

프리아크 개념을 이용한 로봇의 충돌회피 동작 계획

Collision-Free Motion Planning of a Robot Using Free Arc Concept

이 석 원, 남 윤 석, 이 범 희
(Seok-Won Lee, Yun-Seok Nam, and Beom-Hee Lee)

Abstract : This paper presents an effective approach to collision-free motion planning of a robot in the work-space including time-varying obstacles. The free arc is defined as a set composed of the configuration points of the robot satisfying collision-free motion constraint at each sampling time. We represent this free arc with respect to the new coordinate frame centered at the goal configuration, and therefore the collision-free path satisfying motion constraint is obtained by connecting the configuration points on the free arc at each sampling time from initial to goal configuration point. When the collision-free path of a robot in the C-space is determined from the sequence of free arcs, the optimality is determined by the performance index. Therefore, the complicated collision-free motion planning problem of a robot is transformed to a simplified Sub-Optimal Collision Avoidance Problem(SOCAP). We analyze the completeness of the proposed approach and show that it is partly guaranteed using the backward motion. Computational complexity of our approach is analyzed theoretically and practical computation time is compared with that of the other method. Simulation results for two SCARA robot manipulators are presented to verify the efficacy of the proposed method.

Keywords : robot, collision-free motion planning, C-space, free arc, sub-optimal collision avoidance problem

I. 서론

로봇의 충돌회피 동작 계획 방법들은 로봇과 장애물의 종류에 따라 분류할 수 있다. 로봇은 이동 로봇 및 로봇 매니퓰레이터로 나누어 볼 수 있는데, 이동 로봇과 낮은 자유도(degrees of freedom . d.o.f.)를 가진 로봇 매니퓰레이터의 고정 장애물 회피 문제에 대해서는 완전도(completeness)를 보장하면서 효율적인 알고리즘들이 다양하게 제안되었고[1,2], 이 알고리즘들의 계산 복잡도(computational complexity)를 분석하려는 노력도 있었다[3]. 로봇의 고정 장애물 회피 분야에서는 장애물 정보가 부분적으로 주어지거나 전혀 주어지지 않는 동적 환경 속에서 적용 가능한 알고리즘을 찾거나[4,5], 높은 자유도를 가진 매니퓰레이터에 대한 해법을 찾는 것 [6,7] 등이 현재에는 주요한 과제이다.

그러나, 시변 장애물을 포함한 환경에서의 로봇의 충돌회피 동작 계획 문제는 상대적으로 연구가 많이 진행되어 있지 않은 상태이며 지금까지 제안된 접근 방법들 의 대부분은 고정 장애물 회피 동작 계획 방법을 용용하거나 확장한 방법들이다. 특히, 두 로봇 시스템의 충돌회피 동작 계획 문제는 일반적인 시변 장애물 회피 문제인 동시에 로봇 매니퓰레이터들의 기구학과 동력학이 복잡하기 때문에 단일 로봇의 시변 장애물에 대한 충돌회피 동작 계획 문제와는 다른 접근 방법을 필요로 한다.

Lee와 Lee[8]는 듀얼 로봇(dual-robot)의 충돌회피 문제를 연구하였다. 그들은 두 로봇들의 경로와 궤적 정보

를 이용하여 충돌 지도(collision-map)를 구성함으로써 우선 순위가 낮은 로봇의 속도 프로파일을 변형시키는 시간 조절 알고리즘을 제안하였다. 이 연구는 나수 로봇 시스템의 충돌회피 동작 계획을 위한 실제적이고 사용하기 쉬운 방법을 제공하였으나 주어진 경로의 변경이 허용되지 않기 때문에 그 경로 위에서 충돌회피 궤적이 존재하지 않을 수도 있다는 단점이 있다.

Shin과 Bien[9]은 정해진 경로상의 위치를 지정하는 변수들로 구성되는 조절 공간(coordination space)의 개념을 도입하고 그 조절 공간상에서 충돌 영역을 정의하였다. Bien과 Lee[10], 그리고 Lee[11]는 조절 공간상에서 두 대의 로봇 매니퓰레이터를 위한 충돌회피 및 최적 시간 궤적 계획의 해석적인 방법을 제안하였다. 이 방법은 Kant와 Zucker[12]에 의해 제안된 경로-속도 분해 기법을 기반으로 한 것으로서, 로봇의 충돌회피 동작 계획 문제를 고정 장애물에 대한 경로 계획 문제와 시변 장애물에 대한 속도 계획 문제로 분리하고 출발점으로부터 도착점까지의 최소 경로를 찾기 위하여 VGRAPH 알고리즘을 사용하였다. 즉, 그들은 우선 고정 장애물만을 고려하여 두 로봇의 경로들을 독립적으로 계획한 후에 한 로봇의 출발 지연 시간을 최소화시키는 방향으로 두 로봇들의 속도 프로파일을 조절함으로써 두 로봇 사이의 충돌을 회피하는 방법을 제안하였다. 그러나므로, 이상의 방법들[9-12]은 다른 로봇 이외의 시변 장애물을 고려할 수 없고 경로 계획 단계에서 기존의 고정 장애물 회피 방법을 사용해야 한다.

한가지 다른 접근 방법으로서 Lozano-Perez와 Wesley[13]에 의해 제안된 형상 공간(configuration space : C-space)상에서 로봇의 시변 장애물에 대한 충돌회피 동작 계획 문제를 해결할 수 있다. 형상 공간을

이용한 방법에서 로봇은 한 점으로 표현되며 장애물들은 그 로봇의 형상 공간으로 변형시킬 수 있다. 그러므로, 로봇의 충돌회피 동작 계획 문제는 초기 형상으로부터 목표 형상까지 형상 공간 장애물들과의 충돌을 회피 할 수 있는 경로를 찾아내는 문제로 단순화된다. 형상 공간에서의 방법은 여러가지 장점을 가지고 있는 반면에 근본적인 문제점을 갖고 있는데, 첫째, 로봇의 자유도가 증가함에 따라 계산 복잡도가 지수적으로 증가한다는 점과 둘째, 형상 공간 장애물을 구하는 데 너무 많은 시간이 걸린다는 점이다. 특히, 시변 형상 공간 장애물의 경우에는 계산 시간이 훨씬 더 많이 걸린다. 시변 장애물을 포함한 환경에서 이러한 형상 공간을 사용한 방법들의 문제를 해결하기 위해서, 본 논문에서는 시변 형상 공간 장애물을 효율적으로 구성하는 방법을 제안하고자 한다.

II장에서는 문제 구성에 필요한 개념들을 정의하고 로봇의 시변 장애물 회피 문제를 일반적으로 구성한다. III장 1절에서는 M 차원 형상 공간에서 프리아크의 개념과 표현 방법을 설명하고, III장 2절에서는 프리아크 개념을 이용하여 충돌회피 동작 계획 문제를 준최적 충돌 회피 문제(Sub-Optimal Collision Avoidance Problem : SOCAP)로 변환시키는 방법을 기술한다. IV장에서는 로봇의 동작 제한 조건 및 충돌 제한 조건을 고려하여 2 차원 형상 공간에서 프리아크를 찾아내는 과정을 유도한다. V장 1절에서는 제안한 방법으로 완전도를 보장할 수 있는 방법을 알아보고, V장 2절에서는 기존의 슬라이스 프로젝션 방법과 계산 시간을 비교한다. VI장에서는 제안한 방법을 두 대의 SCARA형 로봇 매니퓰레이터들의 충돌회피 동작 계획 문제에 적용한 모의 실험 결과를 분석하고, 연구 결과에 대한 결론을 VII장에서 기술한다.

II. 로봇의 충돌회피 동작 계획 문제의 구성

로봇의 충돌회피 동작 계획 문제를 구성하고 그 해법을 제안하기 위해서 사용한 기호들은 다음과 같다.

M : 로봇의 자유도

$q_i(t)$: 시작 t_0 에서 로봇의 i 번째 일반화 관절 변수 값

$q(t) = (q_1(t), q_2(t), \dots, q_M(t)) \in R^M$: M 차원 형상 공간에서 로봇의 형상점

q_0 : 로봇의 초기 형상

q_N : 로봇의 목표 형상

t_i : 로봇의 초기 형상으로부터의 출발 시간

t_f : 로봇이 목표 형상에 도착해야 할 시간

$T_f = t_f - t_i$: 로봇의 총 운동 시간

Δt : 샘플링 기간(sampling period)

$N = T_f / \Delta t$: 총 샘플링 회수

로봇의 충돌회피 동작을 계획할 때에도 기본적으로 로봇의 동작은 작업 공간을 벗어날 수 없으므로 먼저 형상 공간에서 로봇의 작업 공간을 다음과 같이 정의 한다.

정의 2 1. 로봇의 형상 공간에서의 작업 공간 CS 는

다음과 같이 정의된다.

$$CS = \{q | q_{i,\min} \leq q_i \leq q_{i,\max}, i = 1, 2, \dots, M\} \quad (1)$$

여기서 $q_{i,\min}$ 과 $q_{i,\max}$ 는 각각 로봇의 i 번째 관절의 동작 범위의 하한값과 상한값이다.

