

# 최단시간 포병진지 구축계획 수립을 위한 연구

안문일, 최인찬\*

고려대학교 산업경영공학과

## Study of shortest time artillery position construction plan

Moon-Il Ahn, In-Chan Choi\*

Department of Industrial Management Engineering, Korea University

**요약** 본 논문은 포병진지 구축계획 문제의 효과적인 해결을 위해 해당 문제를 최적화 모형으로 형성하고, 현실규모 문제 해결을 위한 휴리스틱 알고리즘을 제시한다. 포병진지 구축계획은 공병부대의 지원 포병부대 결정과 공병 지원팀 편성, 편성된 공병 지원팀의 포병부대 지역 내 포병진지 구축일정을 수립하는 의사결정을 포함한다. 군에서는 포병진지 구축계획을 담당자의 경험적 지식을 바탕으로 수립한다. 본 논문에서는 공병부대의 포병진지 구축 일정계획 문제를 효과적으로 해결하기 위한 혼합정수계획 모형을 제안한다. 해당 모형의 목적은 공병 지원팀으로 구축이 요구되는 모든 포병진지를 구축 완료하는 데 걸리는 시간을 최소화하는 것이다. 또한, 현실규모 문제 해결을 위해 포병진지 구축계획 문제를 분할하여 해결하는 휴리스틱 알고리즘을 제안한다. 제한한 휴리스틱 알고리즘의 성능을 확인하기 위해 두 가지 실험을 실시한다. 제한한 휴리스틱 알고리즘은 소규모 문제를 이용한 최적해 근접성 실험에서 최적해 대비 평균 6.44% 오차범위의 해를 도출했으며, 포병진지 200개 이상인 문제를 이용한 현실규모 문제 해결 가능성 실험에서 문제를 해결하는 데 평균 79.8초가 소요되었다.

**Abstract** This paper addresses the problem of the construction planning of artillery positions, for which we present an optimization model and propose a heuristic algorithm to solve problems of practical size. The artillery position construction plan includes the assignment of engineers to support the artillery and the schedule of the support team construction sequence. Currently, in the army, managers construct the plan based on their experience. We formulate the problem as a mixed integer program and present a heuristic that utilizes the decomposition of the mixed integer model. We tested the efficacy of the proposed algorithm by conducting computational experiments on both small-size test problems and large-size practical problems. The average optimality gap in the small-size test problem was 6.44% in our experiments. Also, the average computation time to solve the large-size practical problems consisting of more than 200 artillery positions was 79.8 seconds on a personal computer. The result of our computational experiments shows that the proposed approach is a viable option to consider for practical use.

**Keywords** : Artillery position, Construction plan, Scheduling, Optimization, Heuristic algorithm

### 1. 서론

포병진지 구축계획 문제는 다수의 포병진지를 공병 지원팀으로 최단시간 내에 구축하는 일정을 수립하는 문제이다. 포병진지란 대포의 직접피격 확률 감소와 파편에 의한 2차 피해를 막는 토건 보호벽이다. 공병 지원팀은 공병부대가 보유한 장비로 편성되는 포병부대 지원을

위한 포병진지 구축팀이다.

포병진지는 공병부대의 작업 효율성과 포병부대의 생존성을 향상시키기 위해 최단시간 내에 구축되어야 한다. 포병부대는 전쟁 발발 시 신속한 화력대응을 위해 작전지역으로 이동하여 임무를 수행해야 한다. 공병부대는 포병부대가 작전지역으로 이동하여 포병진지를 사용하기 이전에 포병진지 구축을 완료해야 한다. 공병부대는

\*Corresponding Author : In-Chan Choi(Korea Univ.)

Tel: +82-2-3290-3773 email: ichoi@korea.ac.kr

Received March 24, 2016

Revised (1st April 28, 2016, 2nd May 17, 2016)

Accepted June 2, 2016

Published June 30, 2016

포병진지 구축 임무 외 다수의 임무를 수행하므로 공병부대 임무 수행 효율성 증대를 위해 포병진지를 최단 시간 내에 구축해야 한다. 공병 지원팀을 형성하는 장비자원은 구축이 요구되는 포병진지의 수에 비해 한정되어 있다. 군은 공병 지원팀의 효율적인 포병진지 구축을 위해 포병진지 구축계획을 수립한다.

군에서는 담당자의 경험적 지식을 바탕으로 포병진지 구축계획을 다음과 같이 수립한다. 먼저, 상급 공병부대는 포병부대들이 구축을 요구하는 포병진지의 정보를 수집한다. 이때, 포병진지 정보는 포병진지를 사용하는 포병부대, 포병진지의 위치, 포병진지의 작업량 등을 포함한다. 다음으로 상급 공병부대는 수집된 포병진지 정보와 예하 공병부대의 장비 현황을 바탕으로 해당 공병부대에 포병진지를 할당한다. 이때, 포병진지는 포병부대 단위로 묶어 공병부대에 할당한다. 마지막으로 포병진지를 할당받은 공병부대는 보유 장비로 지원팀을 편성하고, 할당받은 포병진지를 대상으로 지원팀의 구축계획을 수립한다.

