

☒ 응용논문

퍼지 조립라인밸런싱 문제 해결을 위한 주노드법에
기초한 휴리스틱 절차 개발

-The development of critical node method based heuristic procedure for Solving
fuzzy assembly-line balancing problem-

이 상 완*

Lee, Sang Wan

박 병 주**

Park, Byung Joo

Abstract

Assembly line balancing problem is known as one of difficult combinatorial optimization problems. This problem has been solved with linear programming, dynamic programming approaches. but unfortunately these approaches do not lead to efficient algorithms. Recently, genetic algorithm has been recognized as an efficient procedure for solving hard combinatorial optimization problems, but has a defect that requires long-run time and computational complexities to find the solution. For this reason, we adapt a new method called the Critical Node Method that is intuitive, easy to understand, simple for implementation. Fuzzy set theory is frequently used to represent uncertainty of information.

In this paper, to treat the data of real world problems we use a fuzzy number to represent the duration and Critical Node Method based heuristic procedure is developed for solving fuzzy assembly line balancing problem

1. 서론

라인 밸런싱 문제는 75년전 포드 시스템의 출현 이후 제조 업체에서 사용된 역사적인 관리 기술이라 할 수 있다. 대량 생산에 적용되어졌고, 실제 이 기술은 고용 생산을 개선하는 중요한 도구로서의 역할을 했다. 라인 밸런싱에 대한 최초의 접근법은 Salveson[1]에 의해 연구되었고, 다른 중요 연구로는 Baybars[2], Talbot[3], Johnson[4], McCahon[5] 등이 있고, 최근에는 유전자 알고리즘을 이용한 접근법[6]들이 개발되어지고 있다.

조립라인 밸런싱 문제(Assembly-line Balancing Problem :ABP)는 무수히 많은 수의 실행가능해들을 가진다. 전체의 열거가 불가능할 정도이다. 이 문제는 정수계획 문제로 정식화될 수 있지만 불행히도 이 접근법은 NP-hard 최적화 문제인 이 문제의 효과적인 알고리즘은 되지 못했다. 이 때문에 휴리스틱 의사결정 규칙들에 관한 연구를 하였고 대부분의 연구 노력들은 이들 휴리스틱 기법의 효율성을 개선하는데로 확장되어졌다.

최근 유전자 알고리즘은 NP-hard 문제를 해결하기 위한 강력한 대체 알고리즘으로 많이 사용되어지고 있다. 이는 진화과정에 기초한 휴리스틱 알고리즘으로 진화과정의 메카니즘을 사용해 염색체 구조로 해를 코드화 하고 그리고 진화과정에서 그들 염색체에 대해 단순한 연산자들을 반복 적용 할 수 있다. 큰 규모의 문제에서도 최적해에 근접한 해를 구할 수 있다는 장점이 있다. 그러나 이 접근법은 오랜 계산시간과 복잡성, 쉽게 이해하기 어려워 실제 적용시키기 어렵다는 단점을 가지고 있다. 그래서 보다 쉽게 이해하고, 실행하기에 편리하면서 좋은 해를 얻어낼 수 있는 접근법의

* 동아대학교 산업시스템공학과 교수

** 동아대학교 산업시스템공학과 박사과정 수료

개발의 필요성이 부각되고 있다. 그래서 보다 쉽게 적용할 수 있는 주노드 방법(critical node method:CNM)을 활용하고자 한다. 이는 네트워크 분석을 위한 한 도구로써 잘 알려진 주경로법(CPM)과 그래프 이론에 기초하고 있다. CNM은 CPM(Critical Path Method)의 직관적이고, 이해하기 쉽고, 실행의 단순함으로 넓은 응용분야를 가지고 있다는 장점을 유지하면서 CPM의 몇몇 결점을 보완한 새로운 기법이다.

현실세계 문제의 데이터는 불확실하다. 그리고 입력 데이터는 오직 불확실성 하에서 추정할 수 있다. 이 불확실성은 퍼지수로 나타내어질 수 있다. 또한, 퍼지 데이터의 구성함수 값은 의사결정자의 만족의 정도를 나타낸다. 퍼지수의 개념은 각 작업의 가공시간과 같은 불확실한 자료를 다루기 위해 도입되어진다.

