

# 컨베이어 상의 물체 획득을 위한 로봇의 최소시간 알고리즘

## Minimum-Time Algorithm for Intercepting an Object by the Robot on Conveyor System

신익상, 문승빈\*  
(Ik Sang Shin and Seungbin Moon)

**Abstract :** This paper focuses on planning strategies for object interception by the robotic manipulator on a conveyor system in minimum time. The goal is that the robot is able to intercept object with minimum time on a conveyor line that moves at a given speed. The search algorithm for minimum time solution is given in detail for all possible cases for initial locations of robot. Simulations results show the validity of the given algorithm.

**Keywords :** minimum time, interception, robot, conveyor, trajectory planning

### I. 서론

공장 자동화 시설의 한 요소인 로봇시스템의 중요한 특징 중 하나는 생산라인 상에서 연속적으로 진행되는 물체를 손실 없이 빠르게 처리하는 것이다. 효율적으로 물체를 처리하기 위해서는 물체를 최소시간으로 획득하는 능력이 필수적이다. 일반적으로 작업 대상물을 획득하는 공통적인 방법은 예측, 계획, 실행 기법이다[1,2]. 로봇의 작업 영역상에서 물체의 이동을 예측하기 위해서는 물체 감지가 필수적이며 감지 센서 중에서 카메라와 같은 비전센서가 많이 이용되고 있다. 특히 제품의 특성상 생산 주기가 짧은 생산 시스템에서 비전센서는 빠르고 효율적으로 물체의 정보를 획득할 수 있게 해준다[3,4,11]. 물체 정보를 알기 위한 방법으로 Papanikolopoulos는 컨베이어상의 물체 추적을 위한 비전 정보를 찾는데 SSD optical flow 기법으로 물체의 모션을 계산하였다[5,6]. Tsakiris는 3차원 물체추적을 위한 수학적 이론을 제시[7]하였고, Rembold는 물체의 위치와 방위를 템플릿 매치방식으로 계산하고 획득 영역은 칼만 필터를 이용하여 연속적으로 추정하는 방법을 제시하였다[13].

최소 이동 경로를 찾기 위한 최소시간 계획기법으로서 Rajan[8]은 두 지점간의 최소 경로를 찾기 위해 경로를 파라미터화하고 경로 planner로부터 얻은 경로를 주어진 경로와 비교하여 가장 가까운 경로를 찾았고, Shin[9]은 두 지점 사이를 로봇에 의해 이동하는데 필요한 시간의 하위 경계(lower bound)를 도출하고 이 경계를 최소화하는 경로의 형태를 결정하는 방법을 제시하였다. 로봇의 공구단(end-effector)과 목표점간을 최소시간으로 이동하기 위해서는 위에서 언급된 로봇 계획 기법과 물체 획득 시간의 최소화 기법이 필요하다. Croft 와 Fenton [10,12]는 이동중인 물체를 획득하기 위하여 개선된 APPE(Active Prediction, Planning and Execution) 시스템을 제안하였다. 제안된 방식에서 로봇이동곡선을 업데이트 하기 위하여 매회마다 다항식 근사함수를 이용하여 로봇이동곡선

을 업데이트 하였다. 이 방식은 매회 알고리즘을 반복할 때마다 로봇이동곡선을 업데이트 할 경우 많은 시간이 소요된다. 뿐만 아니라 로봇이 목표점에 이동하기 위해서는 반복적인 역기구학 계산이 필요하다. 그러나 컨베이어상에서의 대상물은 시간에 따라 이동되므로 작업물의 위치에 대하여 알고리즘 수행시간은 제한을 받게 된다.

본 논문에서는 물체획득시간을 최소화하기 위하여 로봇이동곡선을 표현하는 세 개의 위치 좌표를 매회마다 교점에 가까워지는 새로운 좌표로 갱신하도록 하였다. 이 좌표를 교점 가까이로 축소 함으로서 실제 로봇이동곡선과 매회 갱신되는 근사로봇이동곡선과의 오차를 감소 시키면서 해의 탐색 영역은 더욱 축소되도록 하였다. 본 논문의 2장과 3장에서는 각각 시스템 개요와 물체 획득계획 및 로봇의 최소시간 획득을 위한 알고리즘을 제시하였다. 그리고 4장과 5장에서 시뮬레이션 결과와 결론을 서술하였다.

### II. 시스템 개요

본 논문에서 제안한 알고리즘을 적용할 컨베이어 시스템의 구성은 그림 1과 같은 고정형 CCD 카메라, FARA AT2(삼성)로봇, 로봇 제어기 그리고 비전처리를 담당하는 마스터 컴퓨터와 컨베이어 장치로 구성된다. 본 시스템에서 컨베이어의 방향은 로봇 좌표(로봇 베이스)에 대하여 X축 방향으로 이동하며 카메라가 Z축에 설치되어 물체를 포착한다.