Lee[14]는 로봇의 동작 계획시 부과할 수 있는 제한 조건들을 환경 제한 조건과 매니퓰레이터 제한 조건의 두 종류로 구분하였다. 본 연구에서는 환경 제한 조건으로서 충돌 제한 조건을 고려하고 매니퓰레이터 제한 조건으로서 캐직의 유연 제한 조건을 문제에 포함시킨다. 충돌 제한 조건은 축 동작의 위치를 충돌회피 형상 공간으로 제한시킨다. 본 연구에서는 한 로봇 매니퓰레이터의 모든 링크들과 시변 장애물들 사이에 발생하는 모든 충돌을 고려한다. 그러나, 한 로봇이 초기 또는 목표 형상에 있을 때에는 다른 장애물과 충돌하지 않는다는 가정이 필요하다. $OS(t)$ 를 형상 공간 장애물에 포함된 모든 형상점들의 집합이라 하면 충돌 제한 조건은 다음과 같이 정의된다.

정의 2 2 충돌 제한 조건은 다음과 같이 정의된다.

$$q(t) \in CN_c(t), \text{ for all } t \in [t_0, t_f] \quad (2)$$

여기서

$$CN_c(t) = \{q(t) | q(t) \in CS - OS(t)\}. \quad (3)$$

위의 정의에서 $CS - OS(t)$ 는 로봇 매니퓰레이터의 충돌 회피 형상 공간을 가리킨다.

캐직 유연 제한 조건은 축 동작을 위한 캐직의 유연성을 유지하기 위하여 축 동작의 속도와 가속도를 제한한다. VB_i 와 AB_i 를 각각 i 번째 축 동작의 속도 한계값과 가속도 한계값이라 하면 유연 제한 조건은 다음과 같이 정의된다

정의 2 3 유연 제한 조건은 다음과 같이 정의된다.

$$q(t) \in CN_s(t), \text{ for all } t \in [t_0, t_f] \quad (4)$$

여기서

$$CN_s(t) = \{q(t) | \dot{q}_i(t) \leq VB_i, \ddot{q}_i(t) \leq AB_i, i = 1, 2, \dots, M\}. \quad (5)$$

이상에서 기술한 로봇의 형상 공간에서의 작업 공간과 제한 조건들에 대한 정의를 이용하여 로봇의 시변 장애물에 대한 충돌회피 동작 계획 문제를 구성한다. 형상 공간에서의 충돌회피 동작 계획 문제는 초기 시작 t_0 에 q_0 의 형상을 갖고 있는 로봇이 동작 제한 조건을 만족시킬면서 형상 공간 장애물들을 피해 시작 t_f 에 목표 형상 q_N 을 갖도록 동작을 계획하는 것으로서, 다음과 같이 표현된다.

충돌회피 동작 계획 문제

$t \in [t_0, t_f]$ 인 모든 t 에서 다음의 제한 조건과 경계 조건을 만족하는 $q(t)$ 를 구한다.

- 1) 제한 조건 : $q(t) \in CS$ and $q(t) \in CN_c(t)$
and $q(t) \in CN_s(t)$
- 2) 경계 조건 $q(t_0) = q_0, q(t_f) = q_N$

III. 프리아크 개념을 이용한 로봇의 충돌 회피 동작 계획 기법

II장에서 기술한 문제를 풀기 위하여 형상 공간에서 프리아크의 개념을 재안한다. 프리아크는 매 샘플링 시간마다 충돌회피 동작 제한 조건을 만족하는 형상점들로 구성되는 집합으로 정의한다. 그러므로, 동작 제한 조건을 만족하는 충돌 회피 경로는 초기 형상점으로부터 목표 형상점까지 매 샘플링 시간의 프리아크상에 존재하는 형상점을 연결함으로써 구해진다. II장에서 정의한 충돌회피 동작 제한 조건에 대한 정의들을 이용하여 프리아크를 다음과 같이 정의한다.

정의 3.1 시간 t_k 에서의 프리아크는 다음과 같이 정의된다.

$$FA_k = \{q_k \mid q_k \in CN_s(t) \cap CN_c(t)\} \quad (6)$$

여기서 q_k 는 $(q_{1k}, q_{2k}, \dots, q_{Nk})$ 로 구성된 시간 t_k 에서의 형상점이다.

그리면, 남은 문제는 프리아크를 어떻게 효과적으로 찾아내고 표현하는가 하는 문제이다. 이 문제를 해결하기 위하여 몇 개의 가정들을 기본으로 하는 방법을 제안한다. 우선 로봇의 M 차원 형상 공간을 목표 형상을 중심으로 하는 N 개의 M 차원 구의 경계를 따라 이산화시킨다. 이때, N 은 이동 시간 T , 동안의 샘플링 회수이다. 우선 매 샘플링 시간마다 가능한 형상점들이 M 차원 동심구의 경계상에 존재한다는 가정이 필요하다. 이 가정으로 인해 시간 t_k 에서의 프리아크는 각각의 M 차원 동심구의 경계의 부분 집합으로 표현된다. 시간 t_k 에서 M 차원 동심구의 경계식은 다음과 같다.

$$\|q_k - q_n\| = r_k \quad (7)$$

여기서 r_k 는 M 차원 구의 반지름을 가리킨다.

프리아크를 효과적으로 표현하기 위해서 목표 형상점을 중심으로 하는 또 다른 좌표계를 도입한다. 이 좌표계는 2차원 형상 공간의 경우 극좌표계의 형태를 가지며 3차원 형상 공간의 경우 구형 좌표계의 형태를 갖는다. 모든 차원의 형상 공간에서 형상점 q_k 의 M 개 성분은 M 차원 구의 반지름인 r_k 와 그 좌표계를 중심으로 표현되는 $M-1$ 개의 다른 성분으로 구성된 벡터 ψ_k 에 의해서 다음과 같이 표현된다.

$$q_{ik} = f_i(r_k, \psi_k), \quad i=1, 2, \dots, M. \quad (8)$$

여기서 r_k 는 매 샘플링 시간마다 고정된 상수이며 M 차원의 벡터 ψ_k 가 로봇의 형상을 결정한다. 그러므로, 프리아크는 ψ_k 의 범위로써 표현할 수 있으며 충돌회피 동작 계획 문제를 $M-1$ 차원 문제인 것처럼 다룰 수 있다. 이상에서 설명한 표현 방법을 기본으로 시간 t_k 에서 프리아크를 다음과 같이 다시 쓸 수 있다.

$$FA_k = \{q_k \mid (f_1(r_k, \psi_k), f_2(r_k, \psi_k), \dots, f_M(r_k, \psi_k)) \in CN_s(t) \cap CN_c(t)\}. \quad (9)$$

다음에서 2차원 형상 공간의 경우에 대하여 프리아크의

기본적인 형태를 알아본다.

Case 1 : 2차원 형상 공간 ($M = 2$)

2차원 형상 공간에서는 목표 형상점을 중심으로 한 새로운 좌표계로서 극좌표계를 선택한다. 이 경우에 M 차원 동심구의 경계는 q_n 을 중심으로 하는 동심원이 되며 프리아크상의 형상점은 다음과 같이 표현된다.

$$\begin{aligned} q_{1k} &\equiv f_1(r_k, \psi_k) = r_k \cos \theta_k + q_{1N}, \\ q_{2k} &\equiv f_2(r_k, \psi_k) = r_k \sin \theta_k + q_{2N} \end{aligned} \quad (10)$$

여기서 $\psi_k = [\theta_k]$ 그리고, $\theta_k \in [0, 2\pi]$.

그림 1은 목표 형상점을 원점으로 하는 극좌표계에 대하여 r_k 와 θ_k 에 의해서 표현되는 형상점 q_k 를 보여준다.

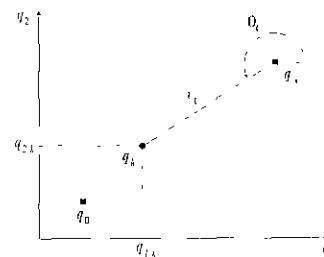


그림 1. 2차원 형상 공간에서의 극좌표계.

Fig. 1. The polar coordinate frame attached to the goal configuration.

프리아크 FA_k 는 충돌회피 동작 제한 조건을 만족하는 형상점들의 집합으로 정의되었기 때문에 충돌회피를 위한 동작 계획 문제는 각 샘플링 시간의 프리아크의 집합 $\{FA_0, \dots, FA_k, \dots, FA_N\}$ 을 찾아내는 문제로 단순화 된다. 각 샘플링 시간의 프리아크의 집합을 찾아내기만 하면 $M-1$ 차원 벡터들의 집합 $\{\psi_0, \dots, \psi_k, \dots, \psi_N\}$ 을 결정할 수 있다. 연속해서 형상 공간에서의 충돌회피 경로 $\{q_0, \dots, q_k, \dots, q_N\}$ 는 ψ_k 로부터 q_k 로의 변환식을 이용하여 구할 수 있다.

