□ 硏 究 論 文 □

韓國國防經營分析學會誌
第36卷 第3號, 2010. 12. 31.

무장 할당문제에서 휴리스틱 방법 효율성 비교: 이행성 규칙이 성립하는 무장성능차이를 중심으로*

(Performance Comparison of Heuristics for Weapon-Target Assignment Problem with Transitivity Rules in Weapon's Kill Probability)

† 임 동 순(Yim, Dong-Soon)**, 최 봉 완(Choi, Bong Wan)***

ABSTRACT

운용과학의 군사 응용 분야 중 하나인 무장-표적 할당문제는 NP-complete 문제로 알려져 있어 주어진 시간 내에 최적해를 구할 수 없으므로 휴리스틱 방법에 의해 빠른 시간 내에 우수한 해를 구하는 것이 더 의미가 있다. 본 연구에서는 보다 효율적으로 해를 도출할 수 있는 방법을 개발하기 위해 전형적인 문제를 재구성하여 단순화 시켰다. 이러한 문제 하에서 두가지 유전자 알고리즘인 표적번호 표현 방법과 순열 표현방법을 비교하였고, 구성적 휴리스틱, 향상적 휴리스틱들을 개발하여 비교하였다. 무장의 파괴확률 간에 이행성 규칙이 존재하는 경우를 대상으로 실험을 수행한 결과 구성적 휴리스틱의 해를 초기해로 하여 교환에 기초한 향상적 휴리스틱을 적용한 결과가 계산 시간이나 해의 질 측면에서 가장 우수한 해를 생성하였다. 그러나, 구성적 휴리스틱의 효율성은 무장 성능 간 이행성 규칙에 민감한 결과를 나타내었다.

ABSTRACT

In this study, the weapon-target assignment problem arising in military application of operations research is considered. We reformulated the problem in order to simplify the solution methods based on genetic algorithms and heuristics. Since the problem is well known as NP-complete and cannot be solved in polynomial time, such solution methods have been widely used to obtain good solutions. Two chromosome representations--target number representation and permutation representation--in genetic algorithm are compared. In addition, a construction heuristic and three improving heuristics are developed. Several experiments under the condition of transitivity rules in weapon's kill probability have been accomplished. It shows that the construction heuristic and exchange-based improving heuristic guarantees good solutions within a second and the performance of construction heuristic is sensitive to transitivity rules.

Keywords : 무장 할당 문제(**Weapon Assignment Problem**), 휴리스틱(**Heuristics**), 효율성(**Performance**)

* 이 논문은 2010년도 한남대학교 학술연구조성비 지원에 의하여 연구되었음.

논문접수일 : 2010년 9월 9일 심사(수정)일 : 2010년 11월 12일 논문게재확정일 : 2010년 11월 29일

** 한남대학교 산업경영공학과

*** 한남대학교 국방전략대학원/무기체계 • M&S 연구센터

† 교신저자

1. 서 론

무장 할당(Weapon-Target Assignment)이란 적의 표적에 가해지는 손상을 최대화하기 위하여 보유하고 있는 무장을 표적에 할당하는 의사결정을 의미한다. 1950년대부터 연구되기 시작한 무장 할당 문제는 주어진 시간 내에 최적 해를 구하기가 매우 어려운 NP-complete 문제에 속한다[7]. 그러나 현실적인 문제에서는 빠른 의사결정을 위해 짧은 시간 내에 해를 구하여야 하는 제약이 존재한다. 따라서 무장 할당에 대한 최근의 연구들은 최적해 보다는 근사 최적해를 구하기 위한 효율적인 경험적 방법들을 제안하고 있다.

Ahuja 등[1]은 branch and bound 방법을 이용하여 최적해를 구하려는 시도를 하였으나 그들의 실험에 따르면 어떤 문제에서는 48시간 내에서도 최적해를 구하지 못하였다고 보고하였다. 이에 따라 보다 효율적인 휴리스틱 방법들을 제안 하였는데 이에는 최소비용 흐름문제에 기초한 구성적 휴리스틱(construction heuristic)과 multiexchange neighborhood structure 에 기초한 향상적 휴리스틱(improving heuristic)인 VLSN(Very Large-scale Neighborhood) 탐색이 포함 되었다.

유전자 알고리즘과 시뮬레이티드 애닐링, 개미군집 알고리즘 등과 같은 메타 휴리스틱(meta heuristic)도 우수한 해를 구하기 위한 방법으로 이용 되었다. 일반적으로 메타 휴리스틱은 통상적인 휴리스틱에 비해 다양한 해를 생성할 수 있으나, 보다 많은 계산시간을 필요로 하는 단점을 갖는다. Lee 등[6]의 연구에서는 유전자알고리즘과 부분 최적화(local optimization)방법을 함께 적용하여 우수한 해를 생성하려는 시도를 하였다.

본 연구에서는 무장 할당문제를 휴리스틱 방법으로 접근하며, 전형적인 무장할당 문제를 단순하게 변형하고 유전자 알고리즘, 구성적 휴리스틱, 그리고, 향상적 휴리스틱 방법의 효율성을 비교 평가한다. 효율성의 평가기준으로는 해의 질과 계산시간을 고려하였다. 특히, 무장간의 성능 순위가 모든 표적에게 동일하게 적용되는 이행성 규칙(transitivity rule)이 성립하는 상황으로 가정하여 실험을 수행한 결과를 제시한다.

2. 단순화된 무장 할당문제

무장의 종류가 m이고, 위협이 되는 표적의 수가 n이라고 하자. j번째 표적은 가치를 나타내는 값 V_j가 주어졌고, 목표에 할당할 수 있는 i번째 무장의 수를 W_i 라고 하자. i번째 무장 하나가 발사되어 j번째 표적을 파괴할 확률을 p_{ij}로 정의한다. $q_{ij}=1-p_{ij}$ 는 j번째 표적이 i번째 무장 하나의 발사에 대해 생존할 확률이다. j번째 표적에 할당된 i번째 무장의 수를 x_{ij}라고 할 때 이 무장 발사에 대한 표적의 생존 확률은 $q_{ij}^{x_{ij}}$ 가 된다. 한 표적에 대해 여러 종류의 무장을 동시에 할당 가능하다고 가정하자. 무장 할당 문제는 표적들의 잔존 가치를 최소화하기 위해 각 무장 종류에 대해서 각 표적에 할당되는 무장 수인 x_{ij}를 구하는 것이다. 이 문제는 다음과 같이 비선형 문제(Nonlinear programming problem)로 정의된다.

$$Min \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} q_{ij}^{x_{ij}} \right)$$

Subject to)

$$\sum_{j=1}^{n} x_{ij} \le W_i, i=1,\cdots,m$$

$$x_{ij} \ge 0 \text{인 정수}, i=1,2,\cdots,m, j=1,2,\cdots,n$$

전형적인 무장 할당 문제에 부가하여 표적에 할당할 수 있는 무장의 수에 대한 제약, 표적에 할당 가능한 총무장의 수에 대한 제약, 표적 생존률의 하한 등 실제적인 제한 조건들을 포함 시킬 수 있다[1].

