

본 연구는 1996년도 교육부 기계공학 학술연구조성비의 지원으로 수행되었음

## 다수의 로봇을 이용한 컨베이어상의 조립순서 계획

박 장 현\*

### Assembly Sequence Planning for Multiple Robots Along a Conveyer Line

Jahng-Hyon Park\*

#### ABSTRACT

In order to increase productivity of an assembly system composed of multiple robots along a conveyer line, an efficient sequence planning is necessary because the assembly time is dependent upon the assembly sequence. In this paper, a two-robot assembly system is considered in which two robots operate simultaneously and transfer parts from the part feeders to the workpiece on the conveyer one by one. In this case, the distance from the feeder to the workpiece varies with time because the workpiece moves at a constant speed on the conveyer. Hence, the sequence programming is not a trivial problem. Also, the two robots may interfere with each other kinematically and dynamically due to the simultaneous operation, so the sequence should be programmed to avoid the interferences. In this paper, the task sequence optimization problem is formulated and is solved by employing the simulated annealing which has been shown to be effective for solving large combinatorial optimizations.

**Key Words** : Multiple robot (다수의 로봇), Assembly (조립), Sequence optimization(작업순서 최적화), Simulated annealing(모사풀림기법)

#### 1. 서 론

조립공정의 생산성 향상을 위하여 조립기계의 고속작업이 요구되고 있다. 특히 전자제품조립의 경우, 현재의 조립기들은 대부분 이미 고속의 작업으로 생산성이 양호하나 부품공급기를 포함한 거대한 구조물로 이루어져 있고 고속, 고가속도 운동을 한다. 따라서, 커다란 질량의 고가속도 운동에 의한 진동 및 반력 때문에 현존하는 기계구

조로는 작업속도 및 생산량을 더 이상 증대시키기 어려운 실정이다. 이러한 점을 감안하여, 다수의 로봇과 컨베이어로 이루어진 새로운 조립시스템이 최근에 Li 등에 의하여 제안된 바 있다<sup>(1)</sup>.

이 시스템은 컨베이어 주위에 분산되어 위치한 로봇들이 일정 속도로 이동하는 컨베이어 위의 기판에 반도체를 올려놓는 작업을 동시에 수행하도록 되어 있다. 이러한 구조의 조립시스템은 그 구조가 비교적 단순하나, 그 운영이

\* 한양대학교 기계공학부

복잡한 면이 있다. 즉, 컨베이어 상의 기판이 연속적으로 이동하므로 반도체의 출발지인 부품공급기로부터 목적지인 기판까지의 거리가 시간에 따라 변화하여 로봇의 운동시간이 달라지므로 조립순서에 따라 전체의 작업시간에 영향을 미친다. 따라서 생산성 향상을 위해서는 작업순서 계획이 필요하다.

또한 다수의 로봇이 움직이는 컨베이어 위에서 동시에 작업을 하게 되므로 로봇들 상호간의 간섭현상이 일어날 수 있다. 그 현상으로는 로봇간의 충돌을 의미하는 기하학적 간섭뿐만 아니라, 고가속도 운동에 의한 거대한 반력 때문에 동역학적 간섭이 가능하다. 그러므로 다수의 로봇을 위한 작업순서를 계획하는데 있어 이러한 간섭현상을 고려하여야 할 것이다. 두 대의 로봇의 충돌을 방지하기 위한 경로계획은 Lee 등<sup>(2)</sup> 및 Shin 등<sup>(3)</sup>에 의하여 제시된 바가 있다. 그들은 기하학적인 경로를 먼저 결정한 후, 그 경로상의 움직임을 시간에 대한 함수로 나타내었다. 최단 시간에 동작을 완료하기 위하여, 두 대 중에 한 대의 로봇은 원위치에서 일정기간동안 정지상태에 있어야 됨을 증명하였다. 그러나 이러한 방법은 위에서 언급한 동역학적인 간섭 때문에 본 조립시스템에 그대로 적용할 수 없다. 어느 한 로봇이 부품을 올려놓기 위해 정밀한 위치제어를 시도하고 있을 때 다른 로봇이 고가속도 운동을 한다면 외란에 의하여 정밀도에 영향을 미칠 수 있기 때문이다. 이러한 동역학적인 간섭현상을 배제하기 위해서는 두 대의 로봇이 동시에 작동하여야 한다.

본 논문에서는 두 로봇의 경로와 양끝 점을 적절히 배열함으로써 두 로봇간의 충돌을 방지할 수 있다고 가정한다. 일정 속도로 움직이는 컨베이어 위에 조립을 하는 두 대의 로봇의 작업순서를 작업시간에 대하여 최적화시키는 문제를 연구한다. 이러한 조립순서의 최적화 문제를 풀기 위하여 본 논문에서는 대형의 조합최적화 문제에 효율적인 모사풀림기법(Simulated Annealing)을 사용할 것이다.

