

표면실장기의 조립시간 최소화를 위한 진화 알고리즘

An Evolutionary Algorithm for Minimizing the Assembly Time of Surface Mounting Machines

이 성 한, 이 영 대, 이 원 식, 이 범 희

(Sung Han Lee, Young Dae Lee, Won Sik Lee, and Beom Hee Lee)

Abstract · This paper considers the problem of improving the productivity of surface mounting machines in the printed circuit board(PCB) assembly line. This problem is generally divided into two problems: reel assignment problem and pick-and-place sequencing problem, which are known to have no polynomial time algorithms. In the last ten years, algorithm designers have been trying to solve them separately. However, they need to be solved jointly because they are highly interrelated. This paper proposes an evolutionary algorithm which can consider the two problems jointly and thus yield a better solution. In order to evaluate the proposed algorithm, computer simulation is performed on real-life surface mounting machines. The proposed algorithm is expected to reduce the assembly time of surface mounting machines and thus improve the productivity.

Keywords : productivity, surface mounting machines, printed circuit board, evolutionary algorithm

I. 서론

과거의 인쇄회로기판(PCB: Printed Circuit Board)은 부품을 해당 구멍에 삽입한 후 납땀하여 조립하는 방법을 사용하였다. 그러나 부품 제조기술이 발달함에 따라 부품의 소형화, 경량화가 이루어져 과거의 삽입방법 대신 표면실장하는 방식이 보편화되고 있다. 인쇄회로기판에 실장되는 부품을 표면실장부품이라 하고 표면실장부품을 인쇄회로기판에 탑재하는 기계를 표면실장기(surface mounting machine)라 한다. 표면실장부품의 제작에서부터 인쇄회로기판에 조립하는 과정에 이르기까지의 제반 관련기술들을 통칭하여 표면실장기술[1]이라 일컬으며 전자제품의 소형화, 경량화, 고신뢰성화, 다기능화가 추구됨에 따라 표면실장기술의 개발 및 보급이 한층 가속화되고 있다.

표면실장기를 제작하는 생산자들은 표면실장기의 생산효율을 높이기 위해 기구적인 면을 개선하는 연구를 해왔다. 최근 들어 생산효율이 높은 고속기계를 개발하는 것도 물론 중요하지만 표면실장기를 보다 효율적으로 사용하게 하는 운용 알고리즘도 중요한 것으로 생산자들은 깨닫게 되어 현재 많은 연구가 진행되고 있다. 관련된 최적화 문제로는 흡장착 순서 문제와 릴 할당 문제가 있다. 흡장착 순서 문제는 조립할 부품들의 작업 순서를 결정하는 문제로서 부품의 공급위치는 고정되어 있다고 가정하며 표면실장기의 조립시간을 최소화하는 것이 목적이다. 릴 할당 문제는 조립할 부품의 공급위치를 결정하는 문제로서 궁극적인 목적은 흡장착 순서 문제의 최적해를 구하기 위한 환경을 제공함에 있다. 이 두 문제는 조합 최적화 문제(combinatorial optimization problem)로서 NP-하드(hard)라고 알려져 있다[2]. 최근까지의 연구는 이들 문제의 복잡

도로 인하여 각 문제를 단독으로 고려하여 해를 구한 다음 이들 해를 조합하는 방법으로 전체 문제의 해로 제시하고 있다. 그러나 위의 문제들은 서로의 해에 영향을 주는 밀접한 관계에 있기 때문에 PCB 조립라인의 조립시간을 최소화하기 위해서는 궁극적으로 이 두 문제를 동시에 풀어야 한다. 본 논문에서는 진화 알고리즘(evolutionary algorithm)을 사용하여 흡장착 순서 및 릴 할당 문제를 동시에 푸는 방법을 제안한다.

II. 표면실장기의 구조

본 논문에서 대상으로 하는 표면실장기의 개략도는 그림 1과 같다. PCB는 컨베이어 벨트에 의해서 표면실장기에 공급되며 조립을 하는 동안은 고정된다. 부품은 릴에 의해서 공급되며 각각의 릴은 한 종류의 부품만을 공급할 수 있다. 컨베이어 벨트의 양쪽으로는 두 개의 피더 선반이 고정되어 있으며 피더 선반은 릴을 설치할 수 있도록 다수의 슬롯을 가지고 있다. 부품을 조립하는 장치인 헤드에는 공압에 의해서 부품을 흡장착하는 노즐이 달려 있다.