위에서 정의된 무장 할당문제에서의 결정변수 x_{ij}는 정수의 값을 갖는다. 본 연구에서는 결정변

수 x_{ij}가 0 또는 1의 값을 갖도록 문제를 변형하였다. 특정의 무장 종류에 대해 W개의 무장이 있다면 각각의 단일 무장에 대해 j번째 표적에 대한 할당 여부를 x_{ij}로 나타내도록 한다. 즉, x_{ij}가 1이면 i번째 단일 무장이 j번째 표적에 할당 되었다는 것이고, 0이면 할당되지 않았다는 의미이다. 따라서 결정변수의 수는 무장의 종류가 m이고, 표적의 수가 n일 때 원래의 문제에서 nm이었지만, 이 같은 변형에서는 $n\sum_{i=1}^{m} W_i$으로 증가한다. 원래의 문제에서는 m이 무장 종류의 수를 의미했지만, 이 같은 변형에 의해 다음과 같이 정의되는 무장 할당 문제에서는 m이 전체 무장의 수를 의미하도록 한다.

$$Min \sum_{j=1}^{n} V_j \left(\prod_{i=1}^{m} q_{ij}^{\,x_{ij}} \right)$$

Subject to)

$$\sum_{j=1}^{n} x_{ij} \le 1, i = 1, \cdots, m$$

$$x_{ij} = 0 \text{또는} 1, i = 1, 2, \cdots, m, j = 1, 2, \cdots, n$$

위 문제의 최적해 에서 $\sum_{j=1}^{n} x_{ij} = 1$이 됨은 자명하다. 목적함수의 특성에 의해 표적에 할당되지 않은 무장은 없고, 각 무장은 필히 하나의 표적에 할당 된다. 위 식에서는 결정변수의 수가 증가되지만, 결정변수에 대한 제약이 단순화된다. 이러한 단순함은 다음 장에서 설명될 유전자 알고리즘에서 의미 있는 해의 표현방법이 가능하고, 경험적 방법에서 보다 효율적인 계산이 가능토록 한다.

정의된 문제의 해 공간은 m차원으로 가능한 해의 수는 m^n개이다. 만약, 10개의 무장을 10개의 표적에 할당한다면 10^{10}개의 대안이 있게 된다. 즉, 무장의 수가 증가할수록, 그리고, 표적의 수가 증가할수록 해의 대안 수는 기하급수적으로 증가하여 문제의 정확한 해 보다는 제한된 시간 내에 근사 해를 제공 하는 것이 더욱 의미가 있다. 따라서 정확한 해를 구하기 위한 최적화 방법 보다는 근사 해를 구하기 위한 경험적 방법이 많이 사용되고 있다.

3. 단순화된 무장 할당문제에서의 유전자 알고리즘

유전자 알고리즘은 여러 개의 개체가 동시에 병렬적으로 주어진 환경에 따라 적자생존의 방법으로 진화하여, 궁극적으로 최적의 상태에 도달하는 생태계의 진화이론에서 도입되었다. 이 알고리즘은 여러 개의 개체로 구성된 군집이 진화할 때 구 세대가 얻은 환경에 대한 정보는 염색체에 저장되어 다음 세대로 전달된다. 이 때 조상의 염색체가 그대로 복제되어 자손에게 전달되는 것이 아니라 조상의 염색체에 교배(crossover), 돌연변이(mutation), 전위(inversion) 등의 연산을 가하여 얻은 염색체로 자손에게 전달된다. 구 세대 중에서 한 개체가 선택되어 자손에게 유전정보를 남길 확률은 일반적으로 그 개체가 주위환경, 그리고 나머지 개체와 어떻게 상호 작용하는 가에 의존하는 적응 값(fitness)에 따라 변한다. 일반적으로 개체 값이 좋을수록 자손을 남길 확률이 높아지는 적자생존의 법칙이 적용된다. 이 알고리즘은 여러 가지 종류의 최적화 문제에 응용되어 좋은 결과를 낳고 있다.

유전자 알고리듬을 무장 할당 문제에 적용하기 위하여는 이 문제의 해결에 적합한 염색체의 표현방법(chromosome representation)과 그 표현에 알맞은 교배(crossover), 돌연변이(mutation) 연산자 등을 정의하여야 한다. 다음과 같은 두 가지 염색체 표현방법을 고려할 수 있다.

표적번호 표현 방법

정의된 무장 할당 문제에서는 각 무장이 필히 하나의 표적에 할당되어야 한다는 점을 고려하여 각 무장에 대해 할당된 표적 번호를 표현해 해를

직접적으로 나타내는 방법이다. 즉, m개의 무장이 있을 경우 이를 n개의 표적에 할당하는 해의 표현방법은 $(i_1,i_2,\cdots,i_m)$이다. 여기서 j번째 정수 i는 j번째 무장이 할당된 표적 번호로 1부터 n 까지의 값을 갖는다.

순열 표현방법

순열 표현방법에서의 해는 m개의 서로 다른 정수로 구성된 리스트로 표현된다[5]. 즉, 무장 할당의 경우 m개의 무장을 n개의 표적에 할당하는 표현방법 $(i_1,i_2,\cdots,i_m)$은 표적번호 표현방법과 다르게 해를 직접적으로 표현하지 않는다. 순열 표현을 하나의 유효한 해로 복호화(decoding)하기 위하여 다음과 같은 알고리즘을 고안하였다.

순열 표현방법에서의 복호화 알고리즘

1) 각 표적의 잔존가치 z_j를 원래의 가치 V_i로 초기화한다.
2) i_1부터 시작하여 i_m까지 차례대로 다음을 수행한다.
   2-1) i_k번째 무장에 대해 다음 조건을 만족하는 j^*번째 표적을 할당한다.

$$\max_j \{z_j \cdot p_{ij}\}$$

   2-2) 할당된 표적의 잔존가치 z_{j^*}를 $z_{j^*}(1-p_{ij^*})$로 변경한다.

복호화 알고리즘이 적용되는 순열표현의 염색체 $(i_1,i_2,\cdots,i_m)$에서 i_1번째 무장은 할당을 위한 가장 높은 우선순위를 가지고, 마지막 i_m번째 무장이 가장 낮은 우선순위를 가진다. 따라서 한 염색체는 할당을 위한 무장의 우선순위에 대한 정보를 포함하고 있게 된다.

