

# 진화 알고리즘에서의 벡터 휴리스틱을 이용한 조합 최적화 문제 해결에 관한 연구

안 종 일<sup>†</sup> · 정 경 숙<sup>†</sup> · 정 태 충<sup>††</sup>

## 요 약

본 논문에서는 진화 알고리즘에 기반하여 조합 최적화 문제를 해결하고자 한다. 진화 알고리즘은 대규모 문제 공간에서 최적화 문제를 해결하는데 적합한 알고리즘이다. 본 논문의 조합 최적화의 예는 경수로 원자로로부터 나온 폐연료를 중수로에서 재사용하는데 필요한 폐연료의 조합 문제이다. 이와 같은 조합 최적화 문제는 0/1 knapsack 문제와 같이 NP-Complete 문제에 해당한다. 이러한 문제를 해결하기 위해서는 고전적인 진화 알고리즘의 전략에 기반하여 랜덤 연산자를 이용하되 평가 함수 값이 좋은 방향으로만 탐색을 수행하는 방법, 그리고 벡터 연산자를 이용하여 최적의 해를 보다 빨리 얻을 수 있는 휴리스틱을 사용하는 방법이 있다. 본 논문에서는 중수로 연료 조합 문제 영역의 모든 지식을 벡터화하여 벡터의 연산만으로 가능성 검사, 해를 평가하는 방법을 소개한다. 또한 벡터 휴리스틱이 고전적인 진화 알고리즘에 비해 어느 정도의 성능을 보이는지 비교한다.

## Vector Heuristic into Evolutionary Algorithms for Combinatorial Optimization Problems

Jong Il Ahn<sup>†</sup> · Kyung Sook Jung<sup>†</sup> · Tae Choong Chung<sup>††</sup>

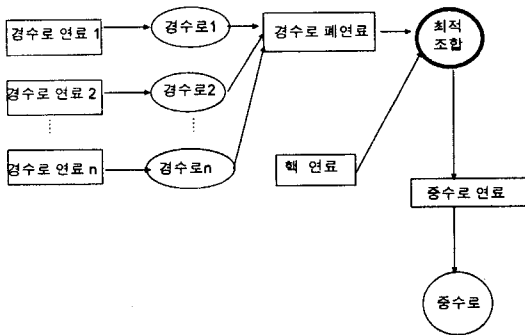
### ABSTRACT

In this paper, we apply the *evolutionary algorithm* to the combinatorial optimization problem. Evolutionary algorithm useful for the optimization of the large space problem. This paper propose a method for the reuse of wastes of light water in atomic reactor system. These wastes contain several reusable elements, and they should be carefully selected and blended to satisfy requirements as an input material to the heavy water atomic reactor system. This problem belongs to an NP-hard like the 0/1 knapsack problem. Two evolutionary strategies are used as approximation algorithms in the highly constrained combinatorial optimization problem. One is the traditional strategy, using random operator with evaluation function, and the other is heuristic based search that uses the vector operator reducing between goal and current status. We also show the method which perform the feasible test and solution evaluation by using the vectored knowledge in problem domain. Finally, We compare the simulation results of using random operator and vector operator for such combinatorial optimization problems.

† 준 회 원: 경희대학교 전자계산공학과  
†† 정 회 원: 경희대학교 전자계산공학과  
논문접수: 1996년 10월 4일, 심사완료: 1997년 5월 10일

### 1. 서 론

최근 원자력 발전소에 대한 관심은 환경 보호 문제와 관련되어 매우 비중 있게 다루어지고 있다. 이와 관련하여 경수로에서 사용된 폐연료를 중수로에서 재사용하는 방안이 모색되고 있다. 중수로의 경우는 경수로와 연료 조건이 다르다. 경수로의 폐연료들은 여러가지의 원소를 포함하고 있는데, 각각의 원자로와 생성된 시기에 따라 각 원소의 함량이 다양하다. 이들 각 폐연료들의 주요 관심 구성 원소는 U-235, U-238, PU-239, PU-240, PU-241의 5가지이다. 이 폐연료를 재사용하기 위해서는 각 구성 원소들이 단위 무게 당 필요로 하는 요구량을 만족해야 한다. 그러나 이 폐연료들은 각 원소의 함량이 일정치 않아 그 원소들을 분리, 추출하여 사용할 수 없는 제약조건을 갖고 있다. 따라서 폐연료를 혼합하여 모든 구성 원소의 함량이 요구량에 도달하여야 재사용이 가능하다. 문제 발생의 위치는 (그림 1)과 같으며, 본 연구의 목적은 여러 폐연료들 중 어떤 것들을 혼합하는 것이 모든 원소의 조건을 만족할 수 있는지에 대해 탐색하는 것이다.