## 2. 두 로봇의 동작

위에 언급된 조립시스템은 그림 1에 보는 바와 같이 두 대의 로봇과 부품 공급기, 컨베이어로 이루어져 있다. 여러 개의 피이더가 배열된 부품 공급기는 거리상 어느 한 로봇이 와 닿을 수 없는 부분이 있다. 따라서 각 로봇이 도달할 수 있는 공간을 구분할 필요가 있다. 각각의 피이더에는 일련번호가 정해져 있고 로봇 A와 B의 도달 가능

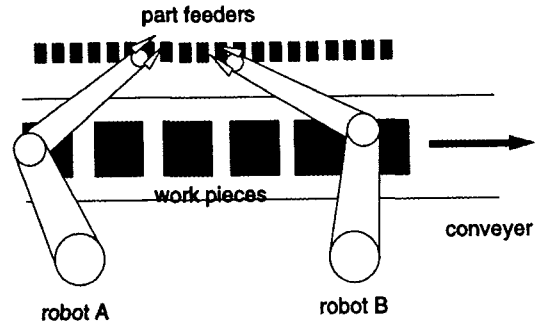


Fig. 1 Dual-robot assembly station with a conveyor

공간을 각각 A, B라고 정의한다. 조립될 부품들은 피이더에서 로봇 손끝에 달려 있는 그리퍼에 의하여 컨베이어상의 작업물위에 옮겨져 조립된다.

한개의 작업물위에 N개의 부품이 조립되고 N은 짝수라고 가정하자. 각각의 로봇은 N/2개의 부품을 조립하게 되고 N/2번의 동시작업을 통하여 하나의 작업물이 조립된다. i번째 작업 시에 로봇 A, B가 집어 올릴 부품의 피이더 번호를 각각  $n_i^A$ ,  $n_i^B$ 라고 하자. 한 개의 작업물을 조립하기 위한 N/2번의 작업의 순서는 다음과 같다.

$$\begin{aligned} n^A &= (n_1^A, n_2^A, \dots, n_{N/2}^A) \\ n^B &= (n_1^B, n_2^B, \dots, n_{N/2}^B) \end{aligned} \quad (1)$$

이 때,  $1 \leq i \leq N/2$ 에 대하여  $n_i^A \in A$ ,  $n_i^B \in B$ 이다.

또한 컨베이어 상의 공간을 살펴보면 여러 개의 작업물들이 일정속도로 움직이고 있다. 그 중에서 M개의 작업물이 로봇에 의하여 도달 가능한 공간에 있고 그들은 일련번호로 표시된다고 하자. 그러면 각 로봇은 부품공급기에서 집어 올린 부품을 어느 작업물 위에 올려 놓는가하는 문제에 접한다. i번째의 작업에서 로봇 A, B가 부품을 내려놓는 작업물의 번호를 각각  $m_i^A$ ,  $m_i^B$ 라고 하면 조립순서를 나타내는 벡터는 다음과 같이 표시된다.

$$\begin{aligned} m^A &= (m_1^A, m_2^A, \dots, m_{N/2}^A) \\ m^B &= (m_1^B, m_2^B, \dots, m_{N/2}^B) \end{aligned} \quad (2)$$

이 때,  $m_i^A, m_i^B \in \{1, 2, \dots, M\}$ .

그런데 앞 절에서 언급한 바와 같이 로봇 상호간의 기하학적 간섭, 즉 충돌 가능성 때문에 두 대의 로봇이 동시에 동일하거나 근접한 공간에 들어서는 것을 방지하여야

한다. 그러므로 작업순서를 결정하는데 있어 이를 고려한 구속조건이 마련되어야 한다.

부품공급기 쪽을 살펴보면 두 대의 로봇이 동일한 부품 피이더에 동시에 접근할 수 없고 일정한 간격을 유지하여야 한다. 예를 들어 두 대의 로봇은 공간  $A \cap B$ 에 동시에 진입할 수 없다. 즉, 만약에  $n_i^A \in A \cap B$  이라면  $n_i^B \notin A \cap B$  이고 그 반대의 경우도 마찬가지이다. 이 경우에 각 공간집합의 원소의 수는  $n_A, n_B, n_{A \cap B}$  로 표시되고  $N = n_A + n_B - n_{A \cap B}$  의 조건이 성립된다.