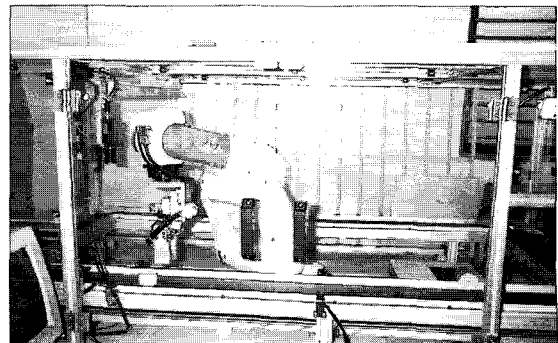


그림 1. 컨베이어 이동장치와 로봇.  
Fig. 1. Conveyor and robot system.

\* 책임저자(Corresponding Author)

논문접수 : 2004. 11. 24., 채택확정 : 2005. 6. 25.

신익상, 문승빈 : 세종대학교 컴퓨터공학과

(sshin@sejong.ac.kr/sbmoon@sejong.ac.kr)

우선 시뮬레이션을 실시하고 그 결과를 토대로 로봇의 실제 실험 장치에 본 알고리즘을 적용시킬 예정이다. 그림 1에서 보이는 바와 같이 작업 영역 중 최초 획득 영역과 마지막 허용 획득 영역 사이에서 최소시간으로 획득 할 수 있는 위치를 여러 후보 영역 중에서 찾는 것이 본 논문의 목적이다.

III. 로봇 이동 계획

1. 물체 획득 모드

물체를 최소 시간으로 획득하기 위해서는 로봇 이동계획이 최적의 성능을 발휘해야 하고 또한 물체를 획득할 적절한 지점을 찾아야 한다. 최적의 획득 지점을 찾기 위해서는 획득 가능한 영역을 먼저 결정 한 후 그 영역 안에서 최소시간으로 물체를 잡을 수 있는 영역을 찾아야 한다. 그림 2와 같이 물체를 획득하는 순간의 상황에 따라 세가지 형태로 분류할 수가 있는데 물체와 로봇이 교점을 이루는 경우, 로봇이 물체보다 일찍 도착하여 대기 하는 경우 그리고 목표점에 대하여 로봇이 나중에 도착하여 물체를 추적하여야 하는 경우이다. 이 세 가지 형태는 모두 컨베이어상에서 발생되므로 물체 궤적은 일 차원 형태로 생각할 수 있다. 따라서 물체 획득 타이밍의 결정문제는 일 차원 탐색과 관련을 갖고 있으며 여러 개의 획득 지점 가운데 최적의 지점을 탐색하는 문제로 볼 수 있다.

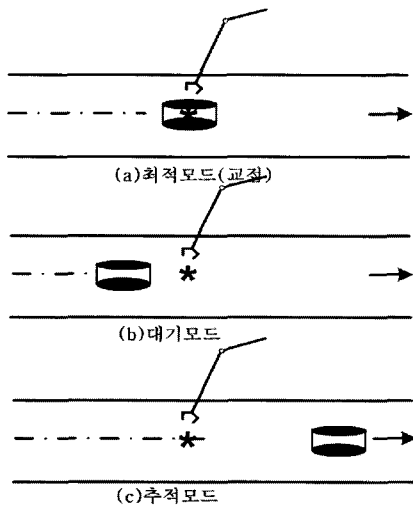


그림 2. 컨베이어 벨트상의 획득모드.  
Fig. 2. Interception modes on a conveyor belt.

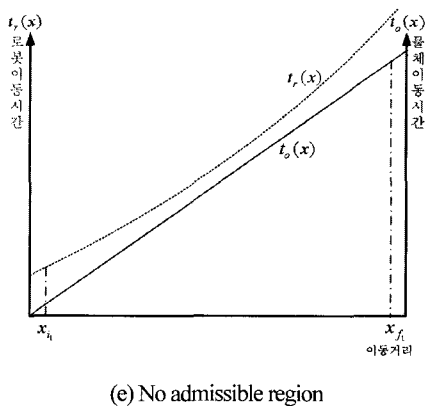
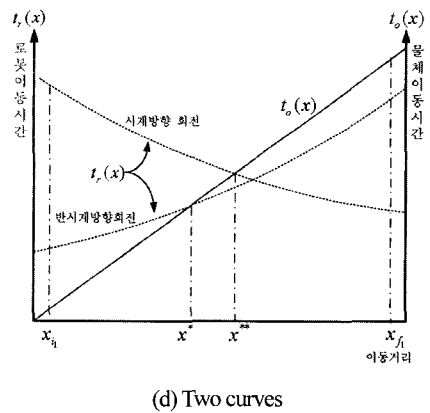
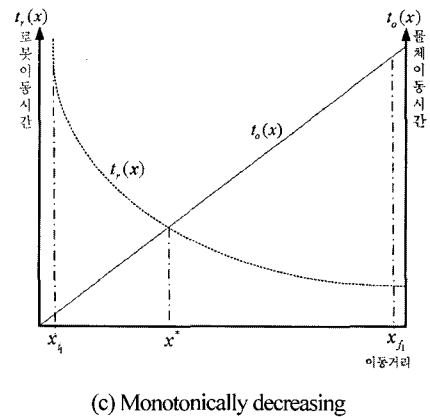
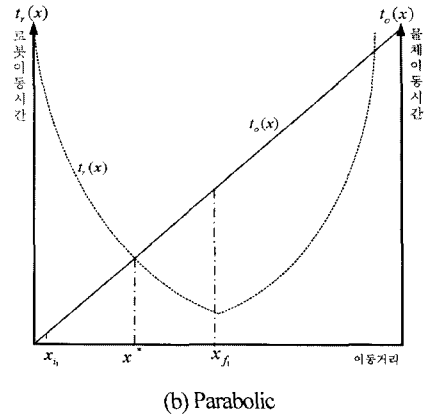
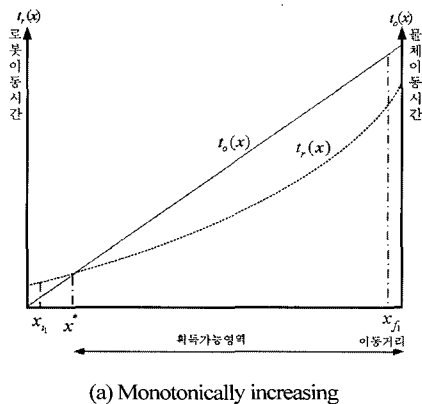