유전자 알고리즘은 각 세대에서의 초기 염색체들에 대하여 적응값에 기초한 선택 확률을 구하고 룰렛 휠(roulette wheel) 방식에 의해 염색체들을 선택한다. 선택된 염색체들은 교배와 돌연변이 연산을 통하여 다음 세대의 염색체들을 생성한다. 각 염색체의 적응값은 본 연구에서 정의된 문제의 목적함수 값에 기초한다.

표적번호 표현방법을 이용하여 무장 할당 문제를 해결한 대표적인 연구로는 Lee등[6]이 있다. 그들은 단순화된 무장 할당 문제와 유사한 문제에 대해 유전자 알고리즘을 적용하였다. 교배연산자로는 EX라는 특별한 방법을 고안하였으며, 돌연변이 연산자로는 임의의 위치에 있는 표적번호를 임의의 다른 표적번호로 바꾸는 방법을 사용하였다.

순열 표현 방법에 기초하여 무장 할당 문제에 적용한 연구는 아직 없는 것으로 사료된다. 순열 표현방법에서는 외판원 문제에 적용되는 패스(path) 표현방법에서의 교배 연산자인 PMX (partially-mapped)[2], OX (order)[3], CX (cycle)[10]등을 사용할 수 있다. 본 연구에서는 교배연산자로 PMX를 사용하였다. 또한, 전형적인 돌연변이 연산자 대신에 염색체에서 임의의 두 위치를 구해 그 위치의 값들을 서로 바꾸는 전위 연산자를 사용하였다.

4. 단순화된 무장 할당문제에서의 휴리스틱 알고리즘

휴리스틱 방법들은 구성적 휴리스틱과 향상적 휴리스틱으로 나뉜다. 2 장에서 설명된 무장 할당 문제에 적용될 수 있는 구성적 휴리스틱으로 본 연구에서 제안한 방법은 다음과 같다.

구성적 휴리스틱(H1)

1) 각 표적의 잔존가치 z_1를 원래의 가치 V_i로 초기화한다. 미할당 무장 집합에 모든 무장 번호를 넣는다. 즉, $S=\{1,2,\cdots,m\}$

2) 미할당 무장집합 S가 공집합이 될 때까지 다음을 수행한다.

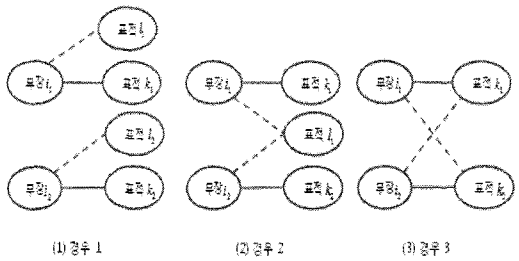
2-1) S의 각 무장($i \in S$)에 대해 각 표적(j)에 할당됐을 때 표적의 잔존가치 변화량 Δz_{ij}을 구한다.

2-2) 음수의 Δz_{ij} 중 가장 작은 값을 갖는 $\Delta z_{ab} = \min\{\Delta Z_{ij}, i \in S, j = 1,2,\cdots,n\}$을 선택하여 a번째 무장을 b번째 표적에 할당한다.

$Z_b = Z_b \cdot (1 - p_{ab}), S = S - \{a\}$

이 구성적 휴리스틱은 특성 상 greedy 방법에 속할 것이라고 생각될 수 있다. 일반적으로 greedy 방법은 최적해와 차이를 보여 좋은 성능을 나타내지 않는다고 알려져 있다. 그 이유는 휴리스틱의 초기 단계에서 근시안적인 대안만을 고려하고, 이러한 한정적인 대안으로부터의 의사결정이 전체적인 측면에서 보면 매우 안 좋은 결정이 될 수 있기 때문이다. 그러나 위 휴리스틱 방법은 이와 다르게 초기 단계에서 전체적인 대안을 고려한다고 볼 수 있다. 단계 2-1)에 따르면 알고리즘의 초기에 한정된 무장과 한정된 표적을 대상으로 하기보다 모든 무장을 대상으로 모든 표적에 대한 할당 가치를 구하고, 이를 바탕으로 가장 좋은 가치를 가지는 할당을 수행한다. 때문에, 일반적인 greedy 방법의 특성과 다르게 우수한 해를 생성할 수 있는 장점을 가지고 있다. 시간 복잡성은 $mn + (m-1)n + \cdots + n$번의 Δz_{ij}를 구하므로 $O(nm^2)$이다.

유전자 알고리듬의 해 또는 구성적 휴리스틱의 해를 보다 좋은 해로 변환하기 위한 부분 최적화 방법으로 향상적 휴리스틱을 적용할 수 있다. 본 연구에서 고려한 방법은 재위치(relocation)와 교환(exchange)이다. 이들 방법은 Harhalakis 등[4]이 시뮬레이티드 아닐링 방법에서 주변해를 생성하기 위한 방법으로 사용한 것에 기초한다. 이들 방법들의 기본 개념을 바탕으로 본 연구에서 정의



〈그림 1〉 2-재위치의 3가지 경우

된 무장 할당 문제에 적합하도록 고안하였다. 고안된 재위치는 두가지로 1-재위치와 2-재위치이다. 1-재위치는 한 무장을 선택하여 현재 할당된 표적에서 다른 표적으로의 할당 변경을 모색한다. 2-재위치는 서로 다른 표적에 할당된 두 무장을 선택하여 다른 표적으로의 할당 변경을 모색한다. 교환은 서로 다른 표적에 할당된 두 무장을 선택하여 서로 상대 표적으로의 할당 변경을 모색한다. 따라서 2-재위치는 교환을 포함하여 <그림 1>과 같이 3가지 경우를 고려한다. 첫번째 경우는 두 무장 i_1, i_2의 현재 할당된 표적 k_1, k_2를 제외한 나머지 $n-2$ 표적 중 서로 다른 2개의 표적 l_1, l_2에 새로이 할당한다. 두번째 경우는 현재 할당된 두 표적을 제외한 $n-2$ 표적 중 한 표적에 두 무장이 새로이 할당된다. 세번째 경우는 교환으로 서로 상대의 표적으로 할당을 변경한다.

