

대안 부품을 고려한 다계층 시스템의 중복 할당을 위한 입자 군집 최적화

정일한[†]

울산과학기술대학교 안전및산업경영공학과

Particle Swarm Optimization for Redundancy Allocation of Multi-level System considering Alternative Units

Chung, Il Han[†]

Department of Safety & Industrial Management Engineering, Ulsan college

ABSTRACT

Purpose: The problem of optimizing redundancy allocation in multi-level systems is considered when each item in a multi-level system has alternative items with the same function. The number of redundancy of multi-level system is allocated to maximize the reliability of the system under path set and cost limitation constraints.

Methods: Based on cost limitation and path set constraints, a mathematical model is established to maximize system reliability. Particle swarm optimization is employed for redundant allocation and verified by numerical experiments.

Results: Comparing the particle swarm optimization method and the memetic algorithm for the 3 and 4 level systems, the particle swarm optimization method showed better performance for solution quality and search time. Particularly, the particle swarm optimization showed much less than the memetic algorithm for variation of results.

Conclusion: The proposed particle swarm optimization considerably shortens the time to search for a feasible solution in MRAP with path set constraints. PS optimization is expected to reduce search time and propose the better solution for various problems related to MRAP.

Key Words: Redundancy, Multi-level, Particle Swarm Optimization, Memetic Algorithm

● Received 29 August 2019, 1st revised 24 September, accepted 4 October 2019

† Corresponding Author(ihchung@uc.ac.kr)

© 2019, The Korean Society for Quality Management

This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-Commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

* 이 논문은 2019년 울산과학기술대학교 교내학술연구비 지원에 의해 수행됨.

1. 서론

시스템을 설계할 때에는 시스템에 요구되는 성능을 비롯해서 고장의 발생 정도, 원자재 비용, 크기 등 다양한 측면에서 검토가 이루어진다. 그 중 신뢰도는 시스템 고장이 발생하였을 경우 치명적인 결과를 양산하는 시스템에 특히 중요하게 다뤄지게 된다. 신뢰도는 신뢰도가 높은 부품을 사용하거나, 유사한 기능을 가진 부품을 중복해서 사용함으로써 향상시킬 수 있다. 동일한 기능을 가진 부품은 다양하게 있으며, 어떤 부품을 사용하느냐에 따라 시스템의 신뢰도, 비용, 크기 등이 달라질 수 있다. 중복구조에 대한 중복할당 최적화 문제는 NP-hard 문제로 Fyffe et al.(1968) 이후로 다양한 문제와 최적화 방법에 대하여 연구가 되었다. 중복할당 문제에서 가장 기본적으로 다루어진 문제는 m 개의 서브시스템을 가지는 시스템의 병렬-직렬 문제로 각 i 번째 서브시스템에 중복 개수 n_i 를 결정하는 문제인데, 이 때 시스템의 개발 비용, 부피, 무게 등이 제약으로 사용된다(Chung, 2015).

다계층 시스템은 부품들 간의 관계가 나무 형태로 상하 구분이 가능한 시스템을 나타낸다. 예를 들어, Figure 1의 시스템은 A, B, C 품목으로 구성이 되며, A는 A1, A2의 품목으로 구성되어 있음을 나타낸다.

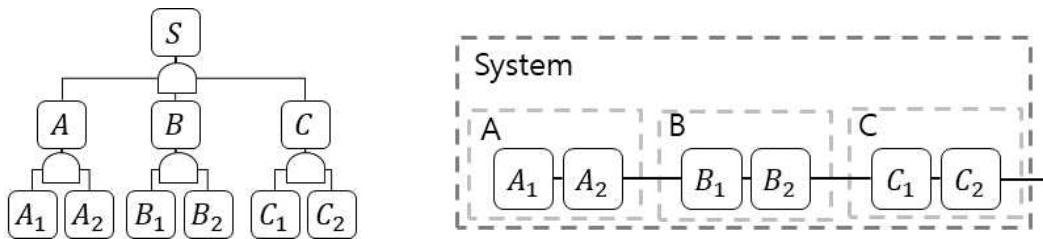


Figure 1. Structure of multi-level system

다계층 시스템의 중복할당 문제는 경로집합에서 단 하나의 품목만 중복이 가능한 다수준 중복 할당 문제(MRAP; multi-level redundancy allocation problem)와 두 개 이상의 품목이 중복이 가능한 다중 다수준 중복 할당 문제(MMRAP; multiple multi-level redundancy allocation problem), 중복 품목에 대해 재중복이 가능한 다수준 중복 할당 최적화 문제(MRAOP; multi-level redundancy allocation optimization problem)로 나뉜다.

