

자원 제약을 고려한 제안 프로젝트 포트폴리오 문제의 휴리스틱 기법

조 윤 재(제1저자)

승실대학교 대학원 프로젝트경영학과 (박사)

백 인 섭(공동저자)

승실대학교 대학원 경영학과 (박사)

김 종 출(공동저자)

승실대학교 대학원 경영학과 (박사과정)

안 태 호(교신저자)

승실대학교 경영학부 (교수)

The Heuristic Procedure for Resource Constrained Proposal Project Portfolio Problem

Cho, Yoon Jae(First Author)

The Dept. of Project Management, Soongsil University (Ph.D.)

Paek, In Soub(Co Author)

The Dept. of Business Administration, Soongsil University (Ph.D.)

Kim, Jong Chool(Co Author)

The Dept. of Business Administration, Soongsil University (Doctoral Student)

Ahn, Tae Ho(Corresponding Author)

The Dept. of Business Administration, Soongsil University (Professor)

Abstract

With regards to the various project to establish information system in the Korean software market, a company intending to win such project considers comprehensively the business detail, expected profits, and probability to win within available resources and comes to determine if to make an offer. Such judgement is one of the most important decision making affecting the overall business management like sales and revenue. Therefore, when the proposed project portfolio is defined as the composition of the project decided to be offered among the ordered projects, Cho, Yoon-Jae(2020) suggests a mathematical model of the problem of maximizing the sum of expected values in constituting the proposed project portfolio and an optimal solution accordingly.

This paper assumes the concept of a project mode to be following: 1) the resources are limited, 2) a linear progression of the probability of the orders to be awarded as the duration of each activity increases, 3) the nature of the proposed project is reflected. Unlike previous studies, this research proposes a heuristic technique as a solution to the composition of the proposed project portfolio. The simulation in this study verifies the effectiveness and efficiency of the proposed algorithm as well.

Keywords : Proposal Project, Portfolio, RCPSP, Project Selection, Heuristic Procedure

접수일(2021년 02월 10일), 수정일(2021년 03월 10일), 게재확정일(2021년 03월 12일)

I. 서 론

국내 소프트웨어(Software) 시장의 경우 시스템 구축, 시스템 운영 및 유지보수 등 다양한 정보시스템 구축사업들이 발주되고 있다. 해당 사업을 수주하려는 기업은 가용한 자원 내에서 해당 사업의 내용, 예상되는 기대수익, 수주확률, 제안에 필요한 비용, 제안과 사업에 필요한 인적자원, 제안 일정 등 다양한 제약조건과 판단기준을 종합적으로 고려하여 해당 사업에 대한 제안 참여 여부를 결정을 하게 된다. 이와 같은

결정은 기업 경영에 있어서 매출과 수익에 직접적으로 관여되는 의사결정이며, 기업의 경영 전반에 영향을 주는 중요한 의사결정 중의 하나이다. 제안하는 기업은 발주된 다수의 사업 중에서 투입 가능한 자원 내에서 기대수익을 최대화할 수 있는 사업들의 조합을 고려하여 제안할 사업을 결정해야 하며, 제안할 사업이 결정되면 제안을 위한 활동을 해야 한다. 이때 제안을 위한 활동이란 제안서 작성, 제안발표 등 사업의 제안에 필요한 모든 과정을 모두 포함한다. 이러한 일련의 과정은 해당 사업과 별도로 하나의 프로젝트로 볼 수 있으며, 본 연구에서는 이를 제안 프로젝트라 지칭한다.

발주된 사업 중에서 제안할 프로젝트를 결정하여 구성하는 것을 제안 프로젝트 포트폴리오라 할 때, 조운재(2020)는 제안 프로젝트 포트폴리오 구성의 기대가치 합을 최대화하는 문제의 수학적 모형과 최적해법을 제시하였다. 이 연구에서는 제안 프로젝트의 특성이 반영되었으며, 활동별 기간이 증가함에 따라 선형적으로 수주확률이 증가한다는 가정 하에 자원의 제약이 있는 경우를 다루었다.

이에 본 연구에서는 이전 연구와 달리, 제안하는 기업의 기대수익 최대화를 위한 제안 프로젝트 포트폴리오의 구성 문제에 대한 해법으로 휴리스틱 기법을 제시하고자 하며, 모의실험을 통해 알고리즘의 유효성 입증하고자 하였다.

II. 이론적 배경

2.1 제안 프로젝트에 관한 선행연구

본 연구에서 제안이란 공공사업에 대한 공개 입찰경쟁에서 사업의 수주를 위해 제안요청서(RFP)의 분석, 제안전략 수립 등의 과정을 거쳐 제안서의 작성과 제안발표 등의 수주를 위한 활동을 하는 것을 의미한다.

관련 선행연구에서 박상우(2013)는 수주형 비즈니스 환경에서의 제안에 대한 의미가 “발주자의 요구사항을 만족시키기 위해 입찰자가 제시하는 공식적인 조건”이라 하였으며, 발주자의 요구사항에 대하여 구체적으로 입찰자의 입장을 밝히는 작업이라 하였으며, 이는 수주를 위한 필수 활동이라 하였다. Lewis(2015)는 경쟁 입찰, 입찰과 제안을 통한 계약, 자금 조달에 관해 기술적 품질과 비용 대비 가치가 높은 제안서를 만드는 방법, 광범위한 조달, 공급과 입찰에 관하여 기술하였고, Astana et al.(2015)은 입찰의 전략과 관련한 다양한 문헌적 관점을 바탕으로 입찰 전략 모델에 대한 개념적 프레임워크를 개발하였다. 또한, Takano et al.(2017)은 입찰자 예상 이익 극대화 모델에서 최적의 자원 배정 방법을 제안하고 모의실험을 통하여 그 효율성

을 입증하였다.

2.2 RCPSP에 관한 선행연구

일반적으로 자원 제약을 고려한 일정 문제(Resource Constrained Project Scheduling Problem, RCPSP)는 활동의 선행관계, 지체보상금, 단위 기간별 가용자원 등의 제약조건이 있는 경우 활동의 우선순위와 모드선택에 관한 문제이다.

