

형상공간 접근 방식에 기반한 모듈식 고정쇠의 적재가능성 분석

Loadability Analysis of Modular Fixtures based on a Configuration Space Approach

유 건 아
(Kyeonah Yu)

Abstract : In modular fixturing systems, a given part or a subassembly is fixed by placing fixture elements such as locators and clamps adequately on a plate with a lattice of holes. It is known that the minimal number of point contacts to restrict translational and rotational motions on a plane is four and the type of three locators and a clamp(3L/1C) is the minimal fixture. Brost and Goldberg developed the complete algorithm to automatically synthesize 3L/1C types of fixtures which satisfy the condition of form closure. Due to the nature of the fixture, the clearance between the fixture and the part is extremely small. It is hard to load the part repeatedly and accurately for human as well as for robot. However the condition of loadability has not been taken into account in the B&G algorithm. In this paper, a new method to decide a given fixture to be loadable by using configuration space is proposed. A method to plan for a part to be loaded by using compliance safely even in the presence of control and sensing uncertainty is proposed is well.

Keywords : computational geometry, fixture loading, compliant motion, robot algorithm

I. 서론

고정쇠는 조립이나 선반가공 등 거의 모든 제조공정 단계에서 부품을 고정시키기 위해 사용되는 기구이다. 모듈식 고정쇠는 여러 가지 고정쇠 구성요소(fixture elements)들을 조합하여 정확한 위치로 부품을 고정시키는 시스템이며 재사용이 가능하다는 장점때문에 많은 관심을 받고 있다. Brost와 Goldberg는 주어진 부품에 대하여 가능한 모든 3L/1C 형태의 모듈식 고정쇠를 찾아내는 최초의 완전 알고리즘(이하 B&G 알고리즘)을 개발하였다[1]. 평면에서 직선 운동과 회전운동을 억제하는 form closure를 위한 접촉점의 최소 수는 4개로 알려져 있으며 3L/1C 고정쇠는 3개의 위치기와 1개의 클램프로 이루어진 최소의 고정쇠(minimal fixture)이다. B&G 알고리즘에서 사용한 위치기는 실린더형으로 2차원 평면에서 보면 지름이 일정한 원형이며 클램프는 클램프 버팀대(strut)와 플린저(plunger)로 이루어져 있다. 그림 1은 풀총(gluegun)을 조립하기 위해 풀총 케이스를 고정하는 예를 논문 [1]에서 인용한 것으로 B&G 알고리즘에 의해 찾아내어진 97개의 고정쇠 형상중 2가지이다. 3L/1C 형태의 고정쇠는 2개의 클램프를 사용하던 기존의 모듈식 고정쇠에 비해 클램핑하는 시간이 절약되지만 부품을 적재하는 일은 어려워진다. 위치기들이 부품과 타이트하게 밀착되어 있어 미세한 오차에도 잘못된 결과를 초래할 수 있기 때문이다. 특히 부품이 로봇에 의해 반복적으로 적재되고 제거되어야 하므로 로봇의 제어와 센싱 과정에서 생길 수 있는 불확실성에 대해 강인함을 보여야 한다. 그러므로 적

재의 용이성은 고정쇠가 만족해야 할 필수조건으로 언급되지만[2,3] B&G 알고리즘은 가능한 모든 고정쇠 형상을 생성하는 데에만 초점을 맞추었을 뿐, 고정쇠의 적재 가능 여부에 대한 언급은 하지 않았다. 본 논문에서는 3L/1C 형태의 고정쇠 적재 문제에 대하여 다룬다.

앞에서도 지적한 바와 같이 부품과 위치기들 사이에는 여유공간이 없으므로 위로부터 최종형상으로 바로 삽입하는 등의 단순 운동에 의한 적재보다는 주변에 이용할 수 있는 위치기나 클램프 중의 일부를 가이드로 사용하여 미끄러지듯 최종 형상으로 회전하여 들어가도록 하는 컴플라이언트 모션에 의한 적재 방식이 적절하다. 센싱이나 운동 오차에도 불구하고 컴플라이언트 운동에 의한 적재가 가능한가 판별하기 위하여 고정쇠 적재 문제를 형상공간(Configuration Space, 이하 C-공간)에서 다룰 것을 제안한다.

C-공간에서는 움직이는 물체를 점으로 표현하고 주변 환경을 구성하는 장애물들을 그에 따라 수정해 줌으로써 물체와 장애물과의 기하학적인 관계를 명확하게 표현해 주며, 그 위에 해결하고자 하는 운동계획문제를 적절히 매핑하여 답을 구한다[4,5]. 움직이는 물체를 점으로 환산하여 표현하는 C-공간에서 운동을 계획하는 일은, 부피가 있는 물체가 충돌없이 움직이도록 계획하는 일보다 훨씬 간단할 뿐만 아니라 컴퓨터 기하학을 이용하여 보다 체계적이고 알고리즘적인 접근을 보장할 수 있다. 이제까지의 연구결과는 주로 로봇이나 환경을 구성하는 장애물들을 다각형으로 제한하고 로봇이 직선이동만을 한다는 전제로 얻어진 것이 대부분이기 때문에, 물체의 회전운동이 필수적이며 비다각형 물체를 다루어야 하는 고정쇠 적재 문제에 활용하기 위해서는 기존 알고리즘의 보완 및 확장이 필요하다[6,7]. 고정쇠 적재문제를 C-공간에서 표현하기 위해서 적재되어야 할 부품을 움직이는 물체로, 고정되어 있는 고정쇠 구성요소들

논문접수 : 2003. 1. 23., 채택확정 : 2003. 4. 23.

유건아 : 덕성여자대학교 컴퓨터과학부(kyeonah@duksung.ac.kr)

※ 본 논문은 한국과학재단 연구비 지원 (R04-2001-000-00048-0)에 의하여 연구되었습니다.

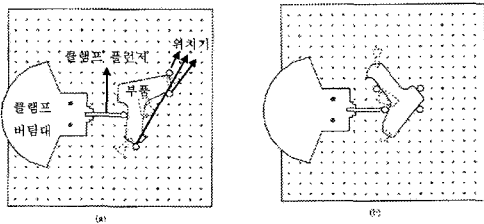


그림 1. 풀총(gluegun) 케이스를 고정하는 3L/1C 고정쇠의 예(참고문헌 [1]에서 인용함).

Fig. 1. Example of 3L/1C fixture immobilizing a gluegun case(Cited from reference [1]).