Yun and Kim(2004)은 다계층 시스템에 대해 MRAP를 처음 소개하였으며 경로집합에서 단 하나의 품목만 중복이 가능한 경우에 대해 유전알고리즘을 활용하여 최적화 방법을 제시하였다. 경로집합이란 최하위 수준의 하나의 품목에서 최상위시스템으로 이어지는 품목의 집합을 의미하며 Figure 1에서는 S-A-A1, S-A-A2, S-B-B1 등이 있다(Yun et al, 2017).

Kumar et al.(2008)은 MRAOP를 제시하였으며 계층적 유전 알고리즘, 미미틱 알고리즘, 박테리아 진화 알고리즘 등을 활용한 최적화 방법이 연구되었다(Yun et al, 2017). MRAP와 MMRAP, MRAOP의 문제는 각 품목이 한 가지 종류인 경우에 최적 중복 할당 방법을 나타내며, 동일한 기능을 가진 다양한 종류의 품목에 대한 선택은 고려하지 않고 있다. Chung(2015)은 MRAP에서 각 품목의 종류가 다양하게 존재할 경우 최적의 부품 선정과 중복 할당 수량을 다루었다. 최적화를 위해 유전 알고리즘과 미미틱 알고리즘을 활용하였으며, 이에 대한 비교 결과를 제시하였다. MRAP는 경로집합에서 단 하나의 중복만 허용함으로 인해 유효해를 찾는 과정이 복잡하고 어려우며, 이로 인해 탐색시간이 길어지는 단점을 가지고 있다. 본 연구에서는 대안부품을 가지는 MRAP에서 입자 군집 최적화(particle swarm optimization) 방법을 다루며 실험을 통해 미미틱 알고리즘과의 성능을 비교하고자 한다.

2. 대안부품을 가진 다계층 시스템 모형

단, 본 연구에서는 모든 품목이 중복 설계시 고려 대상이 되지만, 경로집합에서 단 한 개의 품목만이 중복 대상으로 선정되어지는 MRAP 모형을 기반으로 하고 있으며, 그 외 중복 설계 모형을 위한 가정 및 기호는 아래와 같다 (Chung, 2015).

<가정>

- (1) 각 품목들은 직렬로 연결되어 있다.
- (2) 중복 품목 및 모든 품목들의 고장시간은 통계적으로 독립이다.
- (3) 각 품목들의 신뢰도는 알려져 있고 확정적이다.
- (4) 각 품목과 그 품목의 대안 품목들 중에 단지 하나의 품목만이 중복 가능하다.
- (5) 경로집합에서 단 한 개 수준에 품목만이 중복 대상으로 선정될 수 있다.

위에서 언급한 ‘품목’은 시스템, 모듈, 부품을 나타내는 공통된 의미로 사용되며, 시스템은 최상위 품목, 부품은 최하위 품목, 모듈은 중간 수준 품목을 의미한다. 대안 부품을 고려하기 위해 Figure 1에서 A, B, \dots, C_1, C_2 등을 품목의 ‘군’으로 표현하며, 군에 사용 가능한 품목이 2종류 있는 경우 $A^1, A^2, \dots, C_1^1, C_1^2$ 와 같이 표현한다.

<기호 및 정의>

- R_s : 전체 시스템 신뢰도
- i_f : i 군의 경로 집합
- m_i : 위치 i 에 대체 가능한 대안 부품들의 수
- N : 품목 군의 수
- l_c : 최하위 품목 군의 집합
- y_i^j : i 군의 j 번째 중 품목의 사용 여부 (0, 1)
- x_i^j : i 군의 j 번째 중 품목의 사용 개수
- p_p^k : p 군의 k 번째 중의 품목이 지역 최적화 대상 품목으로 선택될 확률
- R_p^k : p 군의 k 번째 중 품목의 신뢰도
- $C(x$: 중복할당 수를 고려하였을 경우 p 군의 k 번째 중으로 인한 소요 비용
- C_T : 시스템 비용한계

$$\text{Max } R_s = \prod_{i=1}^N \prod_{j=1}^{m_i} (1 - y_i^j (1 - R_i^j)^{x_i^j}) \tag{1}$$

subject to

$$\sum_{i=1}^N \sum_{j=1}^{m_i} C(x_i^j) \leq C_T, \quad C(x_i^j) = c_i^j \times x_i^j + \lambda_j^{x_i^j} \tag{2}$$

