

# 표적 할당과 사격 순서의 동시 결정 문제를 위한 발견적 기법

김동현\* · 이영훈\*

## Heuristic for the Simultaneous Target Allocation and Fire Sequencing Problem

Dong Hyun Kim\* · Young Hoon Lee\*

### ■ Abstract ■

In this study the artillery fire system is investigated in consideration of the characteristics of the troop and the target. Two kinds of decision are to be made on the target allocation with fire ammunition and the fire sequencing for the target with duties in charge. The objective is to minimize the completion time for all troops. Each target has the specified amount of load of fire, which can be accomplished by a single troop or the combination of the troops having different capabilities. Mathematical model is suggested, and the heuristic algorithm which yields a solution within a reasonable computation time is developed. The algorithm consists of iterative three steps : the initial solution generation, the division improvement, and the exchange improvement. The performance of the heuristic is evaluated through the computational experiment.

Keyword : Target Allocation, Fire Sequencing, Artillery Fire System

논문접수일 : 2008년 09월 11일 논문게재확정일 : 2009년 12월 31일

논문수정일(1차 : 2009년 09월 05일)

\* 연세대학교 정보산업공학과

† 교신저자

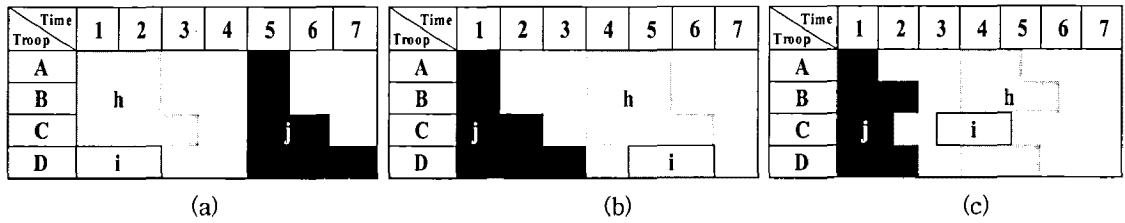
## 1. 서론

과거와 현재까지의 전쟁수행개념은 기동부대 위주의 전투수행에 의해 진행되었으나 미래의 전쟁수행개념은 화력이 결정적 전투수행으로 진보할 것이다. 따라서 현대의 전장 양상도 우수한 무기체계의 확보와 함께 다양한 화력의 통합운용능력에 따라 전장에서 결정적인 역할을 담당하여 변화되고 있으며 각종 무기체계의 효율성을 극대화하기 위한 과학적 운용방법과 능력이 요구되고 있다. 예를 들어 걸프전이 재래식 대량 화력의 중요성과 첨단무기의 위력이 시현된 장이었다면 이라크와 아프가니스탄 전에서는 재래식 무기체계와 최첨단의 정밀유도 무기체계의 효율적인 동시 통합위주의 화력전 양상을 보였다. 이와 같이 현존하는 각종 무기체계에 대한 통합 화력운용방법의 중요성과 연계하여 효율성이 강조되고 있는 것이 현실이다. 현대 화력전의 수행 및 계획의 주체이자 현존하는 대표적인 사격방법 및 수단으로써 포병사격을 들 수 있다. 특히, 신속한 표적 할당과 사격 순서 결정은 화력의 효과적인 운용방법의 중요한 요소이다. 신속한 포병사격은 사격 후 수분 이내에 사격한 부대의 위치가 탐지수단에 노출되어 상대방의 대응사격으로부터 생존을 위협받을 수 있다. 이에 신속한 사격임무는 적의 위협을 최소화시킬 수 있는 효과적이고 적극적인 생존방법일 뿐만 아니라 다양한 포병부대의 화포 특성을 고려한 화력의 효율적인 통합운용으로 작전의 목표달성을 극대화시킬 수 있는 방법이다. 표적 할당(Target Allocation)은 사격부대에 표적의 전체 또는 일부에 대한 사격할 포탄의 양을 할당하는 것이며 사격 순서(Fire Sequencing)의 결정은 사격부대별 할당된 표적들에 대한 사격할 순서의 조합을 말한다. 전장상황에서 사격부대가 사격임무를 수행하는 시간을 단축하는 만큼 아군은 적군으로부터 사격하는 부대의 사격위치가 탐지되어 사격부대를 위협하는 사격을 회피할 수 있는 구체적인 생존방법 중에 하나로서 단시간 집중 사격 후 사격한 진지를 신속히 이탈하는 타격수단의 기본운용방법인 사격

과 진지이탈(Shoot and Scoot)의 개념(포병운용, FM 2-2, 2007)을 고려할 때 작전 개시 전 미리 사격할 표적을 할당하고 사격 순서를 계획하는 상황뿐만 아니라 작전 진행 간 사격이 요구된 표적을 최단시간 내 할당함과 동시에 사격 순서를 결정하는 방법은 중요하다. 특히, 사격부대가 수직 열세인 경우 단시간 내 사격이 요구되는 대량 표적의 발생은 사격을 계획하는데 많은 제한이 따르며 신속성이 배제되어 효과적인 사격이 될 수 없다. 본 연구는 작전 개시 전과 진행 중에 사격부대별 능력의 차이를 고려하고 최단시간 내 표적 할당과 동시에 사격 순서를 계획하여 작전의 신속성과 동시에 사격부대의 생존성 추구에 밀접한 사격임무를 수행하는 시간을 단축시켜 작전의 효율성을 달성하는데 목적이 있다. 이 문제를 동시 할당 및 순서결정 문제(Simultaneous Allocation and Sequencing : SAS)라고 정의한다. 이에 전체 사격완료시간을 최소화하는 문제로 다음과 같이 정의한다.

- 사격부대와 표적은 다수이다.
- 표적에 대한 사격의 우선순위는 동일하다.
- 사격부대별 능력(효율, 속도)이 서로 다르다.
- 사격부대의 능력에 따라 각각 사격 할 수 있는 표적이 다르다.
- 사격부대는 1개 표적을 사격한 후 다음 표적을 사격하기 위한 준비시간이 필요하다.
- 사격부대는 한 번에 1개 표적만 사격 할 수 있다.
- 1개 표적에 1개 이상 사격부대가 사격할 수 있으며 2개 이상 사격부대가 참여 시 시작시간은 같다.
- 사격부대별 포탄 1발에 대한 사격의 효과와 소요 시간은 각각 다르다.
- 사격 발수와 각종 시간은 정수(integer)이다.
- 목적함수는 최종 사격완료시간의 최소화이다.

일반적인 사격 순서결정 문제는 사격부대가 사격할 표적이 미리 결정된 상태에서 사격에 소요되는 시간을 최소화하는 사격 순서를 결정하는 것이다. 본 연구에서는 사격부대에 사격할 표적을 할당하고 할



[Figure 1] Examples of how to change target allocation and fire sequencing

당된 표적을 사격하는 순서를 동시에 결정하는 문제를 다룬다.

[Figure 1]은 사격 순서를 결정하는 문제의 예로써 (a)와 (b) 같이 기존 문제의 순서 결정문제는 표적할당이 완료된 후 표적의 형태가 변하지 않고 사격 순서만을 결정하여 좋은 해를 구하는 방법으로 표적이 사격부대에 사전 할당된 상태로 표적에 대한 사격부대의 변화 없이 사격 순서를 결정한다. 이는 최초 표적은 h, i, j로 표적별 사격하는 사격부대는 사전 고정되어 있는 상태이며 사격은 표적 h(사격부대 A, B, C) → i(D) → j(A, B, C, D) 순으로 사격하며 최초 사격부대에 할당된 표적 형태의 변화 없이 표적들의 사격 순서를 표적 j → h → i 순으로 변경함으로써 좋은 해를 구하는 것이다. 본 연구는 (c)와 같이 표적할당이 완료된 후 표적의 형태가 변하여 해를 구하는 방법으로 표적에 대한 기존 사격수요(Demand)는 변경 없이 사격부대별 능력과 표적에 대한 사격할 발수를 고려하여 표적의 형태를 변경하고 사격부대에 할당함과 동시에 사격 순서를 결정한다. 이는 표적 j(사격부대 A, B, C, D) → i(C) → h(A, B, D) 순으로 사격부대에 변경된 표적의 형태로 할당함으로써 (a)와 (b)의 방법과는 다른 문제의 성격을 갖고 있다. 이에 (c)는 사격의 특수성을 포함한 사격계획체계에 대한 사격 순서결정 연구의 새로운 접근방법으로 사격부대별 능력의 차이와 표적의 크기인 사격수요를 고려하여 표적별 사격에 참가하는 사격부대의 선정과 함께 사격 발수, 사격 순서를 동시에 고려하여 최적의 사격계획을 수립하는 방법이다. 사전에 표적을 할당하는 시간과 노력을 표적 할당과 사격 순서를 동시에 고려함으로써 전장에서의 각종 상황과 여건에 따라 변화

하는 유동성을 감안하여 표적할당과 사격 순서를 동시에 결정하는 복합적인 접근방법이라 할 수 있다.

## 2. 관련 연구

본 연구는 제한된 시간 또는 시간 단축을 위한 목표를 달성하기 위해 각종 작업(job) 또는 활동(activity)의 상관관계를 계획하는데 있어 다양한 표적과 사격부대의 특성에 따른 작업 및 활동의 조화를 이루어 총체적인 효율성을 높여야 하는 특성을 갖고 있는 병렬기계(parallel machines)에서의 스케줄링 문제의 유형을 가지고 있다. Sundararaghavan and Ahmed(1984)는 동일한 병렬기계(identical machines)에 대한 최적 스케줄링 알고리즘을 제시하였다. 보편적이고 현실적인 생산 공정의 현장 상황을 고려하여 Plekar et al.(1991), Martell et al.(1997), Yu et al.(2002) 등은 서로 다른 능력의 병렬기계(uniform machines)에서 MS(makespan)와 평균 시스템 내 머문 시간(mean flow time)에 관한 연구를 제시하였으며 Hariri and Potts(1991)는 기계의 능력이 서로 다른 병렬기계(unrelated machines)에서 총 처리시간을 최소화시키는 일정계획으로 1단계에서 선형계획법으로 몇몇 작업을 기계에 할당하고, 2단계에서 나머지 작업 할당하는 2단계(two-phase) 발견적 기법에 대해서 연구하였다.

