Balancing Assembly Line In an Electronics Company

박 경 철, 강 석 훈, 박 성 수
한국과학기술원 산업공학과

김 완 희
삼성데이타시스템

In general, the line balancing problem is defined as the problem of finding an assignment of the given jobs to the workstations under the precedence constraints given to the set of jobs. Usually, the objective is either minimizing the cycle time under the given number of workstations or minimizing the number of workstations under the given cycle time. In this paper, we present a new type of an assembly line balancing problem which occurs in an electronics company manufacturing home appliances. The main difference of the problem compared to the general line balancing problem lies in the structure of the precedence given to the set of jobs. In the problem, the set of jobs is partitioned into two disjoint subsets. One is called the set of fixed jobs and the other, the set of floating jobs. The fixed jobs should be processed in the linear order and some pair of the jobs should not be assigned to the same workstation. Whereas, to each floating job, a set of ranges is given. The range is given in terms of two fixed jobs and it means that the floating job can be processed after the first job is processed and before the second job is processed. There can be more than one range associated to a floating job.

We present a procedure to find an approximate solution to the problem. The procedure consists of two major parts. One is to find the assignment of the floating jobs under the given (feasible) assignment of the fixed jobs. The problem can be viewed as a constrained bin packing problem. The other is to find the assignment of the whole jobs under the given linear precedence on the set of the floating jobs. First problem is NP-hard and we devise a heuristic procedure to the problem based on the transportation problem and matching problem. The second problem can be solved in polynomial time by the shortest path method.

The algorithm works in iterative manner. One step is composed of two phases. In the first phase, we solve the constrained bin packing problem. In the second phase, the shortest path problem is sloved using the phase 1 result. The result of the phase 2 is used as an input to the phase 1 problem at the next step. We test the proposed algorithm on the set of real data found in the washing machine assembly line.

1. 서론

작업장균형문제(line balancing problem)는 조립산업에 있어서 가장 빈번히 나타나는 생산관리 문제 중의 하나이다. 일반적으로 작업장균형문제는 주어진 작업(task)들을 우선순위 제약(precedence constraint)하에서 작업장에 할당하는 문제이다. 목적함수는 주어진 작업장 수제한하에서 cycle time을 최소화하거나 또는 주어진 cycle time하에서 작업장수를 최소화하는 것이다.

본 논문에서는 가전제품을 조립생산하는 공장에서 발생하는 작업장균형문제를 다루고 있다. 이 문제는 일반적인 작업장균형문제와는 몇가지 다른 특질을 나타낸다. 먼저 전체작업 집합은 두 개의 부분집합으로 나뉜다. 첫번째 집합은 고정작업의 집합이고 두번째 집합은 유동작업의 집합이다. 고정작업의 수를 m 이라 하고 고정작업집합을 M 이라 하자. 즉, $M = \{1, ..., m\}$ 이라 하면, 이들 작업간에는 다음과 같은 우선순위 제약이 주어진다.

$$1 \rightarrow 2 \rightarrow 3 \rightarrow \cdot \cdot \cdot \rightarrow m$$

여기서 → 은 선행관계를 나타낸다. 즉, 고정작업들은 1 번부터 순서대로 작업되어야 함을 의미한다. 고정작업간에는 선행관계제약외에 다른 종류의 제약이 있다. 문제의 입력으로 M 의 부분집합 IMC 가 주어진다. i 를 IMC 의 원소라 하면, i 는 i-1 과는 다른 작업장에 할당되어야 함을 나타낸다. 이를 합병불가 조건이라 한다. 유동작업의 수를 n 이라 하고 유동작업의 집합을 N 이라 하자. N 의 각 원소 j 에 대하여 (a_j,b_j) 형태의 범위가 주어진다. 여기서 $a_j,b_j \in M$ 이고, $a_j<b_j$ 를 만족한다. 이 범위는 유동작업 j 가 고정작업 a_j 를 수행한 이후에 수행될 수 있고 고정작업 b_j 를 수행하기 전에 수행되어야 함을 의미한다. 하나의 유동작업 j 에 대하여 하나이상의 범위가 명시되어 있으며, 이 범위의 집합을 $R(j)$ 로 나타내기로 한다. 따라서, 본논문에서 다루는 작업장균형문제는 고정작업과 유동

작업을 고정작업간의 우선순위 제약과 합병불가 조건을 만족시키고 유동작업의 범위 제약을 만족시키면서 작업장에 할당하는 문제이다. 목적함수는 주어진 작업장수 제약하에서 cycle time 을 최소화하는 것이다.

