

유연 PCB 자동삽입라인의 부하 평준화를 위한 작업흐름선택모델

Job Route Selection Model for Line Balancing of Flexible PCB Auto-insertion Line

함호상*, 김영휘**, 정연구*

Ho-Sang Ham*, Young-Hui Kim** and Yun-Koo Chung*

Abstract

We have described the optimal process route selection model for the PCB(printed circuit board) auto-insertion line. This PCB assembly line is known as a FFL(flexible flow line) which produces a range of products keeping the flow shop properties. Under FFL environments, we have emphasized the balancing of work-loads in order to maximize total productivity of PCB auto-insertion line. So we have developed a heuristic algorithm based on a work-order selection rule and min-max concept for the job route selection model.

1. 서론

80년대 초부터 전자조립업체들은 인력난을 극복하기 위하여 생산활동의 대부분을 차지하고 있는 인쇄회로기판(PCB : Printed Circuit Board)의 부품삽입작업을 자동화설비로 대체하였다. 이와같은 자동화 추세는 삽입부품의 크기가 점점 소형화됨에 따라 더욱

불가피하리라 예상된다. 따라서 Board에 전자부품을 자동으로 삽입하는 자동삽입기에 대한 투자는 향후 지속적으로 확대될 것이 틀림없기 때문에, 이에대한 효율적인 운영은 전자조립산업의 생산성향상과 직결될 것이다.

현재 국내 전자조립산업의 PCB 자동삽입기에 대한 운영실태를 살펴보면, 자동삽입공

* 한국과학기술연구원, 시스템공학연구소

** 고려대학교, 산업공학과

정은 그림 1과 같이 Sequencing 공정, Axial 공정, Radial 공정, Robot 공정, SMT(Surface Mounting Technology) 공정으로 구성되며, 각각의 Board들은 이들 공정을 순차적으로 거치면서 부품이 삽입되어 완성된다. 한편 각각의 공정은 하나 이상의 동일한 설비로 구성되고, 공정간 제품의 이동은 Magazine이라고 불리는 이동장비에 의하여 이송된다. 또한 공정과 공정사이에는 제품을 보관할 수 있는 저장 Buffer가 존재하는데, 이 곳에서 공정간 생산 불균형이 부분적으로 흡수된다. 그러나 저장 Buffer가 가득(Full)차는 경우 전(前)공정의 작업이 일시적으로 중단되는데 이를 Blocking이라 칭하고, 반대로 저장 Buffer의 재공품이 비어있어 후(後)공정에서 생산할 작업물을 공급받지 못하여 작업이 중단되는 경우 이를 Starvation이라 한다.

어야 할 최우선의 과제이다. 특히 최근 고객의 욕구를 만족시키기 위한 방안으로 제품의 수명이 크게 단축됨에 따라 제품종류의 다양화, 주문의 소ロット화 및 단납기화가 초래되었다. 이로 인하여 작업교체가 빈번히 발생하므로 NC프로그램 변경, 부품의 Sequencing을 위한 부품의 재배열등과 같은 부수적인 준비작업이 증가되는 결과를 가져왔다. 또한, 부적합한 제품조합으로 인하여 자삽기 공정간의 부하불균형이 발생하고, 이로인한 Blocking 및 Starvation의 발생으로 자삽기의 Idle시간이 증대되고, 공정내 재공물량의 정체가 심화되어 결국에는 생산성저하를 유발시키고 있다. 그 결과 불필요한 자삽기의 설비투자를 발생시키기 때문에 기업은 과잉 시설투자를 하게되므로 원가부담 및 자금압박을 받게 된다.

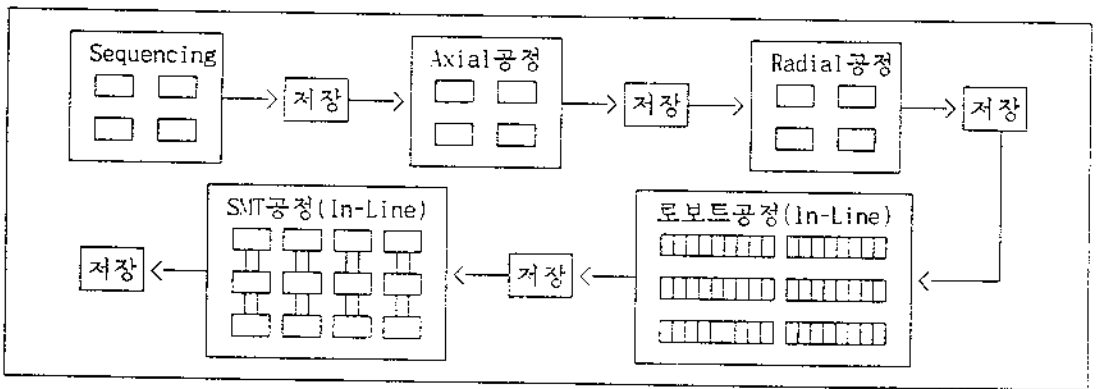


그림 1. PCB 자동삽입공정의 공정흐름도

이처럼 PCB 자동삽입공정은 여러 공정을 흘러 조립되는 관계로, 공정내 작업물량 및 공정간 부하에 대한 관리수준에 따라 기업의 생산성은 크게 좌우된다. 이와같은 이유 때문에 자동삽입기의 효율적인 운영은 기업이 풀

이러한 이유때문에 90년대 들어 PCB 조립 공정을 대상으로 생산시스템의 성능향상을 위한 연구가 많이 발표되고 있다. 일례로 Johri는 PCB조립라인에 대한 최적 설계를 위하여, 필요한 설비의 최적 대수, 저장 Buffer

의 크기, 투입되는 Lot 크기, 그리고 작업순서방법들에 대한 의사결정을 전형적인 제품 조합이나 작업시간의 예측치 또는 추정된 Downtime등의 개념을 이용하여 분석하고 있다[1]. 또한 Kung and Changchit는 시스템의 성능을 향상시키기 위한 방안으로, 현존하는 PCB조립라인에 대한 시뮬레이션 모델을 개발하였다. 이 모델에서는 Lot Size, 간판의 수, PCB Type의 수 그리고 순서계획에 대한 규칙등의 변수를 최적화시키기 위하여 JIT(Just In Time) 개념을 기본으로하여 모의실험을 수행하고 있다[2]. 이밖에도 최근에는 객체지향형 전문가시스템에 대한 연구도 수행되고 있다. Pardhy and Dwivedi는 객체지향의 Paradigm을 기본으로한 전문가 시스템인 PCAAD(Printed Circuit Board Assembly Advisor)을 개발하여 PCB조립공정을 분석함으로써, 보다 더 우수한 PCB조립공정을 위한 대안을 수립하고 있다[3]. 그러나 이들 논문들에서 사용되는 스케줄링 문제는 주로 PCB자삽라인을 Flexible Flow Line으로 특성지어, 작업순서계획에 국한시켜 다루고 있다. 일례로 Wittrock는 단순한 Flow 라인에 대한 순서계획알고리즘을 개발하였으며[4], Kochhar and Morris는 작업리드타임의 최소화를 목표로 생산공정에 투입되는 서열계획을 수립하였고[5], Johri는 납기와 작업준비시간을 고려한 작업순서를 결정하는 알고리즘을 개발하였다[6].

