

Fuzzy-MOEH : 퍼지 개념을 이용한 자원제약 프로젝트 스케줄링 알고리즘

(Fuzzy-MOEH : Resource Constraints Project Scheduling Algorithm with Fuzzy Concept)

고 장 권 [†] 신 예 호 ^{**} 류 근 호 ^{***} 김 흥 기 ^{***}
(Jang Kwon Koh) (Ye Ho Shin) (Keun Ho Ryu) (Hong Gi Kim)

요 약 자원제약 하에서의 프로젝트 스케줄링 문제는 많은 불확실성의 요소들을 내포하고 있으며 스케줄의 구성은 전문가에 의한 자의적 판단에 따르는 것이 일반적 현상이다. 전문가는 스케줄을 구성하기 위해 자신의 경험을 토대로 프로젝트 진행을 위한 작업(activity)들을 식별하고 이 작업들 사이의 선행관계를 규정하며 각 작업들에 대한 예상 시간을 이용하여 스케줄을 작성한다. 이 때 대부분의 스케줄링 방법들은 비용과 작업기간 두 요소들 중 한 요소에 집중하게 된다. 또한 스케줄 작성의 중요한 요소인 작업기간 결정이 전문가의 경험에 의존하여 결정됨으로써 결정된 작업기간의 불확실성을 초래할 수 있으며 따라서 이 불확실한 작업기간을 이용하여 구성된 스케줄의 불확실성이 증대되는 문제를 내포하고 있다. 이와 같은 문제 즉 스케줄 구성에 있어 작업기간의 불확실성과 작업비용을 함께 고려하지 못하는 문제를 해결하기 위해 이 논문에서는 Fuzzy 개념을 이용한 작업기간의 정형화를 시도하며 아울러 정형된 퍼지 작업기간과 결합 가능한 퍼지 작업비용을 도입하여 작업기간과 작업비용을 함께 고려하는 Fuzzy-MOEH 스케줄링 알고리즘을 제안한다. 아울러 이 논문에서 제안한 Fuzzy-MOEH 알고리즘과 기존 MOEH 알고리즘의 수행 결과에 대한 비교를 통해 Fuzzy-MOEH 알고리즘의 효용성 및 특성을 분석한다.

Abstract Project scheduling under resource constraint conditions have contained to many uncertain factors and it is performed by human experts. The expert identifies the activities of the project, decides the precedent relationships between these activities, and then construct the schedule using expected activity's duration. At this time, most of the scheduling methods concentrate on one of scheduling factor between activity duration and cost. And the activity duration, which is the most important factor in scheduling, is decided by heuristic of expert. Therefore it may cause uncertainty of activity duration decision and the use of this activity duration may increase the uncertainty of constructed schedule. This paper proposes Fuzzy-MOEH scheduling algorithm, which is the aggregation of the fuzzy number for deciding activity duration and applies the cost function for solving the problems of previous scheduling methods. This paper also analyze the utility and property of Fuzzy-MEOH algorithm through the comparison between Fuzzy-MEOH algorithm and existing MEOH algorithm.

1. 개요

소프트웨어 개발 프로젝트에서 일정관리를 결정하는 스케줄링이 차지하는 비중은 매우 크다. 왜냐하면 프로젝트 수행 일정을 규정하고 있는 스케줄은 프로젝트 진도를 관리할 수 있는 주요한 수단일 뿐만 아니라 프로젝트의 수행과정에서 발생할 수 있는 많은 문제점들을 추적 관리하는데 있어 주요한 지표로 이용될 수 있기 때문이다. 소프트웨어 개발 프로젝트에서 관리자는 프로젝트의 스케줄을 수립하기 위해서 여러 가지 동원 가능한 자

· 이 연구는 KAIST 해킹 바이러스 연구센터의 연구비 지원으로 수행되었음.

[†] 정 회 원 : 군장대학 컴퓨터응용학부 교수
jkkoh@kunjang.ac.kr

^{**} 비 회 원 : 충북대학교 컴퓨터과학과
snowman@dbl.ab.chungbuk.ac.kr

^{***} 총신회원 : 충북대학교 전기전자컴퓨터공학부 교수
khryu@dbl.ab.chungbuk.ac.kr
hgkim@cbucc.chungbuk.ac.kr

논문접수 : 1999년 10월 5일

심사완료 : 2001년 3월 28일

원을 생각하면서 소요작업을 분석하고, 일정을 수립해야 한다. 또한 작업간의 상호 의존성을 감안하여, 가능하면 병행작업을 해야하고, 예상되는 문제점이나 추진일정의 지연요인, 병목현상 등에 대한 대책 등을 수립해야 한다. 이와 같은 문제들을 해결하기 위해 많은 스케줄링 도구들이 개발되었다[1]. 1차 대전 이후 고안된 간트 차트는 일정표를 그림으로 나열한 것으로서 스케줄링 도구의 가장 기본적인 형태를 갖추고 있다. 이후 프로젝트의 시작에서 끝까지 수행되어야 하는 작업을 그림이나 표로 표현해주는 태스크 네트워크를 만들어내는 PERT/CPM(Program Evaluation and Review Technique/Critical Path Method)이 스케줄링 방법으로 등장하였다.

그러나 간트 차트는 작업들 사이의 선행 관계를 기술할 수 없으며 따라서 작업의 지연이 전체 스케줄에 미치는 영향을 파악하기가 어렵다. 반면 PERT/CPM 방법은 작업 사이의 선행 관계를 태스크 네트워크를 이용하여 도식적으로 표현함으로써 지연되는 작업에 의해 발생하는 문제 등은 쉽게 해결할 수 있으나 자원제약의 문제라든지 프로젝트 수행 비용 등과 같은 요구사항을 수용할 수 없다[2]. 이와 같은 문제들을 해결하기 위한 방법으로 [3]에서는 스케줄링의 수행에 있어 자원제약 및 선행제약 조건을 만족하면서 시간 창(time window)의 개념을 이용하여 특정 시간대에 특정 작업을 진행시킬 수 있도록 하는 것과 같은 다중의 요구사항을 수용할 수 있는 MOEH(Multiple Objective Exchange Heuristic)알고리즘을 제시하였다. 그러나 MOEH 알고리즘이 기존의 방법들보다 개선되기는 하였으나 다음과 같은 문제점들을 갖고 있다.

- MOEH에서의 작업기간은 전문가의 경험에 의해 산출된 불확실한 값으로서 정형화되지 못하였다.
- MOEH스케줄링 알고리즘에서 스케줄링의 주요한 목적을 작업기간으로 한정함으로써 스케줄 수행과정에서 발생하는 비용 문제를 해결하지 못하였다.

이와 같은 두 가지 문제점은 거의 대부분의 스케줄링 알고리즘에서 갖고 있는 공통적 문제점으로서 스케줄링에서의 본질적 문제에 해당하는 것들이다. 먼저 첫 번째 문제점의 원인은 일반적으로 프로젝트를 구성하는 각 작업들의 작업기간을 산출하는 정형화된 방법이 존재하지 않기 때문이며 두 번째 문제점인 비용의 문제는 스케줄 과정에서 거의 고려되지 못하거나 또는 비용이나 기간의 한 측면만이 강조되고 이들 두 요소를 조화롭게 고려하지 못하는 현실을 반영하고 있다. 그러나 이러한 문제는 프로젝트 수행에 있어 매우 중요한 문제로서 스케줄의 신뢰도 향상은 물론 프로젝트의 성공 여부를 결

정하는 매우 중요한 요소에 해당하는 문제들이다.

이에 이 논문에서는 이들 문제점들을 해결하기 위해 작업기간의 정형화를 위해 작업기간을 퍼지수(fuzzy number)로 환원하는 퍼지 기법을 도입하고 도입된 퍼지수와 결부된 비용함수를 적용하여 스케줄에 대응한 프로젝트 수행비용을 산출할 수 있는 새로운 스케줄링 알고리즘을 제시한다.

이를 위한 논문의 구성은 다음과 같다. 먼저 2장에서는 스케줄링과 관련된 기존 연구들을 검토하고 3장에서는 Fuzzy 수와 스케줄링의 관계 및 스케줄링에서의 불확실한 정보를 퍼지수로 합성하기 위한 방법에 대해 고찰한다. 4장에서는 MOEH알고리즘에 퍼지수를 확장 적용한 새로운 스케줄링 모델 및 이의 알고리즘을 제시하고 5장에서는 4장에서 제시한 스케줄링 모델에 대한 실험 및 검증을 수행한다. 마지막으로 6장에서 결론과 향후 연구방향에 대해 기술한다.