결국 한 무장이 현재의 표적에서 다른 표적으로 할당을 변경했을 경우, 또한, 서로 다른 두 무장의 현재 표적을 다른 표적으로 변경하는 경우, 평가함수 값의 변화량을 계산하고, 변화량에 따라 무장의 재할당 여부를 결정한다. 현재의 해에서 i번째 무장이 k번째 표적에 할당되어 있다고 하고, 모든 무장의 할당 결과 k번째 표적의 잔존가치를 z_k라고 하자. i번째 무장을 l번째 표적으로 1-재위치 시킬 경우 목적함수 값의 변화량(Δz^1)은 다음과 같다.

$$\Delta z^1(i,l) = \frac{z_k}{1-p} + z_1(1-p_{ij}) - z_k - z_l = \frac{p_{ik}}{1-p} z_k - p_{il} z_l$$

또한, <그림 1>의 경우 1과 같이 k_1번째 표적에 할당된 i_1번째 무장과 k_2번째 표적에 할당된 i_2번째 무장에 대해 표적을 각각 l_1, l_2번째 표적으로 재위치 시킬 때 목적함수 값의 변화량은 다음과 같다.

$$\Delta z^{21}(i_1,i_2,l_1,l_2) = \frac{p_{i_1k_1}}{1-p_{i_1k_1}} z_{k_1} - p_{i_1l_1} z_{l_1} + \frac{p_{i_2k_2}}{1-p_{i_2k_2}} z_{k_2} - p_{i_2l_2} z_{l_2}$$

<그림 1>의 경우 2와 같이 두 무장이 같은 표적 l_1에 할당된다면 목적함수 값의 변화량은 다음과 같다.

$$\Delta z^{22}(i_1,i_2,l_1) = \frac{p_{i_1k_1}}{1-p_{i_1k_1}} z_{k_1} + \frac{p_{i_2k_2}}{1-p_{i_2k_2}} z_{k_2} - (1-p_{i_1l_1})(1-p_{i_2l_2}) z_{l_1}$$

또한 <그림 1>의 경우 3과 같이 상대의 표적으로 교환한다면 목적함수 값의 변화량은 다음과 같다.

$$\Delta z^{23}(i_1,i_2) = \frac{p_{i_1k_1} - p_{i_2k_1}}{1-p_{i_1k_1}} z_{k_1} - \frac{p_{i_2k_2} - p_{i_1k_2}}{1-p_{i_2k_2}} z_{k_2}$$

위 식들을 이용한 향상적 휴리스틱 알고리즘은 다음과 같다.

1-재위치 휴리스틱(H2)

1) 다음 절차를 반복한다.

1-1) 모든 각 무장(i)에 대하여 현재의 표적에서 다른 각 표적(l)으로 할당을 변경하였다고 가정하여 평가함수의 순 변화량 $\Delta z_1(i,l)$들을 구한다.

1-2) $\Delta z_1(i,l)$중 가장 작은 것을 선택하여 $\min\{\Delta z^1(i,l)\} < 0$이면 해당 무장의 할당을 변경하고, 변경된 할당의 두 표적에 대해 새로운 평가함수 값을 구한다. 그렇지 않고, $\min\{\Delta z^1(i,l)\} \geq 0$이면 종료한다.

교환 휴리스틱(H3)

1) 다음 절차를 반복한다.

1-1) 서로 다른 표적에 할당된 모든 두 무장(i_1,i_2)에 대하여 각자의 할당된 표적을 서로 교환 했다고 가정하여 평가함수의 순 변화량 $\Delta z^{23}(i_1,i_2)$을 구한다

1-2) $\Delta z^{23}(i_1,i_2)$중 가장 작은 것을 선택하여 $\min\{\Delta z^{23}(i_1,i_2)\} < 0$이면 해당 두 무장의 표적을 교환시키고, 변경된 두 표적에 대해 새로운 평가함수 값을 구한다. 그렇지 않고, $\min\{\Delta z^{23}(i_1,i_2)\} \geq 0$이면 종료한다.

2-재위치 휴리스틱(H4)

1) 다음 절차를 반복한다.

1-1) 서로 다른 표적에 할당된 모든 두 무장(i_1,i_2)에 대하여 다음을 계산한다.

1-1-1) 각자의 할당된 표적을 서로 다른 두 표적(l_1,l_2)로 변경했다고 가정하여 평가함수의 순 변화량 $\Delta z^{21}(i_1,i_2,l_1,l_2)$들을 구한다

1-1-2) 각자의 할당된 표적을 동일한 표적(l_1)으로 변경했다고 가정하여 평가함수의 순 변화량 $\Delta z^{22}(i_1,i_2,l_1)$들을 구한다

1-1-3) 각자의 할당된 표적을 교환했다고 가정하여 평가함수의 순 변화량 $\Delta z^{23}(i_1,i_2)$들을 구한다

1-2) Δz^{21}, Δz^{22}, Δz^{23} 들 중 가장 작은 것을 선택하여 $\min\{\Delta z^{21}(i_1,i_2,l_1,l_2), \Delta z^{22}(i_1,i_2,l_1), \Delta z^{23}(i_1,i_2)\} < 0$

이면 해당 두 무장의 표적을 변경하고, 변경된 두 표적에 대해 새로운 평가함수 값을 구한다. 그렇지 않으면 종료한다.

세가지 향상적 휴리스틱은 반복 횟수를 예측할 수 없으므로 시간 복잡성을 계산하기 어렵다. 가장 좋은 경우 1-재위치 휴리스틱은 한번의 반복을 수행할 때, $m(n-1)$번의 순 변화량을 구한다. 교환 휴리스틱은 한번의 반복에서 ${}_mC_2 = m(m-1)/2$번의 순 변화량을 계산한다. 2-재위치 교환 휴리스틱은 한번의 반복에서 ${}_mC_2 \times n^2 = m(m-1)n^2/2$번의 변화량을 계산하여 교환 휴리스틱에 비해 n^2배의 많은 시간 복잡성을 갖는다.

유전자 알고리즘의 해에 부분 최적화 방법을 적용한 대표적 연구로 이전에 언급한 Lee 등[6]의 연구를 들 수 있다. 그들은 Greedy Eugenics 라는 특별한 방법을 제안하였으나 이 방법은 향상적 휴리스틱이라기 보다는 본 연구에서의 구성적 휴리스틱 방법과 유사하다. 1 단계에서는 유전자 알고리즘의 해에서 무장이 할당된 표적 만을 대상으로 하여 구성적 휴리스틱 방법을 적용하였고, 모든 대상 표적이 할당되면 단계를 종료하여 할당이 안 되는 무장이 발생될 수 있도록 하였다. 2단계에서는 할당이 안된 무장을 대상으로 구성적 휴리스틱과 동일한 방법을 적용하였다. 유전자 알고리즘의 한 세대에서 생성된 모든 population에 대해 휴리스틱을 적용하여 2배의 population을 생성하였고, 이 들을 대상으로 룰렛 휠(roulette wheel) 방식에 의해 원래 크기의 population을 선택하였다. 부분 최적화 방법으로 구성적 휴리스틱과 유사한 방법을 사용하였으므로 새로이 생성된 population들은 동일한 해를 다수 포함하여 빠른 수렴성을 보장하지만 유전자 알고리즘의 장점인 해의 다양성을 가지지 못하게 될 것은 분명하다. <표 1>은 Lee 등의 연구에서 사용한 방법과 본 연구에서 사용한 방법의 차이를 나타낸다. 본 연구에서는 유전자 알고리즘을 통해 생성된 각 세대에서의 가장 좋은 해에 부분 최적화를 적용하여 계산시간의 단축을 도모하였고, 부분 최적화의 해는 순열 표현으로의 코딩이 용이하지 않은 이유로 유전자 알고리즘에 반영하지 않았다.

