Journal of the Society of Korea Industrial and Systems Engineering
Vol. 34, No. 4, pp.82－88, December 2011.

복수의 부품 및 조립 흐름공정의 총흐름시간 최소화

문기주† · 이재욱

동아대학교 산업경영공학과

Minimizing Total Flow Time for Multiple Parts and Assembly Flow Shop

Gee Ju Moon† · Jae Wook Lee

Dept. of Industrial and Management Systems Engineering, Dong-A University

A typical job sequencing problem is studied in this research to improve productivities in manufacturing companies. The problem consists of two-stage parts and assembly processes. Two parts are provided independently each other and then two sequential assembly processes are followed. A new heuristic is developed to solve the new type of sequencing problem. Initial solution is developed in the first stage and then the initial solution is improved in the second stage. In the first stage, a longer part manufacturing time for each job is selected between two, and then a sequence is determined by descending order of the times. This initial sequence is compared with Johnson's sequence obtained from 2-machine assembly times. Any mismatches are tried to switch as one possible alternative and completion time is calculated to determine whether to accept the new sequence or not to replace the current sequence. Searching process stops if no more improvement can be made.

Keywords : Flow Shop, Heuristic, Makespan, Job Sequencing

1. 서 론

본 연구는 부품의 생산과 조립으로 이루어진 작업장의 효율을 높이기 위해 어떤 순서로 작업물을 처리할 것인가를 결정하는 작업순서결정(job sequencing)문제를 다룬 것이다. 대상 문제는 두 단계의 작업으로 구성되어 있는데, 1단계는 각 부품들을 상호독립 병렬라인으로 생산하며, 2단계는 1단계에서 만들어진 부품들을 직렬라인으로 조립하는 공정으로 되어있다. 이러한 부품생산기계와 조립기계로 이루어진 문제를 2단계 조립 일정계획 문제(2AFS : 2-Stage Assembly Flow shop Scheduling Problem)라고 하며 Lee et al.[8]의 연구에서 제시되었다. 현실의 예로 소방차의 조립문제를 들 수 있는데 차체와 엔진을 동시 별개의 생산라인에서 생산한 뒤 엔진과 함께 조립하여 완성하는 경우이다. 한 대의 소방차를 조립하기 위해 차체와 엔진은 동시에 생산이 가능하나 조립공정은 차체와 엔진 생산이 모두 완료되어야 시작할 수 있다. 이와 같은 작업장의 경우 부품 생산공정은 병렬구조이므로 유휴시간 없이 진행이 되나, 조립공정은 조립을 위한 해당 부품들의 생산이 모두 완료되어야 시작 가능하며 1차 조립이 끝나야 2차조립을 할 수 있으므로 부품생산과 1차조립공정 상황에 제약

논문접수일 : 2011년 10월 10일　　게재확정일 : 2011년 10월 21일
† 교신저자 gjmoon@donga.ac.kr
※ 이 논문은 동아대학교 교내연구비 지원에 의하여 연구되었음.

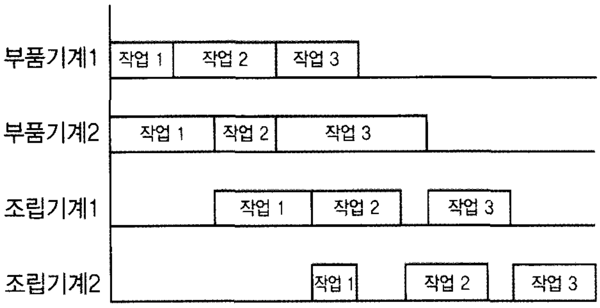
을 받는다.