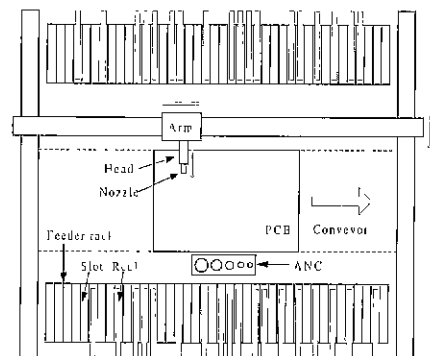


그림 1. 표면실장기의 개략도.

Fig. 1. Schematic diagram of a surface mounting machine.

접수일자 : 1999. 8. 24., 수정완료 : 2000. 2. 16

이성한 : 삼성전자 통신연구소

이영대 : 세명대학교 제어계측공학과

이원식, 이범희 : 서울대학교 전기공학부

노즐은 부품의 크기에 따라서 다른 종류를 사용해야 한다. 따라서 작업도중 노즐을 교환해야 할 필요가 발생하는데 이런 경우를 위하여 노즐을 자동으로 교환하여 주는 장치인 자동 노즐 교환기(Auto Nozzle Changer, ANC)가 있다.

표면실장기의 작업 최적화 문제는 헤드의 x/y-축 이동 시간을 최소화하는 문제로 인식된다. 헤드의 x/y-축 이동 시간은 릴의 공급위치와 헤드의 이동경로에 의해서 결정되는데 릴의 공급위치를 결정하는 문제를 릴 할당 문제, 헤드의 이동경로를 결정하는 문제를 흡장착 순서 문제라 일컫는다. Leu(1993)는 적응 유전자 알고리즘(adaptive genetic algorithm)을 사용하여 릴 할당 문제의 해를 제시하였고 또한 흡장착 순서 문제에도 적용하여 해를 구하였다[3]. Kumar(1995)는 릴 할당 문제를 최소 가중 부합 문제(minimum weight matching problem)로 모델링하여 해를 구하였고 흡장착 순서 문제는 순회판매원문제(TSP: Traveling Salesman Problem)로 모델링하여 해를 제시하였다[4]. 이들 알고리즘들은 릴 할당 문제와 흡장착 순서 문제를 서로 독립된 문제로 분리하여 각각의 해를 구한 다음 이를 조합함으로써 표면실장기의 작업 최적화 문제의 해로 제시하였다. 그러나 실제로는 릴 할당에 따라서 흡장착 순서가 달라져야 하며 또한 흡장착 순서에 따라서 릴 할당이 달라져야 한다. 즉 릴 할당 문제와 흡장착 순서 문제를 분리하여 해를 구하는 방법은 성능면에서 비효율적일 수 밖에 없다[5]. 본 논문에서는 진화 알고리즘을 사용하여 두 문제를 동시에 고려하는 방법을 제시함으로써 보다 효율적인 해를 구하고자 한다. 제안된 방법의 성능을 검증하기 위해서 릴 할당 문제와 흡장착 순서 문제를 각각 분리하여 진화 알고리즘을 적용한 결과와 비교한다.

III. 진화적 접근법

1. 진화 알고리즘의 개요

최근 진화 알고리즘이 최적화 문제의 전역해나 근사해를 구하는 방법으로 각광을 받고 있다. 진화 알고리즘은 자연의 진화 현상을 모방한 확률적인 최적화 기법으로 휴리스틱한 기법의 일종이라고 할 수 있다. 진화 알고리즘은 풀고자하는 문제에 대한 정보가 별로 없는 경우에 대하여 강인하고 전역적인 탐색 방법으로서 여러 가지 문제에 대하여 적용이 가능하며, 주어진 상황에 적용적으로 대처하여 탐색을 행하는 특징을 가지고 있다[6]. 현재 진화 알고리즘은 복잡한 문제를 푸는 데 있어서 그 가능성을 주목받고 있으며 일정 계획이나 신뢰성 설계, 기계 최적 설계, TSP, 차량 배치, 그룹 기법, 공장 설계 및 수송 문제 등에 광범위하게 적용되고 있다[7].