그러나 PCB 자동삽입공정의 경우 스케줄링문제에 앞서 작업부하량에 대한 분석 및 공정간 부하평준화에 대한 연구가 선행되어야 한다. 왜냐하면 아무리 훌륭한 스케줄링 기법이라도 처리 가능한 생산능력 이상의 작

업물량이 초기에 주어지는 경우 최적의 Solution을 도출할 수 없기 때문이다. 즉 일정한 생산기간동안 생산능력과 작업부하량의 균형이 중요하다. 이때 생산능력에 비해 작업부하량이 적은 경우, 스케줄링상에서는 최적해의 도출이 가능하나, 반대로 생산능력에 비하여 작업부하량이 큰 경우에는 Feasible한 최적해는 불가능하다. 따라서 스케줄링이전에 생산능력을 증가시키거나, 작업부하량을 감소시켜 생산능력과 작업부하의 균형을 유지시켜야 한다.

일반적으로 생산능력과 작업부하량간의 균형을 유지시키는 방안으로는 표 1과 같이 4가지의 범주로 나누어 생각할 수 있다. 그러

표 1. 생산능력 및 작업부하량 조정 방안

구 분	생산 능력 조정	작업 부하량 조정
기업 내적인 해결방안	① 공정간의 설비대수를 조정한다. ② 작업시간을 조정한다.(잔업시간)	① 작업흐름(Route)변경한다. ② 작업우선순위를 조정한다.
기업 외적인 해결방안	① 새로운 설비를 도입한다. ② 신규 작업인력을 채용한다.	① 수주납기를 조정한다. ② 일부물량을 외주 처리한다.

나 기업 외적인 해결방안의 경우에는 추가로 투자나 경비지출이 요구된다. 일례로 새로운 설비를 도입하는 경우 설비투자가 요구되고, 외주작업을 내보내는 경우에는 추가의 경비지출이 따르게 된다. 또한 거래처의 수주에 대한 납기를 조정하는 방안은 최근과 같은 무한 경쟁의 시장여건하에서는 기업의 신뢰도와 직결되므로 추진상에 문제가 있다. 한

편 기업 내적인 해결방안의 경우에는 생산능력을 증대시키는 방안과 Bottle-neck공정의 작업부하량을 감소시키는 방안으로 나누어 생각할 수 있다. 그러나 현재 국내 대다수 전자조립산업의 PCB 자동삽입공정은 3교대 작업으로 24시간 Full로 가동되기 때문에 잔업시간을 활용한 생산능력증대는 불가능하다. 따라서 본 논문에서는 최적의 작업흐름을 편성하여 Bottle-neck공정을 최소화하는 방안에서 부하분석을 수행하고자 한다.

2. 문제 정의

PCB 자동삽입공정은 다양한 종류의 Board를 조립할 수 있는 전형적인 유연흐름생산형태로 되어 있기 때문에 단기간의 생산주기 동안 여러 종류의 Board들에 부품이 조립된다. 그런데 Board가 교체되는 경우, NC프로그램 및 부품의 재배열등에 많은 시간의 준비작업이 필요하다. 이와같은 문제점을 해결하고자 대부분의 PCB자동삽입공정에서는 유사한 부품을 사용하는 Board를 그룹지워 생산함으로써 준비시간을 단축시키고 있다. 일례로 유사성을 지닌 Board들의 작업교체시에는 몇개의 부품만을 재배열하여도 작업이 가능하므로 전체 부품을 재배치시키는 경우보다 작업교체시간 및 노력을 크게 절감시킨다. 이와같은 절감효과에 준하여 Sequencing, Radial등의 대공정들은 여러개의 소공정으로 특화되어 운영되고 있다.

이처럼 하나의 대공정을 여러개의 소공정으로 구분하여 운영하기 때문에, Board에 부품을 조립하는 경로는 하나가 아닌 다수의 작업흐름을 지닌다. 왜냐하면 Board마다 하

나의 작업흐름으로 고정시킬 경우, 선호도가 높은 일부 소공정은 많은 부하가 걸리게 되나, 그렇지 않은 다른 소공정에는 Idle이 발생하기 때문이다. 따라서 국내 전자조립산업의 PCB 조립공정에서는 그림 2와 같이 Board마다 여러개의 Route을 지님으로써, 특정 제품조합에 관계없이 공정간 부하 평균화를 유도시키고 있다.

따라서 이 논문의 주요 관심사항은 주어진 생산기간(일반적으로 1주일)동안 생산코자 하는 Job들에 대한 작업 Route의 선택은 물론, 실제 생산기간동안 긴급으로 발생하는 Job들에 대한 Route도 실시간으로 선택할 수 있는 시스템의 구축에 있다. 즉 PCB자동삽입공정의 부하평균화를 위하여 Bottle-neck공정을 최소화시키는 방안에서, 작업선호도를 고려한 최적의 작업흐름선택모델의 개발이 본 논문의 주 목적이다.

3. 수학적 모델링

3.1 가정

작업 선택을 위한 모델링을 위하여 다음과 같이 5가지의 가정을 설정하고자 한다.

- 각 Board별 작업준비시간과 작업시간은 일정한 값을 갖는다.
- 작업선택모델링을 위한 Time Bucket은 일단위로 한다.
- 작업공정간 물류이동시간은 무시한다.
- 각 공정에서 각 Job별 준비작업은 1회로 이루어짐을 원칙으로 한다.
- 초기 설정된 Lot은 생산도중에 분리되지 않는다.

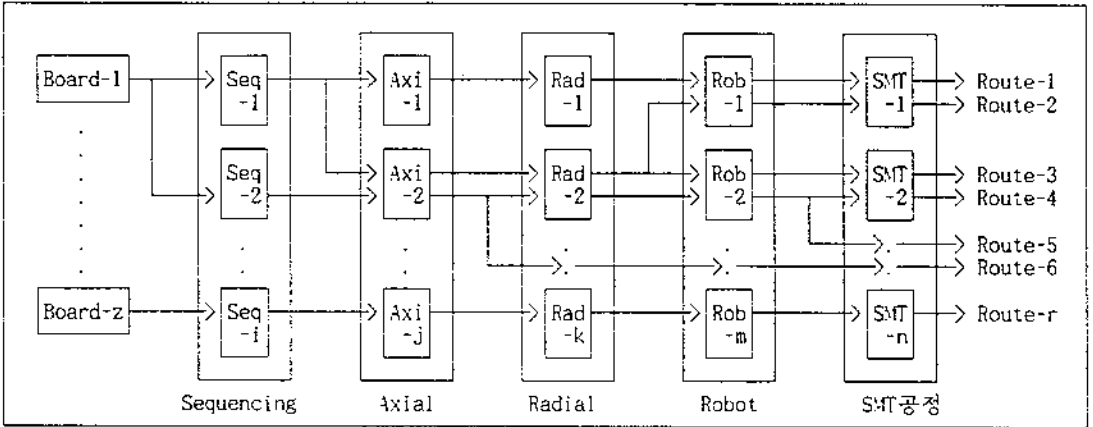


그림 2. Board별 작업흐름(Route)도

3.2 용어정의

- t : 작업분석시작일부터 t 번째 Time-bucket(일자).
- p : 대공정 단위의 작업장 변수.
- s : 대공정내의 소공정 단위의 작업장 변수. 따라서 $s \in p$.
- w : 작업지시에 따른 Work-order의 변수.
- r_w : Work-order(w)에 대한 작업흐름.
- x_{w,r_w} : Work-order(w)가 작업흐름(r_w)를 선택하는 변수.
- $l_{s,w,r_w}(t)$: t 시점에서 Work-order(w)의 작업흐름(r_w)중 소공정(s)에서의 작업부하량.
- $L_s(t)$: 시작시점부터 시간 t 시점까지 소공정(s)에 부과되는 누적된 작업부하량.
- $c_s(t)$: 시간 t 시점에 소공정(s)에서의 처리가능한 생산 Capacity.
- $C_s(t)$: 시작시점부터 시간 t 시점까지 소공정(s)에서의 누적된 총 생산 Capacity.
- N_p : 분석코자하는 최대 대공정 갯수.
- N_s : 분석코자 하는 최대 소공정 갯수.