그림 3. 물체이동곡선 및 로봇이동곡선.  
Fig. 3. Object motion curves and robot motion curves.

2. 로봇이동곡선 및 물체이동곡선

컨베이어상에서 이동하는 물체를 로봇이 최소시간에 획득하기 위한 알고리즘을 설명하기 위하여 다음과 같이 로봇이동곡선과 물체이동곡선을 정의한다. 그림 3(a)에서 가로축은 컨베이어상의 위치를 나타내고 세로축은 각각 로봇이 컨베이어상의 특정위치에 도착하는 시간을 나타내는 로봇이동곡선  $t_r(x)$  과 물체가 컨베이어상의 특정위치에 도착하는 시간을 나타내는 물체이동곡선  $t_o(x)$  로 표시 하였다. 그림 3에서 로봇이동곡선은 점선으로 나타내고 물체이동곡선을 실선으로 표시하였다. 여기서,  $[x_{i1}, x_{f1}]$  은 최적해를 구하기 위한 검색 알고리즘에서 사용할 최초의 후보영역이다.

일반적으로 로봇이 이동 물체를 획득하기 위해서는 로봇 이동시간이 컨베이어상의 물체이동시간보다 빨라야 가능하다. 따라서 물체를 획득할 수 있는 획득가능영역(admissible region)은 그림 3(a)에서 로봇이동곡선이 물체이동곡선보다 아래에 있는 영역이 된다.

컨베이어 상에서 이동하는 물체의 이동시간곡선은 일반적으로 미리 알려져 있으며,  $t_o(x) = k \cdot x$  ( $k$ 는 상수)라고 가정한다. 그러나 로봇이동곡선은 일반적으로 미리 알려지지 않고 특히, 그림 4에서 보는 바와 같이 로봇의 초기위치에 따라 달라지게 된다. 즉, 로봇의 공구단(end-effector)이 그림 4(a)와 같이 컨베이어의 진입부에 위치하는 경우에는 그림 3(a)에서 보여주는 바와 같이 단조증가함수 형태를 이루게 되며, 그림 4(b)와 같이 로봇의 공구단이 컨베이어의 중앙에 위치하는 경우에는 로봇이동곡선의 형태는 그림 3(b)와 같이 포물선의 형태가 된다. 이런 경우 그림에서 보는 바와 같이 물체이동곡선과 로봇이동곡선의 교점은 2개가 되나, 최적해는 최단시간에 해당하는 것으로 포물선의 최저점보다 왼쪽에 위치한 것이 된다. 문제를 단순화하기 위하여 이 경우에 초기 후보검색 영역이 포물선의 중앙보다 더 왼쪽에 있다고 가정한다. 이러한 가정은 로봇의 초기 위치가 컨베이어의 중앙에 위치하고 여기서 기다리고 있어도 물체를 획득할 수 있다는 점에서 볼 때, 초기 후보검색영역을 중앙보다 더 시작 위치와 가까운 쪽으로 잡는 것은 타당한 가정이라고 하겠다.

그리고 그림 4(c)와 같이 로봇의 공구단이 컨베이어 상의 마지막 부분에 위치하는 경우 로봇이동곡선의 형태가 그림 3(c)와 같이 단조감소함수 형태를 이루게 된다.

또한, 그림 4(d)와 같이 로봇의 공구단이 로봇후방에 위치하고 있는 경우에는 그림 3(d)와 같이 로봇이동곡선은 로봇 베이스 1축의 초기회전방향이 시계방향인 경우에는 단조증가함수 형태가 되고 시계반대방향인 경우에는 단조감소함수 형태가 된다. 이러한 경우에는 두개의 로봇이동곡선과 물체이동곡선 사이의 해를 각각 구한 후에 둘 중에 더 작은 값으로 최적해를 결정한다.