한계열거법은 작업의 소요기간에 따라 자원의 요구가 변화되며, 작업의 연속성 관련 다양한 가정들이 허용되는 장점과 문제 크기의 증가에 따라 급격한 해 공간의 증가가 있다는 단점을 지닌 방법이며(남재덕, 2008), Davis & Heidorn(1971)가 제안한 최적해법으로 시작되었다(민택기, 2005). Johnson(1967)과 Schrage(1970)에 의해 분지한계법(Branch and Bound)이 연구되었으며, Stinson et al.(1978)은 우월 규칙(Dominance Rule)의 수립을 통하여 분지한계나무(Branch and Bound Tree)상의 열등 해들을 효율적으로 제거하였다. 0-1계획법은 주어진 제약조건에 0 또는 1의 값을 가진 변수를 곱하는 방법으로, Bowman(1959)은 활동 수행에 이용하였으며, Patterson & Roth(1976)의 경우 순서 문제(Sequencing Problem)를 수식화하는데 이용하였다.

RCPSP가 단일모드(Single Mode)인 경우 일반적으로 목적함수는 완료 시간 최소화로 표현되고(남재덕, 2008), 복수모드(Multi Mode)인 경우 모드의 선택에 따라 프로젝트 총 비용과 총기간이 변경되는 특징을 지니고 있다(Horio & Suzuki, 2010).

RCPSP에서 많이 연구되고 있는 복수모드의 자원 제약을 고려한 일정 문제(RCPSP with Multi-Modes, RCPSPMM)는 자원 제약 하에서 활동의 시작 시점과 그 수행 모드의 선택에 대한 의사결정을 다루고 있으므로 매우 현실적이다(백인섭, 2020).

2.3 프로젝트 선택 및 포트폴리오에 관한 선행연구

Archer & Ghasemzadeh(1999)는 프로젝트 포트폴리오 관리를 특정한 조직의 후원 하에 수행되는 프로젝트의 그룹을 관리하는 것이라 정의하였으며, Blichfeldt & Eskerod(2008)는 프로젝트 포트폴리오를 전체 프로젝트를 우선순위를 통해 정렬하는 것, 정렬된 포트폴리오 내에 있는 프로젝트들을 다시 우선 순위화하는 것, 마지막으로 우선순위에 따라 할당된 자원을 프로젝트들에 배정하는 것 등의 범주를 통하여 정의하였다. 안태호(2007)는 연구개발 프로젝트 포트폴리오 구성에 관한 수학적 모형과 휴리스틱 기법을 소개하였으며, 여기서 포트폴리오는 다기준으로 평가되었다. 김동욱

과 이원영(2018)은 포트폴리오 투입인력 최적화를 위해 유전자 알고리즘에 기반한 휴리스틱 기법을 제안하였다. 또한, Hassanzadeh et al.(2014)은 시간 변화와 자원 제약에 따른 트레이드오프 문제에 대해 프로젝트의 선택 및 일정의 최적화에 대한 모델을 제시하였고, Huang & Zhao(2014)은 연구개발 프로젝트 포트폴리오에서 선행된 프로젝트가 후행되는 프로젝트의 일정 등에 영향을 주는 경우를 다루었다. 상기 주요 선행연구를 표로 정리하면 <표 1>과 같다.

<표 1> 주요 선행연구의 주요 내용 및 적용기법

분야	주요 내용	측정문항	비고
연구개발 프로젝트 (R&D Project)	다속성을 가진 연구 프로젝트들이 제한된 자원을 놓고 경합할 때 최적의 구성방안 제시	휴리스틱 기법	안태호
	연구 프로젝트의 수행 비용 추정치의 불확실성을 반영한 트레이드오프 문제	—	Hassanzad eh et al.
	연구 프로젝트간의 상호작용을 반영한 포트폴리오 최적화 모델 제시	휴리스틱 기법	Huang & Zhao
IT 프로젝트	전문 인력의 투입비용, 균등 부하 등의 요소를 반영한 투입인력의 최적화에 관한 모델	휴리스틱 기법	김동욱& 이원영

III. 문제의 모형

3.1 제안 프로젝트의 특징 및 사례

일반적으로 공공사업의 제안 프로젝트는 일반적인 프로젝트와 몇 가지 다른 특징을 갖는다. 첫째, 일반적인 프로젝트에서는 자원이 활동 단위로 투입되나, 제안 프로젝트에서는 대부분 자원이 프로젝트 단위로 투입된다. 즉, 제안 프로젝트 A의 활동1에 투입된 자원이 있을 경우 그 자원은 대부분의 경우 프로젝트 A의 종료 시점까지 프로젝트 A에 전속된다. 따라서 제안 프로젝트에서 투입된 자원은 (1) 일반 프로젝트의 경우와 같이 특정한 활동에만 투입될 수 있는 자원과 (2) 해당 프로젝트 전 기간에 전속되어 투입되는 자원 그룹으로 구분하는 것이 필요하다. 둘째, 제안 프로젝트에서 제안서 작성 완료 시점은 제안 마감일보다 빠를 수 없다. 왜냐하면, 제안서의 작성 완료가 조기에 이루어질 때도 자원은 제안의 품질 향상을 위해 제안 마감일까지 지속해서 수정 보완 작업에 투입되기 때문이다. 셋째, 제안 프로젝트에서는 여유 기간(Slack)을 허용하지 않는다. 만일 활동 간의 여유 기간이 발생하면 이전 활동의 종료

시점을 다음 활동의 시작 시점까지 연장하여 자원의 유희를 허용하지 않는다.

3.2 문제의 가정

본 문제의 가정은 다음과 같다.

- (1) 제안 가능한 프로젝트의 목록, 각 사업의 발주금액, 제안 마감일, 수주확률 등은 사전에 알려져 있다.
- (2) 모든 제안 프로젝트의 종료 시점은 해당 사업의 제안 마감일을 넘을 수 없다.
- (3) 모든 제안 프로젝트에서 활동의 수와 각 활동 간의 선행관계는 사전에 알려져 있다. 또한, 각 제안 프로젝트별 활동 간 선행관계는 반드시 준수되어야 한다.
- (4) 제안 프로젝트 내에서 활동 간의 여유 기간(slack)은 없다.
- (5) 각각의 제안 프로젝트는 필요한 자원의 종류와 수량이 다른 복수의 프로젝트 모드를 가지고 있으며 제안 프로젝트별 하나의 프로젝트 모드만을 배정받는다.
- (6) 제안 프로젝트에서 각 활동은 수행 기간과 필요자원의 종류와 수량이 다른 복수의 모드를 가지고 있으며 활동 별 하나의 모드만을 배정받는다.
- (7) 제안 프로젝트에서 시작 시점부터 종료 시점까지 매 기간별 필요자원의 수는 해당 기간에 배정된 프로젝트 모드와 활동 모드에 따라 결정된다.
- (8) 제안 프로젝트에서 활동의 품질은 수행 기간이 연장됨에 따라 향상될 수 있으며, 활동의 품질이 향상되는 경우 해당 제안 프로젝트의 수주확률도 향상된다.
- (9) 제안 프로젝트에서 활동은 선정되는 모드에 따라 연장할 수 있는 기간과 그에 따른 수주확률 증가분이 달라진다.
- (10) 활동의 수행 기간은 선정된 모드의 수행 기간과 연장으로 인해 추가된 기간으로 결정된다.
- (11) 활동의 연장 기간과 수주확률의 증가분은 선형관계를 가진다.
- (12) 수주확률 증가분의 총합은 해당 제안 프로젝트에서 각 활동에 대한 수주확률 증가분의 합이다.
- (13) 제안 프로젝트 포트폴리오가 실행되는 모든 기간에서 매 기간별 가용자원의 종류와 수량은 알려져 있다.
- (14) 제안 프로젝트 포트폴리오가 실행되는 모든 기간에서 매 기간별 각 자원 요구량은 선정된 제안 프로젝트들의 매 기간별 각 자원 요구량의 합이며, 매 기간별 각 자원의 요구량은 가용자원의 수량을 초과할 수 없다.