본논문에서는 이 문제에 대하여 최단경로문제(shortest path problem)와 수송문제(transportation problem)를 이용한 해법을 제시한다. 2 장에서 문제를 두 개의 딸림문제로 분해하여 그 성질을 분석하고 3 장에서 각 딸림문제의 해법과 이를 이용한 원문제의 해법을 제시한다. 4 장에서는 결론을 제시한다.

2. 문제분석

앞 장에서 제시된 작업장균형문제를 다음과 같은 두 가지 딸림문제로 분해한다.

(문제 1) 각 작업장에 대한 고정작업의 할당이 결정된 상태에서 유동작업의 할당을 결정한다.

즉, (문제 1)은 우선순위제약과 합병불가조건을 만족하는 고정작업의 할당이 결정되었다고 할 때, cycle time 을 최소화하는 유동작업의 할당을 결정하는 문제이다. 이를 정식화하면 아래와 같다. 먼저 다음과 같은 기호를 정의하자.

$W(k)$: 작업장 k 에 할당된 고정작업의 집합
w : 작업장의 수
$B(j)$: 유동작업 j 가 할당될 수 있는 작업장의 집합
$A(k)$: 작업장 k 에 할당될 수 있는 유동작업의 집합
p_i : 고정작업 i 의 작업시간
a_j : 유동작업 j 의 작업시간
C : cycle time
b_k : 작업장 k 의 여유시간

$$(\; b_k = C - \sum_{i \in W(k)} p_i \;)$$

여기서 $B(j)$ 는 $W(k)$ 와 $R(j)$ 로부터 결정할 수 있다.
다음과 같은 문제를 정의한다.

$$\text{(TP)} \quad \min \sum_{j=1}^{n} \sum_{k \notin B(j)} x_{jk}$$

$$\sum_{k=1}^{w} x_{jk} = a_j \text{ for all } j \in N$$

$$\sum_{j=1}^{n} x_{jk} + s_k = b_k \text{ for all } k = 1,\dots,w$$

$$x_{jk} \geq 0 \text{ for all } j \in N,\ k = 1,\dots,w$$

$$s_k \geq 0 \text{ for all } k = 1,\dots,w$$

여기서 s_k 는 여유변수이다. 이문제는 다음과 같은 성질을 가진다.

정리 1.

(1) 유동작업이 주어진 cycle time C 이내에 할당가능하면, (TP)의 최적목적함수는 0 의 값을 갖는다.

(2) 유동작업이 주어진 cycle time C 이내에 할당가능한 필요충분조건은 (TP) 의 최적목적함수값이 0 이고, 모든 여유변수가 기저변수인 최적정점해가 존재하는 것이다.

따라서, cycle time C 의 값을 변화시켜가면서 (TP)를 풀면, 주어진 고정작업의 할당하에서 가능한 cycle time 의 하한을 구할 수 있다.

다음으로 딸림문제 2 를 고려한다.

(문제 2) 유동작업에 대하여 선형우선순위관계(linear precedence)가 주어졌을 때, cycle time을 최소화하는 할당을 구한다.

여기서 선형우선순위관계는 고정작업에 주어진 것과 같은 우선순위관계를 의미한다. 이 관계는 물론 가능한 관계(feasible relation)이어야 한다. (문제 2)는 유동작업간의 고려순서를 고정하고 작업할당을 하는 문제이다. 이 문제는 우선순위가 주어진 bin packing 문제의 일종으로 볼 수 있다. 일반적으로 bin packing 문제는 NP - hard 이지만, (문제 2)의 경우는 동적계획법을 이용한 최단경로문제로 다항식시간(polynomial time)안에 최적해를 구할 수 있다.

