

플랜인지 에이전트의 플랜 라이브러리 구축 방법론 연구

김 진섭, 김 도완
배재대학교 정보통신공학과

**A Study of the Plan Library Structure for
Plan Recognition Agent**

Jin-Sup Kim, Do-Wan Kim
Dept. of Information and Communications Engineering, PaiChai University

1. Introduction

현대를 대표하는 키워드 중의 하나로서 ‘정보화’를 들 수 있으며 소프트웨어 산업의 발달은 기술적 측면에서 시스템 성능의 대형화와 다기능화를 가능케 하였지만 결과적으로 시스템의 복잡도를 증가하게 만들었으며 그로 인해 사용자 인터페이스는 더욱 중요한 소프트웨어 개발 요소가 되었다. 본 연구는 시스템의 지능화를 위한 기본적인 Plan Recognition 방법과 Design 구성의 방법을 살펴보고 효율적인 Plan Library 구축 방법론에 관하여 다루고 있다.

주어진 입력으로부터 사용자의 플랜을 유추하는 기술을 플랜인지라 한다. 인간-컴퓨터 인터랙션(Human-Computer Interaction)에서 인간과 컴퓨터는 독립된 객체를 형성하며, 이때 인터랙션을 가능케 하는 인터페이스는 중요한 의미를 갖는다. 인터페이스란 원래 이질적인 두 가지 물질이 접촉한다는 의미의 화학용어에서 탄생하였고, 일반적으로 두 종류의 서로 다른 세계가 상호교섭하는 장을 의미한다[1]. 사용자의 플랜을 인지하는 것에 관련된 이슈들과 사용자의 인터페이스에 적용하는 플랜정보 지원은 인공지능 문헌에서 많이 다루어지고 있다. 본 논문은 플랜인지 및 플랜라이브러리 구축에 관계되는 항목들을 살펴보고, 기존의 1 차원적 플랜라이브러리 구축 방법을 보다 간결하게 처리할 수 있는 2 차원 플랜라이브러리 구축 방법론을 연구 제시한다. 2 장에서는 Temporal Reasoning Algorithms 을 통한 플랜인지 및 플랜라이브러리에 대하여 토론되어 졌다. 3 장에서는 플랜인지와 관련된 항목들이 소개된다. 4 장은 플랜인지 알고리즘을 일차원이 아닌 이차원적인 영역에서 접근함으로써 플랜 라이브러리 구축을 쉽고 효율적으로 할 수 있는 기술에 대하여 서술하고 있다. 5 장에서는 향후 연구 방향이 제시되었다.

2. Plan Recognition through the Temporal Reasoning Algorithms

Allen[1983a]은 새로운 제약조건(constraints)을 다른 것에 전달하는 추론 알고리즘과 모순된 점을 검색하는 방식을 제안했다. 제약조건(constraints)사항을 추론하기 위해 교차점(Intersection)과 구성(Composition)이 정의 되었다. 교차점(Intersection)은 두 제약조건(Constraints)합이고, 두 제약조건(Constraints)의 구성(Composition)은 그 기초 관계의 pair-wise 곱셈의 집합으로 되어있다.

$$C1 \circ C2 = \{ a \times b \mid a \in C1 \text{ and } b \in C2 \}$$

$a \times b$ 의 결과는 다음 정의된 테이블 [Allen, 1983 a] 에서 찾아 볼 수 있다.

Primitives	Symbols	Inverses	Examples
------------	---------	----------	----------

X Before Y	X b Y	Y bi X	XXX YYY
X Meets Y	X m Y	Y mi X	XXXYYY
X overlaps Y	X ° Y	Y oi X	XXX YYY
X Starts Y	X s Y	Y si X	XXX YYYYY
X During Y	X d Y	Y di X	XXX YYYYYYY
X Finishes Y	X f Y	Y fi X	XXX YYYYYY
X equal Y	X eq Y	Y eq X	XXX YYY

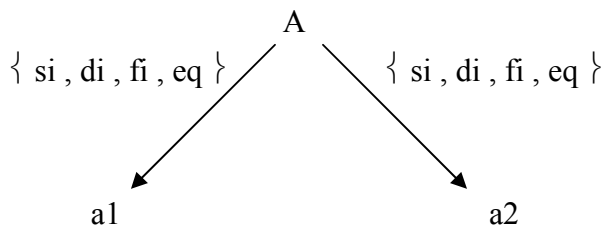
예를 들자면 $C1 = \{ b, m, o \}$, $C2 = \{ m, o \}$ 라면

$C1 \circ C2$ 의 구성은 $\{ b, m, o \}$ 이다.

하지만 Allen 의 알고리즘은 분배(decomposition)를 포함하는 계획에 적용 할 때 부족한 결과를 제공한다.