(그림 1) 문제 발생의 위치  
(Fig. 1) Location of Problem

이 문제는 전체 집합으로부터 사용 목적에 맞는 부분 집합을 조합하는 것으로 최적화 문제에 해당한다. 최적화 문제에는 대표적으로 0/1 Knapsack 문제[7]를 꼽을 수 있다. 이와 같은 문제는 NP-complete에 속하는 문제로서 전통적인 탐색 알고리즘으로는 시간과 비용의 문제로 해결이 곤란하다. 특히, 각 집합의 원

소가 분해될 수 없고, 그 경우의 수가 매우 많은 경우는 문제의 복잡도는 더욱 더 커지게 된다. 이러한 문제에 대한 효과적인 해결 방법으로는 유전자 알고리즘(Genetic Algorithm), 진화 알고리즘(Evolutionary Algorithm)[4][5][6]등이 있다.

유전자 알고리즘, 진화 알고리즘 등은 적자 생존에 착안한 알고리즘으로서 문제 공간을 사전에 알 수 없는 상태에서 무작위 선택(random) 방법을 사용하여 언덕 오르기 탐색(Hill climbing)을 하거나, 패턴을 분류하는 방법을 사용한다.[1][2][3] 그러나 위의 방법들은 문제 공간에 대한 다양한 정보가 사전에 주어진 상태임에도 불구하고, 그 정보를 제대로 이용하지 못한다는 단점이 있다. 따라서 사전 정보를 이용할 수 있도록 연산자를 설계하여 탐색의 속도를 개선하는 방법이 필요하다. 폐연료의 최적 조합을 찾는 문제의 경우 이미 알 수 있는 유용한 정보로는 전체 폐연료 집합에 포함되어 있는 원소들의 분포에 대한 정보이다.

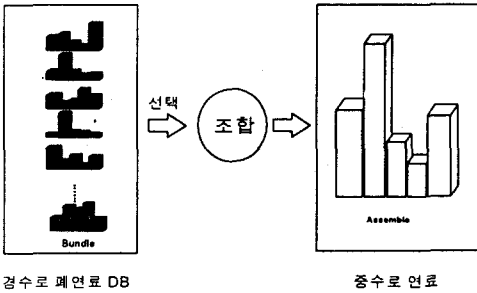
따라서 본 연구에서는, 원소의 분포를 벡터화하여 벡터 연산에 기반한 진화 알고리즘의 연산자를 설계하여 해에 빨리 접근하도록 유도함으로써 탐색의 속도를 단축시키고 최적의 폐연료 집합을 찾는 방법과 탐색 전에 미리 문제 공간을 분석하여 목표 해를 구할 수 있는지를 검증하는 가능성 검사의 방법을 제안하고자 한다.

### 2. 문제 정의

경수로의 연료로 핵연료를 사용한 후 그 폐연료를 재사용하기 위해서는 몇가지 고려되어야 할 점이 있다. 경수로에서 사용된 폐연료를 번들(Bundle)이라 하며, 이것은 여러가지 원소들로 구성되어 있다. 그리고 각 번들의 원소 함량은 생성 장소와 생성 일시에 따라 각각 다르다. 이 폐연료를 중수로의 연료로 재사용하기 위해서는 폐연료의 각 원소들이 단위 무게 당 필요로 하는 요구량을 함유하고 있어야 하지만, 단독 번들로는 그 요구 조건을 만족시키지 못한다. 따라서 여러개의 적절한 번들들을 혼합하여 어셈블(Assemble)을 구성한다.

(그림 2)에서 보는 바와 같이 본 연구에서는 U-235, U-238, PU-239, PU-240, PU-241의 다섯가지 원소에 대해서 그 함량을 고려하며, 다양한 형태의 번들로

어셈블을 구성하였을때, 어셈블내의 각 원소의 함량이 단위 무게 당 요구량에 가장 근사하게 접근하는 번들들을 찾고자 한다.