진화 알고리즘에서 각각의 해(개체)는 목적함수의 적합도에 따라 평가된다. 개체 집단에 적용되는 선정, 교배 및 돌연변이와 같은 진화 연산자는 기본적으로 다음과 같은 역할을 수행한다. 선정은 다음 세대에 각 개체가 더 좋은 적합도를 가지게 함으로써 탐색점들을 더 좋은 탐색영역으로 유도한다. 교배 및 돌연변이는 탐색공간에 새로운 점들을 생성시킨다. 교배 연산은 임의의 두 개체의 정보를 혼합하는 방식으로 각 세대에 걸쳐 부모개체 집단의

정보를 자손개체 집단에 전달한다. 돌연변이 연산은 일종의 노이즈를 탐색점들에 가해 이들을 새로운 위치로 이동시키는 역할을 한다. 진화 알고리즘의 초기화 과정에서 개체집단은 임의로 초기화되고, 반복적인 세대교체 과정이 도입된다. 먼저 각 개체들의 적합도가 목적함수를 이용하여 계산된 후, 선정연산이 개체집단에 대해 행해진다. 다음에 교배 및 돌연변이 연산이 현재의 개체집단에 대해 적용되면, 새로운 점들이 생성되어 다음 세대의 개체집단을 형성하게 된다. 결과적으로 충분한 세대교체 후에 개체집단의 최우수 개체는 문제의 최적해나 준최적해를 표현하게 된다.

2. 부분링크 기법

2.1 링크 정의

진화 알고리즘의 구성부는 유전인자이다. 일련의 유전인자, 즉 염색체를 링크라 일컫는다. 링크를 두 개의 부분링크로 나눈다. 앞쪽에 위치하는 부분링크 A는 릴 할당을 위한 것이고 뒤쪽에 위치하는 부분링크 B는 흡장착 순서를 결정하기 위한 것이다. 부분링크 A를 다음과 같이 정의한다.

$$\vec{g} = [g(1), \dots, g(L)] \quad (1)$$

$$\text{where } g(i) \in L, \quad \forall i \in \{1, \dots, |L|\} \quad (2)$$

$$g(i) \neq g(j), \quad \text{if } i \neq j, \forall i, j \in \{1, \dots, |L|\} \quad (3)$$

유전인자 $g(i)$ 는 릴 $i \in R$ 가 할당되는 슬롯번호를 나타낸다. 예를 들면, 부분링크 $[5, 7, \dots]$ 은 1번 릴을 5번 슬롯에 할당하고 2번 릴을 7번 슬롯에 할당함을 표시한다. 따라서 앞부분에 있는 $|R|$ 개의 유전인자만이 릴 할당에 직접적으로 관여하게 되고 나머지 $|L| - |R|$ 개의 유전인자는 사실상 릴 할당에는 아무런 영향을 미치지 않는다. 두 번째 부분링크 B는 다음과 같이 정의한다.

$$\vec{g}^* = [g^*(1), \dots, g^*(|M|)] \quad (4)$$

$$\text{where } g^*(i) \in M, \quad \forall i \in \{1, \dots, |M|\} \quad (5)$$

$$g^*(i) \neq g^*(j), \quad \text{if } i \neq j, \forall i, j \in \{1, \dots, |M|\} \quad (6)$$

유전인자 $g^*(i)$ 는 작업 순서상 1-번째에 장착될 부품을 나타낸다. 예를 들면, 링크 $[8, 3, \dots]$ 은 8번 부품을 맨 처음으로 조립한 후 3번 부품을 두 번째로 조립함을 표시한다. 부분링크 A와 부분링크 B를 그림 2와 같이 연결함으로써 완성된 링크를 구성할 수 있다. 최종 링크는 총 $|L| + |M|$ 개의 유전인자를 가지게 된다.

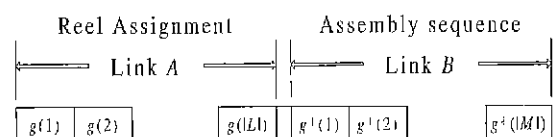


그림 2. 부분링크를 갖는 링크

Fig. 2. Link consisting of two partial links.