을 장애물로 간주한다. 본 논문에서는 B&G 알고리즘에서 사용한 고정쇠 구성요소인 실린더형의 위치기와 다각형이나 호를 포함한 다각형 모양의 클램프를 가정한다. 즉, 다루는 도형군이 다각형에 제한되지 않고 호와 선분으로 이루어진 일반화 다각형(*generalized polygons*)으로 확대되는 것이다. 또한 적재해야 할 부품의 회전운동이 필수적이므로 결과적으로 3차원의 일반화다각형 C-공간 장애물을 다루어야 한다. 본 논문에서는 주어진 3L/1C 고정쇠가 적재 가능 여부를 C-공간 표현을 이용하여 판별하는 방법을 제시한다. 이를 위해 기존의 다각형 환경에서 개발된 C-공간 장애물을 구성 알고리즘과 경로계획 알고리즘들을 시간 복잡도가 나빠지지 않으면서 일반화 다각형 환경으로까지 확장하는 방법을 보여준다.

본 논문의 구성은 다음과 같다. II장에서는 고정쇠 문제를 C-공간에서 표현하는 방법에 대하여 설명하고 III장에서는 C-공간 표현을 이용하여 적재가능성을 판단하는 방법과 적재 가능한 고정쇠에 대해 적재하는 방법을 계획하는 과정을 설명한다. IV장에서는 B&G 알고리즘과 본 논문에서 제안한 알고리즘의 구현 및 실행 결과를 보여주며 V장에서는 결과에 대한 의미를 살펴보고 향후에 계속되어야 할 과제에 대한 제안으로서 맺음 한다.

II. C-공간에서의 고정쇠 적재 문제

고정쇠 적재문제를 C-공간에서 표현하기 위해서 적재되어야 할 부품을 움직이는 물체로, 고정되어 있는 고정쇠 구성요소들을 장애물로 간주한다. 서론에서도 언급한 바와 같이 부품의 방향을 고려하면 C-공간은 3차원이 되는데 이 장에서는 고정쇠 적재 문제를 위하여 일반화 다각형의 3차원 C-공간 장애물을 구성하는 방법에 대해 논한다. 3차원 C-공간 장애물은 2차원 C-공간 장애물을 부품의 방향 축으로 쌓음으로서 형성이 되며 이 때, 연속적인 부품의 방향 공간을 어떻게 이산화(*discretization*)하는가가 중요한 문제이다.

1. 2차원 C-공간 장애물의 구성

움직이는 부품의 방향이 고정되어 있다면 C-공간 장애물은 2차원 유클리디안 공간에서 나타내어진다. 효과적으로 2차원 C-공간 장애물을 구하기 위해 다각형과 원형 장애물의 특징을 고려하여 각기 다른 알고리즘에 의해 2차원 C-공간 장애물을 구성하고 나중에 합집합한다. 알고리즘을 구

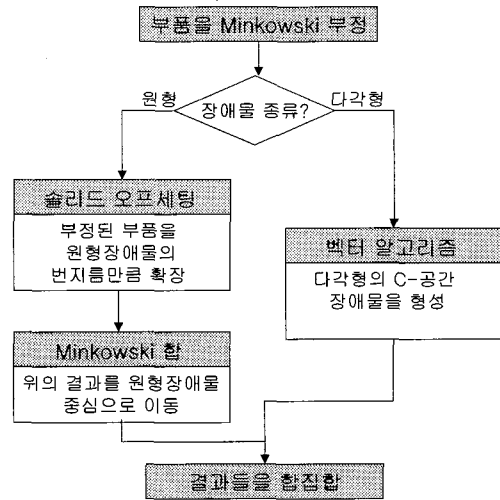


그림 2. 2차원 C-공간 장애물 구성 알고리즘.

Fig. 2. Algorithm for constructing 2-dimensional C-space obstacles.

성하는 각 단계에 대한 설명은 다음과 같다.

■ 솔리드-오프세팅

원형 장애물을 C-공간 장애물로 변환하기 위해서는 부품을 원형 장애물의 반지름만큼 확장하는 양(*positive*)의 솔리드 오프세팅연산, \uparrow^* 을 적용한다[8]. 원형 장애물의 반지름을 r 이라고 하고 부품을 집합 S 로 표시하면 S 의 오프셋은 $S \uparrow^* r = \{p \in \mathbb{R}^2 \mid \exists q \in S, \|p - q\| \leq r\}$ 로 정의된다. 이 식을 구현하면 꼭지점이 볼록(*convex*)인 경우 그 꼭지점을 중심으로 하고 양변에서 외향 수직벡터(*outward normal*) 방향으로 r 만큼 확장된 점들을 양 끝점으로 하는 호로 변환되고 오목(*concave*)인 경우에는 다시 오목 꼭지점으로 변환되어 “선분과 원호로 둘러싸인 닫힌 R^2 영역”, 즉 일반화 다각형이 생성된다.

■ Minkowski 부정과 합

Minkowski 부정은 집합 S 에 대해 $\ominus S = \{-e \mid e \in S\}$ 와 같이 정의되며 2차원 평면상에서 원점에 대한 대칭으로 구현된다. Minkowski 합은 집합 P 와 L 에 대해 $P \oplus L = \{p+l \mid p \in P, l \in L\}$ 과 같이 정의된다. 솔리드 오프세팅 연산에 의해 원형 장애물은 점으로 환산되어 집합 P 와 점 L 과의 Minkowski 합은 P 를 L 의 좌표만큼 직선 이동시켜 얻어진다.

■ 벡터 알고리즘

다각형 장애물에 대한 C-공간 장애물은 [4]에서 제안된 벡터 알고리즘에 의해 구현한다. Minkowski 부정으로 얻어진 도형 P 와 다각형 장애물 O 의 둘레를 각각 시계반대방향(CCW)으로 돌며 선분에 대한 벡터 리스트를 생성한다. 생성된 두 개의 리스트를 각도가 증가하는 순서로 정렬하여 합친다.

■ 합집합

일반화 다각형의 합집합은 기존의 *sweep-line* 알고리즘을 선분과 호의 교점을 구하도록 확장한 알고리즘[9]을 사용한다. *Sweep-line* 알고리즘의 핵심은 선분의 시작점과 끝점을

을 *sweep-line*의 방향과 수직 방향으로 정렬하는 것인데 원의 호는 *y*축 혹은 *x*축을 따라 단순 증가 혹은 단순 감소하는 성질을 만족하지 못하므로 호의 끝점들을 정렬한다는 것이 의미가 없다. 그러므로 호를 단순 증가 및 감소하는 성분으로 나누어 처리하면 기존의 알고리즘의 시간 복잡도를 저하시키지 않고도 일반화다각형으로 확장할 수 있다. 그림 3은 원과 다각형 장애물에 대해 위의 과정을 적용하여 2차원 C-공간 장애물을 생성하는 예를 보여준다.

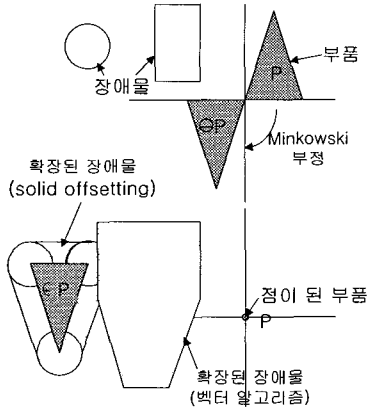


그림 3. 원과 사각형 장애물에 대한 2차원 C-공간 장애물의 구성과정.
 Fig. 3. Constructing process of 2-dimensional C-space obstacles for a circle and a square obstacles.