(그림 2) 문제의 정의  
(Fig. 2) Define of Problem

3. 폐연료 재활용을 위한 시스템 구성

3.1. 문제 공간의 분석

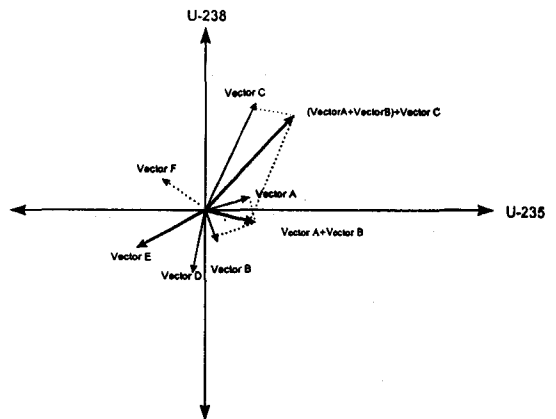
문제의 공간은 5개의 원소, U-235, U-238, PU-239, PU-240, PU-241로 구성된 폐연료의 집합이다. 먼저 각 폐연료, 즉 번들들을 섞어서 기준 함량에 맞추기 위하여 각 원소들을 벡터화 하였다. 여기서, 벡터화란 각각의 원소들을 순서쌍의 성분으로 나타내어 그 구성의 분포를 보는 것이다. 구성 원소가 5가지 이므로 5차원의 벡터가 만들어지는데, 5차원의 경우는 그래프상에 표현하기 어려우므로 여기서는 2차원, 즉 U-235와 U-238의 두가지 원소를 예로 들어서 설명하기로 한다. <표 1>에서 각 원소의 단위 무게 당 요구량은 U235인 경우는 0.16kg이고, U238의 경우는 17.20kg이다. 이것을 상대적인 기준량 즉, 각 기준량을 0.0kg에 해당하는 비율로 바꾸고, 각 번들의 함유량을 상대적인 과부족량으로 계산함으로써 각 번들의 형태를 결정할 수 있다. 여기서 각 원소의 요구량을 상대적인 기준량으로 바꾸어 계산하는 것은 각 원소마다 기준량을 정규화한 것에 해당한다. 두가지 원소만을 고려하였을 경우는 2차원의 벡터(U-235, U-238)로 나타낼 수 있으며, 이 벡터의 사분면은 <표 1>의 TYPE 열에서 볼 수 있다.

(그림 3)과 같은 경우 벡터(vector) A, B, C인 번들로 어셈블을 구성하며, 이 벡터의 크기가 어셈블의

<표 1> U235, U238 원소의 벡터 분면  
<Table 1> Vectorizing of U235, U238

원소	U235	U238	원소	U235	U238	TYPE
단위 요구량	0.16	17.20	단위요구량에 대한 상대 기준량	0.00	0.00	사분면
단위	1.41	19.30	상대	1.25	2.10	1사분면(+, +)
	0.08	20.15		-0.08	2.95	2사분면(-, +)
	0.20	14.70		0.04	-2.50	4사분면(+, -)
무게당	1.05	15.05	기준량에	0.89	-2.15	4사분면(+, -)
	0.18	19.65		0.02	2.45	1사분면(+, +)
원소	0.21	20.70	대한	0.05	3.50	1사분면(+, +)
	0.11	17.05		-0.05	-0.15	3사분면(-, -)
함량	0.14	12.65	과부족량	-0.02	-4.55	3사분면(-, -)
	0.09	19.75		-0.07	2.55	2사분면(-, +)
	0.07	13.65		-0.09	-3.55	3사분면(-, -)

오차이다. 이 오차를 줄이기 위하여 벡터 A, B, C중에서 어셈블 벡터와 방향 성분이 유사한 벡터를 제거한다. 여기서의 합 벡터와 유사한 벡터 C를 제거한다. 대치시킬 벡터는 어셈블에 포함되어 있지 않은 벡터 D, E, F 중에서 선택한다. 이때 벡터 C를 제거한 어셈블은 벡터 A와 벡터 B의 합이므로 이 두 벡터의 합 벡터와 방향 성분이 반대인 벡터를 선택한다. 그러므로 벡터 D 또는 E를 선택하는 것보다 벡터 F를 선택

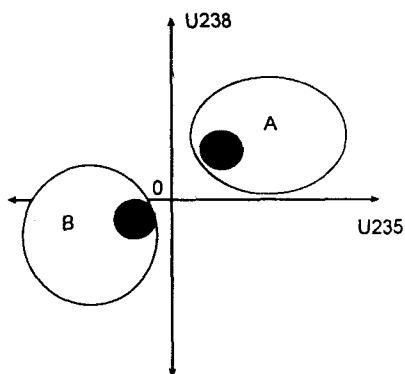


(그림 3) 원소들의 벡터화  
(Fig. 3) Vectorizing of Elements

하는 것이 어셈블의 오차를 상대적으로 빨리 줄일 수 있다.