본 연구에서는 주노드 방법이라 불리는 새로운 방법을 이용하여 현실세계의 불확실성을 반영한 퍼지 조립라인 밸런싱 문제를 해결하기 위한 휴리스틱 기법을 제안하고자 한다.

2. 조립라인 밸런싱의 네트워크 표현

ALB 문제의 네트워크 표현은 다른 여러 문제들에 대한 표현 보다 더욱 쉽게 표현될 수 있다. 그 네트워크는 선행 방향성 그래프(precedence diagram)로 불리워지고 실제로 활동을 마디에 나타낸 네트워크(activity-on-node network)와 같은 형태이다.

ALB 방향성 그래프는 50년대 중반에 개발되어졌고, activity-on-node 구조는 약 10년쯤 후에 제안되어졌다. 많은 연구자들이 여러 분야에 성공적으로 이 방법들을 도입했었다.

2.1 Arrow-Precedence Diagram(APD)

네트워크의 표현에 두 개의 유사한 표현 방법이 있는데 그 중 AOA(activity-on-arrow) 표현에서 모든 노드들은 사건을 표현하기 위해 번호가 매겨진다. 그리고 모든 화살표들은 작업(task)을 표현하기 위해 이름 또는 번호가 붙여진다. 덧붙여 각 화살표는 대응되는 작업들의 기간(duration)을 포함하고 있다.

AON(activity-on-node) 표현에서 모든 노드들은 작업을 표현하기 위해 번호가 부여되고 그리고 각 노드는 대응되는 작업들의 기간들을 포함한다. 이 구조에서 화살표는 작업들의 순서 관계만을 나타내기 때문에 번호는 사용되지 않는다.

APD(Arrow-Precedence Diagram)라 불리는 표현방식은 위의 두 구조에서 발전된 새로운 구조이다. 이 새 구조에서 네트워크들은 AON 표현과 유사한 구조를 가진다. 모든 노드들은 작업을 표현하기 위해 번호가 부여되고 작업 기간은 노드 대신 작업과 관련되는 노드에서 출발하는 화살표에 기입된다. 그러므로 그 화살표들은 논리적인 관련성 뿐만 아니라 활동의 기간까지 나타낸다.

이 APD는 크게 AON 구조와 다르지 않지만, 이 구조는 작업들 사이의 관련성을 더욱 직관적으로 만들 수 있고 그리고 더욱 중요한 것은 CNM의 기초가 된다. 더욱 일반적으로 APD의 모든 형태를 가지는 구조로서 arrow-and-node-precedence diagram을 정의 할 수 있다. 덧붙여, 그 노드들은 노드들의 능력(capacity)과 같은 값이 할당되어진다. 이 구조는 수송 문제나 네트워크 흐름 문제를 표현하기에 편리하다.

APD는 비순환 방향성 그래프(acyclic diagraph)이다. 같은 노드에서 출발하는 화살표들은 그 노드에 의해 표현되는 작업들의 기간과 같은 값을 가진다.

표 1. 작업의 목록

작업	기간	직접 후행 노드
1	8	4,5
2	15	5
3	12	6,7
4	27	10,11
5	6	11
6	11	8,9
7	11	9,12
8	6	11
9	18	11
10	16	
11	9	
12	6	

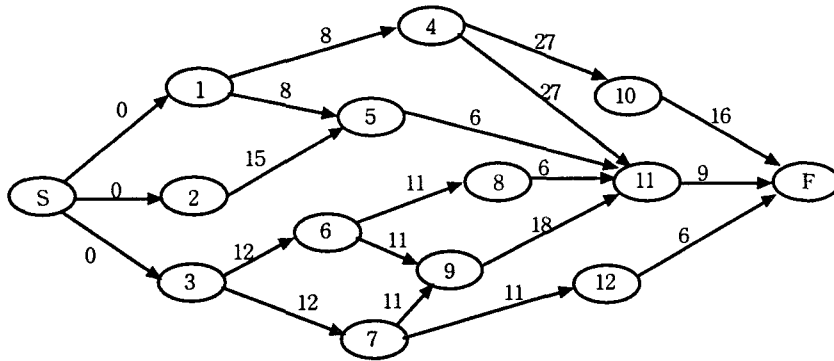


그림 1. Arrow-Precedence 표현

3. 주노드법(Critical Node Method)

APD 구조에서는 한 노드에서 출발하는 모든 화살표들은 같은 값의 가중치를 가진다. 이 형태는 더욱 효과적인 절차를 만들기 위해 사용되어진다. 그리고 다음의 정의들이 CNM을 나타낼 때 사용된다.

