

# 진화 알고리즘을 이용한 경수로 폐연료의 중수로 재사용을 위한 최적 조합 탐색에 관한 연구※

안종일\*·정경숙\*\*·정태충\*\*\*

## A Study for searching optimized combination of Spent light water reactor fuel to reuse as heavy water reactor fuel by using evolutionary algorithm

Ahn, Jongil Jung\*·Kyung Sook Chung\*\*·Tae Choong\*\*\*

### Abstract

These papers propose an evolutionary algorithm for re-using output of waste fuel of light water reactor system in nuclear power plants. Evolutionary algorithm is useful for optimization of the large space problem. The wastes contain several re-useable elements, and they should be carefully selected and blended to satisfy requirements as input material to the heavy water nuclear reactor system. This problem belongs to a 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 performs the feasible test and solution evaluation by using the vectorized data in problem. Finally, We compare the simulation results of using random operator and vector operator for such combinatorial optimization problems.

---

※ 본 연구는 학술진흥재단에서 지원하는 96학제간 연구 과제인 '경·중수로 연계 핵연료 제조기의 폐연료 최적 조합 시스템 개발에 관한 연구'의 지원에 의해 연구 되었음

\* 경희대학교 전자계산공학과 박사과정

\*\* 경희대 전자계산공학과 석사

\*\*\* 경희대 전자계산공학과 교수

## 요 약

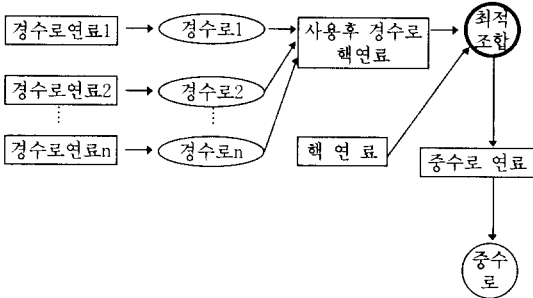
본 논문에서는 경수로 원자력 발전소의 사용 후 핵연료를 중수로의 핵연료로 재사용하기 위해 사용 후 경수로 핵연료의 최적 조합을 찾는데 진화 알고리즘(Evolutionary Algorithm)을 이용하여 해결해 보고자 한다. 진화 알고리즘은 대규모 문제 공간에서 최적화 문제를 해결하는데 적합한 알고리즘이다. (Keyword : 진화 알고리즘, 원자력 발전소, 경수로, 폐연료, 중수로, 벡터화 휴리스틱) 사용 후 경수로 핵연료에는 중수로에서 사용할 수 있는 유용한 원자들을 많이 포함하고 있지만 핵연료 봉마다 그 함량이 다양하고, 중수로 연료가 되기 위한 제약 조건 때문에 최적 조합 전략이 필요하다. 사용 후 핵연료의 조합 문제는 알고리즘 분야에서 대표적인 조합 최적화 문제인 0/1 Knapsack 문제와 같이 Non-Polynomial(NP) Complete 문제에 해당한다. 이러한 문제를 해결하기 위해서는 고전적인 진화 알고리즘의 전략에 기반하여 랜덤 연산자를 이용하되 평가 함수 값이 좋은 방향으로만 탐색을 수행하는 방법이 있으나 이것은 탐색의 효율면에서 좋지 않다. 따라서 본 연구에서는 벡터 연산자를 이용하여 최적의 해를 보다 빨리 얻을 수 있는 휴리스틱을 사용하는 방법을 제안한다. 본 논문에서는 경수로 핵연료 조합 문제 영역의 모든 지식을 벡터화하여 벡터의 연산만으로 가능성 검사, 해를 평가 하는 방법을 소개한다. 또한 벡터 휴리스틱이 고전적인 진화 알고리즘에 비해 어느 정도의 성능을 보이는지 비교한다.

## 1. 서 론

최근 원자력 발전소에 대한 관심은 환경 보호 문제와 관련되어 매우 비중 있게 다루어지고 있다. 이와 관련하여 사용 후 경수로 핵연료를 중수로에서 재사용하는 방안이 모색되고 있다. 경수로 핵연료로 사용되는 연료는 U235가 3.5~4.4%정도 농축된 것을 사용하고 있고 사용 후에도 타고 남은 U235와 새로 생긴 Pu를 합쳐 약 1.5%의 핵분열물질이 포함되어 있다. 이것은 천연우라늄보다 두 배 이상 높은 값으로 중수로에 다시 사용하기에 충분하다. 중수로는 중성자 이용 효율이 우수하여 천연우라늄 연료로 운전이 가능하기 때문이다.