표적 할당 및 사격 순서 문제에 관한 기존 연구로는 권오정(1997)이 최초 확정적 모형의 접근으로 표적에 사격하는 스케줄링을 이용한 다중처리(multi-purpose) 일정계획 문제로 하나의 작업이 여러 설비를 공유할 때 시작시간과 완료시간이 동일하다는 조건을 만족하는 경우로 국한하며, 1단계에서 사전

표적 할당을 완료한 후 2단계에서의 할당된 표적으로 사격 순서 결정 문제에 대한 연구를 하였다. 그리고 김기호(2002)는 권오정 외(1997)의 연구를 기초로 제시한 모형으로 개별생산 시스템 수리모형을 이용하여 하나의 작업이 여러 설비를 공유할 때 시작시간만 동일하다는 조건을 만족하는 경우로 사격 순서 결정 문제를 개선하였다. 또한 김기호(2002)는 개별생산 시스템 문제를 포병사격문제로 해석하여 이론적인 틀을 마련하였으며 김태현, 이영훈(2003)은 사전 표적 할당이 완료된 여러 표적 중 2개의 표적을 1개의 조합(Pair)으로 구성하여 사격 순서를 결정하고 전체 사격완료시간을 최소화하는 사격 순서 문제로 발견적 기법을 제시하였다. 황원식과 이재영(2007)은 전장의 여러 전술적인 상황을 고려하며 개선된 사격 순서 문제를 연구하였다.

이와 같이 일반적인 사격 순서 결정에 관한 연구들의 공통점은 사전에 사격부대별 표적 할당이 완료되어 계획된 표적만으로 사격 순서를 고려하거나 표적 할당 후 다음단계에서 사격 순서를 결정하는 문제를 연구하였으나 본 연구는 전장 상황을 기초로 효율성과 유동성을 고려하여 사격이 요구되는 표적들을 최단시간 내 사격부대별 표적을 할당함과 동시에 사격부대의 표적별 사격 순서를 결정하는 문제로 일반적인 사격 순서 결정 문제와 다른 복합적인 문제라고 할 수 있다. 특히 본 연구가 황원식, 이재영(2007)과 다른 점은 첫째 기존의 가정이 표적에 대해 발사되는 사격부대와 사격발수가 정해져 있는데 반하여 본 연구는 사격부대와 사격발수를 효과를 고려하여 결정하는 문제이며 둘째, 표적의 이동시 일정시간의 준비시간이 필요한 현실적인 제한을 기존의 연구에서 무시한 반면 본 연구에서는 준비시간이 반영된 실질적 연구라고 할 수 있다.

### 3. 최적화 수리모델

표적 할당과 사격 순서의 동시 결정 문제인 포병 사격체계를 수리모형으로 표현하면 다음과 같다.

#### • Notation

$i$  : 표적( $i \in I$ )

$j$  : 사격부대( $j \in J$ )

$k$  : 1개의 사격부대가 사격하는 표적의 사격 순서( $k=1, 2, \dots, K$ )

$W_j$  : 사격부대  $j$ 가 사격하는 표적  $i$ 의 집합

$F_{ij}$  : 사격부대  $j$ 가 표적  $i$ 의 사격 가능 여부,  
 $F_{ij} \in \{0, M\}$

$D_i$  : 표적  $i$ 의 사격수요(Demand)

$E_j$  : 사격부대  $j$ 가 사격하는 포탄 1발의 효과(Efficiency)

$PT_j$  : 사격부대  $j$ 가 표적을 1발 사격하는데 소요되는 시간(Processing Time)

$SUT_j$  : 사격부대  $j$ 가 1개 표적을 사격한 후 다른 표적 사격에 필요한 준비시간(Setup Time)

#### • Decision Variable

$ST_{ij}$  : 사격부대  $j$ 가 표적  $i$ 를 사격한 시작시간(Starting Time)

$CT_{ij}$  : 사격부대  $j$ 가 표적  $i$ 를 사격한 완료시간(Completion Time)

$MT$  : 최종 사격완료시간(Makespan Time)

$R_j$  : 사격부대  $j$ 가 표적  $i$ 를 사격한 포탄의 발수(Round)

$X_{ijk}$  : 사격부대  $j$ 가 표적  $i$ 를  $k$ 번째 사격할 경우 1, 아니면 0

#### • Formulation

Minimize  $MT$

s.t

$$MT \geq CT_{ij} \quad \forall i, j \quad (1)$$

$$ST_{ij} = ST_{iy} \quad \forall i, y, j, \quad j, y \in J, \quad j \neq y \quad (2)$$

$$CT_{ij} = ST_{ij} + PT_j R_j \quad \forall i, j \quad (3)$$

$$CT_{ij} + SUT_j - M(1 - X_{ijk}) \leq ST_{hj} + M(1 - X_{hjk+1}) \quad \forall i, h, j, \quad k(k=1, \dots, K-1) \quad i, h \in I, \quad i \neq h \quad (4)$$

$$M \times \sum_k X_{ijk} \geq R_j \quad \forall i, j \quad (5)$$

$$\sum_j R_j E_j \geq D_i \quad \forall i \quad (6)$$

$$\sum_i X_{ijk+1} \leq \sum_i X_{ijk} \quad \forall j, k(k=1, \dots, K-1) \quad (7)$$

$$\sum_i X_{ijk} \leq 1 \quad \forall j, k \quad (8)$$

$$\sum_k X_{ijk} \leq 1 \quad \forall i, j \quad (9)$$

$$R_{ij} \leq F_{ij} \quad \forall i, j \quad (10)$$

$$\sum_k X_{ijk} \leq F_{ij} \quad \forall i, j \quad (11)$$

$$X_{ijk} \in \{0, 1\} \quad \forall i, j, k \quad (12)$$

$$ST_{ij}, CT_{ij}, MT, R_{ij} \geq 0 \quad \forall i, j \quad (13)$$

$$M \text{은 큰 상수} \quad (14)$$

목적함수는 주어진 전체 표적에 대한 최종 사격 완료시간을 최소화하는 것이며 식 (1)에서  $MT$ 는 사격부대  $j$ 가 표적  $i$ 를 최종 사격을 완료하는 시간인  $CT_{ij}$  보다 크거나 같은 것을 의미하며, 식 (2)는 1개 표적을 2개 이상의 사격부대가 사격 시 표적을 동일한 시작시간으로 사격해야 하며, 식 (3)은 사격부대의 사격완료시간은 사격시작시간에 1발에 대한 사격소요시간을 사격 발수를 고려한 시간을 의미한다. 즉, 1발의 사격소요시간을 사격발수로 곱하여 산출된 시간이 순수하게 표적을 사격하는 시간이 된다. 식 (4)는 사격부대는  $k$ 번째의 표적을 사격한 이후  $k+1$ 번째의 표적을 사격하기 위해서는 사격부대별 사격준비시간이 필요하다는 것을 의미한다. 즉,  $X_{ijk} = 1, X_{ijk+1} = 0$ 인 경우  $ET_{ij} + SUT_j \leq M$ 이며  $X_{ijk} = 0, X_{ijk+1} = 1$ 인 경우  $-M \leq ST_{hj}, X_{ijk} = 1$ 이다.  $X_{ijk+1} = 1$ 인 경우  $ET_{ij} + SUT_j \leq ST_{hj}, X_{ijk} = 0, X_{ijk+1} = 0$ 인 경우  $-M \leq M$ 으로 성립된다. 식 (5)는 사격부대가 사격 시에만 표적에 대한 사격 발수가 할당이 되며 식 (6)은 사격부대별 1발의 포탄효과와 사격 발수를 고려하여 표적에 필요한 사격수요를 만족한다. 식 (7)은 1개의 사격부대가 표적을  $k$ 번째로 사격해야만 다음 표적을  $k+1$ 번째로 사격이 가능하다는 의미로 1개의 사격부대가 1개의 표적을  $k$ 번째로 사격하지 않고서는  $k+1$ 번째에 다른 표적을 사격할 수 없다는 것을 의미한다. 식 (8)은 1개의 사격부대는  $k$ 번째에 표적 1개만 사격할 수 있으며 식 (9)는 1개의 사격부대는 1개의 표적에 대해 한번만 사격이 가능하다는 것을 의미하는 것으로 선후행 관계를

나타낸다. 식 (10)은 표적별 사격이 가능한 사격부대가 사전에 지정이 되어 있어 사격을 해야만 사격 발수가 할당이 되며 식 (11)은 사격이 가능하면  $M$ , 불가능하면 0(Zero)이 되며 식 (12)~식 (14)는 결정변수 값의 범위를 위한 제약조건이다.

위 문제는 혼합정수계획법(mixed integer programming)문제로서 최적해를 구하는데 많은 시간이 소요되는 문제점을 가지고 있다. 실제로 수리모델에 의한 작은 규모의 실험 예제로 CPLEX 10.1을 이용하여 최적해를 산출해본 결과 사격부대 3개와 표적 6개인 경우 최소 14분에서 최대 1시간 이상의 계산시간이 소요되었으며 사격부대 4개와 표적 10개인 경우 최소 6시간 34분이 소요되었다. 본 논문에서는 현장에서 적용 가능하도록 효과적인 발견적 기법을 제시하여 빠른 시간 내에 근사해를 구하고자 한다.

#### 4. 발견적 기법

권오정 외(1997)의 표적 할당 후 사격 순서 결정 문제에 대한 발견적 기법은 동일한 표적에 대해 할당된 사격부대의 수가 많은 표적에 우선순위를 두고 할당하는 방법으로 구성과정은 시간 축을 설정하고 한 단위씩 증가시켜 할당이 완료되지 않은 표적에 대해 해당 사격부대가 많이 남은 표적을 우선 할당하여 대체적으로 우수한 성능을 보이나 우선 할당할 수 있는 표적 중에서 선택함으로써 좋지 않은 할당과정을 생성할 수 있다. 이에 본 연구는 1개 이상의 사격부대를 표적에 할당함과 동시에 사격 순서를 결정하며 사격부대별 포탄 1발에 대한 사격효과와 소요시간 뿐만 아니라 준비시간을 고려한다. 이는 표적과 사격부대의 조합에 따라 시간을 단축할 수 있는 방법으로 전체 표적에 대한 할당과 사격하는 순서의 조정은 기존 표적 할당의 문제와 일반적인 사격 순서 결정 문제를 종합한 하나의 새로운 문제로서 정의하고자 함에 복잡성이 증대되었다.