### 3.2. 가능성 검사

주어진 자료를 분석한다. 기존 자료의 분포가 재 활용 가능한 어셈블을 구성할 수 있는 원소합량을 포함하는지 아닌지에 대한 분석이다. 자료 분석 후 적당한 조합으로 요구량을 만들 수 없다면 중수료에 적합한 요소를 많이 가진 핵연료를 추가해야 한다.



(그림 4) 완전해를 얻을 수 없는 자료의 분포 예  
(Fig. 4) Data distribution of worst case

(그림 4)와 같이 모든 자료의 분포가 A 위에 있거나 B 위에 있으면 해를 구할 수 없다. 이 경우는 오차가 가장 적도록 C에 위치한 번들을 선택하여 어셈블을 구성하였다 할지라도 기준량을 초과하게 된다. 따라서 검사 방법은 모든 자료의 집합에서 각 원소별로 값이 최소인 번들을 찾아 각각 어셈블을 구성한다. 그리고 각 원소의 합량이 기준량을 초과한다면 해를 구할 수 없다고 판단한다. 반대로 최대값을 갖는 번들로 어셈블을 구성하였을 경우 원소의 합량이 기준량보다 미달이라면 해를 구할 수 없다. 따라서 가능성 검사는 다음과 같은 수식으로 표현할 수 있다.

• 해가 가능한 경우의 자료 분포

$$\frac{1}{K} \sum_{k=1}^K \{Y_i(x)_k\} < D_i(x) < \frac{1}{K} \sum_{k=N-K+1}^N \{Y_i(x)_k\}, i=1, \dots, 5$$

• 해가 없는 경우

$$D_i(x) < \frac{1}{K} \sum_{k=1}^K \{Y_i(x)_k\}$$

$$\text{또는 } D_i(x) > \frac{1}{K} \sum_{k=N-K+1}^N \{Y_i(x)_k\}, i=1, \dots, 5$$

$D_i(x)$ : i번째 원소의 기준량

$Y_x$ : i번째 원소의 실제량

$\{Y_i(x)\}$ : i번째 원소를 작은 값부터 차례로 정렬

$K$ : 한 어셈블 내의 번들의 개수

$N$ : 전체 자료의 개수

### 3.3. 평가 함수

본 연구에서 사용되어지는 한 어셈블에 대한 평가 함수는 다음과 같이 정의한다.

$$F(x) = \sum_{j=1}^K \left\{ \sqrt{\sum_{i=1}^E \{Y_i(i) - D_i(x)\}^2} \right\}_j$$

$Y_i(x)$ : 어셈블에 포함된 i번째의 단위 원소 합량

$D_i(x)$ : 요구되는 i번째 단위 원소 합량

$E$ : 원소의 개수

$K$ : 번들의 개수

즉,  $D_i(x)$ 는 단위 무게 당 요구되는 원소 합량이고,  $Y_i(x)$ 는 어셈블에 포함된 각 원소들의 합량을 단위 무게 당 합량으로 바꾸어 계산한 것이다. 그리고 평가 함수  $F(x)$ 는  $Y_i(x)$ 와  $D_i(x)$ 의 오차를 계산한 것이다. 즉, 모든 원소를 벡터화 시켰으므로  $F(x)$ 는 오차의 크기를 나타내는 것이다. 따라서 최적화는 평가 함수  $F(x)$ 의 값이 가장 적게 되도록 어셈블을 구성하는 것이다.

### 3.4. 연산자

1) 벡터화 기법을 이용한 연산자

연산자에서는 초기에 구성된 어셈블로부터 제거할 번들과 삽입할 번들을 교환하여 평가 함수의 평가값을 최소화시키는 방법을 사용한다. 어셈블에서 제거할 번들과 삽입할 번들의 선택 과정을 벡터 수식으로 정의하면 다음과 같다.