■ 시간 복잡도

임의의 다각형 *P*를 솔리드 오프세팅을 하기 위한 시간 복잡도는 *P*의 꼭지점의 개수가 *m*이라고 할 때, $O(m)$ 이다. 솔리드 오프세팅에 의해 생성된 일반화 다각형의 꼭지점의 수는 최대 2배이며 Minkowski 부정이나 Minkowski 합은 다각형의 꼭지점의 수만큼 실행하면 되므로 시간 복잡도는 $O(m)$ 이 된다. 벡터 알고리즘에 의한 C-공간 장애물을 구성하는 데에는 장애물의 꼭지점의 개수가 *n*이라고 할 때, 모두 볼록 다각형인 경우, $O(n+m)$ 이고 오목다각형을 포함하는 경우, $O(nm)$ 이다. 두 일반화 다각형의 합집합을 구할 때, 선분의 수가 *n*인 경우, 기존의 *Sweep-line* 알고리즘의 시간 복잡도가 $O(n \log n)$ 인데 호는 최대 4개의 호로 분할되어 같은 방식으로 실행되므로 전체 시간 복잡도에는 변함이 없다.

2. 3차원 C-공간 장애물의 구성

움직이는 부품의 방향이 고정되어 있지 않다면 C-공간은 부품의 위치 (*x*, *y*)와 방향 θ 의 3차원 유클리디안 공간이다. C-공간 장애물도 3차원이 되며 θ 축에 따른 모양은 비정형이어서 정형화된 식으로 표현하기가 어렵기 때문에 $[-\pi, \pi]$ 구간 사이에 여러 슬라이스를 쌓아 놓은 형태로 모델링한다. 그러나 슬라이스 표현법에서 *n*분의 1 식으로 연속적인 θ 영역을 일정 간격으로 이산화하는 경우, 원하는 각도의 답을 놓칠 수 있는 단점이 있다. 특히 고정쇠나 조임새와 같이 물체의 미세한 회전이나 이동이 중요한 문제에서는 더욱 치명적인 단점이 된다. 그러므로 본 장에서는 완전한 3차원 C-공간 장애물을 구성하지 않고도 존재하는 답을 놓

치지 않도록 이산화하는 임계치-기반 알고리즘(criticality-based algorithm)을 제안한다.

1) 임계각의 충분조건

그림 1과 같이 원형 위치기에 부품을 삽입할 때, 움직이는 부품의 방향에 따라 C-공간 장애물이 어떻게 변하는가를 관찰하여 임계각의 충분조건을 알아낸다. 2차원 C-공간 장애물 경계선을 부품의 방향 θ 의 식으로 표현했을 때, 그 식의 표현이 달라지는 각도를 임계각으로 정의한다. 즉, 임계각 슬라이스 사이의 장애물 표현식이 동일하도록 임계각을 정의한다. 부품이 적재되어야 할 대상인 세쌍의 위치기들만을 고려했을 때, 임계각이 될 충분 조건은 다음과 같이 구분된다. 위치기 *i*에 대한 C-공간 장애물을 $CO(i)$ 라고 할 때 다음 경우에 C-공간 장애물의 기하학적 모양이 변한다.

- 조건 1: $CO(i_j)$ 의 끝점이 $CO(i_k)(i \neq k)$ 의 선분이나 호와 교차
- 조건 2: $CO(i_j)$ 의 호가 $CO(i_k)(i \neq k)$ 의 선분이나 호와 외접

그림 4는 이 두 가지 임계각이 발생하는 예를 보여준다. 그림 4 (a)와 (b)의 점선 부분이 각각 끝점이 선분과 교차하는 예와 호가 선분과 외접하는 예를 보여준다. 부품의 방향이 최종 목표를 0° 로 하여 CCW 방향으로 돌 때, 임계각을 기준으로 그 전과 후의 C-공간 장애물의 기하학적 특성이 변한다. 그림 4에서 임계각이 발생하는 예의 하단에 화살표로 나타낸 것이 바로 그 전후 관계를 나타낸다. 임계각 사이의 C-공간 장애물은 부품의 각도가 증가함에 따라 모양이 연속적으로 변화하지만 기하학적 특징은 같기 때문에 도형을 θ 의 식으로 일반화할 수 있다. 임계각을 만나고 그 이상으로 돌면 C-공간 장애물의 기하학적 모양은 바뀌게 되어, 새로운 도형의 식을 적용하여야 한다.

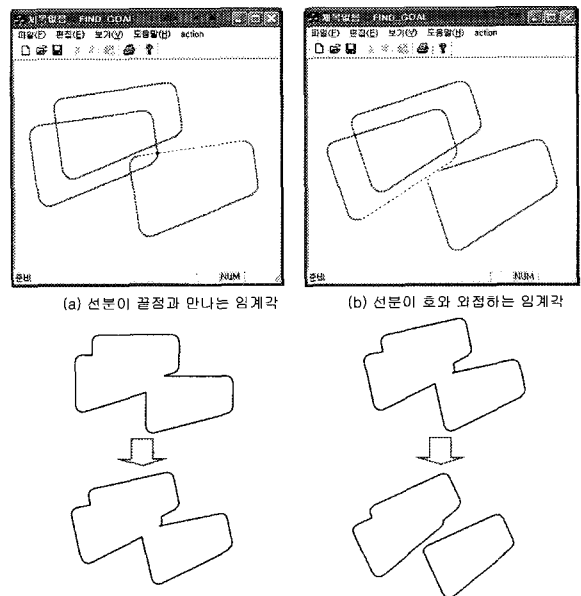


그림 4. 임계각의 발생과 그 전후 C-공간 장애물의 기하학적 모양의 변화.
 Fig. 4. Occurrence of critical orientations and changes of geometrical shapes of C-space obstacles before and after its occurrence.

2) 임계각의 계산

위치기들은 모두 반지름이 일정한 원이기 때문에 모양은 변하지 않고 부품에 각도에 따라 C-공간 장애물의 각도만 변하는 특징을 갖고 있다. 그러므로 앞에서 소개한 두 가지 조건을 만족하는 닫힌 형태(closed-form)의 답을 구할 수 있는데 이를 위해 사용할 여러 가지 표기법은 그림 5에 도시하였다. 그림 5에서는 부품의 기준점을 원점으로 하는 좌표계에서 일반화 다각형으로 확장된 부품과 장애물이 되는 위치기가 있을 때, 부품의 점들과 위치기의 중심을 어떻게 표기하였나 보여주고 있다.