포병진지 구축계획에 관련하여 할당된 포병진지 구축을 위한 공병 지원팀의 이동경로계획 연구가 수행되었다. 문정현(2011)은 지원팀의 이동경로를 최소화하기 위해 해당 문제를 차량경로문제(Vehicle routing problem)로 보고 이를 해결하기 위한 혼합정수계획 모형과 PSO(Particle swarm optimization) 방법론을 활용한 휴리스틱 알고리즘을 제안하였다[1].

본 논문에서는 공병부대의 포병진지 구축계획문제를 일정계획문제로 보고 이를 해결하기 위한 혼합정수계획 모형을 제안한다. 또한, 혼합정수계획 모형의 현실규모 문제 해결을 위해 휴리스틱 알고리즘을 제안하고, 실험을 통해 현실규모 문제 해결 가능성을 검증했다.

논문은 1장 서론 이후 5개의 장으로 이루어진다. 2장에서 문제를 정의하고, 3장에서 수리모형을 제안한다. 4장에서 실험결과를 바탕으로 휴리스틱 알고리즘을 제안하고, 5장에서 실험을 통해 휴리스틱 알고리즘의 성능을 확인한다. 마지막 6장에서 결론을 도출한다.

## 2. 문제정의

본 논문이 제안하는 혼합정수계획 모형은 부여된 포병진지의 구축을 최단시간 내에 완료하기 위해 포병진지

를 공병부대에 할당하고, 공병부대의 보유 장비로 지원팀을 편성하며, 할당된 포병진지에 대해 지원팀이 언제 어떤 포병진지를 구축할지를 결정한다.

일정계획모형은 부여된 작업을 최단시간 내에 완료하기 위해 가용 기계에 작업을 할당하는 작업 할당계획 수립 의사결정을 포함한다. 이러한 일정계획모형은 작업, 기계, 최적의 조건 등 3가지 특성에 따라 분류한다[2-4]. Job-shop 일정계획모형, Parallel machine 일정계획모형, Resource constraint project 일정계획모형 등을 예로 들 수 있다[5~10].

문제 구성요소의 유사성을 통해 포병진지 구축계획문제를 일정계획문제의 일종으로 보고 모형을 구성하기 위해 다음과 같이 정의한다.

포병진지 구축문제의 구성요소 중 포병부대는 일정계획문제의 작업에 대응된다. 또한, 포병부대 지역 내 포병진지는 일정계획문제에서 작업을 완료하기 위해 수행되어야 하는 세부작업과 대응된다. 포병부대와 포병진지가 가지는 특성은 다음과 같다.

첫 번째, 포병부대는 단일 공병부대의 지원을 받을 수 있으나 공병부대는 다수의 포병부대를 지원 가능하다.

두 번째, 포병진지를 구축하는 데 걸리는 시간은 기계의 작업 성능에 따라 결정된다. 즉, 각 포병진지 구축에 필요한 흙의 양은 동일하다고 가정한다.

세 번째, 구축 작업과 수행 기계는 일대일로 대응되며 작업의 중단은 허용하지 않는다.

네 번째, 포병진지 구축 작업은 우선순위에 따라 수행한다. 구축 우선순위는 3순위까지 있으며, 작업 시작 시 높은 순위의 포병진지 구축을 우선하여 작업한다.

다섯 번째, 포병진지의 작업시간은 구축시간과 이동시간을 더하여 계산한다. 포병진지 구축을 위해서는 해당 포병진지로의 이동이 선행되어야 한다. 지원팀은 공병부대에서 포병진지로, 포병진지에서 포병진지로 이동하며 작업을 수행한다.

공병부대가 보유하는 지원팀은 일정계획 문제의 기계에 해당하는 요소이다. 지원팀의 특성은 다음과 같다.

첫 번째, 지원팀은 공병부대가 할당받은 포병부대 지역 내의 포병진지만 구축 가능하다.

두 번째, 포병진지를 구축하는 데 걸리는 시간은 지원팀의 구축능력에 의해 결정된다. 지원팀의 구축능력은 보유한 장비 구축능력의 합으로 계산하며, 장비의 구축능력은 단위 시간당 포병진지 구축량으로 나타낸다.(m/h)

세 번째, 지원팀은 부대별 동원완료 시각에 따라 서로 다른 포병진지 구축 시작시각을 가진다. 군은 동원을 통해 일정 시간 이후 민수용 장비를 추가로 사용할 수 있으며, 해당 장비로 지원팀의 편성이 가능하다.