여기에 행동 간격을 나타내는 A 와, 두 개의 서브 행동간격 a1, a2 를 사용하자. a1 과 a2 사이에 아무런 제약조건도 없는 Plan 라이브러리의 초기값을 가정해보자.

$A \{ si, di, fi, eq \} a1$ and $A \{ si, di, fi, eq \} a2$ 라고 할 때 다음과 같이 결정할 수 있다.

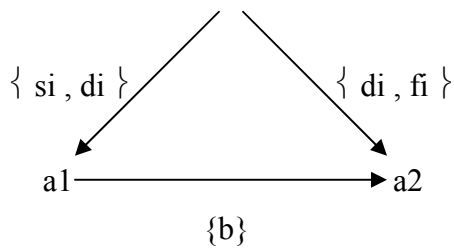


< An Example of One Decomposition >

a1 과 a2 사이에 새롭게 제한된 $\{ b \}$ 를 가정해 보자.

Allen 의 알고리즘에 기초해서 우리는 다른 제약(constraints)에 전달 할 수 있고 그림과 같다.

A



<Strong Results Desired>

하지만 a1,a2 가 Allen 의 알고리즘에 일관된 네트워크이지만 분배(decomposition)를 대표하는 것은 아니다. 예를 들자면 만약 추상(abstract)적인 행동의 간격이 서브행동에 의해 일시적으로 문제가 생기면 분배(decomposition)는 끝난다.

최근의 연구 대부분은 abstract action 이 low-level 로 분배되는 계층 구조를 강조했다. 하지만 플랜은 action 이 실행되는 시간 간격의 temporal network 로 보여 질 수 있다. 이러한 시간 사이의 temporal 관계는 Allen's interval calculus(Allen, 1983)에 의해서 명확 해진다.

Allen 의 high-level 관계로 나타내면 다음과 같다.

Precedes(a,b) = before(a,b) meets(a,b) overlaps(a,b)

Includes(a,b) = starts(b,a) during(b,a) finishes(b,a) equal(a,b)

위의 두 관계는 temporal network 구성에 도움을 준다.

3. Previous work

인공지능에서 플랜인지는 복잡한 응용 시스템에 적합한 개선의 방법이나 자연어 이해를 위한 어떤 현상을 설명하기 위해서 처음 연구 되었다. 플랜인지는 또한 자연어 연구의 Question – answering 시스템에서도 다루어 졌다(Cohen et al., 1982). 시스템이 이해할 수 있는 언어학적 현상의 범위가 넓어지고 사용자에게 보다 친숙한 시스템을 제공하기 위해서 플랜인지가 도움말 시스템에 적용되거나(Allen, 1983) 문장의 낱말을 이해 하는데(Allen, 1983), 생략(Allen, 1983 ; Carberry, 1985; Litman and Allen, 1987), 간접 표현 행동(Allen, 1983), 담화를 통한 말하는 사람의 흐름을 추적(Sidner.), 사용자 지식과 시스템 사이의 모순을 정확하고 완전하게 다루는 것(Pollack)들에 다루기 쉬운 알고리즘을 제공함으로써 적용되었다. Cohen 과 Levesque(1990)는 합리적인 상호작용과 통신이론에 대한 기초를 사용함으로써 의도(intention)를 형식화하는 logic 을 제공했다. perrault(1990)는 Speech set utterance 의 결과를 추출해서 애매한 말을 정확히 예측 할 수 있는 Speech acts 의 default logic 을 제시했다.

최근에 플랜인지 연구는 플랜인지에 대한 의미를 형식화하고 구체화 함으로써 플랜인지 문제에 대한 엄격하고 구체적인 해결방안을 제시하는데 집중되고 있다. 즉 특정 도메인에서 가능한 모든 사용자 액션과 액션의 관계를 플

랜라이브러리로 상세하게 구축하고(close-world model), 구축된 플랜 라이브러리 상에서 추적 함으로서 사용자의 플랜을 인지하는 방식이다. 플랜인지 알고리즘은 본 논문에서 상세하게 다루어지지 않았다

4. 플랜 라이브러리 구축 방법론

플랜 라이브러리는 플랜인지 에이전트의 지식베이스로서, 응용 도메인에서 가능한 모든 사용자의 액션과 연속적인 액션의 관계 및 의미를 표상 (Representation)하고 있어야 한다.