성능 지수를 $\{\psi_0, \dots, \psi_k, \dots, \psi_N\}$ 의 함수로 선정함으로써 프리아크의 집합 $\{FA_0, \dots, FA_k, \dots, FA_N\}$ 으로부터 $\{\psi_0, \dots, \psi_k, \dots, \psi_N\}$ 를 선택할 때 최적화를 고려할 수 있으므로 출최적 충돌회피 문제를 다음과 같이 구성할 수 있다.

$$\begin{aligned} \text{Find } & \{\psi_0, \dots, \psi_k, \dots, \psi_N\} \quad \text{s.t.} \\ \text{minimize } & G(\psi_0, \dots, \psi_k, \dots, \psi_N) \\ \text{subject to } & (f_1(r_k, \psi_k), f_2(r_k, \psi_k), \dots, f_M(r_k, \psi_k)) \\ & \in CN_s(t) \cap CN_c(t) \\ & \text{for } k = 0, 1, \dots, N. \end{aligned}$$

IV. 실제 형상 공간상의 문제에 대한 적용 방법

III장에서는 M 차원 형상 공간에서 프리아크의 개념에 대하여 설명하였다. 이번 장에서는 2차원 형상 공간의 경우에 프리아크를 찾아내는 방법을 설명한다. (6)에서 프리아크는 다음과 같이 정의되었다.

$$FA_k = \{q_k \mid q_k \in CN_s(t)\} \cap \{q_k \mid q_k \in CN_c(t)\} \quad (11)$$

여기서, 궤적의 유연 제한 조건을 만족하는 프리아크를 $FA_{s,k}$ 라 하고 충돌 제한 조건을 만족하는 프리아크를 $FA_{c,k}$ 라 하면 다음과 같이 쓸 수 있다.

$$FA_k = FA_{s,k} \cap FA_{c,k} \quad (12)$$

$FA_{c,k}$ 를 구하는 과정은 IV.1절과 IV.2절에서 전개하고 $FA_{s,k}$ 를 구하는 과정은 IV.3절에서 설명한다.

1. 작업 공간에 포함되는 초기 프리아크의 구성

충돌 제한 조건을 만족하는 프리아크를 찾아내는 방법을 다음 절에서 설명하기 전에 이 절에서는 작업 공간에 포함된 초기 프리아크를 구하는 과정을 유도한다. 인접한 M 차원 구들 사이의 거리가 모두 같은 경우에 k 번째 샘플링 시간에서의 로봇 매니퓰레이터의 형상 q_k 는 다음 식을 만족해야 한다.

$$\|q_k - q_0\| = \|q_0 - q_N\| \cdot (N-k)/N \quad (13)$$

2차원 형상 공간의 경우에 (13)을 만족하는 $q_k = (q_{1,k}, q_{2,k})$ 의 집합은 중심이 q_N 에 있고 반지름이 $\|q_0 - q_N\| \cdot (N-k)/N$ 인 원으로 그려진다. (13)을 만족하는 q_k 의 집합을 간단한 형태로 표현하기 위하여 극좌표계를 도입한다. 이 원을 q_N 에 원점을 갖는 극좌표계를 중심으로 표현하면 (10)에서처럼 반지름 r_k 와 θ_k 의 범위로써 나타낼 수 있다. 여기서 r_k 는 다음과 같은 식을 만족한다.

$$r_k = \|q_0 - q_N\| \cdot (N-k)/N \quad (14)$$

이 동심원을 C_k 로 표현하면 C_k 는 다음과 같은 형상 공간 점들의 집합으로 정리할 수 있다.

$$C_k = \{q_k = (r_k \cos \theta_k + q_{1,N}, r_k \sin \theta_k + q_{2,N}) \mid r_k = \|q_0 - q_N\| \cdot (N-k)/N \text{ and } 0 \leq \theta_k < 2\pi\} \quad (15)$$

그런데, 작업 공간 밖의 형상들은 의미가 없기 때문에 다른 제한 조건들을 고려하기 전에 이 동심원들은 작업 공간의 경계선으로 잘라내야 한다. (1)로부터 2차원 형상 공간에서의 작업 공간은 다음과 같다

$$CS = \{q \mid q_{1,\min} \leq q_i \leq q_{1,\max}, i=1,2\} \quad (16)$$

그리면, 충돌 및 동작 제한 조건을 고려하지 않은 작업 공간에 포함된 초기 프리아크 $FA_{CS,k}$ 는 다음과 같이 유도된다.

$$FA_{CS,k} = CS \cap C_k \quad (17)$$

(15)와 (16)으로부터 프리아크 $FA_{CS,k}$ 는 다음과 같이 θ_k 의 범위로써 나타낼 수 있다.

$$FA_{CS,k} = \{q_k \mid \theta_{CS}^{\min}(k) \leq \theta_k \leq \theta_{CS}^{\max}(k)\} \quad (18)$$

여기서 $\theta_{CS}^{\min}(k)$ 와 $\theta_{CS}^{\max}(k)$ 는

- 1) 다음 식들을 만족하고

$$\begin{aligned} q_{1,\min} &\leq r_k \cos \theta_k + q_{1,N} \leq q_{1,\max} \\ q_{2,\min} &\leq r_k \sin \theta_k + q_{2,N} \leq q_{2,\max} \end{aligned} \quad (19)$$

- 2) 다음 식에서 얻어지는 θ_0 를 포함하는 θ_k 의 구간

을 구함으로써 계산할 수 있다.

$$\begin{aligned} r_0 \cos \theta_0 + q_{1,N} &= q_{1,0} \\ r_0 \sin \theta_0 + q_{2,N} &= q_{2,0} \end{aligned} \quad (20)$$

θ_0 가 $FA_{CS,k}$ 에 포함되어야 하는 이유는 θ_0 를 포함하지 않는 다른 구간은 동작 계획 문제에서 물리적 의미를 갖지 못하기 때문이다. 프리아크 $FA_{CS,k}$ 의 기본적인 개념은 그림 2에서 살펴 볼 수 있다.

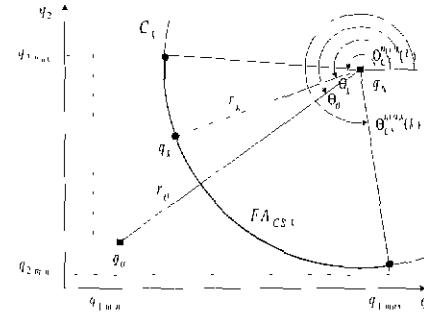


그림 2. 작업 공간에 포함되는 초기 프리아크 $FA_{CS,k}$ ($q_k = (q_{1,k}, q_{2,k})$)

Fig. 2. Initial free arc $FA_{CS,k}$ included in the work-space.

2. 충돌 제한 조건을 만족하는 프리아크의 구성

충돌 제한 조건의 정의를 고려하면 충돌 제한 조건을 만족하는 프리아크는 다음과 같이 쓸 수 있다.

$$FA_{c,k} = FA_{CS,k} - OS_k \quad (21)$$

여기서 OS_k 는 시간 t_k 에서 형상 공간 장애물과 충돌하는 모든 형상점들의 집합을 나타낸다. 매 샘플링 시간마다 로봇의 형상을 $FA_{CS,k}$ 상으로 제한함으로써 $FA_{CS,k}$ 를 따라 로봇의 충돌을 검사하면 시변 형상 공간 장애물을 구성할 수 있다. θ_k 의 이산화 분해능을 선정하면 매 샘플링 시간마다 형상 공간 장애물에 포함되는 형상점들의 집합을 다음과 같이 구할 수 있다.

$$OS_k = \{q_i \mid q_i \in \bigcup_{i=1}^{P(k)} OS_k(i)\} \quad (22)$$

여기서 $OS_k(i)$ 는 시간 t_k 에서의 고정 장애물 또는 시간 구간 $[t_k, t_{k+1}]$ 동안 시변 장애물이 쓸고 지나간 영역 (swept region)과의 i 번째 충돌 영역을 가리키고, $P(k)$ 는 시간 t_k 에서의 충돌 영역의 개수를 가리킨다. OS_k 는 θ_k 의 $P(k)$ 개 구간으로 구성되어 있기 때문에 다음과 같은 형태의 OS_k 를 얻을 수 있다.

$$OS_k = \{q_i \mid \theta_{OS}^{\min}(k,i) \leq \theta_k \leq \theta_{OS}^{\max}(k,i), i=1,2,\dots,P(k)\} \quad (23)$$

(18)과 (23)을 대입하면 $FA_{c,k}$ 는 다음과 같이 정리된다.

$$FA_{c,k} = \{q_k \mid \theta_{CS}^{\min}(k) \leq \theta_k < \theta_{OS}^{\min}(k,1), \theta_{OS}^{\max}(k,1) < \theta_k < \theta_{OS}^{\min}(k,2), \dots, \theta_{OS}^{\max}(k,P(k)) < \theta_k \leq \theta_{CS}^{\max}(k)\} \quad (24)$$

인접한 프리아크 사이에 존재하는 잠재적인 충돌 영

역을 고려하여 안전 동작을 위한 여유각으로 θ_{margin} 을 설정한다. 그러면, (24)는 다음과 같은 형태로 정리할 수 있다.

$$FA_{C,k} = \{q_k \mid \theta_{CAC}^{\min}(k,i) < \theta_i < \theta_{CAC}^{\max}(k,i), i=1, \dots, P(k)+1\} \quad (25)$$

여기서

$$\begin{aligned} \theta_{CAC}^{\min}(k,1) &= \theta_{CS}^{\min}(k) \\ \theta_{CAC}^{\min}(k,i) &= \theta_{OS}^{\max}(k,i-1) + \theta_{margin} \text{ for } i=2,3,\dots,P(k)+1 \\ \theta_{CAC}^{\max}(k,P(k)+1) &= \theta_{CS}^{\max}(k) \\ \theta_{CAC}^{\max}(k,i) &= \theta_{OS}^{\max}(k,i) - \theta_{margin} \text{ for } i=1,2,\dots,P(k) \end{aligned}$$

샘플링 시간 t_k 에서 두 곳의 충돌 영역이 존재할 때, $\theta_{margin}=0$ 인 프리아크 $FA_{C,k}$ 는 그림 3과 같다.