경수로 사용 후 핵연료들은 여러 가지의 원자를 포함하고 있는데, 배출한 각각의 원자로와

생성된 시기에 따라 각 원자의 함량이 다양하다. 이들 각 폐연료들의 주요 관심 구성 원자는 U235, U238, Pu239, Pu240, Pu241의 5가지이다. 이 핵연료를 재사용하기 위해서는 각 구성 원자들이 단위 질량 당 필요로 하는 요구량을 만족해야 한다. 그러나 이 핵연료들은 각 원자의 함량이 일정치 않다. 또한 경수로의 폐연료 내에는 중수로에서 사용하는 필요한 원자의 함량을 충분히 갖고있지 못하다. 그러므로 사용 후 경수로 핵연료를 혼합하여 모든 구성 원자의 함량이 요구량에 도달하여야 재사용이 가능하다. 문제의 위치는 <그림 1>과 같으며, 본 연구의 목적은 여러 사용 후 경수로 핵연료들 중 어떤 것들을 혼합하는 것이 모든 원자의 조건을 만족할 수 있는지에 대해 탐색하는 것이다.



〈그림 1〉 문제 발생의 위치

이 문제는 전체 집합으로부터 사용 목적에 맞는 부분 집합을 조합하는 것으로 최적화 문제에 해당한다. 최적화 문제에는 대표적으로 0/1 Knapsack 문제를 꼽을 수 있다(Olsen,1994; Hinterding, 1994). 이와 같은 문제는 NP-complete에 속하는 문제로서 전통적인 탐색 알고리즘으로는 시간과 비용의 문제로 해결이 곤란하다(Hrowitz and Stataj, 1978). 특히, 각 집합의 원자가 분해될 수 없고, 그 경우의 수가 매우 많을 때는 문제의 복잡도는 더욱 더 커지게 된다. 이러한 문제에 대한 효과적인 해결 방법으로는 유전자 알고리즘(Genetic Algorithm)(Goldberg, 1989; Koza, 1989; Belew and Booker, 1991), 진화 알고리즘(Evolutionary Algorithm)등이 있다(Martin and Joorg, 1995).

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. 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 개선하

는 방법이 필요하다. 사용후 핵연료의 최적 조합을 찾는 문제의 경우 이미 알 수 있는 유용한 정보로는 전체 사용후 핵연료 집합에 포함되어 있는 원자들의 분포에 대한 정보이다.

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

2장에서는 경수로 원자력 발전소의 폐연료가 중수로 연료로서 사용 가능성에 대해 살펴보고 3장에서는 전체 시스템의 개략적인 소개를 한다. 4장에서는 본 논문에서 제안한 벡터 휴리스틱을 이용한 최적 조합의 탐색 방법을 소개하고 5장에서 일반적으로 사용되는 무작위 휴리스틱과의 효율성을 비교해 본다. 마지막으로 6장에서 최종 결론 및 향후 연구 방향에 대해 소개한다. Basic 4.0을 이용하여 개발하였다.

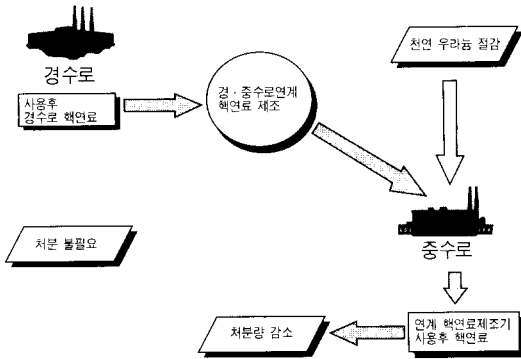
## 2. 경·중수로 핵연료 연계 타당성

우리 나라는 경수로와 중수로를 동시에 운전하고 있는 세계 유일의 원자력 발전 국가로서 경·중수로 핵연료 연계를 활용할 수 있는 모범적인 위치에 있다.

