY
E 1소대장 탱크			
E 1소대 1호탱크			
E 1소대 2호탱크			

<그림 4-3> mergeCGF 도구를 사용한 임무 계획 작성

청군 1소대는 최초 ‘집결지 행동’ 수행 중, 사용자 명령에 의해 ‘행정적 이동’으로 방어진지까지 이동하여, ‘진지 점령’ 과업을 수행한다. 홍군 1소대는 최초 ‘집결지 행동’ 수행 중, 사용자 명령에 의해 ‘행정적 이동’ 과업으로 공격출발진지로 이동 한 다음, 공격개시시간이 되면 ‘전술적 이동’으로 공격 기동을 실시한다. 이 때, ‘전술적 이동’에서 이동 경로와 ‘진지 점령’에서의 위치 좌표를 설정하여 2차원 상황도 상에 해당 경로 및 지점을 전시한다.

아래 <그림 4-4>는 청군 전차 1소대가 적 직사화기에 의한 피해 상황 및 적 포탄 낙하 상황 발생에 따라 행위 자동 계획 기법에 의해 예상되는 행위 계획 수립 개념도이다. 청군 전차 1소대는 아래 그림처럼 우발 상황 발생에 따라 수행중인 과업을 중단하고 새로운 과업을 계획하여 수행할 것이다.



2. 과업 및 행위 모델링

위 절에서 임무 계획을 작성하기 위하여 과업과 그 과업을 구성한 행위를 사전에 모델링해야 한다. 또한, 우발 상황 발생시 새로운 계획을 수립할 수 있도록 필요한 몇 개의 과업을 추가로 모델링하였다. 과업 및 행위는 XML 형태로 작성되지만, 본 논문에서는 표 형태로 그 내용을 기술하였다.

모의 시나리오에 최초 계획된 과업은 ‘전술적 이동’, ‘행정적 이동’, ‘집결지 행동’, ‘진지 점령’이며, 우발 상황에 의해 발생 가능한 과업은 ‘우회 이동’,

‘대응사격’, ‘포탄 낙하시 조치’ 이다. 각 과업의 설명과 구성 행위, 그리고 시작 및 종료 조건은 아래 [표 4-1]부터 [표 4-7]에 기술하였다.

아래 [표 4-1]의 ‘전술적 이동’ 과업은 전술적인 목적을 가지고 주어진 경로 따라 이동하는 것으로, 이동간 유효사거리 내 적을 탐지하거나, 적으로부터 직접 사격을 받으면 대응사격을 실시하는 과업이다. 이 과업은 전투대형 결정, 경로 탐색, 경로 이동 순으로 행위를 수행한다.

[표 4-1] ‘전술적 이동’ 과업 모델링

과업 명	전술적 이동	입력 파라미터	이동 경로
과업 설명	전술적인 목적으로 개체 또는 부대가 이동하는 것. 이동간 적으로부터 직접사격을 받았을 때 대응 사격을 실시함.		
구성 행위	전투대형 결정 → 경로 탐색 → 경로 이동		
시작 조건	(상황→없음) AND NOT (목적지까지거리→도착)	종료 조건	목적지까지거리→도착

아래 [표 4-2]의 ‘행정적 이동’ 과업은 이동 대형에 따라 주어진 경로를 따라 이동하는 것으로, 전술적 이동과 유사하지만, 적 출현이 발생하지 않는 지역에서의 이동을 의미한다.

[표 4-2] ‘행정적 이동’ 과업 모델링

과업 명	행정적 이동	입력 파라미터	이동 경로
과업 설명	임무에 적합한 이동대형으로 이동로를 따라 이동함.		
구성 행위	전투대형 결정 → 경로 탐색 → 행군 이동		
시작 조건	(상황→없음) AND (탐지표적→없음) AND NOT (목적지까지거리→도착)	종료 조건	목적지까지거리→도착

아래 [표 4-3]의 ‘집결지 행동’ 과업은 지속 시간 동안 지정된 공간에서 사주 경계를 실시하는 것이며, 정지된 상태에서 경계 행위를 수행하는 것이다.

[표 4-3] ‘집결지 행동’ 과업 모델링

과업 명	집결지 행동	입력 파라미터	지속 시간
과업 설명	지정된 공간에서 사주경계를 실시함		
구성 행위	경계		
시작 조건	조건 없음	종료 조건	조건 없음

아래 [표 4-4]의 ‘진지 점령’ 과업은 지시된 방어진지 또는 공격출발진지에 도착하여 진지를 편성하고 사주 경계를 수행하는 과업이다. 이 과업은 진지로 이동하는 행위와 경계 행위의 조합으로 구성하였다.

[표 4-4] ‘진지 점령’ 과업 모델링

과업 명	진지 점령	입력 파라미터	진지 위치
과업 설명	지시된 진지(방어진지, 공격출발진지)에 도착하여 분대원별 위치를 정하여 진지를 편성하고 사주 경계를 수행함.		
구성 행위	진지로 이동 → 경계		
시작 조건	NOT (목적까지거리→도착)	종료 조건	목적지까지거리→도착

아래 [표 4-5]의 ‘우회 이동’ 과업은 계획된 경로로 이동 중 자연·인공 장애물에 봉착했을 때 우회하여 전진하는 과업이다. 이 과업은 전술적 이동 과업과 유사하지만, 우회 경로 탐색 행위를 수행하는 점이 다르다.