그림 3(e)와 같이 로봇이동곡선이 물체이동곡선보다 항상 위에 존재하는 경우에는 획득가능영역(admissible region)이 존재하지 않으므로 최적해가 존재하지 않는다.

3. 최소 시간 알고리즘

로봇이 물체를 최소시간에 획득하기 위한 해를 구하는 문제는 로봇이동곡선과 물체이동곡선과의 교점을 찾는 문제가

된다. 이 교점을 찾기 위하여 검색 알고리즘이 필요하게 되고, 최초의 검색후보 영역을 결정하여야 하는바, 이것을  $[x_{i1}, x_{f1}]$  이라고 표시한다.  $[x_{i1}, x_{f1}]$  내에 최적의 해가 유일하게 하나가 존재한다고 가정한다. 이러한 가정은 일반적으로 로봇이동곡선이 주어진 영역내에서 단조함수인 경우에 해당하는 것으로 3.2절에서 살펴본 바와 같이 초기 검색 영역을 알맞게 설정하면 만족할 수 있는 것이다.

일반적으로 로봇이동곡선의 형태는 미리 주어지지 않는다. 물론, 충분한 수의 중간 경로 포인트들에서 값을 구한후 로봇이동곡선을 근사치로 구하는 방법이 있으나, 이 방법은 실시간으로 사용될 수 없다는 결점이 있다. 따라서, 본 논문에서는 주어진 2개의 초기점  $x_{i1}$  과  $x_{f1}$  의 중간점을  $x_{m1}$  이라고 하고 이 세 점을 직선으로 연결하는 근사로봇이동곡선  $\tilde{t}_r(x)$  를 그림 5에서와 같이 구한 후에 이  $\tilde{t}_r(x)$  과 물체이동곡선  $t_o(x)$  와의 교점을 찾아서 최적해를 단계별로 검색하는 알고리즘을 다음에서 제시한다.  $t_r(x_{m1})$  에서의 물체의 위치를  $x_{m1}$  이라고 하면 이 지점에서  $t_r(x_{m1}) = t_o(x_{m1})$  이 됨으로,  $x_{m1}$  은 (1)과 같이 구할 수 있다.

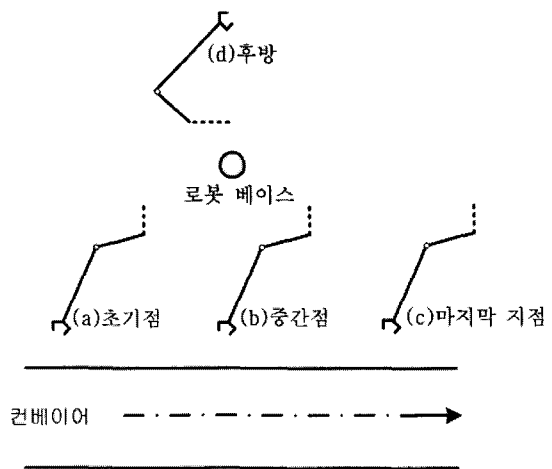


그림 4. 로봇초기위치. Fig. 4. Initial positions of the robot.

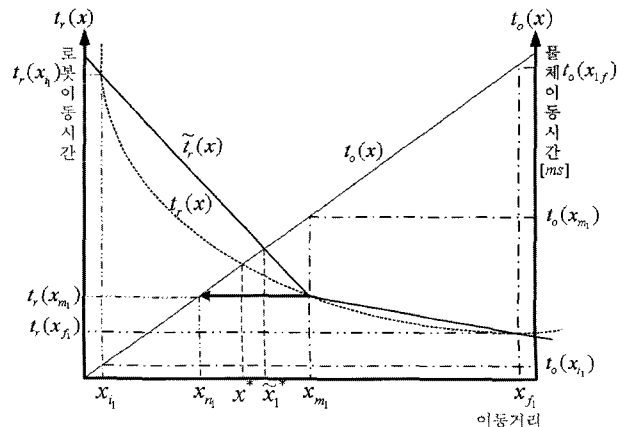


그림 5. 최적해 검색 알고리즘. Fig. 5. Algorithm for searching optimal solution.

$$x_{m_1} = \frac{t_r(x_{m_1})}{k} \quad (\text{여기서 } t_o(x) = k \cdot x) \quad (1)$$

그림 5에서 보는 바와 같이 근사로봇이동곡선  $\tilde{l}_1(x)$  와 물체이동곡선  $t_o(x)$  의 교점을  $\tilde{x}_1^*$  라고 하자.  $\tilde{x}_1^*$  는 첫 번째 단계에서의 교점의 근사 해라고 할 수 있다. 두 번째 반복일 경우  $x_{i_2}, x_{m_2}, x_{f_2}$  는 다음에서  $k=1$ 로 하고 구한다.