이송기 상의 작업물 쪽을 살펴보면, 기구학적인 구속조건은 두 로봇이 접근하는 두 작업물 사이의 거리를 최소 기준치인  $d_{min}$  이상으로 유지함으로써 만족시킬 수 있다. M개의 작업물 중에서 j번째 작업물의 위치를 이송기의 길이방향의 좌표로 표시하였을 때  $d(j)$ 라고 하자. 이 때의 충돌방지를 위한 조건은  $d(m_i^A) + d_{min} < d(m_i^B)$  으로 나타낼 수 있다.

### 3. 최적화 문제의 설정

본 논문에서 고려하고 있는 조립작업을 다시 기술하면 다음과 같다: 각 로봇은 피이더로부터 부품을 공급받아 이송기상의 작업물위의 주어진 위치에 부품을 올려놓는다. 그 다음에 조립할 부품이 장전되어 있는 피이더로 이동하여 공급받은 후 같은 작업을 계속 수행한다. 컨베이어 위에는 로봇의 운전 가능한 영역 안의 M개의 작업물이 일정 속도로 흐르고 있다. 이 때, N개의 부품을 하나의 작업물 위에 모두 조립하는 데에 걸리는 시간을 최소화시키는 것이 본 연구의 주목표이다.

그림 2는 조립과정에서의 로봇의 움직임을 보여 준다. 앞 절에서 설명된 동역학적 간섭현상을 피하기 위하여 두

로봇의 동작은 동시에 진행되어야 한다. 어느 한 로봇이 작업물위에서 조립을 위하여 정밀한 위치제어를 수행하고 있을 때 다른 로봇이 고가속도의 운동을 한다면 로봇 작업 및 컨베이어 운동의 정밀도에 악영향을 미칠 수 있기 때문이다. 따라서 두 대의 로봇 중 작업물 위에 먼저 도착한 로봇은 다음의 피이더로 이동하기 전에 늦게 도착한 로봇이 조립작업을 마칠 때까지 현위치에서 대기하여야 한다. 로봇 A, B가 i번째 순서에서 조립에 걸리는 시간을 각각  $T_i^A, T_i^B$  라고 하면 i번째 순서의 조립시간은 둘 중에 더 큰 값이 된다:

$$T_i = \max(T_i^A, T_i^B) \tag{3}$$

조립시작 후 i번째 순서가 종료된 후의 시간을  $t_i$ 라고 하면 다음의 관계가 성립된다.

$$t_{i+1} = t_i + T_i \tag{4}$$

여기에서 최적화 문제는 두 로봇에 대하여 각각의 피이더와 작업물의 순서를 결정하는 문제로 정의되며, N개의 부품을 조립하기 위하여 N/2번의 동작이 필요하므로 다음과 같은 목적함수를 갖는다.

$$t_{N/2} = \sum_{i=1}^{N/2} T_i \tag{5}$$

매 순서마다의 로봇  $l(l = A, B)$ 의 작업시간  $T_i^l$ 를 구하기 위해서는 각 로봇의 이동시간  $T_{travel}^l$ 를 구해야 하는데 이는 각 운동의 시작점과 끝점의 위치, 로봇 팔의 특성 및 모터의 용량, 제어기에 의하여 결정된다. 로봇팔 및 구동장치가 주어져 있다고 보면  $T_{travel}^l$ 는 운동의 양끝 점에 의해서만 결정된다. i번째 순서에 피이더 j에서 작업물 k에 부품이 이동된다고 하자. 피이더 j의 위치를  $x_p(j)$ 로, 작업물 k의 위치를  $x_b(k)$ 로 나타내면 로봇 l의 이동시간은 다음과 같다.

$$\begin{aligned} T_{travel_i}^l &= f^l(x_p(j), x_b(k)) \\ &= f^l(x_p(n_i^l), x_b(m_i^l)) \end{aligned} \tag{6}$$

여기에서 함수 f의 형태는 로봇의 제어시스템에 의존한다. 1절에서 기술된 로봇링크는 직접구동(Direct Drive) 모터로 구동된다. 로봇 팔이 두 링크간의 상호작용이 작도록 설계되었다고 가정하면 로봇의 이동시간은 각 관절