### 3. 수리모형

#### ○ Notation

- $f$  : 포병부대 인덱스.  $f \in F = \{0\} \cup \{1, 2, \dots, |F|\}$
- $c$  : 공병부대 인덱스.  $c \in C = \{1, 2, \dots, |C|\}$
- $i$  : 포병진지 인덱스.  
 $i \in J = \{1, 2, \dots, |J|\} \cup \{|J|+1\}$
- $J_f$  : 포병부대  $f$  지역 내 포병진지 집합.  $J_f \subset J$
- $(i, j)$  : 구축 선행관계 쌍.  $(i, j) \in R, i \in J, j \in J$
- $h$  : 동원완료 여부 인덱스.  
-  $h = 0$  : 동원완료 전  
-  $h = 1$  : 동원완료 후
- $k$  : 지원팀 인덱스.  $k \in K = \{1, 2, \dots, |K|\}$
- $K_{c,h}$  : 공병부대  $c$ 와 동원완료 여부 인덱스  $h$ 에 따른 지원팀 집합.  $K_{c,h} \subset K$
- $e$  : 보유 장비 인덱스.  $e \in E = \{1, 2, \dots, |E|\}$
- $E_{c,h}$  : 공병부대  $c$ 와 동원완료 여부 인덱스  $h$ 에 따른 보유 장비 집합.  $E_{c,h} \subset E$
- $q$  : 지원팀의 구축능력 인덱스.  
 $q \in Q = \{1, 2, \dots, |Q|\}$
- $a_q$  : 구축능력 인덱스  $q$ 의 구축능력 값
- $g_e$  : 장비  $e$ 의 구축능력
- $p_{q,i}$  : 구축능력 인덱스가  $q$ 인 지원팀이 포병진지  $i$ 를 구축하는 데 걸리는 시간
- $d_{i,j}$  : 포병진지  $i$ 에서  $j$ 로의 이동시간
- $t_k$  : 지원팀  $k$ 의 동원완료 시각
- $A$  : 공병부대가 지원하는 최소 포병부대 수
- $B$  : 지원팀의 최소 구축능력
- $M$  : 충분히 큰 상수

#### ○ Decision variable

- $C_{\max}$  : 모든 포병진지가 구축 완료되는 시각
- $s_i$  : 포병진지  $i$ 의 구축 시작시각
- $x_{k,i,j}$  : 이진변수. 지원팀  $k$ 가 포병진지  $i$ 에 이어 연

속으로  $j$ 를 구축하면 1, 아니면 0

- $y_{k,i}$  : 이진변수. 지원팀  $k$ 가 포병진지  $i$ 를 구축하면 1, 아니면 0
- $u_{k,e}$  : 이진변수. 지원팀  $k$ 에 장비  $e$ 가 할당되면 1, 아니면 0
- $v_{c,f}$  : 이진변수. 공병부대  $c$ 에 포병부대  $f$ 가 할당되면 1, 아니면 0
- $w_{k,q}$  : 이진변수. 지원팀  $k$ 의 구축능력 인덱스가  $q$ 이면 1, 아니면 0

상기한 표기법(Notation)과 결정변수(Decision variable)를 사용하여 다음과 같이 수리모형을 제시한다.

$$\text{Min } C_{\max} \quad (1)$$

*s.t*

$$s_i \geq t_k y_{k,i}, \forall i \in J_0, \forall k \in K \quad (2)$$

$$y_{k,i} = 1, \forall k \in K, \forall i \in J_0 \quad (3)$$

$$s_i \leq s_j, \forall (i, j) \in R \quad (4)$$

$$\sum_{c \in C} v_{c,f} = 1, \forall f \in F \setminus \{0\} \quad (5)$$

$$\sum_{f \in F} v_{c,f} \geq A, \forall c \in C \quad (6)$$

$$\sum_{k \in K_{c,0} \cup K_{c,1}} y_{k,i} = v_{c,f}, \quad (7)$$

$$\forall i \in J_f, \forall f \in F \setminus \{0\}, \forall c \in C$$

$$\sum_{j \in J} x_{k,i,j} = y_{k,i}, \forall k \in K, \forall i \in J \setminus \{|J|+1\} \quad (8)$$

$$\sum_{i \in J} x_{k,i,j} = y_{k,j}, \forall k \in K, \forall j \in J \setminus J_0 \quad (9)$$

$$\sum_{k \in K_{c,h}} u_{k,e} = 1, \forall e \in E_{c,h}, \forall c \in C, \forall h \in \{0, 1\} \quad (10)$$

$$\sum_{q \in Q} w_{k,q} = 1, \forall k \in K \quad (11)$$

$$\sum_{q \in Q} w_{k,q} a_q = \sum_{e \in E_{c,h}} u_{k,e} g_e, \quad (12)$$

$$\forall k \in K_{c,h}, \forall c \in C, \forall h \in \{0, 1\}$$

$$\sum_{q \in Q} w_{k,q} a_q \geq B, \forall k \in K \quad (13)$$

$$s_i + \sum_{q \in Q} w_{k,q} p_{q,i} + d_{i,j} \leq s_j + (1 - x_{k,i,j})M, \quad (14)$$