먼저 화살표의 가중치는 AOA구조와 APD 둘다 각 화살표들은 작업기간과 같은 값을 가지는데 이를 가중치로 고려한다. 경로의 가중치(Path of Weight)는 경로 상의 모든 화살표에 대한 가중치의 합이다. 주경로(Critical Path)는 한 쌍의 노드 사이의 모든 경로들 중 가중치가 가장 큰 경로를 말한다. 노드의 가중치는 한 노드에서 종점(sink) 노드까지의 가장 긴 경로의 가중치이고 주 노드(Critical Node)는 노드들의 집합에서 가장 큰 가중치를 가지는 한 노드를 말한다.

3.1 주노드법의 절차

먼저 AOA, AON 구조를 포함하는 비순환 방향성 그래프에 적용할 수 있는 기본적인 CNM 절차는 먼저 네트워크를 APD로 표현하여야 한다. 그리고 이 절차는 최장 경로 문제나 최단 경로 문제에도 쉽게 적용되어질 수 있다.

CNM 절차는 종점노드에서 출발노드까지 후진기법이 이용된다. 위에서 주어진 정의에서 종점노드는 가중치가 0이 될 것이다. 어떤 다른 노드에 있어서, 만약 모든 후행 노드들이 그들 가중치들에 따라 표시되어졌다면 이 노드의 가중치는 각 직접 후행 노드(direct successor)의 가중치와 연결 화살표의 가중치를 합해서 그 합들 중에서 최대인 값이 그 노드의 가중치가 된다. 만약 한 노드가 가중치가 없는 후행 노드를 가진다면, 그것은 즉각적으로 가중치를 부여할 수 없다. 그래서 항상 출발노드를 향하여 후진할때 오직 가중치를 가지는 후행 노드들을 가지고 가중치를 가지지 않은 노드를 찾기 위해 노력해야 한다. 이 과정은 출발노드가 가중치를 가질때 까지 되풀이 된다. 다음의 관계는 항상 참이 된다.

$$\text{노드 집합} = \text{가중치를 가진 부분집합} + \text{가중치를 가지지 않은 부분집합}$$

가중치를 가지지 않는 노드들의 집합은 특별한 관심을 가지게끔 한다. 가중치가 있는 후행노드를 가지는 가중치가 없는 노드들이 있다. 그들 가중치들은 절차에 따라 정확하게 할당할 수 있는데 그들을 가중치를 부여할 수 있는 노드들이라 부른다. 그들 개념들은 쉽게 전진 과정에서도 도입될 수 있다.

앞에서 쉽게 얘기 했던것 처럼, 비순환 방향성 그래프는 그래프 상의 모든 노드들은 번호가 부여될 수 있고 그것의 선행 노드들은 더욱 작은 숫자가 부여된다. 그리고 그것의 후행 노드들은 이 특정 노드보다 더욱 큰 수가 부여되어진다.

이 형태를 고려해서, 주 노드를 정하는 절차는 다음과 같이 순서적으로 네트워크에 숫자를 부여해 나가는 과정이라 할 수 있다.

단계 1) 종점(sink) 노드에는 0의 가중치를 할당한다.

단계 2) 가장 큰 노드 숫자를 가진 노드 중 가중치를 가지지 않은 노드를 고려한다. 그것의 가중치는 직접 후행 노드와 연결 화살표의 각 쌍으로 부터의 가중치의 가장 큰 합으로 할당되어진다.

단계 3) 단계 2에서 노드의 가중치가 결정된 노드와 이 노드에서 후행 노드의 연결 화살표를 점선 화살표로 바꾼다.

단계 4) 만약 어떤 가중치를 가지지 않은 노드가 있다면, 단계 2로 가고 그렇지 않으면 정지.

주 경로의 확인은 단계 3에서 이루어진다. 주 경로는 점선 화살표로 나타낸다. 이 절차가 수행된 후 적어도 하나 이상 점선 화살표가 각 노드에서 종점(sink) 노드까지 생기게 될 것이다. 위의 절차에서, 주어진 어떤 한 노드에 있어 그곳에는 이 노드를 출발하는 적어도 하나 이상의 점선 화살표가 있다는 것을 쉽게 확인할 수 있다. 시작노드에서 종점노드까지의 점선 화살표는 전통적인 주 경로이다.