Lee et al.[8]은 최대작업완료시간(maximum complete time)의 최소화를 목적함수로 하였고 본 문제가 NP-complete임을 증명하였다. 이후 Potts et al.[10]은 이전 2개의 부품 개수를 여러 개로 확장한 문제에 대해 발견적 해법을 제시하고 Hariri and Potts[5]는 분지한계알고리즘(Branch and Bound Algorithm)을 제시하였다. Sun et al. [11]은 Lee et al.[8]과 동일한 문제에서 보다 더 효과적인 발견적해법을 제시하였다. Kovalyov et al.[7]은 Batch Size 결정을 추가한 문제에 대해 연구하였고 Lin et al. [9]은 동일한 문제에서 2단계의 조립기계가 일괄조립(batch assembly)을 수행하는 상황을 가정하여 해법을 제시하였다. Juhn et al.[6]과 전재호[4]의 연구에서는 이전의 적시조달시간 제약을 개선하여 외주조달 상황을 가정한 뒤 시간제약을 함께 고려하여 문제의 난이도가 NP-complete임을 규명하고 발견적 해법을 제시하였다. 이익선 등[3]의 연구에서는 2AFS 문제에서 총 작업완료시간(total complete time)을 목적함수로 하고 혼합정수계획모형과 발견적해법을 제시하였다.

이상에서 기술한 연구들은 모두 부품 생산 후 조립공정이 1차로 제한된 것이다. 그러나 자동차 공장 등 일반적 생산현장에서 조립이 1차로 완료되는 경우는 거의 없다. 엔진조립, 변속기조립 등 다수의 후속 조립 공정을 거치게 된다. 그러므로 조립공정수가 복수이며 순차적인 2AFS에 대한 해법의 개발이 필요하다 하겠다.

2. 2단계조립 일정계획 문제

본 연구에서는 조립공정이 복수이며 순차적인 2AFS 문제를 다룬다. 부품생산은 병렬구조로 동시에 별도의 라인에서 생산이 가능하다. 복수의 부품 중 최대 부품 생산시간을 가지는 공정이 완료되어야 조립공정으로 넘어갈 수 있게 된다. 그러나 후속 조립공정은 직렬구조를 가지므로 1차 조립이 끝나야만 2차 조립이 시작 가능하다. 이러한 작업의 예시를 간트차트 형식으로 표시하면 <그림 1>과 같다. 조립기계1에서의 유휴시간은 그대로지만 1차 조립이 완료되어야 2차 조립이 시작되므로 2차 조립공정에서 유휴시간이 더욱 크게 발생하는 것을 볼 수 있다. 총소요시간은 작업순서의 마지막으로 배정된 작업의 2차 조립완료시간을 의미한다. 목적함수는 총소요시간의 최소화이므로 각 작업간의 유휴시간을 최소화하는 작업순서의 결정이 중요하게 된다. 본 연구의 대상인 조립공정이 복수인 2AFS 문제는 다음의 가정을 가지며 일반적인 순서계획문제와 같이 목적함수를 완료시간의 최소화[1]로 한다.



<그림 1> 복수부품 2단계 순차 조립 2AFS 예시

① 모든 작업들은 시작시점에서 처리 가능한 상태이다.
② 각 기계에서의 처리시간은 확정적이다.
③ 각 기계에서의 작업은 시작 후 완료까지 연속적이다.
④ 각 기계는 한 번에 하나의 작업만 생산이나 조립이 가능하다.
⑤ 작업준비시간(setup time)은 각 작업시간에 포함되어 있다.
⑥ 공정간 운반시간은 고려하지 않는다.
⑦ 부품의 생산순서가 조립기계에서도 유지되어야 한다.
⑧ 2개의 병렬부품공정과 2개의 순차적 조립공정으로 구성되어 있다.

3. 해법의 설계

3.1 초기해의 생성

본 연구에 관련된 수리모형과 기호는 문헌상의 2AFS 연구[2, 3, 4]를 참조 및 수정 인용하였다.

n	작업수
m	각 작업의 부품 종류의 수. 2가지만 현모형에서 존재하므로 $m \in \{1, 2\}$
j	작업 $j(j \in \{1, \cdots, n\})$를 표시하는 첨자
C_j	작업 j의 조립이 완료되는 시점
$a_{j,m}$	1단계 부품생산기계에서 작업 j에 사용되는 부품 생산시간
$b_{j,m}$	2단계 조립기계에서 작업 j에 사용되는 조립시간
π	임의 일정계획의 해. $\pi = ((\pi(1), \cdots, \pi(n)),$ 여기서 $\pi(i)$는 일정계획해 π의 i번째 작업
π^*	최적일정계획 해
$C_{\max}^{\pi}$	임의 일정계획의 해 π의 최대작업완료시간

목적함수는

Minimize $C_{\max}^{\pi}$

이익선 등[3]은 총 작업완료시간을 목적함수로 하여 혼합정수계획모형을 만들었다. 그러나 규모가 큰 혼합정수계획모형의 최적해를 적정시간 내에 구하는 효율적인 해법은 존재하지 않으므로 발견적 해법을 제시하였다.