- N_w : 분석코자 하는 최대 Work-order 갯수.
- N_{r_w} : Work-order(w)가 지나는 최대 Route 갯수.
- T : Route 선택모델 분석을 위한 최후의 종료시점.

3.3 Route선택모델의 목표

본 모델의 목표는 Work-order마다 작업흐름을 최적으로 선택함으로써 모든 공정의 생산능력과 작업부하량간의 균형을 유지하는데 있다. 따라서 이를 수식으로 표현하면 식(1)과 같다.

$$C_s(t) = L_s(t) \text{ for } s = 1, 2, 3 \dots N_s \text{ and } t = 1, 2, 3 \dots T \dots \dots \dots (1)$$

단 $C_s(t) = \sum_{t_1=1}^t c_s(t_1)$ for $s = 1, 2, 3 \dots N_s$ and $t = 1, 2, 3 \dots T$ 이고,

$$L_s(t) = \sum_{t_1=1}^t \sum_{w=1}^{N_w} \sum_{r_w=1}^{N_{r_w}} (l_{s,w,r_w}(t_1) * x_{w,r_w}) \text{ for}$$

$s = 1, 2, 3 \dots N_s$ and $t = 1, 2, 3 \dots T$ 이다.

그러나 모든 공정마다, 모든 시점에서 부하균형을 유지시키는 것은 거의 불가능하다. 그러므로 우리는 하나의 방안으로 분석기간 내에서 적절하게 부하량이 산적되고, 산분된다는 가정하에서 종료시점 T 에서의 누적된 생산능력과 누적된 작업부하량의 편차를 누적부하편차 y 라는 새로운 변수로 정의하자.

$$\begin{aligned}
 y_s^+ &= C_s(T) - L_s(T) \quad \text{단 } C_s(T) \geq L_s(T) \\
 &= 0 \quad \text{단 } C_s(T) < L_s(T) \\
 y_s^- &= L_s(T) - C_s(T), \quad \text{단 } C_s(T) < L_s(T) \\
 &= 0 \quad \text{단 } C_s(T) \geq L_s(T)
 \end{aligned}$$

위에 치환된 변수를 활용하여 작업선택모델의 목표를 새롭게 정의하면, 식(2)와 같이 구해진다.

$$\text{Min } y = \sum_{s=1}^{N_s} (y_s^+ + y_s^-) = \sum_{s=1}^{N_s} |C_s(T) - L_s(T)| \quad \dots\dots(2)$$

3.4 부하평준화를 위한 최적 작업선택 모델링

한편 제약조건식을 살펴보면, 모든 공정에서 종료시점 T 에서의 누적된 작업부하량과 누적부하편차의 합은 누적된 생산능력과 일치하여야 한다. 그리고 모든 Work-order는 오직 하나의 Route만을 선택하므로 선택된 Route에 대한 매개 변수 x_{w,r_x} 만 1의 값을 취하고 나머지 다른 Route에 대한 매개변수의 값은 모두 0(Zero)이 된다. 즉 하나의 Work-order가 지나는 모든 Route에 대한 선택변수의 합은 1이다. 따라서 부하평준화를 위한 최적 작업선택모델은 식(3)과 같이 BIP(Binary

Integer Programming)으로 정의된다[7].

$$\text{Min } y = \sum_{s=1}^{N_s} (y_s^+ + y_s^-) \quad \dots\dots\dots(3)$$

$$\text{Subject to : } \sum_{t=1}^T \sum_{w=1}^{N_w} \sum_{r_x=1}^{N_{r_x}} (l_{s,w,r_x}(t) * x_{w,r_x}) +$$

$$(y_s^+ - y_s^-) = C_s(T) \quad \text{for } s = 1, 2, 3 \dots N_s$$

$$\sum_{r_x=1}^{N_{r_x}} x_{w,r_x} = 1 \quad \text{for } w = 1, 2, 3 \dots N_w$$

$$y_s^+, y_s^- \geq 0 \quad \text{for } s = 1, 2, 3 \dots N_s$$

$$x_{w,r_x} = 0 \text{ or } 1 \quad \text{for } w = 1, 2, 3 \dots N_w$$

$$\text{and } r_x = 1, 2, 3 \dots N_{r_x}$$

그러나 이와같은 BIP모형을 실제 생산현장에 적용시켜 활용하는데는 계산규모와 여러 가지 제약으로 상당한 문제가 있다. 특히 일반적으로 부하분석주기가 일주일이라는 사실에 비추어 이 기간동안 생산될 물량이 수백 개의 Work-order에 이르며, 각 Work-order에 대한 Route와 Time Bucket을 고려할 때, 본 모형은 상당히 큰 규모의 수학적인 계산과정을 요구한다. 이는 대부분의 수학적인 알고리즘이 NP-hard문제로서 실제 실용화단계에서는 활용상의 한계가 있음을 설명하고 있는 것이다. 따라서 본 논문에서는 이와같은 수학적인 Background하에서 실제적인 자동삽입공정의 최적 작업선택을 위한 해결방안으로 Heuristic한 접근방법을 근간으로한 Min-Max 알고리즘을 개발하였다. 이에대한 상세한 내용은 다음 절에서 설명한다.

4. Heuristic Search에 의한 알고리즘 개발

4.1 전반적인 설명

앞절에서 수립한 수학적 모델이 BIP문제임을 우리는 알고있다. 이 경우 이론적으로는 해의 도출이 가능하나, 실제적으로 Solution을 구하는 경우 상당히 많은 시간이 소요된다. 이러한 현실적인 문제때문에 본 논문에서는 Heuristic접근방법을 기본으로 한 계산적인 알고리즘(Enumerative Algorithm)을 개발하여 문제를 해결코자 한다. 수학적 모델링에 의한 해의 도출은 Global한 접근방법으로 동시에 모든 조건과 제약을 고려하여 최적의 Solution을 구하지만, 본 논문의 접근 방법에서는 근시안적인(Myopic Policy) 접근방법으로 Local Optimal Solution을 점진적으로 산출함으로써 원하는 최종의 해를 구하게 된다.

한편 이 접근방법은 그림 3에서 보듯이 Heuristic Search에 의한 Work-order선택부분과 부하율에 대한 Min-Max알고리즘을 활용한 Work-order의 작업흐름선택부분으로 구분된다. 먼저 Heuristic Search는 Work-order단위로 작업흐름을 선택하기 위하여 PCB자동삽입공정의 특징 및 작업부하분석의 특징을 고려하여 Work-order를 선택하는 순위기준을 수립한다. 그리고 부하율에 따른 작업흐름의 선택에서는 각 Work-order가 지나는 모든 작업흐름(Route)에 대하여 부하율을 계산하여 가장 Bottle-neck의 발생을 억제하는 최적의 Route를 설정하게 되므로, Enumerative한 접근방법이라 할 수 있다. 한편 이와같은 두단계의 처리과정은 모든 Work-order에 대하여

반복되어 수행된다.