(가)  $x_{n_k} \leq x_{m_k}$  인 경우

if ( $x_{n_k} == x_{m_k}$ )

$$x_{i(k+1)} = x_{i_k}, \quad x_{m(k+1)} = \tilde{x}_k^*, \quad x_{f(k+1)} = x_{f_k}$$

else

(a) ( $x_{n_k} \leq \tilde{x}_k^* \leq x_{m_k}$ ) 이면

$$x_{i(k+1)} = x_{n_k}, \quad x_{m(k+1)} = \tilde{x}_k^*, \quad x_{f(k+1)} = x_{m_k}$$

(b) ( $x_{i_k} \leq \tilde{x}_k^* < x_{n_k}$ ) 이면

$$x_{i(k+1)} = x_{i_k}, \quad x_{m(k+1)} = \tilde{x}_k^*, \quad x_{f(k+1)} = x_{n_k}$$

(나)  $x_{n_k} > x_{m_k}$  인 경우

(a) ( $x_{m_k} \leq \tilde{x}_k^* \leq x_{n_k}$ ) 이면

$$x_{i(k+1)} = x_{m_k}, \quad x_{m(k+1)} = \tilde{x}_k^*, \quad x_{f(k+1)} = x_{n_k}$$

(b) ( $x_{n_k} < \tilde{x}_k^* \leq x_{f_k}$ ) 이면

if ( $Y < 0$ )

$$x_{i(k+1)} = x_{n_k}, \quad x_{m(k+1)} = \tilde{x}_k^*, \quad x_{f(k+1)} = x_{f_k}$$

else

$$x_{i(k+1)} = x_{i_k}, \quad x_{m(k+1)} = \tilde{x}_k^*, \quad x_{f(k+1)} = x_{n_k}$$

여기서  $Y$ 는 로봇의 원점을 기준한 좌표.

알고리즘의 매회 반복 시 구해지는  $x_{n_k}$  는 매회 갱신되는 근사로봇이동곡선을 이루는 세 지점 중 한 지점으로 사용된다. 알고리즘의 종료시기는 이전에 계산된 후보 획득지점의 시간과 현재 획득지점의 시간차의 절대값인 (2)의 좌측 항이 임의 상수  $\delta$  값 보다 작으면 알고리즘을 종료 한다.

$$|(t_r(\tilde{x}_k^*) - t_r(\tilde{x}_{k-1}^*))| < \delta \quad (2)$$

여기서 최소시간 물체획득 알고리즘을 다음과 같이 기술하였다.

• 최소시간 획득 알고리즘

Step 1:  $k = 1$ 로 하고  $\tilde{x}_0^* = 0$  으로 설정하며 주어진  $x_{i_1}, x_{f_1}$

을 이용하여  $x_{m_1}$  을 구한다.

Step 2:  $x_{n_k}$  및  $\tilde{x}_k^*$  를 구한다.

Step 3:  $|(t_r(\tilde{x}_k^*) - t_r(\tilde{x}_{k-1}^*))| < \delta$  이면  $x^* = \tilde{x}_k^*$  로 하고

종료하고, 조건이 맞지 않으면 다음 Step으로 이동한다.

Step 4:  $k = k + 1$  으로 하고,  $x_{i_k}, x_{f_k}, x_{m_k}$  구한 후 Step 2로 이동한다.

#### IV. 시뮬레이션

##### 1. 로봇 시뮬레이터

본 논문에서 제시한 최소시간 물체획득 알고리즘의 성능을 알아 보기 위하여 실제 실험환경과 비슷한 환경으로 시뮬레이터를 설계하고 알고리즘을 적용하는 실험을 수행하였다. 시뮬레이터의 외형은 그림 6 과 같으며 시뮬레이터는 메인 스프레드외에 이동 스프레드, 디스플레이를 담당하는 스프레드, 가감속 스프레드, 통신 스프레드 등으로 나누어 각 스프레드간에 데이터 통신이 이루어 지도록 하였다.

##### 2. 시뮬레이션

시뮬레이션 조건은 로봇의 각 축이 동기적으로 동작하며, 최대 180 deg/s로 움직이도록 하였다. 실험에서 기준이 되는 좌표는 로봇의 베이스 (0.0, 0.0, 0.0)을 기준으로 하며 로봇의 초기위치에서 목표 위치로 이동하되 가·감속구간은 최대 속도 까지 320[ms]동안 가·감속하고 등속구간은 최대 속도로 이동한다. 로봇의 1회 모션 계산 시간은 대략 10ms가 소요된다. 이때의 컨베이어 속도는 160mm/s ~ 600mm/s로 가변 가능하고 이 실험에서는 물체는 각각 400, 600mm/s 의 속도로 이동한다. 또한 컨베이어의 목표위치는 보통 작업 방향이 단 방향 이므로 X좌표를 제외한 Y, Z 좌표는 각각 425mm, 200mm로 고정된 위치를 취하며 선형적으로 이동되는 것으로 가정하였다.

그림 7에서 보여주는 바와 같이 세로축은 로봇 및 물체 이동곡선을 표시하며 가로축은 컨베이어상의 이동위치로서 컨베이어상의 물체는 (-)X좌표에서 (+)좌표로 이동된다.