본 해법은 초기해를 생성하는 1단계와 그 해를 개선하는 탐색으로 이루어진 2단계로 구분할 수 있다. 초기해는 부품생산시간 오름차순, 조립시간 내림차순, 그리고 부품생산시간과 조립시간 합 내림차순의 성능을 평가하여 우수한 결과를 생성하는 것으로 하였다. 부품생산시간 오름차순은 각 작업에 대해 $\max(a_{j,1}, a_{j,2})$ 값을 구한 뒤 이 값을 기준으로 오름차순으로 정렬하는 것이다. 조립시간 내림차순은 각 작업에 대해 $c_j = (b_{j,1} + b_{j,2})$를 계산한 뒤 그 값을 기준으로 내림차순으로 정렬한 것이다. 부품생산시간과 조립시간 합 내림차순은 각 작업에 대해 $d_j = \max(a_{j,1}, a_{j,2}) + c_j$를 구하여 내림차순으로 정렬한 것이다.

이 3가지 초기해 생성 방안의 수행도를 평가하기 위해 작업 수 20개, 각 작업 시간치$(a_{j,1}, a_{j,2}, b_{j,1}, b_{j,2})$는 1~999까지의 정수를 임의로 생성하고 30개의 문제를 만들었다. 3가지 방안에 추가하여 작업순서를 임의로 생성하는 방안도 함께 평가하였다. 그 결과 최소값의 개수가 임의 순서는 2개, 부품생산시간 오름차순은 8개, 조립시간 내림차순은 15개 그리고 부품생산시간과 조립시간 합 내림차순이 5개가 나왔다. 평균값 역시 조립시간 내림차순이 제일 작은 값을 나타내었다. 따라서 조립시간 내림차순이 가장 좋은 수행도를 보이므로 초기해 생성에 사용하였다. 물론 초기해의 선정과 개선으로 이루어진 발견적 해법에서 초기해의 성능이 우수하다고 해서 우수한 최종해가 구해진다고는 할 수 없다. 그러나 초기해를 좋은 값으로 선정 할 때 보다 나은 개선해가 구해질 가능성과 탐색 시 계산과정을 줄일 수 있는 가능성을 위해 이 방안을 채택하였다.

3.2 개선해의 탐색

2단계는 1단계에서 구한 초기해를 개선하는 과정으로 구성되어 있다. 이 개선은 다음과 같은 순서를 따른다.

① 초기해의 작업순서를 배열 A라 둔다.
② 각 작업을 $b_{j,1}, b_{j,2}$를 기준으로 존슨(Johnson)규칙에 따라 작업순서를 결정하고 이를 배열 B라 둔다.
③ <그림 2>의 탐색 순서도에 따라 배열 A를 교체.
④ 종료 후 배열 A를 최종 근사해를 선정한다.

탐색 순서는 우선 배열 A에서 A_1과 A_2의 순서를 배열 B와 비교한다. 예를 들어 작업의 수는 5개이고 A = [1 3 5 2 4], B = [5 3 2 4 1] 일 때 처음 선택되는 작업은 A_1 = 1, A_2 = 3이다. 따라서 작업 1의 완료 후 작업 3이 시작하는 순서를 따른다. 그러나 배열 B를 보면 작업 1이 작업 3보다 뒤에 있으므로 작업 3의 완료 후 작업 1이 시작되는 것을 알 수 있다. 이것을 순서가 다르다고 표현하며 순서도에서 N라인을 따르게 된다. 그 뒤 순서가 다른 A_1과 A_2를 교체한 작업순서 [3-1-5-2-4]를 배열 A'로 둔 뒤 총소요시간을 구한다. 만약 A`의 총소요시간이 A보다 작게 되면 A'가 새로운 A가 되며 다시 A_1 = 3, A_2 = 1이 되며 배열 B와 비교하게 된다. 이번에는 순서가 같으므로 총소요시간을 구하지 않고 k가 작업 수(n = 5)와 같은지 비교한다. 아니면 k에 1을 더해 A_2와 A_3를 비교하게 된다. 즉 위의 단계에서는 최종적으로 A_4, A_5까지 순서를 비교한 뒤 개선을 탐색하고 종료한다.