2. 관련연구

소프트웨어 공학분야에서 스케줄링에 관한 문제는 프로젝트 계획수립의 일 부분으로서 스케줄링 방법론에 대한 연구보다는 스케줄 수립에 있어 고려해야 할 사항들이나 스케줄 수립과정에서 어떻게 공학적 관점을 유지할 것인가와 같은 문제에 집중되었으며 스케줄링 방법론에 대한 부분은 기존에 제시되었던 간트 차트 또는 PERT/CPM등을 복합적으로 이용하는 경우가 대부분이었다[4].

반면 스케줄링 기법 자체에 대한 연구는 1차 대전 중에 고안된 간트차트와 50년대 말 개발된 CPM 및 PERT를 중심으로 전개되었다[5]. 간트차트는 단위 작업들을 계획하여 일정관리를 하고 프로젝트를 미리 예견해서 계획한 일정관리를 수정 보완할 수 있도록 하였다. CPM 및 PERT기법은 간트차트로는 적용하기 어려웠던 대규모 프로젝트에 대한 일정관리를 수행할 수 있게 하였으나 간트차트의 경우 작업의 일정속에 포함된 특정작업의 지연이 전체 프로젝트 수행에 미치는 영향을 파악하기가 어렵다는 문제점을 안고 있었으며 CPM과 PERT는 자원이 무제한적이라고 가정하고 스케줄의 목적도 작업기간을 최소화하는 것일 때만 효과적인 뿐, 자원이 제한 되어있고 이용 가능한 자원의 형태가 광범위하다면 상충하는 자원의 갈등을 극복할 수 없는 문제점을 안고있다[1].

PERT/CPM이 이와 같이 작업완료 시간이라는 단일의 스케줄 목적 달성에 적합한 반면 일반적인 스케줄링 환경은 자원의 제약이라든지 비용의 문제등 보다 복잡한

요인들을 포함하는 경우가 대부분이다. 이와 같이 작업 완료일을 최소화시키는 것 외에 병목현상을 없애거나 처리중인 목록을 최소화하는 등의 목적이 있는 다른 환경에서는 일정관리에 완료 기간, 준비 시간 및 자원제약 등과 같은 작업 특성들이 선행관계에 첨가되어야 한다. 그래서 처리 순서를 결정해주는 단순한 규칙의 집합인 constructive 알고리즘이 개발되었다[6]. 한편 Lawler는 선행 제약들을 관리하는데 있어 처리비용을 최소한으로 단축하는 알고리즘을 개발하였는데 작업의 비용은 완성 시간과 내림차순의 관계가 있어야 하며 작업을 연기하는 것은 이익이 없다는 결론을 얻었다[7].

이와 같은 스케줄링 알고리즘들을 유형별로 분류하면 분석기법과 경험적 기법으로 분류할 수 있다. 분석 기법은 최적 해를 찾기 위한 많은 연산 시간을 필요로 하지만 경험적 기법에서는 최적해가 아닌 좋은 해를 빨리 일반화시킨다[3]. 자원제약 하에서의 스케줄링은 NP-complete한 문제이다[6, 8]. 따라서 분석적 기법을 이용한 스케줄링은 매우 많은 시간을 필요로 하기 때문에 스케줄링의 목적을 자원제약 또는 작업완료시간 등과 같이 스케줄에서 요구되는 여러 가지 목적들 중 하나의 목적만 적용할 수 있다. 반면 경험적 기법은 비록 최적해는 아니지만 좀더 빨리 스케줄링을 완료할 수 있기 때문에 스케줄링 목적을 다중화 할 수 있으며 이들 다중의 목적을 조합해서 적용하는 것이 가능하다. 이렇게 다중화된 알고리즘의 대표적 예가 Yang에 의해 제안된 EH(exchange heuristic) 알고리즘이다[9]. EH 알고리즘은 자원제약 프로젝트 스케줄링 문제를 해결하기 위한 것으로 자원제약, 선행제약 등을 만족하면서 완료시간을 최소화 할 수 있도록 하였다.

MOEH 스케줄링 알고리즘은 경험적 기법의 일종인 EH 알고리즘에 작업이 반드시 수행되어야 할 시간 구간으로 정의되는 시간 창(time window) 개념을 도입하여 시간 제약에 대한 일반성을 확보한 다 목적 자원 제약 스케줄링 알고리즘이다. 이 스케줄링 알고리즘은 스케줄링을 수행하는데 있어 1) 자원제약(resource constraints)과 선행제약(precedent constraints)을 위배하지 않아야 하고 2)작업의 시작 시간 범위를 지정하는 시작 시간 창(start time window)을 위배하지 않아야 하며 3)작업이 끝나는 시간 범위를 지정하는 작업 종료 시간 창(end time window)을 위배하지 않아야 하고 4)프로젝트 완료 시간 즉 스케줄 완료시간(makespan)을 최소화할 수 있도록 한다. 그러나 이와 같은 경험적 기법은 비용과 시간을 동시에 고려하지 못하며 스케줄링의 가장 중요한 요소 중 하나인 작업기간에 대한 정형화 과정을 포함하지

않는다.

한편 경험적 기법이 스케줄링 과정에서 스케줄에 영향을 끼칠 수 있는 다양한 요인들을 포함할 수 있기는 하지만 생성된 스케줄을 이용하여 프로젝트를 진행하는 과정에서 발생하는 작업기간의 지연 등에 대한 문제를 포함하지는 못한다. 왜냐하면 이와 같은 문제를 해결하기 위해서는 프로젝트 수행 과정에서 프로젝트에 영향을 끼칠 수 있는 모든 요인들을 예측해야 하는데 이는 거의 불가능한 문제일 뿐만 아니라 예측이 가능하다 하여도 예측된 기능성들을 모두 또는 일부 반영할 경우 스케줄링 비용이 반영된 경우의 수에 비례하여 지수적으로 증가하는 문제가 있기 때문이다. 경험적 기법의 개선을 위해 [10]에서는 전문가 시스템을 이용하는 방법을 제시하고 있으나 [10]역시 프로젝트의 수행 과정에서 발생하는 작업기간의 지연 등에 대한 문제는 해결하지 못하고 자원의 재사용이나 새로운 자원의 할당 등과 같이 기존의 제약사항들을 개선하는데 그치고 있다.

프로젝트 스케줄링에 대한 또 다른 연구 분야로 퍼지 개념을 적용하는 연구는 [11, 12, 13, 14] 등과 같이 다수 존재하기는 하나 이는 여러 개의 후보 스케줄을 생성하여 이들 생성된 후보 스케줄들에 퍼지 비교 기법을 적용하여 가장 유효한 스케줄을 선택하는 방법들만을 제시하였으며 이 논문에서 제시하고 있는 바와 같이 스케줄 과정에서 직접 퍼지 개념을 적용하는 경우는 없다. 특히 비용 문제를 함께 고려한 경우는 그 사례를 찾아볼 수 없다.

3. Fuzzy수와 일정관리

2장에서 스케줄링에 기법에 대한 기존 연구들을 고찰하였다. 이 방법들은 대부분 스케줄의 주요 인자로서 주어지는 단위작업들의 작업기간을 정형화하지 못하고 경험자의 경험에 의한 불확실한 추정값만을 이용하였으며 스케줄링의 비용에 대한 문제를 고려하고 있지 못한다. 따라서 이 장에서는 이와 같은 문제를 해결하기 위해 Fuzzy 기법을 이용하여 단위 작업들의 작업기간에 대한 정형화를 시도하고 아울러 이 정형화된 작업기간과 작업 수행에 소요되는 비용을 결합 확장한 Fuzzy수를 이용한 단위 작업들의 작업기간 모델을 제시한다.