그림 7(a-f)에서 보이는 바와 같이 로봇의 초기위치(그림 8)와 로봇의 회전방향 그리고 컨베이어의 속도(본 실험에서는 400mm/s, 600mm/s) 에 따라 획득위치가 달라짐을 알 수 있으며 특히 후방의 경우 로봇의 회전방향과 컨베이어의 속도가 빨라짐에 따라 획득 실패확률이 높았다. 본 실험에서 획득 실패확률이 높은 경우는 600mm/s일 때 ⑦번 위치에서 로봇의 모든 방향에서 존재하고, 400mm/s일 경우는 ⑤번 위치에서

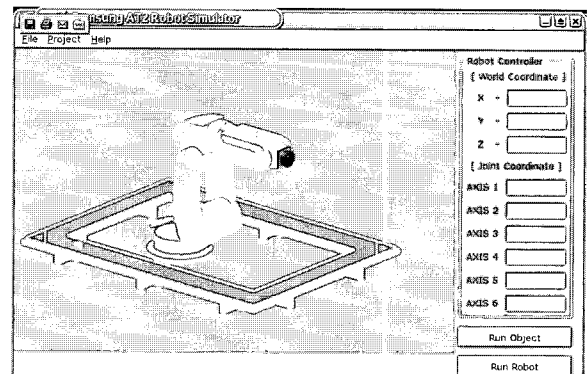
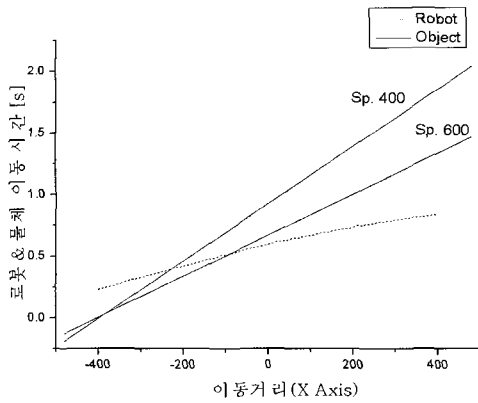
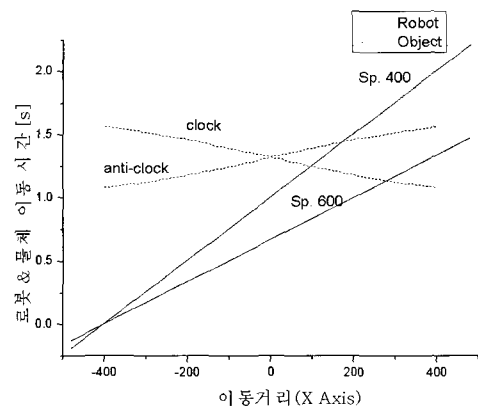


그림 6. 로봇 시뮬레이터.

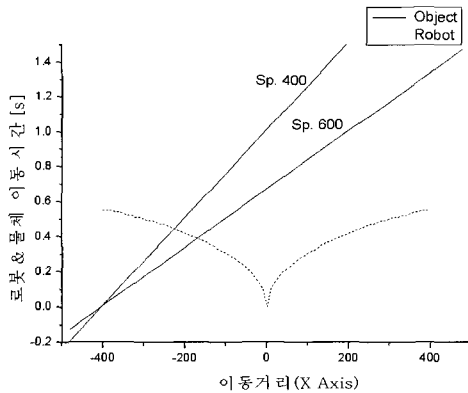
Fig. 6. Robotic simulator.



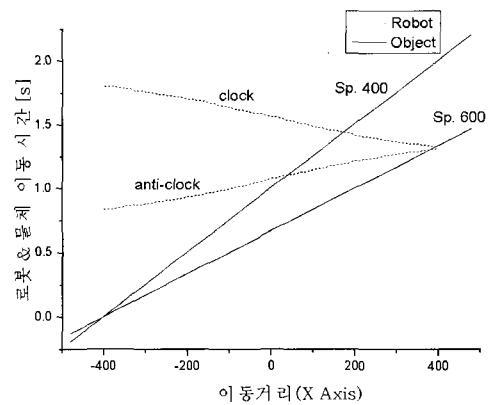
(a) Point ①



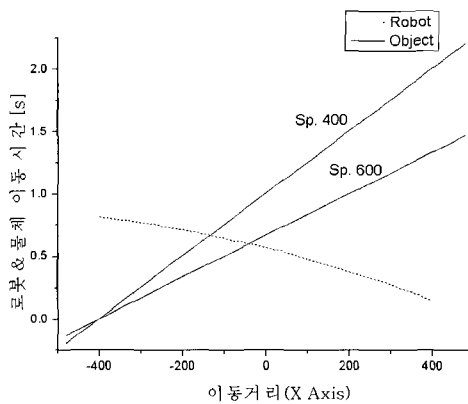
(e) Point ⑥



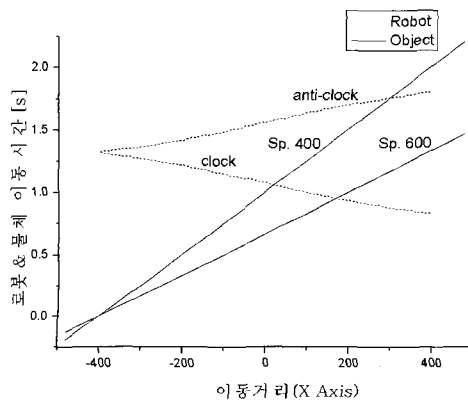
(b) Point ②



(f) Point ⑦



(c) Point ③



(d) Point ⑤

그림 7. 실험결과(그림 8의 초기위치에 따른).