[표 4-5] ‘우회 이동’ 과업 모델링

과업 명	우회 이동	입력 파라미터	우회 경로
과업 설명	계획된 경로로 이동 중 자연 · 인공 장애물에 봉착하였을 때 우회하여 전진함		
구성 행위	전투대형 결정 → 우회경로 탐색 → 경로 이동		
시작 조건	((상황→장애물봉착) OR (상황→포탄낙하)) AND NOT (목적지까지거리→도착)	종료 조건	(상황→없음) AND (목적지까지거리→도착)

아래 [표 4-6]의 ‘대응 사격’ 과업은 적으로부터 직접 사격을 받았을 때 적을 제압하기 위한 과업으로써, 탄약이 충분하고, 직접사격에 대한 피해가 발생되면, 탐지된 표적에 대해 사격 행위를 수행한다.

[표 4-6] ‘대응사격’ 과업 모델링

과업 명	대응사격	입력 파라미터	이동 경로
과업 설명	적으로부터 직접사격을 받았을 때 유효사거리내 탐지된 표적에 대하여 적을 제압하기 위한 사격 실시		
구성 행위	사격		
시작 조건	NOT (탐지표적→없음) AND (탄약보유→충분) AND NOT (표적거리→멀다) AND (직접사격피해→있음)	종료 조건	(탐지표적→없음) AND (직접사격피해→없음)

아래 [표 4-7]의 ‘포탄 낙하시 조치’ 과업은 적 간접화기에 의한 포탄 낙하 상황시, 우회 경로가 없으면 제자리에서 일시 정지하고, 우회 경로가 있으면 우회 이동을 실시하는 과업이다.

[표 4-7] ‘포탄 낙하시 조치’ 과업 모델링

과업 명	포탄 낙하시 조치	입력 파라미터	없음
과업 설명	적 포탄 낙하시의 조치. 우회 경로가 없으면 일시 정지하고, 우회 경로가 있으면 우회 이동을 실시함.		
구성 행위	우회 이동(우회 경로 존재) 또는 경계(우회 경로 미존재)		
시작 조건	(상황→포탄낙하)	종료 조건	상황→없음

각 과업을 구성하는 행위는 ‘경계’ 행위, ‘전투대형 결정’ 행위, ‘경로 탐색’ 행위, ‘우회경로 탐색’ 행위, ‘경로 이동’ 행위, ‘행군 이동’ 행위, ‘진지로 이동’ 행위, ‘사격’ 행위이며, 각 행위에 대한 설명과 시작 및 종료 조건은 아래 [표 4-8]에 기술하였다.

[표 4-8] 행위 모델링

행위 명	행위 설명	시작 조건	종료 조건
경계	전투 대형을 유지하면서 지정된 경계 방향을 지향하여 경계함 적 발견 시 또는 적 기습사격 시 위치를 변경하지 않고 사격을 실시함	조건 없음	조건 없음
전투대형 결정	임무 및 지형 조건에 따라 전차 소대의 전투 대형을 종대 또는 횡대 대형으로 결정함	전투대형→미정	전투대형→종대 OR 전투대형→횡대
경로 탐색	계획된 경로에 대하여 실제 이동할 최적 경로를 탐색함	전투대형→종대 OR 전투대형→횡대	탐색경로→결정
우회경로 탐색	계획된 경로에 대하여 우회 경로를 탐색함	(상황→장애물붕착 OR 상황→포탄낙하) AND (전투대형→종대 OR 전투대형→횡대)	탐색경로→결정
경로 이동	부대이동, 공격을 위한 기동, 철수, 수색정찰 등 모든 형태의 이동으로서 사전 계획되거나 지시된 경로로 이동함	NOT 목적지까지거리→도착 AND 탐색경로→결정	목적지까지거리→도착 AND 상황→없음
행군 이동	행정적 이동을 위한 이동으로서, 적의 출현이 예상되지 않는 지점으로 사전 계획되거나 지시된 경로로 이동함	탐지표적→없음 AND NOT 목적지까지거리→도착 AND 탐색경로→결정	목적지까지거리→도착 AND 상황→없음
진지로 이동	지시된 진지 위치로 이동함	NOT 목적지까지거리→도착	목적지까지거리→도착
사격	탐지된 표적에 대하여 모든 직사화기 사격을 실시함	NOT 피해발생→없음 AND NOT 탐지표적→없음 AND 탄약보유→충분	피해발생→없음 AND 탐지표적→없음