2.2 부분링크 연산자

일반적인 교배 및 돌연변이 연산자를 부분링크를 갖는

링크에 적용했을 경우 무효한 링크를 생산할 수 있는 위험이 있다. 따라서 부분링크를 갖는 링크에 적용할 수 있는 교배 및 돌연변이 연산자를 본 절에서 정의하며 앞으로의 진화 과정에서는 이 두 부분링크 연산자를 사용하기로 한다. 교배 연산자를 그림 3과 같이 정의한다. 우선 각 부분링크내에서 임의적으로 두 위치를 선택한다. 한 부모개체로부터 선택된 두 위치 사이에 있는 유전인자를 그대로 자손 개체에 유전시킨다. 다음으로 다른 부모개체로부터 현재 사용되지 않은 유전인자를 순서대로 추출하여 자손 개체의 정의되지 않은 부분에 복제한다. 부분링크를 갖는 링크의 돌연변이 연산자로서 교환 (exchange), 반전 (inversion) 및 회전 (rotation) 연산자를 정의한다. 교환 연산자는 각 부분링크내에서 임의로 두 유전인자를 선택하여 값을 교환하는 역할을 한다. 반전 연산자는 각 부분링크내에서 임의로 두 위치를 선택한 다음 두 위치 사이의 유전인자 값을 역순으로 재정의하는 역할을 한다. 회전 연산자는 각 부분링크내에서 임의로 두 위치를 선택한 다음 두 위치 사이의 유전인자 값을 왼쪽이나 오른쪽으로 이동시키는 역할을 한다. 돌연변이 연산자 중 교환 연산자의 수행결과를 보이면 그림 4와 같다.

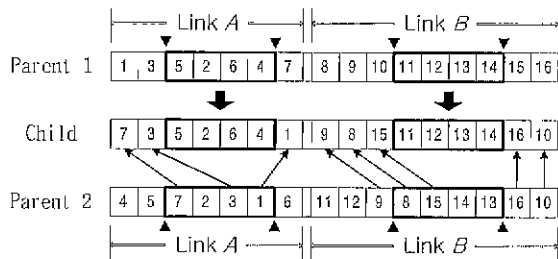


그림 3. 부분링크 교배 연산자.
Fig. 3. Partial link crossover operator.

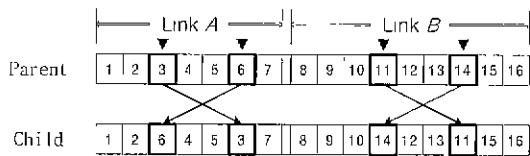


그림 4. 부분링크 교환 연산자.
Fig. 4. Partial link exchange operator.

23 적합도 함수 정의

적합도 함수는 PCB의 조립시간으로 정의한다. 조립작업을 시작하기 전에 처음으로 조립할 부품의 노즐을 미리 조립헤드에 설치해 두며 해당 슬롯 위에 조립헤드를 위치시켜 둔다. 따라서 조립작업의 전체 경로는 슬롯에서 시작해서 PCB상의 마지막 장착점에서 끝나게 되며 노즐 교환은 PCB에서 슬롯으로 이동할 때 이뤄진다. 이와 같은 내용을 고려하여 링크 $[\tilde{g}, \tilde{g}^*]$ 에 해당하는 적합도 함수 $F(\tilde{g}, \tilde{g}^*)$ 를 수식으로 표현하면 다음과 같다.

$$F(\tilde{g}, \tilde{g}^*) = \sum_{m=1}^{M} [T(g(b(g^*(m))), g^*(m)) + F_1(\tilde{g}, \tilde{g}^*, m)], \quad (7)$$

$$F_1(\tilde{g}, \tilde{g}^*, m) = \begin{cases} 0 & \text{if } m = |M|, \\ T(g^*(m), g(b(g^*(m+1)))) & \text{if } e(g^*(m)) = e(g^*(m+1)), \\ T(g^*(m), ANC) + T_{anc} & + T(ANC, g(b(g^*(m+1)))) \\ & \text{otherwise.} \end{cases} \quad (8)$$

$e(m) \in E$ 는 $m \in M$ 에 사용되는 노즐을 나타내며 T_{anc} 는 노즐 교환 소요 시간을 나타낸다. (7)의 첫 번째 항 $T(g(b(g^*(m))), g^*(m))$ 은 조립순서상 m -번째 조립될 부품이 할당된 슬롯에서 해당 장착점까지의 이동 시간을 나타내며 두 번째 항 $F_1(\tilde{g}, \tilde{g}^*, m)$ 은 m -번째 조립되는 부품의 장착점에서 $(m+1)$ -번째 조립될 부품이 할당된 슬롯까지 이동하는 데 소요되는 시간을 나타낸다. 두 번째 항은 경우에 따라서는 노즐 교환 시간을 포함하기도 한다. 2.4 매개 변수

진화 알고리즘의 적용에 있어서 알고리즘의 성능에 영향을 미치는 변수는 다양하다. 우선 링크와 적합도 함수를 정의하는 방법에 따라서 성능이 변할 수 있다 또한 개체집단의 크기, 세대수, 교배 확률, 돌연변이 확률 등의 매개 변수 값에 따라서도 알고리즘의 성능은 크게 변한다. 다양한 실험을 통하여 구한 매개 변수의 추천 값은 표 1과 같다.

