

글로벌경영학회지
제18권 제1호 2021년 2월 pp.198~217
글로벌경영학회

자원 제약을 고려한 제안 프로젝트 포트폴리오 문제의 최적해법*

조운재** · 안태호***

〈요 약〉

국내의 소프트웨어시장에서는 시스템 개발, 시스템 유지관리 및 하드웨어 설치 등 다양한 정보 시스템 구축사업이 발주되고 있다. 이에 이를 수주하려는 기업은 가용자원 내에서 사업내용, 예상 수익, 수주가능성 등의 요소를 종합적으로 고려하여 입찰 참여여부를 결정하게 된다. 이는 기업의 경영에 큰 영향을 주는 중요한 의사결정이며 고도의 정밀성과 과학성이 요구된다. 따라서 이러한 의사결정을 지원하기 위하여, 입찰할 사업에 대한 제안 프로젝트를 선택하여 포트폴리오를 구성하는 최적화 수학적모형과 알고리즘의 개발이 필요하다.

본 연구에서는 제안 프로젝트의 특성을 반영한 프로젝트 모드라는 개념을 설정하고 활동별 활동 모드에 따라 활동 기간을 연장함으로써 선형적으로 증가되는 수주확률을 고려하였다. 이를 통해 제한된 자원 하에서 전체 기대가치의 합을 최대화하는 수학적 모형과 최적화 알고리즘을 제시하였으며, 모의실험을 통해 본 알고리즘의 유효성을 입증하였다. 모의실험은 프로젝트의 수, 프로젝트 모드의 수, 활동모드의 수 등 다양한 계수값을 조정하면서 수행하였다. 실행가능한 해가 존재하지 않는 해 영역 또는 현재의 해보다 개선된 해가 존재하지 않는 해 영역을 체계적으로 제거하는 기법인 개선 규칙들을 사용하였다. 이를 통해 수작업으로는 불가능하며 많은 연산시간이 소요되는 프로젝트 포트폴리오 최적화 해법을 비교적 적은 시간 내에 산출하였고, 작은 계수값으로 수행한 일부 모의실험에서는 주어진 제한시간(3,600초)내에 계산결과를 도출하는 결과를 보였다.

본 연구는 활동의 기간이 연장됨에 따라 수주 확률이 선형적으로 증가한다는 가정 하에, 수주 기대가치의 총합을 최대화하는 문제로서, 최적해 도출 알고리즘을 새롭게 제시하였다는 점에 의의가 있다.

주제어 : 제안 프로젝트, 프로젝트 선정, 포트폴리오, RCPS, 최적해법

* 본고는 조운재의 2020년도 박사학위 논문 「자원 제약을 고려한 제안 프로젝트 포트폴리오 최적화 문제」의 일부를 발췌하여 재작성한 것임을 밝힙니다.

** 주저자, 숭실대학교 대학원 프로젝트경영학과 박사 (dawnvits@naver.com)

*** 교신저자, 숭실대학교 경영학부 교수 (ahnt@ssu.ac.kr)

최초접수일 : 2021년 01월 29일 심사수정일 : 2021년 02월 10일 게재확정일 : 2021년 02월 15일

I. 서론

국내의 소프트웨어(Software) 시장에서 다수의 사업들이 발주되고 있으며, 이들 사업에 입찰을 하는 기업은 사업의 내용, 예상수익, 제안비용, 인적자원, 일정 등 다양한 평가기준과 제약조건을 판단하여 입찰(제안) 참여 여부를 결정하게 된다. 다수의 사업 중에서 입찰할 사업을 선택하는 것은 그 기업의 투입 가능한 가용자원(인적·물적 자원)과 해당 사업의 수주 가능성을 감안하여 기대 수익의 최대화를 가져 올 수 있는 사업들의 조합(Combination)을 선정하는 것이며, 이는 고도의 정밀성과 과학성이 요구되는 작업이다.

입찰자는 사업에 입찰하기 위해서는 제안활동을 해야 한다. 여기서 제안활동이란 제안서 작성, 제안 발표 등을 비롯하여 제안에 필요한 제반 과정을 모두 포함하는 것이다. 이러한 일련의 제안 과정은 하나의 프로젝트이며, 본 연구에서 프로젝트라 함은 제안 프로젝트를 지칭한다. 제안 프로젝트는 일반적인 프로젝트와는 다른 몇 가지 특성들이 존재한다. 일반적인 프로젝트에서는 활동 단위로 자원이 투입되고 해제되나, 제안 프로젝트에서는 대부분의 자원이 프로젝트 단위로 투입되고, 해제된다. 이는 프로젝트 초기에 투입된 자원은 프로젝트 종료 때까지 유지된다는 것을 의미한다. 이 밖에 제안 종료일을 기준으로 일정을 역방향으로 수립한다는 점, 여유시간(Slack)을 허용하지 않는다는 점 등이 제안 프로젝트가 갖는 특성들이다. 기존에는 이러한 제안 프로젝트의 특성이 반영된 포트폴리오 최적화에 관한 선행연구는 미흡한 상태이며, 실무에서는 제안 참여에 관한 의사결정은 오직 영업부문의 주장에 의존하는 경향이 많으므로 종합적으로 검토되지 않은 제안활동은 많은 자원의 낭비와 재정적 손실을 초래할 수 있다. 따라서, 과학적인 의사결정 방식에 의한 제안 프로젝트의 선택을 위한 최적화 모델의 개발이 필요한 실정이다.

본 연구의 목적은 제한된 자원으로 기대 수익을 최대화하는 제안 프로젝트의 포트폴리오 최적화를 통해 사업선택의 가능해(Feasible solution)을 도출하는 알고리즘을 구현하여 그 유효성을 입증하고 데에 있다. 본 연구에서는 제안 프로젝트의 특성을 반영하고, 실무 현장의 다양한 평가기준과 제약조건을 최대한 고려하여 제안 프로젝트 포트폴리오 최적화를 위한 산출모형을 찾고 이를 최적화하는 방법을 연구하고자 한다.

II. 이론적 배경

2.1 경쟁 제안관련 선행연구

본 논문에서 제안이란 경쟁제안을 의미하는 것으로써, 공개 입찰경쟁에서 수주를 위한 영업행위의 일환으로 진행되는 제안요청서 분석, 제안전략 수립, 제안서 작성, 제안발표 등의 활동들을 의미한다. 즉, 수주형 비즈니스 환경에서 제안의 의미를 “발주자의 요구사항을 만족시키기 위해 입찰자가 제시하는 공식적인 조건”이라 하였고 발주자의 요구사항에 대해 구체적으로 입찰자의 입장을 밝히는 작업이며 수주를 위한 필수적 활동이라 하였다(박상우, 2013).