다음 단계는 배열 B와의 순서비교는 같지만 배열 A의 선택패턴이 바뀌게 된다. 즉 이전 단계에선 A_1과 A_2를 비교한 뒤, A_2와 A_3를 비교하였지만 이번 단계에선 A_1과 A_3를 비교하게 된다. 역시 개선이 이루어지면 A`는 A가 되어 다시 1단계부터 탐색을 시작한다. 다른 순서가 없거나 A_4, A_5까지 개선이 되지 않으면 종료한다.

본 해법은 교체를 반복하여 최종해를 도출하는 방식이므로 반복수를 줄이는 것이 해법의 성능에 큰 영향을 미친다. 예를 들어 작업 수(n) 5개를 기준으로 A_4와 A_5의 교체는 A_α와 A_β를 교체하는 규칙만을 사용하면 10회의 교체 반복을 필요로 한다. 하지만 A_k와 A_{k+1}을 추가하는 규칙을 적용하면 4회 반복으로 종료할 수 있다. A_α와 A_β에서 α와 β의 차이가 큰 경우 A_k와 A_{k+1}을 교체하는 규칙의 추가로 탐색 횟수가 더 증가할 수 있다. 하지만 다수의 예시 문제에 적용한 결과 이를 감안하더라도 결론적으로 탐색을 줄이는 경우가 더 많이 관측되었다. 따라서 이상의 규칙을 추가하여 더 효율적인 탐색을 가능하게 하였다.

4. 수치실험 및 결과 분석

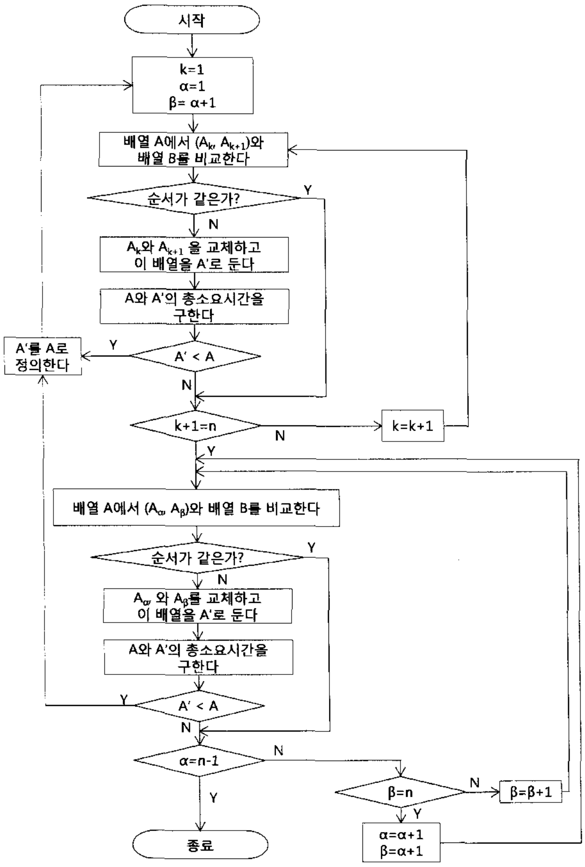
4.1 실험의 설계

일반적으로 개발된 발견적해법의 성능 평가는 기존 연구에서 제시된 해법과 동일한 문제를 풀어 그 결과의 비교로 이루어진다. 그러나 이전 연구의 2AFS 문제에서 1차로 종료되던 작업을 본 연구에서 2차로 구성

되어있으므로 기존 해법을 그대로 적용하여 평가하는 방법은 가능하지 않다. 그러므로 본 연구에서 제안하는 발견적 해법의 평가를 다음과 같이 2가지 유형의 실험으로 수행하였다.