■ 조건 1: 끝점과 세그먼트의 교점

여기서 세그먼트는 선분과 호를 총칭한다. 우선 $CO(l_1)$ 의 선분이 $CO(l_2)$ 의 선분의 끝점과 교차하기 위한 식은 다음과 같다.

$$R_\theta \cdot X_i + X_{L1} = a(R_\theta \cdot X_j + X_{L2}) + (1-a) \cdot (R_\theta \cdot X_k + X_{L2}) \quad (1)$$

$$R_\theta(X_i - a \cdot X_j - (1-a) \cdot X_k) = X_{L2} - X_{L1} \quad (2)$$

(1)에서 양변의 norm을 취하면 $|R_\theta^2| = 1$ 이므로 다음과 같은 a의 2차식을 얻을 수 있다.

$$a^2(X_j - X_k)^2 + 2 \cdot a \cdot (X_j - X_k) \cdot (X_k - X_i) + (X_k - X_i)^2 - (X_{L2} - X_{L1})^2 = 0 \quad (3)$$

(3)으로부터 구한 a가 $0 \leq a \leq 1$ 을 만족하면 R_θ 식에 대입하여 θ 를 구한다. 끝점과 호의 교점에 대해서도 유사하게 θ 를 구할 수 있다.

■ 조건 2: 호와 세그먼트의 외접

$CO(l_1)$ 의 호가 $CO(l_2)$ 의 호와 외접하기 위한 식은 다음과 같다.

$$\begin{aligned} |X_{c1} - X_{c2}| &= 2r \\ |(R_\theta X_{c1} + X_{L1}) - (R_\theta X_{c2} + X_{L2})|^2 &= 4r^2 \\ 2 \cdot R_\theta \cdot (X_{c1} - X_{c2}) \cdot (X_{L1} - X_{L2}) &= \\ 4r^2 - |X_{c1} - X_{c2}|^2 - |X_{L1} - X_{L2}|^2 & \end{aligned} \quad (4)$$

(4)에 R_θ 식을 대입하여 정리하면 $A \cos \theta + B \sin \theta = C$ 형태의 식이 유도되어 삼각함수 합성에 의해 두 개의 임계각도

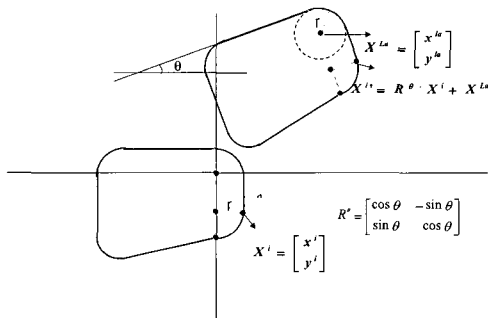


그림 5. 확장된 부품과 위치기의 중심으로 이동한 부품의 좌표 표기.

Fig. 5. Notation for coordinates of the part expanded and translated to the center of a locator.

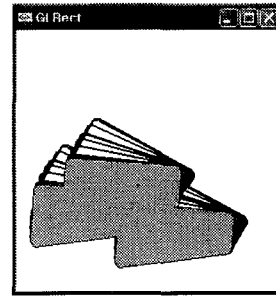


그림 6. $\theta=0^\circ$ 인 면으로부터 올려본 2차원 C-공간 장애물의 더미.

Fig. 6. Stacks of 2-dimensional C-space obstacles viewed from underneath, $\theta=0^\circ$.

가 구해진다. 호와 선분의 외접에 대해서도 유사하게 구할 수 있다.

이와 같이 구해진 임계각들에 대해 2차원 C-공간 장애물을 각각 구하고 임계각 사이의 C-공간 장애물을 θ, x, y 의 식으로 표현하여 3차원 C-공간 장애물을 구성하면 그림 6과 같이 2차원 장애물의 더미(pile)로 표현할 수 있다. 임계각을 구하기 위한 시간 복잡도는 C-공간 장애물의 꼭지점수를 n 이라고 할 때 세그먼트와 세그먼트 및 세그먼트와 꼭지점의 모든 조합에 대하여 위의 식을 적용해야 하므로 $O(n^2)$ 이다.

III. 적재 가능성과 적재 플래닝

적재는 클램프가 제거된 상태에서 일어나게 되므로 이 장에서 목표형상이라 함은 클램프를 제거한 상태의 위치기 3개로 이루어진 형상을 말한다. C-공간에서 목표 형상이 다음 3가지 조건을 만족할 때 그 고정쇠 형상은 '적재 가능하다'고 한다.

- 접근성(Accessibility): 목표 형상이 자유공간에 있는가?
- 유일성(Uniqueness): 목표 형상이 유일한 위치 설정인가?
- 오차허용성(Tolerance): 불확실성의 존재에도 목표에 도달 가능한가?

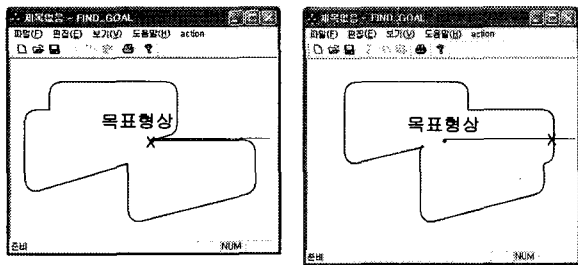
B&G 알고리즘에서 생성된 고정쇠 형상들은 II장에서 소개된 C-공간 장애물로 변형되고 변형된 C-공간 장애물에 대해 위의 세 가지 조건을 차례로 테스트하여 한가지라도 만족하지 않으면 그 고정쇠 형상은 적재 가능하지 않다고 판단한다. 적재 가능하다고 판단된 고정쇠 형상들에 대하여는 불확실성이 존재하더라도 적재가 가능한 컴플라이언트 운동을 계획한다.

1. 접근성

컴플라이언트 운동은 장애물을 가이드로 이용하는 운동을 허용하므로 자유공간이라 함은 C-공간 장애물의 경계선까지를 포함한 것을 의미한다. C-공간에서 목표형상이 자유공간에 포함되어 있으면 '이 목표 형상은 접근 가능하다'고 한다. 고정쇠 적재 문제에서 최종 목표는 점으로 표현되므로 점이 C-공간 장애물의 내부에 있지 않으면 자유공간에

있음을 의미한다. 컴퓨터 기하학 분야에서 주어진 점과 도형에 대해 점의 위치가 도형의 내부인지 외부인지를 판단하는 방법으로는 여러 가지가 알려져 있는데, 그 중 구현이 용이한 Ray crossing 알고리즘을 이용한다[10].