Lewis(2015)는 경쟁 입찰, 입찰 및 제안을 통한 계약 및 자금 조달에 대한 기술적 품질과 비용 대비 가치가 뛰어난 제안서를 만드는 방법과 광범위한 조달, 공급 및 입찰에 관하여 설명하였고, I Astana et al.(2015)은 입찰전략과 관련된 다양한 문헌의 관점을 바탕으로 입찰 전략 모델의 개념적 프레임워크를 개발하였다. 또한 Takano et al.(2017)은 입찰자의 예상 이익을 극대화하기 위한 모델에서 최적의 자원 할당 방법을 제시하고 모의실험을 통해 그 효율성을 입증하였다.

2.2 RCPSP에 관한 선행연구

일반적으로 RCPSP(Resource Constrained Project Scheduling Problem)는 활동의 선행관계, 마감기한, 지체보상금, 최대 가용자원 등의 제약조건하에서 활동의 우선순위와 활동모드의 선택에 관한 문제를 다룬다. 이 문제의 해결을 위해 이전 연구에서는 한계열거법, 분지한계법, 선형계획법, 0-1 계획법 등을 사용하였다.

한계열거법은 Davis & Heidorn(1971)가 최적해법을 제안함으로 시작되었다(민택기, 2005). 장점으로 작업의 소요기간에 따라 자원 요구가 변화되며 작업의 연속성에 관한 다양한 가정들이 허용되며, 단점으로 문제의 크기가 증가함에 따라 급격히 해 공간이 증가한다는 점이다(남재덕, 2008). 분지한계법(Branch and Bound)은 Johnson(1967)과 Schrage(1970)에 의해 연구되었으며, 선형계획법은 Stinson et al.(1978)이 노드 선택 휴리스틱을 이용하여 수식화하였다. 이들은 분지한계트리(Branch and Bound Tree)상에서 열등 해를 효율적으로 제거하는 우월 규칙(Dominance Rule)을 수립하여 효율성을 높였다. 0-1계획법은 주어진 제약조건에 대해 0 또는 1로 만드는 변수를 곱하는 것으로 Bowman(1959)이 활동의 수행에 이용하였고, Patterson & Roth(1976)는 순서 문제(Sequencing Problem)의 수식화에 이용하였다. 단일모드 RCPSP의 경우 목적함수는 일반적으로 완료시간의 최소화로 표현되며(남재덕, 2008), 복수모드인 경우는 단일모드와 달리 어떤 모드를 선택하느냐에 따라 프로젝트의 총 기간과 총 비용이 변경되는 특징을 가지고 있다(Horio & Suzuki, 2010). RCPSP 중 많이 연구되고 있는 복수의 모드를 지닌 문제(RCPSP with Multi-Modes, RCPSPMM)의 경우 작업(활동)의 시작시간과 그 실행방법(모드)의 선택을 의사결정 변수로 가지고 있으며, 자원이 한정된 경우를 다루고 있기 때문에 실제 프로젝트 일정수립에 활용될 가능성이 높다(백인섭, 2020).

2.3 프로젝트 선택 및 포트폴리오 관련 선행연구

Archer & Ghasemzadeh(1999)에 의하면 프로젝트 포트폴리오 관리란 특정한 조직의 후원 혹은 관리 하에서 수행되는 프로젝트의 그룹들을 관리하는 것이라 정의하였다. Blichfeldt & Eskerod(2008)는 전체 후보 프로젝트를 우선순위를 통해 필터링하는 것, 필터링된 포트폴리오 내에 있는 프로젝트들을 동시에 다시 우선 순위화하는 것, 그리고 그 우선순위에 따라 프로젝트들에 할당된 자원을 배정하고 재조정하는 것 등 3가지 범주를 사용하여 프로젝트 포트폴리오를 정의하였다. 안태호(2007)는 R&D 프로젝트 포트폴리오 구성문제에 대한 수학적 모형과 휴리스틱 기법을 소개하였다. 여기서 프로젝트들은 자원이 제약되고 복수의 모드를 가지며 포트폴리오는 다기준으로 평가되었다. 김동욱과 이원영(2018)은 프로젝트 포트폴리오 투입인력 최적화를 위하여 유전자 알고리즘에 기반한 휴리스틱 기법을 개발하였다. Hassanzadeh et al.(2014)은 시간에 따른 자원 제약이 있는 프로젝트의 선택 및 일정 문제를 위한 최적화 모델을 제시하였으며, Huang & Zhao(2014)은 R&D 프로젝트로 구성된 포트폴리오에서 선행 프로젝트가 후행 프로젝트의 일정 등에 영향을 끼치는 경우를 다루었다.

III. 문제의 모형

3.1 제안 프로젝트의 특징 및 사례

제안 프로젝트는 일반적인 프로젝트와 대비하여 몇 가지 특징을 갖는다. 첫째, 일반적인 프로젝트에서는 활동 단위로 자원이 투입되지만, 제안 프로젝트에서는 대부분의 자원이 프로젝트 단위로 투입된다. 즉, 제안 프로젝트의 경우에는 활동에 투입된 자원 중 대부분은 프로젝트가 종료될 때까지 해당 프로젝트에 전속된다. 그러므로 제안 프로젝트의 경우, 투입되는 자원을 (1) 프로젝트 전 기간에 투입되는 자원 그룹과 (2) 특정 활동에만 투입될 수 있는 자원으로 구분하는 것이 필요하다. 둘째, 제안 프로젝트의 경우, 제안서 작성이 완료되었다 하더라도 제안 마감일보다 먼저 제안서를 제출할 수 없다. 설령 제안서 완료가 조기에 이루어진다고 하더라도, 제안품질의 향상을 위해 프로젝트에 투입된 자원을 조기에 해산하는 것이 아니라 계속적으로 제안서의 수정 보완 작업에 투입된다. 셋째, 제안 프로젝트는 여유시간(Slack)을 허용하지 않는다. 만일 활동에 여유시간이 발생하는 경우, 해당 활동기간을 다음 활동 시작일까지 연장하여 자원의 유휴 자원화를 허용하지 않

는다.

3.2 문제의 가정

본 문제에 대한 가정은 다음과 같다.

- (1) 참여 가능한 제안 프로젝트의 후보 목록, 각 프로젝트의 발주금액과 제안서 제출 마감, 제안 기간 그리고 수주 확률 등은 사전에 알려져 있다.
- (2) 각 프로젝트의 제안서 제출 마감일은 반드시 준수되어야 한다.
- (3) 각 프로젝트의 활동의 수와 선행관계는 사전에 알려져 있다. 각 프로젝트별로 활동 간 선행 관계는 준수되어야 한다.
- (4) 모든 활동의 여유시간(slack)은 0으로서 여유시간을 허용하지 않는다.
- (5) 각 프로젝트 별로 복수의 프로젝트 모드를 가지며, 하나의 프로젝트 모드만을 배정받는다.
프로젝트 모드에 따라 프로젝트 수행 기간 중 필요한 자원의 종류와 수량은 달라진다.
- (6) 프로젝트의 각 활동 별로 복수의 모드를 가지며, 오직 하나의 모드만을 배정받는다.
- (7) 각 활동의 모드에 따라 활동의 수행 기간과 필요한 자원의 종류와 수량이 달라진다.
- (8) 프로젝트 p의 시작부터 종료까지 매 기간별 필요한 자원의 수는 프로젝트 모드와 해당기간에 수행 중인 활동들의 활동 모드에 따라 결정된다.
- (9) 각 활동의 품질은 수행 기간을 연장하여 향상될 수 있으며, 각 활동의 품질이 향상되면 해당 프로젝트의 수주 확률도 향상된다.
- (10) 각 활동의 모드에 따라 연장 가능한 기간과 수주 확률 증가분이 달라진다.
- (11) 각 활동의 기간은 선정된 모드의 기간과 연장된 기간으로 결정된다.
- (12) 프로젝트 수주 확률 증가분은 각 활동의 수주 확률 증가분의 단순 합이다.
- (13) 포트폴리오가 실행되는 기간 중 매 기간별 가용한 자원의 종류와 수량은 알려져 있다.
- (14) 포트폴리오의 매 기간별 자원별 요구량은 선정된 프로젝트의 매 기간별 자원별 요구량의 합이다. 포트폴리오의 매 기간별 자원별 요구량은 자원의 가용량을 초과할 수 없다.