3.1 퍼지수의 도입

M을 퍼지수 즉 실수 R의 정규형이며 특별한 특성을 갖는 퍼지 부분집합이라 하고 $M = \{(x, \mu_M(x)) \mid x \in R\}$ 로 정의하며 $\mu_M(x)$ 는 [0, 1]사이의 값을 갖는 소속함수이고 x가 M에 속하는 정도를 나타낸다. M의 α -level 집합은 집합 $I(M, \alpha) = \{x \in R \mid \mu_M(x) \geq \alpha, \alpha \in (0, 1]\}$ 이

다[11]. 어떤 α -level 집합 $I(M, \alpha)$ 의 하한과 상한은 각각 $\inf_{x \in R} I(M, \alpha)$ 와 $\sup_{x \in R} I(M, \alpha)$ 이며 둘 다 유한 집합이라 할 때 이러한 α -level 집합을 표현하는 방법으로 $M=(m, \bar{m}, \alpha, \beta)_{LR}$ 으로 표현되는 L-R type의 플랫폼 퍼지수를 이용한다. 여기서 $[m, \bar{m}]$ 은 M 의 core이며, m 과 \bar{m} 은 M 의 하한 및 상한중심이고 α, β 는 core의 좌 우측 경사도 결정 값이다. 이 소속함수는 다음과 같은 사다리꼴 모양의 참조함수 L 과 R 로 표현된다.

$L(0)=R(0)=1$ 이고 $L(1)=R(1)=0$ 이며,

$$\mu_M(x) = \begin{cases} L[(m-x)/\alpha] & \text{if } x < m, \\ 1 & \text{if } x \in [m, \bar{m}], \\ R[(x-\bar{m})/\beta] & \text{if } x > \bar{m}, \end{cases} \alpha, \beta \in R^+ \text{이다.}$$

이 논문에서는 그림 1과 같이 최대 3개의 선형 조사를 가진 L-R 플랫폼 퍼지수로 정형화하는 것을 제안한다. 이러한 표현에 대한 정의는 $M=(m^\lambda, m, \bar{m}, \bar{m}^\lambda)$ 이다. 여기서 m, \bar{m} 는 퍼지수 M 의 코어이며 $m^\lambda, \bar{m}^\lambda$ 은 절단 수준 값 λ 에서의 퍼지수를 의미한다. 절단 수준 λ 는 퍼지수가 퍼지 집합에 소속되는 정도를 결정하는 것으로서 λ 이하의 소속 정도는 그 정도가 너무 작아 소속 관계를 무시할 수 있다. 또한 중심값 m, \bar{m} 은 퍼지 소속 정도가 1 즉 완전히 포함되는 부분의 퍼지수를 의미한다.

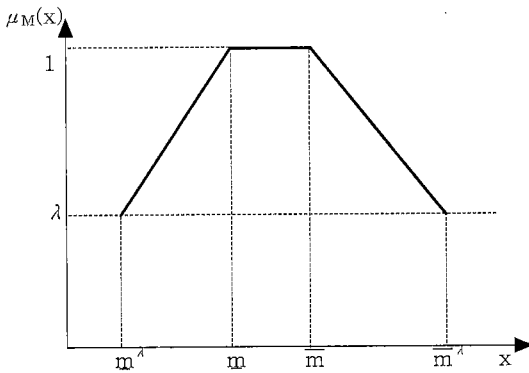


그림 1 Flat 퍼지수

3.2 일정관리에서 퍼지 정보의 합성

불확실한 정보를 포함한 의사결정 문제를 해결하기 위하여 가장 빈번한 접근방법은 다음의 3단계로 요약될 수 있다[11].

- (i) 불확실성의 정형화
- (ii) 불확실한 정보를 가진 문제를 연관된 확실한 문제로의 변환
- (iii) 연관된 확실한 문제를 해결하여 그것을 검증하고

필요하면 변환변수를 바꾸어 (ii)로 반복

이러한 절차를 프로젝트 일정관리 문제에 적용하면 다음과 같이 진행된다.

① 퍼지수의 간결한 형태는 전문가에 의해서 만이 잘 표현될 수 있으므로, 퍼지 자료의 적합한 소속함수를 구하는 실질적인 방법으로 전문가는 어떤 특정한 소속정도 α_j 를 설정하고 시작 한다. 예를 들면

$\alpha=1 : \mu(x) = 1$ 은 x 가 가능한 값들의 집합에 확실히 속하는 것을 의미하고,

$\alpha=\lambda : \mu(x) = \lambda$ 는 $\mu(x) \geq \lambda$ 인 x 가 가능한 값들의 집합에 속할 가능성이 높다는 것을 의미하며 이보다 작은 범위로 $\alpha=\epsilon$ 인 매우 작은 값을 설정할 수도 있으나 이 경우 $\mu(x) = \epsilon$ 인 x 가 가능한 값들의 집합에 속할 가능성이 매우 적기 때문에 $\mu(x) < \lambda$ 인 x 에 관련된 값을 무시할 수 있다.

② 퍼지수의 모든 α -level 절단은 그림 2에서와 같이 낙관과 비관적이라 불리는 하한선과 상한선의 쌍으로 특징지어진다[15]. 기간을 최소화 해야하는 일정관리 문제에서는 낙관적이라 함은 level α 에서 작업기간의 하한선이며 완료시기의 상한선이다.

반대로 비관적이라 함은 level α 에서 작업기간의 상한선이며 완료시기의 하한선이다. 각 퍼지수에 대하여 프로젝트 일정관리 알고리즘에서 4개의 실수 값이 사용되기 때문에 불확실성하의 프로젝트 일정관리 문제는 각각 2개의 값을 가진 시간 변수인 낙관과 비관적인 두 개의 시나리오 집합을 가진 4개의 분명한 문제와 관련되어 진다.

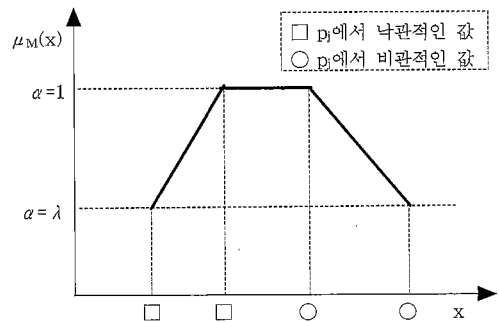


그림 2 퍼지수에서 두 개의 α -level cut과 그에 대한 4개의 실수 값

③ 연관된 확실한 문제를 해결하는 단계에서는 비관과 낙관적인 시나리오에 2개의 α -level cut을 따로따로 적용하여 반복적으로 시행한다. 따라서 최적의 시간에

대한 4개의 서로 다른 일정과 4개의 결과를 얻어 종합적인 퍼지 결과로 합성할 수 있다.

4. Fuzzy 수를 이용한 MOEH 알고리즘의 확장

3장에서 고찰한 바와 같이 Fuzzy 수는 불확실한 정보를 정형화하는데 효과적인 도구로 활용할 수 있다. 이 절에서는 이러한 Fuzzy 수의 특성을 MOEH 알고리즘에 도입하여 알고리즘에서 갖는 불확실성의 문제에 대한 정형화를 시도한다. 일반적으로 스케줄링에 사용되는 기본 인자들로 프로젝트를 구성하는 단위 작업들과 이 작업들에 대한 선행관계, 작업을 수행하는데 소요되는 자원의 수 및 작업 소요시간등을 이용한다. 이와 같은 스케줄 인자들의 값을 결정하는 문제는 프로젝트 스케줄을 수립하는 관리자의 경험을 토대로 도출되는 값으로서 정형화되지 못한 값을 갖고 있다. 이들 인자들 중 단위 작업들 및 이 작업들 사이의 선행관계는 스케줄 수립의 기본 요소로서 정형화의 대상이 될 수 없다. 반면 작업기간과 작업에 필요한 자원 문제는 프로젝트 관리자의 경력 등에 종속적인 값으로써 프로젝트의 수행 비용을 결정하는 중요한 문제이면서도 정형화된 값을 도출하는 방법이 제시되지 못하였다. 이는 작업기간이나 자원 소요량이 작업 자체의 문제뿐만 아니라 많은 외적 요인에 영향을 받을 수 있으며 이 외적요인들은 대부분 정량화하기 힘든 부정형의 요인들이 되기 때문이다. 이와 같이 많은 불확실성 요소들을 내포하는 작업기간 및 자원소요량에 대한 문제를 정형화하고 이 정형화된 값을 바탕으로 수립된 스케줄은 정형화를 하지 않은 스케줄보다 신뢰도를 향상시킬 수 있으며 보다 높은 생산성을 담보할 수 있다. 이를 위해 이 논문에서는 작업기간 및 자원 소요량에 대한 정형화를 하기 위해 먼저 작업기간을 퍼지수를 이용하여 정형화하고 이 정형화된 작업기간과 비용을 결합하여 자원에 대한 불확실성의 문제에 적용한다.