이 단순한 절차에 따라, 전통적인 주 경로를 확인할 수 있었다. 덧붙여 모든 노드에서 주 경로를 가질 수 있고, 이 주경로의 길이는 경로의 출발노드의 가중치로 주어진다.

3.2 주노드법의 예

위의 절차를 예를 들어 설명하면, 먼저 단계 1에서 그림 2의 노드 6(sink)에 가중치 0을 할당한다. 단계 2에서 가중치를 가지지 않은 노드들 중 가장 큰 숫자를 가진 노드는 5이다. 노드 5는 하나의 후행 노드를 가지고 있기 때문에 비교는 필요하지 않다. 그리고 그것의 가중치는 $3=3+0$ 이 할당된다. 단계 3에서 노드 5에서 6까지의 화살표를 이중 화살표로 바꾼다. 그리고 다시 단계 2로 돌아가서 노드 4에 가중치 $8=8+0$ 를 할당한다.

이 시점에서, 단계 2는 가중치를 가지지 않은 노드들 중 가장 큰 수를 가진 노드 3을 선택한다. 노드 3은 두개 직접 후행 노드로, 노드 4와 노드 5를 가진다. 노드 5와의 연결의 합보다 큰 노드 4와 연결 화살표의 합이 $10=8+2$ 이기 때문에, 노드 3은 단계 3에서 가중치 10이 할당된다. 그리고 노드 4에서 연결 화살표는 단계 3에서 점선 화살표로 바뀌어진다. 유사하게, 노드 2와 1도 가중치를 계산할 수 있다.

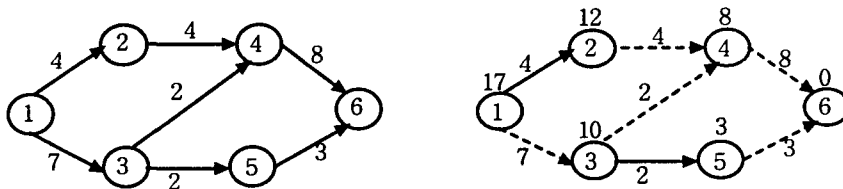


그림 2. 일반적인 (APD로 표현된) 네트워크에서의 CNM 절차

4. 퍼지조립라인 밸런싱(Fuzzy Assembly Line Balancing:FALB) 문제

ALB는 조립라인에서 부과된 제약을 어기지 않고 특정목적을 최적화하기 위해 작업장에 작업들을 할당하는 문제와 관련된다. ALB 문제에서 여러 목적들이 고려되는데, 가장 흔히 다루는 목적으로는 사이클 타임이 주어지고 작업장의 수를 최소화 하는 것으로 이러한 상황을 type I 문제라 부른다. 그리고 작업장 수가 주어지고 사이클 타임을 최소화 하는 것인데 이를 type II 문제라 부른다. 연구의 대부분은 type I 문제를 다루는데 type II 문제를 해결하기 위해 사용되어지는 방법 대부분은 type I 문제를 해결하기 위한 방법으로 도입되어지기 때문에 실제 전체 ALB 문제를 해결하는 것은 2단계 문제로 보여질 수 있다. 본 연구에서는 type I 문제를 다룬다.

ALB의 일반적인 연구들에서는 기간, 또는 가공시간(processing time)이 확정적인 값을 가진다는 가정 하에 이루어져 왔으나 실제 현장에 있어서는 기간을 확정적인 값으로 나타낸다는 것이 비현실적일 경우가 많다. 특히, 경험이 없는 작업, 주로 사람의 손으로 이루어지는 작업, 자동화 설비가 제대로 갖추어지지 못한 작업장에서 운반시간을 고려할 경우에 확정적인 값으로 시간을 나타낸다는 것은 비현실적이

다. 이럴 경우 작업기간을 확정적인 값으로 나타내기 보다는 구간(interval)으로 표현하는 것이 더욱 현실적일 것이다.

이와 같이 구간으로 나타내어진 작업기간은 Kaufmann과 Gupta[8]의 확신 구간(interval of confidence)의 개념을 이용한 퍼지수로 표현할 수 있다. 퍼지수로 표현함으로써 얻을 수 있는 이점은 관리자에게 작업시간을 정확한 값으로 표현하도록 강요 할 필요가 없고, 구간으로 나타내어진 가공시간을 표현하기 위해 확률분포를 구할 필요가 없다는 것이다.