3.3 문제의 수식화

본 문제에 대한 목적함수와 제약식은 다음과 같다.

$$\begin{array}{ccc}
 \text{M} & \text{a} & \text{x} \\
 \sum_{p=1}^P V_p \times \left[Prob_p + \sum_{q=1}^{Q_p} \left(\sum_{i=1}^{I_p} \sum_{m=1}^{M_{pq}} Prob_{pqim}^+ \times z_{pqim} \times u_{pqim} \right) \times y_{pq} \right] \times x_p & & [1]
 \end{array}$$

subject to

$$\sum_{q=1}^{Q_p} y_{pq} = 1 \quad \forall p, x_p = 1 \quad [2]$$

$$\sum_{m=1}^{M_{pqi}} z_{pqim} = 1 \quad \forall pi, x_p = 1, y_{pq} = 1 \quad [3]$$

$$u_{pqim} \leq Max_Ext_{pqim} \quad \forall pqim, x_p = 1, y_{pq} = 1, z_{pqim} = 1 \quad [4]$$

$$d_{pi} = \sum_{q=1}^{Q_p} \left[\sum_{m=1}^{M_{pqi}} (D_{pqim} + u_{pqim}) \times z_{pqim} \right] \times y_{pq} \quad \forall pi, x_p = 1 \quad [5]$$

$$f_{pi} - (f_{pj} - d_{pj}) = 0 \quad \forall (pi, pj) \in H_p, x_p = 1 \quad [6]$$

$$f_{p, I_p+1} = Due_Date_p \quad \forall p, x_p = 1 \quad [7]$$

$$\sum_{p \in PS_t} \sum_{q=1}^{Q_p} PR_{pqr} \times y_{pq} + \sum_{p \in PS_t} \sum_{i \in AS_{pt}} \sum_{q=1}^{Q_p} \sum_{m=1}^{M_{pqi}} AR_{pqimr} \times y_{pq} \times z_{pqim} \leq C_r \quad \forall t, r \quad [8]$$

$$d_{p,0}, d_{p, I_p+1} = 0 \quad \forall p, x_p = 1 \quad [9]$$

$$f_{p,0} \geq 0 \quad \forall p, x_p = 1 \quad [10]$$

$$x_p, y_{pq}, z_{pqim} \in \{0, 1\} \quad \forall pqim \quad [11]$$

$$u_{pqim} \geq 0 \quad \forall pqim \quad [12]$$

여기서,

P	프로젝트의 수
p	프로젝트 인덱스, $p = 1, 2, \dots, P$
Q_p	p 번째 프로젝트의 프로젝트 모드의 수
q	프로젝트 모드 인덱스, $q = 1, 2, \dots, Q_p$
I_p	p 번째 프로젝트의 활동의 수
i	활동 인덱스, $i = 0, 1, 2, \dots, I_p, I_p+1$; 0과 I_p+1 은 해당 프로젝트의 시작과 종료를 의미하는 가상활동이다.
M_{pqi}	p 번째 프로젝트가 q 번째 프로젝트 모드로 수행될 때 활동 i 가 선택할 수 있는 활동 모드의 수
m	활동 모드 인덱스, $m = 1, 2, \dots, M_{pqi}$

R	자원 종류의 수
r	자원 종류 인덱스, $r = 1, 2, \dots, R$
V_p	p 번째 사업을 수행할 때의 예상 가치
$Prob_p$	p 번째 제안 프로젝트를 기본 품질 수준으로 수행할 때의 사업 수주 확률
$Prob_{pqim}^+$	p 번째 프로젝트가 q 번째 프로젝트 모드로 수행되고 활동 i 가 m 번째 모드로 수행될 때 연장 기간 당 발생하는 수주 확률 증가분
d_{pi}	p 번째 프로젝트의 활동 i 의 실행 기간
D_{pqim}	p 번째 프로젝트가 q 번째 프로젝트 모드로 수행되고 활동 i 가 m 번째 모드로 수행될 때 정상적인 실행 기간
u_{pqim}	p 번째 프로젝트가 q 번째 프로젝트 모드로 수행되고 활동 i 가 m 번째 활동 모드로 수행될 때 D_{pqim} 을 초과한 연장 기간; $u_{pqim} = d_{pi} - D_{pqim}$
x_p	p 번째 입찰 프로젝트의 선정 여부를 나타내는 0-1 변수
y_{pq}	p 번째 프로젝트의 q 번째 프로젝트 모드 선정 여부를 나타내는 0-1 변수
z_{pqim}	p 번째 프로젝트가 q 번째 프로젝트 모드로 수행될 때, 활동 i 가 m 번째 활동 모드로 선정되는 지를 나타내는 0-1 변수
Max_Ext_{pqim}	p 번째 프로젝트가 q 번째 프로젝트 모드로 수행되고 프로젝트 p 의 활동 i 가 m 번째 활동 모드로 배정될 때, 활동 pi 의 연장 기간의 최대값
f_{pi}	프로젝트 p 의 활동 i 의 종료시기를 나타내는 변수
H_p	프로젝트 p 의 선행관계 집합; $(pi, pj) \in H_p$ 면, pi 가 종료되는 시점에 pj 가 시작되어야 한다.
Due_Date_p	프로젝트 p 의 마감일
C_r	r 번째 자원의 가용량
PS_t	시점 t 에 진행 중인 프로젝트들의 집합
AS_{pt}	시점 t 에 진행 중인 프로젝트 p 의 활동들의 집합
AR_{pqimr}	프로젝트 p 가 q 번째 프로젝트 모드로 실행되고 활동 i 의 배정 활동 모드가 m 일 때, 활동 i 의 전 기간에 걸쳐 매 기간별 필요한 r 번째 자원의 수량
PR_{pqr}	프로젝트 p 가 q 번째 프로젝트 모드로 실행될 때, 프로젝트 전 기간에 걸쳐 매 기간별 필요한 r 번째 자원의 수량