$$\sum_{h=1}^{m_i} \left\{ y_i^h + \sum_{k \in \{i_j\}} \sum_{j=1}^{m_k} y_k^j \right\} = 1 \quad (3)$$

$$y_i^j = 0 \text{ 또는 } 1$$

$$x_i^j \geq 1 \text{ 이고 정수이고, 식(3)에서 기호 } i \text{ 는 최하위 품목을 나타냄} \quad (4)$$

$$i = 1, 2, 3, \dots, N$$

본 연구에서는 비용한계가 주어졌을 때 시스템 신뢰도를 최대화하는 문제로 식(1)은 다계층 시스템에 중복할당 수가 결정되었을 때 시스템의 신뢰도를 계산하는 목적함수이며, 식(2)는 비용한계, 식(3)은 경로집합의 제약과 한 개의 군에서는 한 종의 품목만을 사용할 수 있다는 가정을 나타낸다.

3. 최적 중복할당을 위한 입자 군집 최적화

입자 군집 최적화는 자연에서 물고기, 새와 같은 군집 활동을 하는 행위에 착안하여 Kennedy and Eberhart(1995)에 의해 개발되었다. 유전알고리즘이나 개미알고리즘에서 중요한 개념으로 사용되는 교차, 돌연변이, 페르몬과 같은 연산을 수행하지 않으며, 입자들의 변화는 과거의 군집활동을 바탕으로 이루어진다.

x_i 와 v_i 는 i 번째 입자의 위치와 속도 벡터, ϵ_1 와 ϵ_2 는 0과 1사이 난수값, α 와 β 는 가중치, g^* 와 x_i^* 는 현재까지 전체 최적해와 i 번째 입자의 최적해를 나타낼 때, $t+1$ 세대에서의 i 번째 입자의 위치와 속도는 다음과 같이 구할 수 있다.

$$v_i^{t+1} = v_i^t + \alpha \epsilon_1 \odot [g^* - x_i^t] + \beta \epsilon_2 \odot [x_i^* - x_i^t]$$

$$x_i^{t+1} = x_i^t + v_i^{t+1}$$

입자 군집 최적화 방법은 연속적 비선형 함수를 최적화 하는데 많이 활용 되어지고 있다. 본 연구에서 다루고자 하는 중복 할당 최적화는 각 품목의 종류와 중복 수량을 결정해야 하는 것으로 최적화의 변수가 양의 정수의 형태로 나타난다. 이러한 문제에 대해 Yeh(2009)는 Yun et al.(2007)이 제시한 MMRAP에 대해 이단계 이산 입자 군집 최적화 방법을 제시하였으며, Yeh et al.(2016)에서는 GMRAP(general multi-level redundancy allocation problem)을 소개하고 Yeh(2009)에서 제시한 방법을 활용하여 최적화 방법을 제시하였다. 본 연구에서는 대안 품목을 고려한 MRAP에서 이산 입자 군집 최적화를 위해 해의 표현, 해의 상한값, 해의 평가, 최적화 절차를 제시하고 최적화 절차에서 $t+1$ 세대에서의 i 번째 입자의 위치와 속도 산출 방법은 Yeh(2009)가 제시한 방법을 인용하여 적용하고자 한다.

3.1 해의 표현

입자 군집 최적화 방법을 적용하기 위해서는 먼저 각 입자를 해의 형태로 표현을 해야 한다. 해의 표현은 경로제약에 따른 유효해 탐색과 다음 위치의 해를 탐색하는데 많은 영향을 주기 때문에 문제의 특성에 맞게 적절한 방법으로 표현해야 한다. 본 연구에서 해의 표현은 중복할당 수를 나타내는 x_i^j 의 배열로 나타낸다. x_i^j 가 0의 값을 가질 경우 해당 i 품목 군의 j 종은 중복대상에서 제외됨을 의미한다. 따라서 k 입자의 해 v_k 는 다음과 같이 나타낼 수 있

으며, 이러한 표현 방법은 각 입자의 유효해 여부를 해석하는데 용이하므로 해를 평가하는데 소요되는 시간을 줄일 수 있다.

$$v_k = [x_{k1}^1, x_{k1}^2, x_{k2}^1, \dots, x_{ki}^1, x_{ki}^{m_i}] \quad (5)$$