제 1유형 : 작업 수(n)를 5개, 부품생산시간(a)과 조립시간(b)은 1부터 999사이의 정수로 임의 선정하였다. 이 문제의 모든 가능한 작업순서의 개수는 5!이다. 따라서 120개의 작업순서를 완전열거법으로 최적해를 구한 뒤 본 연구에서 제안하는 발견적 해법으로 구한 최종해와 비교하여 성능을 평가하였다.

제 2유형 : 작업 수(n)를 20개, 부품생산시간(a)과 조립시간(b)은 1부터 999사이의 정수로 임의 선정하였다. $20! = 2.4329 \times 10^{18}$이므로 모든 작업순서를 평가하는 것은 현실적이지 않다. 따라서 총 10문제를 만들고 한 문제당 100개와 1000개의 임의 작업순서를 만들어 본 연구에서 제안한 발견적해법의 최종해와 비교하여 성능을 평가하였다.



<그림 2> 제안해법의 흐름도

작업의 개수가 많을수록 한 작업순서의 총소요시간을 산출하기 위한 시간은 늘어나게 된다. 그러므로 효율적인 해법은 최종해가 최적해에 얼마나 근접한가도 중요하지만 최종해를 도출하는 시간을 줄이는 것도 중요하다고 할 수 있다. 제안한 해법으로 최종해를 도출할 때 한 문제당 몇 회의 총소요시간에 대한 계산이 이루어졌는지를 같이 평가하였다.

제안 해법의 평가는 IBM PC 호환 Intel Core 2 Duo 2.93GHz, 2GB RAM의 컴퓨터를 사용하여 수행하였다. 계산프로그램은 Microsoft Office Excel 2007과 Microsoft Visual Basic을 사용하였다. 난수 발생은 Excel의 RAND 함수를 사용하였으며 주로 임의순서로 작업 시 반복 계산은 Excel의 매크로 기능을 이용하여 처리하였다. 해법을 설계할 때 Excel의 가시적인 데이터 처리가 직관성을 높여 주어 많은 도움을 주었다. 결과 데이터의 평가 및 정리는 Minitab Ver. 15와 ExpertFit Ver. 8.00(beta) 프로그램을 사용하였다.

4.2 실험 및 결과

제 1 유형 실험 결과

제 1 유형 실험에 사용된 자료는 <표 1>과 같다.

<표 1> 사례연구 1

	작업 1	작업 2	작업 3	작업 4	작업 5
부품생산 1(a_j, 1)	75	602	765	597	212
부품생산 2(a_j, 2)	315	345	180	511	67
조립 1(b_j, 1)	37	332	751	183	371
조립 2(b_j, 2)	661	276	566	483	816

5개의 작업에 대해 2개의 부품 생산시간과 1차, 2차 조립시간치를 1~999사이의 정수로 임의 생성하였다. 제안하는 발견적 해법을 적용한 단계적 예시는 다음과 같다.

우선 Johnson's rule에 의한 작업순서는 [1-4-2-5-3]이며 이것은 배열 B로 된다. 초기해는 [3-5-1-4-2]이며 배열 A로 된다. 이상을 <그림 2>의 개선해 탐색 순서도에 대입하면 <표 2>와 같은 결과가 나온다.

좌측의 초기해 = A는 초기해 선정으로 만들어진 배열 A를 뜻하며 하단의 A는 A'가 개선이 된 경우 새로운 A가 선정되었음을 뜻한다. 우측 반복(회)는 탐색이 완료되기까지 총소요시간을 계산한 횟수를 의미한다. 반복 9회에 탐색이 최종 완료 되었고 최종해를 나타내는 최종 배열 A는 [1-5-3-4-2]임을 알 수 있다. 따라서 최종해의 총소요시간은 3154로서 완전열거법으로 구한 2개의 최적해중 하나와 동일함을 알 수 있다.

120개의 작업순서를 가지는 문제를 9회 계산으로 최적해와 동일한 해를 구하였으므로 본 문제에 대한 제안 해법은 유효하다고 할 수 있겠다.