4.1 프로젝트 스케줄링에서의 비용 문제

3장에서 작업기간의 정형화를 수행하기 위한 L-R 퍼지수를 결정하기 위한 퍼지함수를 고찰하였다. 이 L-R 퍼지함수를 통해 얻어진 퍼지수를 이용하여 작업기간의 정형화를 시도할 수 있으며 이렇게 정형화된 작업기간은 작업수행에 필요한 비용과 결합함으로써 불확실한 정보의 합성을 달성할 수 있다. 퍼지수로 정형화된 정보를 작업비용과 합성하는 과정은 먼저 작업기간에 대응하는 비용함수를 결정하고 결정된 비용함수에 의해 구해진 비용을 스케줄에 반영하여 스케줄이 비용을 고려할 수 있도록 한다.

일반적으로 프로젝트의 진행과정에서 비용에 대한 문제는 직접 경비와 간접경비 두 측면을 고려한다. 이들 비용은 작업기간이 길어지는 것과 비례하여 직접경비는 감소하나 간접 경비는 증가하는 경향을 보이고 반대로 작업기간이 줄어들면 직접경비는 증가하나 간접경비는 감소한다[2]. 직접경비와 간접경비를 합한 총 비용을 개념적으로 표현하면 다음 그림 3과 같다.

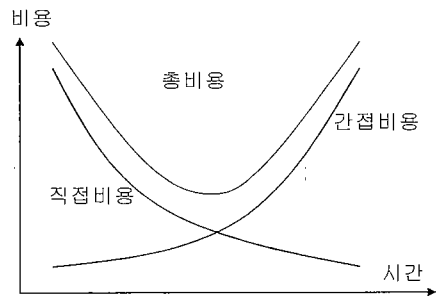


그림 3 전형적인 비용곡선

이러한 비용 문제는 일반적으로 경험상 관측되는 값들을 개념적으로 인식하고 있을 뿐 이를 정량화 하는 연구는 거의 이루어지지 않고 있다. 이는 프로젝트의 규모, 소요 자원의 제한, 작업기간 등과 같은 불확실한 변수들에 따라 매우 불특정 하게 나타날 수 있기 때문이다. 따라서 이 논문에서는 작업의 비용을 결정하는 비용함수를 정량화하기 위한 비용 모델을 설정하고 이 비용 모델을 근거로 비용 함수를 도출한다.

4.2 퍼지수를 이용한 비용 모델

3장에서 퍼지수를 이용한 작업기간의 정형화를 고찰하였고 4.1절에서 프로젝트 스케줄링에서의 비용 문제를 고찰하였다. 이 절에서는 이러한 내용을 토대로 정형화된 퍼지수와 비용과의 관계를 이용한 퍼지 비용 모델을 제시한다. 다음의 표 1은 비용 모델을 정의하기 위한 요소들을 정리한 것이다. 표 1에서 $u_{-}d_i$, $m_{-}d_i$, $\bar{m}_{-}d_i$, $o_{-}d_i$ 은 퍼지수로 환원된 작업기간이다.

표 1에서 $ratio()$ 로 표기된 작업기간의 비용 기여도 함수는 정형화된 퍼지 작업기간이 비용에 미치는 영향을 계산해 내는 함수로서 이 논문에서는 직접 경비에 한정해서 적용한다. 직접경비는 작업기간에 반비례하는 특성을 나타내므로 이 함수는 반비례 관계의 곡선을 결정하는 함수가 된다. 이러한 함수의 종류로는 지수함수, 이차 함수 등이 있을 수 있으나 이 논문에서는 지수 함수를 이용한 비용 기여도 계산을 수행한다. 이는 지수

표 1 비용평가 모델 요소

k :	자원리스트에서 자원을 식별하기 위한 색인값
i :	단위 작업들을 식별하기 위한
u_d_i :	작업 i 에 대한 낙관적 퍼지 작업기간
$\underline{m_d}_i$:	작업 i 에 대한 하한 중심 작업기간
$\bar{m_d}_i$:	작업 i 에 대한 상한 중심 작업기간
o_d_i :	작업 i 에 대한 비관적 퍼지 작업기간
$ratio()$:	퍼지 확장된 작업기간에 대한 비용 기여도 함수
r_k :	자원 k 에 대한 소요량
$Unit_k$:	자원 k 의 단위 비용
$Cost_i$:	작업 i 에 대한 퍼지 비용

함수가 작업기간에 대한 비용 적응성이 뛰어나고 반비례 관계의 모델링에 적합하기 때문이다. 표 1의 요소들을 이용한 비용 모델은 퍼지수로 정형화된 각각의 작업기간에 대한 비용 산출을 토대로 구성한다. 다음은 퍼지수로 정형화된 작업기간에 대한 비용함수이다.

- $Cost(u_d_i) = \sum_k (ratio(u_d_i) / ratio(\underline{m_d}_i)) \times Unit_k \times r_k$
- $Cost(\underline{m_d}_i) = \sum_k (ratio(\underline{m_d}_i) / ratio(\underline{m_d}_i)) \times Unit_k \times r_k$
- $Cost(\bar{m_d}_i) = \sum_k (ratio(\bar{m_d}_i) / ratio(\bar{m_d}_i)) \times Unit_k \times r_k$
- $Cost(o_d_i) = \sum_k (ratio(o_d_i) / ratio(\bar{m_d}_i)) \times Unit_k \times r_k$

이 비용 함수는 평균 작업기간을 기준으로 상대적 비용을 계산하는 모델이다. 따라서 이 비용 모델을 이용할 경우 적어도 작업기간에 대한 정형화된 퍼지수를 적용하지 않은 스케줄링 모델과 비교하여 동등한 정도의 스케줄을 얻을 수 있음을 보장한다. 그림 4는 위 비용식을 도식적으로 표현한 것이다.

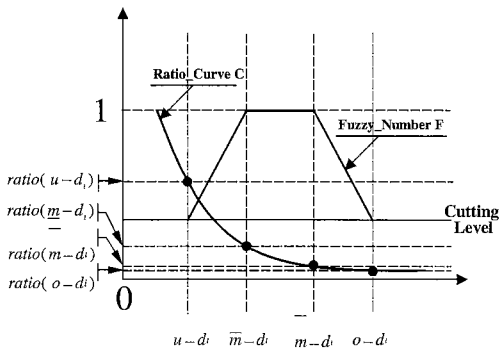


그림 4 비용함수를 이용한 비용의 산출

그림 4에서 절단 수준(cutting level)은 λ 에 해당하는 것으로 작업 관리자에 의해 주어지며 퍼지수로 환원되는 작업기간의 퍼지 소속정도에 대한 하한을 나타낸다. 작업기간에 대한 퍼지수의 궤적을 표현하는 F 또한 작업관리자에 의해 주어지는 것으로서 작업관리자는 L-R 퍼지함수를 이용하여 퍼지화 되지 않은 작업기간으로부터 확장한 퍼지 작업기간의 궤적을 표현한 것이다. 이 비용모델을 이용한 비용계산 절차는 다음과 같다. 먼저 퍼지 곡선 F와 절단수준과의 교차점에서 수직선을 긋는다. 다음으로 수직선과 비용 기여도 함수 곡선 C가 만나는 점의 퍼지 값을 구한다. 그런 다음 구해진 퍼지 값을 중간 값을 이용하여 나눈다. 이렇게 하면 각각의 퍼지 값은 중간 값에 대한 비율 형태로 계산된다. 이 비율을 자원에 대한 단위비용과 곱한 후 이 계산된 비용을 다시 자원 요구량과 곱하여 각 자원에 대한 비용계산을 수행한다.