3.2 상한값

각 입자에서 각 품목별 중복할당 수는 세대가 반복됨에 따라 변경되어진다. 품목별 중복할당 수의 가능한 범위를 예측할 수 있다면, 탐색 범위를 좁힐 수 있으며 이로 인해 탐색 시간을 줄일 수 있다. 중복할당 수와 관련된 제약은 비용에 대한 제약으로 식(2)와 연관되어져 있다. 이에 대해 Yeh(2009)는 최적화 과정에서 허용 가능한 총 비용을 사용해서 각 품목에 대한 중복할당 상한을 적용하고 있다. 본 연구에서는 한 개의 품목에 다양한 군이 존재하고 각 군의 비용이 다르므로 인해 각 품목 군에서 종의 중복 상한값은 최하위 품목이 중복을 가지지 않을 때 최소값을 산출한다. 따라서, k 품목군에 p종의 중복수량 상한값 U_k^p 는 k 품목군 p종의 중복으로 인한 비용과 나머지 품목군에서 중복 수량이 1개일 때 최소 비용을 산출하여 비용한계 C_T 를 초과하지 않는 x_k^p 를 찾아 적용하게 된다.

$$U_k^p = x_k^p \text{ if } C_T \in [C(x_k^p - 1) + \sum_{i \in l_c, i \neq k_j} \text{Min}\{c_i^1 + \lambda_1, c_i^2 + \lambda_2, \dots, c_i^{m_i} + \lambda_{m_i}\}, C(x_k^p) + \sum_{i \in l_c, i \neq k_j} \text{Min}\{c_i^1 + \lambda_1, c_i^2 + \lambda_2, \dots, c_i^{m_i} + \lambda_{m_i}\}]$$

3.2 초기 모집단

본 연구에서는 임의의 생성 방법으로 초기해를 생성할 경우 제약식을 만족시키는 해를 만드는데, 상당한 시간이 소요됨으로 처음에는 최하위 품목 군에서 첫 번째 종을 사용하는 것으로 정한다. 또한, 각 품목 군의 첫 번째 종의 중복 개수는 적용이 가능한 중복 수의 상한을 고려해서 임의로 생성한다.

3.3 해의 평가

비용 한계가 주어졌을 때 시스템 신뢰도를 최대로 하는 중복 최적화 문제에서 해의 평가는 각 해의 신뢰도로 주어질 수 있다. 각 해는 식(8)과 같이 시스템 신뢰도에서 벌금을 제하여 계산하고 있으며, 벌금은 각 해가 비용 한계를 초과하였을 경우 초과한 비용과 비용한계의 비율에 벌금함수 비율 α 를 적용하여 산출하고 있다. 따라서, k 입자 v_k 에서 i군 품목의 j종의 중복 수가 x_{ki}^j 일 때 k 입자의 평가 결과는 다음과 같다.

$$eval(v_k) = \prod_{i=1}^N \prod_{j=1}^{m_i} (1 - y_{ki}^j (1 - R_i^j)^{x_{ki}^j}) - \alpha \sum_{r=1}^l \text{Min} \left\{ \left(\sum_{i=1}^N \sum_{j=0}^{m_i} C(x_i^j) - C_T \right) / C_T, 0 \right\} \quad (8)$$

3.4 이산 입자 군집 최적화

MRA에서 다음 세대의 입자를 결정하는데 있어 중복 대상 품목의 변경은 경로집합 제약으로 인해 유효하지 않은 입자를 생성할 수 있다. 유효하지 않는 입자는 임의적 위치 수정을 통해 유효 입자로 변경해야 하는 어려움과 이로

인해 많은 탐색 시간을 소요하게 된다. 본 연구에서는 경로 집합에 따라 유효하지 않는 입자가 발생하는 것을 방지하기 위해 중복 대상 품목과 수량을 결정할 때 항상 유효 입자를 유지 할 수 있도록 절차를 개발하여 제안한다. 해의 임의성을 향상시키기 위해 Step 1과 Step 2에서는 품목 군의 순서를 임의적으로 변경하여 중복대상 품목과 수량을 결정하며, Step 4에서는 경로집합의 제약에 따라 같은 군과 경로집합에 있는 품목의 중복할당 수량을 0으로 변경시키게 된다. 이렇게 모든 입자에 대해 Step 1~Step 5를 실시하여 t 세대의 입자를 구할 경우 모든 입자의 위치는 유효한 입자로 생성이 가능하게 된다.