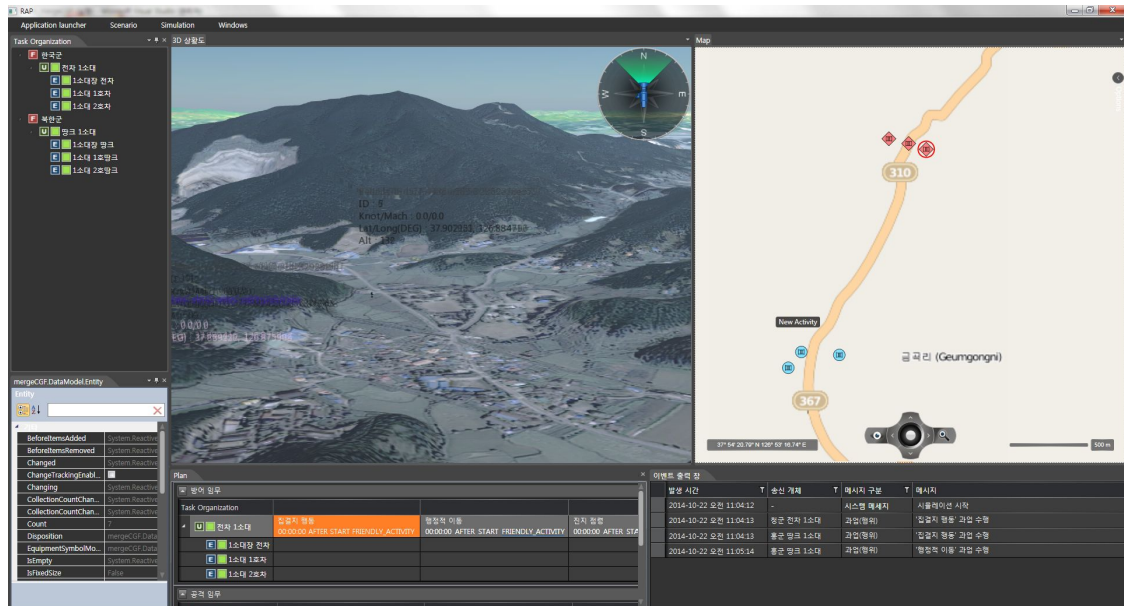
과업 및 행위의 시작 조건 또는 종료 조건이 없는 경우는 행위 연결시 어떠한 조건도 필요 없다는 의미이기 때문에, 어떠한 행위도 해당 행위 앞 또는 뒤로 연결될 수 있다. 그리고 과업과 행위의 시작/종료 조건에 사용된 퍼지 팩트는 아래 [표 4-9]과 같이 정의하였다. 퍼지 집합에 사용된 퍼지 팩트 함수를 결정하는 것은 내용 전문가의 경험과 다양한 실험을 통하여 정확한 값을 사용해야 신뢰성 있는 계획 수립 결과를 얻을 수 있지만, 본 실험의 제한 사항으로 인하여 임의의 퍼지 팩트 함수와 구간값을 정의하여 사용하였다.

[표 4-9] 퍼지 팩트 정의

퍼지 집합	퍼지 팩트 함수	퍼지 집합	퍼지 팩트 함수
목적지까지 거리	도착, 가깝다, 멀다	탐지 표적	없음, 적음, 많음
상황	없음, 장애물봉착, 포탄낙하	탄약 보유	없음, 적음, 충분
표적 거리	가깝다, 중간, 멀다	직접사격피해	없음, 있음
지형 속성	개활지, 좁은 도로	전투 대형	미정, 종대, 횡대
탐색 경로	결정, 미결정	-	-

제 3 절 모의 실시 및 결과 분석

위 절의 모의 준비 내용과 같이 과업과 행위를 모델링하고, 모의 시나리오의 임무 계획이 작성되면, 모의 소프트웨어(mergeCGF Tool)를 사용하여 행위 자동 계획을 실험한다. 아래 <그림 4-4>는 모의 소프트웨어의 실행 화면이다. 모의가 시작되면, 그림 하단 중앙의 임무 작성 창에 SAF 개체가 현재 수행중인 과업이 주황색 색상으로 표시되고, 그림 하단 우측의 이벤트 전시 창에서 과업의 변경 상황을 실시간으로 확인할 수 있다. 또한 그림 중앙의 2차원 및 3차원 상황도 상에 SAF 개체의 위치 및 상태를 확인할 수 있다. SAF 개체는 시나리오에 작성한 과업을 순차적으로 수행하며, 종료된 과업은 임무 작성 창에서 검정색 색상으로 표시된다.



<그림 4-5> mergeCGF 소프트웨어를 사용한 모의 실행

모의가 시작되면, 청군 전차 1소대는 지속 시간 동안 집결지 행동 후, 방어 진지까지 행정적 이동을 실시한다. 홍군 탱크 1소대는 지속 시간동안 집결지 행동 후, 공격출발진지 까지 행정적 이동을 하고, 전술적 이동으로 공격 기동을 실시한다. 홍군 탱크 1소대는 전술적 이동 중 청군 전차를 탐지하게 되면, 보유 화기로 선제 사격을 실시한다. 이때의 과업 진행 상황과 3차원 상황도의 화면은 아래 <그림 4-5>와 <그림 4-6>과 같다.