4.1 계층구조에 기초한 1 차원적 플랜 라이브러리 구축방법론의 문제점

하나의 플랜은 하나 이상의 액션으로 구성되며, 목표에 도달하기 위한 성공적인 사용자 플랜이란 사용자에게 의하여 취해지는 액션의 연속이 통사론적, 의미론적인 제약조건을 만족시키는 경우이다(2 장에서 Allen(1983) 표 참조). 계층구조에 기초한 플랜 라이브러리에서는 이미 언급한 바와 같이 모든 가능한 플랜들이 상세하게 구축되어 있어야 한다. 텍스트 에디터를 예로 들어 보자. 몇 개의 가능한 사용자 액션을 적어보면 다음과 같다.

- 액션 a1: 커서 정위치
- 액션 a2: 마우스 클릭
- 액션 a3: 마우스 움직임
- 액션 a4: 마우스 왼쪽버튼 누른채로 움직임
- 액션 a5: 텍스트 마크
- 액션 a6: 마우스 왼쪽버튼 다운
- 액션 a7: 메뉴 편집에서 “복사하기” Item 선택
- 액션 a8: 메뉴 편집에서 “잘라내기” Item 선택
- 액션 a9: 메뉴 편집에서 “붙이기” Item 선택
- 액션 a10: 키 다운/업
- 액션 a11: 마우스 왼쪽버튼 업

위 10 개의 액션을 가지고 가능한 플랜 라이브러리를 구축하여 보면 아래와 같다.

- 플랜 p1(커서 정위치): 커서 정위치
- 플랜 p2(커서 정위치): 마우스 움직임>마우스 클릭
- 플랜 p3(텍스트 에디팅): 커서 정위치>키 다운/업
- 플랜 p4(텍스트 에디팅): >키 다운/업
- 플랜 p5(텍스트 마크): 마우스 움직임>마우스 클릭>마우스왼쪽버튼 다운>마우스왼쪽버튼 누른채로 움직임>마우스왼쪽버튼 업

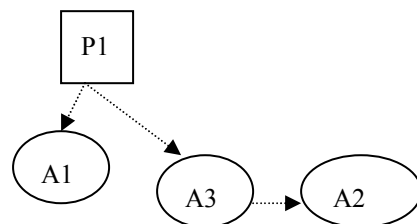
- 플랜 p6(텍스트 마크): 커서정위치> 마우스왼쪽버튼 다운>마우스왼쪽버튼 누른채로 움직임>마우스왼쪽버튼 업
- 플랜 p7(텍스트 복사): 마우스 움직임>마우스 클릭>마우스왼쪽버튼 다운>마우스왼쪽버튼 누른채로 움직임>마우스왼쪽버튼 업> 메뉴편집에서”복사하기”Item 선택>마우스움직임>마우스클릭>메뉴편집에서 “붙이기”Item 선택
- 플랜 p8(텍스트 복사): 커서 정위치>마우스왼쪽버튼누른채로움직임>마우스왼쪽버튼업>메뉴편집에서”복사하기”Item 선택>마우스움직임>마우스클릭>메뉴편집에서 “붙이기”Item 선택
- ...

위와 같이 단 10 개의 액션을 가지고 실제 사용자가 실행 가능한 성공적인 플랜을 구축하더라도 쉽게 10 여 개 이상의 플랜들이 생겨나며 액션의 수가 많아 질수록 플랜의 수 또한 기하급수적으로 늘어나게 되어 플랜라이브러리의 용량 또한 커지게 된다. 이는 결과적으로 플랜 추적을 어렵게 하는 요인으로 작용하게 될 뿐만이 아니라 플랜 라이브러리 구축에 있어서 비용을 증가시킨다. 또한 각각의 플랜들을 살펴보면 텍스트 복사를 위한 플랜 p7 에는 p5 가 Sub-Plan 으로 포함(Meet 또는 Include 관계)되어 있음을 알 수 있다. 또한 p5 에는 p2 가 포함되어 있다. 위와 같이 계층적인 1 차원 플랜 라이브러리에서는 많은 플랜들이 다른 플랜의 Sub-Plan 이 되어 중복되어 구축되는 것을 알 수 있다. 이러한 문제는 플랜 추적 및 인지를 더더욱 어렵게 만드는 요인으로 작용하는데, 예로 플랜 추적 엔진이 하나의 플랜이 인지되었을 때, 인지된 플랜이 다른 플랜의 Sub-Plan 인지 또는 인지의 종료와 새로운 플랜의 추적이 시작되어야 하는지 판단을 어렵게 한다.