4.2 Heuristic Search방법에 의한 Work-order 선택

휴리스틱 기법은 완벽함에 대한 보장은 기대할 수 없는 대신에, 가능한 탐색코자하는 과정을 효과적으로 유도하는 기법이다. 즉 Global한 최적해를 보장할 수 없으나, 탐색과정을 보다 단순화하기 때문에 복잡한 실제적인 문제해결방안으로 많이 활용되고 있다[8]. 특히 PCB 자동삽입공정의 경우에는 특정한 생산기간동안에 생산되는 Board가 수백종에 이르며, Work-order는 Due-date라는 시간적인 제약과 선택해야하는 다수의 Route를 지니고 있기 때문에 동시에 모든 공정의 부하 평준화를 만족시키면서 모든 Work-order에 대한 최적의 작업흐름을 결정하기란 거의 불가능하다. 따라서 본 논문에서는 근시안적인 접근방법에 의하여 Local Optimal Solution을 단계적으로 산출함으로써 Global한 최적해는 아니지만 타 접근방법에 비하여 우수한 작업선택의 Solution을 도출한다.

PCB 자동삽입공정의 부하평준화를 만족시키는 접근방법은 크게 두가지 방안을 고려할 수 있다. 첫번째 방법은 자동삽입공정은 작업흐름의 특성상 Flow-shop의 개념으로 구성되어 있기 때문에 공정을 Stage로 고려하는 방법이고, 두번째 방법은 Work-order를 Stage로 간주하는 방법이다. 즉 작업흐름이란 Work-order마다 어떠한 공정을 거쳐 작업이 수행되는지를 결정하는 것이기 때문에 전자의 방법에서는 공정을 단계적으로 고정시킨 후 모든 Work-order에 대하여 최적 Route를 수립하는 방안이고, 후자의 방법에서는 정해

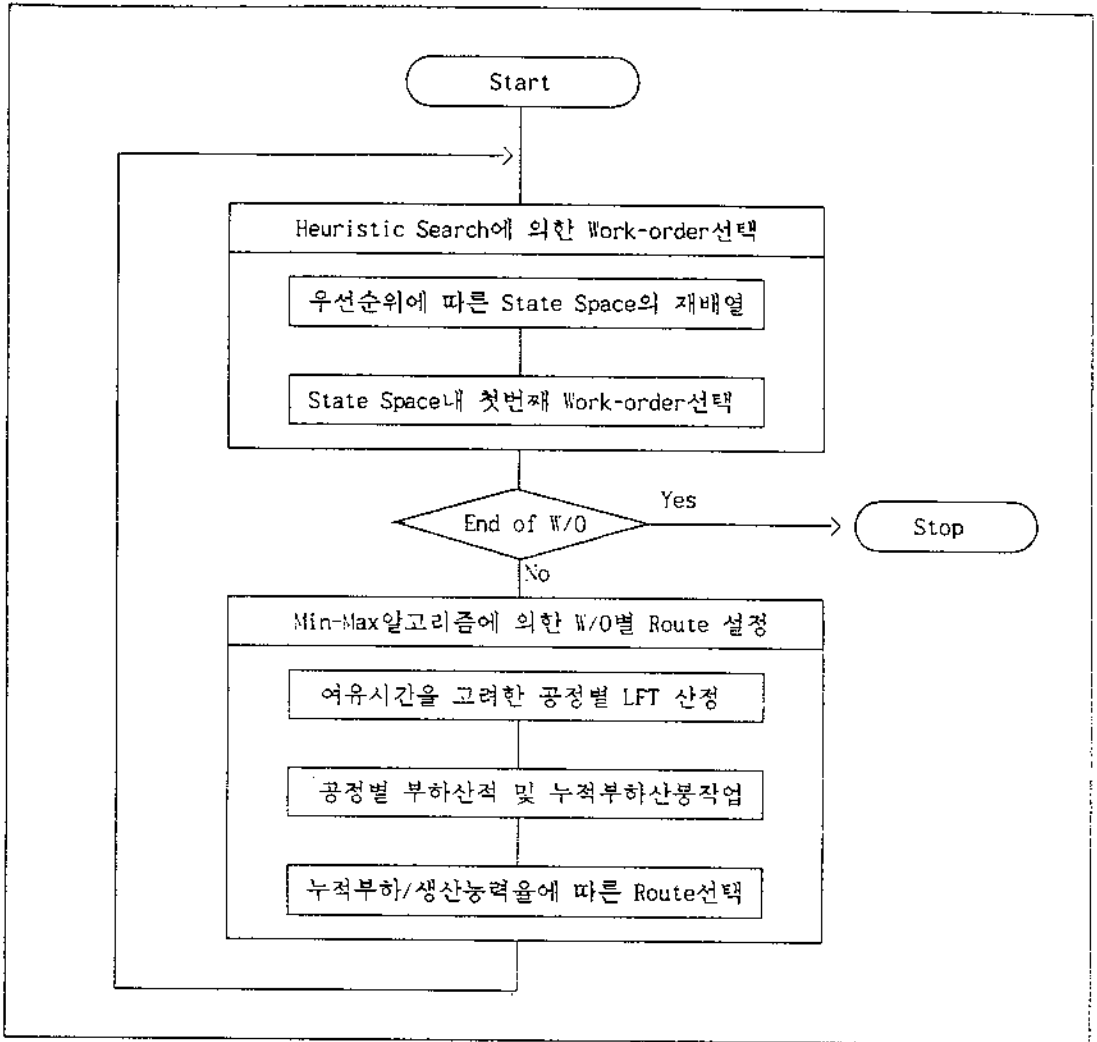


그림 3. 자동삽입공정의 Route선택을 위한 알고리즘 흐름도

진 기준에 의하여 임의의 Work-order를 선택하고 모든 공정에 대하여 부하를 분석한후 공정간 부하를 평준화시키는 작업흐름방안을 선택하는 방안이다. 그런데 전자의 방법에서는 단계마다 수립된 Solution간에는 연계성이 존재하므로 Backtracking이 발생한다. 즉 앞공정에서 최적의 작업흐름을 선택하였으나, 후 공정에서는 불균형을 이루는 경우 앞공정으

로 Backtracking하여 다시 최적의 solution을 구해야한다. 그러나 후자의 방법은 단계마다 독립성이 유지되기 때문에 Backtracking이 발생하지 않는다. 따라서 본 논문에서는 후자의 방법을 활용코자 한다.