4.3 퍼지 비용을 이용한 스케줄링

이 절에서는 4.2절에서 언급한 비용 모델을 토대로 계산된 비용을 이용하여 스케줄링을 수행하는 과정에 대해 기술한다. MOEH 스케줄링 알고리즘은 먼저 스케줄되지 않은 작업들 중 자원제약, 선행제약 등을 위배하지 않는 작업에 대해 스케줄 후보로서 선택한다. 그런 다음 이 후보를 스케줄러를 통해 스케줄에 삽입한다. 이와 같은 MOEH 스케줄링 알고리즘에서 퍼지 비용은 작업 선택 과정에 적용할 수 있다. 이를 위해 먼저 4.2절에서의 퍼지비용 계산을 수행한 후 계산된 비용 중에서 관리자가 지정하는 정책에 적합한 퍼지기간을 선택한다. 지정할 수 있는 정책으로는 1)가장 적은 비용, 2)가장 적은 작업기간, 3)가장 큰 작업비용 4)가장 큰 작업기간 등을 선택할 수 있으며 1)번의 경우 계산된 비용 중에서 가장 적은 값을 갖는 작업기간을 선택하면 되고 2)의 경우는 낙관적 작업기간을 선택하면 된다. 다음 그림 5는 MOEH 스케줄링 알고리즘을 수행하는 스케줄러의 구조를 퍼지 비용을 수용할 수 있도록 확장한 그림이다. 그림 5에서 퍼지 개념을 이용한 스케줄링을 수행하기 위해 확장된 부분은 입력 요소들이 퍼지 스케줄링에 대응할 수 있도록 작업기간을 정형화된 퍼지 작업기간으로 확장하였으며 4.2절의 퍼지비용 모델을 적용하기 위해 각각의 자원에 대한 단위 비용이 추가되었고, 스케줄러 역시 퍼지개념을 적용하여 확장된 입력을 이용할 수 있도록 확장하였다. 이는 스케줄러에서 작업 선택을 담당하는 작업 선택기에 퍼지비용을 고려한 선택을 수행할 수 있도록 확장한 것으로서 퍼지 작업 선택기 모듈을 구성하며 확장 적용된다. 이러한 확장은 입력의 퍼지변

수들과 이 퍼지 변수들에 대응한 퍼지연산을 통해 MOEH 알고리즘이 퍼지개념에 대응할 수 있도록 하는 주요한 수단이다.

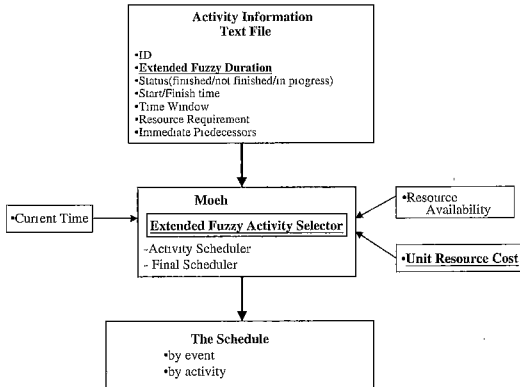


그림 5 Fuzzy-MOEH 스케줄러 Architecture

4.4 퍼지수를 이용한 알고리즘의 확장

MOEH 스케줄링 알고리즘의 퍼지 확장은 스케줄링을 위한 작업 선택 모듈에 대한 퍼지 확장을 통해 구현된다. 작업 선택 모듈은 4.3절에서 제시한 네 가지의 퍼지 스케줄링 정책인 최소 비용, 최대 비용, 최소 작업기간, 최대 작업기간을 수용한다. 이 네 가지 정책은 불확실한 시간정보인 작업기간을 퍼지수로 정형화 한 확장된 퍼지 작업기간을 이용하여 수행하며 수행 과정은 확장된

퍼지 작업기간과 퍼지 비용을 모두 고려하여 수행한다. 그림 6은 작업 선택 모듈에 대한 개략적 알고리즘을 기술하고 있다.

그림 6에서 작업 선택 모듈은 지정한 정책에 따라 네 가지 정책에 해당하는 함수를 호출하게 되고 해당 함수들은 정책에 대응한 작업 번호를 반환한다. 이 때 스케줄 정책을 수행하는 정책 함수들 중 작업기간 함수인 MIN_DUR과 MAX_DUR은 퍼지 작업 기간중 가장 작은 또는 가장 큰 작업기간을 반환하는 간단한 연산만을 수행하지만 MIN_COST와 MAX_COST는 퍼지 작업기간과 퍼지 비용을 모두 고려하여야 하며 작업기간과 비용 사이의 상관 관계를 이용한 복잡한 연산을 수행한다. 다음 그림 8은 MIN_COST정책 수행 함수에서 가장 중요한 작업비용 계산 과정을 기술하는 알고리즘이다. 그림 8에서 평균 작업기간에 대한 비율 계산은 항상 1이 나오도록 하고 있다. 이는 상한 및 하한 평균을 모두 포함하는 평균 작업기간은 항상 퍼지 소속 정도 1 즉 스케줄링에서 최대한 보장 해 줄 수 있는 기간을 의미하며 또한 기존의 MOEH알고리즘에서 적용하던 작업기간에 해당하는 값으로서 확장된 퍼지 알고리즘에서 이득을 기대할 수 없는 기간에 해당하기 때문이며, 최소한의 성능 보장을 위한 작업기간이기 때문이다. 반면 낙관적 작업기간 u_d 와 비관적 작업기간 o_d 는 평균 작업기간에 대한 비율로서 제시되는데 이는 평균작업기간에 대한 이득 값을 결정하는 요소로서 적용하기 위해서 이다. 이 때 낙관적 작업 기간인 u_d 는 하한 중심값인

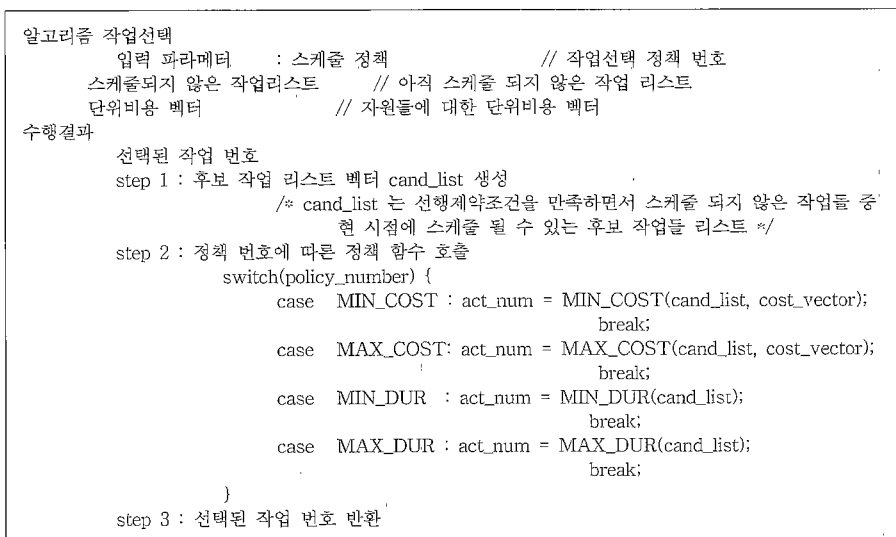


그림 6 퍼지 확장된 작업선택 모듈

알고리즘 비용계산
 입력 파라미터 : 후보작업리스트, 단위비용벡터
 수행결과

각 후보 작업에 대해 계산된 작업수행비용
 step 1 : 사상 함수를 이용 퍼지 작업기간을 퍼지 소속값으로 사상

$$act_i.ratio_0 = fuzzy_mapper(act_i, u_d)$$

$$act_i.ratio_1 = fuzzy_mapper(act_i, \overline{m_d})$$

$$act_i.ratio_2 = fuzzy_mapper(act_i, m_d)$$

$$act_i.ratio_3 = fuzzy_mapper(act_i, o_d)$$

step 2 : 사상된 퍼지 소속기간을 이용한 비용비를 계산

$$act_i.ratio_0 = act_i.ratio_0 / act_i.ratio_1$$

$$act_i.ratio_1 = act_i.ratio_1 / act_i.ratio_1$$

$$act_i.ratio_2 = act_i.ratio_2 / act_i.ratio_2$$

$$act_i.ratio_3 = act_i.ratio_3 / act_i.ratio_2$$

step 3 : 자원들에 대한 단위 비용을 step 2 에서 구한 비율로 환원

$$unit.trans_{i,j,k} = act_i.ratio_j \times resource_k$$
 단 k 는 단위비용벡터내 자원에 대한 색인,
 j 는 작업 i 의 $u_d, \overline{m_d}, m_d, o_d$ 에 대한 색인

step 4 : 환원된 단위 비용을 이용한 비용 계산

$$cost_{t,j} = \sum_k act.dur_i \times unit.trans_{i,j,k} \times num_resource_k$$

그림 7 작업비용 계산 알고리즘

m_d 를 이용하여 비율 계산을 수행하고 비관적 작업 기간인 o_d 는 상한 중심값인 $\overline{m_d}$ 를 이용하여 비율 계산을 수행한다. 이는 비율의 편차를 최소화하기 위한 조치이다.