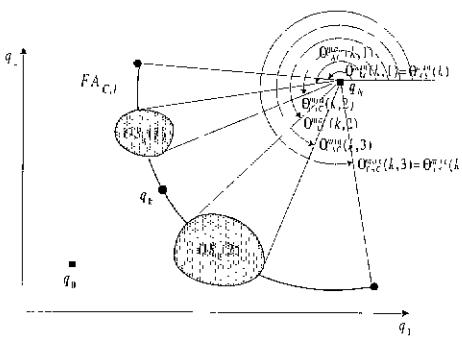


그림 3. 충돌 제한 조건을 만족하는 프리아크 $FA_{C,k}$ ($q_k = (q_{1,k}, q_{2,k})$).

Fig. 3. Free arc $FA_{C,k}$ satisfying collision constraints.

3. 유연 제한 조건을 만족하는 프리아크의 구성

이번 절에서는 (5)에 의해 표현되어지는 축 궤적에 대한 유연 제한 조건을 프리아크에 대한 제한 조건으로 변형시킨다. (5)로부터 축 속도 제한에 대한 다음과 같은 [14]들을 얻을 수 있다.

$$\dot{q}_{i,k}^{\min} \leq \dot{q}_{i,k} \leq \dot{q}_{i,k}^{\max}; i=1,2,\dots,M \quad (26)$$

여기서

$$\dot{q}_{i,k}^{\min} = \max\{-VB_i, -AB_i \cdot \Delta t + \dot{q}_{i,k-1}\}, \quad (27)$$

$$\dot{q}_{i,k}^{\max} = \min\{VB_i, AB_i \cdot \Delta t + \dot{q}_{i,k-1}\}. \quad (28)$$

$k=0 \sim N$ 등안의 $\dot{q}_{i,k}^{\min}$ 의 상한치를 $\dot{q}_{i,\max}$ 이라 하고 $\dot{q}_{i,k}^{\max}$ 의 하한치를 $\dot{q}_{i,\min}$ 라 하자. 축에 관계없이 $|\dot{q}_{i,\max}|$ 과 $|\dot{q}_{i,\min}|$ 이 $\dot{q}_{i,\max}$ 라는 같은 값을 갖는다고 가정하면 로봇은 시간 Δt 등안에 형상 공간에서 q_k 로부터 다음 영역 내에서 움직일 수 있다.

$$|q_{i,k} - q_{i,k+1}| \leq \dot{q}_{i,\max} \Delta t, i=1,2,\dots,M \quad (29)$$

2차원 형상 공간에서는 로봇의 도달 가능한 영역 (reachable region)이 중심이 q_k 에 있고 한 변의 길이가 $2\dot{q}_{i,\max} \Delta t$ 인 직사각형으로 그려진다. 이 직사각형은 (30)으로 결정되는 원에 의해서 균사화시킬 수 있다.

$$||q_k - q_{i+1}|| \leq \dot{q}_{i,\max} \Delta t \quad (30)$$

위 식으로부터 로봇 매니퓰레이터가 q_{k-1} 로부터 $t=t_k$ 에 도달할 수 있는 영역은 다음과 같이 표현되어진다

$$R_{reach,k} = \{q_k \mid ||q_{k-1} - q_k|| \leq \dot{q}_{i,\max} \Delta t\} \quad (31)$$

그러므로, 축 속도와 가속도에 대한 유연 제한 조건을 만족하는 프리아크 $FA_{S,k}$ 는 다음과 같이 표현되어진다.

$$FA_{S,k} \equiv C_k \cap R_{reach,k} \quad (32)$$

$FA_{S,k}$ 를 그림 4에서처럼 형상 공간에서 기하학적으로 분석해 보면 다음과 같은 θ_{i-1} 과 θ_i 에 대한 선형 제한 조건으로 변형시킬 수 있음을 알 수 있다.

$$FA_{S,k} = \{q_k \mid |\theta_{i-1} - \theta_i| \leq \theta_{safe}(k)\} \quad (33)$$

여기서 $\theta_{safe}(k)$ 는 코사인 법칙(the law of cosines)을 사용하여 계산할 수 있다. 도달 가능 영역의 반지름을 r_{col} 이라 하면 다음과 같은 식이 성립하고

$$r_{col}^2 = r_{i-1}^2 + r_i^2 - 2r_{i-1}r_i \cos\theta_{safe}(k) \quad (34)$$

위 식을 정리하면 $\theta_{safe}(k)$ 를 구할 수 있다.

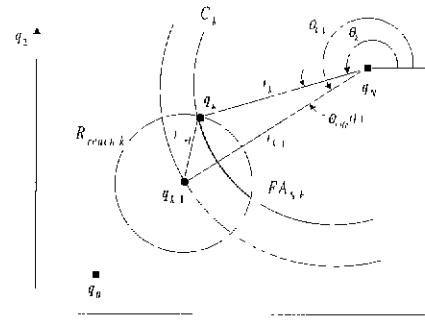


그림 4. 유연 제한 조건을 만족하는 프리아크 $FA_{S,k}$ ($q_k = (q_{1,k}, q_{2,k})$).

Fig. 4. Free arc $FA_{S,k}$ satisfying smoothness constraints.

4. 충돌 회피 문제의 구성

시간 t_k 에서 최종 프리아크 FA_k 는 지금까지의 결과로부터 다음과 같이 정리할 수 있다.

$$\begin{aligned} FA_k &= FA_{C,k} \cap FA_{S,k} \\ &= \{q_k \mid \theta_{CAC}^{\min}(k,i) < \theta_i < \theta_{CAC}^{\max}(k,i), i=1,2,\dots,P(k)+1 \\ &\quad \text{and } |\theta_{i-1} - \theta_i| \leq \theta_{safe}(k)\} \end{aligned} \quad (35)$$

프리아크는 충돌을 회피할 수 있는 형상점들의 집합이므로 충돌회피를 위한 동작 계획 문제는 프리아크 $\{FA_0, \dots, FA_1, \dots, FA_N\}$ 상에 있는 형상점들의 집합을 찾아내는 문제로 축소된다. 프리아크들의 집합만 구해지면 충돌회피 경로점들인 $\{\theta_0, \dots, \theta_1, \dots, \theta_N\}$ 를 선정할 수 있고 연속해서 형상 공간에서의 충돌회피 경로 $(q_0, \dots, q_1, \dots, q_N)$ 이 결정된다.

$\theta_0, \dots, \theta_k, \dots, \theta_N$ 들의 함수로써 성능 지수를 선정하면 프리아크들의 집합 $\{FA_{0, \dots, 0}, FA_{k, \dots, k}, FA_N\}$ 으로부터 충돌회피 경로점들인 $\{\theta_0, \dots, \theta_k, \dots, \theta_N\}$ 를 선정할 때 최적화를 고려할 수 있으므로 준최적 충돌회피 문제가 구성되어진다.

$$\begin{aligned} & \text{Find } \{\theta_0, \dots, \theta_k, \dots, \theta_N\} \quad s.t. \\ & \text{minimize } G(\theta_0, \dots, \theta_k, \dots, \theta_N) \\ & \text{subject to } \theta_{CAC}^{\min}(k, i) < \theta_k < \theta_{CAC}^{\max}(k, i) \\ & \quad |\theta_{k-1} - \theta_k| \leq \theta_{\text{gap}}(k) \quad \text{for } k = 0, 1, \dots, N. \end{aligned}$$

이 최적화 문제에서 선형부등 제한조건(linear inequality constraint)에 포함된 지수 i 는 이 최적화 문제가 여러개의 독립적인 문제들로 나누어 질 수 있음을 보여준다. 이러한 최적화 문제의 분리를 위해 다음과 같이 매 샘플링 시간마다 단일 구간을 포함하는 프리아크들로 구성되는 경로 트리(path tree : PT_i)를 정의한다.

$$PT_i = \{FA_{C,0}(i_0), \dots, FA_{C,k}(i_k), \dots, FA_{C,N-1}(i_{N-1})\} \quad (36)$$

여기서 $FA_{C,k}(i)$ 는 다음과 같이 정의되며

$$FA_{C,k}(i) = \{q_k \mid \theta_{CAC}^{\min}(k, i) < \theta_k < \theta_{CAC}^{\max}(k, i)\} \quad (37)$$

첫번째 조건 1) $q_0 \in FA_{C,0}(i_0)$ 와 두번째 조건 2) $FA_{C,N-1}(i_{N-1})$ 에 속한 θ_{i-1} 과 $FA_{C,k}(i_k)$ 에 속한 θ_k 의 공동 영역이 모든 $j=1, \dots, N-1$ 에 대해서 존재해야 한다는 조건들을 만족해야 한다. 첫번째 조건은 프리아크 $FA_{C,0}$ 의 구간 중 초기 형상을 포함하는 구간만이 유효함을 의미하며 두번째 조건은 θ_k 의 공동 구간이 존재하지 않는 인접한 프리아크의 구간들은 그 사이에서 충돌의 위험이 있기 때문에 제외시켜야 함을 의미한다. 그러면, 각 경로 트리 PT_i 에 대해서 SOCAP이 상태 변수 θ_k 에 대한 선형부등 제한조건을 갖는 최적화 문제로 구성된다. 각 경로 트리에서 구성된 SOCAP들의 해를 서로 비교하여 그들 중 최적의 성능 지수를 갖는 최종 해 $\{\theta_0, \dots, \theta_k, \dots, \theta_N\}$ 를 선택하면, 최종 충돌회피 경로 $\{q_0, \dots, q_k, \dots, q_N\}$ 를 구할 수 있다.