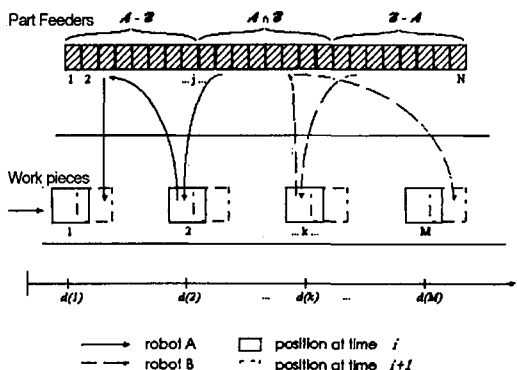


Fig. 2 Coordinated motion of the two robots

의 변위량에 의하여 결정된다. 이를 위하여 직교좌표계로 나타낸 양끝 단의 위치  $x = (x \ y)^T$ 를 관절에 대한 좌표계  $q = (\theta_1, \theta_2)^T$ 로 나타내도록 한다. 좌표변환의 관계식  $x = g(q)$ , 즉  $q = g^{-1}(x)$ 를 이용하여 각 동작의 시작점  $x_s$ 과 끝점  $x_f$ 를  $q_s$ 와  $q_f$ 로 각각 좌표변환을 시킨다. 그러면 각 관절은  $|\theta_{is} - \theta_{if}|$  ( $i = 1, 2$ )만큼의 변위를 갖게 된다. 각 모터의 최대 토오크를  $\tau_{i, \max}$ , 각 링크의 관성모멘트를  $J_i$ 라고 하고 각 관절에서 최대 토오크로 가속 및 감속을 한다고 가정하면 각 관절의 이동시간은 다음과 같이 주어진다.

$$T_{\theta_i} = 2 \sqrt{\frac{J_i |\theta_{is} - \theta_{if}|}{\tau_{i, \max}}} \quad (7)$$

위에서 가장한대로 두 관절이 완전히 독립적이라면 로봇의 이동시간은 각 관절의 이동시간 중 긴 쪽에 의해서 결정될 것이다.

$$T_{travel} = \max(T_{\theta_1}, T_{\theta_2}) \quad (8)$$

부품을 집어 올리고 내리는 시간  $T_{pick}$ ,  $T_{place}$ 을 고려하면 로봇이  $i$ 번째 작업에서 부품을 공급기에서 작업물에 조립하는데 걸리는 전체시간은 다음과 같이 주어진다.

$$T_i^l = T_{pick} + T_{place} + T_{travel}^l + T_r^l \quad (9)$$

여기에서  $T_{travel}^l$ 은 피이더에서 작업물까지의 이동시간,  $T_r^l$ 은 작업물에서 다음 순서에 조립하여야 할 피이더까지의 이동시간을 나타낸다.

#### 4. 모사플립기법

본 논문에서 활용하려는 모사 플립기법은 TSP(traveling salesman problem)을 비롯한 조합의 최적화 문제에 이용되어 왔다. 이 방법은 TSP, 회로설계, 조립방법설계, 생산계획<sup>(9, 10)</sup> 등에 유용하게 응용된 바가 있다. 본 논문에서 다루는 문제는 N개의 부품을 조립하는 데에 로봇의 이동순서를 최적화 함으로써 전체의 작업시간을 단축하는 데에 목적이 있다. 따라서, 주어진 모든 도시를 여행하면서 최소 비용이 드는 경로를 정하기 위한 TSP 문제와 흡사하다고 할 수 있다. 그러나, 다음과 같은 차이점들이 있다.

1. 두 사람(로봇)의 이동경로를 구한다.

2. 목적지(작업물)의 위치가 시간에 따라 변한다.
3. 각 목적지 도착 후 집(부품공급기)으로 다시 귀환한다.
4. 이동방향에 따라 비용(시간)이 달라진다. 즉,  $d_{ij} \neq d_{ji}$ .
5. 앞 단계에서의 경로(작업순서)에 따라 비용(시간)이 달라진다. 즉,  $d_{ij}$ 가 일정치 않다.

이와 같은 문제를 풀기 위하여 가능한 경우를 모두 탐색하는 것은 적합하지 않은 방식이다. N이 증가함에 따라 경우의 수는 N! (factorial)로 증가하여 계산량이 크게 증가하여 탐색이 불가능해진다. 지금까지 TSP와 같은 문제를 풀기 위한 여러 방법들이 개발되어 왔으나 특수한 경우를 제외하고는 다항식의 차수( $N^p$ ) 이내의 시간 내에 정확한 답을 구할 수 있는 방법이 아직 존재하지 않는다<sup>(4)</sup>. 대신에  $O(N^p)$ 차수 이내의 시간에 근사적인 답을 구하기 위한 방법들이 시도되어 왔고 그 중에서 Kirkpatrick 등이 고안한 모사플립 방법이 TSP를 비롯한 대형의 조합 최적화 문제를 푸는데 있어서 효율적임이 입증되었다<sup>(5)</sup>.