$$\forall i \in J, \forall j \in J, \forall k \in K$$

$$C_{\max} \geq s_{|J|+1} \quad (15)$$

$$s_i \geq 0, \forall i \in J \quad (16)$$

$$x_{k,i,j} = \{0, 1\}, \forall k \in K, \forall i \in J, \forall j \in J \quad (17)$$

$$y_{k,i} = \{0, 1\}, \forall k \in K, \forall i \in J \quad (18)$$

$$u_{k,e} = \{0, 1\}, \forall k \in K, \forall e \in E; \quad (19)$$

$$v_{c,f} = \{0, 1\}, \forall c \in C, \forall f \in F \setminus \{0\} \quad (20)$$

$$w_{k,q} = \{0, 1\}, \forall k \in K, \forall q \in Q \quad (21)$$

(1)번 식은 모든 포병진지가 구축 완료되는 시각을 나타내는  $C_{max}$ 의 값을 최소화하는 목적식이다.

(2), (3)번 식은 동원완료 시각에 따라 상이한 지원팀의 작전 시작 가능시각에 관한 제약식이다.  $J_0$ 는 지원팀의 작전 시작 가능시각을 결정하기 위해 각 지원팀의 최우선 수행이 요구되는 더미(Dummy) 포병진지의 집합이다.

(4)번 식은 구축 선행관계 제약식으로써 전순위 포병진지의 구축이 후순위 포병진지의 구축보다 먼저 시작되도록 제한한다.

(5), (6)번 식은 공병부대에 포병부대를 할당하는 제약식으로써, 포병부대는 단일 공병부대에 할당하고, 공병부대에 할당되는 포병부대 수는  $A$  이상이다.

(7)~(9)번 식은 지원팀에 포병진지를 할당하는 제약식이다. 각 지원팀은 공병부대에 할당된 포병부대 지역 내 포병진지만 구축 가능하고, 각 포병진지의 구축은 선행 포병진지의 구축과 후행 포병진지의 구축이 수반되어야 한다.

(10)~(13)번 식은 지원팀 편성과 관련한 제약식이다. 장비는 단일 지원팀에 할당되어야 하고, 각 지원팀은 단일 구축능력을 가지며, 지원팀의 구축능력은 할당된 장비 구축능력의 합이다. 이때, 각 지원팀의 구축능력의 하한선은  $B$ 이다.

(14)번 식은 각 포병진지의 구축 시작시각을 결정한다. 각 포병진지의 구축 시작시각은 이전 수행된 포병진지의 구축시작 시각에 구축시간과 이동시간을 더하여 결정된다.

(15)번 식은  $C_{max}$  값을 모든 포병진지의 구축 완료 시각 중 가장 큰 값으로 제약한다.

마지막으로 (16) ~ (21)번 식은 결정변수의 비음 조건 및 이진변수 조건을 나타낸다.

#### 4. 휴리스틱 알고리즘

현실규모 문제는 포병진지 200개 이상으로 구성되어 있다. 그러나 제시한 모형은 문제의 규모가 커짐에 따라 해 도출에 필요한 계산시간과 메모리 공간이 기하급수적으로 증가했다. 모의실험 결과 포병진지를 12개로 설정한 문제는 해를 도출하였으나, 포병진지 15개 이상의 문제는 계산시간과 메모리 부족으로 해결하기 어려웠다.

본 논문에서 제안하는 휴리스틱 알고리즘은 원 문제를 공병부대와 포병부대의 할당 문제, 지원팀 편성 문제, 지원팀 일정계획문제로 분할하여 해결한다.

휴리스틱 알고리즘의 개발을 위해 포병부대 수 4개, 공병부대 수 2개, 지원팀 수 8개, 포병진지 수 12개로 설정한 문제와 실험의 결과를 분석하였다. 다음은 해당 문제의 실험 결과이다.

Table 1은 공병부대에 포병부대를 할당한 결과이고, Table 2는 지원팀 편성 결과이다.

Table 1. Artillery assignment results

Spec.	Assignment
Engineer 1	Artillery 1, Artillery 3
Engineer 2	Artillery 2, Artillery 4

Table 2. Support team formation results

Spec.	Assignment equipments	Ability (m/h)
Support team 1	4, 5, 6, 7, 8, 9	255
Support team 2	1, 2, 3, 10, 11, 12	255
Support team 3	13, 14, 15, 22, 23, 24	255
Support team 4	16, 17, 18, 19, 20, 21	255
Support team 5	25, 26, 27, 34, 35, 36	255
Support team 6	28, 29, 30, 31, 32, 33	255
Support team 7	38, 39, 40, 41, 42	235
Support team 8	37, 43, 44, 45, 46, 47, 48	275

Table 3은 지원팀의 일정계획이 수립된 결과이다.