Fig. 7. Experimental results with respect to given initial points in fig.8.

로봇이 반 시계방향 회전 시, 그리고 ⑥번 위치에서 반 시계 방향회전, ⑦번 위치에서 시계 방향 회전인 경우 물체를 획득하지 못할 가능성이 컸다. 또한 로봇이 물체획득에 실패할 경우는 600mm/s일 때 ⑤번 위치에서 로봇이 반 시계방향 회전 시와 ⑥번 위치에서 로봇이 반 시계방향 회전일 경우로 나타났다. 이 알고리즘의 결과는 (2)를 기준으로 측정한 데이터로서 기준 상수  $\delta$  보다 오차가 작으면 수렴하게 된다. 따라서 표 2-1은 알고리즘의 수렴결과를 보이고 있으며 표 2-1, 2-2에서 보이는 바와 같이 각각 컨베이어 속도가 각각 400mm/s와 600mm/s일 경우 이진알고리즘과 최소시간 알고리즘의 비교 결과를 보이고 있다.

본 결과에서 각 로봇의 초기 위치와 컨베이어의 속도에 따라 최적 좌표지점이 달라지며 오차는 알고리즘이 수렴하였을 경우 실제물체이동좌표와 최적좌표간의 차이를 나타낸 것으로서 경우에 따라 이진 알고리즘이 적은 경우도 발생하였다. 그러나  $\delta$  을 좀더 높일 경우 반복회수와 정밀도면에서 제안된 알고리즘이 우수한 것으로 나타났다. 본 실험에서 가장 빨리 획득할 수 있는 로봇의 초기좌표 제안 알고리즘과 이진알고리즘이 모두 동일하게 400mm/s일 경우 ①번 위치이고 600mm/s일 경우 ②번 위치였고, 가장 늦게 물체를 획득하는 곳은 모두 ⑦위치로 나타났다.

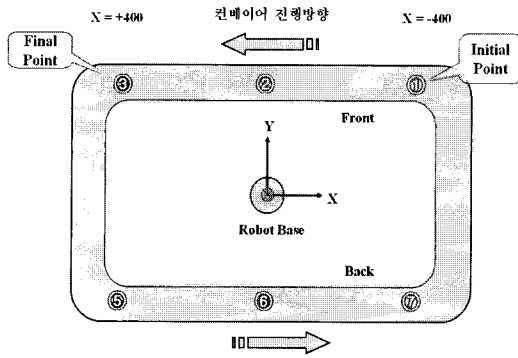


그림 8 컨베이어상의 로봇의 초기 위치.  
Fig. 8. Robot's initial points for experiment.

표 1. 제시 알고리즘 결과.

Table 1. The result of proposed algorithm.

속도 (mm/s)	반복 회수	최적 좌표 (mm)	로봇이동 시간(ms)	실제 좌표 (mm)	오차 (mm)	
400	①	3	-234.8	413	-234.7	0.05
	②	2	-223	438	-224	1
	③	2	-138	658.8	-136	2
	⑤	1	24.4	1059	23.6	0.79
	⑥	2	97.9	1246	98.4	0.5
	⑦	2	174	1433	73.2	0.8
	600	①	2	-92.4	512	-92.8
②		2	-168.7	386	168.4	0.31
③		1	-49.9	588	47.2	2.69
⑤		2	172.2	953	171.8	0.42
⑥		2	280.7	678.6	278.6	2.14
⑦		1	391.7	1319	391.4	0.3

표 2. 이진 알고리즘 결과.

Table 2. The result of binary search algorithm.

속도 (mm/s)	반복 회수	최적 좌표 (mm)	로봇이동 시간(ms)	실제 좌표 (mm)	오차 (mm)	
400	①	7	-231.3	421	-231.6	0.35
	②	10	-224.2	438	-224.8	0.6
	③	6	-137.5	658.8	-136.5	1.02
	⑤	8	21.9	1058	23.2	1.33
	⑥	11	98.1	1247	98.8	0.75
	⑦	10	174.2	1431	172.4	1.82
	600	①	8	-90.6	513	-92.2
②		8	-165.6	386	168.4	2.78
③		7	-43.75	590	46	2.25
⑤		8	171.9	953	171.8	0.07
⑥		9	279.7	1131	278.6	1.08
⑦		9	392.2	1320	392	0.18

V. 결론

본 논문에서는 컨베이어상의 이동물체 획득을 위한 로봇의 최소시간 알고리즘을 제안하였다. 로봇이 최소시간으로 물체를 획득하기 위해서 로봇과 물체의 교점을 포함하는 탐색 영역을 빠르게 축소함으로써 탐색 알고리즘 최적화 하였다.