모사플립방법은 금속재료의 플립 과정에서 온도를 서서히 낮춤에 따라 재료내의 격자들이 가장 안정된 구조로 결정되어 가는 과정을 모방한 것이다. 최적 값을 찾기 위한 단순강하법에서는 기존의 이동순서를 임의의 방향으로 변화를 시킨 다음, 목적함수가 향상되었을 때에만 새로운 순서로 받아들이고, 이 작업을 반복하여 결국에는 국소 최적값 (local optimum)에 도달하게 된다. 이에 반하여, 모사플립 방법은 목적함수가 향상된 경우뿐만 아니라 목적함수가 일시적으로 악화되는 경우에도 통계적인 방법을 이용하여 새로운 조합을 받아들일 수 있다. 이 경우의 확률은 임의로 변경된 조합에 의한 목적함수의 변화량  $\Delta E$ 와 온도라고 불리는 매개변수 T에 다음과 같이 의존한다.

$$p = \exp\left(-\frac{\Delta E}{T}\right) \quad (10)$$

따라서, 목적함수의 악화되는 정도 ( $\Delta E$ )가 커질수록 새로운 조합이 받아들여질 확률이 작아진다. 또한, 온도 T가 높을수록 분자의 운동이 활발하듯이 목적함수의 증가가 받아들여질 확률이 커진다. 온도가 점차로 낮아져서  $T=0$ 인 경우에는 더 이상의 목적함수의 증가는 불가능하여져서 설계변수의 조합은 동결된다. 따라서 모사플립법의 반복작업 중에는 온도 T의 적절한 조절이 필요하다: 초기의 온도는 지역 최적점에서 벗어날 수 있도록 하기

위하여 높은 값이 설정되어야 하며 온도강하는 최적점이 아닌 곳에서의 동결을 막기 위하여 매우 느리게 진행되어야 한다<sup>6)</sup>.

### 5. 조립 순서 계획

3절에서 기술한 조립시간의 최적화 문제는 4절에서 설명된 모사풀림을 이용하여 다음과 같이 프로그램화되었다. 부품 피이더의 순서  $n = (n^A, n^B)$ 와 작업물의 순서  $m = (m^A, m^B)$ 의 초기치는 임의로 발생시켰다. 각각의 순서는 임의의 방향으로 소량의 변화를 주어  $n'$ 와  $m'$ 으로 바꾼다. 새로운 순서조합에 대한 목적함수(작업시간)  $E'$ 를 계산하여 원래의 목적함수  $E$ 와 비교한다. 성능에 향상이 있는 경우 ( $\Delta E = E' - E < 0$ )에는 새로운 순서를 채택하고,  $\Delta E > 0$ 인 경우에도  $\exp(-\Delta E/T)$ 의 확률로써 그 순서를 다음의 반복을 위하여 채택된다. 더 이상의 향상이 관찰되지 않을 때까지 매 반복 시마다 온도  $T$ 는 소량씩 감소시킨다.

모사풀림기법의 효율성은 모든 경우를 탐색하는 방법에 비하여 계산량이 크게 감소되면서도 최적점에 거의 근사한 답을 찾아낼 수 있다는 것이다. 예를 들어, 조립할 부품의 수가  $N=20$ 이고  $n_A = n_B = 10$ ,  $n_{A \cap B} = 10$ 인 경우를 보면, 정지해 있는 하나의 작업물 ( $m=1$ ) 위에 조립을 할 때 두 로봇의 작업 순서의 가능한 경우의 수는  $2.77 \times 10^{13}$ 이고 작업물이 움직인다고 하면 경우의 수는  $3.62 \times 10^{25}$ 이 된다. 부품의 수와 작업물의 수가 증가함에 따라 경우의 수는 매우 급격히 늘어나므로 모든 경우를 일일이 탐색한다는 것은 불가능한 일이다.

모사풀림의 실패를 들기 위하여 다음의 경우를 예로 든다. 컨베어 위에는 6개의 작업물이 로봇의 작업공간 안에 있으며, 각 작업물에 20개의 서로 다른 부품을 올려놓아야 한다. ( $N=20$ ,  $\{1, \dots, 6\} \subset A-B$ ,  $\{7, \dots, 14\} \subset A \cap B$ ,  $\{15, \dots, 20\} \subset B-A$ ,  $M=6$ ). 그리고 당분간, 컨베어는 정지해 있다고 가정한다.