본 연구에서 제시하는 발견적 기법은 초기해(initial solution)의 생성과 해의 개선을 위한 분할(Di-

vision)과 교환(Exchange)단계로 구성되어 있으며 산출된 초기해를 개선하기 위해 분할과 교환의 상호 보완적인 개선절차를 거쳐 최적해에 유사한 값으로 발전시킬 수 있는 방법이다. 초기해 생성은 표적의 사격수요와 사격부대의 능력을 우선 고려하여 사격 수요가 큰 표적부터 효율성이 높은 사격부대에 할당시켜 전체 표적이 할당될 때까지 지속한다. 이후 분할단계는 분할절차(Division procedure), 조정(Adjustment)과 평가(Evaluation)를 거쳐 해의 개선 여부에 따라 교환단계로의 이동이 결정된다. 분할절차는 최대 사격완료시간의 사격부대가 사격하는 표적 중 최대 사격수요의 표적부터 표적의 일부를 분할하여 다른 사격부대에 나누어 할당하는 방법으로 표적을 여러 사격부대가 동시에 사격함으로써 사격완료시간을 단축할 수 있다. 조정 및 평가는 표적과 표적 사이의 사격하지 않는 여유시간에 다른 표적을 이동시켜 사격완료시간을 단축할 수 있는 방법으로 여러 사격부대가 동시에 표적을 사격하기 위해 발생한 대기시간 안으로 다른 표적의 위치를 이동시켜 시간을 단축할 수 있으며 이를 통해 해의 개선여부를 평가한다. 교환단계는 교환절차(Exchange procedure), 조정과 평가를 거쳐 해의 개선 여부에 따라 교환단계의 반복 및 종료가 결정된다. 교환절차는 사격부대별로 사격하는 표적을 상호 교환하여 사격완료시간을 단축하는 방법으로 최대 사격완료시간의 사격부대가 사격하는 표적과 최소 사격완료시간의 사격부대가 사격하는 표적 중 일부를 1대 1로 상호 교환하여 사격완료시간을 단축할 수 있다. 이후 조정 및 평가방법은 분할단계에서 설명한 방법과 동일하다. 위와 같이 초기해에서 개선의 절차를 거쳐 시간단축의 효과를 얻기 위한 방법으로 신속한 해의 산출이 가능하다.

#### 4.1 Initial solution generation

알고리즘의 초기해 생성방법은 최초 표적의 사격 수요가 큰 표적부터 가장 효율적인 사격부대에 우선적으로 표적을 배정하는 Greedy Rule로 모든 표

적을 순차적으로 할당하여 초기해를 생성한다.

#### Initial Algorithm

Step 1 : Set $I' = I, W_j = \emptyset \quad \forall j$
Step 2 : Select the highest target $i^* \in I'$ (The tie breaks arbitrarily)
Step 3 : $R_{i^*,j} = \lfloor D_{i^*} / E_j \rfloor,$ $CT_{i^*,j} = MT_j + R_{i^*,j} \times PT_j + SUT_j \quad \forall j$
Step 4 : Select $j'$ such that $CT_{i^*,j'} = \min_j CT_{i^*,j}$
Step 5 : $W_{j'} = W_{j'} + \{i^*\}$
Step 6 : $MT_{j'} = CT_{i^*,j'}$
Step 7 : Delete $i^*$ from $I'$
Step 8 : If $I' = \emptyset$ , stop, else, go to Step 2

초기해 생성에서 Step 1)사격할 표적들을 할당할 표적의 집합( $I'$ )으로 구성하여 Step 2)표적의 집합에서 표적의 사격수요( $D_i$ )가 제일 큰 최대 표적( $i^*$ )을 선정하며 이때 동일한 사격수요의 표적들이 존재 시 임의로 선택한다. Step 3)이후 최대 표적의 사격수요( $D_{i^*}$ )를 사격부대별 사격하는 포탄 1발의 효과( $E_j$ )에 의한 사격부대별 소요되는 사격 발수( $R_{i^*,j}$ )를 산출하며, 이 때 모든 사격수요를 만족하기 위해 사격하는 1발 미만의 발수는 절상한다. 이후 사격부대별 산정된 사격 발수는 소요되는 시간의 비례에 따라 최대 표적에 대한 사격완료시간( $CT_{i^*,j}$ )을 결정할 수 있으며 Step 4)최대 표적을 사격함에 있어 최소 사격완료시간의 사격부대( $j'$ )를 선정한다. Step 5)사격부대별 최초 할당 받은 최대표적을 포함하여 이후 할당된 표적을 추가하여 해당 사격부대가 사격할 표적의 집합( $W_{j'}$ )을 구성한다. Step 6)최대 표적을 포함한 사격완료시간( $CT_{i^*,j'}$ )은 해당 사격부대의 최종 사격완료시간( $MT_{j'}$ )이 된다. Step 7)최초 할당된 최대 표적을 표적의 집합에서 제외한 후 Step 8)모든 표적이 사격부대에 할당될 때까지 Step 2~8의 과정을 반복한다.

초기해를 구하는 과정을 4개의 사격부대와 7개의 표적의 예를 들어 설명하면 다음과 같다. <Table 1>은 사격부대의 유형별 능력의 차이로써 사격 시 1발의 효과(Efficiency)와 소요시간(Processing Time),

사격준비시간(Setup Time)을 나타낸다.

<Table 1> Types and Characteristics of Troops

Type	A	B	C	D
$E_j$	1	2	3	5
$PT_j$	20sec	30sec	60sec	120sec
$SUT_j$	240sec	300sec	60sec	120sec

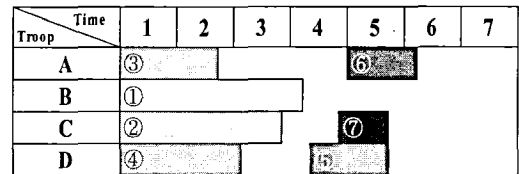
또한 <Table 2>에서 사격부대는 A, B, C, D 유형 별 1개씩 총 4개를 대상으로 하고 표적은 총 7개이며 사격수요( $D_j$ )는  $D_1 = 22, D_2 = 15, D_3 = 10, D_4 = 7, D_5 = 5, D_6 = 4, D_7 = 3$ 이다. 최대 표적( $i^*$ )인 1번째 표적의 사격수요( $D_1$ ) 22는 A사격부대가 사격효과( $E_j$ ) 1로써 22발을 사격해야 되며 이때 1발에 대한 사격 소요시간( $PT_j$ )인 20초를 고려하여 사격완료시간( $MT_j$ )은 440초가 된다는 것을 의미하며 C사격부대는 사격효과 3으로써 사격수요 22를 충족하기 위해서는 7.3발만이 필요하나 8발로 절상하여 사격해야 되며 사격소요시간 60초를 고려하여 사격완료시간은 480초가 된다. 최초 각 사격부대는 사격준비가 완료된 상태로 별도의 준비시간은 필요하지 않으며 이후 할당받는 표적을 사격하기 위해서는 사격부대별 상이한 사격준비시간이 필요하다. 위에서 설명한 바와 같이 한 번의 과정에 1개의 표적이 선정되며 이는 가장 효율적인 사격부대에 배치되고 각 사격부대에 1개 이상의 표적이 할당된 후 사격준비시간( $SUT_j$ )

<Table 2> Example of result of  $R_{i^*,j}, MT_j$

Troop	A		B		C		D	
	$R_{i^*,j}$	$MT_j$	$R_{i^*,j}$	$MT_j$	$R_{i^*,j}$	$MT_j$	$R_{i^*,j}$	$MT_j$
① 22	22	440	11	330	8	480	5	600
② 15	15	300	8	870	5	300	3	360
③ 10	10	200	5	780	4	660	2	240
④ 7	7	580	4	750	3	600	2	240
⑤ 5	5	540	3	720	2	540	1	480
⑥ 4	4	520	2	690	2	540	1	720
⑦ 3	3	820	2	690	1	480	1	720

240초를 포함하여 사격완료시간은 480초가 되어 가장 빠른 시간 안에 사격이 완료되는 D사격부대에 5번째 표적이 할당된다. 이와 같은 과정의 모든 표적이 할당될 때까지 지속되며 완성된 할당 및 순서는 [Figure 2]와 같다. 유형별 사격부대의 포탄 1발의 효과, 사격 소요시간과 준비시간을 고려하여 초기해를 산출하는 과정은 <Appendix 1>과 같다.

이때 사격부대별 동일한 사격완료시간이 산출된 경우에는 최소 발수로 사격할 수 있는 사격부대 중 임의로 표적을 할당하며 초기해 생성원칙에 따라 1개의 표적은 1개의 사격부대에 1대 1로 할당되어짐을 알 수 있다.



[Figure 2] Example of initial solution

### 4.2 Solution Improvement

초기해는 표적의 사격수요와 사격부대의 효율성에 따라 사격부대에 표적을 1대 1로 할당하는 Greedy Rule로써 이를 개선하기 위한 과정을 통해 개선해를 도출하고자 한다. 개선해는 사격할 표적들을 조합하여 사격완료시간을 단축하는 방법으로써 2개 이상의 사격부대가 1개의 표적을 사격하면서 사격소요시간을 단축하고 가장 큰 표적을 가장 효율적인 사격부대가 사격하는 방법보다는 표적들을 적절하게 조합하고 사격부대를 조정하여 다른 효과적인 사격부대가 사격하는 것이 오히려 전체 사격완료시간을 단축하는 방법일 수 있다는 개념이다. 이에 개선해는 모든 표적이 순차적으로 할당된 초기해를 바탕으로 분할(division)과 교환(exchange)의 2단계 방법으로 해를 개선한다. 표적의 사격수요의 크기와 사격부대의 효율성이 높은 순서에 따라 생성한 초기해를 바탕으로 개선의 절차를 진행하기 위해 사격수요가 큰 표적을 2개 이상의 사격부대가 분할

하여 사격함으로써 표적을 사격하는 완료시간을 단축한다. 그러나 표적의 사격수요를 분할할 수 없거나 분할하여 사격해도 시간단축의 효과가 없는 경우 사격부대별로 표적을 상호 교환함으로써 시간을 단축하는 방법으로 표적과 사격 순서를 동시에 조정하여 최적의 사격부대와 표적의 조합을 찾는 효율적인 방법이 될 수 있다.