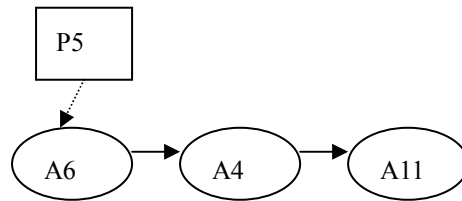
4.2 이차원적 플랜 라이브러리 구축

4.1 장에서 나타나는 문제점은 이차원적인 플랜인지 알고리즘과 2 차원적 플랜추적 알고리즘을 사용함으로써 해결될 수 있다. 2 차원 플랜인지 알고리즘에 대하여 본 논문에서는 다루어 지지 않는다. 2 차원적 플랜라이브러리 구축을 위하여 모든 액션이 먼저 4.1 에서와 같이 정의되어야 한다. 그리고 또한 각각의 플랜들이 하나의 단위로서 정의된다. 이에 따라 플랜 라이브러리는 다음과 같은 양상을 보이게 된다(4.1 장에 액션을 참고).

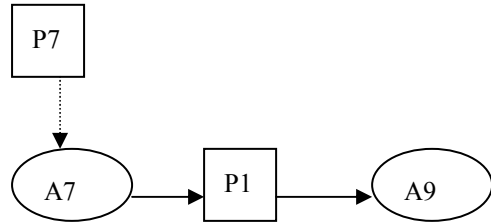
플랜 p1(커서위치): $p1 \ni \{a1, a2\}$



플랜 P5(텍스트마크): $p5 \ni \{a6 > a4 > a11\}$

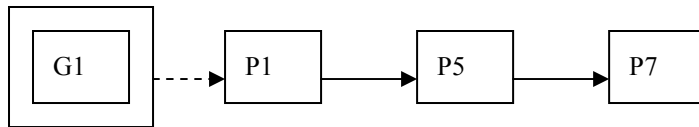


플랜 P7(텍스트복사): $p7 \ni \{a7 > p1 > a9\}$



이러한 플랜 단위에 따라 사용자의 목표(Goal)에 이르는 플랜 라이브러리를 구축하게 되는데, 예로 텍스트의 특정 부분을 복사하여 다른 곳에 붙이기를 하려는 사용자의 목표에 도달하기 위한 사용자 플랜은 다음과 같이 플랜 라이브러리에 표상된다.

목표 G1(텍스트복사): $G1 \supset \{pa > p5 > p7\}$



위에서 보는 바와 같이 2 차원적 플랜 라이브러리 구축 방법론을 적용하게 되면 플랜 라이브러리 구축을 매우 효율적으로 할 수 있으며, 또한 플랜 라이브러리가 간결하고 용량 또한 획기적으로 축소되어 플랜 추적 및 인지를 쉽게 할 수 있다.

4. 향후 연구방향

이러한 2 차원적 플랜 라이브러리를 활용하기 위하여 사용자 플랜 추적 및 인지 또한 2 차원적으로 동시에 행하여져야 한다. 즉 사용자의 액션은 특정한 플랜단위의 구성요소로서 하위 레벨 추적 엔진에서는 **Sub-Plan** 이라 할 수 있는 플랜단위의 추적이 실행되어 플랜단위를 인지하여야 하며, 상위 레벨

추적엔진에서는 하위 레벨 추적엔진이 인지한 플랜단위가 어떠한 사용자 목표를 지향하고 있는지 인지하여야 한다. 2 차원 플랜인지 엔진 개발에 대한 알고리즘들은 향후 특정 응용 도메인과 관련하여 연구 개발 될 것이다.

References

Allen, J.F.:1983, Maintaining Knowledge about Temporal Intervals. Communications of the ACM 26, 832-843.

Carberry,S.:1988,'Modeling the User's Plans and Goals'. Computational Linguistics 14(3), 23-37.

Carberry,s.:1990, 'Incorporating Default Inferences into Plan Recognition. Eighth National Conference on Artificial Intelligence,Boston,Ma.,pp.471-478.

Carver,N.F.,V.R.Lesser,and D.L.McCue:1984, 'Focusing in Plan Recognition'. Fourth National Conference on Artificial Intelligence, Austin,TX.

Cohen,R.,F.Song,B.Spencer,and P.van Beek:1991, 'Exploiting Temporal and Novel Information from the User in Plan Recognition' .User Modeling and User-Adapted Interaction 1(2),125-148.

Eller,R.and Carberry,S.:1991,A Meta-Rule Approach to Flexible Plan Recognition in Dialogue. UMUAI,forthcoming.

Litman,Diane 1985. Plan Recognition and Discourse Analysis: An Integrated Approach for Understanding Dialogues. Ph.D.Dissertation, University of Rochester.

Robert Jerrard, 'Person-System Interaction and Designing: Reestablishing Values', Design Issues, Vol. 4, No.2, Fall, 1993, pp.20-24

H. J. Bullinger and K.P.Faehrich, 'Symbiotic Man-Computer Interfaces and the User Assistant Concept' , Human-Computer Interaction, Proceedings of the First U.S.A. – Japan Conference on Human-Computer Interaction, Honolulu, Hawaii, August 18-20, 1984, pp.17-20