표 1. 매개 변수의 추천값.
Table 1. Recommended values of parameters.

매개 변수	추천값
개체집단 크기	200
세대수	1000
교배 확률	0.5
돌연변이 확률	0.5

2.5 적용 예

알고리즘의 실행 결과를 확인하기 위해서 예제를 다룬다. 4개의 릴이 사용되며 11개의 부품을 가진 PCB를 고려한다. 표 2는 각 릴에 사용되는 노즐 번호와 릴이 공급할 부품 개수를 보여 준다. 그림 5는 각 부품과 슬롯의 위치를 도식적으로 표현한 것인데 편의상 각 부품과 슬롯에 일련번호를 정의하였다.

표 2 릴에 따른 사용 노즐 및 부품 개수.

Table 2. Numbers of nozzles and components for each reel.

릴	a	b	c	d
사용 노즐 번호	1	2	3	4
부품 개수	5	3	2	1

진화 알고리즘을 적용했을 때 세대가 진행됨에 따른 적합도 함수 값의 변화는 그림 6과 같다. 초기 세대에는 적합도 함수 값이 빠르게 감소하다가 세대가 지날수록 일정한 값에 수렴함을 볼 수 있다. 적합도 함수의 수렴 값은 초기

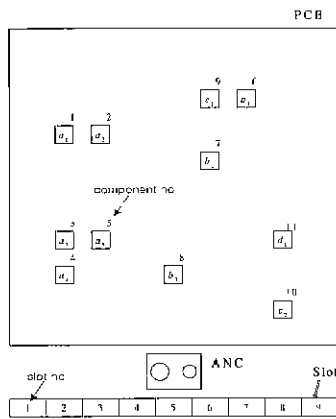


그림 5. 적용 예.
Fig. 5. Example.

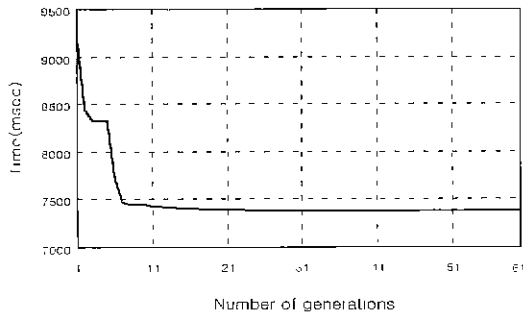


그림 6. 세대에 따른 적합도 함수 값의 변화.
Fig. 6. Fitness value according to generations.

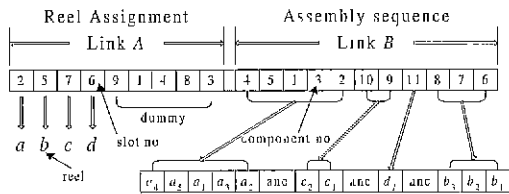


그림 7. 최종해에 해당되는 링크.
Fig. 7. Solution link.

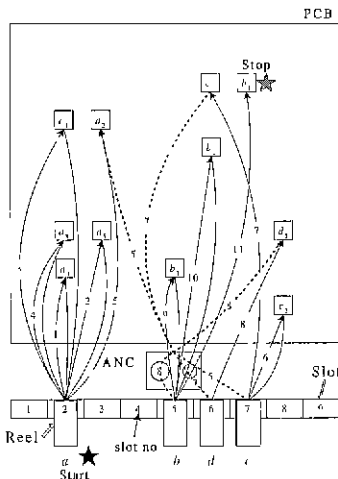


그림 8. 릴 할당 및 흡장착 순서.
Fig. 8. Reel assignment and pick-and-place sequence.

값에 비하여 14%가 감소했으며 그 때의 해당 링크는 그림 7과 같다. 그림 7의 링크 정보를 이용하여 릴 할당 및 흡장착 순서를 나타내면 그림 8과 같다. 각 릴은 장착점의 위치에 가깝게 배치가 되었으며 노즐 교환 회수가 최소가 되게끔 흡장착 순서가 결정되었음을 확인할 수 있다.

