

강인 행동 계획의 자동 생성 방법

이상형, 차병근, 이상훈, 서일홍
한양대학교 정보통신학과

A automatic construction technique of Robust Behavior Plan

SangHyoun Lee, ByungGun Cha, Sanghoon Lee, Il Hong Suh
The Department of Information & Communications

Abstract

In this paper, we propose a planning algorithm which automatically generates robust behavior plans for service robots in the dynamically changing environments. The proposed method searches for paths to perform the given tasks in the physical space and the configuration space where tasks are described. And then, the characteristics of paths for successfully performed task are abstracted and generalized to build an ordered-tree structure. The resulting robust behavior plans guarantee that the given tasks are successfully performed. The validity of our method is tested by simulation work for a pushing-box task.

동적으로 변화하는 상황에서 태스크를 수행하기 위해서는 상황에 유동적으로 대처할 수 있는 행동 계획이 필요하다. 때문에 본 논문에서는 로봇이 동적으로 변화하는 환경에서 태스크를 수행할 수 있도록 reactive plan의 개념[2]을 도입하여 로봇이 태스크를 수행하는 동안 계획의 수정 없이 바로 필요한 행동을 선택할 수 있도록 강인한 행동 계획을 자동으로 생성하는 방법을 제안한다. 강인 행동 계획(Robust behavior plan)은 동적으로 변화하는 환경에서 로봇의 태스크 수행을 보장한다. 그리고 이 방법을 'Pushing -box' 태스크에 적용하고, 이를 모의 실험 환경에서 검증하는 작업을 수행한다.

I. 서론

로봇은 지능을 갖기 위해 행동 계획(plan)과 학습 능력(learning)이 요구된다. 특히, 행동 계획이란 태스크를 수행할 수 있는 일련의 순서를 갖는 행동과 행동을 수행할 수 있는 조건을 말한다. 지금까지의 행동 계획은 일반적으로 planning에 의해 생성되거나 또는 사람이 직접 생성해 왔다. Planning은 현재 상황에서부터 목표까지 계획을 작성해야 하기 때문에 많은 시간과 비용이 필요하고, 동적으로 변화하는 환경에 빠르게 대처하는 것이 힘들고[1], 사람이 행동 계획을 직접 작성하는 경우에는 미리 특정 상황에 따른 행동을 정의해 주어야 한다. 하지만 환경이 동적으로 변화하는 상황에 대해 행동 계획을 미리 작성하는 것은 매우 어렵다. 이렇듯

II. Robust Behavior Plan

로봇이 태스크를 수행하기 위해서는 태스크를 로봇이 수행할 수 있는 도메인으로 나타내는 작업이 필요하다. 때문에 본 논문에서는 로봇 공학에서 흔히 말하는 상태 공간(configuration space)의 개념을 도입하여 Task configuration space를 생성하였다[3]. 하지만 Task configuration space는 상태 공간과는 달리 서로 비독립적인 특징을 가질 수 있기 때문에 태스크를 수행하는데 있어서 발생하는 물리적 제약 조건을 설명해 주는 물리 공간(Physical space)이 필요하다. Task configuration space와 Physical space를 합쳐서 태스크를 설명한 공간을 Task description space(TDS)라고 부른다.

TDS 공간에서 이동 경로는 태스크의 수행 경로를 의미한다. 때문에 이동 경로를 찾는 것이 중요하다. 본 논문에서는 이동 경로를 찾기 위해 리스크(risk)값을 고려한 변형된 A* 알고리즘을 사용해 성공적으로 태스크를 완수한 노드(node)들을 수집하였다. 수집한 노드들을

. 본 연구는 정보통신부에서 수행되고 있는 'URC를 위한 컴포넌트 기술 개발 및 표준화'과제의 지원을 받아 수행되었습니다.

이용해 태스크 설명 공간 상태 벡터(TDS State vector)를 생성하고, 이 공간 상태 벡터에서 하위 목표(sub goal)를 추출해 내기 위해 평균 빈도수 이하의 공간 상태 벡터를 하위 목표 후보에서 제외시킨다. 이렇게 구해진 하위 목표 후보들을 하위 목표로 일반화시키기 위해 공간 상태 벡터를 이용해 상태 전위 튜플 (State Transition Tuple)을 생성하고 이 튜플들로부터 특징¹ (feature)을 추출해 낸다. 이 과정이 끝난 후에 TDS vector 값을 비교하여 추상화시키는 과정을 거쳐 더 일반화시킨다.