V. 제안한 방법의 이론적 분석 및 토의

일반적으로 로봇의 충돌회피 알고리즘의 성능은 그 알고리즘의 완전도와 계산 복잡도로써 평가되어진다. 완전도를 보장하는 알고리즘이란 주어진 상황에서 충돌회피 경로가 존재하면 반드시 찾아내고, 존재하지 않을 때에는 이를 확인할 수 있는 알고리즘을 의미한다. 완전도를 보장하는 알고리즘은 보통 계산 시간이 매우 커지기 때문에 실제 상황에 적용하기 어려우므로 완전도와 계산 복잡도 사이의 균형(compromise)을 이루는 알고리즘이 더욱 가치가 있다고 할 수 있다. 그러므로, 이번 장에서는 제안한 방법의 완전도와 계산 복잡도를 2차원 형상 공간에서 분석하고 비교함으로써 그 가치 및 한계를 밝히고자 한다.

1. 제안한 방법의 완전도에 관한 토의

IV장에서 SOCAP의 해를 구함으로써 로봇의 충돌회피 동작 계획 문제를 풀 수 있다고 주장하였다. 그러나,

SOCAP의 해가 존재하지 않을 때에는 새로운 문제가 발생한다. SOCAP의 해를 찾을 수 없다고 해서 충돌회피 경로가 존재하지 않는다고 말 할 수 있을 것인가? 제안한 방법은 형상 공간의 이산화를 사용하는 방법이고 충돌회피 경로를 찾기 위한 탐색 공간을 전체 형상 공간이 아닌 프리아크 상으로 제한하기 때문에 기본적으로 완전도를 완벽하게 보장할 수는 없다. 형상 공간의 이산화를 사용한다는 점은 다른 알고리즘도 피할 수 없는 문제로서, 충돌회피 경로의 탐색 공간이 제한되어 있지 않다면 제안한 방법도 형상 공간의 이산화를 사용한 다른 알고리즘에서처럼 이론적으로 이산화의 분해능만큼 완전하다고, 즉, 분해능적 완전(resolution-complete)한 알고리즘이라고 말 할 수 있다. 그러나, 탐색 공간의 제한은 완전도에 근본적인 제한을 가하여, 제안한 방법으로는 충돌회피 경로가 존재하는데도 불구하고 그 경로를 찾아 내지 못 할 가능성이 있다. 이 문제를 가능한 해결하기 위해서 몇 가지 상황을 설정하고 고찰하는 과정을 통해 로봇의 역방향 동작(backward motion)을 제안한다. 여기서 역방향이란 형상 공간에서 목표 형상으로부터 점점 멀어지는 방향을 의미한다.

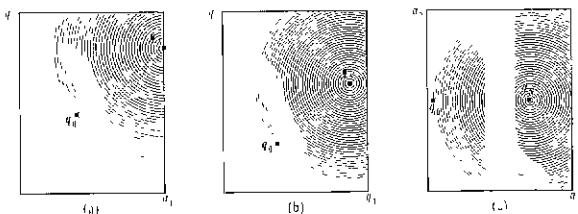


그림 5. 역방향 동작을 설명하기 위한 세 가지 상황.

Fig. 5. Example cases for introducing backward motion.

그림 5의 (a)는 IV장 4절에서 설명한 조건들을 만족시키는 경로 트리는 존재하지만 그 경로 트리로부터 유연 제한 조건을 만족시키는 SOCAP의 해를 찾아낼 수 없는 상황을 보여주고 있다. 이러한 상황은 매우 큰 형상 공간 장애물이 초기 형상이나 목표 형상 근처에 존재할 때 발생할 수 있으며, 유연 제한 조건을 고려하지 않는다면 SOCAP을 풀어 충돌회피 경로를 찾아 낼 수 있다. 그러나, 유연 제한 조건까지 만족시키는 충돌회피 경로는 제안한 방법을 단순히 적용해서는 존재하는데도 찾지 못 할 가능성이 있다. 이러한 가능성을 제거하기 위해서 로봇으로 하여금 초기 형상으로부터 목표 형상에서 멀어지는 방향으로 역방향 동작을 하도록 한다. 초기 형상에서의 프리아크보다 반지름이 큰 새로운 프리아크를 추가함으로써 이러한 역방향 동작이 가능하게 되는데 로봇이 큰 장애물을 돌아서 피해 갈 수 있을 만큼 물리 설 때까지, 즉, SOCAP의 해가 발견될 때까지 이 과정을 반복한다. 필연적으로 하나의 새로운 프리아크를 추가할 때마다 로봇의 운동 시간은 두 샘플링 기간만큼 증가하게 된다. 이러한 역방향 동작은 동일 프리아크 상에서 로봇의 동작을 허락하지 않기 때문에 더

짧은 경로를 따라 이동할 수 있는 가능성을 배제하므로 비효율적으로 보이지만, 시변 형상 공간 장애물을 효율적으로 표현하기 위해 로봇의 형상이 매 챕터링 시간마다 각각의 프리아크 위에 위치해야 한다는 기본적인 가정때문에 불가피하다.

그림 5의 (b)는 초기 형상이 시변 형상 공간 장애물에 의해 둘러싸인 상황을 보여 준다. 첫번째 경우와 다른 점은 이 상황에서는 IV장 4절에서 설명한 조건을 만족시키는 경로 트리조차 존재하지 않는다는 사실이다. 이러한 트랩(trap)은 위에서 설명한 역방향 동작으로 역시 벗어날 수 있다. 경로 트리가 성공적으로 형성될 때까지 경로 트리를 찾기 위한 과정이 반복되고 경로 트리를 찾게 되면 그 경로 트리 상에서 SOCAP을 풀어 충돌회피 경로를 찾게 된다. 고려해야 할 마지막 상황은 그림 5의 (c)와 같이 커다란 장애물을 장벽이 초기 형상과 목표 형상을 가로막고 있는 경우이다. 이 경우에는 형상 공간 내에 더 이상 프리아크를 추가할 공간이 없을 때까지 역방향 동작을 반복함으로써 충돌회피 경로가 존재하지 않음을 확인할 수 있다. 이상에서 설명한, 충돌회피 경로의 존재를 탐색하는 역방향 동작 알고리즘을 정리하면 다음과 같다.

Procedure : This procedure addresses the iterative steps in checking the success of the proposed approach

Given travel time T_f , sampling period Δt :

Determine the number of sampling $N = T_f / \Delta t$.

Discretize the C-space with N concentric circles.

repeat Construct the free arcs $\{FA_k, k = 0, 1, \dots, N-1\}$.

Build the path tree set $\{PT_i\}$;

if $\{PT_i\} = \emptyset$, goto backward

else solve the SOCAPs for each PT_i ;

if any feasible solution does not exist,

goto backward

else return success

backward if there remains no more free arc to add in the workspace,

then return failure.

else one-step backward motion by adding one free arc, which results in increasing the travel time by $2\Delta t$, i.e., $N \leftarrow N + 2$. Goto repeat

물론, 이 역방향 동작이 시변 형상 공간 장애물이 존재하는 경우에는 제안한 방법의 완전도를 완전히 보장해 주지는 못 한다. 그 이유는 첫째, 시변 형상 공간 장애물의 경우 그 모양이 역방향 동작을 반복할 때마다 달라지고, 둘째, 동일 프리아크 상에서는 로봇이 이동할 수 없기 때문이다. 그러나, 이 역방향 동작 알고리즘을 이용하여 제안한 방법으로 충돌회피 경로를 찾아낼 수 있는지 확인할 수 있고, 매우 복잡한 장애물이 존재하는 환경이 아니라면 존재하는 충돌회피 경로를 찾지 못하는 경우는 발생하기 어렵다.

2 제안한 방법의 계산 시간에 관한 분석

이번 절에서는 제안한 방법으로 시변 형상 공간 장애물을 표현하는데 걸리는 시간을 가장 널리 쓰이는 기준 방법들 중의 하나인 슬라이스 프로젝션 방법의 계산 시간과 비교한다. Lozano-Perez[15]는 충돌회피 형상 공간을 효율적으로 표현하기 위하여 각 축의 이동각(joint angle)을 이산화시키는 슬라이스 프로젝션 개념을 제시하였다. 그는 1차원 슬라이스들로부터 반복적으로 생성되는 M -1차원 슬라이스들의 집합으로 M 자유도 매니퓰레이터에 대한 형상 공간 장애물을 근사화하였다. 이 방법은 충돌회피 형상 공간을 효율적으로 근사화시킬 수 있으나 형상 공간의 이산화를 이용하는 대부분의 알고리즘들과 같이 완전도를 보장할 수 없다.

우선, 시변 환경 속에서 형상 공간 장애물을 표현하는데 걸리는 각 방법의 계산시간을 2 자유도 매니퓰레이터에 대해서 해석적으로 유도한다. 슬라이스 프로젝션 방법을 사용하기 위해서는 각 축의 이산화 분해능(resolution level)을 결정해야 한다. Δq_1 과 Δq_2 를 각각 첫번째 축과 두번째 축의 이산화 분해능이라 하자. 충 운동 시간동안의 샘플링 회수를 N 이라 하고 충돌검사 루틴(collision checking routine)의 평균 수행시간을 T_c 라 하면, 슬라이스 프로젝션 방법을 사용시 시변 형상 공간 장애물을 계산하는데 필요한 시간은 다음과 같은 범위내에서 결정된다.

$$\frac{Q_1}{\Delta q_1} \cdot (N-1) \cdot T_c \leq t \leq \frac{Q_1}{\Delta q_1} \left(1 + \frac{Q_2}{\Delta q_2}\right) \cdot (N-1) \cdot T_c \quad (38)$$

여기서, Q_1 는 작업 공간의 범위, 즉, $Q_1 = q_{1,\max} - q_{1,\min}$ 를 나타낸다. 충돌회피 형상 공간이 존재하지 않을 때 형상 공간 장애물을 구하는 데 최소의 시간이 걸리고, 첫번째 링크가 어떤 장애물과도 충돌하지 않을 때 가장 많은 시간이 걸린다.