본 문제는 프로젝트를 선정하고, 선정된 프로젝트의 프로젝트 모드를 결정하며, 각 활동 별로 활동 모드, 연장 기간과 종료시기를 결정하는 문제이다. 선정 대상이 되는 P 개의 프로젝트 중 p 번째 프로젝트의 선정여부는 0-1 변수인 x_p 로 표현된다. p 번째 프로젝트에서 선정 가능한 프로젝트 모드의 수는 Q_p 로 표기한다. 0-1 의사결정변수 y_{pq} 는 p 번째 프로젝트에서 q 번째 프로젝트 모드를 선택하면 1, 아니면 0의 값을 갖는다. 식 [2]는 프로젝트 별로 선정 가능한 프로젝트 모드 중 오직 하나의 프로젝트 모드만 선택되어야 함을 나타낸다. 식 [2]는 선정된 프로젝트에 대해서만 적용된다. 즉 모든 p 에 대해 적용되는 것이 아니라, $x_p = 1$ 인 경우에만 적용된다. 식 [3]은 프로젝트 p 가 선정되고 프로젝트 모드로 q 가 배정되었을 때, 각 활동 별로 오직 하나의 모드만 배정되어야 함을 나타낸다. 식 [4]는 p 번째 프로젝트의 i 번째 활동이 모드 $pqim$ 으로 배정될 때, 선택 가능한 추가 투입 기간은 그 모드의 최대 투입 기간을 초과할 수 없음을 나타낸다. 식 [5]는 프로젝트 p 의 활동 i 의 수행 기간 d_{pi} 는 선정된 모드 $pqim$ 에서의 정상 기간 D_{pqim} 에 연장한 기간 u_{pqim} 을 더한 값을 나타낸다. 식 [6]은 $(pi, pj) \in H_p$ 이면 활동 pi 가 종료된 이후에나 활동 pj 가 시작할 수 있음을 의미한다. 활동의 여유시간 배제는 선행활동의 종료시기와 후행활동의 시작시기를 일치시키면 된다. 프로젝트 p 의 종료를 의미하는 활동은 $(I_p + 1)$ 이므로, 프로젝트 p 의 종료시기는 f_{p, I_p+1} 이 된다. 식 [7]은 프로젝트 p 가 자신의 마감일인 Due_Date_p 에 종료되어야 함을 나타낸다. 식 [8]은 매 기간별, 자원 종류 별로 포트폴리오에서 요구되는 수량이 가용량을 초과할 수 없음을 나타낸다. 식 [9]는 프로젝트의 시작과 종료를 나타내는 가상활동들의 수행기간이 0임을 나타낸다. 본 연구에서 포트폴리오의 분석 시작 시점을 0이라 가정한다. 그러므로 식 [10]은 모든 프로젝트의 시작시기는 포트폴리오 분석 시작 시점인 0보다 앞설 수 없음을 의미한다. 식 [11]은 x_p, y_{pq}, z_{pqim} 이 0-1 변수임을, 식 [12]는 u_{pqim} 이 비음조건을 지녔음을 나타낸다. [1]은 목적함수로써 제안 프로젝트 포트폴리오의 기대 가치 최대화를 의미한다.

IV. 알고리즘

본 모형에서 의사결정변수는 x_p, y_{pq}, z_{pqim} 및 u_{pqim} 이다. x_p, y_{pq}, z_{pqim} 과 u_{pqim} 의 값에 따라 각 활동들의 수행기간 d_{pi} 와 종료시기 f_{pi} 가 결정된다. 그러므로 d_{pi} 와 f_{pi} 는 종속변수가 된다. 본 연구에서는 본 모형에 적합한 최적해법을 제시하고자 한다.

4.1 해의 영역

해의 영역에서 프로젝트 선정 변수인 x_p 해 영역은 $x_1, x_2, \dots, x_p, \dots, x_P$ 이다.

프로젝트 모드 선정을 의미하는 y_{pq} 에 대한 해 영역은 아래와 같다.

$$\begin{bmatrix} y_{11} & y_{12} & y_{1q} & \cdots & y_{1Q_1} \\ y_{21} & y_{22} & y_{2q} & \cdots & y_{2Q_2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{p1} & y_{p2} & y_{pq} & \cdots & y_{pQ_p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ y_{P1} & y_{P2} & y_{Pq} & \cdots & y_{PQ_P} \end{bmatrix}$$

z_{pqim} 은 활동 i 의 모드 선정을 의미한다. p 번째 프로젝트가 선정되고($x_p = 1$) q 번째 프로젝트 모드가 선정되었을 때($y_{pq} = 1$) 활동 i 의 모드 선정에 국한하여 z_{pqim} 의 부분해를 나열하면 $z_{pq1}, z_{pq2}, \dots, z_{pqim}, \dots, z_{pqM_{pqi}}$ 이 된다. 위와 같은 부분해는 가능한 pqi 조합만큼 생성된다.

u_{pqim} 은 활동의 기간 연장에 관한 변수이다. 기간 연장 상한값은 활동의 모드에 따라 다르고 기간 연장은 정수 단위로만 허용되므로, p 번째 프로젝트가 선정되고($x_p = 1$) q 번째 프로젝트 모드가 선정되며($y_{pq} = 1$) 활동 i 에 m 번째 모드가 선정되었을 때($z_{pqim} = 1$), u_{pqim} 의 부분해를 나열하면 $0, 1, \dots, u_{pqim}, \dots, Max_Ext_{pqim}$ 이다. 위와 같은 부분해는 가능한 $pqim$ 조합만큼 생성되며, x_p, y_{pq}, z_{pqim} 과 u_{pqim} 의 부분해를 모두 연결하면 본 문제의 해가 완성된다.

4.2 최적화 기법

본 연구에서 제안한 최적화 기법은 아래와 같다.

- (1) $x_p, y_{pq}, z_{pqim}, u_{pqim}$ 의 모든 조합을 체계적으로 생성한다. 생성되는 조합을 σ_{xyzu} 라 하자.
- (2) 위에서 구한 각각의 σ_{xyzu} 로부터
 - (2-1) 선정된 프로젝트의 각 활동의 종료시기와 수행기간을 산정한다.
 - (2-2) 자원제약의 준수여부를 검토한다.
 - (2-3) 모든 제약조건을 충족하면, 목적함수값을 구한다.

실행 가능한 σ_{xyzu} 중에 목적함수값을 최대화하는 σ_{xyzu} 가 최적 σ_{xyzu} 가 된다.

4.2.1 종속변수값 결정

독립변수인 x_p , y_{pq} , z_{pqim} 과 u_{pqim} 이 결정되면, 독립변수의 값을 사용하여 종속변수인 d_{pi} 와 f_{pi} 를 구할 수 있다.