#### 4.2.1 Division

개선해 생성방법의 1단계인 분할은 1개의 사격부대가 사격하는 1개의 표적을 다른 사격부대와 함께 사격을 함으로써 표적에 대한 사격소요시간을 단축할 수 있는 방법이다. 이때 1개 표적에 대해 다른 사격부대가 사격에 추가하여 사격 시에는 사격부대별 사격준비시간의 고려가 필수적이며 전체 사격완료시간의 단축이 있어야 한다.

#### Division Algorithm

<p>Step 1 : Select <math>j^*</math> with <math>\max_{j \in J} MT_j</math> (The tie breaks arbitrarily)</p> <p>Step 2 : Set <math>W'_{j^*} = W_{j^*}</math>, <math>BestMT = MT_{j^*}</math></p> <p>Step 3 : Select <math>i^*</math> with <math>\max\{D_i \mid i \in W'_{j^*}\}</math></p> <p>Step 4 : Divide <math>D_{i^*}</math> into  <math>\max\{0, \lceil (D_{i^*} - E_{j^*})/E_{j^*} \rceil = R_{i^*,j^*}'\}</math>  and <math>\lceil E_{j^*}/E_{j^*} \rceil = R_{i^*,j}</math></p> <p>Step 5 : If <math>\max\{0, \lceil (D_{i^*} - E_{j^*})/E_{j^*} \rceil = R_{i^*,j^*}'\} &gt; 0</math>, insert <math>R_{i^*,j^*}'</math> to <math>j^*</math> and <math>R_{i^*,j}</math> to <math>j</math>, else, go to Step 10</p> <p>Step 6 : If <math>MT_{j^*} \geq MT_j + SUT_j + R_{i^*,j} \times PT_j</math>, allocate <math>R_{i^*,j^*}'</math> to <math>j^*</math>, <math>R_{i^*,j}</math> to <math>j</math>, else, not allocate and go to Step 10 <math>\forall j</math></p> <p>Step 7 : <math>BestMT = MT_{j^*} - PT_{j^*}</math></p> <p>Step 8 : If <math>BestMT \leq MT_{j^*}</math>, best improvement = current solution</p> <p>Step 9 : <math>D_{i^*} = D_{i^*} - E_{j^*}</math> and go to Step 1</p> <p>Step 10 : Delete <math>i^*</math> from <math>W'_{j^*}</math></p> <p>Step 11 : If <math>W'_{j^*} = \emptyset</math>, go to Step 3, else, stop</p>
--

개선해의 1단계 분할에서 Step 1)최대 사격완료시간( $\max MT_j$ )의 사격부대( $j^*$ )를 선정하고 Step 2) 선정된 사격부대( $j^*$ )가 사격하는 표적의 집합( $W'_{j^*}$ )

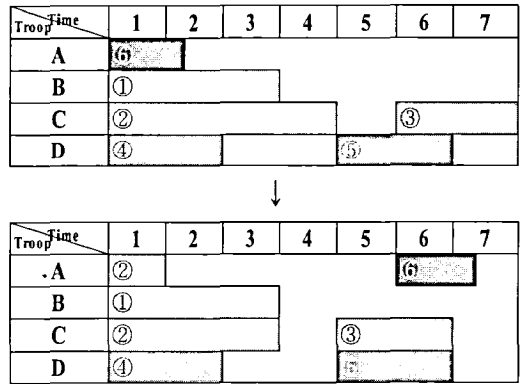
으로 구성하며 현재의 최대 사격완료시간이 현재해 ( $BestMT$ )가 된다. Step 3)최대 사격완료시간의 사격부대( $j^*$ )가 사격하는 표적들 중 표적의 사격수요 ( $D_i$ )가 가장 큰 최대 표적( $i^*$ )과 최소 사격완료시간 ( $\min MT_j$ )의 사격부대( $j'$ )를 선정한다. Step 4)최대 표적의 사격수요( $D_{i^*}$ )를 최소 사격완료시간 사격부대( $j'$ ) 1발의 효과( $E_{j'}$ )를 고려하여  $D_{i^*} - E_{j'}$ 와  $E_{j'}$ 로 최대 사격완료시간의 사격부대가 사격하는 최대 표적의 일부를 최소 사격완료시간의 사격부대가 사격할 수 있는 1발로 표적의 일부를 분할한다. Step 5)이 때 최대 표적의 포탄 1발을 삽입 가능한 최소 사격완료시간 사격부대( $j'$ )에 할당하며 할당할 수 없으면 Step 9로 이동한다. Step 6)최소 사격완료시간의 사격부대( $j'$ )에 포탄 1발을 할당시킨 최대 사격완료시간 사격부대( $j^*$ )의 포탄 1발에 의해 사격소요시간( $PT_{j^*}$ )이 감소되는 사격완료시간( $MT_{j^*}$ )과 포탄 1발을 할당받아 최소 사격완료시간 사격부대( $j'$ )의 변화하는 사격준비시간( $SUT_{j'}$ )과 소요시간( $PT_{j'}$ )이 증가된 사격완료시간( $MT_{j'}$ )을 산출한다. 이와 같이 Step 7)최대 사격완료시간은 포탄 1발에 대한 사격소요시간 만큼 감소되며 Step 8)최대 사격완료시간의 감소가 있으면 현재해는 개선해가 된다. Step 9)이후 최대 표적( $i^*$ )의 사격수요( $D_{i^*}$ )는 다른 사격부대에 할당시킨 1발의 사격수요( $D_{i^*}$ ) 만큼 감소하며 Step 1로 이동하여 더 이상 할당할 수 없을 때까지 반복한다. Step 10)최대 사격완료시간의 사격부대가 사격하는 표적의 집합 내에서 분할되어 할당되거나 분할할 수 없는 표적을 제외시키며 Step 11)다음의 최대 표적을 대상으로 반복한다.

1단계 분할방법은 초기해에서 최대 사격완료시간의 사격부대가 사격할 표적 중 다른 사격부대에 표적의 사격 발수를 분할하여 할당함으로써 시간을 단축하는데 목적이 있다. 사격부대는 A, B, C, D 유형별 1개씩 총 4개를 대상으로 하고 표적은 총 6개이며 사격수요( $D_i$ )는  $D_1 = 6$ ,  $D_2 = 12$ ,  $D_3 = 6$ ,  $D_4 = 10$ ,  $D_5 = 10$ ,  $D_6 = 4$ 이며 포탄 1발의 효과(Efficiency)와 사격소요시간(Processing Time), 사격준비시



간(Setup Time)은 <Table 1>과 같다. [Figure 3]와 같이 Step 1에서 최대 사격완료시간의 C사격부대를 선정하고 Step 2에서 C사격부대가 사격하는 2, 3번 표적을 집합으로 구성하며 사격완료시간 420초는 현재해가 된다. Step 3에서 C사격부대가 사격하는 2, 3번 표적 중 사격수요가 가장 큰 2번 표적과 최소 사격완료시간의 A사격부대를 선정하며 Step 4에서 C사격부대 1발은 A사격부대의 3발의 효과와 동일하며 C사격부대가 사격하는 2번 표적의 사격수요 12발을 포탄 1발의 사격수요 3발을 분할하여 A사격부대에 할당한다. Step 5에서는 2번 표적의 포탄 1발을 A사격부대에 할당하며 Step 6에서 A사격부대에 C사격부대의 포탄 1발을 할당시켜 C사격부대의 사격완료시간은 60초 단축되어 360초가 되며 C사격부대로부터 할당된 1발은 A사격부대에서 3발로써 A사격부대의 사격완료시간은 60초가 증가하게 된다. 이 때 할당된 2번 표적을 A사격부대가 사격하기 위해서는 사격준비시간 240초가 추가되어 A사격부대의 사격완료시간은 기존 6번 표적의 사격완료시간인 80초와 2번 표적을 사격하기 위한 준비시간 240초, 사격소요시간 60초를 합하여 A사격부대의 사격완료시간은 380초가 된다. Step 7에서 C사격부대의 최초 사격완료시간 420초에서 A사격부대에 1발을 할당시켜 감소된 60초를 감하여 360초가 된다. Step 8에서 최초 최대 사격완료시간 420초가 감소하여 360초가 개선해가 된다. 이후 Step 9에서 C사격부대의 2번 표적의 1발을 다시 다른 사격부대에 할당여부를 판단하기 위해서 Step 1로 이동한다. 이후 2번 표적에서는 더 이상 할당할 수 없어 Step 10에서 C사격부대의 2번 표적을 표적의 집합에서 제외한다. Step 11에서는 C사격부대가 사격하는 표적의 집합 내 3번 표적을 대상으로 Step 3에서 분할되어 할당되는 절차를 수행한 후 3번 표적을 대상으로 분할하여 할당 시 사격완료시간의 감소가 없으며 새로운 최대 사격완료시간의 A사격부대를 대상으로 Step 1부터 분할 후 할당하는 절차를 수행하나 해의 개선이 불가능하여 해의 조정

과 평가를 거쳐 교환(exchange)단계로 이동한다. 분할단계의 예제에 대한 연산과정은 <Appendix 2>와 같다.



[Figure 3] Example of how to divide

#### 4.2.2 Adjustment and Evaluation

해의 개선은 1개의 사격부대가 사격하는 표적과 2개 이상의 사격부대가 동시에 사격하는 표적을 대상으로 표적과 표적사이에서 발생한 여유시간으로 사격 순서를 조정하여 사격완료시간을 단축하는 방법으로 표적에 대한 사격부대의 변경 없이 사격부대 내 표적의 사격 순서를 변경한다. 이후 해의 조정 및 평가는 개선해의 각 단계별로 조정과 평가를 실시하며 해의 개선유무에 따라 개선해의 이전단계 또는 다음단계로 이동한다.