방어 임무			
Task Organization			
U 청군 전차 1소대	집결지 행동 00:00:00 AFTER START FRIENDLY_ACTIVITY	행정적 이동 00:00:00 AFTER START FRIENDLY_ACTIVITY	진지 점령 00:00:00 AFTER START FRIENDLY_ACTIVITY
E 1소대장 전차			
E 1소대 1호차			
E 1소대 2호차			
공격 임무			
Task Organization			
U 홍군 탱크 1소대	집결지 행동 00:00:00 AFTER START FRIENDLY_ACTIVITY	행정적 이동 00:00:00 AFTER START FRIENDLY_ACTIVITY	전술적 이동 00:00:00 AFTER START FRIENDLY_ACTIVITY
E 1소대장 탱크			
E 1소대 1호탱크			
E 1소대 2호탱크			

<그림 4-6> 임무 작성 창 의 과업 진행 상황



<그림 4-7> ‘행정적 이동’ 중인 전차 1소대

홍군 탱크 1소대는 전술적 이동 중 청군 전차를 탐지하면 전차포 사격을 수행한다. 이 때, 청군 전차 1소대는 ‘행정적 이동’ 과업 수행 중, 적 전차포 사격에 대한 피해가 발생되고, 우발 상황에 대한 행위 자동 계획을 실시한다. 아래 <그림 4-7>의 이벤트 출력 창에는 청군 1소대가 적 전차 사격에 의해 피해가 발생하여 새로운 행위들을 계획하는 것을 기록한 로그 내용이다.

이벤트 출력 창				
발생 시간	송신 개체	메시지 구분	메시지	
2014-10-22 오전 11:04:12	-	시스템 메시지	시뮬레이션 시작	
2014-10-22 오전 11:04:13	청군 전차 1소대	과업(행위)	'집결지 행동' 과업 수행	
2014-10-22 오전 11:04:13	홍군 탱크 1소대	과업(행위)	'집결지 행동' 과업 수행	
2014-10-22 오전 11:05:14	홍군 탱크 1소대	과업(행위)	'행정적 이동' 과업 수행	
2014-10-22 오전 11:05:44	청군 전차 1소대	과업(행위)	'행정적 이동' 과업 수행	
2014-10-22 오전 11:06:02	홍군 탱크 1소대	과업(행위)	'전술적 이동' 과업 수행	
2014-10-22 오전 11:07:48	홍군 탱크 1소대	사격	탐지 표적에 대한 사격 실시	
2014-10-22 오전 11:07:48	청군 전차 1소대	피해 발생	1소대 2호차 피해 발생 (이동불가)	
2014-10-22 오전 11:07:49	청군 전차 1소대	과업(행위)	'행정적 이동' 과업 중단	
2014-10-22 오전 11:07:49	청군 전차 1소대	행위 계획	계획 수립 : '사격' - '행정적 이동'	
2014-10-22 오전 11:07:49	청군 전차 1소대	과업(행위)	'사격' 행위 수행	

<그림 4-8> 적 전차 사격에 대한 행위 계획 수립 로그

위 그림의 로그를 보면, 청군 전차 1소대가 피해가 발생하여 수행중인 ‘행정적 이동’ 과업을 중단한다. 행위가 중단되면, 플래너는 행위 자동 기법에 의해 계획을 생성한다. 현재 적 전차에 의한 피해가 발생하였고, 적 전차를 탐지한 상황에서, 플래너는 새로운 행위인 ‘사격’을 수행하고, ‘행정적 이동’을 실시하는 것으로 계획을 수립하였다. 즉, ‘행정적 이동’ 중 우발 상황에 대해 적절히 대응 ‘사격’을 하고, 적을 제압하거나, 탐지된 적 전차가 없는 상황이 되면 다시 ‘행정적 이동’을 하여 목적지까지 도달하는 것을 확인할 수 있다.

다른 실험은, 전과 동일한 시나리오를 사용하여 모의 중 적 포탄 낙하 상황을 조성하여 SAF 개체의 행위 자동 계획을 실험하였다. 모의가 시작되면, 앞에서 실험한 것과 같이, 청군 전차 1소대는 지속 시간 동안 집결지 행동 후, 방어진지까지 행정적 이동을 실시한다. 이 때, 사용자는 모의 소프트웨어의 매직 기능을 사용하여 전차의 이동 경로 상에 적 포탄 낙하 상황을 부여하였다. 이때의 3차원 상황도 화면은 아래 <그림 4-8>과 같다.