<표 2> 사례 1에서의 최종해

	작업순서					완료시간	반복회수
A	3	5	1	4	2	4318	1
A	5	3	1	4	2	3714	2
A	5	1	3	4	2	3385	3
A	**1**	**5**	**3**	**4**	**2**	**3154**	4
	1	5	4	3	2	3242	5
	1	4	3	5	2	3846	6
	1	2	3	4	5	4058	7
	1	5	4	3	2	3242	8
	1	5	2	4	3	3568	9

제 2 유형 실험 결과

제 2 유형의 실험의 시간치는 한 문제당 작업 수(n)는 20개로 되어 있으며 각 작업에 사용되는 4개의 시간치(부품생산 1($a_{j,1}$), 부품생산 2($a_{j,2}$), 조립 1($b_{j,1}$), 조립 2($b_{j,2}$))가 순서대로 아래 <표 3>과 같이 표기되어 있다. 실험은 총 10개의 문제로 구성했으며, 사용한 시간치들의 예시는 <표 3>과 같다.

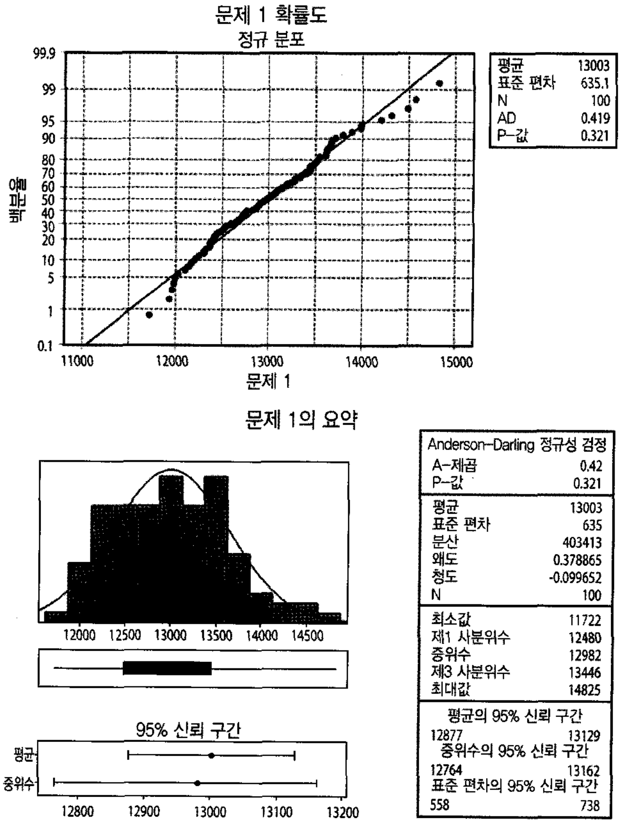
<표 3> 사례연구 2

작업물 / 문제	1	2	…	20
1	971	456	…	632
	151	803		742
	15	784		659
	555	464		864
2	770	307	…	934
	111	131		719
	349	455		914
	29	169		558
⋮	⋮	⋮	⋱	⋮
10	493	225	…	81
	245	549		910
	152	253		383
	972	749		812

20개의 작업순서는 1부터 20까지의 배열로 정의할 수 있다. 이 배열을 Excel로 난수를 발생시켜 정렬하는 방식으로 무작위 생성시켰다. 각 문제당 100개의 무작위 작업순서에 대해 총소요시간을 계산하고 그 데이터들의 분포와 본 연구에서 제안한 해법의 최종해와 비교하였다. <표 3> 사례연구 2의 문제 1에 대한 요약이 <그림 3>에 나와 있다.

임의생성한 100개 순서들의 총소요시간은 정규분포와 유사한 형태를 보인다, 평균은 13003.1(95% 신뢰구간은 12877~13129)이다. 표준편차는 635(95% 신뢰구간은 558~738)이다. 유의수준 0.05을 사용한 Ander son-Darling 정규성검정(A-제곱 = 0.419, P = 0.321)은 문제 1이 정규분포를 따른다는 것을 나타낸다.

본 연구에서 제안하는 발견적 해법으로 최종해를 구한 결과와의 비교는 <표 4>에서 볼 수 있다. 좌측에는 한 문제당 무작위로 생성한 100개의 총소요시간을 바탕으로 평균과 표준편차 그리고 최소값을 나타내었다. 우측에는 본 연구에서 제안하는 해법으로 문제를 푼 결과의 최종해를 나타내었다. 그리고 임의생성된 순서의 최소값과 비교하여 최종해/최소값의 비율을 $\frac{\text{최종해}}{\text{최소값}}$(%)로 나타내었고 평균값과 비교하여 평균값/최소값의 비율을 $\frac{\text{최종해}}{\text{최소값}}$(%)로 나타내었다. 반복회수는 본 해법에서 탐색과정을 거치면서 총소요시간의 계산을 몇 회 수행하였는가를 나타내고 있다.