본 논문에서 제시하는 알고리즘은 반복회수와 정확도 면에서 이진 탐색 알고리즘 보다 좀더 나은 결과를 보여 주었다. 제시 알고리즘은 로봇을 이용한 생산시스템에서 여러 가지의 물체가 컨베이어상에서 이동되는 경우 이 물체를 주어진 시간 내에 획득하여 다른 장소에 놓아야 하는 경우 시간 제한이 다르게 된다. 이러한 환경에선 물체의 최소시간 획득이 필요하다. 따라서 본 알고리즘은 여러 개의 물체가 랜덤하게 진행되는 환경에서 본 알고리즘이 적용될 수 있으리라고 사료 된다.

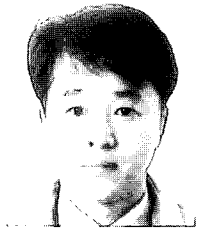
참고문헌

- [1] R. Sharma, J. -Y. Herve, and P. Cucka, "Dynamic robot manipulation using visual tracking," *Proc. of IEEE Int. Conf. Robotics and Automation*, Nice, France, pp. 1844-1849. May 1992.
- [2] P. K. Allen, A. Timcenko, B. Yoshimi, and P. Michelman, "Automated tracking and grasping of a moving object with a robotic hand-eye system," *IEEE Trans. Robotics and Automation*, vol. 9, pp. 152-165, Apr. 1993.
- [3] M. Jarabek and David W. Capson, "Robot position servoing using visual gap measurement," *IEEE Instrumentation and Measurement Tech Conf*, St. Paul, Minnesota, pp. 26-30, May 1998.
- [4] 최규중, 조월상, 안두성, "이동물체 포착을 위한 비전 서보 제어 시스템 개발" 한국동력기계학회지, pp. 96-101, 2002.
- [5] N. P. Papanikolopoulos, P. K. Khosla, and T. Kanade, "Visual tracking of a moving target by a camera mounted on a object: a combination of control and vision," *IEEE Trans. Robotics and Automation*, vol. 9 pp. 14-35, Feb. 1993.
- [6] N. P. Papanikolopoulos and P. K. Khosla. "Feature based robotic visual tracking of 3-D translational motion" *Proc of the 30th IEEE Conf Decision and Control*, pp. 1877-1882, Dec 1991.
- [7] D. P. Tsakiris, "Visual tracking strategies" Master's thesis Dept. of Electrical Eng., Univ. of Maryland College Park, 1988.
- [8] V. Rajan, "Minimum time trajectory planning" *IEEE Int. Conf. Robotics and Automation*, pp.759-764, 1985.
- [9] K. Shin and N. D. McKay, "Selection of near-minimum-time geometric paths for robotic manipulators", *IEEE Trans Automatic Control*, vol. 31 Issue. 6, pp. 501-511, June 1986.
- [10] E. A. Croft, R. G. Fenton, and B. Benhabib, "Time-optimal interception of objects moving along predictable paths," *IEEE Int Symp Assembly and Task Planning*, pp.419 - 425,10-11 Aug. 1995.
- [11] 방성호, 문승빈, "연속된 영상정보를 이용한 물체 추적 시스템 구현," 제18회 한국 정보처리학회 추계 학술 발표대회 논문집 제9권 제2호, pp. 653-656, Nov. 2002.
- [12] D. Hujic, E. A. Croft, G. Zak, R. G. Fenton, J. K. Mills, and B. Benhabib, "The robotic interception of moving objects in industrial setting: strategy development and experiment" *IEEE*

*Trans Mechatronics*, pp. 225-239, Sept. 1998.

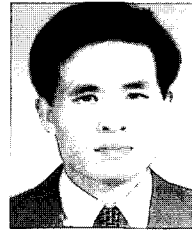
[13] D. Rembold, U. Zimmermann, T. Langle, and H. Worn,

"Detection and handling of moving objects," *IEEE IECON*, pp. 1332-1337, 1998.



**신익상**

1996년 한밭대학교 전기공학과 졸업(학사). 1998년 명지대학교 전기공학과 졸업(석사 전공 : 전기기기 및 전력전자). 2003년 세종대학교 컴퓨터공학과 박사 수료(로보틱스 전공). 관심분야는 로보틱스, 컴퓨터 비전.



**문승빈**

1985년 명지대학교(공학사). 1988년 Univ. of Michigan(공학석사). 1993년 Purdue Univ., U.S.A.(공학박사). 1986년 전자통신연구소 연구원. 1993년~1999년 삼성전자 수석연구원. 1999년~현재 세종대학교, 부교수. 관심분야는 RTOS기반 로봇 제어기, 임베디드 시스템, 컴퓨터 비전.