<그림 4-9> 3차원 상황도에서의 포탄 낙하 상황

아래 <그림 4-9>의 이벤트 출력 창에는 청군 1소대가 ‘행정적 이동’ 과업 수행 중, 적 간접화기에 의한 포탄 낙하 상황이 발생되어 행위 자동 계획에

의해 새로운 행위들이 동적으로 계획되는 것을 확인할 수 있는 로그가 나타나 있다. 청군 전차 1소대가 반경 100m 이내 포탄 낙하 상황을 탐지하면, 현재 수행중인 '행정적 이동' 과업 시작 조건의 평가값이 거짓이 되어, 과업 수행을 중단하게 된다. 행위가 중단되면, 플래너는 행위 자동 기법에 의해 계획을 생성한다. 플래너는 우회 경로가 존재한다고 판단하여, 우회 이동을 실시하도록 '전투대형 결정', '우회경로 탐색', '경로 이동', 행위 순으로 계획을 수립한다. 수립된 계획은 순차적으로 실행하여, 전투대형을 결정하고, 우회경로를 탐색하여 우회 이동을 실시한다. 우회 이동이 종료되면, 다시 '행정적 이동' 과업을 다시 실행하여 주어진 임무를 수행한다.

이벤트 출력 창				
발생 시간	송신 개체	메시지 구분	메시지	
2014-10-22 오전 11:32:44	-	시스템 메시지	시뮬레이션 시작	
2014-10-22 오전 11:32:45	청군 전차 1소대	과업(행위)	'집결지 행동' 과업 수행	
2014-10-22 오전 11:32:45	홍군 탱크 1소대	과업(행위)	'집결지 행동' 과업 수행	
2014-10-22 오전 11:33:45	홍군 탱크 1소대	과업(행위)	'행정적 이동' 과업 수행	
2014-10-22 오전 11:34:15	청군 전차 1소대	과업(행위)	'행정적 이동' 과업 수행	
2014-10-22 오전 11:34:45	홍군 탱크 1소대	과업(행위)	'전술적 이동' 과업 수행	
2014-10-22 오전 11:35:27	-	사격	청군 1소대장 전차 전방에 간접화기에 의한 포탄 낙하 발생	
2014-10-22 오전 11:35:28	청군 전차 1소대	탐지	반경 100m 이내 포탄 낙하 탐지	
2014-10-22 오전 11:35:29	청군 전차 1소대	과업(행위)	'행정적 이동' 과업 중단	
2014-10-22 오전 11:35:29	청군 전차 1소대	행위 계획	계획 수립 : '전투대형 결정' - '우회경로 탐색' - '경로 이동' - '행정적 이동'	
2014-10-22 오전 11:35:30	청군 전차 1소대	과업(행위)	'전투대형 결정' 행위 수행	
2014-10-22 오전 11:35:30	청군 전차 1소대	물리 모델	대형 변경 : 종대대형	
2014-10-22 오전 11:35:31	청군 전차 1소대	과업(행위)	'우회경로 탐색' 행위 수행	
2014-10-22 오전 11:35:32	청군 전차 1소대	물리 모델	이동경로 결정 완료	
2014-10-22 오전 11:35:33	청군 전차 1소대	과업(행위)	'경로 이동' 행위 수행	

<그림 4-10> 적 포탄낙하 상황에 대한 행위 계획 수립 로그

위의 실험 결과처럼, SAF 개체는 적 기습사격 및 포탄낙하 상황에 대한 새로운 계획을 수립하여 적절히 대응하는 것을 확인할 수 있다. 또한, 기존 위 게임 모의 모델과 상용 SAF 운용도구도 위 실험 결과와 유사한 결과를 얻을 수 있다. 그러나, 아래의 [표 4-10]과 같이, 기존 모델의 상황 대처 기능은 제안하는 행위 자동 계획 기법과는 많은 차이가 있다.

[표 4-10] 기존 위게임 모델, 상용 SAF 운용도구와의 차이점 비교

구분	기존 위게임 모의 모델	상용 SAF 운용도구	행위 자동 계획 기법
상황 판단 조건 표현 형태	- 소스코드 또는 스크립트로 구현 (if .. then .. else ..) - 정해진 수치값 사용 (정수, 실수, 부울값 등)		- XML 형태 (논리 표현식) - 퍼지를 사용하여 애매모호한 교리 내용 표현
조건 수정 여부	불가능	가능	가능
모의 형태	미리 정의한 조건에 따른 행위만 수행함		실시간으로 다양한 행위들을 조합할 수 있기 때문에, 다양한 행위 수행이 가능함
새로운 행위 추가시	- 적 기습 사격시 또는 포탄 낙하시 대응에 대한 조건식이 if-then 형태로 하드코딩 또는 스크립트로 되어 있음 - 새로운 행위가 추가될 경우 소스코드 또는 스크립트를 직접 수정해야 함 - 하나의 행위를 추가하더라도 발생 가능한 모든 케이스에 대해서 모두 if-then 형태의 조건문 작성 필요		- 사격, 우회와 같은 행위의 시작/종료조건을 XML 형태로 작성 ‘철수’와 같은 새로운 행위 추가시 기존 ‘사격’, ‘우회’ 행위의 조건 수정 필요없음 - 새로운 행위의 시작/종료조건만 퍼지 논리 표현식으로 작성함 - 발생 가능한 모든 케이스에 대해 고려하지 않더라도 현 상황에 맞는 적절한 행위 계획 가능