이와같은 개념하에서 Heuristic Search를 표현하는 State Space를 정의하면, 식(4)와 같다. 따라서 Initial State와 Goal State는 식(5)

와 식(6)과 같이 표현된다.

$$\text{State Space } S = \{\text{Route을 선택하지 못한 Work-order}\} = \{w_1, w_2, \dots, w_k\} \quad (4)$$

$$\text{Initial State } S = \{\text{생산코자하는 모든 Work-order}\} = \{w_1, w_2, \dots, w_{N_x}\} \quad (5)$$

$$\text{Goal State } S = \{\text{Null}\} \quad (6)$$

단, w_i : 정해진 우선순위에 따라 State Space내에 i 번째로 정렬된 Work-order

다음은 State Space내 존재하는 Work-order 들에 대하여 어떠한 기준으로 서열지워서 선택하느냐가 우리의 관심사항이다. 일반적으로 우선순위기준에 의하여 탐색과정이 수립되기 때문에 순위기준에 대한 설정은 매우 중요하다. 이와같은 중요성에 비추어, 본 논문에서는 PCB자동삽입공정의 유연성 측면과 부하산봉작업의 특성을 고려하여 다음과 같은 우선순위기준을 설정하였다.

순위기준 1 : Board가 지나는 Route의 갯수가 적은 순서로 배열한다.

일부의 Board는 작업흐름이 고정되어 있지만, 대부분의 Board의 경우에는 하나 이상의 Route을 지니고 있다. 일례로 Main-Board의 경우에는 제품의 중요성 및 품질안정때문에 고정된 작업흐름을 갖고 있지만, Sub-Board의 경우에는 공정간의 부하를 고려하여 작업을 진행하기 위하여 하나 이상의 작업흐름을 지니고 있다. 이와같은 현실속에서 고정된 Route을 지닌 Board를 제일 나중에 처리하는

경우에는 부하불균형으로 작업이 불가능하게 된다. 따라서 Route의 갯수가 적은 Board부터 State Space에 배열한다.

순위기준 2 : 납기순으로 Work-order를 배열한다.

누적부하에 대한 산봉작업은 부하량이 Peak를 이루는 Time-bucket보다 이전의 Time으로 산봉되고, 이후의 시점으로는 산봉되어 지지 않는다. 따라서 납기순으로 W/O를 배열하여 부하를 LFT에 따라 해당 Time-bucket에 산적함으로써 공정간의 부하에 대한 산봉작업을 효율적으로 수행할 수 있다. 일례로 뒤에 제일 납기가 빠른 작업을 처리한다면, 기존의 평균화된 공정간의 부하를 혼란시키는 결과를 가져온다.

순위기준 3 : 작업부하량이 큰 작업부터 Work-order를 배열한다.

우리는 가정에서 초기에 설정한 Lot을 도중에 분리시키지 않기로 정의하였다. 따라서 작업부하량이 큰 Work-Oder순으로 배열함으로써 공정간의 불균형을 사전에 방지할 수 있다. 일례로 부하량이 적은 순으로 배열시키는 경우, 최종단계에서 큰 부하량이 산적되기 때문에 이전의 공정간 균형상태는 깨질 수 있다.

그리고 이들 순위기준들은 순위가 높은 순서로 Tie-break 규칙을 적용시킴으로써 Work-order를 State Space내 정렬시킨다. 일례로 2개의 Work-order의 1순위값이 동일한 경우에는 2순위수치를 비교하여 정렬시키고, 만약 2순위값마저도 동일한 경우에는 3순위의 기준을 적용시킨다. 따라서 Initial State로 부터 Goal State로 전개되는 단계는 Work-order갯

수만큼 반복되어 진행된다. 그러므로 본 논문에서 제안하는 Heuristic Search방식을 활용하는 경우에는 N_w 번의 처리과정을 거쳐 Goal State에 접근할 수 있다. 그러나 일반적인 Search방법에서는 N_w 번 만큼의 단계를 처리하여야 Goal State에 도달할 수 있다. 이상과 같은 Heuristic Search방법을 종합적으로 요약하면, 다음과 같은 절차로 표현할 수 있다.

Procedure : 1) Initial State를 평가한다. 만약 그것이 Goal State라면, 과정을 중단한다. 그렇지 않으면, initial State를 Current State로 한다.

2) Current State가 Goal State에 이룰때 까지 다음 과정을 반복한다.

- ① Current State내에서 가장 우선순위가 높은 Work-order를 선택한다.
- ② 누적부하율에 대한 Min-Max 알고리즘을 통한 최적 Route를 설정한다.
- ③ Route가 결정된 Work-order를 Current State에서 제외시킨후, 새롭게 Current State를 수립한다.

4.3 Min-Max알고리즘에 의한 작업흐름(Route) 선택

Work-order별 작업흐름(Route)의 선택은 수학적인 모델링에서 언급되었듯이 공정간 누적된 부하량의 균형을 이루는 목적하에서 최적이 수립된다. 따라서 본 과정은 다음과 같이 3단계로 구분하여 생각할 수 있다.

- 1) 여유시간의 공정간 배분에 따른 공정별 LFT산정
- 2) 공정별 LFT를 기준으로한 부하산적 및

누적부하 산봉작업

3) 누적부하율에 대한 Min-Max알고리즘에 의한 Route선택

4.3.1. 총 여유시간의 공정간 배분에 따른 공정별 LFT산정

공정흐름에 따른 일련의 작업활동을 위한 유용한 시간 간격들은 가장 빠른 시작시점(EST)과 가장 늦은 종료시점(LFT)에 의하여 표현된다. 그러나 본 논문에서는 LFT만을 주로 활용하는데, LFT는 총 여유시간을 어떠한 기준에 의하여 배분시키는가에 따라 정의될 수 있다[9]. 먼저 임의의 Work-order에 대한 총 여유시간($ST(w)$: Total Slack Time of Work-order w)을 정의하면 식(7)과 같다.

$$ST(w) = \left\{ [Due-date(w) - EST(w)] - \sum_{p=1}^{N_p} l_{w,p} \right\} \dots\dots(7)$$

단, $l_{w,p}$: 대공정 p 에서의 Work-order(w)에 의한 작업부하량

$$\left\{ l_{w,p} = \sum_{s \in p} \sum_{t=1}^{\infty} \sum_{r_w=1}^{N_{r_w}} (l_{s,w,r_w}(t) * x_{w,r_w}) \right\}$$

이와같이 산정된 총 여유시간을 공정마다 배분함에 있어서 여러가지 방법이 존재하나, 일반적으로 공정의 부하비율(R : Ratio of Load)에 비례하여 분배시킨다. 따라서 임의의 공정 p 에 대한 여유시간을 정의하면 식(8)과 같으며, 이를 활용한 공정 p 에 대한 LFT는 식(9)와 같다.

$$ST(w,p) = R(w,p) * ST(w) \quad \text{단,} \quad \sum_{p=1}^{N_p} R(w,p) = 1 \dots\dots\dots(8)$$

$$LFT(w,p) = EST(w) + \sum_{p_j=1}^p [l_{w,p_j} + ST(w,p_j)] \dots\dots\dots(9)$$

단, $R(w,p)$: Work-order(w)의 전체작업부하에대한 공정 p 에서의 부하율

$$\left\{ R(w,p) = \frac{l_{w,p}}{\sum_{p=1}^{N_p} l_{w,p}} \right\},$$

그리고, $ST(w,p)$: 공정 p 에서의 Work-order(w)가 지나는 여유시간.

4.3.2. 공정별 부하산적 및 누적부하 산봉작업

앞에서 산출된 공정별 LFT를 기준으로 각각의 Work-order의 작업부하량은 산적한다. 이때 산적되는 작업부하량은 “Machine * Hour”의 단위로, Board가 지나는 작업흐름에 따라 공정마다 정해진 작업준비시간과 실작업시간의 합으로 계산한 후, 해당 공정의 LFT와 일치하는 Time-bucket에 누적한다. 한편 생산능력은 설비대수와 당일 작업시간의 곱으로 산출되는데 만약 해당 공정내 일부 설비가 예방정비 및 설비의 고장으로 작업이 불가능한 경우에는 이를 사전에 고려해야 한다.