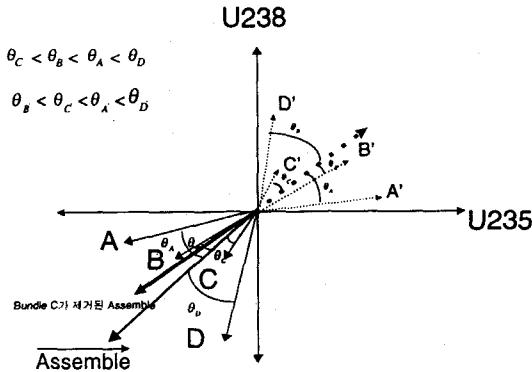
$$\begin{aligned} \text{Assemble} &= \overrightarrow{\text{Bundle}_1} + \overrightarrow{\text{Bundle}_2} + \dots + \overrightarrow{\text{Bundle}_K} \\ \text{Error} &= \overrightarrow{\| \text{Assemble} \|} \end{aligned}$$

• 번들의 제거와 삽입

$$\overrightarrow{Assemble} = \overrightarrow{Assemble} - \overrightarrow{Bundle}_R + \dots + \overrightarrow{Bundle}_I$$

즉, 제거 번들인  $Bundle_R = \text{Min}_{i \rightarrow K} (|\text{Assemble} - Bundle_i|)$

으로 이 번들을 제거할 때 오차가 최소가 되며, 삽입 번들인  $Bundle_I = \text{Min}_{i \rightarrow N} (|\text{Assemble} + Bundle_i|)$ 으로 이 번들의 삽입시 오차가 최소가 된다. 그러므로  $Bundle_R$ 는 어셈블의 오차를 최대로 했던 번들이며,  $Bundle_I$ 는 어셈블에 속해 있지 않으면서 어셈블 벡터와 반대 방향의 번들이다.  $K$ 는 한개의 어셈블 내의 번들의 갯수이며,  $N$ 는 폐연료 전체 자료의 갯수이다.



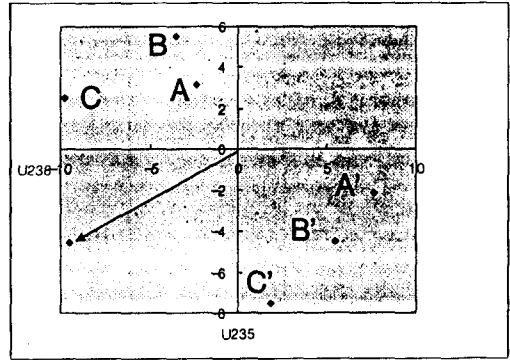
(그림 5) Assemble 과 번들이 이루는 각 (Fig. 5) Angle between Assemble and Bundle

(그림 5)에서 A, B, C, D는 어셈블을 구성하고 있는 번들이고, A', B', C', D'는 어셈블의 반대 방향에 있고 어셈블에 속하지 않은 번들이다. 이 가운데에서 제거할 번들은 어셈블과 이루는 각  $\theta$ 가 가장 작은 것이다. 두 벡터가 이루는 각은 다음 식에 의해 구할 수 있다.

$$\cos \theta = \frac{\text{Assemble} \cdot \text{Bundle}}{\|\text{Assemble}\| * \|\text{Bundle}\|}$$

따라서 그림에서의 제거할 번들은 C이고, 만약 각의 크기가 같은 번들의 경우는 크기가 큰 것을 선택한다. 삽입할 번들은 C가 제거된 새로운 어셈블의 반대 방향에 가장 유사한 번들인 B'를 선택한다.

2) 랜덤방법에 의한 연산자



(그림 6) 반대 분면에 데이터가 없는 예 (Fig. 6) Data distribution of wrong case

(그림 6)은 어셈블이 3사분면에 속해 있다. 이것은 U235, U238의 두 원소 모두가 기준량에 미달되는 것을 의미하므로 직관적으로 1사분면에 있는 번들과 교환해야 한다는 것을 알 수 있다. 그러나 번들의 분포가 1사분면에 없는 경우가 있다. 이러한 경우 제거할 번들을 찾는 사분면의 선택에서 두번째로 사용할 수 있는 정보로는 어셈블의 U235와 U238의 상대적인 차이값이다. (그림 6)에서 얻을 수 있는 정보는  $|U235| > |U238|$ 로서 U235의 양이 상대적으로 더 모자람을 알 수 있다. 따라서 4사분면의 A', B', C'의 번들과 교환하는 것이 유리하다. 그러나 이러한 차선책으로써의 사분면의 선택은 예외의 값을 줄일 수 있는 가능성을 항상 갖고 있는 것은 아니다. 따라서 벡터 기법으로 탐색을 실패할 경우는 랜덤 방법에 의한 전통적인 전화 알고리즘의 연산자를 사용한다. 또한 랜덤 연산자의 특징은 국부 최소화의 극복 대안으로 사용될 수 있다.