Step 1 : 다음 세대의 위치를 결정하기 위해 입자 ν_i 에서 품목 군의 순서를 임의적으로 변경한다. 단, 같은 군의 품목의 순서는 연속되게 한다. 예를 들어, ν_i 는 다음과 같이 순서 변경이 될 수 있다.

$$\nu_i = [x_{i1}^1, x_{i1}^2, x_{i1}^3, x_{i2}^1, x_{i3}^1, x_{i4}^1, x_{i5}^1, x_{i5}^2] \rightarrow \nu'_i = [x_{i5}^1, x_{i5}^2, x_{i3}^1, x_{i4}^1, x_{i1}^1, x_{i2}^1, x_{i1}^3, x_{i1}^2]$$

Step 2 : ν'_i 에서 $x_{ij(t)}^k, x_{ij(t)}^k$ 값이 정해지지 않은 가장 첫 번째 품목을 선택한다.

Step 3 : 0에서 1사이의 임의의 $\rho_{ij(t)}^k$ 를 생성하고, $x_{ij(t)}^k$ 를 구한다.

$$C_w = c_w, C_p = C_w + c_p, C_g = C_p + c_g, \Delta_i = \frac{1 - C_g}{U_k + 1} \text{ 일 때,}$$

$$x_{ij(t)}^k = \begin{cases} x_{ij(t-1)}^k & \text{if } \rho_{ij(t)}^k \in [0, C_w) \\ p_{ij(t-1)}^k & \text{if } \rho_{ij(t)}^k \in [C_w, C_p) \\ g_i & \text{if } \rho_{ij(t)}^k \in [C_p, C_g) \\ x & \text{if } \rho_{ij(t)}^k \in [C_g + x \cdot \Delta_j, C_g + (x + 1) \cdot \Delta_j) \end{cases}$$

여기서, c_w, c_p, c_g 는 현재 해 유지, 입자 최적해 적용, 전체 최적해 적용 가중치를 나타내며, $p_{ij(t)}^k$ 와 g_i 는 i 입자에서 j 품목군 k종에서 t세대 최적해와 t세대까지 전체 최적해를 의미한다.

Step 4 : 경로집합 품목의 중복할당 수량을 0으로 변경한다.

Step 4-1 : $x_{ij(t)}^k > 0$ 면, i 입자의 j 군과 동일한 품목($x_{ij(t)}^1, x_{ij(t)}^2, \dots, x_{ij(t)}^{m_j}$)의 중복할당 수량을 0으로 변경한다.

Step 4-2 : i 입자의 j 군의 경로집합 품목의 중복할당 수량을 0으로 변경한다.

Step 5 : 모든 품목의 $x_{ij(t)}^k$ 값이 정해지면 종료하고, 그렇지 않으면 Step 2로 간다.

3.5 탐색정지 조건

해의 탐색 정지를 위하여 일반적으로 반복회수(세대 수) n을 지정하는 방법과 탐색 중 해의 개선 없이 동일한 최적해가 반복적으로 n회 나타났을 경우 정지하는 방법을 적용한다. 본 연구에서는 세대 수를 지정하여 탐색을 실시하고 해의 탐색을 정지시키게 한다.

4. 실험 및 분석

4.1 3 수준 시스템 실험

제안된 알고리즘의 실험을 위해서 Yun and Kim(2004)에서 적용한 시스템 구조를 활용하여 Chung(2015)에서 적용한 데이터를 사용한다. Figure 2는 계층이 3개인 시스템을 나타낸다. 시스템은 A, B, C 모듈로 구성되어 있으며, 모듈 A, B, C는 각각 3개, 2개, 2개의 하위 품목군으로 구성이 된다.

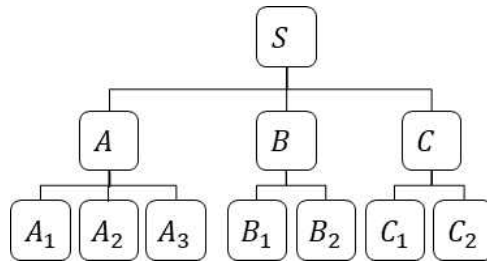


Figure 2. System structure in numerical example proposed at Yun and Kim(2004)

각 품목군은 Table 1과 같이 1~4종의 대안 부품의 적용이 가능한 것으로 하였다.