〈표 1〉 표적번호 표현과 순열 표현의 유전자 알고리즘 비교

구분	Lee 등[6]	GA+LOCALOPT
염색체 표현방법	표적번호 표현	순열 표현
부분 최적화 방법	Greedy Eugenics	재위치, 교환
부분 최적화 대상	각 세대에서 생성된 모든 유전자 알고리즘의 해에 적용	각 세대에서 생성된 유전자 알고리즘의 해 중 가장 좋은 해에 적용
부분 최적화 해의 피드백	유전자 알고리즘의 선택 절차에서 대상 염색체에 포함	부분 최적화 해는 유전자 알고리즘으로 피드백되지 않음

무장 할당 문제에 좋은 결과를 가져온다고 보고된 향상적 휴리스틱인 VLSN[1]은 본 연구에서 설명된 교환 휴리스틱과 많은 공통점을 가지고 있다. VLSN은 두 무장의 표적 교환뿐 만 아니라 K개 무장의 표적을 차례대로 바꾸는 일반적인 절차를 포함한다. 예를 들어, 3개의 무장에 대한 교환은 1번 무장의 표적은 2번 무장의 표적으로 바꾸고, 2번 무장의 표적은 3번 무장의 표적, 마지막으로 3번 무장의 표적은 1번 무장으로 바꾸는 사이클 적인 교환을 의미한다. 그러나 교환되는 무장의 수가 클수록 탐색하여야 하는 사이클의 복잡성이 증가하여 특별한 절차가 요구된다. Ajoua 등[1]의 연구에서는 3개 이상 무장에 대한 교환은 큰 향상을 나타내지 않고, 2개 무장의 교환이 가장 큰 향상을 가져온다고 보고하였다. 본 연구에서는 이 같은 결과에 기초하여 2개 무장의 교환만을 고려하였다.

5. 실험 및 분석

5.1 실험대상 휴리스틱 방법

실험 대상은 <표 2>와 같이 두가지 유전자 알고리즘인 표적번호 표현방법과 순열 표현방법에 기초한 것, 유전자 알고리듬에 부분 최적화 방법을 추가한 것, 그리고, 구성적, 향상적 휴리스틱의 8가지로 구성하였다. 유전자 알고리즘의 파라미터들인 개체수, 교배확률, 전위확률은 Lee 등[6]의 연구에서 사용한 동일한 값들을 사용하였고, 세대수는 모든 문제에 대해 500으로 한정하였다. 표적번호 표현 방법의 유전자 알고리즘인 GA1의 교배, 돌연변이 연산과 부분 최적해 방법으로 교환휴리스틱을 적용한 GA1+H3에서 두 알고리듬의 연결 방식은 <표 2>에서 언급한 Lee 등의 방법에 기초한다. 다만, GA1+H3의 부분 최적해 방법으로 greedy Eugenic 대신에 교환 휴리스틱을 적용하였다. 향상적 휴리스틱인 H2, H3, H4는 구성적 휴리스틱 H1의 해를 초기해로 하였다. 실험대상 알고리즘들은 JAVA 프로그래밍 언어로 구현되어 Intel 2.33 GHz CPU 와 2 GB의 메모리를 갖춘 PC에서 실행되었다.

〈표 2〉 실험대상 휴리스틱 방법

방법	설명	비고
GA1	표적번호표현방법 유전자 알고리즘	개체수: 무장수× 표적수 세대수: 500 교배확률: 0.8 전위확률: 0.4
GA1+H3	표적번호표현방법 유전자알고리즘과 교환 휴리스틱에 의한 부분 최적화	
GA2	순열표현방법 유전자 알고리즘	
GA2+H3	순열표현방법 유전자 알고리즘과 교환 휴리스틱에 의한 부분최적화	
H1	구성적 휴리스틱	
H1+H2	H1의 해를 초기해로 하여 1-재위치 휴리스틱에 의한 부분 최적화	
H1+H3	H1의 해를 초기해로 하여 교환 휴리스틱에 의한 부분 최적화	
H1+H4	H1의 해를 초기해로 하여 2-재위치 휴리스틱에 의한 부분 최적화	

5.2 랜덤의 파괴확률

실험을 위한 랜덤의 문제를 만들기 위해 Ahuja 등[1]에서 제안한 방법을 사용하였다. 표적의 가치는 25 ~ 100사이의 균등분포에 따르도록 했고, 무장의 표적에 대한 파괴확률은 0.6부터 0.9사이의 균등분포에 따르도록 하였다. 23가지의 무장의 수와 표적의 수를 대상으로 각 경우에서 10개의 문제를 생성하여 총 230개의 문제를 만들었다.

<표 3>과 <표 4>는 23개 문제인 각 (무장수, 표적수)에 대해 8가지 방법들에 대한 실험결과로 차이율과 계산시간을 나타낸다. GA2+H3에 의한 해가 대체로 가장 우수하여 이 해의 목적함수 값을 기준으로 각각의 방법들에 대한 차이율을 계산하였다.

$$\text{차이율} = \frac{(\text{휴리스틱에 의한 값}) - (GA2+H3\text{에 의한 값})}{(GA2+H3\text{에 의한 값})} \times 100$$

<표 3>의 유전자 알고리즘에 대한 결과에 따르면 두가지 유전자 알고리듬인 GA1과 GA2 중 GA1이 해의 질적 측면에서 더 나쁜 결과를 가져왔다. 표적번호 표현방법에 따른 GA1은230개 문제에 대한 평균 차이율이 273.5%로 GA2의 평균 차이율 27.3%에 비해 10배 이상이나 되었다. 이는 GA2에서 사용한 순열표현의 복호화 방법에 의한 해의 해석 결과가 랜덤성을 가지는 유전자 알고리즘의 GA1에 비교하여 약간의 시간을 더 소요했지만 상대적으로 우수한 해를 생성했다고 볼 수 있다. 유전자 알고리즘에 부분 최적해를 적용한 결과는 GA2+H3이 GA1+H3에 비해 우수한 결과를 가져왔다. 특히, GA2+H3은 8가지 방