반면에, 제안한 방법을 사용하기 위해서는 단지 프리아크의 이산화 분해능만 결정해 주면 된다. 이 이산화 분해능을 $\Delta\theta$ 라 하면, 시변 형상 공간 장애물을 표현하는데 걸리는 계산 시간은 다음과 같은 범위 내에서 결정된다.

$$\frac{90^\circ}{\Delta\theta} \cdot (N-1) \cdot T_c \leq t \leq \frac{360^\circ}{\Delta\theta} \cdot (N-1) \cdot T_c \quad (39)$$

목표 형상이 작업 공간의 모서리 부분에 위치하여 모든 동심원들의 1/4 영역에서만 충돌을 검사하는 것으로 충분할 때 최소 시간이 걸린다. 반면에, 최대 시간이 걸리는 경우는 목표 형상을 중심으로 하는 모든 동심원들이 완전히 작업 공간내에 포함될 때 발생한다. 그림 6은 두가지 경우의 예를 보여주고 있다.

(38)과 (39)를 비교해 보면, 작업 공간과 프리아크의 이산화 분해능이 지나치게 작지만 않다면 제안한 방법의 계산 시간이 슬라이스 프로젝션 방법보다 매우 작음을 확인할 수 있다. 이 분석을擴증하기 위해서 두 대의 2 자유도 평면형 로봇으로 구성된 두 로봇 시스템의 네가지 다른 경우를 조사해 본다. 그림 7은 이

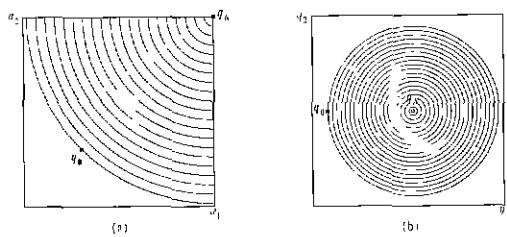


그림 6. 제안한 방법으로 시변 형상 공간 장애물을 구할 때. (a) 최소 시간이 걸리는 경우, (b) 최대 시간이 걸리는 경우($N=20$).

Fig. 6. (a) The minimum time case, (b) the maximum time case to represent time-varying C-obstacles in the proposed method($N=20$).

예들을 보여주고 있다. 이 예들에서, 로봇 1이 로봇 2와의 충돌을 회피해야 하며, 두 대의 로봇의 링크 길이는 $l_1 = l_2 = 350(\text{mm})$ 로서 똑같다. 경우 I과 경우 II에서 두 로봇 사이의 거리는 1,200(mm)로써 로봇 1의 첫번째 축은 로봇 2와 전혀 충돌이 발생하지 않는다. 경우 III과 경우 IV에서는 두 로봇 사이의 거리가 900(mm)로써 앞의 두 경우보다 가깝게 놓여져 로봇 1의 첫번째 축도 로봇 2와 충돌할 수 있으므로 두 로봇 사이의 충돌을 회피하기가 훨씬 어려워진다. 로봇 2의 초기 형상과 목표 형상은 각각 $(-90^\circ, 60^\circ)$ 및 $(60^\circ, -30^\circ)$ 로 고정시켜 놓았고 로봇 1의 초기 및 목표 형상은 그림 6의 예들과 비슷한 경우들을 만들어내기 위해서 각 경우마다 변화를 주었다. 슬라이스 프로젝션 방법에서 사용하는 각 축의 이산화 분해능과 프리아크의 이산화 분해능은 0.5° 의 같은 값으로 설정하였고, $\theta_{\text{margin}} = 50^\circ$, $N = 40$, $\Delta t = 0.1[\text{sec.}]$ 의 값을 사용하였다.

제안한 방법을 사용하여 구성한 각각의 경우에 해당하는 형상 공간 장애물을 그림 8에 도시하였고 표 1에서는 두 방법들 사이의 비교 결과를 보여주고 있다. 여기서, T_{FA} 는 제안한 프리아크 방법을 사용하여 형상 공간 장애물을 계산하는데 걸리는 시간을 나타내고, T_{SP} 는

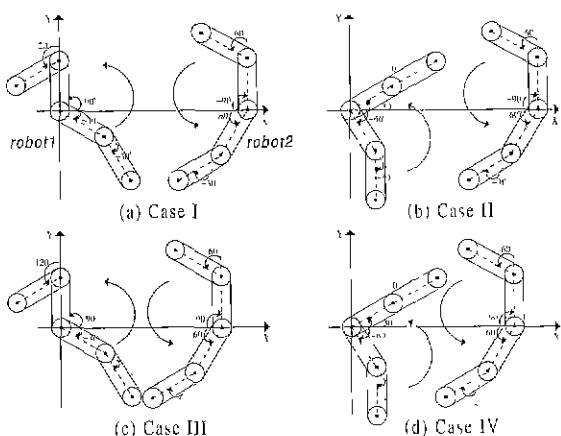


그림 7. 계산 시간 비교를 위한 네 가지 경우.

Fig. 7. Four example cases.

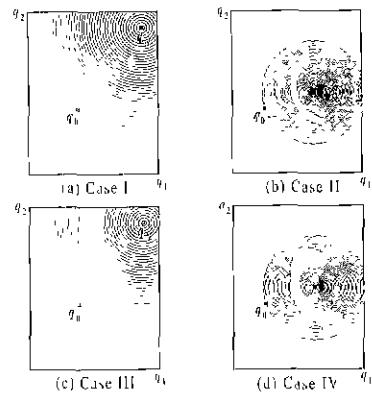


그림 8. 제안한 방법으로 구성한 형상 공간 장애물.

Fig. 8. C-obstacles constructed by the proposed method.

슬라이스 프로젝션 방법을 사용했을 때의 계산 시간을 가리킨다. 결과로부터, 제안한 방법이 대부분의 경우에 슬라이스 프로젝션 방법보다 계산 시간을 크게 단축시킬 수 있음을 알 수 있다. 예상했던 대로, 제안한 방법의 계산 시간은 두 로봇 베이스 사이의 거리에 의해 결정되는 잠재적 충돌 영역의 크기와 관계가 없는 대신에 초기와 목표 형상의 상대적인 위치에 따라 달라짐을 확인할 수 있었다. 그림 6(a)의 최소 시간 경우와 비슷한 경우들인 경우 I과 III의 형상 공간 장애물을 계산하는데 경우 II와 IV보다 더 적은 시간이 소요되었다. 반면에, 슬라이스 프로젝션 방법을 사용했을 때에는 잠재적 충돌 영역의 크기가 작을 때 더욱 많은 시간이 필요하였다.

VI. 실험 결과 및 토의

본 모의 실험에서는 제안한 방법을 각각 2 자유도 매니퓰레이터로 근사화시킬 수 있는 두 대의 SCARA 형 로봇 매니퓰레이터의 충돌회피 동작 계획 문제에 적용하였다[16]. 두 대의 로봇은 같은 규격과 기구학적 특성을 갖고 있다고 가정하였으며 두 대의 로봇 중 로봇 2는 미리 정해진 경로를 따라 등속도로 이동하고 로봇 1은 장애물인 로봇 2와의 충돌을 회피하고 각 축의 속도 제한을 만족시킬 수 있는 자신의 충돌회피 경로를 계획해야 한다. 각 로봇의 초기 및 목표 형상은 한 로봇이 초기 또는 최종 위치에 있을 때에는 충돌이 발생할

표 1. 제안한 방법과 슬라이스 프로젝션 방법의 계산 시간 비교.

Table 1. Comparison of computation time between the proposed method and the slice projection

경우	q_0 [deg.]	q_N [deg.]	D [mm]	T_{FA} [sec.]	T_{SP} [sec.]
I	(-30,-30)	(90,120)	1,200	0.97	284
II	(-60,-30)	(30,0)	1,200	2.37	285
III	(-30,-30)	(90,120)	900	0.95	249
IV	(-60,-30)	(30,0)	900	2.34	248

수 없도록 선정하였으며, 링크들의 길이를 고려하여 두 로봇 매이스 사이의 거리는 1,000[mm]로 설정하였으므로 공동 작업 영역에 들어 가면 각 로봇은 서로 다른 로봇과의 잠재적인 충돌 가능성을 갖게 된다. 로봇의 총 운동 시간은 10[sec]로서, 이 모의 실험에 사용된 두 로봇 시스템의 관련 데이터는 표 2에 정리하였고, 각 로봇의 초기 위치와 최종 위치를 그림 10에서 확인할 수 있다. 우선 작업 공간에 포함되는 초기 프리아크 $FA_{CS,i}$ 를 구하고 프리아크의 분해능을 1°로 설정하여 시변 형상 공간 장애물을 구성하면 그림 9에서와 같이 충돌 제한 조건을 만족하는 프리 아크 $FA_{CS,i}$ 가 구해진다.

표 2. 모의 실험을 위한 입력 데이터.

Table 2. Data for computer simulation..