3.5. 알고리즘의 구성

- step 1. 가능성 검사
- step 2. 전체 자료의 그룹화
- step 3. Assemble(t)의 초기화,  $t=0$ .
- step 4. 최대 반복 횟수나 오차가 허용치 내에 있으면 종료
- step 5.  $Bundle_R$ 과 반대 분면의  $Bundle_I$ 의 선택

step 6. 전체 자료에서 만약 Error를 줄일 수 있는  $Bundle_i$ 가 없으면

step 6-1.  $Bundle_i$ 를 random 선택

step 6-2. Assemble(t)을 mutation 시켜 Assemble(t')을 구성

step 6-3. Assemble(t) = Assemble(t')

step 7. 만약 step6의 조건이 아니라면

step 7-1. Assemble(t)을 mutation 시켜 Assemble(t')을 구성

step 7-2. Evaluation(t) > Evaluation(t')이면 Assemble(t) = Assemble(t')

step 9. step4로 반복

4. 실험 및 결과

4.1. 실험 과정

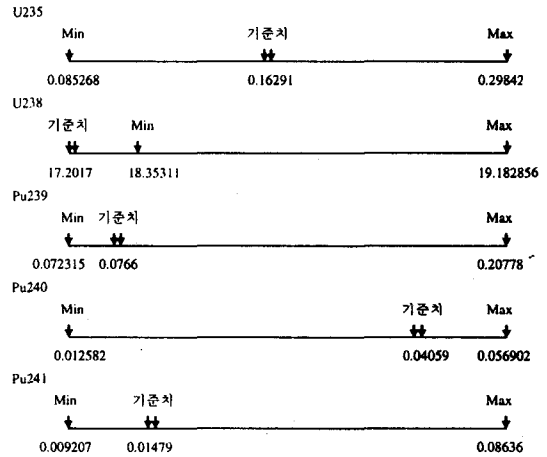
먼저 주어진 자료를 분석하여 해를 구하는 것이 가능한지 아니면 최소 오차의 번들을 구성할 것인지를 결정한다. 자료 분석 후에는 최적의 어셈블을 구성하기 위해 각 번들을 그룹으로 나눈다. 그룹을 나누는 방법은 각 번들의 형태를 계산하여 같은 형태의 번들을 하나의 그룹에 속하게 한다. 따라서 5가지 원소를 기준치에 대해 32개의 공간 그룹으로 나눈다. 그 후 각 번들의 형태를 보고 그룹화한다. 본 연구에서는 1985년 7월부터 1992년 9월까지의 영광 1호기와 1986년 9월부터 1993년 4월까지의 영광 2호기에서 나온 폐연료를 사용하였다. 전체 자료의 갯수는 847개이고, 번들의 평균 무게는 약 420Kg이며 구성해야 할 어셈블의 크기는 20,000Kg이다. 즉, 번들 47개로 구성된 어셈블을 생성한다. 20,000Kg의 어셈블을 생성할 때의 각 원소의 기준치는 다음과 같다.

<표 2> 원소의 기준 함량  
<Table 2> Unit of Elements

원소 무게	U235	U238	PU239	PU240	PU241
Bundle 기준치 (20.24483)	0.16291	17.2017	0.0766	0.04059	0.01479
Assemble 기준치 (20.000)	160.9398	16993.6719	75.6736	40.0991	14.6111

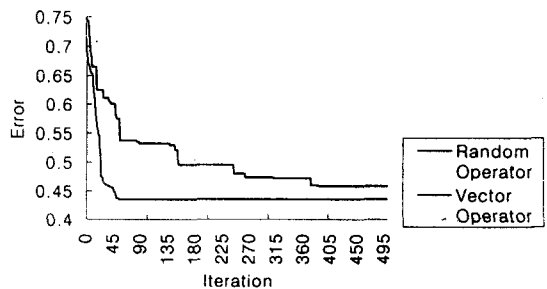
4.3. 실험 결과

전체 자료의 가능성 검사의 결과는 다음과 같다.