〈표 3〉 랜덤의 파괴확률 문제에서 유전자 알고리즘들의 실험결과

문제	GA1		GA1+H3		GA2		GA2+H3	
	차이율(%)	시간(초)	차이율(%)	시간(초)	차이율(%)	시간(초)	차이율(%)	시간(초)
(10,5)	35.9	0.048	0.0	0.291	0.1	0.034	0.0	0.039
(10,10)	39.1	0.112	0.0	0.679	0.0	0.097	0.0	0.100
(10,20)	12.8	0.234	2.9	1.488	0.0	0.254	0.0	0.257
(20,10)	145.9	0.403	4.2	8.739	20.9	0.357	0.0	0.387
(20,20)	99.7	0.872	29.3	20.542	6.9	0.964	0.0	0.985
(20,40)	25.2	1.943	12.7	43.843	0.4	2.899	0.0	2.914
(20,80)	9.7	4.769	5.4	88.322	0.0	9.466	0.0	9.471
(40,10)	697.7	1.646	15.8	125.090	122.5	1.298	0.0	1.563
(40,20)	322.4	3.457	57.5	303.579	46.4	3.643	0.0	3.904
(40,40)	155.2	7.594	71.9	716.771	15.3	11.069	0.0	11.326
(40,80)	35.5	18.186	21.7	1409.918	1.0	36.892	0.0	37.018
(40,120)	20.2	32.097	12.2	2175.647	0.3	77.944	0.0	78.025
(80,20)	1851.8	15.147	89.4	4597.592	197.4	14.905	0.0	17.242
(80,40)	539.3	32.805	134.3	10835.589	56.2	44.740	0.0	46.963
(80,80)	202.9	77.634	121.4	24029.872	16.7	149.270	0.0	151.309
(80,160)	42.2	199.905			1.3	529.814	0.0	530.986
(80,320)	16.0	574.836			0.2	1977.428	0.0	1978.117
(100,50)	614.8	70.170			56.4	104.009	0.0	108.698
(100,100)	216.0	167.961			16.4	351.736	0.0	355.822
(100,200)	45.2	437.609			1.2	1262.137	0.0	1264.632
(200,100)	860.0	785.917			50.9	1463.591	0.0	1502.807
(200,200)	252.9	1986.196			15.1	5148.610	0.0	5183.573
(200,400)	50.7	2009.674			1.1	8765.853	0.0	8789.592
평균	273.5		38.6		27.3		0.0	

〈표 4〉 랜덤의 파괴확률 문제에서 휴리스틱의 실험결과

문제	H1		H1+H2		H1+H3		H1+H4	
	차이율(%)	시간(초)	차이율(%)	시간(초)	차이율(%)	시간(초)	차이율(%)	시간(초)
(10,5)	9.2	<0.001	9.2	<0.001	5.2	<0.001	5.2	<0.001
(10,10)	10.9	<0.001	10.9	<0.001	1.7	<0.001	1.7	0.002
(10,20)	1.9	<0.001	1.9	<0.001	0.9	<0.001	0.9	<0.001
(20,10)	11.3	<0.001	11.3	<0.001	3.0	<0.001	3.0	0.002
(20,20)	7.0	<0.001	7.0	<0.001	2.3	<0.001	2.3	0.003
(20,40)	1.6	<0.001	1.6	<0.001	1.0	<0.001	1.0	0.017
(20,80)	0.4	<0.001	0.4	<0.001	0.3	<0.001	0.3	0.059
(40,10)	12.8	<0.001	12.8	<0.001	3.8	<0.001	3.8	0.008
(40,20)	9.7	<0.001	9.7	<0.001	2.2	<0.001	2.2	0.034
(40,40)	3.5	<0.001	3.5	<0.001	0.8	<0.001	0.8	0.125
(40,80)	0.8	<0.001	0.8	<0.001	0.4	<0.001	0.3	0.541
(40,120)	0.4	0.002	0.4	0.002	0.2	0.002	0.2	0.877
(80,20)	8.3	<0.001	8.3	<0.001	-1.2	0.002	-1.2	0.204
(80,40)	1.4	0.002	1.4	0.002	-2.0	0.002	-2.0	0.674
(80,80)	0.4	<0.001	0.4	<0.001	-1.4	<0.001	-1.4	3.184
(80,160)	0.1	0.003	0.1	0.003	-0.1	0.005	-0.2	10.803
(80,320)	0.0	0.005	0.0	0.005	0.0	0.005	0.0	28.138
(100,50)	1.8	0.002	1.8	0.002	-1.9	0.003	-1.9	2.259
(100,100)	0.3	0.003	0.3	0.003	-1.5	0.005	-1.5	9.195
(100,200)	0.0	0.002	0.0	0.002	-0.2	0.002	-0.2	32.635
(200,100)	-2.0	0.005	-2.0	0.005	-3.9	0.009	-3.9	50.883
(200,200)	-1.4	0.019	-1.4	0.019	-2.4	0.025	-2.3	156.836
(200,400)	-0.3	0.042	-0.3	0.045	-0.3	0.045	-0.3	395.699
평균	3.4		3.4		0.3		0.3	

법 중 가장 우수한 해를 생성하였다. 그러나 계산시간 측면에서는 두 알고리즘 모두 나쁜 결과를 초래하였다. 무장, 표적의 수가 각각 200, 400인 가장 큰 문제에서는 GA2+H3가 평균 8790초(=2.44시간)이 걸렸다. GA1+H3은 (80, 80)의 문제에서 조차 평균 24029.872초(6.68시간)가 소요되어 그 이상의 큰 문제에서는 계산시간의 부담으로 실행을 하지 않았다. 따라서, 유전자 알고리즘에 부분 최적화를 적용한 방법은 비교적 작은 문제에서 조차 적지 않은 시간이 소요되어 실제적인 상황에서 실시간으로 해를 도출하는데 어려움이 있을 것이다.

<표 4>의 4가지 휴리스틱에 대한 결과에서 구성적 휴리스틱 H1은 모든 방법 중 가장 적은 시간이 소요되었고, 해의 차이율이 평균 3.4%에 불과하였다. 작은 문제에서는 0.001초 보다 작은 시간이 걸렸고, (200, 400)의 가장 큰 문제에서도 0.042초가 걸렸다. 향상적 휴리스틱 H1+H2의 차이율과 계산시간은 구성적 휴리스틱 H1과 거의 동일하다. H1의 해를 초기해로 하여 향상적 휴리스틱을 실행하였다는 점을 고려할 때 H2에서 적용된 1-재위치의 기여가 거의 일어나지 않았다고 볼 수 있다. 교환 알고리듬인 H1+H3의 차이율은 0.3%로 H1의 차이율 3.4%에 비해 10배 이상의 해의 향상을 가져왔고, 계산시간은 가장 큰 문제에서도 0.045초가 걸려 매유 효율적인 방법임을 의미한다. 2-재위치 방법인 H1+H4는 해의 차이율이 H1+H3과 거의 동일한 반면 계산시간은 평균 8000배 이상이 소요되었다. 따라서 2-재위치에서 고려한 3가지 경우 중 교환만이 해의 향상에 큰 기여를 했고, 나머지 두 경우는 적지 않은 계산시간이 소요된 반면 해의 향상에 거의 기여를 하지 않았다고 볼 수 있다. 결론적으로 교환 휴리스틱에 기인한 H1+H3가 해의 질적 측면이나 계산시간면에서 가장 바람직한 결과를 가져왔다.