기 호	값[단위]	
	로봇1($r = 1$)	로봇2($r = 2$)
q'_{\min}	-180[deg.]	-180[deg.]
q'_{\max}	180[deg.]	180[deg.]
q'_{max}	40[deg./sec.]	40[deg./sec.]
q'_i	(-30,-60)[deg.]	(90,60)[deg.]
q'_N	(60,90)[deg.]	(-120,-30)[deg.]
l_1	400[mm]	
l_2		200[mm]
θ_{margin}		5.0[deg.]
N		20
Δt		0.5[sec.]

그림 9에서 볼 수 있듯이 초기 형상을 포함하고 인접한 프리아크들 사이의 공통 구간이 항상 존재하는 두 개의 경로 트리를 구성할 수 있고 각 경로 트리 PT_1 과 PT_2 에 대하여 SCAP을 만들 수 있다. 본 모의 실험에서는 Yoshikawa[17]에 의하여 제안된 조작성 지수(measure of manipulability · MOM)를 성능 지수로 사용한다. MOM은 2 자유도 평면형 로봇 매니퓰레이터의 경우 다음과 같이 계산되어진다.

$$MOM = l_1 l_2 |\sin q_2| \quad (40)$$

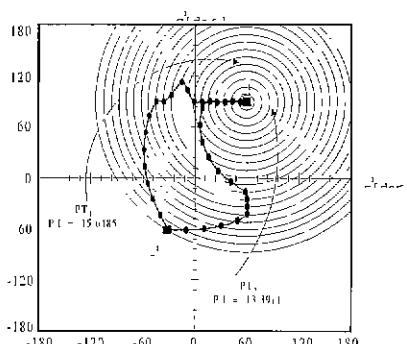


그림 9. SCARA 로봇 시스템 중 로봇 1의 준최적 충돌회피 경로

Fig. 9. Sub-optimal collision-free path of robot 1.

여기서 l_1, l_2 는 링크의 길이이며 q_2 는 두 번째 축의 위치를 나타낸다. l_1, l_2 는 정해진 상수이고 각 샘플링 시간에서의 MOM의 합을 최대화시키는 것이 목적이므로 SCAP은 다음과 같은 형태를 갖는다.

$$\text{Find } \{\theta_0, \dots, \theta_i, \dots, \theta_N\} \text{ s.t.}$$

$$\text{minimize } - \sum_{i=1}^N |\sin(n \sin \theta_k + q_{2,N}^i)|$$

$$\text{subject to } \theta_{\text{CAC}}^{\text{min}}(k) < \theta_k < \theta_{\text{CAC}}^{\text{max}}(k)$$

$$|\theta_{k-1} - \theta_k| \leq \theta_{\text{age}}(k) \quad \text{for } k = 0, 1, \dots, N.$$

각 SCAP들을 풀은 결과로부터 각 경로 트리마다 성능 지수 -15.6485와 -13.3911의 값을 갖는 준최적 충돌회피 경로를 얻을 수 있었고, 그림 9에서 이 결과를 도시하였다. 각 SCAP으로부터 얻은 성능 지수들을 비교하면 PT_1 으로부터 구한 경로가 로봇 2와의 충돌을 회피하고 동작 제한 조건을 동시에 만족시키면서 최대의 MOM값을 갖는 준최적 경로로 선택되어진다. 그림 10에서 두 대의 로봇 매니퓰레이터가 계산한 준최적 충돌회피 경로를 따라 충돌 없이 각각의 원하는 최종 위치로 이동하는 것을 확인할 수 있다.

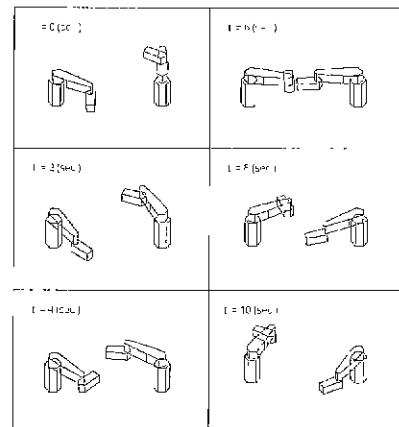


그림 10. SCARA 로봇 시스템의 준최적 충돌회피 동작.

Fig. 10. Sub-optimal collision-free motion of SCARA robot system..

V.1절에서 설명한 역방향 동작의 기능을 확인하기 위하여 V.2절에서 설정한 4가지 경우에 대해서도 모의 실험을 수행하였다. 경우 I과 II에서는 그림 8에서 구성한 환경에 따라 각각 두 개의 경로 트리가 형성되고 각 경로 트리마다 준최적 충돌회피 문제를 구성할 수 있다. 성능 지수로서 MOM의 합과 최소 이동 거리의 두 지수를 사용하였다. 로봇 1의 이동 거리를 최소화하기 위해서는 다음과 같이 성능 지수를 선정한다.

$$G(\theta_0, \dots, \theta_i, \dots, \theta_N) = \sum_{i=1}^N (\theta_{k-1} - \theta_k)^2 \quad (41)$$

앞에서 설명한 방법으로 준최적 충돌회피 문제들을 풀은 결과를 그림 11과 12에 도시하였다. MOM의 합을

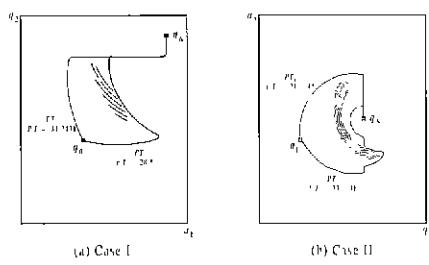


그림 11. 경우 I과 II에서 MOM 을 최대화시키는 준최적 충돌회피 경로.

Fig. 11. Sub-optimal collision-free path maximizing MOM for Case I and II.

성능 지수로 사용했을 때 경우 I에서는 경로 트리 PT_1 으로부터, 경우 II에서는 경로 트리 PT_2 로부터의 해가 최종 경로로 선정되었다. 반면에, 최소 이동 거리가 성능 지수로 사용되었을 때에는 경우 I에서는 PT_2 로부터, 경우 II에서는 PT_1 으로부터의 해가 최종 경로로 선정되었다. 그림 11에서는 로봇 1이 MOM 의 합을 최대화하기 위하여 그림 12에서의 이동 거리를 최소화하기 위한 경로보다 더 긴 경로를 따라 움직인다. 이것은 (40)에서 알 수 있듯이 각 셀플링 시간에서의 MOM 의 값을 크게 하기 위하여 q_2 를 가능한 $\pm 90^\circ$ 에 가까운 값으로 유지시키기 때문이다. 물론, 이 최종 경로들은 제안한 방법의 특성상 로봇의 형상이 매 셀플링 시간마다 프리아크상으로 그 위치가 제한되기 때문에 최적 경로가 아닌 준최적 경로이다. 준최적이라 함은 제안한 방법으로 충돌을 회피하는 것이 가장 중요한 목표이고 이 목표를 만족시키는 범위 내에서 최적화 과정을 거친을 의미한다. 그러므로, 프리아크 방법은 가장 중요한 목적이 시변 장애물과의 충돌을 효율적으로 회피하는 것이고 특정 성능 지수를 동시에 최적화하고자 할 때 유용하다.

그림 8에서 볼 수 있듯이 초기에는 경우 III에서는 1개의, 경우 IV에서는 두 개의 경로 트리가 형성되고, 각 경로 트리마다 준최적 충돌회피 문제들이 성공적으로 구성된다. 그러나, 모든 준최적 충돌회피 문제에서 유연 제한 조건을 만족시킬 수 있는 해가 존재하지 않는다.

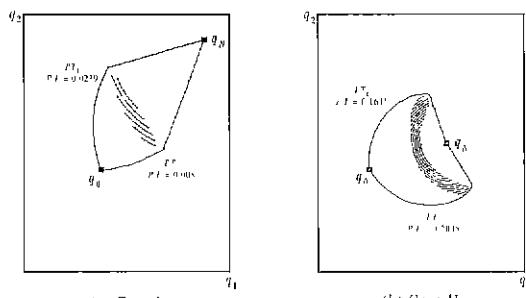


그림 12. 경우 I과 II에서 이동거리를 최소화시키는 준최적 충돌회피 경로.

Fig. 12. Sub-optimal collision-free path minimizing moving distance for Case I and II.

리 두 대의 메니퓰레이터 사이의 거리가 링크들의 길이에 비해서 상대적으로 짧아졌기 때문에 충돌을 회피하기 매우 어려워졌다라는 것을 의미한다. 이 문제를 해결하기 위해서 V.1절에서 제안한 역방향 동작을 포함하는 알고리즘을 적용한다. 즉, 유연 제한을 만족하는 충돌회피 경로를 찾아내기 위해서 매 스텝마다 초기 형상으로부터 바깥쪽으로 프리아크를 하나씩 추가시키고 로봇의 역방향 동작을 수행시킨다. 결과적으로 로봇은 형상 공간에서 유연 제한을 만족시키면서 장애물을 피해 돌아갈 수 있는 경로를 찾아내게 된다. 경우 III에서는 13 스텝의, 경우 IV에서는 19 스텝의 역방향 동작 알고리즘을 반복한 후에 각 준최적 충돌회피 문제의 가능해(feasible solution)를 찾을 수 있었다. 로봇 1의 MOM 의 합을 최대화시키는 준최적 충돌회피 경로를 그림 13에, 로봇 1의 이동 거리를 최소화시키는 준최적 충돌회피 경로를 그림 14에 도시하였다. 이 그림들에서 전방향 동작과 역방향 동작시의 형상 공간 장애물들을 동시에 표현하였으므로, 로봇 1의 역방향 동작시 만나는 부분은 실제 장애물이 아닌 전방향 동작시의 장애물이 표현된 것이다. 이상의 결과들은 제안한 방법이 로봇의 시변 장애물 회피 문제들 중 상대적으로 여유가 있는 문제들은 쉽게 풀 수 있는 동시에 매우 어려운 문제도 전체 동작 시간이 증가하기는 하지만 역방향 동작을 이용하여 풀 수 있음을 보여주고 있다.