Ray crossing 방법은 주어진 점과 도형이 있을 때, 이 점과 무한히 멀리 있는 점을 연결한 선분을 가정하고 이용하는 방법이다. 이 선분의 끝점에서 시작점으로 따라가다 도형과의 첫 번째 교점이 생기면, 이는 점이 도형의 내부에 위치함을 의미한다. 그 후 두 번째 교점을 만나면 도형의 내부에서 외부로 빠져나간 것임을 알 수 있다. 이를 일반화하면, 선분과 도형의 교점의 개수가 짝수이면 점이 도형의 외부에 위치하고, 홀수이면 점이 도형의 내부에 위치한다고 볼 수 있다. 단, 점으로부터 한쪽으로 뺄어나가는 선분과의 교점의 개수와 반대 방향으로 뺄어나가는 선분과의 교점의 개수에 대한 모듈 값이 서로 다른 경우는 점이 도형을 이루는 경계선 위에 위치하는 특수한 경우로서 앞서 말한 바와 같이 경계선도 자유공간의 일부이므로 접근 가능한 형상으로 간주한다. 그림 7은 이 방법에 의해 접근 가능한 목표와 그렇지 않은 목표를 판별하는 예를 보여준다. 이 알고리즘은 일정한 선분과 C-공간 장애물의 경계의 교차 여부를 확인하면 되므로 C-공간 장애물의 꼭지점수를 n 이라고 할 때, 시간 복잡도는 $O(n)$ 이 된다.



(a) Ray와 장애물과 교점이 없을 때에는 목표형상이 자유공간에 속한 경우이다. (b) Ray와 장애물과 교점이 홀수번 있을 때에는 목표형상이 장애물 내부에 있는 경우이다.

그림 7. Ray-crossing 알고리즘의 적용 예.
Fig. 7. Example of applying the Ray-crossing algorithm.

2. 유일성

고정쇠에서 부품이 부정확하게 자리를 잡으면, 그 후 일어나는 모든 공정과정에서 오차를 초래하게 된다. 그러므로 고정쇠에 대해 부품이 정확하게 위치해 있는가를 인식하는 일은 매우 중요하다. 고정쇠 적재 문제에서 목표의 인식은 3개의 위치기와 부품이 모두 접촉하는 순간 이루어지므로 적재 과정에서 부품이 3개의 위치기 모두와 맞는 방식이 유일해야만 한다. 즉, 적재과정에서 목표형상 이외의 다른 형상이 3개의 위치기와 동시에 접촉할 가능성이 있는 고정쇠 형상은, 목표 형상을 잘못 인식할 가능성이 있는 것으로 허용하면 안 된다. 위치기와의 접촉을 인식하기 위해서는 [11]과 [12]에 소개된 바와 같이 간단한 on-off 센서를 이용할 수 있다.

C-공간에서 목표형상은 3개의 선분의 교차점으로 표현된다. 이 3개의 선분이 한 점에서 유일하게 만나게 되는 충분

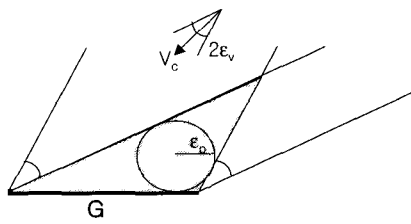
조건은 세 선분이 $[x \ y]$ 벡터의 1차 결합으로 표시하였을 때, 첨가 행렬의 계수가 2인 것이다. 계수는 널리 사용되고 있는 Gauss 소거법 알고리즘에 의하여 구할 수 있으며 이 알고리즘의 시간 복잡도는 행렬의 차수를 n 이라고 할 때, $O(n^3)$ 으로 알려져 있다.

3. 오차허용성

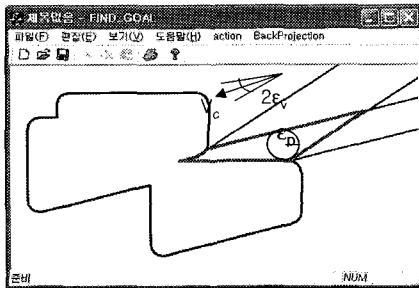
고정쇠와 같이 부품과 위치기 사이의 여유 공간이 없는 응용 분야에서 부품을 위치기 사이로 적재하는 경우, 미세한 오차에도 잘못된 결과를 초래할 수 있다. 이런 경우에는 장애물의 일부를 가이드로 이용하여 슬라이딩에 의해 목표에 도달하는 컴플라이언트 운동이 적합하다고 할 수 있다. 제어와 센싱 오차가 존재함에도 불구하고 항상 목표형상에 도달하는 것을 보장해 주는 컴플라이언트 운동계획이 가능해야 주어진 고정쇠 형상이 '적재 가능하다'고 말할 수 있다. 오차허용성이란 이와 같이 센싱이나 제어 오차가 존재해도 항상 목표에 도달할 수 있는가에 관한 조건이다. 그러므로 오차허용성의 확인은 컴플라이언트 운동에 위한 적재를 계획하는 일과 병행하여 이루어진다.

제어오차를 팔호 안의 원뿔로 모델링한다면(즉은 명령속도(commanded velocity), 꼭지점은 초기위치, 각은 $2\epsilon_v$ 인 원뿔) 명령속도 V_c 로 목표형상 G 에 안전하게 도달하기 위해서는 물체의 가능한 초기 위치는 그림 8(a)와 같이 빗금 친 부분이 되며 이를 preimage라고 부른다[13]. 센싱 오차를 고려하면 부품을 위해 가장 안전한 초기 위치는 그림 8 (a)와 같이 preimage에 포함된 가장 큰 원의 중점이 되며 이때 허용오차는 ϵ_p 가 된다. ϵ_p 가 매니플레이터의 센싱 오차의 범위를 보장하지 못하면 오차허용성 조건은 만족하지 못하게 되어 적재 가능하지 않다고 판단하게 된다. 그림 8 (b)와 (c)는 고정쇠 적재 문제에서 계산된 preimage와 ϵ_p 로서 오차허용성 조건이 만족된 경우와 그렇지 않은 경우의 예를 보여준다. preimage를 구하는 알고리즘은 역투사 알고리즘(backprojection algorithm)[14,15]이 있으며 이 알고리즘의 시간 복잡도는 C-공간 장애물의 꼭지점수를 n 이라고 할 때, $O(n \log n)$ 이다.