따라서 공정별 작업부하량 및 생산능력에 대한 Level을 도표로서 도시하면, 그림 4와 같이 나타낼 수 있다. 특히 맨 하단부분의 계산된 누적 부하량/누적 생산능력의 비율은 생산능력을 고려한 작업 Load수준을 표현하기 때문에 이에대한 수치를 활용하여 공정마다의 부하수준을 판단할 수 있다. 또한 이들 누적된 부하량과 생산능력간의 비율(Rate)은

최고의 산봉된 부하수준을 의미한다. 일례로 4번째 시점에서 1.03이라는 수치는 이 Peak-time의 시점이전의 모든 기간에서 1.03의 작업부하량이 부과된다는 사실을 의미한다. 따라서 이들 수치를 활용하여 생산현장에 걸리는 산봉된 부하수준을 표현하면 그림 5와 같이 나타낼 수 있다. 이때 기간중 최고의 Peak를 이루는 산을 허물고 난후, 이후 시점에 대해서는 새롭게 부하산봉작업을 수행하여야 한다. 결과적으로 산봉작업은 댐을 막아 수위를 조정하는 개념으로 생각할 수 있다.

4.3.3. 누적부하에 대한 Min-Max알고리즘에 의한 Route선택

산봉작업단계에서는 가장 높은 부하의 산을 앞쪽으로 붓겨시켜 Peak-time시점에 걸리는 부하를 평균화시킨다. 그리고 이후 시점에 대하여는 동일한 방법으로 두번째 높은 부하의 산을 발견하여 산봉시킨다. 만약 두번째의 산봉작업이후에 산봉할 기간이 존재한다면 동일한 방법으로 산봉시킴으로써 전체적인 분석기간동안의 공정부하를 평균화시킨다.

이때 평균화된 수위는 생산능력과 비교하여 작업지시된 부하량을 소화할 수 있는지 여부를 결정하는 중요한 변수가 된다. 일례로 수위기준치가 1보다 작은 값일 경우에는 현재의 생산능력으로 부과된 모든 작업을 수행할 수 있음을 의미하며, 반대로 1보다 큰 경우에는 현 생산능력으로는 이 공정에 전달된 모든 작업을 수행하는데 어려움이 있음을 의미한다. 따라서 임의의 Route가 $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow s_4$ 공정으로 정의된 경우라면, 먼저 각 공정마다 계산된 수위기준치중 가장 높은 수치를 해당 Route의 누적부하/누적Capa율로

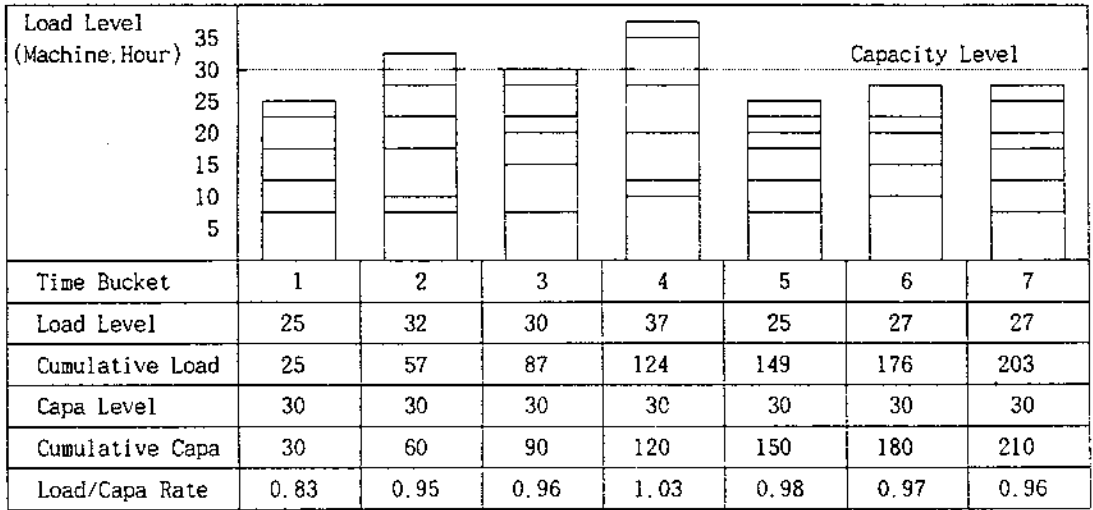


그림 4. 부하산적 및 누적부하에 의한 Load/Capa Rate 산출도표

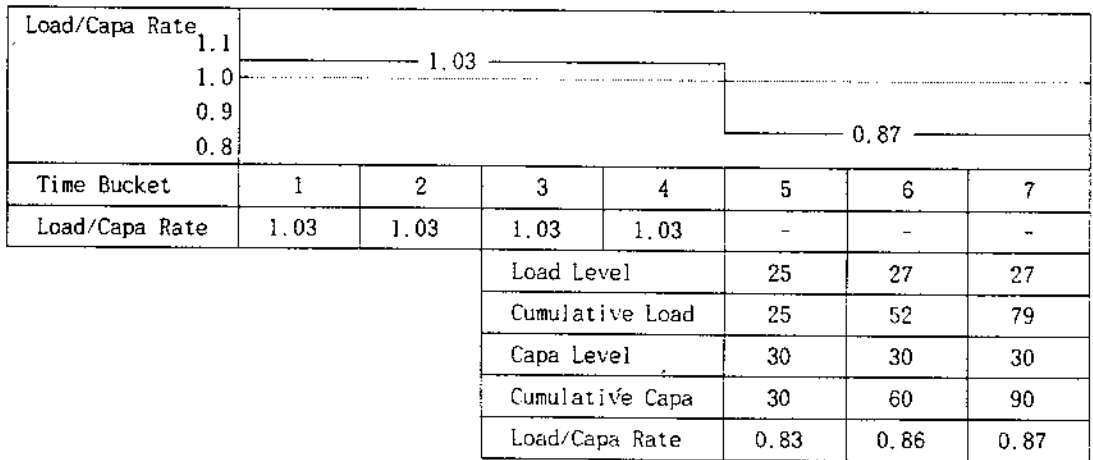


그림 5. 산봉작업결과에 따라 평준화된 작업부하율 산출도표

정의할 수 있다. 이와같은 방법에 의하여 산출된 수위기준치를 근거로 하여 해당 수위기준치가 가장 낮은 Route을 선택하면, 공정의 Bottle-neck을 최소화하는 결과를 가져온다.

한편 다수의 Route에 대하여 선호도가 존재하는 경우 이에대한 가중치도 적용이 가능하다. 일례로 하나의 Board에 2개의 Route가

존재하는 경우, 일상적으로는 Route-1을 선택하나, 이로인하여 특정 설비의 부하율이 높고, 다른 Route에 의한 설비의 부하율이 낮은 경우 Route-2를 활용한다면, 가중치를 활용하여 선호도에 따른 작업흐름을 결정할 수 있다. 따라서 이상의 Route 선택과정을 알고리즘으로 정리하면 다음과 같은 절차로 요약

된다.