퍼지 작업비용을 계산하는 그림 7은 MOEH 스케줄링 알고리즘의 퍼지 확장을 위한 핵심적 부분으로서 그림 6의 Extended Fuzzy Activity Selector 모듈을 구성하는 주요한 알고리즘이 된다. 그림 8의 알고리즘에서 단계 1은 퍼지수로 환원된 작업기간의 비용기여도를 구하는 부분으로서 $act_i.ratio_j$ 는 퍼지 비용 모델에서의 작업기여도 함수 $ratio()$ 에 의해 구해진 비용기여도 값이고 단계 2에서는 퍼지수로 환원된 각 작업기간의 이익률을 계산하는 부분이며, 단계 3은 단계 2에서 계산된 이익률을 이용하여 각 자원에 적용할 수 있는 단위 비용을 산출한다. 이 때 각 자원은 단계 2에서 계산한 작업기간 사이의 이익률을 자원에 대한 단위비용에 곱함으로써 작업기간의 증감에 따른 비용 증감을 반영할 수 있다. 단계 3은 작업기간 증감에 따른 비용 증감을 반영한 단위 비용을 이용하여 작업의 소요 비용을 계산하는 부분이다.

단계 3에서 첨자 i 는 스케줄러 내의 작업번호를 의미하고 첨자 j 는 작업 내에서 이익률로 환원된 퍼지 작업기간을 색인 한다. 첨자 j 의 번호는 0 ~ 3의 범위를 갖으며 각각 $u_d \sim o_d$ 에 대응하는 작업의 이익률이 된다. 또한 첨자 k 는 자원에 대한 단위비용 벡터 내에서 자원에 대한 식별자 역할을 하는 것으로

서 0 ~ 2의 변위를 갖으며 각각 자원 $R[0]$ 의 단위비용 ~ $R[2]$ 의 단위 비용에 대응한다. 한편 $act_i.ratio_j$ 는 i 번째 작업의 j 번째 작업기간의 이익률을 의미하고 $resource_k$ 는 자원 k 의 단위 비용을 의미한다. 따라서 $unit.trans_{i,j,k}$ 는 작업 i 의 작업기간 j 에 대응하는 자원 k 의 환원된 단위 비용이 된다.

5. 실험 및 평가

5.1 Fuzzy-MOEH 알고리즘 실험

이 장에서는 앞에서 제시한 Fuzzy-MOEH 알고리즘의 동작 특성에 대한 실험을 통하여 그 효율성을 검증한다. 실험을 위한 프로젝트 모델은 선행 제약 및 자원 제약을 갖는 작업으로 구성되었으며 각 작업들은 최대 세 가지 종류의 자원을 필요로 하고 시작시간 창 및 종료시간 창을 갖고 있다. 다음의 표 2는 실험에 이용될 스케줄링 모델을 기술한 것이다.

표 2의 모델에 표시되진 않았으나 퍼지 스케줄링에 있어 중요한 요소로 각 자원에 대한 단위 비용도 중요한 요소이다. 이 단위 비용은 실험 모델에 추가하여 표현한다. 표 2의 모델의 주요한 특징은 작업기간이 L-R 퍼지 함수에 의한 네 개의 정형화된 시간 변수로 확장되었다는 점이다. 이들 시간 변수는 비용과 결합되어 스케줄링 과정에서 중요한 역할을 수행한다. 다음의 표 3은 표 2의 스케줄링 모델을 이용하여 구성한 실험 모델이다.

표 2 Fuzzy-MOEH 스케줄링 모델

번호	기호	의미
1	ID	작업 식별자
2	$u-d_j$	낙관적 퍼지 작업기간
3	$\underline{m}-d_j$	평균 퍼지 작업기간의 하한
4	$\bar{m}-d_j$	평균 퍼지 작업기간의 상한
5	$o-d_j$	비관적 퍼지 작업기간
6	W1	시작시간 창(Start Time Window)
7	W2	종료시간 창(End Time Window)

번호	기호	의미
8	R[0]	R[0] 유형의 자원 요구량
9	R[1]	R[1] 유형의 자원 요구량
10	R[2]	R[2] 유형의 자원 요구량
11	P[0]	첫번째 Immediate Predecessor
12	P[1]	두번째 Immediate Predecessor
13	P[2]	세번째 Immediate Predecessor

표 3 Fuzzy-MOEH 실험 모델

ID	$u-d_j$	$\underline{m}-d_j$	$\bar{m}-d_j$	$o-d_j$	W1	W2	R[0]	R[1]	R[2]	P[0]	P[1]	P[2]
1	2	3	3	4.5	0	0	3	2	1	-1	0	0
2	3.5	5	5.5	6	0	5	2	4	2	-1	0	0
3	2	6	6.5	8	0	9999	3	1	2	1	-1	0
4	2	2	2	3	0	9999	4	3	1	1	-1	0
5	1.5	3	3	4.5	0	9999	2	0	3	4	-1	0
6	2	3	3.5	4	0	9999	1	1	1	4	-1	0
7	2	4	4	5.5	0	9999	3	1	1	2	-1	0
8	2.5	5	5	7	0	9999	2	2	2	6	7	-1
9	3	4	4.5	6	0	9999	3	2	3	3	-1	0
10	1	2	2	3	0	9999	4	1	0	5	9	-1
11	2.5	3	3	4	0	9999	5	4	2	10	-1	0

λ	0.6
-----------	-----

자원타입	단위비용
R[0]	5
R[1]	3
R[2]	6

자원타입	자원총량
R[0]	6
R[1]	7
R[2]	6

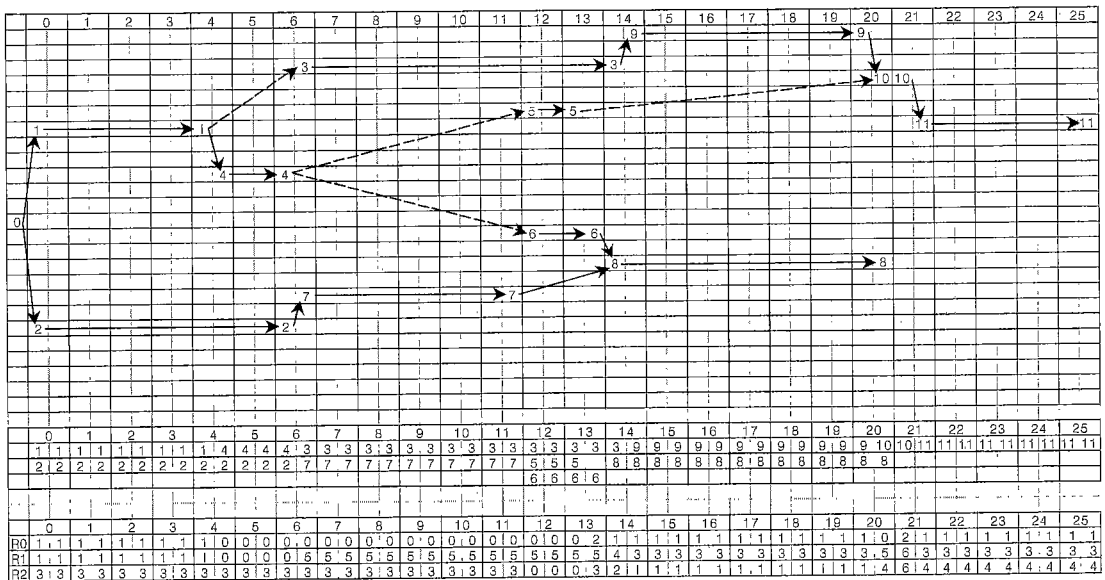


그림 8 Fuzzy-MOEH 스케줄링 상태 네트워크

표 3의 실험 모델은 MOEH 및 Fuzzy-MOEH의 특성을 비교하기 위해 설계된 11개의 작업들로 구성되어 있다. 실험을 실제 데이터를 이용하여 수행하는 것이 알고리즘의 특성을 보다 정확하게 파악할 수 있는 방법이겠으나 실제로 공개되어 있는 프로젝트 수행 정보가 전무할 뿐만 아니라 실제 진행중인 프로젝트의 수행 정보에 대한 획득이 불가능하기 때문에 가상적으로 알고리즘의 특성을 파악할 수 있는 실험 모델을 설계하였다. 이 실험 모델의 설계에 있어 주요한 고려사항은 작업들 사이의 선행 관계 및 자원 제약에 따른 지연 요인 등을 반영할 수 있도록 하는 것이다. 이는 알고리즘이 스케줄링의 수행에 의해 생성된 스케줄의 적합성을 평가할 수 있는 기준이 되기 때문이다. 이와 같이 설계된 실험 모델에서 각각의 작업들에 대한 작업기간을 퍼지 확장하기 위해 L-R 퍼지함수를 이용하였으며 자원량은 최소 세 번 이상의 지연을 갖을 수 있도록 지정하였다. 자원들에 대한 단위 비용과 퍼지 절단 수준 값 λ 는 임의