<그림 3> 문제 1의 결과 요약

<표 4> 사례연구 2의 결과(100개의 임의생성 작업순서 평가)

사례	100개의 임의생성된 작업순서			제안해법			
	평균	표준편차	최소값	최종해	최종해/최소값 (%)	최종해/평균값 (%)	반복회수
1	13002.96	635.15	11722	10951	93.42	84.22	216
2	11163.18	547.52	10204	9555	93.64	85.59	204
3	13824.84	618.43	12551	12329	98.23	89.18	167
4	14444.59	595.65	13402	12838	95.79	88.88	310
5	13359.91	547.24	12132	11339	93.46	84.87	126
6	13372.67	696.26	11839	11839	100.00	88.53	96
7	13682.57	548.88	12499	11286	90.30	82.48	173
8	12408.41	665.79	10841	10657	98.30	85.89	177
9	14280.42	517.08	13072	12400	94.86	86.83	313
10	12707.37	604.74	11465	10724	93.54	84.39	177
평균	13224.69	597.67	11972.7	11391.8	95.15	86.09	195.9

예를 들어 5번 문제의 경우 임의로 100회의 작업순서를 구한 경우 그 최소값은 12132이며 평균은 약 13359.91, 표준편차는 547.24이다. 이 경우 p값이 0.118로 정규분포를 따르고 있음을 알 수 있다. 여기에서 본 연구에서 제안한 해법으로 구한 최종해가 11339인데, 이 값이 임의생성한 정규분포를 따르는 모집단에서 어느 수준의 값인지를 평가해 보기 위해 Z값을 구하면 -3.69임을 알 수 있다. 따라서 본 해법의 최종해보다 작은 해가 존재할 확률은 0.011% 이하라고 해석할 수 있다. 여기에서 Z값은 (최종해-평균값)/표준편차로 구한 것이다.

본 연구의 해법이 찾은 최종해는 임의생성된 해들의 최소값 대비 6.54% 적은 값이 보였으며, 평균값 대비해서는 15.13%의 총소요시간 감소를 나타내었다. 반복회수는 126회로 나타났는데, 이는 최종해를 구하기까지 해의 개선과정에서 총소요시간 계산을 126회 반복 수행하였다는 것을 뜻한다.

10개의 문제에 대한 최종해의 평균은 $Z \leq -3.11$의 값이 나왔고 이는 최종해보다 작은 해가 존재할 확률이 0.09% 이하임을 뜻하므로 본 연구에서 제시한 해법이 유효하다고 하겠다. 좀 더 정확한 평가를 위해 임의생성한 작업순서를 1000개로 증가시켜 비교해 보았다. <표 5>는 위와 동일한 문제에 대해 임의작업순서를 1000개로 확대하여 본 연구 해법의 최종해와 비교한 것이다.

임의 생성한 작업순서의 개수를 1000개로 앞 보다 10배 증가시켜 실험하였으나 개선의 폭은 미비하였다. 결론적으로 본 해법으로 구한 최종해보다 작은 해는 발견되지 않았다. 1000회의 실험에서 구한 최소값들의 평균은 11712.5이며 본 해법의 최종해 평균은 11391.8이다. 본 연구의 해가 2.74% 작은 값을 찾았다. 감소의 폭이 다소 미미하나 도출한 최종해는 모두 비교대상보다 작은 값이며 평균 반복수 195.9회 만으로 1000회의 임의생성한 순서보다 작은 값을 탐색하였으므로 본 연구에서 제시한 해법은 유효하다고 하겠다.