IV. 모의실험

제안된 알고리즘의 성능 평가를 위하여 실제 PCB 조립 라인에서 사용되고 있는 표면실장기를 대상으로 모의 실험을 수행한다. 대상으로 삼은 표면실장기는 1개의 헤드와 100개의 슬롯을 가지고 있으며 헤드의 x/y-축 방향 속도는 1.5 m/s이다. 한 개의 부품을 흡착 또는 장착하는데 소요되는 시간은 0.15초이며 노즐을 교환하는데 필요한 시간은 0.85초이다.

알고리즘의 성능을 객관적으로 평가하기 위해서는 많은 종류의 대상 작업이 필요하게 된다. 본 절에서는 난수 발생기(random number generation program)를 사용하여 만든 다수의 가상 PCB를 대상으로 모의 실험을 수행한다. 가상 PCB의 크기는 가로 55cm, 세로 40cm로 제한되고 장착점의 위치가 난수 발생기에 의해서 만들어진다. 가상 PCB의 일련번호(serial No.)에 따른 릴과 부품개수를 그래프로 나타내면 그림 9와 같다. 릴과 부품개수 모두 PCB의 일련번호가 커짐에 따라서 선형적으로 증가함을 볼 수 있다. 이것은 PCB의 일련번호를 문제의 복잡도로 간주해도 무방함을 의미한다.

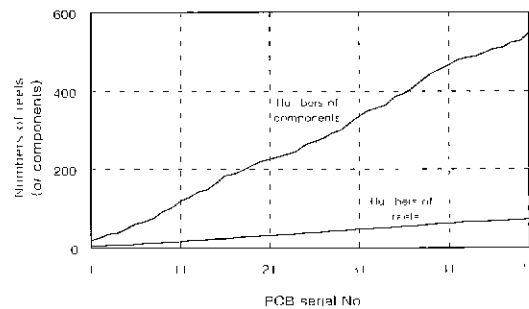


그림 9. 실험대상 PCB에 따른 릴과 부품개수.
Fig. 9. Numbers of reels and components according to board serial no

본 논문이 제안한 알고리즘의 성능 검증을 위해서 릴 할당 문제와 흡장착 순서 문제를 분리하여 해를 구하는 두 가지 경우와 비교한다. 우선 실제 생산라인에서 보편적으로 사용되고 있는 휴리스틱 알고리즘과 비교한다. 휴리스틱 알고리즘은 그리디(greedy) 알고리즘에 의해서 릴을 할당한 다음, TSP를 풀기 위한 대표적인 휴리스틱 기법인 최단점 TSP 기법을 사용하여 흡장착 순서를 결정한다. 두 번째 경우로는 진화 알고리즘을 릴 할당 문제와 흡장착 순서 문제에 각각 적용하여 해를 조합하는 방법을 비교 대상으로 삼는다. 그림 10은 대상 PCB에 따른 각 알고리즘의 실험 결과를 나타낸다. 서술의 편의를 위해서 휴리스틱 알고리즘을 HA(Heuristic Algorithm), 본 논문이 제안한 방법을 IEA(Integrated Evolutionary Algorithm), 그리고 진

화 알고리즘을 릴 할당 문제와 흡장착 순서 문제에 각각 적용하는 방법을 SEA(Separate Evolutionary Algorithm)라고 명명한다 그림 10에 의하면 실험한 모든 PCB에 대해서 IEA가 SEA와 HA보다 성능이 우월함을 확인할 수 있었다. HA 및 SEA에 대한 IEA의 효율을 보다 쉽게 보여 주기 위해서 그림 11에 작업 시간 감소율을 나타내었다. HA에 대한 IEA의 작업시간 감소율은 평균 1.8%, SEA에 대한 IEA의 작업시간 감소율은 평균 1.3%이었으며 PCB의 일련번호가 증가할수록, 즉 PCB에 사용되는 릴과 부품 개수가 증가할수록 성능이 향상되었다. 그러나 릴과 부품 개수가 모의 실험된 범위를 벗어나는 경우에도 IEA의 효율이 계속 향상된다는 보장은 할 수 없다. 다만 모의 실험에 사용된 PCB의 범위가 본 논문이 대상으로 하는 표면실장기가 실제 생산 현장에서 조립하는 PCB중 가장 복잡한 경우도 포함하고 있기 때문에 현실적으로는 IEA의 성능이 PCB의 복잡도가 증가할수록 향상된다고 말할 수 있다.