위와 같은 조건을 이용하여 계산이 수행되었다. 그림 3에는 모사풀림기법을 이용한 경우를 실선으로 나타내었고 이용하지 않은 경우(단순강하법)를 점선으로 비교하였다. 반복작업의 초기에는 온도가 높기 때문에 모사풀림법의 경우 에너지(목적함수)의 등락이 심하다. 반복의 횟수가 증가할수록 온도가 낮아져 나중에는 고에너지 상태로의 천이가 줄어들다. 에너지의 증가는 최소화하려는 방향과는 반대방향으로 이동하므로 수렴속도가 느린 편이

다. 단순강하법은 초기에는 빠른 속도로 수렴하나 지역 최소점에 걸려서 빠져 나오지 못함을 볼 수 있다. 따라서, 마지막의 결과는 모사풀림법이 우수함을 알 수 있다.

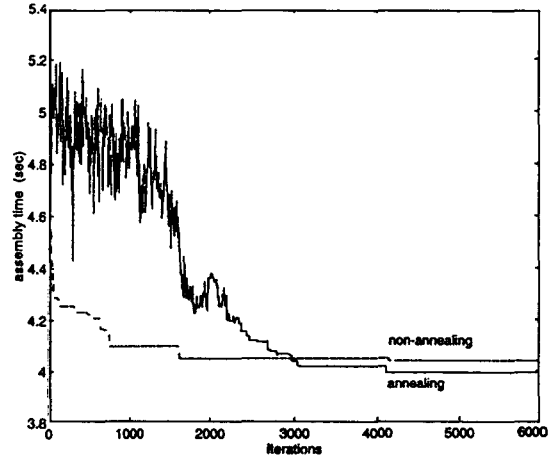


Fig. 3 Optimization iterations

### 6. 컨베어가 정속운전하는 경우

컨베어가 일정속도  $v_{conv}$ 로 움직이는 경우를 고려하자. 작업물은 컨베어 상에서 이동하므로 한 순간의 작업물의 위치는 시간의 함수이며 앞 단계에서의 동작이 얼마나 오래 걸렸는가에 의하여 결정이 된다.

$$x_b(t) = x_b(0) + v_{conv}f \quad (11)$$

따라서, 어느 한 단계에서의 이동시간은 앞 단계에서의 작업시간에 따라 달라진다. 컨베어의 이동속도  $v_{conv}$ 가 주어졌을 때 시간에 따른 작업물들의 위치가 결정이 되며 이에 따라 최적의 조립순서를 정할 수 있다.

모사풀림방법에서는 주어진 작업순서에 소량의 변화를 주어서 다음 단계에서 시도할 새로운 순서를 만들게 되는데 주어진 순서를 변화시키는 방법에 따라 수렴속도가 달라진다. TSP의 경우, 순서변화의 주 방법으로 k-opt transition<sup>7)</sup>을 들 수 있다. 그러나 이 방법은 본 논문에서 다루고 있는 순서 계획에는 사용할 수 없다. 그 이유는 순서변화 후에도 충돌 방지를 위한 기하학적 구속조건을 항상 만족시켜야 하기 때문이다.

가장 간단한 방법으로는 그림 4a에서 보는 바와 같이 두 로봇의 작업순서  $n^A$ 와  $n^B$ 의 임의의 두 요소를 서로 맞바꿈으로써 새로운 순서를 만드는 것이다. 단, 이 때 기

$$\begin{cases} \pi_A = (1\ 2\ 3\ \textcircled{4}\ 5\ 6\ 7\ 8\ 9\ 10) \\ \pi_B = (1\ 2\ 3\ 4\ 5\ 6\ \textcircled{7}\ 8\ 9\ 10) \end{cases}$$



$$\begin{cases} \pi_A = (1\ 2\ 3\ \textcircled{7}\ 5\ 6\ 7\ 8\ 9\ 10) \\ \pi_B = (1\ 2\ 3\ 4\ 5\ 6\ \textcircled{4}\ 8\ 9\ 10) \end{cases}$$

or

$$\begin{cases} \pi_A = (1\ 2\ \textcircled{3}\ 4\ 5\ 6\ 7\ \textcircled{9}\ 10) \\ \pi_B = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10) \end{cases}$$



$$\begin{cases} \pi_A = (1\ 2\ \textcircled{9}\ 4\ 5\ 6\ 7\ \textcircled{3}\ 10) \\ \pi_B = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10) \end{cases}$$

(a) Two element exchanges

$$\begin{cases} \pi_A = (1\ 2\ 3\ \textcircled{4}\ 5\ 6\ 7\ 8\ \textcircled{9}\ 10) \\ \pi_B = (1\ 2\ 3\ \textcircled{4}\ 5\ 6\ 7\ 8\ \textcircled{9}\ 10) \end{cases}$$



$$\begin{cases} \pi_A = (1\ 2\ 3\ \textcircled{9}\ 5\ 6\ 7\ 8\ \textcircled{4}\ 10) \\ \pi_B = (1\ 2\ 3\ \textcircled{9}\ 5\ 6\ 7\ 8\ \textcircled{4}\ 10) \end{cases}$$