3.3 문제의 수식화

본 문제에서 목적함수와 제약식은 다음과 같다.

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

subject to

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

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

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

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

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

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

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

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

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

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

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

여기서,

P	후보 제안 프로젝트의 수
p	후보 제안 프로젝트의 인덱스, $p = 1, 2, \dots, P$
Q_p	p 번째 후보 제안 프로젝트에서 프로젝트 모드의 수
q	후보 제안 프로젝트에서 프로젝트 모드의 인덱스, $q = 1, 2, \dots, Q_p$
I_p	p 번째 후보 제안 프로젝트에서 활동의 수
i	활동 i 의 인덱스, $i = 0, 1, 2, \dots, I_p$, I_p+1 에서 0과 I_p+1 은 해당 제안 프로젝트의 시작과 종료를 의미하는 가상의 활동이다.

M_{pqi}	p 번째 제안 프로젝트를 q 번째의 프로젝트 모드로 수행할 때, 활동 i 가 선택 가능한 모드의 수
m	활동이 선택 가능한 모드 인덱스, $m = 1, 2, \dots, M_{pqi}$
R	자원 종류의 개수
r	자원 종류의 인덱스, $r = 1, 2, \dots, R$
V_p	p 번째 사업을 수행하는 경우에 예상되는 가치
$Prob_p$	p 번째 제안 프로젝트를 기본적인 품질 수준으로 수행할 때, 사업의 수주 확률
$Prob_{pqim}^+$	p 번째 제안 프로젝트를 q 번째 프로젝트 모드로 수행하고, 제안 프로젝트에서 활동 i 가 m 번째 모드로 수행할 때, 연장되는 단위 기간별 증가하는 수주 확률의 증가분
d_{pi}	p 번째 제안 프로젝트에서 활동 i 의 실행 기간
D_{pqim}	p 번째 제안 프로젝트를 q 번째 프로젝트 모드로 수행하고, 제안 프로젝트에서 활동 i 가 m 번째 모드로 수행할 때, 정상적인 수행 기간
u_{pqim}	p 번째 제안 프로젝트가 q 번째 프로젝트 모드로 수행할 때, 제안 프로젝트에서 활동 i 가 m 번째 활동 모드로 수행할 때, D_{pqim} 을 초과하여 연장된 기간; $u_{pqim} = d_{pi} - D_{pqim}$
x_p	p 번째 제안 프로젝트의 포트폴리오 선정 여부를 나타내는 0-1 변수
y_{pq}	p 번째 제안 프로젝트의 q 번째 프로젝트 모드에 대한 선정 여부를 나타내는 0-1 변수
z_{pqim}	p 번째 제안 프로젝트를 q 번째 프로젝트 모드로 수행할 때, 제안 프로젝트에서 활동 i 에 대한 m 번째 모드의 선정 여부를 나타내는 0-1 변수
Max_Ext_{pqim}	p 번째 제안 프로젝트를 q 번째 프로젝트 모드로 수행하고, 제안 프로젝트 p 의 활동 i 를 m 번째 모드로 배정할 때, 활동 pi 의 연장 기간 최댓값
f_{pi}	제안 프로젝트 p 에서 활동 i 의 종료 시점을 나타내는 변수
H_p	제안 프로젝트 p 에서 각 활동간 선행관계 집합; $(pi, pj) \in H_p$ 인

경우, pi 가 종료되는 시점에 pj 는 시작할 수 있다.

Due_Date_p 제안 프로젝트 p 의 제안 마감일

C_r r 번째 종류의 자원 가용량

PS_t 시점 t 에 진행 중인 제안 프로젝트들의 집합

AS_{pt} 시점 t 에 진행 중인 제안 프로젝트 p 의 활동들의 집합

AR_{pqimr} 제안 프로젝트 p 를 q 번째 프로젝트 모드로 수행하고 제안 프로젝트에서 활동 i 에 배정된 모드가 m 일 때, 활동 i 의 모든 기간에 걸쳐 매 기간별로 필요한 r 번째 자원의 수량

PR_{pqr} 제안 프로젝트 p 를 q 번째 프로젝트 모드로 실행할 때, 제안 프로젝트의 전 기간에 걸쳐 매 기간별로 필요한 r 번째 자원의 수량

식 [1]은 목적함수로써 본 문제가 제안 프로젝트를 선정하고, 선정된 제안 프로젝트의 프로젝트 모드를 결정하며, 선정된 제안 프로젝트의 각 활동별로 활동의 수행 모드, 연장되는 기간과 종료 시점을 결정하는 문제임을 보여주며, 제안 프로젝트 포트폴리오의 기대가치 최대화 문제임을 보여준다. 선정 가능한 P 개의 제안 프로젝트 중 p 번째 프로젝트의 선정 여부를 0-1 변수인 x_p 로 표현하며, p 번째 제안 프로젝트에서 선정 가능한 프로젝트 모드의 수를 Q_p 로 표기한다. 0-1 의사결정변수 y_{pq} 는 p 번째 제안 프로젝트가 q 번째 프로젝트 모드를 선택하면 1, 아니면 0의 값을 갖는다. 식 [2]는 제안 프로젝트별 결정 가능한 프로젝트 모드가 오직 하나임을 나타내며, 선정된 제안 프로젝트에 대해서만 적용된다. 즉, 모든 제안 프로젝트 p 에 대해서 적용되는 것이 아니고 $x_p = 1$ 인 경우에만 적용된다. 식 [3]은 제안 프로젝트 p 가 선정되고 프로젝트 모드 q 가 배정될 경우 제안 프로젝트의 각 활동에 오직 하나의 모드만 배정됨을 나타낸다. 식 [4]는 p 번째 제안 프로젝트에서 활동 i 가 모드 m 으로 배정될 때, 추가 투입을 통해 선택할 수 있는 기간이 해당 모드의 최대 투입 기간을 초과할 수 없음을 나타낸다. 식 [5]는 제안 프로젝트 p 에서 활동 i 의 수행 기간(d_{pi})이 선정된 모드 m 에서의 정상 기간(D_{pqim})에 연장된 기간(u_{pqim})을 더한 값을 나타낸다. 식 [6]은 제안 프로젝트 p 에서 활동 간의 선행관계 집합이 H 이고 $(pi, pj) \in H_p$ 인 경우, 활동 pj 가 시작 시점이 pi 가 종료 시점임을 나타낸다. 따라서 제안 프로젝트 p 의 종료를 의미하는 활동은 $(I_p + 1)$ 이므로, 프로젝트 p 의 종료 시기는 f_{p, I_p+1} 이 된다. 식 [7]은 제안 프로젝트 p 가 자신의 제안 마감일인 Due_Date_p 에 반드시 종료되