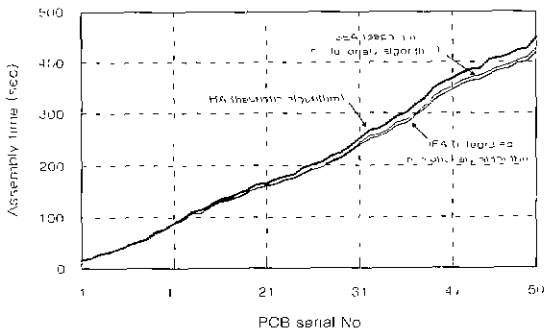


그림 10 알고리즘에 따른 조립시간.
Fig. 10. Assembly time according to algorithm.

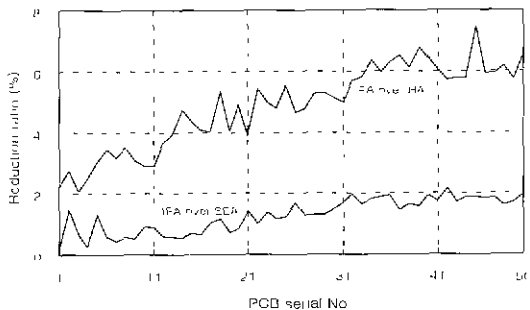


그림 11 조립시간 감소율.
Fig. 11. Reduction ratio of assembly time.

V. 결론

본 논문에서는 표면실장기의 작업시간을 최소화하기 위한 알고리즘을 제안하였다. 표면실장기의 작업시간 최소화 문제는 릴 할당 문제와 흡장착 순서 문제로 구분할 수 있는데 문제의 복잡도로 인하여 이들 두 문제를 별개의 문제로 분리하여 각각의 해를 구한 다음 다시 조합하는 방법이 사용되어 왔다. 그러나 실제로는 릴 할당에 따라서 흡장착 순서가 달라져야 하며 또한 흡장착 순서에 따라서 릴 할당이 달라져야 하기 때문에 릴 할당 문제와 흡장착 순서 분

제를 분리하여 해를 구하는 방법은 성능면에서 비효율적일 수밖에 없다. 본 논문에서는 진화 알고리즘을 사용하여 두 문제를 동시에 고려하는 새로운 방법을 제안하였다. 제안된 알고리즘의 성능 평가를 위해서 실제 PCB 조립라인에서 사용되고 있는 표면실장기를 대상으로 모의 실험을 수행하였다. 비교 평가를 위해서 릴 할당 문제와 흡장착 순서 문제를 각각 휴리스틱하게 푼 경우와 각각의 문제에 진화 알고리즘을 적용한 경우의 실험 결과와 비교하였다. 실험된 50종의 PCB 모든 경우에 있어서 제안된 알고리즘이 우월함을 확인할 수 있었으며 휴리스틱 알고리즘에 비해서 4.8%, 진화 알고리즘을 각각의 문제에 적용한 경우에 비해서는 1.3%의 작업시간을 감소시킬 수 있었다. 또한 실험된 PCB의 릴과 부품개수가 많은 경우에도 일정 수준 이상의 성능을 보였으며 이는 PCB의 복잡도에 상관없이 성능을 유지함을 의미한다. 본 논문에서 제안한 표면실장기의 작업시간을 최소화하기 위한 알고리즘은 PCB 조립시간을 줄임으로써 생산율을 향상시킬 것으로 기대된다.

한편 표면실장기의 작업시간을 기구적으로 줄이기 위한 방법으로 다수의 헤드를 사용하는 경우가 있다. 다수의 헤드를 사용함으로써 한번의 수직 운동으로 여러 개의 부품을 동시에 흡착할 수 있기 때문에 흡착 소요 시간을 획기적으로 줄일 수 있게 된다. 따라서 이 경우에는 이동 거리의 최소화에 초점을 맞춘 본 논문의 접근 방향과는 달리 동시 흡착 회수의 최대화에 중점을 두어야 한다. 이에 기초한 다수 헤드에 대한 알고리즘 개발을 추후 연구 과제로 제시한다.