기존 위게임 모의 모델과 상용 SAF 운용도구는 소스코드나 스크립트 형태의 if-then 문장으로 기술되어 있지만, 본 행위 자동계획 기법은 XML 형태의 퍼지 논리 표현식으로 기술된다. 조건문을 퍼지 논리 표현식으로 사용함으로써, 애매모호한 교리의 내용을 그대로 반영할 수 있고, 조건 표현식을 소스코드가 아닌 XML로 작성하기 때문에, 수정 및 변경이 용이한 장점이 있다. 또한, 기존 위게임 모의 모델이나 상용 SAF 운용도구는 정의된 조건에 따른 행동만을 수행하고, 조건이 없는 상황에 대해서는 처리가 불가능하지만, 행위

자동 계획 기법을 사용하면 실시간으로 적합한 행위들을 조합하기 때문에, 다양한 행위 수행이 가능하다.

즉, 기존 위게임 모델에서는 적 기습사격시 또는 포탄낙하시 대응에 대한 조건식을 if-then 형태로 소스코드에 하드코딩 되어 있기 때문에 새로운 행위가 추가될 경우 소스코드를 직접 수정해야 될 뿐만 아니라, 하나의 행위를 추가하더라도 발생 가능한 모든 케이스에 대해서 전부 if-then 형태의 조건문으로 처리해야 한다. 하지만, 행위 자동 계획 기법을 사용하면, 사격, 우회와 같은 행위에 대한 시작 및 종료 조건을 XML 형태로 작성하며, ‘철수’와 같은 새로운 행위를 추가하더라도 기존의 ‘사격’, ‘우회’ 행위의 조건 수정이 전혀 필요없고, 단지 새로운 행위의 시작 및 종료 조건만 퍼지 논리 표현식으로 작성하면 되며, 발생 가능한 모든 케이스에 대해 고려하지 않더라도 현 상황에 맞는 적절한 행위의 계획이 가능하다.

앞서 실험한 SAF의 행위 자동 계획 기법에 대한 두 가지 실험은, 행위를 분류한 방법과 그 행위의 시작 조건과 종료 조건, 사용된 퍼지 팩트의 내용, 그리고 퍼지 팩트 함수의 구간 값에 따라 시뮬레이션 결과와 행위 조합의 적절성이 달라질 수 있다. 위 실험에서는 적의 기습 사격 시 및 포탄 낙하 시 두 개의 상황에 대해 대응 사격과 우회 이동을 계획하도록 시작 및 종료 조건을 단순화 하였고, 사용된 퍼지 팩트도 제한적으로 사용하였다. 행위를 모델링하는 전문가가 행위의 해상도에 맞는 수준으로 행위의 시작 및 종료조건을 상세하게 기술하고, 사용되는 퍼지 팩트와 퍼지 팩트 함수의 구간 값을 정확하게 기술한다면 최적의 계획 결과를 얻을 수 있다. 또한, 조건식에 들어가는 퍼지 팩트의 추상화 수준을 높인다면, 예측하지 못한 우발 상황에서도 대처 가능한 행위를 계획할 수 있다. 제안된 설계 기법은 행위의 내용과 추상화 수준과는 독립적으로 설계하였기 때문에, 도메인 전문가들이 구축한 높은 수준의 행위들을 그대로 적용할 수 있다.

이와 같이 본 논문에서 제안한 SAF의 행위 자동 계획 기법을 활용하여 SAF 개체가 우발 상황에 대해 능동적으로 대처할 수 있는 SAF 모의 모델을 개발할 수 있다.

제 5 장 결 론

본 논문에서는 인공지능 계획 기법을 사용하여 SAF 개체가 우발 상황에 대해 능동적으로 대처할 수 있는 행위 자동 계획 기법을 설계하였다.

인공지능 계획 기법을 적용함으로써, 기존 위게임 모의 모델이나 규칙기반 시스템에서 적용된 모의논리 또는 규칙에 의한 모델링 방식보다 높은 추상화 수준에서 행위 모델링이 가능하고, 동적으로 행위를 구성하기 때문에 우발 상황에 대한 대처 능력이 소스코드나 스크립트에 의한 기존 동작 방식보다 더 유연하다고 볼 수 있다.

SAF의 행위 자동 계획을 위하여 STRIPS 기반 계획 기법과 HTN 기반 계획 기법을 혼합하여 설계하였다. SAF 개체가 수행하는 행위의 구조는 HTN 기반 계획(HTN-based planning) 기법의 과업 구조를 활용하여 행위를 계층적으로 구성할 수 있는 구조로 설계하였다. 따라서 행위는 계층적인 구조를 가지며, 행위를 수행하기 위한 필수 조건인 시작조건과, 행위를 종료하기 위한 조건인 종료 조건을 퍼지 표현식으로 사용하였다.

행위의 최종 목적을 달성하기 위하여 필요한 행위 목록들을 자동으로 생성하는 행위 계획 수립 기법은 STRIPS 기반 계획(STRIPS-based planning) 기법을 활용하여 설계하였다. 행위 계획 수립 시 그 탐색 영역을 줄이기 위하여 인공지능의 계획 방법 중의 하나인 목표 상태로부터 현재 상태를 역으로 찾는 목표 지향 행위 계획(Goal Oriented Action Planning, GOAP) 방법을 활용하였고, 사용되는 행위 연결 탐색 알고리즘과 시작/종료조건 매칭 알고리즘을 설계하였다.