Procedure : 1) 초기 Route를 선택한다.

만약 Route가 오직 하나만 존재하는 경우에는 Current—Route에 해당 Route번호를 기록한후, 3)으로간다.

그렇지 않으면, 초기치(Current—Route, Current—Load—Rate등)를 부여한후, 2)를 진행한다.

2) 선택할 새로운 Route가 없을 때까지 다음 과정을 반복한다.

① Route에 정의된 각 공정마다의 최종작업종료시점(LFT)을 구한다.

② 각공정의 Peak Time의 누적 부하/생산능력을 산출한다.

③ 공정마다 산정된 최대 부하/생산능력을 비교하여, 이들중 최대값을 해당 Route의 부하율로 선택한다.

④ Route의 부하율에 선호도의 가중치를 곱한후 Current—Load—Rate와 비교한다.

만약 새로운 Route의 부하율이 Current—Load—Rate보다 작다면, Current—Route를 새로운 Route번호로 수정하고, 이때의 부하율을 Current—Load—Rate로 변경한다.

그렇지 않으면, 2)번으로 되돌아 간다.

3) Current—Route를 현 Work—Order의 최적 Route로 결정한다.

그리고 Route에 의해 정의된 각 공정의 LFT를 산정한후, 공정마다의 부하량을 LFT를 기준으로하여 부하Table에 산적시킨다.

5. 사례분석

5.1. 개요

실제적인 생산현장의 Route선택방법을 살펴보면, 첫공정이나 Bottle-neck공정만을 대상으로 부하를 분석함으로써 Work-order의 Route를 결정한다. 이 경우 부분적인 공정만을 국한하여 부하를 분석하였기 때문에 다른 공정에서 부하의 불균형을 발생시킬 수 있다. 따라서 본 사례분석에서는 그림 6과 같은 생산공정을 대상으로 단일공정의 부하분석을 통한 Route선택방법과 전체공정에 대한 부하분석결과에 의한 Route선택방법을 가동율이라는 지표로 분석하였다. 이때 가동율이라함은 순수하게 작업에 소요되는 시간을 기준으로 한 설비운전율을 의미하기 때문에 작업준비시간이나 설비이상사고등으로 인한 유휴시간등은 포함되지 않는다. 이를 역으로 표현하면, Blocking이나 Starvation으로인한 작업손실율을 제외한 가동율이라고 표현할 수 있다.

$$\begin{aligned} \text{가동율} &= (\text{W/O의 부하시간}) / (\text{순수한 작업 가능 시간}) \\ &= 1 - (\text{Blocking이나 Starvation으로 인한 작업손실율}) \end{aligned}$$

한편, 시뮬레이션을 통한 모의실험을 위하여 6개의 가정을 설정하면 다음과 같다.

- 모든 Work-order는 일련의 3개 대공정을 거쳐 작업이 완료된다.

- 각 대공정에서의 작업은 대공정내 존재하는 소공정중 하나에서 작업이 수행된다.

- 대공정 및 소공정간의 생산능력은 평균화되어 있다.
- Work-order의 Route갯수는 최대한 10개로 제한한다.
- 각 Work-order에 의하여 걸리는 각 공정별 작업부하는 동일하다.
- 공정간 물류이동시간이나 작업준비시간은 무시한다.

법이 기존의 단일공정만을 고려한 Route선택 방법에 비하여 약 5%정도수준의 설비가동율을 증가시킬 수 있었다.

한편 위의 결과에 대하여 대공정수, 소공정수, 최대Route의 갯수를 변화시켜 시물레이션을 수행한 결과 (표 2)에서 보듯이 아래와 같은 결과를 도출하게 되었다.

1) Work-order가 선택할 수 있는 Route갯

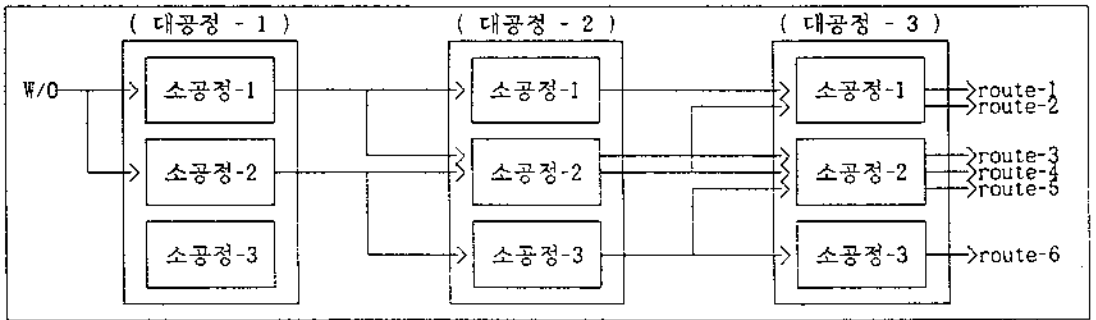


그림 6. 생산공정의 구성도 및 작업흐름도

5.2. Simulation결과 분석

초기 Simulation에서는 대공정수($N_p=3$), 소공정수($N_s=3$), Work-order수($N_w=300$), 최대 Route수($N_r=10$)등의 변수를 설정하여 첫공정의 부하를 기준으로한 가동율과 모든 공정의 부하를 고려한 가동율을 분석한 결과 그림 7과 같이 산출되었다. 한편 본 Simulation은 앞에서 수립한 두개의 과정을 C언어를 활용하여 PC-486상에서 구현하였다.

분석된 결과에 의하면, 단일공정의 부하만을 고려한 Route선택의 경우 순수한 가동율은 W/O갯수가 150개에 이르러 95%에 수렴하고 있으며, 전공정의 부하를 고려한 Route선택방법의 경우에는 110개의 W/O수에 이르러 가동율이 99%에 수렴하고 있음을 확인할 수 있었다. 즉 본 모델에 의한 Route선택방

수가 증가하는 경우 가동율은 점진적으로 증가한다. 즉 제품에 대해 고정된 Route에 의한 작업보다는 다수의 Route를 지너 제품의 유연성을 증가시키는 경우 공정간의 부하 평균화로 인한 설비의 가동율을 증가시킬 수 있었다.

2) 대공정수가 증가하면 할수록, 가동율은 점차로 감소한다. 즉 소공정수를 고정시킨후, 대공정수를 1개부터 10개까지 증가시켜 Simulation을 수행한 결과, 대공정수가 증가하면 할수록 생산공정의 불균형으로 Blocking과 Starvation이 증가되어 설비의 가동율은 감소되었다.