참고문헌

- [1] 이성환, 홍지민, 김대원, 이범희, "표면실장기계를 위한 조립작업의 효율향상기법", 한국자동제어 학술회의 논문집, pp. 112-115, 1995. 10.
- [2] P. V. Laarhoven and W. Zijm, "Production preparation and numerical control in PCB assembly", *International Journal of Flexible Manufacturing Systems*, vol. 5, no. 3, pp. 187-207, 1993.
- [3] H. Wong and M.C. Leu, "Adaptive genetic algorithm for optimal printed circuit board assembly planning", *Annals of the CIRP*, vol. 42, no. 1, pp. 17-20, 1993.
- [4] R. Kumar and H. Li, "Integer programming approach to printed circuit board assembly time optimization", *Trans. on Components, Packaging and Manufacturing Technology. Part B : Advanced Packaging*, vol. 18, no. 4, pp. 720-727, 1995.
- [5] 이성환, "켄트리형 표면실장기에 의한 PCB 조립작업의 최적화에 관한 연구", 공학박사학위논문, 서울대학교 대학원, 1999. 8.
- [6] D. Fogel, "An introduction to simulated evolutionary optimization", *IEEE trans. on Neural Network*, vol. 5, pp. 3-14, 1994.
- [7] 이영대, "로봇 매니플레이터의 최소시간 동작계획을 위한 진화연산적 접근", 공학박사학위논문, 서울대학교 대학원, 1998. 2.

기호 설명

$N \equiv \{1, \dots, n\}$ 1부터 n 까지의 정수 집합
 $C \equiv \{c_1, \dots, c_n\}$ 유전인자의 집합($i \neq j$ 일 때 $c_i \neq c_j$)
 $\tilde{g} \equiv [g(1), \dots, g(n)]$ 유전인자로 구성된 링크($g(i)$ 는 N 로부터 C 로의 함수이며 $i \neq j$ 일 때 $g(i) \neq g(j)$ 를 만족)
 $G \equiv \{ \tilde{g}_1, \dots, \tilde{g}_w \}$ 가능한 모든 링크의 집합(w 는 가능한 링크의 총 개수)
 F G 로부터 실수집합으로의 적함도 함수

$|S|$ 집합 S 의 원소 개수
 $L \equiv \{1, \dots, |L|\}$ 슬롯의 집합
 $R \equiv \{1, \dots, |R|\}$ 조립에 사용될 밀의 집합
 $M \equiv \{1, \dots, |M|\}$ 조립될 부품(장착점)의 집합
 $E \equiv \{1, \dots, |E|\}$ 조립에 사용될 노즐의 집합
 $T(b_1, b_2)$ b_1 위치에서 b_2 위치까지의 이동 소요 시간
 $e(m) \in E$ $m \in M$ 에 사용되는 노즐
 T_{anc} 노즐 교환 소요 시간



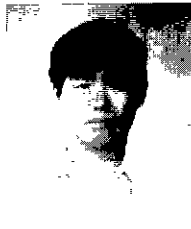
이 성 한

1992년 서울대 제어계측공학과 졸업. 동대학원 석사(1994), 동대학원 박사(1999). 1999년~현재 삼성전자 통신연구소 선임연구원. 관심분야는 인공지능, 공장자동화, 무선이동통신.



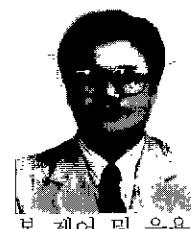
이 영 대

1985년 서울대 기계설계학과 졸업. 동대학원 석사(1987), 동대학원 제어계측공학과 박사(1998년). 1987년~1991년 대우중공업 중앙연구소. 1989년~1991년 한국생산성본부 공장자동화사업부. 1998년~1999년 한국과학기술연구원. 1999년~현재 세명대학교 전기전자공학부 교수. 관심분야는 인공지능 및 공장자동화.



이 원 식

1987년 서울대 제어계측공학과 졸업. 동대학원 석사(1989), 동대학 박사수료(1991). 1991년~1996년 대우중공업 중앙연구소. 1996년~현재 동대학원 전기공학부 박사과정. 관심분야는 인공지능, 공장자동화, RTOS.



이 범 희

1978년 서울대 전자공학과 졸업. 동대학원 석사(1980). Univ. of Michigan. Computer, Information & Control Eng. 박사(1985). 1987년~현재 서울대학교 전기공학부 교수. 관심분야는 로봇 제어 및 응용, 지능시스템.