(1) 선정된 프로젝트의 활동 기간 d_{pi} 값 결정

주어진 x_p , y_{pq} , z_{pqim} , u_{pqim} 로부터 각 활동의 기간 값을 결정한다. 만약 p 번째 프로젝트가 선정되었다면, i 번째 활동의 기간은 다음과 같다.

$$d_{pi} = \sum_{q=1}^{Q_p} \left[\sum_{m=1}^{M_{pq}} (D_{pqim} + u_{pqim}) \times z_{pqim} \right] \times y_{pq}$$

(2) 선정된 프로젝트의 활동별 종료시기 f_{pi} 값 결정

활동 간의 선행관계와 프로젝트 마감일이 주어졌고(x_p , y_{pq} , z_{pqim} , u_{pqim})로부터 활동들의 수행 기간을 구할 수 있으므로, 각 활동의 종료시기를 결정할 수 있다. 전술한 바와 같이 제안 프로젝트의 특성 상 프로젝트 시작 시점에 제약이 있는 것이 아니라 프로젝트 종료 시점에 제약이 존재하므로, 프로젝트 마감일로부터 역순으로 활동들의 종료시기를 결정한다. 역순으로 결정하므로, j 가 활동 i 의 후행활동이라면, 활동 i 의 종료시기는 후행 활동 j 의 시작시기(종료시기에서 수행 기간을 제한 값)가 결정된 이후에나 확정된다. 프로젝트 p 가 선정된 경우, 활동 i 의 종료시기는 다음과 같고, 프로젝트 종료를 의미하는 가상활동 I_{p+1} 의 경우, 종료시기는 다음과 같다.

$$f_{p, I_{p+1}} = Due_Date_p$$

가상활동(dummy node를 의미)이 아닌 활동 i 의 종료시기는 i 의 후행활동들의 시작시기 중 가장 작은 값과 같다. 이를 수식으로 표현하면 $f_{pi} = \underset{j | (i,j) \in H_p}{MIN} (f_{pj} - d_{pj})$ 이 된다.

4.2.2 목적함수값 산정

독립변수 x_p , y_{pq} , z_{pqim} 과 u_{pqim} 이 결정되면, 목적함수값

$$\sum_{p=1}^P V_p \times \left[Prob_p + \sum_{q=1}^{Q_p} \left(\sum_{i=1}^{I_p} \sum_{m=1}^{M_{pq}} Prob_{pqim}^+ \times z_{pqim} \times u_{pqim} \right) \times y_{pq} \right] \times x_p$$

은 쉽게 구할 수 있다.

프로젝트 \hat{p} 가 선정되면, 목적함수값은 일단 $V_{\hat{p}} \times Prob_{\hat{p}}$ 로 산출된다. 기간 연장의 크기가 \hat{u} 라면, 목적함수값은 $V_{\hat{p}} \times Prob_{\hat{p}}^+ \times \hat{u}$ 만큼 증가한다.

4.3 개선 규칙(Bounding Rule)

프로젝트 모드의 수, 활동의 수, 활동 모드의 수 그리고 선택 가능한 기간 연장의 수가 늘어나면, 가능한 해(σ_{xyzu})의 수는 기하급수적으로 증가하게 하며 전체 가능한 해를 검토하는 데 필요한 연산 시간 또한 기하급수적으로 증가한다. 이후에 소개하는 개선 규칙들은 실행가능한 해가 존재하지 않은 해 영역 또는 현재의 해보다 개선된 해가 존재하지 않는 해 영역을 체계적으로 제거하여, 전체를 탐색하는 데 소요되는 연산 시간을 획기적으로 줄이는 기법들이다.

4.3.1 x_p 를 생성하는 단계에 적용하는 규칙들

(1) 규칙 1 : 포트폴리오 목적함수값의 상한값(Upper Bound)

현재까지 발견된 실행 가능한 해값 중 가장 좋은 해값을 Best_Sol_Value라 하자. 이 규칙은 포트폴리오에서 생성할 수 있는 어떤 기댓값도 이미 알려진 Best_Sol_Value를 초과할 수 없을 때, 해당 포트폴리오 탐색을 중지하는 기법이다. 선정된 프로젝트별로 해당 프로젝트 기댓값의 상한값을 구한다. 예를 들어, \hat{p} 가 선정되었다고 하자. 수주확률 최대값에 예상 가치 $V_{\hat{p}}$ 를 곱한 값이 프로젝트 \hat{p} 의 기대가치 최대값이 된다. 즉, 가장 높은 수주율 증가분만을 취하여 계산한 것이므로 상한값이 되는 것이다. 만일 이렇게 구한 값이 Best_Sol_Value 이하라면 이때 선정된 프로젝트들로 구성된 포트폴리오에서는 이 이상 더 좋은 목적함수값을 나올 수 없음을 보여준다. 따라서, 이 경우 현재의 x_p 조합은 더 이상 검토할 필요가 없으므로 검토를 중지하고, 다음 번 x_p 조합을 생성하는 것이다.

(2) 규칙 2 : 포트폴리오 자원 최소 요구량

자원제약으로 인해 선정된 프로젝트들을 동시 배정할 수 없다면, 해당 포트폴리오를 검토할 필요가 없다는 규칙이다. 이 규칙을 적용하기 위해서는 첫째, 각 프로젝트 별로 프로젝트의 LS(Latest Start Time)과 마감일을, 둘째, 프로젝트가 수행되는 매 기간별로 필요한 자원 r 의 최소 요구량을 계산해야 한다. 이 규칙의 적용은 다음과 같은 단계로 수행한다.

Step-1 : 프로젝트 인덱스 순으로 LS기간부터 마감일까지 최소 자원 요구량은 합산한 후, 가용량 초과여부를 검토한다. 만약 \hat{p} 번째 프로젝트에서 합산 요구량이 가용량을 초과했으면, Goto Step-2

Step-2 : 현 포트폴리오는 자원 제약으로 인해 실행 불가능한 포트폴리오다. 검토할 다음 번 포트폴리오는 아래와 같다.

- 현 포트폴리오에 속한 프로젝트 중 \hat{p} 보다 인덱스가 작은 프로젝트는 다음 번 포트폴리오에 속한다.
- \hat{p} 는 다음 번 포트폴리오에 속하지 못한다.
- 프로젝트 $\hat{p}+1, \hat{p}+2, \dots, P$ 는 모두 다음 번 포트폴리오에 속한다.

4.3.2 x_p, y_{pq}, z_{pqim} 과 u_{pqim} 를 생성하는 단계에 적용하는 규칙들

이 규칙들은 σ_{xyzu} 를 실제 배정하는 단계에 적용하는 규칙들이다. 배정 순서는 다음과 같다. 선정된 프로젝트 중 인덱스가 작은 프로젝트부터 배정한다. 동일 프로젝트에서 활동의 배정 순서는 활동 인덱스가 큰 것부터 한다. 활동 인덱스 역순으로 배정하는 이유는 프로젝트 마감일 이전에 프로젝트 종료 불가능하기 때문이다.