어야 함을 나타낸다. 식 [8]은 제안 프로젝트 포트폴리오에서 요구되는 매 기간별, 자원 종류별 자원의 수량이 자원 가용량을 초과할 수 없음을 나타낸다. 식 [9]는 제안 프로젝트의 시작과 종료를 나타내는 가상의 활동들에 대한 수행기간은 0임을 나타낸다. 본 연구에서는 제안 프로젝트 포트폴리오의 분석 시작 시점을 0이라 가정한다. 따라서 식 [10]은 모든 제안 프로젝트의 시작 시기가 포트폴리오의 분석 시작 시점인 0보다 앞설 수 없음을 나타낸다. 식 [11]은 x_p , y_{pq} , z_{pqim} 이 0-1 변수임을 나타내며, 식 [12]는 u_{pqim} 이 0이상의 값을 가짐을 나타낸다.

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 활동의 수행 기간 d_{pi} 와 종료 시점 f_{pi} 가 결정된다. 그러므로 d_{pi} 와 f_{pi} 는 종속변수가 된다. 본 연구에서는 본 모형에 적합한 휴리스틱 기법을 제시하고자 한다.

4.1 해의 영역, 종속변수와 목적함수

본 연구의 해의 영역과 종속변수는 다음과 같다.

4.1.1 해의 영역

해의 영역에서 프로젝트 선정 변수인 x_p 해의 영역은 $x_1, x_2, \dots, x_p, \dots, x_P$ 이다. 프로젝트 모드의 선정을 의미하는 y_{pq} 에 대한 해의 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 에 대한 모드의 선정을 의미한다. p 번째 프로젝트가 선정되고($x_p = 1$) q 번째 프로젝트 모드가 선정되었을 때($y_{pq} = 1$), 활동 i 에 대한 모드의 선정에 한정하여 z_{pqim} 의 부분해를 나열하면 $z_{pq1}, z_{pq2}, \dots, z_{pqim}, \dots, z_{pqM_{pi}}$ 가 된다. 이와 같

은 부분해는 가능한 pqi 의 조합의 수만큼 생성된다.

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

4.1.2 종속변수 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} 의 값 결정

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

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

(2) 선정된 제안 프로젝트의 활동별 종료 시점 f_{pi} 의 값 결정

각 활동 간의 선행관계와 제안 프로젝트의 마감일이 알려져 있고, ($x_p, y_{pq}, z_{pqim}, u_{pqim}$)로부터 각 활동의 수행 기간을 구할 수 있으므로 각 활동의 종료 시점도 결정할 수 있다. 앞에서 기술한 바와 같이 제안 프로젝트 특성상 프로젝트의 시작 시점에 대한 제약이 있는 것이 아니라 프로젝트 종료 시점에 대한 제약이 존재하므로, 프로젝트의 마감일로부터 역순으로 각 활동의 종료 시점을 결정한다. 종료 시점이 역순으로 결정되므로, j 가 활동 i 의 후행 활동이면, 활동 i 의 종료 시점은 후행 활동 j 의 시작 시점(종료 시점에서 수행 기간을 제한 값)이 결정된 이후에나 확정된다. 프로젝트 p 가 선정된 경우, 활동 i 의 종료 시점은 다음과 같고, 프로젝트 종료를 의미하는 가상활동 I_{p+1} 의 경우, 종료 시점은 다음과 같다.

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

가상의 활동(dummy node를 의미)이 아닌 활동 i 의 종료 시점은 i 의 모든 후행 활동들의 시작 시점 중 가장 빠른 값과 같다. 이를 수식으로 표현하면 다음과 같다.

해당 프로젝트가 배정할 때 자원 제약 조건이 위배되면, 해당 프로젝트 배정을 취소한다. 지금까지 배정된 프로젝트들로 최초의 가능해를 구성한다. 종료한다.

Step 3에서 x 가 결정된다. 자원제약 위배 없이 배정된 프로젝트들의 x 는 1의 값을 갖고 미배정된 프로젝트들의 x 는 0의 값을 갖는다. Step 3에서 결정된 x 와 Step 1에서 결정된 y, z 와 u 를 결합하여 최초의 가능해를 구성한다. 만약 모든 프로젝트가 선정되었다면, 여기서 구한 가능해는 문제의 최적해가 된다. 그렇지 않다면, 다음의 개선 규칙들을 적용하여 해의 개선을 시도한다.

4.2.2 개선규칙 1 - 미배정 프로젝트의 배정 시도

개선규칙 1은 현재의 가능해에서 미배정된 프로젝트들을 대상으로 한다. 현 가능해에서 프로젝트 p 가 미배정 상태($x_p = 0$)라 하자. 현 가능해에서 선정된 프로젝트들의 배정은 고정한 채 프로젝트 p 를 추가로 배정될 수 있다면, 해의 값이 당연히 향상된다. 그러므로 x_p 를 0에서 1로 변경할 때, 모든 자원 제약 조건을 충족시키는 y, z 와 u 가 존재하는지, 존재한다면 해값을 최대로 만드는 y, z 와 u 의 값은 무엇인지를 조사할 필요가 있다. 프로젝트 모드의 수 $Q_p = 2$, 활동의 수 $I_p = 4$, 활동 모드의 수 $M_{pqi} = 2$, 초과 연장 기간 $u_{pqim} = 2$ 인 경우, 총 가능한 yzu 조합의 수는 $Q_p \times [M_{pqi} \times (u_{pqim} + 1)]_p^I = 2 \times [2 \times (2 + 1)]^4 = 2 \times 6^4 = 2,592$ 가 된다. 본 연구에서는 본 저자의 박사학위 논문 “자원 제약을 고려한 제안 프로젝트 포트폴리오 최적화 문제”(조윤재, 2020)에서 소개된 최적해법을 사용하여 가능한 모든 yzu 조합을 검토한다.