<표 5> 사례연구 2의 결과(1000개의 임의생성 작업순서 평가)

사례	1000개 임의순서		제안해법		
	평균	최소	최종해	최종해/최소값 (%)	최종해/평균값 (%)
1	12986	11512	10951	95.13	84.33
2	11163	9843	9555	97.07	85.60
3	13904	12378	12329	99.60	88.67
4	14485	13014	12838	98.65	88.63
5	13310	11915	11339	95.17	85.19
6	13381	11839	11839	100.0	88.48
7	13717	12174	11286	92.71	82.28
8	12421	10841	10657	98.30	85.80
9	14279	12696	12400	97.67	86.84
10	12635	10913	10724	98.27	84.88
평균	13228	11712	11391	97.26	86.07

5. 결 론

본 연구에서는 부품을 생산하는 1단계와 만들어진 부품을 복수의 순차 조립하는 2단계 공정으로 이루어진 일정계획문제(2-Stage Assembly flow shop Scheduling Problem; 2AFS)를 다루었다. 제시된 해법의 성능을 평가하기 위해 작업의 수가 작은 모형과 큰 모형 2가지의 조건을 적용하였다. 작업의 개수가 작은 모형에서는 완전열거

법으로 최적해를 구하여 해법의 최종해와 동일함을 보였다. 작업의 개수가 큰 모형에서는 작업순서를 임의로 100개를 생성하여 이 100개 해의 집단이 정규분포를 따르고 있음을 보이고, 본 해법의 최종해가 임의생성한 평균값과 표준편차의 모집단에서 최종해보다 작은해가 존재할 확률이 평균 0.09%임을 보였다.

1000개의 임의 작업순서를 생성한 결과는 총 10개의 문제를 대상으로 한 결과 9개의 문제에서 1000개의 해보다 우수함을 보였고 1개의 문제에서 해의 결과가 동일하였다. 본 해법의 최종해와 임의 생성한 해의 최소값과의 차이가 크지는 않았지만 20!개의 해를 가지는 문제를 195.9회의 개선탐색 반복 계산으로 최종해를 도출하였으므로 그 효율성이 높다고 할 수 있겠다.

참고문헌

[1] 문기주 옮김, Dileep R. Sule; "일정계획이론", 시그마프레스, 2002.

[2] 윤상흠, 전재호; "부품외주를 고려한 조립형 Flowshop 일정계획 해법 개선", *Journal of the Society of Korea Industrial and Systems Engineering*, 31(2) : 80-93, 2008.

[3] 이익선, 윤상흠, 하귀룡, 전재호; "부품 생산과 조립으로 구성된 2단계 조립 일정계획의 Flowtime 최소화 연구", *Journal of the Society of Korea Industrial and Systems Engineering*, 33(4) : 45-57, 2010.

[4] 전재호; "부품외주를 고려한 조립형 Flowshop 일정계획문제 연구", *Journal of the Society of Korea Industrial and Systems Engineering*, 29(4) : 34-42, 2006.

[5] Hariri, A. M. A. and Potts, C. N.; "A branch and bound algorithm for the two-stage assembly scheduling problem," *European Journal of Operational Research*, 103 : 547-556, 1997.

[6] Juhn, J., Sung, C. S., and Yoon, S. H.; "Analysis of heuristics for a two-stage assembly scheduling problem with component available time constraint," *Submitted to Operations Research Letters for publication*, 2006.

[7] Kovalyov, M. Y., Potts, C. N., and Strusevich, V. A.; "Batching decisions for assembly production systems," *European Journal of Operational Research*, 157 : 620-642, 2004.

[8] Lee, C. Y., Cheng, T. C. E., and Lin, B. M. T.; "Minimizing the makespan in the 3-machine assembly-type flowshop scheduling problem," *Management Science*, 39 : 616-625, 1993.

[9] Lin, B. M. T., Cheng, T. C. E., and Chou, A. S. C.; "Scheduling in an assembly-type production chain with batch transfer," *OMEGA : The International Journal of Management Science*, 35(2) : 143-151, 2007.

[10] Potts, C. N., Sevas'janov, S. V., Strusevich, V. A., Wassenhove, L. N. V., and Zwaneveld, C. M.; "The two-stage assembly scheduling problem : Complexity and approximation," *Operation Research*, 43 : 346-355, 1995.

[11] Sun, X., Morizawa, K., and Nagasawa, H.; "Powerful heuristics to minimize makespan in fixed, 3-machine, assembly-type flowshop scheduling," *European Journal of Operational Research*, 146 : 498-516, 2003.