(b) Four element exchange

$$\begin{cases} \pi_A = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10) \\ \pi_B = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9\ 10) \end{cases}$$



$$\begin{cases} \pi_A = (7\ 8\ 9\ 10\ 1\ 2\ 3\ 4\ 5\ 6) \\ \pi_B = (7\ 8\ 9\ 10\ 1\ 2\ 3\ 4\ 5\ 6) \end{cases}$$

(c) Block exchange

Fig. 4 Generating perturbed set of sequences

하학적인 구속조건을 위반하지 말아야 할 것이다. 순서의 최적화 과정에서 전체의 최적값을 구하기 위해서는 가능한 한 다양한 방법의 순서변화를 시도하여야 하는데 반하여 이와 같은 2요소 교환법은 순서변화를 시키는 양이 적으므로 상대적으로 수렴속도가 느린 편이다. 이를 개선하기 위하여 그림 4b, 4c에 보인 4요소 교환법과 묶음교환법을 이용하여 다양한 방식의 순서변화를 시도하였다.

4요소 교환법에서는  $n^A$  과  $n^B$  의 같은 위치에 있는 각각의 2요소들이 교환이 되므로 2요소 교환법을 2번 실시한 효과를 갖는다. 묶음교환법은 개별요소단위가 아닌 집단의 요소들이 로봇 A, B에 대하여 동시에 교환이 되므로 순서변화의 양이 큰 편이다. 모사풀림방법의 매 반복작업에서 순서 변화를 시도할 때 이와 같은 세 가지의 순서변화 방식을 일정 비율로써 번갈아 가면서 사용함으로

써 다양한 순서 변화를 시도할 수 있으며 따라서 수렴속도와 계산결과에 상당한 이득을 가져왔다.

2요소 교환법을 사용한 경우와 세 가지 방식의 혼용한 경우의 계산결과가 표 1에 나타나 있다. 이에 따르면, 같은 수의 반복계산을 했을 때 세 가지 방법을 혼용한 경우가 단순교환법의 경우보다 더 나은 결과를 가져옴을 알 수 있다. 이는 단순교환법만을 사용한다면 최적에 근사한 값에 도달하기 위해서는 훨씬 더 많은 수의 반복작업이 필요하다는 것을 의미한다. 이러한 결과는 '수렴속도를 향상시키기 위해서는 순서의 큰 변화를 시도하여야 한다'는 White의 결론과 일치한다.<sup>(6)</sup>

Table 1 Results comparing the perturbation methods (seconds)

trial	two-element exchange	mixed exchange
1	4.02	3.94
2	4.10	3.96
3	3.94	3.94
4	3.99	3.94
average	4.01	3.945

로봇 A, B가 한 개의 작업물에 각 10개씩 20개의 부품을 조립할 때의 경우를 살펴보자. 각 작업물은 컨베어를 따라 로봇의 작업공간에 들어서게 되고 로봇 A에 의하여 10개, 로봇 B에 의하여 10개의 부품을 수집하여 작업공간을 나서게 된다. 다른 작업물들이 그 앞뒤에서 연속적으로 이동하면서 같은 과정을 거쳐서 조립물이 완성된다. 이러한 연속작업에서 한 개의 작업물에 걸리는 평균 조립시간은 초기의 준비작업시간을 제외하면 로봇들이 20개의 부품을 모두 올려놓는 작업에 걸리는 시간과 같다.

그림 5는 각 단계별 조립시간  $T_i^A, T_i^B$  을 최적화된 경우와 임의의 조립순서에 대하여 보여주고 있다. 여기에서 최적의 조립시간에 대하여 조립순서는 유일한 답이 아니라 여러 개의 답을 가질 수 있다. 그 이유는 각 단계별 조립시간은 각 로봇의 작업시간 중에서 최대값으로 결정되고 각 단계별 최소값이 변하더라도 최대값보다 작다면 조립시간은 변하지 않기 때문이다. 또한, 그림 5b를 살펴보면 조립순서가 최적화 된 경우  $T_i^A, T_i^B$  는 서로 근사한 값을 갖도록 쌍을 이루고 있음을 알 수 있다. 따라서, 한 로봇이 각 단계별 작업을 끝낸 상태에서 동적 간섭현상을 줄이기 위하여 다른 로봇이 대기상태에 있는 시간이 줄어

들어 순수작업시간 이외의 허비된 시간이 최소화되었음을 알 수 있다 (그림 5c). 그 반면에 임의로 계획된 작업순서의 경우에는  $T_i^A$ ,  $T_i^B$  가 조화로운 쌍을 이루지 못하여 대기상태에 소비된 시간이 크고 로봇의 이동순서가 최적화되지 못하여 작업물 하나 당 전체작업시간이 5.21초로 최적화된 경우의 3.94초에 비하여 32%가 증가하였다.