정리 2. (문제 2)는 최단경로문제를 이용하여 다항식시간안에 최적해를 구할 수 있다.

다음 장에서는 이들 두가지 딸림문제를 이용한 원문제의 해법절차를 고려한다.

3. 해법절차

해법절차는 2 장에서 다룬 2 가지 딸림문제를 조합함으로써 구성된다. 먼저 (문제 1)의 해를 (문제 2)의 입력으로 사용하여 (문제 2)를 풀고, 다시 (문제 2)의 결과를 (문제 1)의 입력으로 사용한다. 우선 각 딸림문제의 해법에 대하여 기술하기로 한다.

3.1 (문제 1) 의 해법

3 장에서 기술한 대로 (문제 1)은 (TP) 의 여유변수가 모두 기저변수인 최적해를 구함으로써 해결할 수 있다. 그러나, 이문제는 NP - hard 이다. ([6]) 따라서, 본 연구에서는 다음과 같은 발견적 기법을 이용하여 이 문제를 해결한다. 먼저 cycle time 의 하한을 구한다. 이 하한은 작업시간의 합을 고려함으로써 얻어질 수 있다. 하한값이 얻어지면, cycle time 을 이 하한값으로 두고 (TP) 를 푼다. 만일 최적목적함수의 값이 0 보다 크다면, 이는 주어진 cycle time 내에서는 유동작업을 할당할 수 없음을 의미한다. 이때, (TP) 의 결과를 이용하여, 상한을 구한다. 다음으로 cycle time 을 상한과 하한사이에서 2 분 탐색 (binary search) 기법을 사용하여 조정해 가면서, (TP) 를 푼다. 이를 통해 최적 cycle time 의 가능하한이 얻어진다. 이를 C' 이라 하자. 만일 C' 에서 (TP) 를 풀었을 때, 최적기저해에 모든 여유변수가 들어가 있으면, 문제는 해결된다. 만일 그렇지 않다면, 비기저 여유변수에 대하여, 선회연산(pivoting)을 수행하여, 여유변수를 기저변수로 만들어 준다. 이 결과 모든 여유변수가 기저에 들어가면, 문제는 해결된다. 그렇지 않은 경우에는 2 개 이상의 작업장에 할당된 유동작업이 존재하게 된다. 이들 유동작업의 집합을 P 라 하자. 그러면, 다음의 결과가 성립한다.

성질 1. $|P| \le w-1$

즉, 2 개이상의 작업장에 할당된 유동작업의 수는 작업장의 수보다 작다. 이들 유동작업들에 대하여 작업장을 정해주기 위해 다음의 문제를 푼다.

$$\text{(MP)} \quad \min \quad \max_k \sum_{j \in P} w_{jk} x_{jk}$$

$$\sum_{k \in B(j)} x_{jk} = 1$$

$$\sum_{j \in A(k)} x_{jk} \le 1$$

$$x_{jk} = 0,\ 1 \text{ for all } j,k$$

여기서, $w_{jk} = \max(0, a_j - (C' -$ 작업장 k 에 할당된 고정,유동작업의 작업시간의 합))
(MP) 는 min max matching 문제이다. 목적함수는 유동작업 j 를 작업장 k 에 할당할 때 증가되는 cycle time 의 값을 최소화하는 것이다. 그래프이론의 정리에 의해 (MP)는 항상 가능해를 갖는다. ([1])

성질 2. (MP) 는 항상 가능해를 갖는다.

정리하면, (문제 1)의 해법은 먼저 (TP) 를 풀어 가능하한을 구하고, 여유변수에 대해 선회연산을 수행한 후 min max matching 문제를 풀어서 최종적인 할당을 정하는 것이다.