**Table 3.** Support team schedule

Spec.	Time(hour)											
	6			12			18			24		
Support team 1	Artillery position 1						Artillery position 2					
Support team 2	Artillery position 7						Artillery position 8					
Support team 3							Artillery position 3					
Support team 4							Artillery position 9					
Support team 5	Artillery position 4						Artillery position 5					
Support team 6	Artillery position 10						Artillery position 6					
Support team 7							Artillery position 11					
Support team 8							Artillery position 12					

문제와 실험결과를 바탕으로 다음과 같이 휴리스틱 알고리즘을 제안한다.

#### 4.1 공병부대에 포병부대 할당

Table 4는 각 포병부대의 중심 간 거리를 계산한 값을 나타낸다.

**Table 4.** Distance between each artillery center

Spec.	Artillery 1	Artillery 2	Artillery 3	Artillery 4
Artillery 1		5.27	1.75	7.05
Artillery 2	5.27		6.09	2.76
Artillery 3	1.75	6.09		7.26
Artillery 4	7.05	2.76	7.26	

Table 1에서 공병부대 1에 할당된 포병부대 1과 3은 중심 간 거리가 1.75km이며, 공병부대 2에 할당된 포병부대 2와 4는 중심 간 거리가 2.76km로써 그 합이 4.51km이다. 이는 공병부대에 포병진지가 할당되는 경

우 중 포병부대 중심간 거리의 합이 가장 작은 경우이다.

Table 5는 군집을 형성한 포병부대 지역 내의 구축 최우선순위 집합에 속한 포병진지와 공병부대 간 거리를 나타낸다.

**Table 5.** Distance between engineer and first priority artillery positions in each group

Spec.	Group 1	Group 2
	Artillery 1, Artillery 3	Artillery 2, Artillery 4
Engineer 1	13.52	2.89
Engineer 2	25.48	13.91

Table 5에서 Table 2의 결과와 같이 공병부대 1에 군집 1이 할당되고, 공병부대 2에 군집 2가 할당되면, 해당 군집의 구축 최우선순위 집합에 속한 포병진지와 공병부대 간 거리의 합 27.43km는 반대 경우의 거리 합 28.37km보다 작다.

이를 이용하여 포병부대 중심 간 거리를 최소화하는 군집을 형성하고, 군집화된 포병부대 지역 내의 구축 최우선순위 집합에 속한 포병진지들과 공병부대 간 거리를 최소화하는 것을 목적으로 공병부대에 포병부대를 할당하는 모형을 제안한다.

$$\text{Min} \sum_{c \in C} \sum_{f \in F} \sum_{f' \in F} d_{f,f'} z_{c,f,f'} + \sum_{c \in C} \sum_{f \in F} r_{c,f} v_{c,f} \quad (22)$$

s.t

$$\sum_{c \in C} v_{c,f} = 1, \forall f \in F \quad (23)$$

$$\sum_{f \in F} v_{c,f} \geq A, \forall c \in C \quad (24)$$

$$\sum_{f \in F} z_{c,f,f'} = v_{c,f'}, \forall f' \in F, \forall c \in C \quad (25)$$

$$\sum_{f' \in F} z_{c,f,f'} = v_{c,f}, \forall f \in F, \forall c \in C \quad (26)$$

$$v_{c,f} \in \{0, 1\}, \forall c \in C, \forall f \in F \quad (27)$$

$$z_{c,f,f'} \in \{0, 1\}, \forall c \in C, \forall f \in F, \forall f' \in F \quad (28)$$

#### ○ Notation

- $f$  : 포병부대 인덱스.  $f \in F = \{1, 2, \dots, |F|\}$
- $c$  : 공병부대 인덱스.  $c \in C = \{1, 2, \dots, |C|\}$
- $d_{f,f'}$  : 포병부대  $f$ 의 중심에서 포병부대  $f'$ 의 중심까지의 거리
- $r_{c,f}$  : 공병부대  $c$ 에서 포병부대  $f$ 의 구축 최우선순위

집합에 속한 포병진지까지의 거리

○ Decision variable

- $z_{c,f,f'}$  : 이진변수. 공병부대  $c$ 에 포병부대  $f$ 와  $f'$ 가 모두 할당되면 1, 아니면 0
- $v_{c,f}$  : 이진변수. 공병부대  $c$ 에 포병부대  $f$ 가 할당되면 1, 아니면 0

(22)번 식은 공병부대에 할당된 포병부대 중심 간 거리의 합과 구축 최우선순위 집합에 속한 포병진지와 공병부대 간 거리의 합을 더한 값을 최소화하는 목적 식이다.

(23), (24)번 식은 기존 모형의 (5), (6)번 식과 동일한 제약식이다.

(25), (26)번 식은 공병부대에 할당되는 포병부대를 균집화된 포병부대로 제한한다.

(27)번 식과 (28)번 식은 결정변수의 이진변수 조건을 나타낸다.