로 설정한 것으로서 작업 수행기간이 퍼지수로 환원된 작업기간 중 낙관적 또는 비관적 작업기간으로 수행될 경우의 확률이 최소한 60% 이상이 되어야 함을 의미한다. 이 실험 모델을 대상으로 최소비용 정책 실험 결과는 다음 그림 8과 같다.

아울러 표 3의 실험 모델에서 중간 하한인 $m-d_i$ 를 이용한 스케줄링 결과를 다음의 그림 9에 도시하였다. 이는 MOEH 알고리즘과의 비교를 위한 것으로서 Fuzzy-MOEH에서 적용하고 있는 작업기간인 네 개의 Fuzzy 작업기간이 중간 하한 값을 근거로 확장된 것이기 때문에 이 중간 하한 하나의 작업기간만을 이용하여 스케줄링을 수행할 경우 MOEH 알고리즘과 동일한 결과를 얻어낼 수 있기 때문이다. 이 MOEH 알고리즘의 수행은 Fuzzy-MOEH와 동일 한 절차를 거쳐 수행되지만 Fuzzy-MOEH에서 고려하고 있는 확장된 Fuzzy 작업기간을 이용하지 않고 단일의 작업기간만을 이용하며 비용 계산을 포함하지 않는다는 점이 차이가 있다.

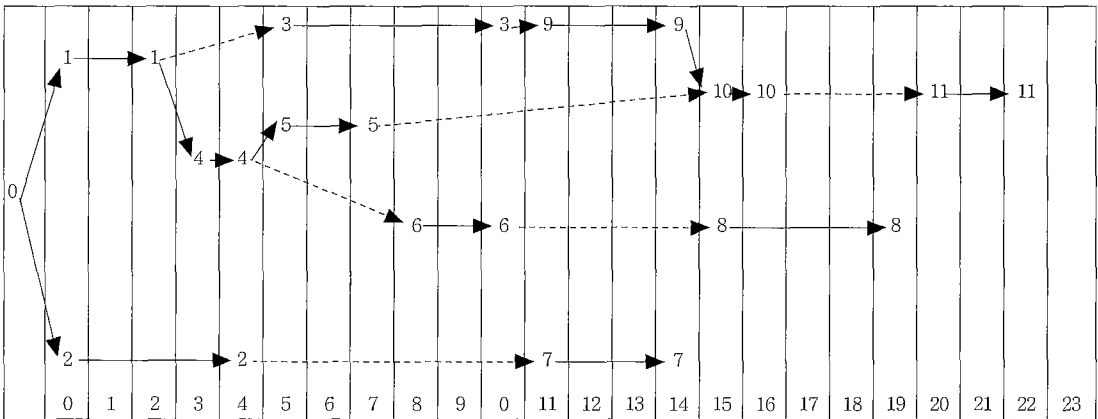


그림 9 MOEH 스케줄링 상태 네트워크

표 4 Fuzzy 비용 모델을 적용한 최소비용 산출 결과

Activity	u_d	m_d	\bar{m}_d	o_d	$R [0]$	$R [1]$	$R [2]$	u_m_{Cost}	\bar{m}_{Cost}	m_{Cost}	o_m_{Cost}
1	2	3	3	4.5	3	2	1	75.36307	81	81	73.69348
2	3.5	5	5.5	6.5	2	4	2	196.19783	170	187	158.35342
3	2	6	6.5	8	3	1	2	227.62007	180	195	145.56736
4	2	2	2	3.5	4	3	1	70	70	70	74.30006
5	1.5	3	3	4.5	2	0	3	69.24629	84	84	76.42286
6	2	3	3.5	5	1	1	1	39.07715	42	49	42.45715
7	2	4	4	5.5	3	1	1	93.49123	96	96	80.06205
8	2.5	5	5	7	2	2	2	161.06831	140	140	100.62976
9	3	4	4.5	6	3	2	3	163.28665	156	175.5	141.92817
10	1	2	2	3	4	1	0	32.09909	46	46	49.44066
11	2.5	3	3	4.5	5	4	1	126.99624	129	129	117.36368

5.2 Fuzzy-MOEH 알고리즘 평가

이 절에서는 앞 절에서의 실험 결과에 대해 정량적 비교 분석을 수행한다. 이를 위해 먼저 MOEH 알고리즘과 Fuzzy-MOEH 알고리즘의 실험 모델에 대한 비용산출을 수행한다. 다음의 표 4은 표 3의 Fuzzy-MOEH 실험 모델에 대해 Fuzzy 비용 모델을 적용하여 계산한 비용 산출 표이다. 비용의 산출은 4.2절에서 제시한 비용 모델을 이용하여 산출하였다.

MOEH 알고리즘에 대한 비용은 작업기간이 표 3의 Fuzzy-MOEH 실험 모델에서 중심구간인 m 과 \bar{m} 에 의해 표현되므로 표 4의 m_{Cost} 와 \bar{m}_{Cost} 에 의해 얻을 수 있다. 이 표 4에서 산출한 비용을 이용하여 Fuzzy-MOEH 알고리즘을 수행할 경우 L-R 퍼지 수 환원 기법에 의해 환원된 네 가지 퍼지수중 가장 적은 비용을 요구하는 작업기간을 선택하게 된다. 따라서 선택된 작업기간을 이용한 스케줄링은 가장 적은 비용이 소요되는 스케줄이 될 수 있다. 표 4에서 반전된 항목들이 선택된 작업기간에 해당한다. 이를 이용하여 프로젝트 수행 비용을 계산할 수 있다. 반면 Fuzzy 개념을 적용하지 않은 MOEH 알고리즘은 평균 작업기간인 m_{Cost} 와 \bar{m}_{Cost} 를 이용하여 얻을 수 있다. 이를 이용한 비용 산출 결과는 다음 표 5와 같다. 한편 작업기간에 대한 비교는 그림 8과 9에 의해 각각 26과 23으로 나타나고 있다. 이는 비용을 고려한 결과로서 비용의 절약에 대한 상보관계에 있음을 확인할 수 있다.

표 5에서 MOEH 알고리즘과 Fuzzy-MOEH 알고리즘에 대하여 각각의 정책을 수행하였을 경우의 결과를 비교하고 있다. 표 5에서 변화율에 대한 표시는 이득이 있을 경우 “ - ” 기호를 이용하여 표현하고 손실이 있을 경우 “ + ” 기호를 이용하여 표현하였다. 이러한 기준을 근거로 표 5를 분석하여 보면 우선 최소비용 정책에

의한 스케줄 결과에서 비용에 대한 절약은 낙관적 평균에 비해 약 14%정도, 비관적 평균에 비해 약 18%정도의 이익을 얻을 수 있다. 반면 작업기간에 대한 손실률은 낙관적 평균에 비해 13%정도, 비관적 평균에 대해 20%정도의 손실이 발생함을 알 수 있다. 마찬가지로 최대 작업비용, 최소작업시간 및 최대작업시간 정책에 대한 평가 결과에 대해서도 수행 결과를 평가할 수 있다. 표 5에서 비관적 평균 작업이 낙관적 평균 작업보다 적은 작업완료시간을 나타내는 것은 MOEH 스케줄링 알고리즘의 특성상 나타나는 것으로서 선행계약과 자원제약을 만족하면서 스케줄에서 작업들의 중첩성을 최대한 반영할 수 있기 때문에 발생하는 현상이다.