그러므로 이미 사용한 경수로 핵연료를 직접 처분하는 대신 적절한 방법으로 중수로에 재사용하면 중수로에 소요되는 천연 우라늄 연료가 30%이상 절감되고, 사용 후의 핵연료의 발생량도 전체적으로 1/3정도까지 감소될 수 있다.

또한 〈그림 2〉와 같이 경·중수로 핵연료 연계

공정은 사용 후의 경수로 핵연료에 함유된 민감한 핵 물질을 분리하지 않고 중수로 핵연료로 재가공하기 때문에 안전하고도 경제적이다. 그러므로 경·중수로 핵연료 연계 시스템의 개발이 요구된다. 경수로 핵연료는 U235의 함량이 3.5 ~ 4.4% 정도 농축되어 있으며 사용 후에도 타고 남은 U235와 원자핵 반응으로 인하여 새로 생긴 Pu를 합쳐 약 1.5%의 핵분열 물질이 포함되어 있다. 이것은 천연 우라늄보다 두 배 이상 높은 값으로, 중수로에 다시 사용하기에 충분하다. 그러므로 경·중수로 연계 핵연료를 만들어 중수로에 사용한다면 현재 사용되고 있는 천연 우라늄의 연소도 보다 두 배 이상의 연소도를 얻을 수 있다. 이것은 기존 중수로의 정상 출력으로도 연료를 두 배 이상 오래 태울 수 있다는 뜻이다.



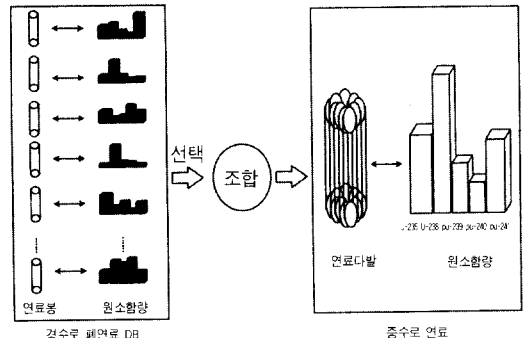
〈그림 2〉 경·중수로 핵연료 연계의 개략도

사용 후의 경수로 핵연료를 기존의 중수로에 다시 사용하려면 이에 적합한 중수로 연료로 재가공하여야 한다. 재가공된 것이 경·중수로 연계 핵연료로서 기존의 천연 또는 저농축 우라늄을 사용하는 중수로의 핵연료로 사용할 수 있다. 경·중수로 연계 핵연료의 재가공은 사용후 경수로 핵연료의 구조물과 피복관을 제거한 후 연료 분말을 산화, 환원으로 특성을 조절하여

성형 및 소결을 거쳐 연료봉과 다발로 가공하는 공정이다. 본 연구에서는 연료봉을 다발로 가공하는 공정에서 여러 개의 연료봉 중에서 어느 것을 선택하여 다발로 만들 것인지를 해결하고자 한다.

### 3. 경·중수로 핵연료 연계를 위한 시스템 구성

중수로의 핵연료로 사용후 경수로 핵연료를 재사용하기 위해서는 고려되어야 할 점이 몇 가지 있다. 한 번에 사용한 폐연료로 만들어진 연료봉을 번들(Bundle)이라 하며, 여러 가지 원자들로 구성되어 있다. 각 번들의 원자 함량은 생성 장소와 생성 일시에 따라 각각 다르다. 이 사용후 핵연료를 중수로의 핵연료로 재사용하기 위해서는 사용후 핵연료의 각 원자들이 단위 질량 당 필요로 하는 요구량을 함유하고 있어야 하지만, 하나의 번들로서는 그 요구 조건을 만족시키지 못한다. 따라서 여러 개의 적절한 번들을 선택한 후 그들의 조합인 다발(Assembly)로 가공하여 그 요구 조건을 만족해야 하는데, 어떻게 적절한 번들의 조합을 찾아 내느냐가 문제이다.



〈그림 3〉 문제의 정의

〈그림 3〉은 번들 내에 포함되어 있는 U235, U238, Pu239, Pu240, Pu241의 다섯 가지 원자의 함량과 이러한 다양한 번들들의 조합으로 어셈블리를 구성하였을 경우, 어셈블리 내에 포함되어 있는 원자들의 함량을 나타내고 있다. 본 연구에서는 어셈블리를 구성하였을 때, 어셈블리 내의 각 원자의 함량이 단위 질량 당 요구량에 가장 근사하게 접근하는 번들들을 찾고자 한다.