(1) 규칙 1 : σ_{xyzu} 목적함수값의 상한값(Upper Bound)

이 규칙은 현 σ_{xyzu} 배정에서 생성할 수 있는 어떤 기댓값도 이미 알려진 Best_Sol_Value 를 초과할 수 없을 때, 현 σ_{xyzu} 배정을 중지하는 기법이다. 본 연구의 목적함수

$$\text{Max} \sum_{p=1}^P V_p \times \left[\text{Prob}_p + \sum_{q=1}^{Q_p} \left(\sum_{i=1}^{I_p} \sum_{m=1}^{M_{pqi}} \text{Prob}_{pqim}^+ \times z_{pqim} \times u_{pqim} \right) \times y_{pq} \right] \times x_p$$

에서 앞부분, $\sum_{p=1}^P V_p \times \text{Prob}_p \times x_p$ 는 프로젝트 기댓가치의 합이며, 뒷부분 $\sum_{p=1}^P V_p \times \left[\sum_{q=1}^{Q_p} \left(\sum_{i=1}^{I_p} \sum_{m=1}^{M_{pqi}} \text{Prob}_{pqim}^+ \times z_{pqim} \times u_{pqim} \right) \times y_{pq} \right] \times x_p$ 는 선정된 프로젝트에서 활동들의 기댓가치 증가분의 합을 나타낸다. 그런데 프로젝트 선정이 완료된 상태이므로, $\sum_{p=1}^P V_p \times \text{Prob}_p \times x_p$ 는 상수이다. 선정된 프로젝트 인덱스의 집합을 S_Proj 라 정의하면, $\sum_{p=1}^P V_p \times \text{Prob}_p \times x_p = \sum_{p \in S_Proj} V_p \times \text{Prob}_p$ 가 된다. 현재 배정 단계가 프로젝트 \hat{p} 에 활동 \hat{i} 라고 하자. 활동 배정은 프로젝트 인덱스 순으로 이루어지므로, 배정이 완료된 활동들은 (i) 프로젝트 인덱스가 \hat{p} 보다 작은 프로젝트에 속한 활동들과 (ii) 프로젝트 \hat{p} 에서 활동 인덱스가 \hat{i} 이거나 \hat{i} 보다 큰 활동들이다. 이 활동들의 경우, 프로젝트 모드, 활동 모드 및 기간 연장의 결정이 이루어진 상태이다. 아직 미배정 상태인 활동들은 (i) 프로젝트 \hat{p} 에서 활동 인덱스가 \hat{i} 보다 작은 활동들과 (ii) 프로젝트 인덱스가 \hat{p} 보다 큰 프로젝트에 속한 활동들이다. 현재 배정 중인 해(부분해)의 목적함수값의 최대값이 이미 알려진 목적함수의 값 Best_Sol_Value 보다 작거나 같으면, 현재 배정된 부분해는 Best_Sol_Value 보다 향상된 해 값을 생성할 수 없으므로 배정을 중단한다.

(2) 규칙 2 : σ_{xyzu} 자원 제약 검토에서 프로젝트별 자원 최소 요구량을 반영하는 규칙

포트폴리오에서 요구하는 자원의 양이 가용량을 초과하게 되면 자원제약 조건에 위배되어

현재 진행 중인 배정은 실행 불가능한 배정이 된다. 이 규칙을 보다 용이하게 적용하기 위해 매 시점별 그리고 종류별 사용가능한 자원량을 $Avail_{tr}$ (시점 t 에 사용 가능한 자원 종류 r 의 수량)이라 하자. 활동들이 배정되기 전 $Avail_{tr}$ 의 값은 모든 시점 t 별로 C_r 이 된다. 어떤 t 와 r 에 대해서도 $Avail_{tr}$ 은 0 이상의 값이 유지되어야 한다. 활동이 배정될 때마다 그 활동의 시작시점부터 종료시점-1까지 자원 종류별로 AR_{pqimr} 만큼 $Avail_{tr}$ 에서 차감된다. 그러므로 미배정 프로젝트 p 마다 시점 LS_p 부터 Due_Date_p -1까지 매 자원 별로 Min_pr 씩 $Avail_{tr}$ 에서 차감한다. 이때 $Avail_{tr} < 0$ 이라면, 현 부분 배정은 자원 제약을 충족시키는 배정을 생성할 수 없다.

V. 모의실험 결과

5.1 모의실험 조건 설정

5.1.1 데이터 생성 기본구성

본 연구에서 사용한 데이터에서 프로젝트의 수, 프로젝트별 프로젝트 모드의 수, 프로젝트를 구성하는 활동의 수, 활동별 활동 모드의 수, 모드 내에서의 연장 가능한 기간 및 자원의 종류는 다음과 같다.

〈표 1〉 데이터셋 기본구성

기호	구분	값
P	프로젝트의 수	14
Q	프로젝트별 프로젝트 모드의 수	2
I	프로젝트별 활동의 수	4
M	활동별 활동 모드의 수	2
E	모드별 최대 연장 기간	1(활동1~3),2(활동4)
R	자원 종류	6

5.1.2 난수값에 의한 데이터 생성 규칙

(1) 본 사업의 가치, 프로젝트 수주 확률, 마감일과 자원의 가용량 생성

본 사업의 가치, 프로젝트 수주 확률과 마감일은 아래와 같이 일양분포를 사용하여 구했다.

사업의 가치 V 는 최소값이 100, 최대값이 500인 일양분포로부터 추출하였고, 프로젝트 수주확률은 최소값이 0.30, 최대값이 0.70인 일양분포로부터, 마감일 최소값은 60, 최대값이 100인 일양분포로부터 구하였다. 포트폴리오 기간 중 가용한 자원의 양은 일양분포를 사용하여 구했다. 자원의 종류는 6이며, 종류별 하한값과 상한값은 아래와 같다.

$$\text{자원별 하한값} = (8, 5, 8, 5, 5, 5), \quad \text{자원별 상한값} = (11, 7, 11, 7, 7, 7)$$

(2) 자원 생성

각 자원별 요구량은 일양분포를 사용하여 생성하였고, 필요한 자원의 수는 정수가 되도록 하였다. 추가되는 자원의 요구량 또한 일양분포를 사용하여 구하였다. 활동의 활동 모드별 자원 요구량 또한 일양분포를 사용하여 구했다.