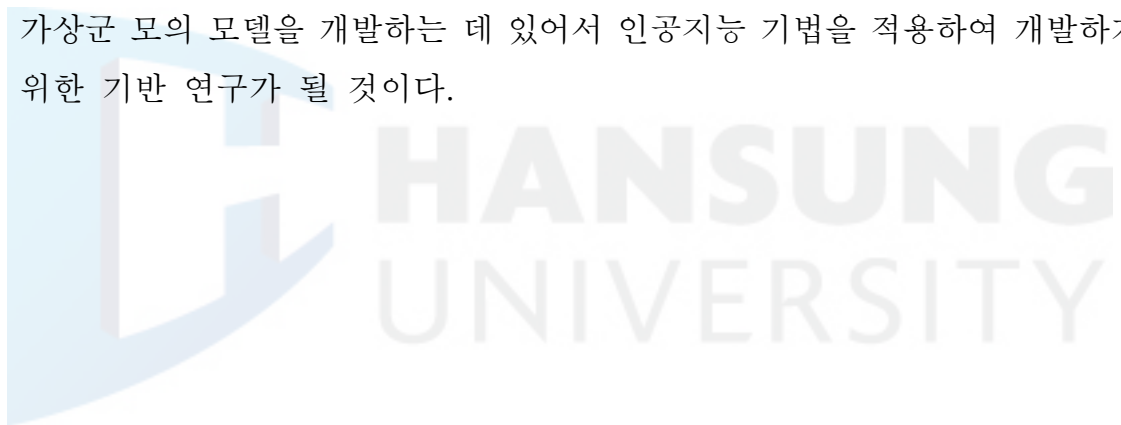
SAF 개체의 행위 모델링 및 모의 실험을 통해서는 행위 자동 계획 기법의 효과와 설계를 검증하고, SAF 개체의 행위 자동 계획 기법의 활용 방안을 제시하였다.

본 논문의 목적은 SAF 개체가 계획하지 않은 우발 상황에 대해 최적의 행위를 선택하도록 하여, 모의 개체가 실 전장 상황에서의 유사한 대응 행동을 자동으로 수행할 수 있는 기법을 개발하는 것이다. 그러나 최적의 행위를 계

획하기 위해서는 본 논문에서 제시한 설계 기법뿐만 아니라, 행위의 시작 및 종료 조건, 퍼지 팩트의 멤버십 함수 구간값 등의 도메인 내용이 전문가의 의견과 실험을 통하여 충실하고 정확하게 기술되어야 한다.

SAF 개체 모의를 포함한 가상군 모의 모델에 활용할 수 있는 기술은 본 논문에서 활용한 계획 기법을 비롯하여, 추론, 협업, 학습 등 다양한 인공지능 분야를 적용할 수 있다. 그러나 인공지능 기술은 도메인의 광범위함, 기술적 어려움 및 복잡성으로 인해 군 도메인에 적용하여 활용하기가 어려운 분야이다. 인공지능 기술이 적용된 가상군 모의 모델을 개발하기 위해서는 핵심기술에 대한 심화된 연구를 통해 인공지능 기술들을 연구개발해야 한다. 그리고 도메인 전문가가 충실하고 정확하게 가상군의 행위를 구축할 수 있도록 행위를 모델링하고 실험할 수 있는 도구 개발이 필요하다.

결론적으로 본 논문을 통하여 제시된 SAF 개체의 행위 자동 계획 기법은 가상군 모의 모델을 개발하는 데 있어서 인공지능 기법을 적용하여 개발하기 위한 기반 연구가 될 것이다.



【참고문헌】

1. 국내문헌

이성기의 2명, 『가상군 기술의 국방 적용방안』, 한국인터넷정보학회, 제13권 제4호, 2012, p17~23.

이성기의 4명, 『자율지능형 컴퓨터 가상객체 기술의 국방 M&S 적용현황 및 활용방안』, 정보과학회지 제26권 제11호, 2008, p27~32.

문형곤외 2명, 『OneSAF 모형 도입/실용화』, 한국국방연구원 연구보고서 모 07-2474, 2008.

이상용, 『인공지능의 세계』, 서울 : 21세기사, 2008.

조영임, 『인공지능시스템』, 서울 : 홍릉과학출판사, 2012.

정영근, 『퍼지 추론을 이용한 최단 경로 탐색 알고리즘의 개발』, 서울대학교 대학원 지구환경시스템공학부 석사학위 논문, 2006.

최대희외 2명, 『위게임에서 가상군의 목표지향행위계획 및 재 계획 방법』, KIIS Fall Conference 2012 Vol. 22, No. 2, 2012.

차명희, 『계획 기반 NPC의 자율적인 학습능력에 대한 동향 분석』, 한국컴퓨터게임학회 논문지 제25권 제3호, 2012.