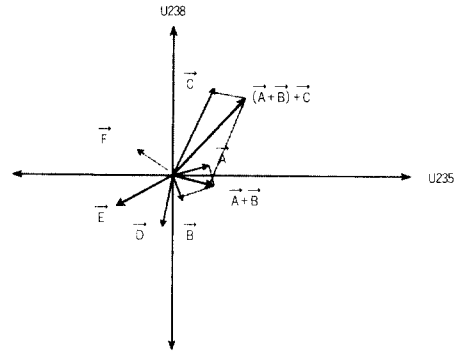
## 4. 문제 해결 방법

### 4-1. 문제 공간의 표현

문제의 공간은 5개의 원자, U235, U238, Pu239, Pu240, Pu241로 구성된 사용후 경수로 핵연료의 집합이다. 먼저 번들들을 섞어서 기준 함량에 맞추기 위하여 각 원자들을 벡터화 하였다. 여기서, 벡터화란 각각의 원자들을 순서쌍의 성분으로 나타내어 그 함량의 분포를 보는 것이다. 구성 원자가 5가지이므로 5차원의 벡터가 만들어 지는데, 각 원자의 요구량과 실제 포함하는 원자의 함량의 차이 값으로 번들의 형태를 결정한다. 이것은 각 원자마다의 기준량을 정규화하고 벡터의 원점이 각 원자에 대한 요구량의 상대적 미달량과 초과량을 표시하는 기준으로 결정하는 것에 해당한다. 이렇게 결정된 번들의 형태에 따라 자료들을 그룹화한다.

〈그림 4〉와 같은 경우 벡터(Vector) A, B, C인 번들로 어셈블리를 구성하면, 이 벡터의 크기가 어셈블리의 오차이다. 이 오차를 줄이기 위하여 벡터 A, B, C중에서 어셈블리 벡터와 방향 성분이 유사한 벡터를 제거한다. 여기서는 합 벡터와 유사한 벡터C를 제거한다. 대치시킬

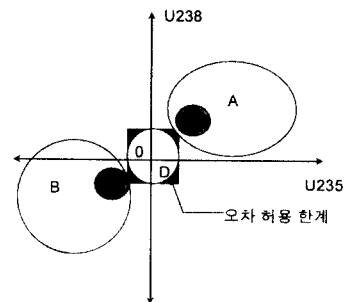
벡터는 어셈블리에 포함되어 있지 않은 벡터 D, E, F 중에서 선택한다. 이 때 벡터 C를 제거한 어셈블리는 벡터 A와 벡터 B의 합이므로 이 두 벡터의 합 벡터와 방향 성분이 반대인 벡터를 선택한다. 그러므로 벡터 D 또는 벡터 E를 선택하는 것보다 벡터 F를 선택하는 것이 어셈블리의 오차를 상대적으로 빨리 줄일 수 있다.



〈그림 4〉 원자들의 벡터화

### 4-2. 가능성 검사

주어진 자료를 분석한다. 기존 자료의 분포가 재활용할 어셈블리를 구성할 수 있는 원자 함량을 포함하는지 아닌지에 대한 분석이다. 자료 분석 후 적당한 조합으로 요구량을 만들 수 없다면 중수로에 적합한 요소를 많이 가진 새로운 핵연료를 추가해야 한다.



〈그림 5〉 완전해를 얻을 수 없는 자료의 분포 예

<그림 5>와 같이 D가 오차 허용 한계이고, 모든 자료의 분포가 A 위에 있거나 B위에 있으면 해를 구할 수 없다. 이 경우는 오차가 가장 작도록 C에 위치한 번들을 선택하여 어셈블리를 구성하였다 할지라도 기준량을 초과하게 된다. 따라서 가능성 검사 방법은 모든 자료의 집합에서 각 원자별로 값이 최소인 번들을 찾아 각각 어셈블리를 구성한 후 각 원자의 합량이 기준량을 초과한다면 해를 구할 수 없다고 판단한다. 반대로 최대값을 갖는 번들로 어셈블리를 구성하였을 경우 원자의 합량이 기준량보다 미달이라면 해를 구할 수 없다. 따라서 가능성 검사는 다음과 같은 수식으로 표현할 수 있다.