3) 반면에 소공정수의 변화는 공정간의 부하불균형과는 무관하였다. 즉 대공정수를 고정시킨후 소공정수를 1개로 부터 10까지 증

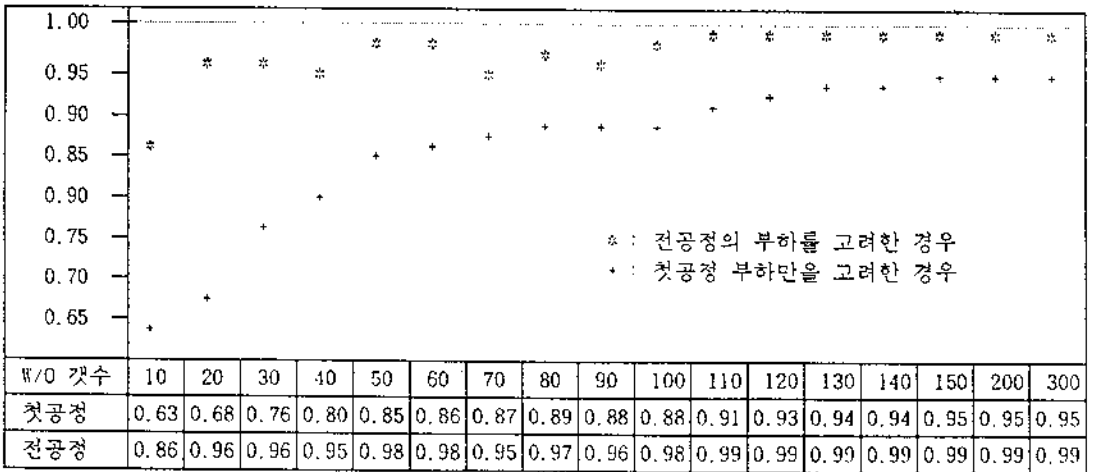


그림 7. Simulation 분석결과에 따른 가동율 분석표

가시킨 결과 가동율은 특정한 추이를 발견할 수 없었다. 그러나 소공정이 1개인 경우에는 Route가 고정되므로 두 방법 모두에서 가동율이 100%에 이르렀으나, 그 이후에는 특정한 흐름으로 감소하거나 증가함이 없이, 주어지는 Work-order의 조건에 따라 가동율이 변화되었다.

6. 결론

이 논문은 PCB 자동삽입라인과 같은 FFL (Flexible Flow Line)을 대상으로 각 공정간의 부하 평균화를 유지시키는 최적의 Route를 결정하는 모형에 관한 연구이다. 사례분석에서도 살펴보았듯이, 공장전체의 생산성

표 2. 변수변화추이에 따른 설비가동율 도표

N _w 를 변화시키는 경우 (N _p = 3, N _s = 3, N _w = 150)			N _p 를 변화시키는 경우 (N _s = 3, N _w = 10, N _w = 300)			N _s 를 변화시키는 경우 (N _p = 3, N _w = 10, N _w = 300)		
N _w	첫공정	전공정	N _p	첫공정	전공정	N _s	첫공정	전공정
1	0.92	0.92	1	1.00	1.00	1	1.00	1.00
2	0.92	0.96	2	0.96	1.00	2	0.98	0.99
3	0.93	0.97	3	0.95	0.99	3	0.95	0.99
4	0.93	0.98	4	0.95	0.99	4	0.97	0.98
5	0.93	0.98	5	0.95	0.99	5	0.94	0.97
6	0.94	0.98	6	0.94	0.99	6	0.98	0.98
7	0.94	0.98	7	0.94	0.98	7	0.99	0.99
8	0.94	0.99	8	0.94	0.98	8	0.94	0.99
9	0.95	0.99	9	0.94	0.98	9	0.98	0.99
10	0.95	0.99	10	0.93	0.98	10	0.95	0.98

은 단일 공정의 부하분석에 의존하기 보다는 공장전체의 부하를 평준화시키는 측면에서 접근하여야한다. 또한 FFL에서는 제품의 유연성측면에서 고정된 Route를 지닌 생산시스템보다는 다수의 Route를 지니는 생산시스템의 경우가 공정간의 균형을 꾀할 수 있다. 또한 다수의 공정으로 생산라인이 복잡하게 구성되어있는 경우 공정간의 Blocking과 Starvation으로 인한 작업중단이 증가되므로 결국에는 생산성이 저하되기때문에 우수한 Route선택모형이 요구된다.

이와같은 이유때문에 많은 논문에서 이 작업흐름선택문제를 최적화시키고자 수학적인 개념을 활용하여 모델링하고 있으나, 이들 수학적인 모델들을 실제 생산현장에 적용시키는 경우, 수많은 변수와 시스템의 복잡성으로 인하여 Solution을 구하는데 어려움이 있다. 따라서 본 논문에서는 작업선택을 위한 우선순위규칙과 작업부하율을 활용한 Min-Max개념을 기본으로한 Heuristic 알고리즘을 개발하였다.

특히 대부분의 생산현장에서는 긴급주문에 따른 새로운 생산지시가 Random하게 발생하기 때문에 기존의 수학적인 모델링에 의한 정적인 방법으로는 문제 해결이 불가능하였다. 그러나 이 논문에서 제시한 모델은 누적된 부하 Table을 활용하여 신규로 발생하는 Work-order에 대한 Route를 단시간내 수립가능하기 때문에 실시간의 개념이 중시되는 최근의 생산시스템에서도 활용이 가능하다.

마지막으로 이 논문에서 부족한 내용으로는 작업선택모델링에서 활용한 순위기준을 기업의 요구조건에 따라 수립하였다는 사실이다. 즉 다른 순위기준 배열(예 순위기준 3

→ 순위기준 1 → 순위기준 2)과의 비교분석이 수행되지 않았다. 일반적으로 조립라인의 부하평준화문제에서는 3 순위기준인 LPT Rule을 최우선적으로 활용하는 것이 좋은 결과를 가져오는 것으로 언급되고 있다. 그러나 이 논문에서는 Main-Board에 대한 품질문제 및 주문에 대한 납기문제를 우선적으로 고려해야하는 기업의 특성으로 인하여 LPT Rule을 3번째의 순위기준으로 설정하였다. 따라서 추후 연구과제로는 이들 순위기준에 대한 비교분석 연구를 수행하고자 한다.

참 고 문 헌

- [1] P.K.Johri, "Engineering a Circuit Board Assembly Line for a Desired Capacity and Flowtime," *Journal of Manufacturing Systems*, Vol.10/No.6, 1991, pp.492-500.
- [2] H.K.Kung and C.Changchit, "A Just-in-time Simulation Model of A PCB Assembly Line," *Computers & Industrial Engineering*, Vol.20/No.1, 1991, pp.17-26.
- [3] S.K.Padhy and S.N.Dwivedi, "PCAAD - An Object-oriented Expert System for Assembly of Printed Circuit Boards," *Expert Systems*, Vol.9/No.1, 1992, pp11-24.
- [4] R.J.Wittrock, "An Adaptable Scheduling Algorithm for Flexible Flow Lines," *Operations Research*, Vol.36/No.3, 1988, pp. 445-453.
- [5] S.Kochhar and R.J.T.Morris, "Heuristic Methods for Flexible Flow Line Scheduling," *Journal of Manufacturing Systems*, Vol.6/No.4, 1987, pp.299-314.

-
- [6] P.K.Johri, "A Heuristic Algorithm for Loading New Work on Circuit Pack Assembly Lines," *International Journal of Production Research*, Vol.28/No.10, 1990, pp.1871-1883.
- [7] F.S.Hillier and G.J.Lieberman, *Introduction to Operations Research*, 5th edn, McGraw-Hill,Inc., 1990, pp.458-466.
- [8] E. Rich and K. Knight, *Artificial Intelligence*, 2nd edn, McGraw-Hill,Inc., 1991, pp.63-101.
- [9] J.K.Lee, M.S.Suh and M.S.Fox, "Contingencies for the Design of Scheduling Expert Systems," *Expert Systems With Applications*, Vol.6/No.3, 1993, pp.219-230.