퍼지수로 나타내어진 작업기간의 크기를 비교하는데는 퍼지수 비교법을 이용해 효과적으로 수행할 수 있다. 현재까지 퍼지수 비교법에 대한 많은 연구가 있었고, 많은 방법들이 제시되어 있다. 이들 퍼지수 비교법 중 적용되어질 상황에 적합한 비교법을 선택하는 것이 중요하다. 왜냐하면 많은 비교 방법들이 개발 되었지만 다 어느 한 상황에 국한된 것이지 완벽하게 모든 상황에 적용되지는 못하기 때문이다. 그러므로 적용되어질 곳의 목표에 잘 맞는 비교 방법을 의사결정자가 잘 선택해야 한다. 본 연구에서는 퍼지수는 삼각퍼지수를 사용하고 퍼지수 비교법은 Lee-Li 퍼지수 비교법[9]을 사용한다.

FALB 문제는 조합 최적화 문제에서 NP-hard 문제에 속한다. 따라서 많은 발견적 기법들이 연구되어 왔다. 이런 발견적 기법들은 크게 두가지 범주로 나눌 수 있는데, 첫번째 범주는 탐색 공간과 탐색 시간을 줄여 해를 찾아 나가는 탐색 알고리즘을 이용하는 경우이고 두번째 범주는 각 문제의 구조에 따른 작업들의 가중치를 계산하여 할당을 거쳐 문제를 해결하는 경우로 작업의 할당 순서는 최초의 부분 순서 관련성을 지키면서 작업의 가중치에 의해 결정한다. 첫번째 범주에 속하는 방법은 좋은 해는 제공하지만, 많은 이론적인 지식없이 적용하기가 어렵고 유연성이 적다는 단점이 있다. 반면 두번째 범주의 방법들은 이해, 실행하기 쉬우며 유연성도 가지고 있고 좋은 해도 제공한다는 장점이 있다. 본 연구에서 제안 하는 방법은 쉬운 적용이 가능한 두번째 범주의 알고리즘이다.

4.1 기호

C : 사이클 타임

\tilde{C} : 퍼지 사이클 타임

\tilde{D} : 퍼지 작업기간

\tilde{W} : 퍼지 가중치

\tilde{TA} : 작업장에서의 활용가능한 시간

A (Available) 집합 : 이미 할당된 선행노드를 가지는 할당되지 않은 노드

$F \& A$ (Fit & Available) 집합 : 이미 할당된 선행노드를 가지는 할당되지 않은 노드 또는 현재 작업장의 활용가능한 시간보다 더 적은 작업시간을 가진 노드들의 집합

$\tilde{\lambda}$: 변환 모수 (위 절차는 두 단계로 이루어 진다. 첫 단계는 네트워크의 전체구조 제약에 초점을 맞추고 두 번째 단계는 사이클 타임의 완전한 사용에 초점을 맞추는데, 이 $\tilde{\lambda}$ 값은 한 단계에서 다른 단계로의 이동을 결정해 주는 값이다. $\tilde{\lambda}$ 의 값이 커지면 선후제약을 보다 중시한 할당이 이루어지고 $\tilde{\lambda}$ 의 값이 작으면 사이클 타임의 제약을 중시한 할당이 이루어진다.)

4.2 CNM에 기초한 휴리스틱 절차

FALB 문제의 특성으로 부터 주어진 시간에 오직 하나의 작업이 작업장에 의해 수행되어진다. 그러므로 FALB 문제를 해결하는 과정은 그 작업들을 수행하는 순서를 결정하는 것을 포함한다. 다른 말로 문제 해결 과정은 부분적으로 순서가 정해진 작업들을 전체적인 작업 순서로 바꾸는 것이다.

FALB 문제를 위한 제안 휴리스틱 절차

단계 1) APD로 FALB 문제를 표현한다.

단계 2) 각 노드의 퍼지수 가중치를 계산한다.