●해가 가능한 경우의 자료 분포

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

$i=1, \dots, 5$

●해가 없는 경우

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

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

- $D_i$  :  $i$ 번째 원자의 기준량
- $Y_i(x)$  :  $i$ 번째 원자의 실제량
- $Y_i(x)_k$  :  $i$ 번째 원자를 작은 값부터 차례로 정렬.
- $K$  : 어셈블리 내의 번들의 개수
- $N$  : 전체 자료의 개수

4-3. 연산자

1) 벡터화 기법을 이용한 연산자

연산자에서는 초기에 구성된 어셈블리로부터 제거할 번들과 삽입할 번들을 교환하여 평가 함수의 평가값을 최소화시키는 방법을 사용한다.

어셈블리에서 제거할 번들과 삽입할 번들의 선택 과정을 벡터 수식으로 정의하면 다음과 같다.

$$\overrightarrow{Assembly} = \overrightarrow{Bundle_1} + \overrightarrow{Bundle_2} + \dots + \overrightarrow{Bundle_K}$$

$$Error = \|\overrightarrow{Assembly}\|$$

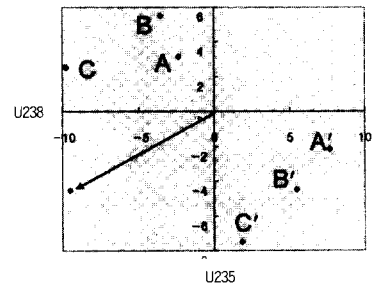
●번들의 제거와 삽입

$$\overrightarrow{Assembly} = \overrightarrow{Assembly - Bundle_K} + \dots + \overrightarrow{Bundle_1}$$

즉, 제거 번들인  $Bundle_K = \text{Min}_{i \rightarrow K} (|\overrightarrow{Assembly - Bundle_i}|)$ 으로 이 번들을 제거할 때 오차가 최소가 되며, 삽입 번들인  $Bundle_i = \text{Min}_{i \rightarrow N} (|\overrightarrow{Assembly - Bundle_i}|)$ 으로 이 번들의 삽입시 오

차가 최소가 된다. 그러므로  $\overrightarrow{Bundle_K}$ 는 어셈블리의 오차를 최대로 했던 번들이며,  $\overrightarrow{Bundle_i}$ 는 어셈블리에 속해 있지 않으면서 어셈블리 벡터와 반대 방향의 번들이다. 한 개의 어셈블리 내의 번들의 개수이며, 은폐연료 전체 자료의 개수이다.

2) 랜덤방법에 의한 연산자



<그림 6> 반대 분면에 데이터가 없는 예

<그림 6>은 어셈블리가 3사분면에 속해 있다. 이것은 U235, U238의 두 원자 모두가 기준량에 미달되는 것을 의미하므로 직관적을 1사분면에 있는 번들과 교환해야 한다는 것을 알 수 있다. 그러나 번들의 분포가 1사분면에 없는 경우가 있다. 이러한 경우 제거할 번들을 찾는 사분면

의 선택에서 두 번째로 사용할 수 있는 정보로는 어셈블리의 U235와 U238의 상대적인 차이 값이다. <그림 6>에서 얻을 수 있는 정보는 |U235| < |U238|로서 U238의 양이 상대적으로 더 모자람을 알 수 있다. 따라서 4사분면의 A, B, C의 번들과 교환하는 것이 유리하다. 그것은 4사분면에 있는 3개의 번들은 U238의 양이 초과된 상태이고 U235의 양은 미달의 상태를 의미하기 때문이다. 그러나 이러한 차선책으로써의 사분면의 선택은 오차 값을 줄일 수 있는 가능성을 항상 갖고 있는 것은 아니다. 따라서 벡터 기법으로 탐색을 실패할 경우는 랜덤방법에 의한 전통적인 진화 알고리즘의 연산자를 사용한다. 또한 랜덤 연산자의 특징은 극부 최소화의 극복 대안으로 사용될 수 있다.

#### 4-4. 평가 함수

본 연구에서 사용되어지는 어셈블리에 대한 평가 함수  $F(x)$ 는 다음과 같이 정의한다.