(그림 7) 가능성 검사의 예  
(Fig. 7) Capability Test

가능성 검사 결과 원소 U238의 경우, 해가 가능한 자료를 갖고 있지 못하며, PU239의 경우는 자료의 분포가 함량 초과분에 집중되어 있어 기본적인 오차를 포함하고 있는 것을 알 수 있다. 여기서는 탐색의 방법을 전통적인 진화 알고리즘의 랜덤 연산자만을 사용하는 방법과 본 연구에서 제안한 방법을 비교하여 보았다.



(그림 8) 랜덤과 벡터 연산자의 성능 비교  
(Fig. 8) Result of Experiment

본 논문의 실험 결과 (그림 8)에 의해서 알 수 있듯이 랜덤 연산자만을 사용한 경우와 벡터 연산자를 사

용한 경우 벡터 연산자가 57회의 탐색 반복한 후 해에 도달하고 있는 반면 랜덤 연산자의 경우 3500회 이상에서 해에 도달하게 되는 것을 알 수 있었다.

### 5. 결 론

본 연구에서는 문제 공간을 벡터 연산자를 사용하여 탐색함으로써 최적화 집합을 빠른 시간내에 찾는 진화 알고리즘을 제안 및 실험하였다.

본 논문의 페연료 재생 에에서 벡터의 방향 성분은 기준 원소 함량과의 오차 정보이고 벡터의 크기는 오차의 크기 정보에 해당한다. 이러한 휴리스틱의 사용은 5가지 원소의 크기가 변수로서 사용되고 또 이 원소가 독립적으로 사용될 수 없다는 제약 조건을 벡터의 방향과 크기로 표현한 것이 된다. 벡터 연산자는 탐색에서 사용되는 모든 정보가 벡터의 연산으로 표현 가능하게 되어 해로 탐색을 인도하는데 효과적인 뿐만 아니라 계산량과 구현에서도 이득을 얻게 된다. 이러한 방법은 0/1 knapsack 문제 등과 같은 고전적인 최적화 문제 및 응용된 다양한 문제에 적용할 수 있다.

### 참 고 문 헌

[1] Richard K. Belew and Lashon B. Booker, "Genetic Algorithm", Morgon Kaufmann Publishers, Inc., 1991.

[2] John R. Koza, "Genetic Programming", The MIT Press, 1992.

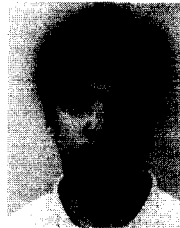
[3] David E. Goldberg, "Genetic Algorithms", Addison-Wesley, 1989.

[4] Robert Hinterding "Mapping, Ordering-Independent Gens And The Knapsack Problem", Proc. ICEC'94, 1994. p13-17.

[5] A. Olsen "Penalty-Functions And The Knapsack Problem", Proc. ICEC'94, 1994. p554-558.

[6] Sami Khuri, Martin Schutz and Joorg HeikKotter "Evolutionary Heuristics For The bin Packing Problem", Artificial Neural Nets and Genetic Algorithms, proc International Conference in Ales, France 1995. p285-288.

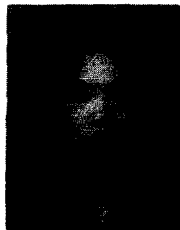
[7] Ellis Horowitz and Sartaj Sahni "Fundamentals of Computer Algorithms", Computer Science Press Inc. 1978.



### 안 종 일

1992년 국민대학교 기계설계학과 졸업(학사)  
 1994년 경희대학교 대학원 전자계산공학과(공학석사)  
 1994년~현재 경희대학교 대학원 전자계산공학과 박사과정 재학중

관심분야: 인공지능, 신경망 이론 등.



### 정 경 속

1995년 경희대학교 수학과 졸업(학사)  
 1995년~현재 경희대학교 대학원 전자계산공학과 석사과정 재학중

관심분야: 인공지능, 신경망 이론, 퍼지 이론 등



### 정 태 충

1980년 서울대학교 전자공학과 졸업(학사)  
 1982년 한국과학기술원 전자계산학과(공학석사)  
 1987년 한국과학기술원 전자계산학과(공학박사)  
 1987년 9월~1988년 3월 KIST

시스템 공학센터 선임연구원

1988년 3월~현재 경희대학교 전자계산공학과 교수

관심분야: 인공지능, 자연어 처리, 멀티미디어 등