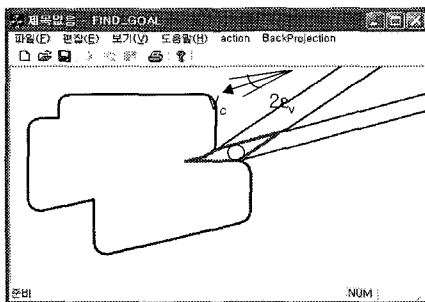
ϵ_p 가 센싱 오차의 범위를 만족했다 하더라도 초기에 부품의 방향(orientation) 오차가 있다면 부품의 운동은 3차원 C-공간에서 표현되며 부품 방향 오차에 대한 분석은 3차원 C-공간에서 이루어져야 한다. 역투사 알고리즘에 의해 초기 위치가 주어지면 V_c 가 가리키는 방향(direction)으로 이동하는데 운동 방향 오차 및 위치 오차 등으로 인해 목표 형상이 아닌 다른 형상으로 진행할 수 있다. 이때 만나게 되는 가능한 모든 형상들을 3차원 C-공간 장애물로부터 예측할 수 있다. 위치기 센서의 on, off 상태를 1과 0으로 표현하여 고정쇠와 부품과의 관계를 3-비트 이진 벡터로 표현한 고정쇠 상태를 정의한다. 이와 같은 표현 방식을 따르면 목표형상 상태는 세 개의 위치기가 모두 부품의 면에 맞는 상태인 $[1 \ 1 \ 1]$ 로 나타내어지며 $[1 \ 0 \ 0]$ 은 위치기 1만 부품에 접촉한 상태이고 나머지 위치기들은 부품에 닿아있지 않음을 의미한다. 일반적인 고정쇠 상태 벡터는 유일한 고정쇠 형상을 나타내주지는 못하지만 제시된 컴플라이언트 운동 중에 발생한 상태 벡터의 변화는 오차를 보장하는데



(a) 제어 및 센싱 오차와 프리이미지



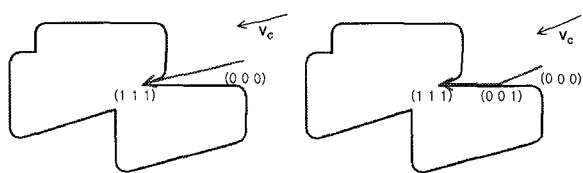
(b) 초기위치 허용오차가 ϵ_p 인 예



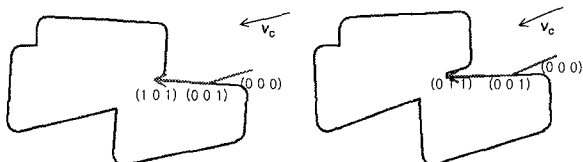
(c) 초기위치 허용오차가 ϵ_p 보다 작은 예

그림 8. 고정식 적재 문제에서의 preimage.

Fig. 8. Preimage in the fixture loading problem.



(a) 부품의 방향오차가 없을 때, (b) 부품의 방향오차가 없을 때, 위치가 3을 자유공간에서 목표로 바로 접근하는 경우 가이드로 하여 목표로 접근하는 경우



(c) 부품이 CW 방향으로 오차가 있을 때, (d) 부품이 CCW 방향으로 오차가 있을 때, (b)와 같이 목표로 접근하는 경우 (b)와 같이 목표로 접근하는 경우

그림 9. 명령 속도 V_c 에 대해 실제 발생 가능한 운동 방향과 형상상태 벡터값.

Fig. 9. Possible velocity direction for the commanded velocity V_c and the corresponding configuration state vector values.

중요한 정보를 제공한다. 그림 9는 그림 8의 운동 속도 V_c 로 진행할 때, 발생 가능한 C-공간에서의 운동 경로를 화살

표로 보여주며 그에 따라 변하는 형상 상태 벡터 값도 같이 보여주고 있다. 그림 9에서 볼 수 있듯이 3D C-공간 장애물로부터 V_c 로 움직이는 과정에서 발생 가능한 모든 형상 상태의 시퀀스들을 찾아 낼 수 있으며 이 시퀀스로부터 현재 부품의 가능한 실제 위치와 방향 범위를 알아낼 수 있다.

이 범위를 알아내면 목표 형상과의 차이를 보정해 주면 되는데 이는 논문 [12]의 SCM(Selective compliance mechanism)과 같이 직선 운동 및 회전 컴플라이언스를 동시에 제공할 수 있는 컴플라이언트 기구를 이용하여 가능하다. 즉, 상태 벡터로부터 정확한 현재 위치는 알 수 없고 가능한 위치의 범위를 나타내는 θ 와 (x,y) 좌표의 궤적을 구할 수 있으므로 정확한 운동 계획보다는, 상태 벡터가 [1 1 1]에 이르기까지 컴플라이언스를 가함으로써 오차를 보정하는 방법을 사용한다[12]. 이를 위해서는 궤적이 선형적으로 변하는 성질을 만족하여야 한다.

예를 들면 그림 9 (d)와 같이 [0 0 0]→[0 0 1]→[0 1 1]과 같은 시퀀스는 현재 부품이 위치가 2와 3에 접촉하고 있음을 나타낸다. 3차원 C-공간 장애물로부터 [0 1 1]이라는 상태 벡터는 부품의 방향 θ 가 CCW 방향으로 $0 < \theta < 19.3$ 임을 알 수 있으며 θ 에 따른 (x,y) 좌표의 궤적을 알아낼 수 있다. 그림 10은 [0 1 1] 상태의 궤적을 나타내는데 SCM에 의해 방향 오차가 보정되기 위해서는 궤적이 선형이어야 하므로 부분 선형을 만족하는 θ 의 구간이 $[0, \epsilon_p]$ 일 때, ϵ_p 가 매니퓰레이터의 센싱 오차의 범위를 보장하지 못하면 오차허용성 조건은 만족하지 못하게 되어 적재 가능하지 않다고 판단하게 된다.

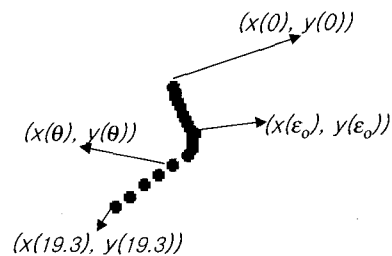


그림 10. [0 1 1] 상태의 궤적으로부터 부품 방향의 가능한 범위 결정하는 과정.

Fig. 10. Process for deciding the range of part orientations possible from the trajectory of the state, [0 1 1].

IV. 구현 및 시뮬레이션 결과

본 논문에서는 적재 가능성의 판별을 위한 모든 알고리즘을 윈도우즈 98환경의 Visual C++로 구현하였다. 또한 Common-LISP로 작성되어 Symbolics XL-1201에서 실행되었던 B&G의 알고리즘을 본 논문의 구현 부분과 연동하기 위하여 Visual C++로 다시 구현하였으며 전체 프로그램은 그림 11과 같은 블록으로 구성되어 있다.

B&G 알고리즘에서 사용한 것과 같은 Qu-Co 모듈식 고정식 셋을 모델로 하였다. 이 셋의 고정식 플레이트는 구멍이 18x18 격자 모양으로 0.75인치 간격으로 배열되어 있어 각종 위치기와 클램프 요소들을 꽂을 수 있고 실린더형의

위치거나 클램프의 반지름은 0.25인치이다. 테스트할 부품으로는 같은 모양으로 크기에서 차이가 나는 사각형 2개와 5각형 1개, 오목다각형으로는 B&G 알고리즘에서 사용된 부품인 풀총(gluegun) 케이스와 힌지 플레이트(hinge plate)를 포함하였다. 각각의 테스트 부품에 대하여 B&G 알고리즘에서 생성된 고정쇠 형상 중, 적재 가능 조건을 한가지씩 거치며 조건에 만족하는 고정쇠 형상이 몇 퍼센트 남아 있는가를 비교하였다. 이 결과는 표 1에 나타나 있다. 이 표에서 볼 수 있듯이 B&G 알고리즘에서 생성된 고정쇠 형상 중, 평균 4.0%정도만이 최종까지 남아있다. 부품에 모양에 따라 제거되는 후보 고정쇠의 비율이 많이 차이가 나는데, 예를 들어 평행한 면이 있고 그 면들이 길면 유일성을 체크할 때, 많이 제거된다.