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

$Y_i(x)$  : 어셈블리에 포함된  $i$  번째의 단위 원자 함량

$D_i(x)$  : 요구되는  $i$  번째 단위 원자 함량

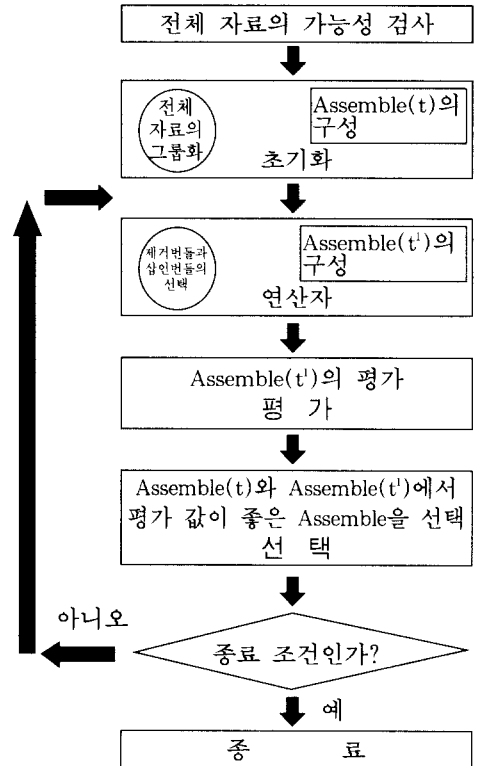
$E$  : 원자의 개수

$K$  : 번들의 개수

즉,  $D_i(x)$ 는 단위 질량 당 요구되는 원자 함량이고,  $Y_i(x)$ 는 어셈블리 전체 내에 포함되어진 각 원자들의 함량을 단위 질량 당 포함되어진 함량으로 바꾸어 계산한 것이다. 그리고 평가 함수  $F(x)$ 는  $Y_i(x)$ 와  $D_i(x)$ 의 오차를 계산한 것이다. 즉, 모든 원자를 벡터화 시켰으므로  $F(x)$ 는 오차의 크기를 나타내는 것이다. 따라서 최적화는 평가 함수  $F(x)$ 의 값이 가장 적게 되

도록 어셈블리를 구성하는 것이다.

#### 4-5. 알고리즘의 구성



<그림 7> 최적 조합 탐색 방법 흐름도

첫번째 수행될 작업은 전체 자료의 가능성 검사를 통해 해를 구하는 것이 가능한지 아니면 최소 오차의 번들을 구성할 것인지를 결정한다. 자료 분석 후에는 최적의 어셈블리를 구성하기 위해 각 번들을 그룹으로 나눈다. 어셈블리를 초기화하여 평가 값을 계산한 후 제거할  $\overrightarrow{Bundle_r}$ 과  $\overrightarrow{Bundle_l}$ 를 선택한다. Assembly(t)을 변이(Mutation)시켜 Assembly(t)을 구성한 후 평가 값이 더 좋은 Assembly을 새로운 Assembly로 하여 종료 조건을 만족할 때까지 반복한다. 대치시킬

## 5. 실험 및 결과

### 5-1. 실험 과정

먼저 주어진 자료를 분석하여 해를 구하는 것이 가능한지 아니면 최소 오차의 번들을 구성할 것인지를 결정한다. 자료 분석 후에는 최적의 어셈블리를 구성하기 위해 각 번들을 그룹으로 나눈다. 그룹을 나누는 방법은 각 번들의 형태를 계산하여 같은 형태의 번들을 하나의 그룹에 속하게 한다. 따라서 5가지 원자를 기준치에 대해 32개의 공간 그룹으로 나눈다. 그 후 각 번들 들의 형태를 보고 그룹화한다. 본 연구에서는 1985년7월부터 1992년 9월까지의 영광 1호기와 1986년 9월부터 1993년 4월까지의 경수로 폐연료를 사용하였다. 전체 자료의 개수는 847개이고, 번들의 평균 무게는 약 420kg이며 구성해야 할 어셈블리의 크기는 20,000kg이다. 즉, 번들 47개로 구성된 어셈블리를 생성한다. 20,000kg의 어셈블리를 생성할 때의 각 원자의 기준치는 다음과 같다.