Table 1. Input data of three-level system

Unit	Parent unit	Reliability	Price	Additive cost
$S^1 \sim S^{12}$ (시스템)	-	0.424~0.327	72	2
A^1	$S^1 \sim S^3$	0.727	26	2
A^2	$S^4 \sim S^6$	0.689	20	2
A^3	$S^7 \sim S^9$	0.646	25	1
A^4	$S^{10} \sim S^{12}$	0.612	16	2
B^1	S^1, S^4, S^7	0.765	19	3
B^2	S^2, S^5, S^8	0.810	23	4
B^3	S^3, S^6, S^9	0.720	18	3
C	$S^1 \sim S^{12}$	0.720	21	2
A_1^1	A^1 or A^2	0.900	5	3
A_1^2	A^3 or A^4	0.800	4	2
A_2^1	A^1 or A^3	0.950	6	4
A_2^2	A^2 or A^4	0.900	4	4
A_3	$A^1 \sim A^4$	0.850	5	4
B_1	$B^1 \sim B^3$	0.900	6	4
B_2^1	B^1	0.850	7	4
B_2^2	B^2	0.900	9	5

Unit	Parent unit	Reliability	Price	Additive cost
B_2^3	B^3	0.800	6	3
C_1	C	0.900	8	3
C_2	C	0.800	7	4

Table 1의 데이터를 바탕으로 비용한계를 150에서 340까지 10씩 증가시키면서 미미틱 알고리즘과 입자 군집 알고리즘과의 비교를 실시하였다. Table 2는 각 비용한계마다 초기난수를 달리하여 10회 반복실험을 실시한 결과 가장 좋은 해와 분산 그리고 해의 탐색시간 평균을 나타내고 있다. 비용한계 20가지 경우에서 미미틱 알고리즘과 입자 군집 알고리즘의 가장 좋은 해는 all enumeration 방법의 최적해와 모두 동일한 결과를 나타내었으며, 분산의 경우 입자 군집 알고리즘이 미미틱 알고리즘보다 적게 나타남을 알 수 있다. 세 번째 열에 탐색시간의 경우 입자 군집 알고리즘이 미미틱 알고리즘에 비해 절반 정도의 시간이 소요됨을 나타내고 있다.

Table 2. Comparison result between MA and PS

Cost limit	All enumeration		Memetic algorithm				Particle Swarm			
	cost	Reliability	cost	Reliability	Variance	Run Time	cost	Reliability	Variance	Run Time
150	150	0.8342	150	0.8342	5.89E-04	239.1	150	0.8342	0	137.1
160	160	0.8620	160	0.8620	2.07E-05	296.0	160	0.8620	0	138.0
170	170	0.8811	170	0.8811	5.13E-06	274.9	170	0.8811	0	137.5
180	178	0.8923	178	0.8923	7.49E-05	275.5	178	0.8923	0	137.7
190	187	0.9086	187	0.9086	6.69E-05	284.9	187	0.9086	1.198E-05	135.9
200	200	0.9202	200	0.9202	1.25E-04	274.3	200	0.9202	0	136.0
210	210	0.9305	210	0.9305	5.59E-06	315.3	210	0.9305	1.303E-07	136.7
220	220	0.9346	220	0.9346	1.79E-05	252.8	220	0.9346	2.551E-06	136.2
230	230	0.9409	230	0.9409	7.41E-06	258.4	230	0.9409	0	136.0
240	240	0.9515	240	0.9515	1.60E-05	309.3	240	0.9515	0	135.9
250	240	0.9515	240	0.9515	4.24E-05	245.0	240	0.9515	0	135.5
260	251	0.9563	251	0.9563	6.56E-06	268.8	251	0.9563	0	135.1
270	269	0.9669	269	0.9669	2.02E-04	223.5	269	0.9669	0	136.4
280	280	0.9717	280	0.9717	6.18E-05	299.0	280	0.9717	0	137.7
290	280	0.9717	280	0.9717	4.26E-05	261.0	280	0.9717	0	137.3
300	296	0.9723	296	0.9723	0	243.6	296	0.9723	0	139.1
310	310	0.9755	310	0.9755	3.35E-06	295.4	310	0.9755	0	136.3
320	316	0.9781	316	0.9781	0	316.6	316	0.9781	0	135.1
330	316	0.9781	316	0.9781	9.63E-07	286.6	316	0.9781	0	135.5
340	335	0.9790	335	0.9790	1.93E-05	268.8	335	0.9790	0	136.7

Table 3은 Table 2의 각 비용한계별 중복할당 결과를 나타낸다. 비용 한계가 낮을 때 A 품목 군이 중복으로 선정되지 않았으나 비용이 증가함에 따라 A 품목 군이 선정되고 비용 변경에 따라 품목 군과 종이 변경되고 있음을 알 수 있다. A 품목 외 B, C 품목군도 비용한계가 증가함에 따라 모듈이 선택되어지는 경향이 나타났다. 이는 중복 수량이 증가할수록 하위 품목의 추가 비용이 급격히 증가하고 상대적으로 모듈의 경우 추가비용이 적게 소요되어 모듈이

중복 대상으로 선택되고 있음을 알 수 있다. 또한, 상대적으로 신뢰도가 낮은 A품목의 중복이 280 이후로 증가 또는 종류의 변경이 나타남을 알 수 있다.