#### Adjustment and Evaluation Algorithm

- Step 1 : Select  $j^*$  with  $\max_{j \in J} MT_j$
- Step 2 : Set  $W'_j = W_j$
- Step 3 : Select  $i'$  with  $\min\{D_i \mid i \in W'_j\}$  (The tie breaks arbitrarily)
- Step 4 : Saving time =  $\max(ST_{j^*(k+1)} - CT_{j^*(k)} - SUT_{j^*}) \forall k$
- Step 5 : If Saving time > 0, insert  $i'$  to Saving time, else, go to Step 10
- Step 6 : Calculate  $MT_j \forall j$
- Step 7 : If  $MT_{j^*} \geq MT_j$ , move  $i'$  to Saving time, else, go to Step 10  $\forall j$

Step 8:  $Best\ MT = MT_{j^*}$ , best improvement = current solution

Step 9: Delete  $i'$  from  $W'_j$

Step 10: If  $W'_{j^*} = \emptyset$ , go to Step 1, else, stop

해의 개선과 평가에서 Step 1) 최대 사격완료시간 ( $\max MT_j$ )의 사격부대( $j^*$ )를 선정하고 Step 2) 선정된 사격부대( $j^*$ )내의 표적을 사격하는 표적의 집합 ( $W'_j$ )으로 구성한다. Step 3) 최대 사격완료시간의 사격부대( $j^*$ )가 사격하는 표적들 중 표적의 사격수요( $D_i$ )가 가장 작은 최소 표적( $i'$ )을 선정하며 동일한 사격수요 시 임의로 선택한다. Step 4)  $ST_{j(k+1)}$ 은 사격부대가 사격하는 표적 중  $k$ 번째로 사격을 시작하는 표적의 사격시작시간을 의미하며  $CT_{j(k)}$ 은 사격부대가 사격하는 표적 중  $k$ 번째로 사격을 종료하는 표적의 사격완료시간이다. 절약시간(Saving time)은 최종 사격완료시간의 사격부대가 사격하는 표적과 표적사이에서 사격준비시간을 제외한 사격 여유시간을 의미한다. 즉, 표적 중에서 1개의 사격부대가 사격하는 효과보다 2개 이상의 사격부대가 사격하여 최종 사격완료시간의 단축효과가 증대될 때 어느 사격부대는 다른 사격부대의 사격이 가능할 시작시간까지 대기하여 2개 이상의 사격부대가 동시에 사격함으로써 발생하며 이는 사격부대가 다음 표적을 사격하기 위해 준비하는 시간을 초과하여 대기하는 시간을 의미한다. Step 5) 여유시간이 발생한 위치에 최소 표적( $i'$ )을 삽입한 후 Step 6) 절약시간에 최소 표적을 삽입한 최대 사격완료시간을 산출하여 Step 7)이 때 최대 사격완료시간은 단축하나 2개 이상의 사격부대가 사격한 최소 표적이면 오히려 다른 사격부대의 사격완료시간의 증가로 전체 시간이 증가할 수 있다. 그러므로 모든 사격부대의 사격완료시간을 다시 산출하여 시간단축이 있을 때만 표적의 위치를 변경하여 이동시킬 수 있다. Step 8) 위와 같이 표적의 이동 후 최대 사격완료시간이 감소되면 현재의 최대 사격완료시간이 현재해( $Best\ MT$ )가 되며 현재해는 개선해가 된다. Step 9) 표적의 집합 내에서 위치를 변경하여 할당되거나 변경할 수

없는 표적을 제외시키며 Step 10) 다음의 최대 사격완료시간의 사격부대가 사격하는 표적들을 대상으로 해의 개선이 없을 때까지 반복한다.

해의 개선과 평가는 [Figure 4]에서 사격부대별 표적과 표적의 사격임무 간 빈 공간인 여유시간으로 이는 다음 표적을 사격하기 위한 준비시간과 다른 사격부대와 같은 표적을 사격하기 위해 대기하는 시간의 합이다. 절약시간은 표적과 표적 사이의 시간이 사격준비시간 이상일 경우 사격준비시간을 감한 대기시간으로 단축할 수 있는 시간이다. 사격부대는 A, B, C, D유형별 1개씩 총 4개를 대상으로 하고 표적은 총 5개이며 사격수요( $D_i$ )는  $D_1 = 8$ ,  $D_2 = 7$ ,  $D_3 = 8$ ,  $D_4 = 3$ ,  $D_5 = 3$ 이며 포탄 1발의 효과(Efficiency)와 사격소요시간(Processing Time), 사격준비시간(Setup Time)은 <Table 1>과 같다. [Figure 4]에서와 같이 Step 1에서 최대 사격완료시간의 C사격부대를 선정하고 Step 2에서는 C사격부대내의 2, 3, 5번 표적을 집합으로 구성한다. Step 3에서 C사격부대가 사격하는 표적 중 사격수요가 최소인 5번 표적을 선택하여 Step 4에서 표적과 표적사이의 시간은 2, 3번 표적 간 180초, 3, 5번 표적 간 60초 중 최대 시간인 180초를 선택한다. 이는 C사격부대의 1개의 표적을 사격 후 다른 표적을 사격하기 위해 필요한 준비시간은 60초이기 때문에 표적사이의 시간인 120초가 여유시간이 된다. Step 5에서 C사격부대의 사격완료시간은 420초에서 여유시간인 120초에서 5번 표적을 사격한 완료시간인 60초를 고려하여 5번 표적을 삽입한 후 Step 6에서 C사격부대의 최대완료시간은 120초의 절약시간으로 5번 표적의 사격완료시간 60초, 이후 3번 표적을 사격하기 위한 준비시간 60초를 고려하여 최초 420초에서 300초로 120초 감소가 가능하였으며 이로 인해 다른 사격부대의 사격완료시간의 증가는 없었다. Step 7에서 절약시간에 따른 표적의 이동으로 최대 사격완료시간이 감소될 수 있으므로 5번 표적을 이동시킨 후 Step 8에서 위와 같이 5번 표적의 이동 후 최대 사격완료시간이 감소되어 개선해가 된다. Step 9에서 5번 표적을 제외시키고 Step 10에

서 Step 1로 이동하여 최대 사격완료시간의 A사격 부대를 대상으로 절차를 반복하나 해의 개선이 없어 다음 단계인 교환(Exchange)으로 이동한다. 이와 같이 사격부대가 사격하는 표적과 표적의 사격 순서를 변경하여 C사격부대의 최대 사격완료시간은 420초에서 A사격부대의 최대 사격완료시간인 380초로 최종 40초를 단축할 수 있으며 시간 단축이 가능할 때까지 반복 적용한다. 해의 개선과 평가의 예제에 대한 세부 연산과정은 <Appendix 3>과 같다.

Troop \ Time	1	2	3	4	5	6	7
A	②					④	
B	①						
C	②				③		⑤
D	①				③		



Troop \ Time	1	2	3	4	5	6	7
A	②					④	
B	①						
C	②		⑤		③		
D	①				③		

[Figure 4] Example of how to adjust and evaluate

### 4.3 Exchange

최초 Greedy Rule에 의한 우선적으로 표적들이 사격부대에 할당된 이후 2개의 사격부대가 사격하는 1개의 표적을 사격함으로써 표적에 대한 사격 소요시간을 더 이상 단축하는데 제한된다. 따라서 개선해 생성방법의 2단계인 교환은 사격부대별로 사격할 표적들의 조합을 변화시켜 적절하게 효율적인 사격부대가 사격하는 것이 오히려 전체 사격완료시간을 단축하는 방법일 수 있다는 개념으로 표적에 대한 사격부대를 교환한다. 이때 변화된 사격완료시간을 산출하기 위해 사격부대별 준비시간과 소요시간을 고려해야 한다.

### Exchange Algorithm

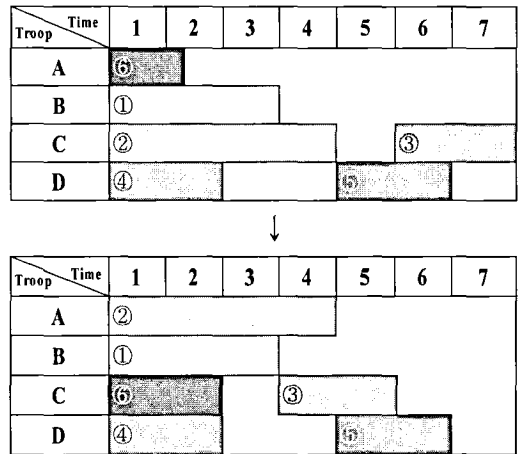
- Step 1 : Select  $j^*$  with  $\max_{j \in J} MT_j$  and  $j'$  with  $\min_{j \in J} MT_j$
- Step 2 : Set  $W_{j^*}' = W_{j^*}$  and  $W_{j'}' = W_{j'}$ ,  
 $BestMT = MT_{j^*}$
- Step 3 : Select  $i^*$  with  $\max\{D_i | i \in W_{j^*}'\}$  and  $i'$  with  $\min\{D_i | i \in W_{j'}'\}$
- Step 4 : Exchange  $i^*$  and  $i'$ , insert  $i^*$  to  $j'$  and  $i'$  to  $j^*$
- Step 5 : If  $MT_{j^*}' + (R_{i^*j^*} - R_{i^*j'}) \times PT_{j^*} \geq MT_{j^*}' + (R_{i^*j'} - R_{i^*j'}) \times PT_{j'}$ ,  $R_{ij} = \lceil D_i / E_j \rceil$ , allocate  $i^*$  to  $j'$  and  $i'$  to  $j^*$ , else, not allocate and go to Step 8
- Step 6 :  $BestMT = MT_{j^*}' + (R_{i^*j^*} - R_{i^*j'}) \times PT_{j^*}$
- Step 7 : If  $BestMT \leq MT_{j^*}$ , best improvement = current solution
- Step 8 : Delete  $i^*$  from  $W_{j^*}'$  and  $i'$  from  $W_{j'}'$
- Step 9 : If  $W_{j^*}' = \emptyset$ , go to Step 1, else, stop