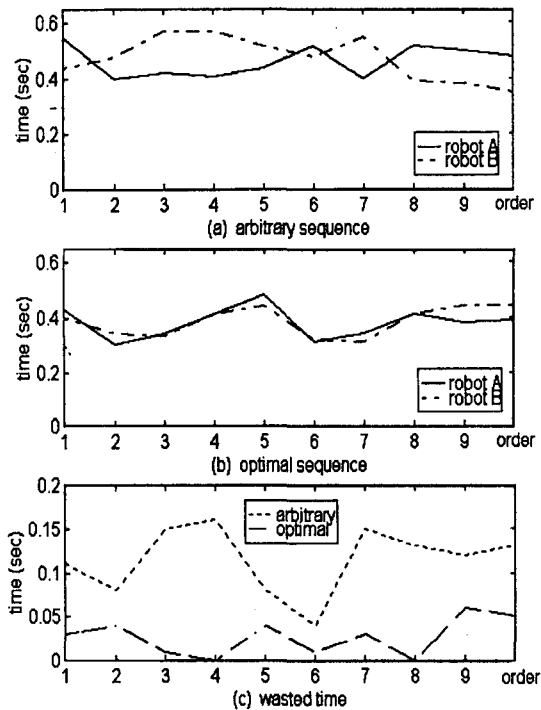


Fig. 5 Assembly time for each step

### 7. 결론

본 논문에서는 두 대의 로봇과 컨베이어 라인으로 구성된 조립시스템의 작업순서 최적화에 대한 연구를 수행하였다. 본 조립시스템의 특징은 조립품을 이송시키는 컨베이어가 연속적으로 움직이므로, 로봇의 조립작업을 위하여 컨베이어가 일정 위치에 정지 후 다시 다음 작업을 위하여 이동을 하는 불필요한 가속 및 감속 운동을 피할 수 있다는 장점이 있다. 그러나, 두 로봇간의 간섭현상을 피하기 위하여 적절한 작업계획이 필요하다.

이를 위하여 로봇간의 동적, 기구학적인 간섭현상을 고려하여 작업순서가 계획되었고 최소시간동안에 작업을 수행할 수 있도록 순서의 최적화 방안을 제시하였다. 그

방법으로 모사플립 기법을 이용하였고 계산의 효율성 제고를 위한 새로운 순서의 조합방법이 제시되었다.

### 참고 문헌

1. Li, S.-H., Fujiwara, N., and Asada, H., "An Ultrahigh Speed Assembly Robot System: Part II Design", *Proc. Japan-USA Symposium on Flexible Automation*, 1994.
2. Lee, B.H. and Lee, C.S.G., "Collision-Free Motion Planning of Two Robots," *IEEE Trans. Syst. Man Cybern.*, vol.SMC-17, no.1, pp.21-32, 1987.
3. Shin, K. G. and Quin, Z., "Minimum-Time Collision Free Trajectory Planning for Dual-Robot Systems," *IEEE Transactions on Robotics and Automation*, vol.8, no.5, pp.641-644, Oct. 1992.
4. Lawler, E. L., et. al, *The Traveling Salesman Problem*, John Wiley and Sons, 1985.
5. Kirkpatrick, S., Gelatt, Jr., C. D., and Vecchi, M. P., "Optimization by Simulated Annealing," *Science*, Vol.220, No.4598, pp.671-680, 1983.
6. Laarhoven, P. and Aarts, E., *Simulated Annealing: Theory and Applications*, 1987.
7. Lin, S. and Kernighan, B.W., "An Effective Heuristic Algorithm for the Traveling Salesman Problem," *Operations Research*, vol.21, pp.498-516, 1973.
8. White, S.R., "Concepts of Scale in Simulated Annealing," *Proc. IEEE Int. Conf. on Computer Design*, pp.646-651, 1984.
9. Peyrol, E., Floquet, P., Pibouleau, L. and Domenech, S., "Scheduling and Simulated Annealing Application to a Semiconductor Circuit Fabrication Plant," *Computer and Chemical Engineering*, vol.17, pp.539-544, 1993.
10. Salomon, M., Kuik, R. and Van Wassenhove, L., "Statistical Search Methods for Lotsizing Problems," *Annals of Operations Research*, vol.41, pp.453-68, May 1993.