Step 1. 현 가능해에서 미배정 상태인 프로젝트들을 하나씩 택한다. 선택한 프로젝트를 프로젝트 p 라 하자.

Step 2. 프로젝트 p 의 가능한 모든 yzu 조합을 검토한다. 해당 yzu 조합으로 배정할 때 자원제약 조건이 모두 충족되면, 이때의 조합을 배정 가능한 yzu 조합이라 하자.

배정 가능한 조합이 존재하면, 목적함수 값을 최대로 하는 배정 가능한 조합을 선택하고 현재의 가능해를 다음과 같이 수정한다. $x_p = 1$, 프로젝트 p 의 yzu 는 해당 yzu 로 한다.

4.2.3 개선규칙 2 - 배정 프로젝트 제거 후 미배정 프로젝트의 배정 시도

개선규칙 2는 이미 배정된 프로젝트를 하나 제거한 후 미배정된 프로젝트 하나를 추가해 본다. 현 가능해에서 프로젝트 p 는 선정($x_p = 1$)이고, q 는 미선정($x_q = 0$)이라 하자. 현 가능해에서 프로젝트 p 의 기대가치를 $Value(p)$ 라 하자. $Value(p)$ 는 현 가능해에서 p 의 yzu 를 $V_p \times \left[Prob_p + \sum_{q=1}^{Q_p} \left(\sum_{i=1}^{I_p} \sum_{m=1}^{M_{pqi}} Prob_{pqim}^+ \times z_{pqim} \times u_{pqim} \right) \times y_{pq} \right]$ 에 대입한 값이다. 현 가능해에서 p 를 제거하고 q 를 추가 선정한다고 하자. q 의 배정 가능한 yzu 조합이 만드는 q 의 기대가치 최대값을 $Value(q)$ 라 하자. $Value(q)$ 가 $Value(p)$ 보다 크다면, p 를 제거하고 q 를 배정하여 가능해를 개선한다.

Step 1. 모든 가능한 (배정된 프로젝트, 미배정 프로젝트) 조합 별로 Step 2를 검토한다.

Step 2. 현 가능해에서 배정된 프로젝트를 p , 미배정된 프로젝트를 q 라 하자. p 를 제거하고 q 를 배정할 때, $Value(q)$ 를 최대화시키는 q 의 배정 가능한 yzu 조합을 찾는다. q 의 배정 가능한 yzu 조합이 존재하고, $Value(q) > Value(p)$ 이면, 다음과 같이 해를 개선한다. $x_p = 0$, $x_q = 1$, q 의 yzu 는 $Value(q)$ 를 만드는 배정 가능한 yzu 로 한다.

4.2.4 개선규칙 3 - 배정 프로젝트의 yzu 조합 변경

개선규칙 3은 이미 배정된 프로젝트의 yzu 조합을 변경해 본다. 현 가능해에서 프로젝트 p 는 배정된 상태($x_p = 1$)이고, 현 가능해에서 p 의 기대가치는 $Value(p)^{CURRENT}$ 라 하자. p 의 배정 가능한 yzu 조합이 만드는 p 의 기대가치 중 최대값을 $Value(p)^{OPTIMUM}$ 라 하자. $Value(p)^{OPTIMUM} > Value(p)^{CURRENT}$ 이면, p 의 yzu 배정은 $Value(p)^{OPTIMUM}$ 을 만드는 배정 가능한 yzu 조합으로 변경한다.

4.2.5 새로운 초기해 생성

개선규칙 1, 2와 3을 적용해도 현재의 가능해가 개선되지 않을 때, 배정된 프로젝트 중 하나를 임의로 선정하여 제거하여 새로운 가능해를 생성한 후 개선규칙 1, 2와 3을 순차적으로 적용한다. 이때 제거된 프로젝트는 Tabu_Project로 명명한다. Tabu_Project는 명명된 이후 개선규칙 1이 최초로 적용될 때는 선택하지 않는다. 명명 이후 최초로 적용되는 개선규칙 1에서 Tabu_Project를 선정된다면, Tabu_Project를 제거하기 이전의 가능해와 동일한 해가 되기 때문이다.

4.2.6 휴리스틱 알고리즘 약술

- Step 1. (최초의 가능해)
 최대 가치 순으로 프로젝트를 배정한다. 현재의 가능해와 해값을 Best_Sol과 Best_Sol_Value로 한다. (4.2.1 참조)
 If 모든 프로젝트가 배정되었으면,
 현재의 Best_Sol_Value가 최적해값이다. 종료한다.
 Else Tabu_Project = -1, Goto Step 2.
- Step 2. (개선 모듈)
 현 가능해값을 Loop_Ini_Value로 한다.
 개선규칙 1(미배정 프로젝트의 배정 시도)을 적용한다.
 단, Tabu_Project는 선정 대상에서 제외한다.
 개선규칙 2(배정 프로젝트 제거 후 미배정 프로젝트의 배정 시도)를 적용한다.
 개선규칙 3(배정 프로젝트의 *yzu* 조합 변경)을 적용한다.
 Tabu_Project = -1
 If 현 가능해값 > Loop_Ini_Value (개선규칙을 통해 개선되었으면)
 Best_Sol, Best_Sol_Value를 현 가능해와 해값으로 한다.
 Goto Step 2. (Step 2를 반복한다.)
 Else Goto Step 3. (개선규칙을 통한 개선이 없었으면, 새로운 해를 생성한다.)
- Step 3. (새로운 초기해 생성)
 If 미리 설정한 연산시간이 경과되지 않았으면,
 현 가능해에서 배정된 프로젝트를 임의로 제거하여 새로운 해를 생성한다.
 제거된 프로젝트를 Tabu_Project로 한다. Goto Step 2.
 Else (미리 설정한 연산시간이 경과되었으면)
 휴리스틱의 해와 해값으로 Best_Sol과 Best_Sol_Value를 제시한다. 종료한다.

Tabu_Project = -1은 Tabu_Project가 설정되지 않았음을 의미한다. Tabu_Project는 Step 3에서 Step 2로 진행될 때만 의미가 있으므로, Step 1에서 Step 2로 진행되거나 Step 2가 반복될 때는 -1로 설정한다.

V. 모의실험 결과

5.1 모의실험 조건 설정

5.1.1 데이터 생성 기본구성

본 연구의 모의실험에 사용한 데이터는 다음과 같다.