앞에서 전개한 과정을 통해 만들어진 클러스터들을 목표 상태(Goal State)와 같은 차이가 나는 상태(State)들을 클러스터링 과정을 통해 최종적으로 강인 행동 계획을 완성한다. 이렇게 생성된 강인 행동 계획은 행동 선택 매카니즘(Action Selection Mechanism)을 통해 로봇이 수행하게 된다[4].

III. 구현

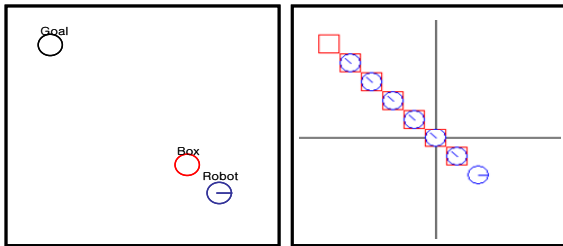


그림 1 실험 환경 및 A* 알고리즘 수행 결과

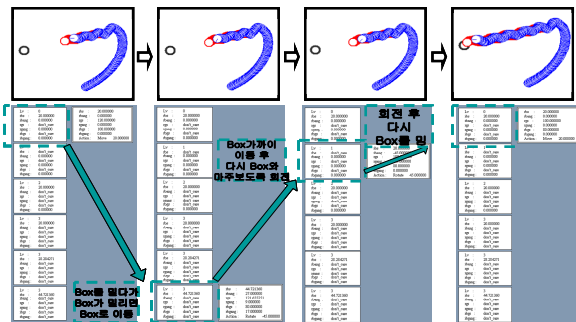


그림 2 Pushing Box Task 에서 Robust Behavior Plan 이 선택되는 예

앞에서 설명한 알고리즘을 이용해 얻어진 강인 행동 계획을 시뮬레이션을 통해 검증하였다. 이 때, 시뮬레이션 환경은 강인 행동 계획을 만들기 위해 사용한

¹ 특징(feature) : State transition tuple 들이 지닌 상태 변화와 상태변화 방향성

환경과 동일하게 하였다. 하지만 검증을 위한 샘플은 새로 생성하여 실험을 진행하였다.

IV. 결론 및 향후 연구 방향

본 논문에서는 강인 행동 계획을 자동으로 생성하는 방법을 제안하였다. 이 방법은 계획의 수정 없이 바로 필요한 행동을 선택할 수 있어 planning 과 달리 시간과 비용의 소모가 많지 않고 바로 필요한 행동을 선택할 수 있다. 때문에 계획의 변경이 용이하고 불확실한 환경의 변화에 대한 대처가 빠르다. [표 1]에서 보는 바와 같이 강인 행동 계획은 동적으로 변화하는 환경과 불확실한 로봇의 움직임을 가지고 있는 상황에서 planning 으로 태스크를 수행하는 것보다 높은 수행 능력을 발휘한 것을 확인할 수 있었다.

표 1 실험 환경 및 A* 알고리즘 수행 결과

	평균 성공 확률	성공 시 평균 수행 시간(sec)
Robust Behavior Plan	97%	6.03
Planning	82%	16.83
無	4%	6.17

< 636회 실험 >

향후 발전 계획은 첫째, 본 논문에서 제안한 강인 행동 계획의 자동 생성 방법으로 만들어진 각각의 계획을 합쳐서 더 복잡한 태스크를 수행할 수 있는지 여부를 확인한다. 둘째, 로봇의 학습과 연결될 수 있도록 한다.

참고문헌

- [1] S. Russell, Peter Norving, "Artificial Intelligence : A Modern Approach," 2nd Edition Prentice Hall, 2003
- [2] R.Brooks, "A Robust Layered Control System for a Mobile Robot," In IEEE Journal of Robotics and Automation, pp.14-23, April, 1986
- [3] John J. Craig, "Introduction to Robotics," 2nd Mechanics and Control, Addison Wesley Longman.
- [4] I.H Suh, S. Lee, W.Y. Kwon, and Y.-J. Cho, "Learning of Action patterns and Reactive Behavior Plans via a Novel Two-Layerd Ethology-Based Action Selection Mechanism," 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp.123201238, August 2-6, 2005, Edmonton, Canada.