개선해 생성의 2단계인 교환에서 Step 1) 최대 사격완료시간의 사격부대( $j^*$ )와 최소 사격완료시간의 사격부대( $j'$ )를 선정한다. Step 2) 사격부대( $j^*$ )가 사격하는 표적의 집합( $W_{j^*}'$ )과 사격부대( $j'$ )가 사격하는 표적의 집합( $W_{j'}'$ )를 구성하며 현재의 최대 사격완료시간이 현재해( $BestMT$ )가 된다. Step 3) 표적의 집합( $W_{j^*}'$ )에서 최대 표적( $i^*$ )과 표적의 집합( $W_{j'}'$ )에서 최소 표적( $i'$ )를 선택하여 Step 4) 사격부대( $j^*$ )가 사격하는 최대 표적( $i^*$ )과 사격부대( $j'$ )가 사격하는 최소 표적( $i'$ )를 교환한다. Step 5) 이때 최대 사격완료시간( $\max MT_j$ )에서 감해지는 최대 표적의 사격 발수( $R_{i^*j^*}$ )와 더해지는 최소 표적의 사격 발수( $R_{i^*j'}$ )의 차이만큼 시간이 감소하며 최소 사격완료시간은 감해지는 최소 표적 대신 최대 표적을 포함하여 증가된 시간의 합이 작거나 같을 때 최대 표적과 최소 표적을 상호 교환하게 되며 최대 사격완료시간의 감소가 없을 때에는 Step 8로 이동한다. 이 때 교환되는 표적은 사격수요( $D_i$ )를 만족하는 사격부대의 1발의 효과( $E_j$ )를 고려하여 새로운 사격 발수를 산출한다. Step 6) 최대 사격완료시간은 교환되는 포탄의 사격 발수의 차이만큼 사격

소요시간이 감소될 수 있으며 Step 7)최대 사격완료시간의 감소가 있으면 현재해는 개선해가 된다. Step 8)각각 표적의 집합 내에서 교환되어 할당되거나 교환할 수 없는 표적을 제외시키며 Step 9)다음의 표적들을 대상으로 반복한다.

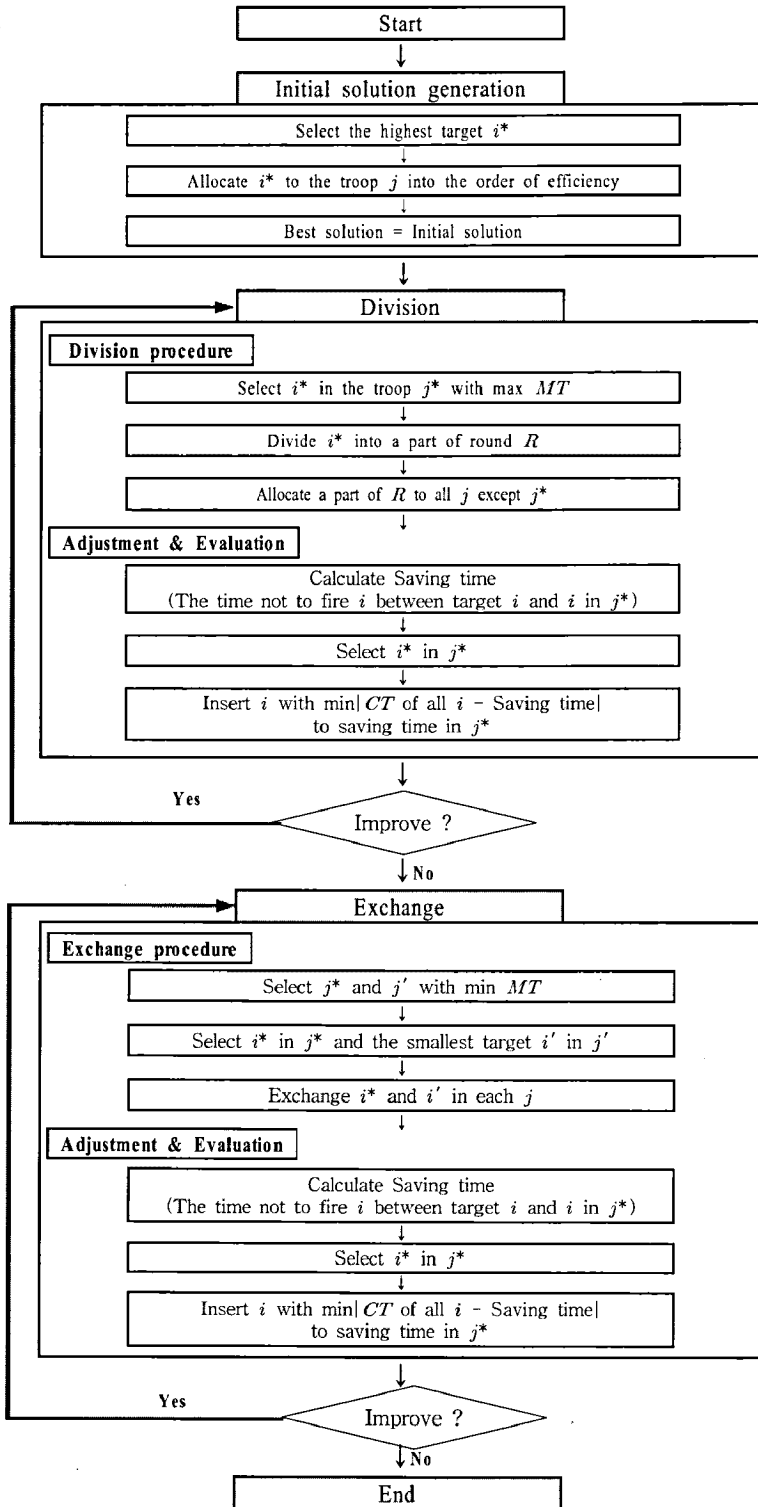
교환단계는 최초에 최대 사격완료시간의 사격부대가 사격할 표적 중 사격부대별 표적과 표적의 상호 교환으로써 시간을 단축하기 위한 목적이라 할 수 있다. 사격부대는 A, B, C, D유형별 1개씩 총 4개를 대상으로 하고 표적은 총 6개이며 사격수요( $D_i$ )는  $D_1 = 8, D_2 = 12, D_3 = 6, D_4 = 5, D_5 = 5, D_6 = 4$ 이며 포탄 1발의 효과(Efficiency)와 사격소요시간(Processing Time), 사격준비시간(Setup Time)은 <Table 1>과 같다. [Figure 5]에서와 같이 Step 1에서 최대 사격완료시간의 C사격부대와 최소 사격완료시간의 A사격부대를 선정하고 Step 2에서 A, C사격부대가 사격하는 표적의 집합을 구성하며 C사격부대의 최대 사격완료시간인 420초가 현재해이다. Step 3에서 C사격부대에서 최대 표적인 2번과 A사격부대에서 최소 표적인 6번 표적을 선택하며 Step 4에서 2번 표적과 6번 표적을 교환하여 할당한다. Step 5에서 교환 시에는 표적의 사격수요에 따라 해당 사격부대의 사격 발수로 산출하고 최초 표적의 사격수요 이상을 충족하여야 한다. 즉, C사격부대가 사격하는 표적의 1발을 A사격부대에 할당할 때 A사격부대는 3발을 사격해야 사격수요를 충족할 수 있으며 A사격부대의 1발은 C사격부대의 1/3발이나 사격부대의 최소 사격 발수를 고려하여 1발을 사격해야함을 고려해야한다. 이와 같이 A사격부대 6번 표적의 사격수요는 4발이며 C사격부대 2번 표적의 사격 수요는 12발이므로 상호간 표적을 교환 시 A사격부대의 사격완료시간은 80초에서 240초로 증가하지만 C사격부대의 사격완료시간은 420초에서 300초로 감소가 가능하므로 표적을 교환하여 할당한다. 이후 Step 6에서 C사격부대의 최대 사격완료시간이 300초로 산출되며 Step 7에서 최초의 최대 완료시간인 420초는 300초로 감소되므로 개선해가 된다. Step 8에서 각각의 표적의 집합에서

교환된 표적을 제외시키며 Step 9에서 다음의 표적들을 대상으로 반복하나 2번과 6번 표적을 교환 후 최소 사격완료시간의 B사격부대의 1번 표적과 최대 사격완료시간의 D사격부대의 4번 또는 5번 표적의 경우 사격수요의 충족과 사격준비시간의 고려에 따라 시간 단축이 제한되어 표적을 교환 할 수 없다. 해의 조정과 평가를 거쳐 개선이 있으면 교환단계를 반복하고 해의 개선이 없으면 종료한다. 해의 조정 및 평가는 분할단계에서 분할절차 이후 조정 및 평가절차에서 설명한 내용과 동일하다. 개선해의 교환단계의 예제에 대한 세부 연산과정은 <Appendix 4>와 같다.



[Figure 5] Example of how to exchange target

위와 같이 본 연구에서 제시하는 발견적 기법은 [Figure 6]에서와 같이 초기해의 생성과 해의 개선을 위한 분할과 교환단계로 구성되어 있으며 초기해를 개선하기 위한 분할과 교환, 조정 및 평가의 상호보완적인 개선절차를 거쳐 최적해에 유사한 값으로 발전시킬 수 있는 방법이다. 초기해에서 여러 경우의 이웃해를 생성하여 해의 가용여부를 선택하는 방법보다 개선해의 변화를 거쳐 시간단축의 효과에 따라 절차의 반복 및 이동을 결정함으로써 신속한 해의 산출과 최적해에 유사한 근사값을 산출할 수 있었으며 지역해(local optimal)로의 제한을 방지할 수 있었다.



[Figure 6] Procedure of Heuristic Search Algorithm

## 5. 실험결과

제시한 발견적 기법에서 동시 할당 및 순서결정(SAS) 알고리즘의 성능을 평가하기 위한 실험으로 포병 사격부대의 운용상황에 근거한 예제를 생성하였다. 1개의 포병대대급 사격부대는 18문의 화포로 구성되어 사격 발수 1발은 실제 18발을 사격하는 효과와 같으며 실험에서 포병대대급 사격부대의 수는 2개~30개로 군단급 제대 이상의 수준까지 구성하였다. 또한 표적의 수는 사격부대 대비 1배~9배 수준에서 결정하였으며 표적의 사격수요인 크기의 생성 방법은 I 유형인 균등분포(uniform distribution)에 의한 생성과 II 유형인 정규분포(normal distribution), III 유형인 무작위(random)에 의한 방법에 의해 생성하여 적용하였다. 최적해의 산출은 ILOG OPL Stu-

dio 5.1을 이용하여 CPLEX 10.1로 산출하였으며 발견적 기법은 Microsoft Visual C#으로 구현하여 PC(Core2 Duo 2.6 GHz)상에서 실험하였다. 최적해 산출에 사격부대 대비 표적의 수를 2배로 적용하여 최적해를 산출한 경우 산출시간은 최소 0.51초에서 최대 6시간 30여분이 소요되어 사격부대와 표적 수의 소폭 증가에도 활용 가능한 최적해 및 상한값 산출에 최적화 프로그램으로는 제한이 되었다. 따라서 최적해 산출이 불가능한 경우와 시간상 제한으로 인해 최적값 대비 근접한 상한값을 3600초로 시간제한(time limit) 기능을 설정하여 산출하였다. 또한 각 표적의 유형별 예제를 근거로 산출한 근사해와 최적해(또는 상한값)의 차이를 최적해(또는 상한값)로 나눈 오차율(Gap, %)을 산출하여 최적해와 근사해를 비교하였다.