유용준, 『계층 구조적 자율형 가상군 모델링 및 시뮬레이션』, 한국항공대학교 대학원 컴퓨터공학과 박사학위 논문, 2011.

최대희외 4명, 『위게임에서 가상군의 자율적 행위를 위한 자동계획 기법』, 한국컴퓨터정보학회 논문집 제16권 제9호, 2011.

김태섭, 『무기체계 분석을 위한 모의엔진 아키텍처 연구』, 한국시뮬레이션학회 논문지 제19권 제2호, 51-62, 2010.

장장훈, 『유전자 알고리즘과 신경망을 이용한 MMORPG의 지능캐릭터 구현에 관한 연구』, 동명정보대학교 컴퓨터공학과 석사학위논문, 2008.

김평원, 『퍼지 논리를 활용한 논증 텍스트 분석 모형 연구』, 국어교육학연구
41권 0호, p339~378, 2011.

2. 국외문헌

E. Long, 『Enhanced NPC Behaviour using Goal Oriented Action Planning』, Master thesis, School of Computing and Advanced Technologies, University of Abertay Dundee, 2007.

Dr Nash, 『Overview of the OneSAF Models Infrastructure』, OneSAF Users Conference, 2009.

J.Orkin, 『Applying Goal-Oriented Action Planning to Games in AI Game Programming Wisdom 2』, Charles River Media, 2003, p217~228.

Glenn Wissing, 『Multi-agent planning using HTN and GOAP』, Luiea University of Technology, Bachelor's Thesis, 2007.

SoengKee Lee 외 6, 『Autonomous Intelligent Simulation Model based on CGF for Military Training and Analysis』, ACM Computational Intelligence and Information Security, 2011.

M. Lekavy and P. Navrat, 『Expressivity of STRIPS-like and HTN-like Planning』, Agent and Multi-Agent Systems : Technologies and Applications Lecture Notes in Computer Science Volume 4496, 2007, p121-130.

『VR-Forces Users Guide』, VT MAK, 2009.

『STAGE 6.0 User's Guide』, PRESAGIS, 2009.

3. 인터넷 사이트

http://ko.wikipedia.org/wiki/컴포지트_패턴

http://ko.wikipedia.org/wiki/A*_알고리즘

<http://en.wikipedia.org/wiki/STRIPS>

http://en.wikipedia.org/wiki/Disjunctive_normal_form

http://en.wikipedia.org/wiki/Esspresso_heuristic_logic_minimizer



ABSTRACT

A Study of Automated Behavior Planning Technique for Semi Automated Forces

Cho, Junho

Major in National Defense Modeling & Simulation

Dept. of National Defense Modeling & Simulation

Graduate School of National Defense Science

Hansung University

Military simulations have been widely applied to various military areas for a long time. While those areas include analysis, training, and acquisition, the recent trend is incorporating areas of modeling and simulation for test & evaluation and combat effectiveness analysis. Moreover, the technique of Computer Generated Forces (CGFs) attracts more attention, in which an autonomous virtual entity reflects the behaviors of a real-world human and the collective logics of a military unit. CGFs can be categorized as SAF (Semi Automated Forces) or FAF (Fully Automated Forces) according to whether any trainee (human participant) is involved in simulation or not, and according to the level of autonomous intelligence of an entity.

Most commercial games or simulation models of existing war-games generally adopt finite state machine or rule-based system. Those techniques are advantageous in establishing knowledge of military experts,

but they have disadvantage in such a way of specifying exhaustively all the rules to counteract every possible situations at modeling phase.

To overcome such disadvantage this work adopts AI planning in which an entity plans in realtime to achieve a goal given to it. The AI planning technique is a process to select a behavior which is more proper to reach the goal. By adopting AI planning technique, we can model in higher abstract level, and since AI planning composes behaviors dynamically it provides more flexible way of planning than the existing ways that plan directly on source code or script.

This work suggests a technique of automated behavior planning that can actively counteract a certain situation by planning automatically to achieve the SAF entity's goal.

To hierarchically construct a behavior that a SAF entity performs, this work uses the structure of task of HTN-based planning. Thus a behavior has hierarchical structure, and it defines two type of conditions in fuzzy logic expression : the start-condition (pre-condition) to initiate the behavior and termination condition (post-condition) to end it.

To automatically generate the sequence of behaviors required to achieve the entity's final goal, it uses the method of behavior plan generation using STRIPS-based technique. In behavior planning (in retrieving candidate behaviors), to minimize the search scope, this work suggests a technique of automated behavior planning, Goal Oriented Action Planning (GOAP), which is a way of automated planning techniques. It uses back propagation from the goal state to the current state.

This work has been verified and shown the effectiveness and design of SAF entity's automated behavior planning technique through behavior modeling and simulation experiments, and suggests the actual usage of the technique.

【Key Words】 Computer Generated Forces (CGF), Semi-automated Forces (SAF), Artificial Intelligence, Automated Planning, Goal Oriented Action Planning (GOAP), Fuzzy, Modeling & Simulation