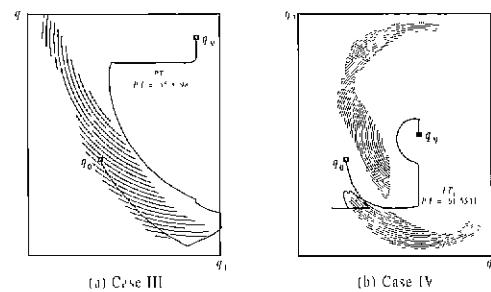


그림 13. 경우 III과 IV에서 MOM 을 최대화시키는 준최적 충돌회피 경로.

Fig. 13. Sub-optimal collision-free path maximizing MOM for Case III and IV.

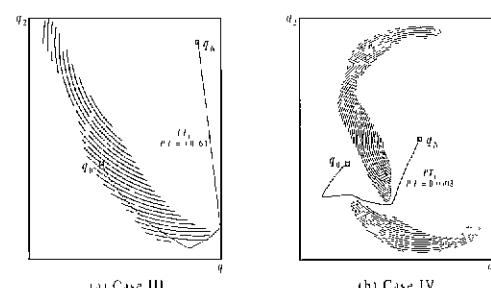


그림 14. 경우 III과 IV에서 이동거리를 최소화시키는 준최적 충돌회피 경로.

Fig. 14. Sub-optimal collision-free path minimizing moving distance for Case III and IV.

VII. 결론

본 논문에서는 로봇의 초기 위치와 최종 위치가 주어져 있고 시변 장애물을 포함하는 작업 공간에 대한 정보가 미리 알려진 경우에 로봇의 충돌회피 동작을 계획하기 위한 새로운 방법을 제안하였다. 기존의 로봇 충돌회피 동작 계획 알고리즘들과 비교했을 때, 제안한 방법은 시변 장애물을 포함하는 환경을 보다 효율적으로 구성함으로써 동작 계획에 소요되는 계산 시간을 감소시킬 수 있었으며, 동시에 로봇의 동작 제한 조건을 고려할 수 있었다.

제안한 방법을 단순히 적용했을 때 충돌회피 경로를 찾아 낼 수 없는 상황들을 구성하고 그 상황들에 대응할 수 있는 알고리즘을 찾아냄으로써 제안한 방법의 완전도를 부분적으로 보장할 수 있음을 보이고 모의 실험을 통해 확인하였다. 제안한 방법으로 시변 형상 공간 장애물을 구성하는데 걸리는 계산 시간을 이론적으로 분석하고 가장 널리 사용되어지는 슬라이스 프로젝션 방법과 비교하여 효율적임을 보였고, 두 대의 로봇 매니퓰레이터로 이루어진 시스템에 두 방법을 적용하여 그 계산 시간을 비교, 확인하였다.

결론적으로, 본 연구의 결과를 통하여 제안한 방법이 다음과 같은 장점을 갖고 있음을 주장할 수 있다.

1) 시변 장애물이 존재하는 작업 환경 내에서 로봇 매니퓰레이터의 충돌회피 동작 계획 문제를 간단한 형태의 최적화 문제(SOCAP)로 변형시킬 수 있다.

2) 시변 형상 공간 장애물을 효율적으로 구성할 수 있고, 다각형(polygon) 또는 다각체(polyhedra)로의 근사화가 필요하지 않기 때문에 장애물을 지나치게 돌아가는 충돌회피 동작을 피할 수 있다.

3) 형상 공간의 차원보다 한 차원 증가되는 형상-시간 공간을 구성하지 않고서도 시간 차원의 고려가 가능하다.

4) 로봇의 속도 제한 조건을 고려할 수 있고, 최소 이동 거리나 조작성 지수와 같은 다양한 척도를 필요에 따라 SOCAP의 성능 지수로 사용할 수 있다.

5) 로봇의 역방향 동작을 이용하여 어려운 충돌회피 동작 계획 문제를 풀 수 있고 알고리즘의 완전도를 부분적으로 보장할 수 있다.

참고문헌

- [1] J. -C. Latombe, *Robot Motion Planning*. Kluwer Academic Publishers, Boston, 1991.
- [2] Y. K. Hwang and N. Ahuja, "Gross motion planning-A survey," *ACM Computing Surveys*, vol. 24, no. 3, pp. 219-291, Sep., 1992.
- [3] J. F. Canny, *The complexity of robot motion planning*. MIT Press, 1987.
- [4] N. S. V. Rao, "Robot navigation in unknown generalized polygonal terrains using vision sensors," *IEEE Trans. Syst., Man, Cybem.*, vol. 25, no. 6, pp. 947-962, June, 1995.
- [5] H. R. Beom and H. S. Cho, "A sensor-based navigation for a mobile robot using fuzzy logic and reinforcement learning," *IEEE Trans. Syst., Man, Cybem.*, vol. 25, no. 3, pp. 464-477, Mar., 1995.
- [6] L. E. Kavraki, P. Svestka, J. -C. Latombe and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Trans. Robotics and Automation*, vol. 12, no. 4, pp. 566-580, Aug., 1996.
- [7] K. Gupta and X. Zhu, "Practical motion planning for many degrees of freedom: A novel approach within sequential framework," *IEEE Int. Conf. on Robotics and Automation*, pp. 2038-2043, 1994.
- [8] B. H. Lee and C. S. G. Lee, "Collision-free motion planning of two robots," *IEEE Trans. Syst., Man, Cybem.*, vol. SMC-17, no. 1, pp. 21-32, Jan./Feb., 1987.
- [9] Y. Shin and Z. Bien, "Collision-free trajectory planning for two robot arms," *Robotica*, vol. 7, pp. 205-212, 1989.
- [10] Z. Bien and J. Lee, "A minimum-time trajectory planning method for two robots," *IEEE Trans. Robotics and Automation*, vol. 8, no. 3, pp. 414-418, June, 1992.
- [11] J. Lee, "A dynamic programming approach to near minimum-time trajectory planning for two robots," *IEEE Trans. Robotics and Automation*, vol. 11, no. 1, pp. 160-164, Feb., 1995.
- [12] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *Int. J. Robotics Research*, vol. 5, no. 3, pp. 72-89, Fall, 1986.
- [13] T. Lozano-Perez and M. A. Wesley, "An algorithm for planning collision-free paths among polyhedral obstacles," *Commun. ACM*, vol. 22, no. 10, pp. 560-570, Oct., 1979.
- [14] B. H. Lee, "Constraints identification in time-varying obstacle avoidance for mechanical manipulators," *IEEE Trans. Syst., Man, Cybem.*, vol. 19, no. 1, pp. 140-143, Jan./Feb., 1989.
- [15] T. Lozano-Perez, "A simple motion-planning algorithm for general robot manipulators," *IEEE Trans. Robotics and Automation*, vol. RA-3, no. 3, pp. 224-238, June, 1987.
- [16] S. W. Lee, B. H. Lee and K. D. Lee, "Sub-optimal collision-free motion planning of a two-robot system using free arc," *Int'l J. of Control Engineering Practice*, vol. 6, no. 8, pp. 935-946, Nov., 1998.
- [17] T. Yoshikawa, "Manipulability of robotic mechanisms," *Int. J. Robotics Research*, vol. 4, no. 2, pp. 3-9, Summer, 1985



이석원

1968년 7월 30일생. 1991년 서울대학교 제어계측공학과 (공학사). 1993년 서울대학교 제어계측공학과(공학석사). 1998년 서울대학교 전기공학부(공학박사). 1998년~현재 삼성황공 정밀기기연구소 선임연구원. 관심분야는 로보틱스, 지능제어, 공장자동화, 네트워크 이론.



이범희

1955년 5월 29일생. 1978년 서울대학교 전자공학과(공학사). 1980년 서울대학교 전자공학과(공학석사). 1985년 Univ. of Michigan. Computer, Information & Control Eng.(공학박사). 1987년 3월~현재 서울대학교 전기공학부 교수. 1990년 3월~현재 IEEE 국제 로보틱스 및 자동화 학술회의 프로그램 위원 및 총장. 1992년 8월~현재 전국 마이크로 로봇 경연대회 조직위원장. 1995년 8월~현재 서울시 중소기업 창업보육센터 소장. 관심분야는 로보틱스 및 공장자동화, 퍼지제어 및 응용, 지능로봇 시스템, 센서퓨전 응용.



남윤식

1963년 9월 11일생. 1987년 서울대학교 제어계측공학과 (공학사). 1989년 서울대학교 제어계측공학과(공학석사). 1997년 서울대학교 제어계측공학과(공학박사). 1997년 9월~1998년 1월 삼성전자 선임연구원. 1998년~현재 한국산업기술대학교 자동화공학과 전임강사. 관심분야는 로보틱스, 공장자동화, 마이크로프로세서 응용, 제어이론.