Table 3. Results of redundancy allocation

Cost limit	Unit	<i>S</i>	<i>A</i>	<i>B</i>	<i>C</i>	<i>A</i> ₁	<i>A</i> ₂	<i>A</i> ₃	<i>B</i> ₁	<i>B</i> ₂	<i>C</i> ₁	<i>C</i> ₂
150	Class			1	1	1	2	1				
	Number			2	2	2	2	2				
160	Class			1		2	2	1			1	1
	Number			2		3	2	2			2	2
170	Class					1	2	1	1	1	1	1
	Number					2	2	2	2	2	2	2
180	Class				1	2	2	1	1	3		
	Number				3	3	2	2	2	2		
190	Class				1	2	2	1	1	1		
	Number				3	3	2	2	2	2		
200	Class		2		1	2	2	1	1	1		
	Number		3		3	3	3	2	2	2		
210	Class				1	2	2	1	1	3		
	Number				3	3	3	2	2	3		
220	Class				1	2	2	1	1	2		
	Number				3	4	3	2	2	2		
230	Class		1		1				1	3		
	Number		3		3				2	3		
240	Class		2		1				1	3		
	Number		4		3				2	3		
250	Class		2	1	1							
	Number		4	3	3							
260	Class		1		1				1	3		
	Number		3		4				2	3		
270	Class		2		1				1	3		
	Number		4		4				2	3		
280	Class		2	1	1							
	Number		4	3	4							
290	Class		2	1	1							
	Number		4	3	4							
300	Class		4	1	1							
	Number		5	3	4							
310	Class		3	1	1							
	Number		5	3	4							
320	Class		2	1	1							
	Number		5	3	4							
330	Class		2	1	1							
	Number		5	3	4							
340	Class		3	1	1							
	Number		6	3	4							

4.2 4수준 시스템 실험

Figure 3은 계층이 4개인 시스템으로 모듈과 부품이 15개로 구성이 되어 있다.

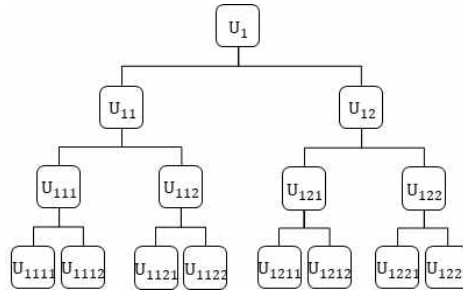


Figure 3. System structure in numerical example proposed at Kumar et al(2009)

Table 4는 Appendix에 Table 5의 데이터를 바탕으로 4 수준 시스템에 대한 실험 결과를 나타낸다. 15개의 비용 조건에서 입자 군집 최적화와 미미틱알고리즘을 비교한 결과 3가지 경우에서 입자 군집 최적화가 미미틱 보다 더욱 좋은 해(* 표시)를 나타내었으며, 1가지 경우에서 미미틱이 더욱 좋은 해를 나타내었다. 비용한계 550에서는 입자 군집 최적화의 경우 비용한계를 넘어서는 552의 결과가 산출되었다. 이에 대해 식(8)의 해의 평가시 벌금함수에 α 값(1에서 1.2로 변경)을 크게 변경함으로써 enumeration의 최적 결과와 동일한 결과를 얻을 수 있었다. 실험의 결과에서 시스템이 복잡해짐에 따라 미미틱 알고리즘의 경우 탐색시간이 상당히 증가됨을 볼 수 있으며, 입자 군자 최적화의 경우 3수준과 비슷하게 나타났다. 이는 시스템의 복잡성으로 인해 유효해를 탐색하는데 필요한 시간이 증가하여 이러한 시간의 차이가 발생함을 나타낸다.