논문 [1]에서도 지적했듯이 B&G 알고리즘은 굉장히 많은 수의 고정쇠를 생성하여 이중 하나를 고정쇠 설계자에게 선택하도록 한다. 특히 설계자는 여러 가지 판단조건을 이용하여 어느 것이 더 나은 고정쇠인가를 판별하는 동시에 고정쇠들이 적재하기에 적당한가를 판별해야 하는데 이는

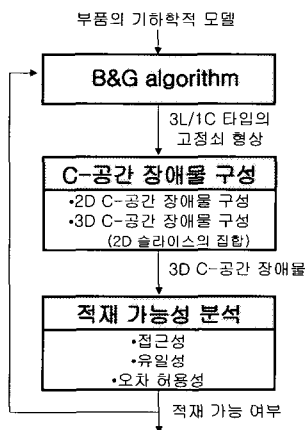


그림 11. 전체 알고리즘의 구성도.

Fig. 11. Flowchart of the entire algorithm.

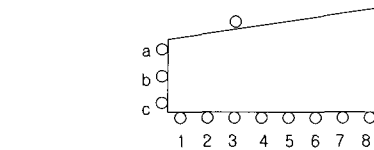
표 1. B&G 알고리즘 결과와 적재 조건들을 통과한 고정쇠 수의 비율.

Table 1. The ratio of the number of fixtures satisfying loadability conditions over the number of fixtures generated by the B&G algorithm.

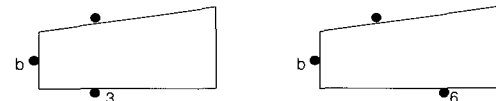
부품의 종류	B&G 알고리즘	접근성	유일성	오차허용성
블록 4각형(소)	42(100.0%)	36(85.7%)	20(47.6%)	4(9.5%)
블록 4각형(대)	1814(100.0%)	1684(92.8%)	1126(62.1%)	134(7.4%)
오각형	2480(100.0%)	2076(83.7%)	210(8.5%)	16(0.6%)
힌지 플레이트	7270(100.0%)	7200(99.0%)	528(7.3%)	94(1.3%)
풀총	364(100.0%)	234(64.2%)	224(61.5%)	4(1.1%)

설계자에게 많은 부담을 주는 일이다. 그러므로 논문 [1]에서는 과도한 수의 고정쇠 후보의 수를 줄이기 위한 한가지 방법을 제안했는데 그것은 베이스 플레이트에 구멍이 한 칸 건너씩(2배수), 혹은 세칸 건너씩(4배수) 있는 것처럼 격자간격을 성글게 하여 시도하는 것이다. 이는 부품의 크기가 격자 간격에 비해 월등히 큰 경우, 그림 12와 같이 부품의 한 면에 대해 좁은 간격의 모든 구멍이 후보를 생성하여 유사한 고정쇠가 많이 생성되게 된다는 데에서 착안한 방법이다. (그림 12의 (a)에서는 같은 패턴의 고정쇠가 1x3x8=24개만큼 생긴다.) 이 방법은 가장 성근 간격(coarse grid)으로 시작하여 유사한 고정쇠를 최소로 생성하고 설계자에게 보인 후 만족하는 고정쇠가 없으면 한 단계 미세한 간격으로 다시 고정쇠를 생성하여 조건에 모두 만족한 고정쇠가 나올 때까지 이 과정을 반복한다. 이 방식의 문제점은 유사하다고 생략한 고정쇠들이 적재 측면에서는 전혀 유사하지 않은 것일 수 있다는 데에 있다. 그림 12 (a)의 예에서 가능한 2가지 고정쇠에 대해 역투사 알고리즘을 적용하여 적재계획을 해 보면 (b) 형상은 적재가 가능하지 않은 반면 (c) 형상은 적재가 가능하다.

본 논문에서 제안한 바와 같이 설계된 고정쇠들에 대해 적재 가능성 여부를 조사하면 최종 고정쇠 후보의 개수를 크게 줄이는 동시에 적재 방법이 함께 계획되어 고정쇠 설계자의 부담을 줄일 수 있는 효과가 있다. 그림 12(c)의



(a) 유사한 패턴의 고정쇠를 생성하는 격자 위치



(b) 적재가 가능하지 않은 고정쇠 (c) 적재가 가능한 고정쇠

그림 12. 격자간격에 비해 큰 부품에 대해 생기는 고정쇠 패턴.

Fig. 12. Fixture pattern produced in case that the size of the part is larger than the lattice grid.

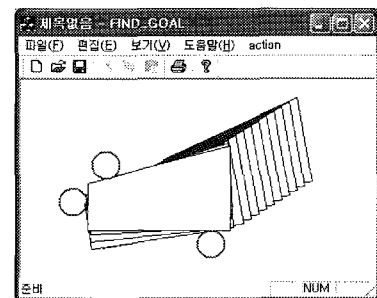


그림 13. 적재 과정(그림 12(c) 고정쇠의 적재 과정).

Fig. 13. Loading process (Loading process of the fixture in Fig. 12(c)).

표 2. 오차허용성을 줄임에 따라 남은 고정쇠 수의 비율.
Table 2. The percentage of remaining fixtures by reducing tolerance.

부품의 종류	$\epsilon_p=0.1\text{cm}$	$\epsilon_p=0.3\text{cm}$	$\epsilon_p=0.5\text{cm}$
블록 4각형(소)	4(9.5%)	2(4.8%)	0(0%)
블록 4각형(대)	134(7.4%)	40(2.2%)	30(1.7%)
오각형	16(0.6%)	16(0.6%)	8(0.3%)
헌지 플레이트	94(1.3%)	50(0.7%)	46(0.6%)
풀축	4(1.1%)	4(1.1%)	0(0%)

고정쇠에 대한 적재계획을 실행하는 과정을 그림 13에서 보여준다. 또한 허용하는 오차의 범위를 조정함으로써, 최종까지 남은 고정쇠의 수를 조정할 수 있어 설계자는 고정쇠 선택에 유연성을 갖게 된다. 특히 적재를 실행하게 될 매니플레이터의 오차허용성에 대해 충분한 수의 후보 고정쇠가 생성된다면, 실제 오차허용 범위보다 작게 설정하여 신뢰도가 좋은 소수의 고정쇠 후보만을 생성되게 할 수도 있다. 표 2는 오차허용성의 변화에 따라 생성되는 후보 고정쇠 수의 비율을 나타낸다. 오차허용성을 엄격하게 할수록 남은 후보 고정쇠의 수가 적어짐을 볼 수 있다.