〈Table 3〉 Result for the small examples

Type	Troop	Target	Solution			Gap (%)	Computation Time(sec)	
			Optimal	Heuristic			Optimal	Heuristic(Initial)
				Initial	Result			
I	2	2	580	580	580	0	1.25	0.12(0.04)
		4	1200	1260	1220	1.66	2.00	0.23(0.04)
		6	1700	1880	1700	0	186.95	0.42(0.04)
	3	3	580	580	580	0	1.25	0.15(0.04)
		5	900	1080	920	2.22	73.00	0.17(0.04)
		6	1080	1200	1080	0	**	2.46(0.04)
II	2	2	320	320	320	0	1.28	0.10(0.04)
		4	720	760	720	0	1.25	0.20(0.03)
		6	1110	1180	1110	0	1.54	0.40(0.04)
	3	3	320	320	320	0	1.03	0.14(0.03)
		5	600	660	620	3.33	4.53	0.28(0.04)
		6	660	740	660	0	15.10	0.40(0.04)
III	2	2	180	180	180	0	0.51	0.34(0.03)
		4	580	580	580	0	0.79	0.43(0.04)
		6	940	960	940	0	1.00	0.42(0.04)
	3	3	180	180	180	0	0.76	0.42(0.04)
		5	480	480	480	0	1.56	0.62(0.04)
		6	540	560	540	0	12.56	1.75(0.04)

주) \*Gap(%) = {Heuristic - Optimal(or Best)} / Optimal(or Best) × 100.

\*\* 3600초 시간제한 하에서 최적해가 산출되지 않아 the best solution으로 비교.

<Table 3>은 사격부대의 수 대비 표적의 수를 2배로 하여 최대 6개까지의 표적을 대상으로 산출된 결과로 대부분 최적해가 산출되었으며 근사해는 최적해와 평균 0.4%의 차이를 나타내며 최적해에 근사한 값을 산출하였다. 그러나 <Table 4>의 경우 표적을 사격부대의 수 대비 9배까지 증가시켜

최소 90개 표적의 경우에도 최적해가 산출이 되지 않아 직접적으로 비교할 수 없었으나 1시간이내에 산출된 상한값을 기준으로 근사해는 우수한 값을 보였다. <Table 3>과 <Table 4>에서 산출된 근사해는 최적해 대비 최대 5%의 오차가 발생하였고 I 유형에서 평균 1.39%, II유형에서 평균 0.37%의

<Table 4> Result for the large examples

Type	Troop	Target	Solution			Gap(%)	Computation Time(sec)
			The best	Heuristic			Heuristic(Initial)
				Initial	Result		
I	4	10	1200	1320	1260	5.00	6.40(0.04)
		20	2620	2680	2640	0.76	16.92(0.06)
		30	7620	4120	4020	-47.2	13.53(0.06)
	10	20	1080	980	980	-9.2	2.79(0.09)
		50	3320	2680	2640	-20.4	21.06(0.10)
		90	-	5040	4920	-	36.60(0.14)
	30	60	-	1320	1260	-	11.51(0.10)
		100	-	2180	2160	-	14.56(0.17)
		150	-	3300	3270	-	104.89(0.28)
II	4	10	840	840	840	0	0.92(0.04)
		20	2460	1900	1800	-26.8	11.32(0.06)
		30	3980	2880	2760	-30.6	13.39(0.06)
	10	20	720	660	660	-8.3	2.15(0.06)
		50	2840	1900	1890	-33.4	34.07(0.09)
		90	-	3540	3480	-	96.03(0.14)
	30	60	-	840	840	-	0.60(0.10)
		100	-	1500	1480	-	13.54(0.17)
		150	-	2400	2280	-	20.87(0.28)
III	4	10	720	720	720	0	0.76(0.04)
		20	1560	1590	1560	0	1.37(0.04)
		30	3600	2400	2340	-35.0	2.20(0.07)
	10	20	560	520	520	-7.1	1.51(0.06)
		50	2180	1640	1590	-27.0	4.26(0.09)
		90	-	3000	2900	-	10.62(0.14)
	30	60	-	810	800	-	6.59(0.10)
		100	-	1260	1260	-	12.40(0.18)
		150	-	2020	1980	-	23.82(0.28)

주)  $Gap(\%) = \{Heuristic - Optimal(or Best)\} / Optimal(or Best) \times 100$ .

The best solution은 3600초 시간제약 하에서 구한 해임, (-)는 3600초 시간제약 하에서 해를 구할 수 없었음.

차이가 발생했으나 III유형에서는 최적해와 근사해의 오차(Gap)가 발생하지 않았다. 각 유형별 오차의 평균은 0.59%로 최적해 대비 근소한 차이를 보였으며 설정된 시간제한 내에서 상한값 조차 산출되지 않는 경우를 제외하고 근사해만 산출이 가능하였다. 이를 고려한 해의 산출시간은 최소 1.5배~최대 442배의 차이로 근사해는 우수하였으며 최소 0.12초~최대 104.89초로 평균 10.97초 만에 신속한 근사해의 산출이 가능하였다. 본 실험결과에서 산출된 최적해를 고려할 때 제한한 동시 표적 할당 및 순서결정 알고리즘(SAS Algorithm)에 의해 산출된 근사해와 합리적 산출시간은 최적해 대비 우수한 성능을 나타냈다.

## 6. 결 론

표적 할당과 사격 순서의 동시 결정 문제는 사격부대별 사격준비시간과 사격소요시간, 포탄의 효과를 동시에 고려하고 2개 이상 다수의 사격부대가 동일 표적을 사격 시 사격시작시간이 동일해야 되는 사격완료시간의 최소화 문제로서 일반적인 사격 순서 결정에 관한 연구들과 다른 새로운 연구라 할 수 있다. 사전에 사격부대별 표적 할당이 완료되어 계획된 표적만으로 사격 순서를 고려하거나 사전 표적을 할당한 후 다음단계에서 사격 순서를 결정하는 문제와는 달리 사격이 요구되는 표적들을 최단시간 내 사격부대별 표적을 할당함과 동시에 사격부대의 표적별 사격 순서를 결정하는 문제이다. 일반 제조 및 서비스산업에서 전자제품회사가 생산한 제품을 예로 가정하면 AS(after service) 센터가 담당하고 있는 지역 내에서 제품별 차별된 기술 서비스를 들 수 있다. 요청된 AS는 최단시간에 해당 서비스를 충족시킬 목적으로 제품별 요구되는 기술 수준에 의한 능력별 서비스 인원의 할당으로써 신속한 최대의 기술 서비스를 구현하는 문제에 적용이 가능하다. 이는 현재 군이 강조하고 있는 효과 중심적 집중사격에 보다 효과적이며 다양한 무기체계에 대한 협동된 통합 화력운용의 효율성이 증대

됨과 함께 사격 후 사격부대의 위치를 추적하는 탐지장비의 발달로 위협받는 사격부대의 생존성을 추구하기 위해서도 자동화 사격체계의 마련이 절실하다. 본 연구는 사격이 요구되는 대량의 표적이 생성된 상황에서 표적을 사격부대에 할당함과 동시에 사격 순서의 결정을 위한 연구로서 제시한 휴리스틱 알고리즘은 최적해 대비 우수한 근사해와 신속한 산출시간을 보여주었으며 인력에 의한 수동적인 사격계획의 수립방법을 개선하여 화력운용의 전술적 분야와 포병사격계획 뿐만 아니라 다양한 사격수단의 전술적 통합 사격지휘와 사격계획절차의 자동화 체계구축에 본 알고리즘이 효과적으로 활용이 가능하다. 본 연구에서 제시한 휴리스틱 알고리즘에 의해 산출된 근사해를 제한된 표적과 사격부대의 수에서만 산출된 최적해로만 비교할 수 있는 제한사항이 있었다. 향후 연구과제는 본 문제의 확장된 형태로서 표적에 대한 사격의 우선권과 사격진지의 변환 등 작전 간 발생 가능한 각종 상황을 반영하고 표적의 생성과 소멸에 따른 기존 사격계획을 실시간 변경하여 적용할 수 있는 표적의 분배와 할당, 사격 순서에 대해 접근을 통해 미래 지향적인 연구과제가 될 것으로 기대된다.

## 참 고 문 헌

- [1] 육군본부, 「포병운용」, FM 2-2, 2002.
- [2] 육군본부, 「화력운용」, FM 3-10, 2008.
- [3] 육군본부, 「화력실무운용」, RM 32-0-47, 2004.
- [4] 김기호, “포병부대 계획표적 사격 순서 결정에 관한 연구”, 고려대학교 논문, 2002.
- [5] 김태현, 이영훈, “공유표적을 포함한 사격 순서 결정에 관한 연구”, 「한국경영과학회지」, 제28호, 제3호(2003), pp.123-134.
- [6] 황원식, 이재영, “스케줄링을 이용한 계획표적 사격 순서의 최적화에 관한 연구”, 「한국국방경영분석학회지」, 제33권, 제1호(2007), pp.105-115.