4.2 지원팀 편성

군은 원활한 작전수행을 위해 지원팀이 균등한 구축 능력을 갖도록 편성한다. Table 2에서는 비교적 균등한 구축능력의 지원팀 편성결과가 나타났다. 이를 고려하여 지원팀의 구축능력 편중을 경감하는 구축능력 하한선 지정과 관련한 제약식은 다음과 같다.

$$\sum_{k \in K_{c,w}} u_{k,e} = 1, \forall e \in E_{c,h}, \forall c \in C, \forall h \in \{0, 1\} \quad (29)$$

$$\sum_{q \in Q} w_{k,q} = 1, \forall k \in K \quad (30)$$

$$\sum_{q \in Q} w_{k,q} a_q = \sum_{e \in E_{c,w}} u_{k,e} g_e \quad (31)$$

$$\forall k \in K_{c,h}, \forall c \in C, \forall h \in \{0, 1\}$$

$$\sum_{q \in Q} w_{k,q} a_q \geq B, \forall k \in K \quad (32)$$

$$u_{k,e} = \{0, 1\}, \forall k \in K, \forall e \in E \quad (33)$$

$$v_{c,f} = \{0, 1\}, \forall c \in C, \forall f \in F \quad (34)$$

$$w_{k,q} = \{0, 1\}, \forall k \in K, \forall q \in Q \quad (35)$$

(29)~(35)번 식은 기존 모형의 (10)~(13), (19)~(21)번식과 동일한 제약식이다.

4.3 지원팀 일정계획

Table 6은 실험결과를 통해 작성한 공병부대 1에 할당된 포병부대의 포병진지 중 구축 우선순위 3순위 포병진지들에 대한 각 지원팀의 구축완료 시각이다.

Table 6. Construction complete time of each artillery position

Spec.	Artillery position 3	Artillery position 9
Support team 1	24.71	24.68
Support team 2	24.76	24.67
Support team 3	24.44	24.44
Support team 4	24.44	24.44

Table 6에서 각 포병진지의 구축완료 시각이 가장 작은 지원팀은 3, 4이다. 실제 실험 결과 포병진지 3은 지원팀 3이, 포병진지 9는 지원팀 4가 할당받았다. 이러한 결과를 반영하여 다음과 같은 알고리즘을 제안한다.

- <Step 1> 공병부대에 할당된 포병부대 지역 내의 포병진지를 구축 우선순위에 맞춰 정렬한다.
- <Step 2> 같은 구축 우선순위를 갖는 포병진지에 대해 구축 지원이 가능한 지원팀별 구축완료 시각을 계산한다.
- <Step 3> 포병진지별 구축완료 시각의 값이 최소인 지원팀과 해당 구축완료 시각을 저장한다.
- <Step 4> 가장 작은 구축 완료시각을 갖는 포병진지에 해당 구축 완료시각을 갖는 지원팀을 할당한다.
- <Step 5> 지원팀이 할당된 포병진지를 해당 우선순위 집합에서 삭제하고 나머지 포병진지에 대해 <Step 2~5>를 반복한다.
- <Step 6> 해당 구축 우선순위 집합에 남은 포병진지가 없으면 하위 구축 우선순위에서 <Step 2~5>를 반복한다.
- <Step 7> 하위 구축 우선순위가 없다면, 나머지 공병부대에 대하여 <Step 1~6>을 반복한다. 이 과정을 모든 공병부대에 대해 시행한다.

5. 실험결과 및 분석

5.1 실험방법

실험은 Linux 기반의 Intel Core i7-3770 CPU @

3.4GHz×8 성능의 PC에서 진행했으며, 솔루션은 Gurobi 6.51을 사용했다.

2가지 실험을 실시하고 결과를 분석한다. 먼저, 소규모 문제에서 휴리스틱 알고리즘을 적용하여 도출한 해와 최적화 솔루션을 적용하여 도출한 최적해를 비교하여 휴리스틱 알고리즘의 성능을 검증한다. 다음으로 현실규모 문제에 휴리스틱 알고리즘을 적용하여 도출한 계산시간을 통해 휴리스틱 알고리즘의 현실규모 해결 가능성을 검증한다.

## 5.2 소규모 문제

### 5.2.1 실험설계

소규모 문제는 포병부대 수 4개, 공병부대 수 2개, 지원팀 수 8개, 포병진지 수 12개로 설정했다. 공병부대가 보유한 장비의 구축능력은 Table 7과 같다.

Table 7. Construction ability of each equipment

Spec.	Construction ability(m <sup>2</sup> /h)	Spec.	Construction ability(m <sup>2</sup> /h)
Equipment 1	50	Equipment 3	40
Equipment 2	45	Equipment 4	35

공병부대가 지원하는 최소 포병부대 수  $A$ 는 2로 설정했고, 지원팀의 최소 구축능력  $B$ 는 공병부대 보유 장비 구축능력 합 of 40%로 설정했다.

상황별 휴리스틱 알고리즘의 성능을 확인하기 위해 동원완료 시각, 보유 장비 수량, 포병진지간 거리, 포병진지 작업량을 3개의 수준으로 설정하고, 4가지 상황을 조합하여 실험을 구성하고 실험을 진행했다. Table 8은 공병부대의 수준별 동원완료 시각이다.