(3) 활동 기간 생성

편의상 $D(p_mode, i, a_mode)$ 는 프로젝트 모드가 p_mode 일 때 활동 i 가 모드 a_mode 로 실행될 때의 기간을 의미한다고 하자. 활동 1부터 3의 기간은 다음과 같은 방법으로 구했다.

$$D(1, i, 1) = U(6, 7)$$

$$D(1, i, 2) = D(1, i, 1) \times U(0.8, 0.9)$$

$$D(2, i, 1) = D(1, i, 1) \times U(0.8, 0.9)$$

$$D(2, i, 2) = D(2, i, 1) \times U(0.8, 0.9)$$

활동 4의 기간은 아래와 같은 방식으로 생성되었다.

$$D(1, i=4, 1) = U(16, 40)$$

$$D(1, 4, 2) = D(1, 4, 1) \times U(0.8, 0.9)$$

$$D(2, 4, 1) = D(1, 4, 1) \times U(0.8, 0.9)$$

$$D(2, 4, 2) = D(2, 4, 1) \times U(0.8, 0.9)$$

(4) 활동 모드별 최대 연장 기간

각 활동별 모드마다 수행 기간이 존재한다. 이 수행 기간은 연장될 수 있으며, 기간이 연장되면 프로젝트의 품질이 향상되어, 결과적으로 프로젝트 수주확률도 그만큼 높아진다. 활동 1, 2 그리고 3의 경우 1기간 연장이, 활동 4의 경우 2기간 연장이 가능하다. 1 기간 연장에 따른 수주 확률 증가분은 아래와 같은 일양분포를 사용하여 생성하였다.

$$\text{활동 1~3의 경우, } U(0.02, 0.04), \quad \text{활동 4의 경우, } U(0.025, 0.075)$$

5.2 모의실험 결과

각각의 데이터 셋은 10개의 문제로 구성되어 있다. 연산 시간은 최대 3,600초까지만 허용하였다. <표 1>처럼 기준이 되는 데이터 셋을 Basic 데이터셋이라 할 때, Basic 데이터셋 문제들의 최적해를 구하는 데 소요된 시간(sec)과 최적해값은 아래 표 <표 2>와 같다.

〈표 2〉 Basic 데이터셋의 모의실험 결과

문제 번호	연산시간(초)	최적해 값
1	311	799
2	269	899
3	108	753
4	528	1,081
5	983	756
6	13	720
7	19	553
8	22	600
9	171	956
10	53	703
평균	247.7	
최대	983	
최소	13	

Basic 데이터셋의 경우, 10문제 모두 제한시간 3,600초 이내에 최적해 값을 찾았으며, 연산 시간의 평균은 247.7 초였다. 본 모의실험에서는 프로젝트의 갯수, 프로젝트 모드의 갯수, 활동모드의 개수 및 가용량크기의 4가지 계수값들의 변화가 연산 시간에 어떤 영향을 끼치는 지를 파악하기 위해 각각의 계수값들을 가감, 조정하여 Basic 데이터셋의 연산 시간과 비교하였다.

4개의 모의실험 결과는 순서대로 <표 3>, <표 4>, <표 5>, <표 6>과 같다.

〈표 3〉 프로젝트(P) 갯수가 12, 14, 16인 데이터셋 모의실험 결과

문제 번호	P=12		Basic데이터셋(P=14)		P=16	
	연산시간	최적해값	연산시간	최적해값	연산시간	최적해값
1	109	732	311	799	275	833
2	4	524	269	899	6	553
3	23	867	108	753	705	1,068
4	306	622	528	1,081	29	604
5	25	723	983	756	3,365	1,040
6	21	782	13	720	551	856
7	2	706	19	553	242	702
8	26	736	22	600	2,046	1,043
9	47	811	171	956	94	898
10	17	757	53	703	3,600+*)	N/A
평균	58		248		N/A	
최대	306		983		3,600+	
최소	2		13		6	

*) 연산 제한 시간 3,600초를 초과함을 의미한다.

〈표 4〉 프로젝트 모드(PM) 갯수가 1, 2, 3인 데이터셋 모의실험 결과

문제 번호	PM=1		Basic데이터셋(PM=2)		PM=3	
	연산시간	최적해값	연산시간	최적해값	연산시간	최적해값
1	604	803	311	799	64	845
2	6	1,024	269	899	628	709
3	9	896	108	753	266	783
4	10	760	528	1,081	14	730
5	8	714	983	756	902	854
6	4	634	13	720	3,600+*)	1,359
7	79	804	19	553	115	715
8	393	895	22	600	17	721
9	9	876	171	956	54	853
10	9	728	53	703	32	723
평균	113		248		N/A	
최대	604		983		3,600+*)	
최소	4		13		14	

*) 연산 제한 시간 3,600초를 초과함을 의미한다.

〈표 5〉 활동모드(AM)의 개수가 1, 2, 3인 데이터셋 모의실험 결과

문제 번호	AM=1		Basic데이터셋(AM=2)		AM=3	
	연산시간	최적해값	연산시간	최적해값	연산시간	최적해값
1	1	647	311	799	3,600+*)	N/A
2	16	731	269	899	104	852
3	2	612	108	753	3,600+*)	N/A
4	34	758	528	1,081	3,600+*)	N/A
5	-	667	983	756	3,600+*)	N/A
6	2	1,150	13	720	2,439	914
7	4	505	19	553	171	863
8	1	500	22	600	202	808
9	12	1,003	171	956	69	778
10	1	710	53	703	1,475	945
평균	7		248		N/A	
최대	34		983		3,600+*)	
최소	-		13		69	

*) 연산 제한 시간 3,600초를 초과함을 의미한다.

〈표 6〉 가용자원(CAP)의 크기를 -1 한 경우 및 +1한 경우 모의실험 결과

문제 번호	CAP-1		Basic데이터셋		CAP+1	
	연산시간	최적해값	연산시간	최적해값	연산시간	최적해값
1	6	777	311	799	2,691	999
2	15	634	269	899	1,219	967
3	44	740	108	753	176	1,011
4	168	847	528	1,081	3,600+*)	N/A
5	301	590	983	756	3,600+*)	N/A
6	7	562	13	720	543	844
7	2	415	19	553	93	623
8	1	557	22	600	249	832
9	28	892	171	956	1,314	1,255
10	14	631	53	703	716	798
평균	59		248		N/A	
최대	301		983		3,600+*)	
최소	1		13		93	

*) 연산 제한 시간 3,600초를 초과함을 의미한다.

상기 4개의 모의실험 결과에서 보듯이 공통적인 경향은 P, PM, AM, CAP의 크기를 Basic 데이터셋에 비해 감소시킨 연산 시간은 각각 23%, 46%, 3.5%, 24%로써 평균 24.1% 감소하였고, 반면 Basic 데이터셋 대비 계수값의 크기를 증가시킨 경우, 연산 시간은 크게 증가하였으며, 제한 시간(3,600초) 이내에 풀지 못한 문제도 다수 발생하였다. 따라서 본 모의실험을 통해 제안 프로젝트 포트폴리오를 구성하는 최적해 방식의 알고리즘이 유효성을 가진다고 말할 수 있다.

VI. 결 론

본 논문은 주어진 제안 프로젝트의 제출 마감일 내에서, 각 제안 프로젝트의 활동에 대한 기간을 연장하거나 자원의 추가투입 시 결과물의 품질이 향상되어 수주확률(수주가능성)이 높아진다는 가정에서 출발하였다. 이로써, 기대수익을 최대화하는 프로젝트들의 조합에 대한 RCPSPP 모형과 최적화 알고리즘을 제시하였고, 모의실험을 통하여 제시된 알고리즘이 유효하다는 것을 입증하였다. 이때 모든 해 영역을 검토하는 데에 있어, 실행가능한 해가 존재하지 않은 해 영역 또는 현재의 해보다 개선된 해가 존재하지 않는 해 영역을 체계적으로 제거하였으며, 개선규칙들을 통하여 전체를 탐색하는 데 소요되는 연산 시간을 줄여 연산의 효율을 높였다. 특히, 프로젝트 모드라는 새로운 독립변수를 설정하고 RCPSPP 기법을 통해 프로젝트 선택(Project Selection)문제와 포트폴리오 문제를 병렬시키는 방안을 모색한 것은 기존 선행연구에는 없는 진일보한 방식이다.