5.3 이행성(Transitivity)규칙에 따르는 파괴확률

두 무장 W1, W2가 있을 때 W1이 표적 T1에 대해 W2보다 큰 파괴확률을 가지는 우수한 무장이라면 또 다른 표적 T2에 대해서도 W1이 W2보다 큰 파괴확률을 가질 것으로 예상된다. 무장간의 성능에 이러한 이행성 규칙이 적용될 때를 대상으로 실험을 수행하였다. 즉, 모든 표적에 대해 다음과 같은 파괴확률간의 관계가 성립하도록 하였다.

$$p_1 > p_2 > p_3 > \cdots > p_{mj}, j = 1,2,\cdots,n$$

무장 1은 어느 표적에 대해서도 가장 큰 파괴확률을 가진다. 그 다음으로 무장 2, 무장 3, 무장 4 등의 순서로 큰 파괴확률을 가진다. 이러한 무장간 파괴확률의 크고 작음은 모든 표적에 동일하게 적용된다.

이와 같은 무장 성능 간 이행성 특성의 데이터를 생성하기 위하여 이전의 랜덤 파괴확률 경우와 동일하게 표적의 가치를 구하였고, 각 표적에 대해 m개의 파괴확률을 동일한 방법으로 구한 후 이들을 내림순으로 정렬하여 1번 무장부터 번 무장까지 차례대로 할당하였다.

<표 5>는 이러한 이행성 규칙에 따르는 데이터를 대상으로 실험한 결과를 나타낸다. GA1, GA1+H3, GA2의 결과는 해의 질이나 실행시간 면에서 우수하지 못하여 생략하였고, H1+H2는 H1과 차이율과 계산시간이 동일하여 생략하였다. 랜덤의 치명확률 경우에서와 같이 GA2+H3가 가장 우수한 해를 생성했지만 큰 계산시간의 부담이 존재한다. 더욱이 랜덤의 파괴확률 문제에 비해 평균 1.17배의 많은 계산시간이 소요되었다. 계산시간은 GA2+H3 뿐 만 아니라 향상적 휴리스틱인 H1+H3, H1+H4에서도 랜덤의 파괴확률 문제에 비해 보다 많은 계산시간이 소요되었다.

랜덤의 파괴확률 경우와 같이 해의 질은 4가지

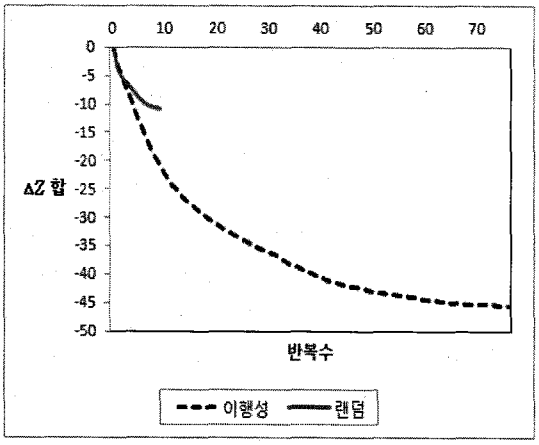
휴리스틱 중에 H1+H3가 가장 좋은 결과를 가져왔다. 그러나 랜덤의 파괴확률 경우에 평균 0.3%의 차이율을 보인 반면 평균 0.9%로 3배 정도의 큰 차이율을 보여 이행성 규칙의 문제에서 더 효율성이 떨어지는 결과를 가져왔다. 이 같은 결과는 H1, H1+H3, H1+H4에서도 동일하다. 따라서 모든 알고리즘들이 해의 질이나 계산시간 측면에서 이행성 규칙이 성립할 때 더 좋지 않은 결과를 초래하는 것으로 파악된다. <그림 2>는 무장, 표적수가 모두 80인 문제에서 랜덤 파괴확률과 이행성 규칙의 파괴확률 경우에 H1+H3에서 H3의 반복수에 따른 순 변화량의 합 $(\sum_{\text{반복}} \min\{\Delta z^{23}(i_1,i_2)\})$을 나타낸다. 랜덤 파괴확률에서는 9번의 반복을 수행한 반면 이행성 규칙의 파괴확률에서는 75번의 반복을 수행하였다. 또한 이행성 규칙에서는 H1의 해를 초기해로 하여 -45.4594의 목적함수 값 감소를 가져왔으나, 랜덤 파괴확률에서는 -10.9123의 감소를 달성하였다. 이 결과는 이행성 규칙 하에서의 초기해가 우수하지 못함을 의미한다. 랜덤 파괴확률의 경우에는 H1에 의한 초기해가 차이율이 3.4%로 우수하여 H3의 향상 휴리스틱에 의한 향상이 적게 이루어진 반면 이행성 규칙하에서는 초기해가 차이율 8.5%로 상대적으로 우수하지 못하여 H3의 반복수가 증가하였고, 많은 향상을 가져왔기 때문이다. 따라서 H1의 성능이 랜덤의 파괴확률 경우에서 보다 이행성 규칙의 파괴확률 하에서 더 뒤 떨어짐을 의미한다.