<표 2> 데이터셋의 기본구성

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

5.1.2 난수값에 의한 데이터 생성 규칙

(1) 본 사업의 기대가치, 프로젝트 수주확률, 제안 마감일과 자원의 가용량 생성

본 사업의 기대가치, 프로젝트 수주확률과 제안 마감일은 일양 분포를 사용하여 구하였다. 사업의 기대가치 V 는 최솟값이 100, 최댓값이 500으로, 프로젝트 수주확률은 최솟값이 0.30, 최댓값이 0.70으로, 제안 마감일의 최솟값은 60, 최댓값이 100인 일양 분포로부터 각각 구하였다. 포트폴리오의 기간 중 가용자원의 양은 일양 분포를 사용하여 구하였다. 자원 종류의 수는 6이며, 자원 종류별 값의 하한과 상한은 아래와 같다.

자원별 하한값 = (8, 5, 8, 5, 5, 5), 자원별 상한값 = (11, 7, 11, 7, 7, 7)

(2) 자원별 필요량의 생성

자원별 필요량은 일양 분포를 사용하여 생성하였으며, 필요자원의 수는 정수가 되도록 하였다. 추가되는 자원 필요량도 일양 분포를 사용하여 구하였다. 활동의 모드별 자원 필요량도 일양 분포를 사용하여 구하였다.

(3) 활동 기간의 생성

편의상 $D(p_mode, i, a_mode)$ 는 프로젝트 모드가 p_mode 이고, 활동 i 가 모드 a_mode 로 실행될 때의 기간이라 하자. 활동 1부터 3까지의 기간은 다음과 같은 방법으로 구하였다.

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

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

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

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

활동 4의 기간은 아래와 같은 방법으로 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) 각 활동의 모드별 최대 연장 기간

각 활동의 모드별 수행 기간이 존재한다. 이 기간은 연장될 수도 있으며, 기간이 연장되는 만큼 프로젝트의 품질도 향상되고, 결과적으로 프로젝트의 수주확률도 그만큼 증가한다. 활동 1, 2와 3의 경우 1기간 연장이, 활동 4의 경우 2기간 연장이 가능하다. 1기간 연장에 따른 수주확률의 증가분은 아래와 같이 일양 분포를 사용하여 생성하였다.

활동 1~3의 경우, $U(0.02, 0.04)$, 활동 4의 경우, $U(0.025, 0.075)$

5.2 모의실험 결과

각각의 데이터 셋은 20개의 문제로 구성되어 있다. 제안한 휴리스틱 기법의 효율성을 측정하기 위해 휴리스틱 기법으로 구한 값과 최적해값을 비교하였다. 최적해는 본 저자의 선행 연구인 “자원제약을 고려한 제안 프로젝트 포트폴리오 최적화 문제”(조윤재, 2020)에서 제안한 기법을 사용했다. 연산 시간은 휴리스틱의 경우 10초이며, 최적해법은 최대 3,600초까지 허용하였다. <표 3>와 같이 기준이 되는 데이터셋을 Basic 데이터셋이라 하자. 이때, Basic 데이터셋 문제들을 휴리스틱으로 구한 해와 최적해값은 아래 표 <표 3>와 같다.

<표 3> Basic 데이터셋 모의실험 결과

문제 번호	휴리스틱 해값(a)	최적해값 (b)	효율(a/b)	최적해 연산시간
1	956	956	100%	594
2	829	829	100%	680
3	992	1,030	96%	364
4	816	851	96%	547
5	1,062	1,062	100%	1,060
6	647	647	100%	1,065
7	985	985	100%	164

8	1,075	1,075	100%	98
9	883	883	100%	114
10	692	690	100%	106
11	1,141	1,141	100%	77
12	775	775	100%	86
13	1,047	1,067	98%	555
14	830	830	100%	29
15	952	952	100%	89
16	666	671	99%	2,034
17	852	852	100%	347
18	882	882	100%	16
19	732	732	100%	9
20	1,209	1,209	100%	800
평균			99%	442
최대			100%	2,034
최소			96%	9

Basic 데이터셋의 경우, 20문제 전부 최적해 값을 제한시간 3,600초 이내에 찾았으며, 평균 연산 시간은 442초였다. 휴리스틱 기법의 평균 효율은 99%에 달했으며, 20문제 중 16 문제의 경우, 최적해 값과 동일한 해의 값을 찾았다.

본 모의실험에서는 제안 프로젝트의 수, 프로젝트 모드의 수, 활동별 모드의 수 및 자원 가용량 크기의 4가지 계수 값의 변화가 휴리스틱의 효율에 어떤 영향을 주는지 파악하기 위해 각 계수들의 값을 가감 및 조정하여 Basic 데이터셋 연산 시간의 결과와 비교하였다. 4개의 모의실험에 대한 결과는 순서대로 <표 4>, <표 5>, <표 6>, <표 7>과 같다.

<표 4> 프로젝트(P) 갯수가 12, 14, 16인 데이터셋 모의실험 결과

문제 번호	P=12				Basic 데이터셋(P=14)				P=16			
	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간
1	895	940	95%	280	956	956	100%	594	848	848	100%	554
2	705	705	100%	16	829	829	100%	680	847	N/A	N/A	3,600+*)
3	734	734	100%	28	992	1,030	96%	364	741	741	100%	303
4	941	941	100%	141	816	851	96%	547	725	725	100%	124
5	580	580	100%	4	1,062	1,062	100%	1,060	691	691	100%	119
6	674	689	98%	735	647	647	100%	1,065	728	728	100%	44
7	757	815	93%	339	985	985	100%	164	964	964	100%	2,074
8	929	929	100%	86	1,075	1,075	100%	98	760	760	100%	552
9	1,102	1,102	100%	71	883	883	100%	114	981	981	100%	278
10	554	554	100%	21	692	690	100%	106	1,237	1,237	100%	2,689
11	1,029	N/A	N/A	3,600+*)	1,141	1,141	100%	77	1,095	1,095	100%	36

12	594	594	100%	45	775	775	100%	86	1,004	1,004	100%	1,242
13	988	988	100%	5	1,047	1,067	98%	555	765	947	81%	8
14	889	940	94%	34	830	830	100%	29	1,125	1,125	100%	1,544
15	849	849	100%	930	952	952	100%	89	842	842	100%	569
16	736	736	100%	112	666	671	99%	2,034	920	920	100%	902
17	552	552	100%	37	852	852	100%	347	837	837	100%	541
18	606	606	100%	56	882	882	100%	16	952	966	99%	157
19	815	815	100%	298	732	732	100%	9	866	866	100%	40
20	823	823	100%	5	1,209	1,209	100%	800	955	955	100%	119
평균			N/A	N/A			99%	442			N/A	N/A
최대			100%	3,600+			100%	2,034			100%	3,600+
최소			93%	4			96%	9			81%	8

*) 연산의 제한시간 3,600초를 초과함을 의미한다.