- 단계 3) 할당을 위해 새로운 작업장을 오픈한다.
- 단계 4) 현재 작업장에 현 F&A 집합으로 부터 주노드를 할당한다. 만약 동틀이면 더 긴 작업기간을 가진 작업을 먼저 할당(기간이 동틀이면 작업번호가 작은 것부터 할당)
- 단계 5) 만약 활용가능한 시간이 $\bar{A} \times \bar{C}$ 보다 크다면, 단계 4로 가고, 그렇지 않으면 단계 6으로 간다.
- 단계 6) F&A 집합에서 가장 긴 작업을 할당한다. F&A 집합이 빌 때까지 반복한다.
- 단계 7) 만약 모든 작업들이 할당되었다면 정지. 그렇지 않으면 단계 3으로 돌아간다.

4.3 FALB을 해결하기 위한 CNM을 기초로 한 휴리스틱 절차의 예

표 2의 자료를 가지고 퍼지 사이클 타임이 (50,50,50) 일때 제안 알고리즘을 사용하여 할당하는 과정을 보이고자 한다. 단계 1과 2는 이미 행해졌기 때문에, 단계3에서 시작한다. 그리고 할당을 위해 첫 작업장을 오픈한다. 시작 노드 다음, 노드 1이 F&A 집합에서 노드 1, 2, 3 사이의 주 작업이 된다. 단계 4는 작업 1을 작업장 1에 할당된다. 그리고 나서 이용가능한 시간은 (41,42,43)으로 감소된다. 그리고 F&A 집합은 노드 2, 3, 4를 포함하는 것으로 수정된다. 단계 5에서 이용가능한 시간은 $\bar{A} \times \bar{C} = 0.5 \times \bar{C} = (25,25,25)$ 보다 더욱 크다. 그래서 단계 4를 되풀이 한다. 그리고 현재 작업장에 새로운 주 작업 3을 할당한다. 이용가능한 시간은 (41,42,43) \ominus (11,12,13) = (28,30,32) > (25,25,25)로 감소된다. 그리고 새로운 F&A 집합은 노드 2, 4, 6, 7을 포함한다. 그리고 여전히 단계 4를 적용하고 주 작업 4를 할당한다. 다시 이용가능한 시간은 (28,30,32) \ominus (24,27,29) = (0,3,8)으로 감소 된다. 그리고 F&A 집합은 공집합이 된다. 다음 두번째 작업장에 할당을 시작한다.

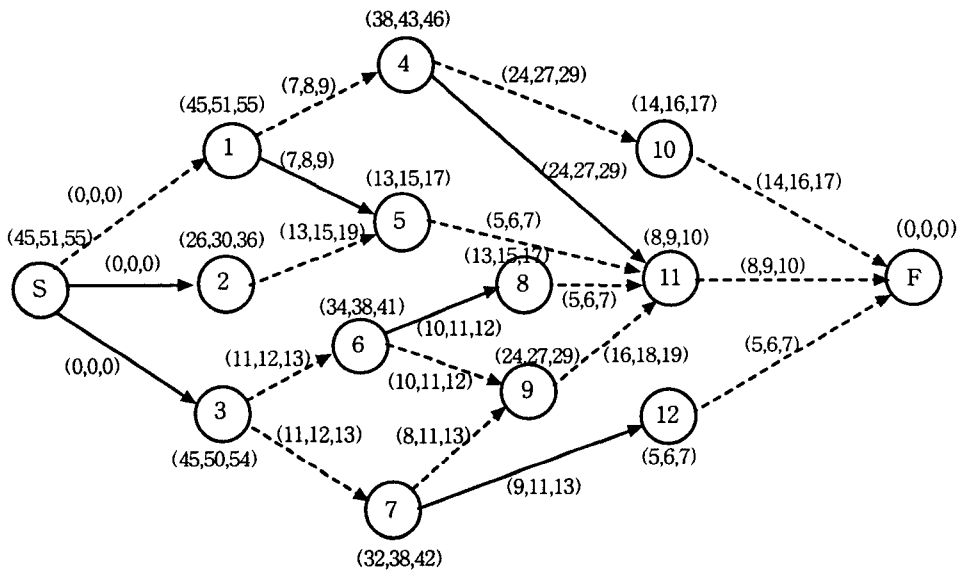
같은 방법으로, 단계 4를 반복하고 작업 6, 7, 2를 할당한다. 이 순간 남아 있는 시간은 (6,13,18) < (25,25,25)이다. 그리고 F&A 집합은 작업 5, 8, 12를 포함한다. 단계 6에 있어서, F&A 집합에서 가장 큰 기간을 가진 작업 5가 할당되어진다. 남아있는 시간 (0,7,13)이다. 그리고 수정된 F&A 집합은 작업 8, 12를 포함한다. 단계 6에서 작업 8이 할당된다. 수정되어 남아 있는 시간은 (0,1,8)이 되고 F&A 집합은 공집합이 된다. 그리고 새로운 작업장에서 할당을 다시 시작한다. 계속 같은 방법으로 남은 작업들을 작업장 3에 할당한다. 그 과정은 표 3과 같고 결과는 그림 3과 같다.