6. 결론

WTA문제를 빠른 시간 내에 해결하기 위한 방

〈표 5〉 이행성 규칙의 파괴확률 문제에서 실험결과

문제	GA2+H3	H1		H1+H3		H1+H4	
	시간(초)	차이율(%)	시간(초)	차이율(%)	시간(초)	차이율(%)	시간(초)
(10,5)	0.050	22.9	<0.001	3.6	<0.001	3.6	<0.001
(10,10)	0.133	9.4	<0.001	0.6	<0.001	0.6	<0.001
(10,20)	0.301	2.8	<0.001	0.3	<0.001	0.3	<0.001
(20,10)	0.444	19.3	<0.001	2.2	<0.001	2.2	0.003
(20,20)	1.133	7.8	<0.001	0.4	<0.001	0.4	0.020
(20,40)	3.369	2.2	<0.001	0.1	<0.001	0.1	0.075
(20,80)	10.884	0.9	<0.001	0.1	<0.001	0.1	0.325
(40,10)	1.776	35.6	<0.001	6.2	<0.001	6.2	0.045
(40,20)	4.496	15.9	<0.001	1.8	<0.001	1.8	0.198
(40,40)	13.058	6.4	<0.001	0.3	<0.001	0.3	0.714
(40,80)	42.636	1.9	<0.001	0.1	<0.001	0.1	2.905
(40,120)	89.197	1.0	<0.001	0.1	<0.001	0.0	7.029
(80,20)	19.585	25.7	<0.001	3.4	0.002	3.4	1.537
(80,40)	54.292	11.7	0.002	1.3	0.011	1.3	6.900
(80,80)	176.487	4.9	<0.001	0.0	0.006	0.0	29.195
(80,160)	617.485	1.4	0.002	0.1	0.011	0.1	106.483
(80,320)	2273.736	0.6	0.006	0.0	0.012	0.0	459.123
(100,50)	125.374	9.9	<0.001	0.9	0.014	0.9	21.890
(100,100)	413.507	3.7	0.002	-0.1	0.017	-0.1	82.750
(100,200)	1470.267	1.3	0.006	0.1	0.022	0.1	337.402
(200,100)	1731.647	7.2	0.012	0.3	0.147	0.3	826.131
(200,200)	5951.648	2.3	0.019	-0.4	0.136	-0.4	3103.914
(200,400)	10076.763	0.9	0.040	0.0	0.170	0.0	12298.651
평균		8.5		0.9		1.0	



〈그림 2〉 (80, 80)의 랜덤파괴확률과 이행성 규칙의 파괴확률에서 H1+H3의 순변화량 합 추이

법으로 휴리스틱이 널리 이용된다. 휴리스틱의 효율성을 분석한 대부분의 연구에서는 표적에 대한 무장의 파괴확률을 랜덤으로 생성한 가상적인 데이터를 이용한다. 본 연구에서는 무장 간의 파괴확률의 크기가 표적에 대해 이행성이 존재하는 경우를 고려하여 가상적인 데이터를 생성하였고, 이를 바탕으로 실험을 하였다. 랜덤파괴확률에서는 순열표현 방법의 유전자 알고리즘에 교환 휴리스틱을 부분최적화를 혼용한 경우의 해의 질이 가장 우수했으나 많은 계산시간으로 실제적으로 활용하기에 부담이 된다. 휴리스틱 방법들은 0.1초 이내에 해를 구하였고, 해의 질은 가장 좋은 유전자 알고리즘의 해에 비해 평균 0.3% 뒤떨어졌다. 이행성 규칙이 존재하는 경우 랜덤의 파괴확률 경우에 비해 휴리스틱의 성능이 뒤떨어지고, 향상적 휴리스틱 방법은 보다 많은 계산시간을 필요로 하였다. 구성적 휴리스틱인 H1의 해를 초기해로 하여 향상적 휴리스틱인 H3를 적용한 경우도 동일한 결과를 보였지만 분석 대상이 된 방법 중 많은 계산시간으로 가장 좋은 해를 생성한 유전자알고리즘 GA2와 부분 최적화 H3의 혼용 결과에 비해 해의 질이 평균 0.3%의 적은 차이가 나는 좋은 결과를 가져왔고, 계산시간은 200개 무장, 400개 표적의 거대한 문제에서도 PC에서 0.17초의 짧은 계산시간을 필요로 하였다.

이행성 규칙이 성립할 때 이러한 구성적 휴리스틱과 향상적 휴리스틱을 혼용하는 경우 본 연구에서 제안된 H1보다 더 우수한 해를 생성할 수 있는 구성적 휴리스틱 방법의 개발이 중요하여 앞으로 이에 대한 연구가 지속되어야 할 것이다.

참고문헌

[1] Ahuja, R. K., Kumar, A., Jha, K. C., and Orlin, J. B., "Exact and Heuristic Algorithms for the Weapon-Target Assignment Problem", Operations Research, Vol. 55. No. 6, pp. 1136-1140, 2007.

[2] Davis, L., "Applying Adaptive Algorithms to Epistatic Domains", Proceedings of the International Joint Conference on Artificial Intelligence, 162-164, 1985.

[3] Goldberg, D. E. and Lingle, R., "Alleles, Loci, and the TSP", Proceedings of the First International Conference on Genetic Algorithms, Lawrence Erlbaum Associates, Hillsdale, NJ, 154-159, 1985.

[4] Harhalakis, G., Proth, J. M., Xie, X. L., "Manufacturing cell design using simulated annealing: An industrial application", Journal of Intelligent Manufacturing, pp. 185-191, 1990.

[5] Jones, D. R. and Bertramo, M. A., "Solving Partitioning Problems with Genetic Algorithms", Proceedings of the Fourth International Conference on Genetic Algorithms, Morgan Kaufmann Publishers, Los Altos, CA, 442-449, 1991

[6] Lee, Z. J., Fu, S. F., and Lee, C. Y., "Efficiently Solving General Weapon-Target Assignment Problem by Genetic Algorithms With Greedy Eugenics", IEEE Transactions on Systems, man,

and cybernetics, Vol. 33, No. 1, pp. 113-121, 2003

[7] Lloyd, S. P. and Witsenhausen, H. S., "Weapon Allocation is NP-complete", Proc. 186 Summer Conf. Simulation, Reno, NV, 1054-1058, 1986

[8] Michaelwicz, Z., Genetic Algorithms + Data Structures = Evolution Programs, Springer-Verlag, 1992.

[9] Murphey, R. A., "Target-based weapon target assignment problems", P. M. Pardalos, L. S. Pitsoulis, eds, Nonlinear Assignment Problems: Algorithms and Applications, Kluwer Academic Publishers, Boston, MA. 39-53, 1999

[10] Oliver, I. M., Smith, D. J., and Holland, J. R. C., "A Study of Permutation Crossover Operators on the traveling Salesman Problem", Proceedings of the Second International Conference on Genetic Algorithms, Lawrence Erlbaum Associates, Hillsdale, NJ, 224-230, 1987.

저자소개

임 동 순(E-mail: dsyim@hnu.kr)

1982	한양대학교 산업공학과 졸업(학사)
1986	한국과학기술원 산업공학과 졸업(석사)
1991	아이오와 주립대학 산업공학과 졸업(박사)
현재	한남대학교 산업경영공학과 교수
관심분야	이산사건 시뮬레이션, 국방 모델링&시뮬레이션

최 봉 완(E-mail: bwchoi@hnu.kr)

1981	해군사관학교 졸업(이학사)
1985	서울대학교 산업공학과 졸업(공석사)
1988	아이오와 주립대학 산업공학과 졸업(공학석사)
1994	아이오와 주립대학 산업공학과 졸업(공학박사)
2000	RAND 연구소 객원 연구원
현재	한남대 국방전략대학원 교수/무기체계 • M&S 연구센터장
관심분야	국방 모델링&시뮬레이션, 획득관리