<표 5> 프로젝트 모드(PM)의 수가 1, 2, 3인 데이터셋의 모의실험 결과

문제 번호	PM=1				Basic 데이터셋(PM=2)				PM=3			
	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간
1	651	651	100%	44	956	956	100%	594	862	862	100%	257
2	849	849	100%	46	829	829	100%	680	707	N/A	N/A	3,600+*)
3	591	591	100%	12	992	1,030	96%	364	1,006	1,006	100%	268
4	992	992	100%	896	816	851	96%	547	709	709	100%	1,022
5	676	676	100%	12	1,062	1,062	100%	1,060	898	898	100%	220
6	895	895	100%	100	647	647	100%	1,065	1,106	1,106	100%	971
7	700	700	100%	10	985	985	100%	164	811	811	100%	102
8	804	804	100%	14	1,075	1,075	100%	98	918	918	100%	16
9	779	787	99%	38	883	883	100%	114	879	879	100%	947
10	644	644	100%	7	692	690	100%	106	819	819	100%	297
11	807	807	100%	21	1,141	1,141	100%	77	773	773	100%	111
12	764	764	100%	70	775	775	100%	86	762	762	100%	1,139
13	752	752	100%	110	1,047	1,067	98%	555	1,269	1,269	100%	532
14	698	698	100%	28	830	830	100%	29	930	930	100%	2,011
15	741	741	100%	115	952	952	100%	89	829	829	100%	311
16	486	486	100%	6	666	671	99%	2,034	885	885	100%	1,919
17	967	967	100%	72	852	852	100%	347	729	729	100%	3,078
18	601	601	100%	2	882	882	100%	16	950	N/A	N/A	3,600+*)
19	893	893	100%	6	732	732	100%	9	740	740	100%	185
20	739	739	100%	—	1,209	1,209	100%	800	830	830	100%	578
평균			99.9%	80			99%	442			N/A	N/A
최대			100%	896			100%	2,034			100%	3,600+
최소			99%	0			96%	9			96%	16

*) 연산의 제한 시간 3,600초를 초과함을 의미한다.

<표 6> 활동모드(AM)의 수가 1, 2, 3인 데이터셋의 모의실험 결과

문제 번호	AM=1				Basic 데이터셋(AM=2)				AM=3			
	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간
1	706	706	100%	3	956	956	100%	594	730	730	100%	129
2	698	698	100%	1	829	829	100%	680	1,077	1,077	100%	909
3	792	792	100%	5	992	1,030	96%	364	832	832	100%	308
4	704	704	100%	3	816	851	96%	547	780	854	91%	1,146
5	971	1,004	97%	6	1,062	1,062	100%	1,060	1,002	1,002	100%	188
6	484	484	100%	3	647	647	100%	1,065	690	N/A	N/A	3,600+*)
7	810	810	100%	2	985	985	100%	164	750	750	100%	565
8	686	686	100%	1	1,075	1,075	100%	98	988	N/A	N/A	3,600+*)
9	550	550	100%	4	883	883	100%	114	1,154	N/A	N/A	3,600+*)
10	844	844	100%	6	692	690	100%	106	769	N/A	N/A	3,600+*)
11	662	662	100%	2	1,141	1,141	100%	77	484	656	74%	24
12	1,008	1,008	100%	5	775	775	100%	86	941	941	100%	1,714
13	751	751	100%	4	1,047	1,067	98%	555	618	618	100%	2,550
14	750	750	100%	1	830	830	100%	29	928	928	100%	2,097
15	967	967	100%	26	952	952	100%	89	590	689	86%	577
16	918	918	100%	6	666	671	99%	2,034	981	N/A	N/A	3,600+*)
17	583	583	100%	2	852	852	100%	347	1,039	N/A	N/A	3,600+*)
18	835	835	100%	10	882	882	100%	16	711	711	100%	646
19	587	587	100%	1	732	732	100%	9	957	957	100%	995
20	1,243	1,243	100%	208	1,209	1,209	100%	800	680	680	100%	448
평균			99.8%	15			99%	442			N/A	N/A
최대			100%	208			100%	2,034			100%	3,600+
최소			97%	1			96%	9			74%	24

*) 연산의 제한 시간 3,600초를 초과함을 의미한다.

<표 7> 가용자원(CAP)의 크기를 -1 한 경우와 +1한 경우의 모의실험 결과

문제 번호	CAP -1				Basic 데이터셋				CAP +1			
	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간	휴리스 틱해값	최적 해값	효율	연산 시간
1	842	842	100%	33	956	956	100%	594	1,178	1,178	100%	3,377
2	703	780	90%	14	829	829	100%	680	891	N/A	N/A	3,600+*)
3	802	802	100%	17	992	1,030	96%	364	1,173	N/A	N/A	3,600+*)
4	601	676	89%	47	816	851	96%	547	945	N/A	N/A	3,600+*)
5	857	857	100%	233	1,062	1,062	100%	1,060	1,343	N/A	N/A	3,600+*)
6	535	535	100%	32	647	647	100%	1,065	896	896	100%	1,693
7	705	705	100%	11	985	985	100%	164	1,179	1,179	100%	2,173
8	896	896	100%	59	1,075	1,075	100%	98	1,087	N/A	N/A	3,600+*)
9	798	798	100%	15	883	883	100%	114	1,081	1,081	100%	2,069

10	453	453	100%	3	692	690	100%	106	787	787	100%	2,236
11	853	853	100%	33	1,141	1,141	100%	77	1,141	1,141	100%	2,897
12	602	602	100%	11	775	775	100%	86	882	882	100%	1,288
13	935	935	100%	46	1,047	1,067	98%	555	1,282	1,282	100%	571
14	561	561	100%	9	830	830	100%	29	949	949	100%	750
15	836	836	100%	17	952	952	100%	89	1,215	1,215	100%	1,665
16	559	559	100%	158	666	671	99%	2,034	726	N/A	N/A	3,600+*)
17	683	683	100%	33	852	852	100%	347	908	N/A	N/A	3,600+*)
18	613	613	100%	10	882	882	100%	16	920	920	100%	163
19	698	698	100%	1	732	732	100%	9	864	864	100%	59
20	1,081	1,081	100%	80	1,209	1,209	100%	800	1,409	N/A	N/A	3,600+*)
평균			99.0%	43			99%	442			N/A	N/A
최대			100%	233			100%	2,034			100%	3,600+
최소			89%	1			96%	9			100%	59

*) 연산의 제한 시간 3,600초를 초과함을 의미한다.