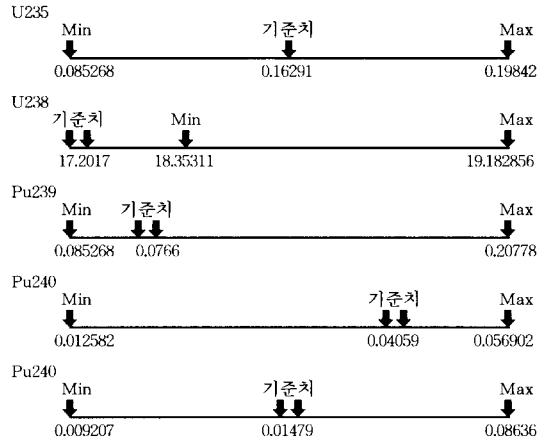
〈표 2〉 원자의 기준 함량

단위 : kg

무게 \ 원자	U235	U238	Pu239	Pu240	Pu241
Bundle기준치 (20.24483)	0.16291	17.2017	0.0766	0.04059	0.01479
Assembly기준치 (20,000)	160.9398	16993.6719	75.6736	40.0991	14.6111

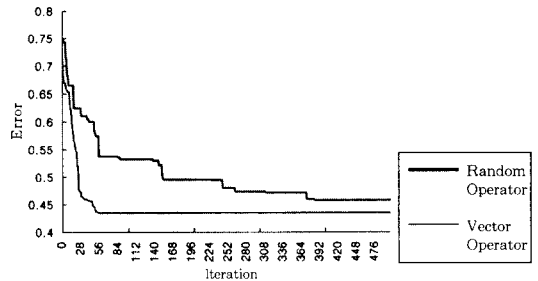
### 5-2. 실험 결과

전체 자료의 가능성 검사의 결과는 다음과 같다.



〈그림 8〉 가능성 검사의 예

가능성 검사 결과 원자 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 9〉 랜덤 과 벡터 연산자의 성능 비교

〈그림 9〉의 실험 결과 랜덤 연산자만을 사용한 경우와 벡터 연산자를 사용한 경우 벡터 연산자가 57회의 탐색 반복한 후 해에 도달하고 있는 반면 랜덤 연산자의 경우 3500회 이상에서 해에 도달하게 되는 것을 알 수 있었다.



## 6. 결 론

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

본 논문의 폐연료 재생 예에서 벡터의 방향 성분은 기준 원자 함량과의 오차 정보이고 벡터의 크기는 오차의 크기 정보에 해당한다. 이러한 휴리스틱의 사용은 5가지 원자의 크기가 변수로서 사용되고 또 이 원자가 독립적으로 사용될 수 없다는 제약 조건을 벡터의 방향과 크기로 표현한 것이 된다. 벡터 연산자는 탐색에서 사용되는 모든 정보가 벡터의 연산으로 표현 가능하게 되어 목표로 탐색을 인도하는데 효과적일 뿐만 아니라 계산량과 구현에서도 이득을 얻게 된다.

이러한 방법은 0/1 Knapsack문제 등과 같은 고전적인 최적화 문제 및 응용된 다양한 문제에 적용할 수 있다.

## 참고문헌

- 1) 비정기 간행물 : A. Olsen, Penalty-Functions And The Knapsack Problem, *Proc. ICEC94*, 1994.
- 2) 비정기 간행물 : David E. Goldberg , Genetic Algorithms , *Addison - Wesley*, 1989
- 3) 비정기 간행물 : Ellis Horowitz, Sartaj Sahnii Fundamentals of Computer Algorithms *Computer Science Press Inc.* 1978.
- 4) 비정기 간행물 : John R. Koza , Genetic Programming, *The MIT Press*, 1992, 643 -644
- 5) 비정기 간행물 : Richard K. Belew, Lashon B.Booker, Genetic Algorithm, *Morgon Kaufmann Publishers, Inc.* ,1991
- 6) 비정기 간행물 : Robert Hinterding Mapping, Ordering-Independent Gens And The Knapsack Problem, *Proc. ICEC94*,1994.
- 7) 비정기 간행물 : Sami Khuri,Martin Schutz and Joorg HeikKotter, Evolutionalry Heuristics For The bin Packing Problem *Artificuil Neural Nets and Genetic Algorithms, Proc International Conference in Ales*, France, 1995