- [7] Fowler, J.W., "Scheduling of unrelated parallel machines an application to PWB manufacturing," *IIE Transactions*, Vol.34(2002), pp. 921-931.
- [8] Hariri, A.M.A. and C.N. Potts, "Heuristic for scheduling unrelated parallel machines," *Computers and Operations Research*, Vol.18, No.3 (1991), pp.323-331.
- [9] Kwon, O.J., D.H. Kang, K.S. Lee, and S.S. Park, "Targeting and Scheduling Problem for Field Artillery," *Computers and Industrial Engineering*, Vol.33(1997), pp.693-696.
- [10] Martello, S., F. Soumis, and P. Toth, "Exact and approximation algorithms for makespan minimization on unrelated parallel machines," *Discrete Applied Mathematics*, Vol.75(1977), pp.169-188.
- [11] Plekar, U.S., N. Rama, and K. Toaffe, "Duality based relaxations for makespan minimization for unrelated parallel machines," *TIME/ORSA Bullutin*, Vol.31, No.21(1991).
- [12] Sundararaghavan, P. and M. Ahmed, "Single machine scheduling with start time dependent processing times," *European Journal of Operational Research*, Vol.78, No.3(1994), pp. 394-403.

## (Appendix)

### 1. Example of Initial solution generation

Step 1 :  $I' = \{1, 2, 3, 4, 5, 6, 7\}$

Step 2 : Select the highest target  $i^* = 1$

Step 3 :  $R_{1A} = \lceil D_1/E_A \rceil = \lceil 22/1 \rceil = 22$ ,  $R_{1B} = \lceil D_1/E_B \rceil = \lceil 22/2 \rceil = 11$ ,

$R_{1C} = \lceil D_1/E_C \rceil = \lceil 22/3 \rceil = 8$ ,  $R_{1D} = \lceil D_1/E_D \rceil = \lceil 22/5 \rceil = 5$ ,

$CT_{1A} = RT_A + R_{1A} \times PT_A + SUT_A = 0 + 22 \times 20 + 0 = 440$ ,

$CT_{1B} = RT_B + R_{1B} \times PT_B + SUT_B = 0 + 11 \times 30 + 0 = 330$ ,

$CT_{1C} = RT_C + R_{1C} \times PT_C + SUT_C = 0 + 8 \times 60 + 0 = 480$ ,

$CT_{1D} = RT_D + R_{1D} \times PT_D + SUT_D = 0 + 5 \times 120 + 0 = 600$

(If  $i^*$  is not allocated to  $j$ ,  $MT_j = SUT_j = 0$ )

Step 4 : Select  $j' = B$  with the  $\min CT_{1B} = 330$

Step 5 :  $W_B = \{1\}$

Step 6 :  $MT_B = CT_{1B} = 330$

Step 7 : Delete 1 from  $I'$ ,  $I' = \{2, 3, 4, 5, 6, 7\}$

Step 8 :  $I' \neq \emptyset$ , go to Step 2

Step 2 : Select the highest target  $i^* = 2$

Step 3 :  $R_{2A} = \lceil D_2/E_A \rceil = \lceil 15/1 \rceil = 15$ ,  $R_{2B} = \lceil D_2/E_B \rceil = \lceil 15/2 \rceil = 8$ ,

$R_{2C} = \lceil D_2/E_C \rceil = \lceil 15/3 \rceil = 5$ ,  $R_{2D} = \lceil D_2/E_D \rceil = \lceil 15/5 \rceil = 3$ ,

$CT_{2A} = MT_A + R_{2A} \times PT_A + SUT_A = 0 + 15 \times 20 + 0 = 300$ ,

$CT_{2B} = MT_B + R_{2B} \times PT_B + SUT_B = 330 + 8 \times 30 + 300 = 870$ ,

$CT_{2C} = MT_C + R_{2C} \times PT_C + SUT_C = 0 + 5 \times 60 + 0 = 300$ ,

$CT_{2D} = RT_D + R_{2D} \times PT_D + SUT_D = 0 + 3 \times 120 + 0 = 360$

Step 4 : Select  $j' = C$  with the  $\min CT_{2C} = 300$

Step 5 :  $W_C = \{2\}$ ,

Step 6 :  $MT_C = CT_{2C} = 300$

Step 7 : Delete 2 from  $I'$ ,  $I' = \{3, 4, 5, 6, 7\}$

Step 8 :  $I' \neq \emptyset$ , go to Step 2

### 2. Example of Division Algorithm

Step 1 :  $\max MT_C = 420$ ,  $j^* = C$

Step 2 :  $W_C' = \{2, 3\}$ ,  $BestMT = MT_C = 420$

Step 3 : Select  $i^* = 2$

Step 4 : Divide  $D_1$  into  $\max\{0, D_1 - E_C\}$  and  $\lceil E_C/E_A \rceil = R_{1A}$ ,  $\max\{0, \lceil (12-3)/3 \rceil = 3\}$  and  $\lceil 3/1 \rceil = 3$ ,

Divide  $D_1$  into  $\max\{0, D_1 - E_C\}$  and  $\lceil E_C/E_B \rceil = R_{1B}$ ,  $\max\{0, \lceil (12-3)/3 \rceil = 3\}$  and  $\lceil 3/2 \rceil = 2$ ,

Divide  $D_1$  into  $\max\{0, D_1 - E_C\}$  and  $\lceil E_C/E_D \rceil = R_{1D}$ ,  $\max\{0, \lceil (12-3)/3 \rceil = 3\}$  and  $\lceil 3/5 \rceil = 1$ ,

$D_2 = 12$ ,  $E_A = 1$ ,  $E_B = 2$ ,  $E_3 = 3$ ,  $E_4 = 5$

Step 5 : If  $3 > 0$ , insert 9 to C and 3 to A,

- If  $3 > 0$ , insert 9 to C and 2 to B,  
 If  $3 > 0$ , insert 9 to C and 1 to D  
 Step 6 :  $MT_C \geq MT_A + SUT_A + R_{1A} \times PT_A$ ,  $420 \geq 80 + 240 + 3 \times 20 = 380$ , go to Step 10,  
 $MT_C \geq MT_B + SUT_B + R_{1B} \times PT_B$ ,  $420 \geq 180 + 300 + 2 \times 60 = 600$ , not allocate,  
 $MT_C \geq MT_D + SUT_D + R_{1D} \times PT_D$ ,  $420 \geq 360 + 120 + 1 \times 120 = 600$ , not allocate  
 Step 7 :  $BestMT = MT_C - PT_C = 420 - 60 = 360$   
 Step 8 :  $BestMT = 360 \leq 420$ , best improvement = current solution  
 Step 9 :  $D_{j^*} = D_{j^*} - E_{j^*}$  and go to Step 1,  $D_2 = D_2 - E_C = 12 - 3 = 9$   
 Step 10 : Delete  $i^* = 2$  from  $W'_C$   
 Step 11 :  $W'_C = \{3\}$ ,  $W'_C \neq \emptyset$ , go to Step 3

### 3. Example of Adjustment and Evaluation Algorithm

- Step 1 :  $MT_C = 420 = 420$ ,  $MT_A = 380$ ,  $MT_B = 180$ ,  $MT_D = 300$ ,  $j^* = C$   
 Step 2 :  $W'_C = \{2, 3, 5\}$ ,  $BestMT = MT_C$   
 Step 3 : Select  $i' = 5$   
 Step 4 : Saving time =  $\max(ST_{j^*(k+1)} - CT_{j^*(k)} - SUT_{j^*})$ ,  
 $ST_{C(2)} - CT_{C(1)} - SUT_C = 240 - 60 - 60 = 120$ ,  
 $ST_{C(3)} - CT_{C(2)} - SUT_C = 360 - 300 - 60 = 0$ , Saving time = 120  
 Step 5 : Saving time =  $120 \geq 0$ , insert  $i' = 5$   
 Step 6 :  $MT_A = 380$ ,  $MT_B = 180$ ,  $MT_C = 300$ ,  $MT_D = 300$ ,  $j^* = A$   
 Step 7 :  $MT_{j^*=C} \geq MT_A = 420 \geq 380$ ,  $MT_{j^*=C} \geq MT_B = 420 \geq 180$ ,  
 $MT_{j^*=C} \geq MT_C = 420 \geq 300$ ,  $MT_{j^*=C} \geq MT_D = 420 \geq 300$ , allocate  $i' = 5$  to Saving time  
 Step 8 :  $BestMT = MT_{j^*} = MT_A = 380$ , best improvement = current solution  
 Step 9 : Delete  $i' = 5$  from  $W'_C = \{2, 3, 5\}$ ,  $W'_C = \{2, 3\}$   
 Step 10 :  $W'_{j^*} \neq \emptyset$ , go to Step 1

### 4. Example of Exchange Algorithm

- Step 1 :  $\max MT_C = 420$ ,  $\min MT_A = 80$ ,  $j^* = C$ ,  $j' = A$   
 Step 2 :  $W'_A = \{6\}$ ,  $W'_C = \{2, 3\}$ ,  $BestMT = MT_C$   
 Step 3 : Select  $i^* = 2$ ,  $i' = 6$ ,  $D_2 = 12$ ,  $D_3 = 6$ ,  $D_6 = 4$ ,  
 Step 4 : Exchange  $i^* = 12$  and  $i' = 4$ , insert  $i^* = 12$  to A and  $i' = 4$  to C  
 Step 5 :  $R_{2A} = \lceil D_2/E_A \rceil = \lceil 12/1 \rceil = 12$ ,  $R_{2C} = \lceil D_2/E_C \rceil = \lceil 12/3 \rceil = 4$ ,  
 $R_{6A} = \lceil D_6/E_A \rceil = \lceil 4/1 \rceil = 4$ ,  $R_{6C} = \lceil D_6/E_C \rceil = \lceil 4/3 \rceil = 2$ ,  
 $MT_C + (R_{6C} - R_{2C}) \times PT_C \geq MT_A + (R_{2A} - R_{6A}) \times PT_A$ ,  
 $420 + (2 - 4) \times 60 = 300 \geq 80 + (12 - 4) \times 20 = 240$ , move 2 to A, 6 to C  
 Step 6 :  $BestMT = MT_C + (R_{6C} - R_{2C}) \times PT_C = 300$   
 Step 7 : If  $BestMT = 300 \leq 420$ , best improvement = current solution  
 Step 8 : Delete  $i^* = 2$  from  $W'_C = \{2, 3\}$  and  $i' = 6$  from  $W'_A = \{6\}$ ,  $W'_B = \emptyset$ ,  $W'_C = \{3\}$   
 Step 9 :  $W'_{j^*} \neq \emptyset$ , go to Step 1