표 2. 작업의 목록

작업	퍼지 기간(\bar{D})	직접 후행 노드
1	(7,8,9)	4, 5
2	(13,15,19)	5
3	(11,12,13)	6, 7
4	(24,27,29)	10, 11
5	(5,6,7)	11
6	(10,11,12)	8, 9
7	(9,11,13)	9, 12
8	(5,6,7)	11
9	(16,18,19)	11
10	(14,16,17)	
11	(8,9,10)	
12	(5,6,7)	

표 3. FALB 문제의 해결 절차

station	step	\bar{TA}	A set	F&A set	rule	task
1	1	(50,50,50)	1,2,3	1,2,3	$\bar{W}1 > \bar{W}3 > \bar{W}2$	1
	2	(41,42,43)	2,3,4	2,3,4	$\bar{W}3 > \bar{W}4 > \bar{W}2$	3
	3	(28,30,32)	2,4,6,7	2,4,6,7	$\bar{W}4 > \bar{W}6 = \bar{W}7 > \bar{W}2$	4
	4	(0,3,8)	2,6,7,10	0		
2	1	(50,50,50)	2,6,7,10	2,6,7,10	$\bar{W}6 = \bar{W}7 > \bar{W}2 > \bar{W}10, \bar{D}6 = \bar{D}7$	6
	2	(38,39,40)	2,7,8,10	2,7,8,10	$\bar{W}7 > \bar{W}2 > \bar{W}10 > \bar{W}8$	7
	3	(25,28,31)	2,8,9,10,12	2,8,9,10,12	$\bar{W}2 > \bar{W}9 > \bar{W}10 > \bar{W}8 > \bar{W}12$	2
	4	(6,13,18)	5,8,9,10,12	5,8,12	$\bar{D}5 = \bar{D}8 > \bar{D}12, \bar{W}5 = \bar{W}8 > \bar{W}12$	5
	5	(0,7,13)	8,9,10,12	8,12	$\bar{D}8 = \bar{D}12, \bar{W}8 > \bar{W}12$	8
	6	(0,1,8)	9,10,12	0		
3	1	(50,50,50)	9,10,12	9,10,12	$\bar{W}9 > \bar{W}10 > \bar{W}12$	9
	2	(31,32,34)	10,11,12	10,11,12	$\bar{W}10 > \bar{W}11 > \bar{W}12$	10
	3	(14,16,20)	11,12	11,12	$\bar{D}11 > \bar{D}12$	11
	4	(4,7,12)	12	12		12
	5	(0,1,7)				



퍼지 사이클 타임 = (50, 50, 50)

작업장 1 = [1, 3, 4], 작업장 2 = [6, 7, 2, 5, 8], 작업장 3 = [9, 10, 11, 12]

그림 3. FALB 문제의 결과

5. 결론

FALB문제들에 대한 테스트 문제집합은 Hoffmann에 의해 확인된 중간크기, 큰 규모의 문제(작업 28~111)등 여러 문제들을 포함한다[10]. 이 집합은 규모가 클 뿐만 아니라 실제 산업현장에 가까우며, 이전

연구들에서 컴퓨터에서 산출한 테스트 문제가 아닌 실제적인 구조를 가진 문제들을 포함하고 있다. 그들 문제들은 ALB 문제에 적용한 휴리스틱 절차들을 평가할때 주로 사용되는 벤치마크 문제들이다. 그리고 그들 대부분은 어려운 문제로 간주되어진다. 예를들어 Hoffman은 사이클 타임이 156에서 70개 작업 문제의 최적해를 찾기 위해 거의 7백만개의 조합을 평가하였다.