3.2 (문제 2) 의 해법

(문제 2)를 풀기 위해 다음과 같은 그래프 $G = (V, A)$를 정의 한다. 각각의 마디는 두개의 첨자로 구성된다. 즉, 임의의 마디는 [i, j]로 표현되며, i 는 고정작업, j 는 유동작업의 첨자이다. 만일 두 개의 마디 [i_1,j_1] 과 [i_2,j_2] 사이에 호가 있다면 이는 한 작업장에서 고정작업 i_1 과 유동작업 j_1 까지 수행되었을 때 그 다음 작업장에서 고정작업 i_2, 유동작업 j_2 까지 주어진 cycle time 내에 수행할 수 있음을 의미한다. 따라서, G 에서 마디 [0,0] 에서 마디 [m,n] 까지의 최단경로를 구하면, 주어진 cycle time 내에서 작업장수를 최소화하는 할당을 구할 수 있게 된다. Cycle time 에 대하여, 하한, 상한을 구한 후 2 분탐색기법을 활용하면, 주어진 작업장수하에서 cycle time 을 최소화하는 할당을 구할 수 있다.

3.3 전체 해법절차

전체해법절차는 다음과 같이 구성된다.

(단계 0) 초기해를 구한다.

(단계 1) 다음을 해의 개선이 없을 때까지 반복한다.

(1) (문제 1)을 푼다.

(2) (문제 2)를 푼다.

(단계 2) 사후 처리(postprocessing) 을 수행한다.

(단계 1) 에서 처음 (문제 1)은 초기해의 결과에서 고정작업의 할당이 정해지므로 이를 입력으로 사용한다. 다음에는 (문제 2)의 결과를 입력으로 사용하면 된다. (문제 2)는 (문제 1) 에서 유동작업의 할당이 주어지면, 이로부터 가능순서를 구하여 입력으로 사용한다.

초기해는 bin packing 문제의 알고리듬을 이용하여 구성할 수 있으며, 사후 처리 절차는 각 작업장의 여유시간을 고려하여 작업부하를 평탄화하는 것이다.

4. 결론

앞에서 제시된 해법을 세탁기조립라인에서 발생하는 실제문제에 대하여 적용시켜 본 결과는 다음과 같다.

[표 1] 실제문제 적용결과

문제번호	작업장수	cycle time	cycle time 하한	상대오차
# 1	37	33	33	0 %
# 2	37	34	33	3 %
# 3	37	30.7	30	2.3 %

위의 표에서 cycle time 의 하한은 작업시간중 최대값과 전체작업시간을 작업장수로 나눈 것 중에서 큰 값을 나타낸다. 상대오차는 하한에 대해 실제 구한 cycle time 의 차이를 백분율로 나타낸 것이다. 실험결과는 제한된 것이기는 하지만 일단 어느정도 좋은 성능을 나타낸다고 볼 수 있다.

본 논문에서는 가전제품조립공장에서 발생하는 작업장균형문제를 제시하고 그에 대한 해법을 제안하였다. 해법절차는 원문제를 두 개의 딸림문제로 분해하고 이들을 이용하여 해를 구한다. 두 개의 딸림문제는 각각 최단경로문제와 수송문제의 해법을 응용하여 해결되었다. Pre-test 결과 제안된 해법은 좋은 성능을 보이는 것으로 나타났다.

참고문헌

[1] Bondy, J.R., Murty, U.S.R., *Graph Theory with Applications*, Elsevier, (1976)

[2] Helgeson, W.P., D.P. Birnie, *Assembly Line Balancing Using the Ranked Positional Weight Technique*, J. of I.E., vol 12, no 6, (1961) pp 396 - 398

[3] Jackson, J.R., *A Computing Procedure for a Line Balancing Problem*, Management Science, vol 2, no 3, (1956) pp 261 - 271

[4] Johnson, R.V., *Optimally Balancing Large Assembly Line with 'FABLE'*, Management Science, vol 34, no 2, (1988), pp 240 - 253

[5] Murty, K.G., *Linear Programming*, John Wiley & Sons, (1983)

[6] Murty, K.G., *Linear Complementarity, Linear and Nonlinear Programming*, Heldermann Verlag Berlin, (1988)

[7] Wee, T.S. , M.J. Magazine, *Assembly Line Balancing as Generalized Bin Packing*, O.R. Letters, vol 1,no 2, (1981), pp 56 - 58