위의 조건들을 사용하여 총 180문제를 최적해법과 휴리스틱 기법으로 해를 구해 보았다. 180문제 중 18개의 문제의 경우 3,600초 이내에 최적해를 구하는 데 실패하였다. 그러므로 최적해 값이 알려진 문제의 수는 162개이다. 최적해 값이 알려진 162 문제의 경우, 휴리스틱이 최적해 값과 동일한 해의 값을 찾은 경우(효율 100%인 경우)는 126문제에 이르며, 162문제의 평균 효율은 99.2%이며 대부분 90% 대의 효율을 보였다. 90% 미만의 효율은 총 3번 관찰되었으며, 그 값은 85.5%, 80.8%와 73.8%였다. 따라서 본 모의실험을 통해 제안한 휴리스틱 기법의 우수성을 확인할 수 있었다.

IV. 결 론

본 연구는 공공사업들에 입찰하려는 기업이 한정된 자원 하에서 제안 프로젝트 포트폴리오를 어떻게 구성하는 것이 기대이익을 최대화하는가에 대한 문제의식에서 출발하였다. 이에 포트폴리오 이론, 프로젝트 선택(Project Selection), RCPS, 입찰에 관한 실증연구 등을 선행연구로 진행하였으며, 제안 프로젝트별 자원투입량의 결정에 프로젝트 모드라는 변수를 도입하여 수학적 모형을 수립하였다. 그리고 휴리스틱 기법을 적용하여 알고리즘을 제시하였으며, 모의실험을 통해 제시한 알고리즘이 효율적이라는 것을 입증하였다.

본 연구의 문제는 해 영역의 크기가 커지면 탐색에 소요시간이 급증하는 전형적인 NP-Hard에 속한다. 일반적으로 NP-Hard에 속하는 문제는 현실적인 시간 내에 최적

해를 구하는 것이 불가능한 경우가 많다. 따라서 현실적인 시간 내에 우수한 해를 도출하는 휴리스틱 기법의 개발이 필요하다. 그러므로 본 연구에서는 휴리스틱 기법을 적용한 결과와 조윤재(2020)의 선행연구인 최적해법 실험결과와 비교하였다. 그 결과, 휴리스틱 기법을 적용한 경우 10초라는 극히 짧은 시간 내에 최적해법의 결과와 비교하여 평균 99.2%의 최선해를 도출하였다. 이는 휴리스틱 기법이 시간과 해의 정확성에 대한 효율 면에서 우수하다는 점을 시사한다. 본 휴리스틱 기법의 알고리즘을 바탕으로 적절한 사용자 Interface를 가진 의사결정지원시스템(DSS, Decision Support System)기반의 제안 프로젝트 포트폴리오 프로그램이 개발된다면 실용적으로 제안참여 의사결정 및 이해관계자 의사소통에 많은 도움이 될 것이다.

마지막으로 본 연구는 모의실험에 가상의 데이터를 사용함으로써 현실과 차이가 있을 수 있다는 한계를 가지고 있으며, 제안 프로젝트가 가지는 현실적인 불확실성(제안 마감일의 변경, 외부 환경변화에 따른 수주확률의 변화 등)이 적용되어 있지 않다. 따라서, 향후 본 연구에서 다루지 못한 부분에 관한 추가적인 연구가 필요하다.

참고문헌

- 김동욱, 이원영 (2018), 유전자 알고리즘을 이용한 프로젝트 포트폴리오 투입인력 최적화 모델에 관한 연구, *한국IT서비스학회지*, 17(4), 101-117.
- 남재덕 (2008), 프로젝트 포트폴리오 평가 문제: 불확실성 하의 최초 구성 및 주기적 개장을 중심으로, *승실대학교 대학원 박사학위논문*.
- 민택기 (2005), 프로젝트 포트폴리오 평가문제: 시뮬레이션을 이용한 다속성 평가를 중심으로, *승실대학교 대학원 박사학위논문*.
- 박상우 (2013), *수주의 기술*, 비즈니스아츠, 95.
- 백인섭 (2019), 자원 제약을 고려한 프로젝트 일정문제 : 재작업 가능한 활동이 있는 경우, *승실대학교 대학원 박사학위논문*.
- 안태호 (2007), R&D 프로젝트 포트폴리오 구성 휴리스틱 기법, *승실대학교 사회과학 연구원 사회과학논총*, 9, 183-195.
- 조윤재 (2020), 자원 제약을 고려한 제안 프로젝트 포트폴리오 최적화 문제, *승실대학교 대학원 박사학위논문*.
- Archer, N. P. and Ghasemzadeh, F. (1999), An integrated framework for project portfolio selection, *International Journal of Project Management*, 17(4), 207-216.
- Astana, I., Rusdi, H. A. and Wibowo, M. A. (2015), Conceptual Framework Of Bidding Strategy In Order To Improve Construction Project Performance. In *Applied Mechanics and Materials (776, 108-113)*, Trans Tech Publications Ltd.
- Blichfeldt, B. S. and 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. and 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. and 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. and Suzuki, A. (2010), A Solution Method for Manpower Scheduling Problems by RCPSP/τ.
- Huang, X. and Zhao, T. (2014), *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. and 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. and Khumawala, B. M. (1978), Multiple resource-constrained scheduling using branch and bound, *AIIE Transactions*, 10(3), 252–259.
- Takano, Y., Ishii, N. and Muraki, M. (2017), Multi-period resource allocation for estimating project costs in competitive bidding, *Central European Journal of Operations Research*, 25(2), 303–323.

*** 저자소개 ***

· **조 윤 재(dawnvits@naver.com)**

숭실대학교 대학원에서 프로젝트경영학 전공으로 경영학 박사학위를 취득하였다. 주요관심 분야는 OM이다.

· **백 인 섭(insoub@gmail.com)**

숭실대학교 대학원에서 경영학 전공으로 경영학 박사학위를 취득하였다. 주요관심 분야는 OM이다.

· **김 종 출(airncountry@gmail.com)**

숭실대학교 대학원 경영학과에서 박사과정을 수료하였다. 주요 관심분야는 OM이다.

· **안 태 호(ahnt@ssu.ac.kr)**

University of Florida(Gainesville)에서 경영학 박사학위를 취득하였다. 현재 숭실대학교 경영학부 교수로 재직 중이며 주요 강의 및 연구 분야는 OM, 의사결정론 등이다.