V. 결론

본 논문에서는 모듈식 고정쇠를 생성하는 완전 알고리즘인 B&G 알고리즘에서 생성되는 고정쇠 형상들에 대해 로봇 매니플레이터가 부품을 적재하는 일이 가능한가를 형상공간(C-공간) 표현을 이용하여 판단하는 방법을 제안하였다. 적재가능성의 조건으로는 부품이 평면 운동에 의해 목표 형상에 도달할 수 있는가에 대한 접근성, 목표형상이 유일한가에 대한 유일성, 로봇 매니플레이터의 센싱 및 제어 오차 범위 내에서 적재가 가능한가를 따져주는 오차허용성 등의 3가지이며 한가지 조건씩 테스트하여 위배되는 고정쇠들을 탈락시켜 나가고 최종적으로 남은 고정쇠에 대해서는 컴플라이언스를 이용한 적재 방법을 함께 제시하게 하였다. 결과에서는 B&G 알고리즘이 생성한 많은 수의 고정쇠 형상을 적재가능성을 확인하면서 효과적으로 줄여주어 고정쇠 설계자의 부담을 덜어 주는 동시에, 센싱이나 제어의 불확실성이 존재해도 로봇 매니플레이터가 안전하게 실행할 수 있는 컴플라이언트 운동을 계획해 줄 수 있음을 보여주었다.

본 논문에서는 적재조건을 검사하는 전 과정을 C-공간에서 실행하는데 이는 고정쇠 요소들이나 움직이는 부품이 있는 원래의 작업환경에서 실행하는 것에 비하여 여러 가지 장점이 있다. 우선 C-공간에서는 움직이는 부품을 점으로 표현하기 때문에 부품과 장애물과의 기하학적인 관계를 명확하게 해줌으로서 부품과 장애물간의 관계를 해석하는 일이 훨씬 간단해진다. 또한 컴퓨터 기하학에서 개발된 체

계적인 알고리즘들을 문제를 해결하는 데에 응용할 수 있는데, 이 알고리즘들은 대개 시간복잡도면에서 최적화되어 있고 예외상황(degenerate case)들에 대한 처리부분도 검증되어 있기 때문에 구현 후에 성능이나 안정성 측면에서 유리하다. 본 논문에서 C-공간 장애물을 구성하는 과정과 역투사를 구하기 위해 사용된 sweep-line 알고리즘이나, 접근성을 테스트하기 위해 이용한 Ray-crossing 알고리즘 등 여러 부분에 이와 같은 컴퓨터 기하학 알고리즘이 사용되었다. 이들 알고리즘을 고정쇠 문제에 적용하기 위하여 원의 호를 포함하도록 확장시켰으며, 응용 범위를 확대하기 위해 앞으로 더 연구가 되어야 할 부분이기도 하다. 예로써 본 논문에서는 위치기는 실린더형(평면으로는 원형)만 가정했고 클램프는 다각형이라고 가정했기 때문에, 실제 모듈식 고정쇠 시스템에서 사용되는 다양한 구성 요소들에 대하여 본 논문에서 제안한 방식을 바로 적용하기 힘들다. 이를 위해서는 C-공간 장애물을 구성하는 알고리즘부터 다양한 모양의 도형군을 포함하도록 확장해야 한다. 또한 적재경로도 부품과 원형 위치기 사이에 점접촉(point contact)만을 고려하여 계획되었으므로 다른 형태의 접촉에도 적용할 수 있는 계획 방식으로 확장하여야 한다. 이와 같이 확장하는 것은 어셈블리 문제나 일반적인 로봇 모션 계획 문제로의 확장을 가능하게 하는 기초가 될 것이다.

참고문헌

- [1] R. Brost and K. Goldberg, "A complete algorithm for synthesizing modular fixtures for polygonal parts", *IEEE Trans. Robotics Automation*, vol. 12(1), pp. 31-46, Feb, 1996.
- [2] H. Asada and A. By, "Kinematic analysis of workpart fixturing for flexible assembly with automatically reconfigurable fixtures", *IEEE Trans. Robotics and Automation*, vol. 1(2), pp. 85-93, 1985.
- [3] D. Ding, G. Xiang, Y. Liu, and M.Y. Wang, "Fixture layout design for curved workpieces", *Proceedings of IEEE Int. Conf. on Robotics and Automation*, pp. 2906-2911, 2002.
- [4] T. Lozano-Perez, "Spatial planning: A configuration space approach", *IEEE Trans. Computers*, vol. 32(2), pp. 108-120, 1983.
- [5] J.-C. Latombe, *Robot Motion Planning*, Kluwer Academic Publishers, 1991.
- [6] E. Sacks and C. Bajaj, "Sliced configuration spaces for curved planar bodies", *International Journal of Robotics Research*, vol. 17, no. 6, pp. 639-651, June 1998.
- [7] E. Sacks, "Deterministic path planning for planar assemblies", *Proceedings of IEEE Int. Conf. on Robotics and Automation*, pp. 1524-1529, 2001.
- [8] J. Rossignac and A.G. Requicha, "Offsetting operations in solid modelling", *Computer Aided Geometric Design*, vol. 3, pp. 129-148, 1986.
- [9] 안진영, 유건아, "일반화 다각형을 위한 sweep-line 알

고리즘의 구현”, 한국정보과학회 춘계 학술지, 29(A)호 pp. 691-693, 2002.

- [10] J. O'Rourke, *Computational Geometry in C*, Cambridge University Press, 1998.
- [11] B. Benhabib, K. Chan, and M. Dai, "A modular programmable fixturing system. ASME Journal Eng. Industry, vol. 113(1), pp. 93-100, 1991.
- [12] K. Yu and K. Goldberg, "A complete algorithm for fixture loading", *International Journal of Robotics Research*, vol. 17(11), pp. 1214, 1998.
- [13] T. Lozano-Perez, M. Mason, and R. Taylor, "Automatic synthesis of fine motion strategies for robots," *International Journal of Robotics Research*, vol. 3(1), pp. 3-24, 1984.
- [14] M. Erdmann, "Using backprojections for fine motion planning with uncertainty", *International Journal of Robotics Research*, vol. 5(1), pp. 29-45, 1986.
- [15] A. J. Briggs, "An efficient algorithm for one-step planar compliant motion planning with uncertainty", *Algorithmica*, 8: 195-208, 1992.



유 건 아

1964년 1월27일생. 1986년 서울대 제어계측공학과 졸업. 동대학원 석사(1988), 미국 University of Southern California 전산학과 박사(1995). 1996년~현재 덕성여자대학교 컴퓨터과학부 교수. 관심분야는 알고리즘 접근

로보틱스, 컴퓨터 기하학, 인공지능.