본 연구는 막대한 수의 전체 해 영역에서 기대가치의 총합을 최대화하는 조합을 찾는 프로젝트 포트폴리오 최적해 도출 알고리즘을 제시하되, 상기한 제안 프로젝트의 특징을 반영하여 새롭게 제시하였다는 점, 수작업 또는 Excel 수식으로는 불가능한 작업이 본 알고리즘을 통해 적정한 연산 시간 내에 최적해를 도출했다는 점에서 학술적 의의가 있다. 본 알고리즘을 바탕으로 적절한 사용자 Interface를 가진 의사결정지원시스템(DSS, Decision Support System)기반의 제안 프로젝트 포트폴리오 프로그램이 개발된다면 실용적으로 제안참여 의사결정에 많은 도움이 될 것이다. 다만, 수수 가능성의 변동, 제안 중 입찰의 포기 등 제안 프로젝트의 불확실성을 반영한 수학적 모형 개발 등과 같이 본 연구에서 다루지 못한 부분에 대한 추가적인 연구가 필요하다.

“본 게재논문에 대한 윤리적 문제는 전적으로 저자에게 책임이 있음을 확인함”

참 고 문 헌

- 김동욱 · 이원영 (2018), “유전자 알고리즘을 이용한 프로젝트 포트폴리오 투입인력 최적화 모델에 관한 연구”, *한국IT서비스학회지* 제17권 제4호, 101-117
- 남재덕 (2008), “프로젝트 포트폴리오 평가 문제: 불확실성 하의 최초 구성 및 주기적 개장을 중심으로”, *숭실대학교 대학원 박사학위논문*.
- 민택기 (2005), “프로젝트 포트폴리오 평가문제: 시뮬레이션을 이용한 다속성 평가를 중심으로”, *숭실대학교 대학원 박사학위논문*.
- 박상우 (2013), “수주의 기술”, *비즈니스아츠*, 95
- 백인섭 (2019), “자원 제약을 고려한 프로젝트 일정문제 : 재작업 가능한 활동이 있는 경우”, *숭실대학교 대학원 박사학위논문*.
- 안태호 (2007), “R&D 프로젝트 포트폴리오 구성 휴리스틱 기법”, *숭실대학교 사회과학연구원 사회과학논총* 제9집, 183-195.
- Archer, N. P., & Ghasemzadeh, F. (1999). “An integrated framework for project portfolio selection”, *International Journal of Project Management*, 17(4), 207-216.
- Astana, I., Rusdi, H. A., & Wibowo, M. A. (2015). “Conceptual Framework Of Bidding Strategy In Order To Improve Construction Project Performance”, *In Applied Mechanics and Materials* (Vol. 776, pp. 108-113). Trans Tech Publications Ltd.
- Blichfeldt, B. S., & Eskerod, P. (2008). “Project portfolio management - There’s more to it than what management enacts”, *International Journal of Project Management*, 26(4), 357-365.
- Bowman, E. H. (1959). “The schedule-sequencing problem”, *Operations Research*, 7(5), 621-624.
- Davis, E. W., & Heidorn, G. E. (1971). “An algorithm for optimal project scheduling under multiple resource constraints”, *Management Science*, 17(12), B-803.
- Hassanzadeh, F., Modarres, M., Nemati, H. R., & Amoako-Gyampah, K. (2014). “A robust R&D project portfolio optimization model for pharmaceutical contract research organizations”, *International Journal of Production Economics*, 158, 18-27.
- Horio, M., & Suzuki, A. (2010). “A Solution Method for Manpower Scheduling Problems by RCPSP/ τ ”.
- Huang, X., & Zhao, T. (2014). “Project selection and scheduling with uncertain net income and investment cost”, *Applied Mathematics and Computation*, 247, 61-71.
- Johnson, T. J. R. (1967). “An algorithm for the resource constrained project scheduling problem” (Doctoral dissertation, Massachusetts Institute of Technology).

- Lewis, H. (2015). *Bids, tenders and proposals: winning business through best practice*. Kogan Page Publishers.
- Patterson, J. H., & Roth, G. W. (1976). "Scheduling a project under multiple resource constraints: a zero-one programming approach", *AIIE transactions*, 8(4), 449-455.
- Schrage, L. (1970). "Solving resource-constrained network problems by implicit enumeration – nonpreemptive case", *Operations Research*, 18(2), 263-278.
- Stinson, J. P., Davis, E. W., & Khumawala, B. M. (1978). "Multiple resource - constrained scheduling using branch and bound", *AIIE Transactions*, 10(3), 252-259.
- Takano, Y., Ishii, N., & Muraki, M. (2017). "Multi-period resource allocation for estimating project costs in competitive bidding", *Central European Journal of Operations Research*, 25(2), 303-323.

The Exact Solution Procedure for Resource Constrained Proposal Project Portfolio Problem

Cho, Yoon Jae* · Ahn, Tae Ho**

〈Abstract〉

In the domestic software market, various information system construction projects such as system development, system maintenance and hardware installation are being ordered. It is a very important decision to establish a portfolio of bidding projects and to participate in the project by judging various factors such as business contents, expected profits, and possibility of winning orders within the available resources held by the company, and it requires a high degree of precision and scientificity. Therefore, to support such decision-making, it is necessary to develop an optimized mathematical model and algorithm that composes a portfolio by selecting a proposal project for a bidding project.

Setting the concept of project mode reflecting the characteristics of the proposal projects, this study considers the probability of winning orders linearly increased by extending the duration of activity according to the activity mode. Based on the considerations, this study presents mathematical models and optimization algorithms that maximize the sum of total expectations under resource constraints, and demonstrate the effectiveness of this algorithm through simulation experiments. The simulation was performed by adjusting various coefficient values, such as the number of projects, the number of project modes, and the number of activity modes. The bounding rules, a technique for systematically eliminate solution space where no viable solution exists or solution space that do not have an improved solution than the current one, were used. Through this, the project portfolio optimization solution, which is impossible by manual and takes a lot of computation time, was calculated in a relatively small amount of time. In some simulations performed with a small coefficient value, the calculation result was derived within the limited time (3,600 seconds).

This study is meaningful in that it presents a new algorithm for deriving optimally as a problem that maximizes the sum of order expectations, assuming that the probability of winning orders increases linearly as the duration of the activity extends.

Key Words : Proposal Project, Project Selection, Portfolio, RCPSP, Optimal Solution

* 1st author, Ph.D. The Dept. of Project Management, Soongsil University

** Corresponding author, Ph.D. Professor, The Dept. of Business Administration, Soongsil University