테스트 문제 집합은 각 문제 구조에 따른 모든 사이클 타임을 포함하고 있다. 그 테스트 결과는 최고로 인정되고 있는 휴리스틱 해와 값들을 비교하여 평가되어졌다.

Hoffman에 의해 수집된 76개의 벤치마크 문제들에 대해 조사해 보았다. 그들 테스트 문제에 대한 최적해는 Talbot & Patterson[11], Baybars[12], Hoffman, Tonge[13] 논문을 참고 하였다.

표 4에서 CNM을 기초한 제안 알고리즘이 다른 알고리즘 보다 우월함을 볼 수 있다. 이러한 평가는 기간들이 확정적인 값을 가지는 문제에 대해 이루어 진것으로 이 방법론을 퍼지화한 제안알고리즘도 퍼지환경하에서 타 알고리즘에 비해 높은 효율성을 보일 수 있으리라 판단된다.

본 연구는 실 현장에서 쉽게 사용할 수 있고, 또한 실제 현장의 불확실성을 반영할 수 있는 간편한 그래프를 기초로한 휴리스틱 절차의 개발로 의사결정자가 생산라인을 운용하는데 있어서의 융통성과 효율성을 제고시킬 수 있으리라 생각된다.

표 4. CNM을 기초한 휴리스틱 절차의 시험 결과

문제의 출처	샘플 크기	CNM에 의해 구해진 최적해의 횟수	적용이 가능한 다른 휴리스틱 결과들의 수	최선의 휴리스틱 해보다 CNM의 결과가 나은 경우	CNM이 최선의 휴리스틱 결과와 같은 횟수
HASKIA	6	6	4	0	4
SAWYER	16	16	11	6	5
K & W	9	9	6	5	1
TONGE	30	18	20	1	11
ARCUS83	7	7	7	0	7
ARCUSIII	8	8	Not available	Not available	Not available

6. 참고문헌

[1]Salvason, M. E., "The Assembly Line Balancing Problem", The Journal of Industrial Engineering, May-June, 1955.

[2]Baybars, I., "An efficient heuristic method for the simple assembly line balancing problem", International Journal of Production Research, Vol. 24, pp.149-166, 1986

[3]Talbot, F., J. Patterson and W. Gehrlein, "A comparative evaluation of heuristic line balancing techniques", Management Science, vol. 32, No. 4, pp. 430-454, 1986.

[4]Johnson, R., "Optimally balancing large assembly lines with FABLE", Management Science, vol.34, No.2, pp.240-253, 1988.

[5]Macahon, C.S. , "Fuzzy Set Theory applied to Production and Inventory Control", Ph. D. Dissertation, Kansas state University, Manhattan, KS, 1987.

[6]Anderson, E. J. and M. Ferris, "Genetic Algorithm for Combinational Optimization: The Assembly Line Balancing Problem", ORSA Journal on Computing, Vol. 6, No. 2, 1994.

[7]Zhaolun Sun, "Critical Node Method and its Applications", Ph. D. Dissertation, The city University of New York, 1997.

[8]Kaufmann, A. and M. M. Gupta, *Introduction to Fuzzy Arithmetic*, North-Holland, Amsterdam, 1985.

[9]Lee, E.S. and R.J. Li, "Comparison of Fuzzy Numbers based on the Probability Measure of Fuzzy Events", Computers Math. Applic. Vol. 15, pp. 887-896, 1988.

[10]Hoffmann, T.R., "Assembly line balancing: a set of challenging problems", International Journal of Production Research, Vol. 28, No. 10, pp. 1807-1815. 1990.

- [11] Talbot, F.B. and Patterson, J.H., "An integer programming algorithm with network cuts for solving the assembly line balancing problem", *Management Science*, Vol. 30, No.1 pp.85-99, 1984.
- [12] Baybars, I. "An efficient heuristic method for the simple assembly line balancing problem", *International Journal of Production Research*. Vol. 24, pp.149-166, 1986.
- [13] Baybars, I. "A survey of exact algorithms for the simple assembly line balancing problem", *Management Science*, Vol. 32. No. 8, pp. 909-932, 1986.
- [14] Tonge, T.M., "Assembly line balancing using probabilistic combinations of heuristics", *Management Science*, Vol. 11. No. 7 pp. 727-735, 1965.
- [15] Dubois, D. and H. Prade, *Fuzzy set and systems :Theory and Application*, Academic Press, 1980.