Table 8. Nongovernment equipment available time

Spec.	Level1	Level2	Level3
Engineer 1	12	9	6
Engineer 2	9	6	3

보유 장비는 수준별로 12대, 9대, 6대이고, 포병진지간 거리는 수준별로 지도상의 거리에 각각 1.5배, 1배, 0.5배를 곱했다. 포병진지 작업량은 3,000m<sup>2</sup>, 2,000m<sup>2</sup>, 1,000m<sup>2</sup>로 설정했다.

### 5.1.2 실험결과 및 분석

Table 9는 소규모 문제의 실험결과를 나타낸다. 실험 결과에서 휴리스틱 알고리즘 해의 최적해와 오차범위는 최저 0.11%에서 최고 18.99%로 나타났다. 평균 6.44% 오차범위를 보였으며, 표준편차는 5.31이다. 오차범위 5% 이내 실험은 총 81개 실험 중 34개로써 41.98%를 차지했다.

Table 9. Optimality gaps of small-size test problem

Spec.	Value
Maximum optimality gap	18.99%
Minimum optimality gap	0.11%
Average of optimality gap	6.44%
Standard deviation of optimality gap	5.31

Table 10은 오차범위가 작은 상위 30%, 24개 실험 결과를 분석한 것으로써 4가지 상황의 수준별 빈도수를 나타낸다.

Table 10. The frequency of each situation level in bottom 30% of small-size test problem optimality gap

Spec.	Level 1	Level 2	Level 3
Nongovernment equipment available time	9	7	8
Equipment number	6	8	10
Artillery position distance	7	7	10
Workload	7	4	13

Table 10에서 포병진지 작업량이 적은경우 즉, 포병진지 작업량 수준 3이 24개 실험 중 13개 실험으로써 54.17%를 차지했다.

## 5.2 현실규모 문제

### 5.2.1 실험설계

현실규모 문제는 포병부대 수 12개, 공병부대 수 4개, 지원팀 수 16개, 포병진지 수 216개로 설정했다. 공병부대가 보유한 장비의 구축능력은 소규모 문제와 동일하다.

공병부대가 지원하는 최소 포병부대 수  $A$ 는 3으로

설정했고, 지원팀의 최소 구축능력  $B$ 는 공병부대 보유 장비 구축능력 합의 40%로 설정했다.

상황별 휴리스틱 알고리즘의 계산시간을 확인하기 위해 소규모 문제와 동일하게 동원완료 시간, 보유 장비 수량, 포병진지간 거리, 포병진지 작업량을 3개의 수준으로 설정하고 서로 다른 조합의 실험을 실시했다. Table 11은 공병부대의 수준별 동원완료 시간이다.

**Table 11.** Nongovernment equipment available time

Spec.	Level1	Level2	Level3
Engineer 1	78	66	54
Engineer 2	72	60	48
Engineer 3	78	66	54
Engineer 4	66	54	48

보유 장비 수량은 수준별로 12대, 9대, 6대로 정했으며, 포병진지간 거리는 수준별로 지도상의 거리에 각각 1.5배, 1배, 0.5배를 곱했다. 포병진지 작업량은 3,000m<sup>2</sup>, 2,000m<sup>2</sup>, 1,000m<sup>2</sup>로 설정했다.

### 5.2.2 실험결과 및 분석

Table 12는 실험결과 도출된 최대 계산시간, 최소 계산시간, 평균 계산 시간을 나타낸다.

**Table 12.** Computation times of large-size test problems

Spec.	Value
Maximum computation time	202.57s
Minimum computation time	30.39s
Average of computation time	79.81s
Standard deviation of computation time	44.75

문제 해결에 평균 79.81초가 소요되었으며, 최장 202.57초, 최단 30.39초가 소요되었다. 해결 시간의 표준편차는 44.75이다.

Table 13은 문제 해결 시간이 짧은 상위 30%, 24개 실험을 분석한 결과이며, 각 상황의 수준별 빈도수를 나타낸다.

**Table 13.** The frequency of each situation level in bottom 30% of large-size test problem optimality gap

Spec.	Level 1	Level 2	Level 3
Nongovernment equipment available time	6	7	11
Equipment number	0	8	16
Artillery position distance	7	8	9
Workload	6	10	8

Table 13에서 보유 장비 수량이 적은 경우 즉, 보유 장비 수량 수준 3이 24개 실험 중 16개 실험으로써 67%를 차지했다.

## 6. 결론

본 논문은 공병 지원팀이 최단시간 내에 모든 포병진지를 구축하기 위한 포병진지 구축계획 문제를 다룬다. 포병진지 구축계획 문제의 효율적 해결을 위해, 해당 문제를 혼합정수계획으로 모형화 하고 현실규모 문제 해결을 위한 휴리스틱 알고리즘을 제안했다.