표 5의 수행 결과는 표 3의 실험 모델에 대한 실험 결과로서 실험 모델이 바뀌면 다른 결과를 나타낼 수 있다. 그러나 표 5의 결과에서 볼 수 있는 바와 같이 비용 및 정형화된 퍼지수를 이용한 작업기간, 확장을 포함하는 Fuzzy-MOEH 스케줄링 알고리즘이 비용을 고려하지 않고 단일 작업기간만을 고려하는 단순 MOEH 스케줄링 알고리즘보다 훨씬 다양한 정보를 제공할 수 있으며 최종 스케줄 결정에 있어 보다 많은 사항들을 고려할 수 있게 하는 장점이 있다. 아울러 퍼지수로 정형화된 작업기간을 이용한 스케줄 결과는 확장된 퍼지수에 의해 신뢰도를 정량적으로 판단할 수 있게 함으로써 산출된 스케줄에 대해 보다 높은 수준의 신뢰성을 보장할 수 있게 한다.

6. 결론 및 향후 연구

이 논문에서는 소프트웨어 개발 프로젝트의 일정 관리를 위한 스케줄 수립에 적용 가능한 스케줄링 알고리즘을 위해 정형화되지 않은 시간변수를 포함하는 자원 계약 프로젝트 스케줄링에서 불확실한 시간 변수를 정

표 5 MOEH와 Fuzzy-MOEH 알고리즘의 비교

		MOEH		Fuzzy-MOEH	MOEH : Fuzzy-MOEH 변화율	
		m	\bar{m}		m	\bar{m}
최소작업비용	비용	1194	1211.5	1028.0209	- 14%	- 18%
	작업기간	23	21.5	26	+ 13%	+ 20%
최대작업비용	비용	1194	1211.5	1323.12747	+ 11%	+ 09%
	작업기간	23	21.5	22	- 05%	+ 02%
최소작업기간	비용	1194	1211.5	1254.44593	+ 05%	+ 03%
	작업기간	23	21.5	15.5	- 33%	- 28%
최대작업기간	비용	1194	1211.5	1060.21815	- 12%	- 13%
	작업기간	23	21.5	30.5	+ 32%	+ 41%

형화된 퍼지수로 환원하였다. 그리고 이 환원된 퍼지수를 작업기간과 관련된 비용과 결합하여 비용 및 작업기간을 동시에 고려하는 퍼지 개념을 이용한 자원제약 프로젝트 스케줄링 알고리즘을 제시하고 이 구조를 설계 구현하였으며 실험을 통한 성능 및 특성에 대한 평가를 수행하였다. 평가 결과 퍼지수로 환원된 작업기간을 이용한 Fuzzy-MOEH 스케줄링 알고리즘이 기존 MOEH 알고리즘에 비해 보다 다양한 스케줄 정책을 투영할 수 있게 하였으며 특히 퍼지수로 환원된 작업기간은 스케줄에 적용된 작업기간에 대한 신뢰도를 정량적으로 평가할 수 있게 함으로써 스케줄에 대한 신뢰도를 증가시킬 수 있게 하였다.

이 논문에서는 퍼지수로 환원된 작업기간을 이용한 스케줄링 과정에서 작업기간과 결부된 비용을 포함하기 위하여 비용 사상함수로서 비용과 작업기간의 반비례관계를 효과적으로 표현할 수 있는 지수 함수를 이용하였다. 그러나 이 함수는 작업기간의 변화에 민감하며 실제계의 환경을 충분히 수용하지 못하는 단점이 있다. 따라서 실 세계 환경을 보다 정확히 반영할 수 있는 비용 사상함수에 대한 연구가 추후 수행되어야 할 중요 연구 과제이며 아울러 다양한 환경에 대응할 수 있는 다양한 비용 사상함수를 준비하고 관리자에 의해 상황에 맞는 비용 사상 함수를 선택할 수 있도록 확장하는 것과 사용자가 임의의 비용함수를 추가할 수 있는 방법에 대한 연구도 필요하다.

참 고 문 헌

- [1] 안태호, 전송탁, "프로젝트 일정 수립의 비교 분석과 새로운 모형", 프로젝트 관리기술, 23, 1997
- [2] 유동선, "제 6장 PERT", 새로운 OR 원론, 교우사, 1988
- [3] Andrew A. Tiger, "An Exchange Heuristic-Based Scheduler for Multiple Objective Resource Constrained Dynamic Systems," Thesis of Doctoral Degree of Industrial Engineering, Huston University, 1995
- [4] R. S. Pressman, "Part 2 : Software Project Management," Software Engineering-A Practitioner's Approach, McGraw-Hill, 1997
- [5] J. J. Moder, C. R. Phillips, and E. W. Davis, "Project Management with CPM, PERT and Precedence Diagramming," 3rd edition, Van Nostrand Reinhold, 1983
- [6] S. French, "Sequencing and Scheduling-An Introduction to the Mathematics of the Job Shop," Ellis Horwood Limited, Chichster, England, 1982

- [7] Lawler, "Optimal Sequencing of a Single Machine Subject to Precedent Constraints," Management Science, 19, 1973
- [8] M. R. Gary, D. S. Jpnhsen, "Computer and Intratability : A Guide to the Theory of NP-Completeness," Freeman, Sanfrancisco, 1979
- [9] T. Yang, J. P. Ignizio, "An Exchange Heuristic Algorithm for Project Scheduling with Limited Resources," Computer Operation Research, 14, 1987
- [10] Y. Zong, T. Yang, and J. P. Ignizio, "An Expert System Using Exchange Heuristic for the Resource-Constrained Scheduling Problem," Expert Systems With Applications, Vol. 6, 1993
- [11] M. Hapke, A. Jaskiewicz, R. Slowinski, "Fuzzy Project Scheduling system for software development," Fuzzy Sets and Systems, 67, 1994
- [12] R. Slowinski, B. Sonewicki, J. Weglarz, "DSS for multiobjective project scheduling," European Journal of Operational Research, 79, 1994
- [13] M. Hapke, R. slowinski, "Fuzzy priority heuristics for project scheduling," Fuzzy Sets and Systems, 83, 1996
- [14] M. Hapke, P. Kominek, R. Slowinski, "FPS Fuzzy Project Scheduling," URL : www-idss.cs.put.poznan.pl/software/fps.html, 1999
- [15] 변중남, "제 2장 퍼지수학기초", 퍼지 논리제어, 홍릉과학 출판사, 1997



고 장 권

1972년 연세대학교 수학과 학사. 1982년 6월 University of New Heaven Computer & Information Science(공학석사). 2000년 2월 충북대학교 대학원 전자계산학과(이학박사). 1994년 3월 ~ 군장대학 컴퓨터응용학부 부교수. 관심분야는 프로젝트 매니지먼트, 인공지능, 퍼지 이론



신 예 호

1996년 군산대학교 컴퓨터과학과 학사. 1998년 충북대학교 대학원 전자계산학과(이학석사). 2000년 8월 ~ 충북대학교 대학원 전자계산학과 박사과정 수료. 관심분야는 시공간 데이터베이스, 시간 데이터베이스, 능동 데이터베이스, 데이터

마이닝



류 근 호

1976년 숭실대 전산학과 학사. 1980년 연세대학교 산업대학원 전산전공(공학석사). 1988년 연세대학교 대학원 전산전공(공학박사). 1976년 ~ 1986년 육군 군수 지원사 전산실(ROTC 장교). 한국전자통신 연구소(연구원). 한국방송통신대학교 전산학과 조교수. 1989년 ~ 1991년 Univ. of Arizona 연구원(TempIS Project). 1986년 ~ 현재 충북대학교 전기전자컴퓨터공학부 교수. 관심분야는 시간 데이터베이스, 시공간 데이터베이스, 객체 및 지식베이스 시스템, 지식기반 정보검색 시스템, 데이터베이스 보안, 데이터 마이닝, Bio-Informatics



김 홍 기

1961년 연세대학교 수학과 학사. 1975년 연세대학교 교육대학원 응용수학 교육학과(교육학 석사). 1985년 중앙대학교 대학원 응용수학(이학박사). 1980년 ~ 현재 충북대학교 전기전자컴퓨터공학부 교수. 관심분야는 퍼지 이론, 정보통신