제안된 수리모형의 해 도출을 위해 필요한 계산시간 및 메모리 공간은 문제규모가 증가함에 따라 기하급수적으로 증가하였다. 따라서 현실규모 문제를 해결하기 위해 원 문제를 세 부분으로 분할하여 해결하는 휴리스틱 알고리즘을 제안하였다. 소규모 문제에서 휴리스틱 알고리즘은 최적해 대비 평균 오차범위 6.44%의 해를 도출했으며, 포병진지 200개 이상의 현실규모 문제를 평균 79.8초에 해결하였다.

본 논문에서는 확정적 상황에서의 포병진지 구축계획 문제 해결을 위한 혼합정수계획 모형과 휴리스틱 알고리즘을 제안하였다. 하지만 전쟁과 같은 불확정적 상황에서 공병부대의 작업환경과 가용자원의 변동에 대한 고려가 필요하다. 예를 들어 도로의 유실이나 토지의 구조변화와 같은 환경변화 및 장비의 고장과 민수용 장비의 수급변동과 같은 가용자원의 변화에 따른 구축계획일정 변동을 고려한 모형의 개발이 필요하다. 차후 연구에서는 불확정적 요소를 고려한 포병진지 구축 문제에 대한 연구의 수행이 요구된다.



## References

- [1] J. H. Moon and S. H. Lee, "Study on vehicle routing problem of artillery position construction for survivability Support", Journal of the Korean Institute of Industrial Engineers, Vol. 37, No. 3, pp. 171-179, 2011. DOI: <http://dx.doi.org/10.7232/JKIIIE.2011.37.3.171>
- [2] P. Brucker, "Scheduling Algorithms", p. 1- 6, Springer, 1995. DOI: <http://dx.doi.org/10.1007/978-3-662-03088-2>
- [3] R. L. Graham, E. L. Lawler, J. K. Lenstra and A. H. G. Rinnooy kan, "Optimization and approximation in deterministic sequencing and scheduling : A survey", Annals of Discrete Mathematics 5, pp.287-326, 1979. DOI: [http://dx.doi.org/10.1016/S0167-5060\(08\)70356-X](http://dx.doi.org/10.1016/S0167-5060(08)70356-X)
- [4] J. Blazewicz, M. Dror and J. Weglarz, "Mathematical programming formulations for machine scheduling : A survey", European Journal of Operational Research 51, pp. 283-300, 2003. DOI: [http://dx.doi.org/10.1016/0377-2217\(91\)90304-E](http://dx.doi.org/10.1016/0377-2217(91)90304-E)
- [5] A. S. Manne, "On the Job-shop scheduling problem", Operations Research, Vol. 8, No. 2, pp. 219-223, 1960. DOI: <http://dx.doi.org/10.1287/opre.8.2.219>
- [6] D. S. Choi and I. C. Choi, "A local search algorithm for jobhop scheduling problems with alternative operations and sequence-dependent setups", Computers&Industrial Engineering 42, pp. 43-58, 2002. DOI: [http://dx.doi.org/10.1016/S0360-8352\(02\)00002-5](http://dx.doi.org/10.1016/S0360-8352(02)00002-5)
- [7] S. Hartmann, D. Briskorn, "A survey of variants and extensions of the resource-constrained project scheduling problem", European Journal of Operational Research 207, pp. 1-14, 2010. DOI: <http://dx.doi.org/10.1016/j.ejor.2009.11.005>
- [8] P. Brucker, A. Drexler, R. Mohring, K. Neumann and E. Pesch, "Resource-constrained project scheduling : Notation, classification, models, and methods", European Journal of Operational Research 112, pp. 3-41, 1999. DOI: [http://dx.doi.org/10.1016/S0377-2217\(98\)00204-5](http://dx.doi.org/10.1016/S0377-2217(98)00204-5)
- [9] E. Mokotoff, "Parallel machine scheduling problems : A survey", Asia-Pacific Journal of Operational Research 18.2, pp. 193-242, 2001.
- [10] E. L. Lawler and J. Labetoulle, "On preemptive scheduling of unrelated parallel processors by linear programming", Journal of the Association for Computing Machinery 25, pp. 612-619, 1978. DOI: <http://dx.doi.org/10.1145/322092.322101>

### 안 문 일(Moon-II Ahn)

[정회원]



- 2008년 3월 : 육군사관학교 (공학사, 군사학사)
- 2014년 7월 ~ 현재 : 고려대학교 산업경영공학과

<관심분야>

경영과학, 스케줄링, 국방

### 최 인 찬(In-Chan Choi)

[정회원]



- 1984년 5월 : 아이오와 주립대학 산업공학과 (공학사)
- 1986년 5월 : 콜롬비아대학 산업공학과 (경영과학 석사)
- 1990년 10월 : 콜롬비아대학 산업공학과 (경영과학 박사)
- 1996년 3월 ~ 현재 : 고려대학교 산업경영공학부 교수

<관심분야>

경영과학, 알고리즘, 이산수학, 계량의사결정