Table 4. Comparison result between MA and PS

Cost limit	All enumeration		Memetic algorithm				Particle Swarm			
	cost	Reliability	cost	Reliability	Variance	Run Time	cost	Reliability	Variance	Run Time
200	200	0.6505	200	0.6505	4.23E-05	33850.0	200	0.6505	2.18E-06	134.4
250	249	0.7844	249	0.7844	1.70E-03	33472.6	249	0.7844	1.37E-32	134.8
300	300	0.8268	300	0.8268	6.61E-04	33444.1	300	0.8268	1.37E-32	138.2
350	344	0.8654	344	0.8654	2.39E-04	37271.2	344	0.8654	1.37E-32	137.7
400	400	0.8942	400	0.8942	3.20E-04	31973.9	400	0.8942	1.37E-32	134.2
450	445	0.9111	445	0.9111	1.68E-04	29470.7	445	0.9111	0	134.7
500	498	0.9289	498	0.9220	5.31E-05	27860.2	498	*0.9289	7.01E-07	133.4
550	530	0.9423	530	0.9423	2.13E-04	29952.5	552	0.9462	0	133.6
600	597	0.9534	589	0.9509	6.34E-05	29176.6	597	*0.9534	5.48E-32	133.6
650	642	0.9623	642	0.9623	5.43E-05	27490.2	642	0.9623	9.27E-06	137.4
700	695	0.9636	666	0.9573	1.26E-05	24421.4	695	*0.9636	1.06E-05	132.5
750	743	0.9674	743	0.9674	6.31E-05	24154.0	743	0.9674	1.37E-32	131.7
800	780	0.9721	780	0.9721	3.10E-05	24997.8	780	0.9721	0	136.7
850	827	0.9738	827	0.9738	3.78E-05	25184.1	833	▽0.9735	0	135.7
900	880	0.9752	833	0.9735	1.00E-05	24617.7	880	*0.9752	7.9E-07	131.8

5. 결론 및 추후 연구과제

본 연구에서는 입자 군집 최적화 방법을 활용하여 다계층 시스템의 최적 중복 할당 방법을 다루었다. 특히, 동일한 기능을 수행하는 대안 부품이 존재할 경우 비용과 신뢰도 측면을 고려하여 최적화 방법을 제시하였다. 3수준과 4수준의 실험 예제를 통해 입자 군집 최적화 방법에 대한 결과를 검증하였으며, 기존에 제시된 미미틱 알고리즘과의 비교 결과 시스템의 구조가 복잡해 질수록 해의 질과 탐색 시간 측면에서 입자 군집 최적화 방법이 더욱 좋은 결과를 보여주었다. 본 연구에서는 한 개의 군에서 한 개의 종만을 선택해서 최적화하는 방법을 다루었다. 향후에는 한 개의 군에서 서로 다른 종을 섞어서 중복을 하는 경우와 가용도 측면에서의 변화와 수리 정책에 대해 고려해 볼 수 있을 것 같다.

REFERENCES

- Chung, I. H. 2015. "Optimization of Redundancy Allocation in Multi Level System considering Alternative Units." *Journal of the Korean Society for Quality Management* 43(1):31-42.
- Fyffe, D. E., Hines, W. W., and Lee, N. K. 1968. "System reliability allocation and a computational algorithm." *IEEE Trans. Reliability* 17(2):64-69.
- Kennedy, J., and Eberhard, R. C. 1995. "Particle swarm optimization." In *Proceedings of IEEE international conference on neural networks*, Piscataway, NJ, USA : 1942-1948.
- Kumar, R., Izui, K., Yoshimura, M., and Nishiwaki, S. 2009. "Optimal multilevel redundancy allocation in series and series-parallel systems." *Computers and Industrial Engineering* 57:169-180.
- Yeh, W. C. 2009. "A two-stage discrete particle swarm optimization for the problem of multiple multi-level redundancy allocation in series systems." *Expert Systems with Applications* 36(5):9192-9200.
- Yeh, W. C., Luo, C. Y., Chung, Y. Y., and Lin, J. S. 2016. "Simplified swarm optimization with modular search for the general multi-level redundancy allocation problem in series-parallel systems." *2016 IEEE Congress on Evolutionary Computation (CEC) Evolutionary Computation (CEC):778-784.*
- Yun, W. Y., and Kim, J. W. 2004. "Multi-level redundancy optimization in series systems." *Computers and Industrial Engineering* 46:337-346.
- Yun, W. Y., Song, Y. M., and Kim, H. G. 2007. "Multiple multi-level redundancy allocation in series systems." *Reliability Engineering and System Safety* 92:308-313.
- Yun, W. Y., Chung, I. H., and Kim, J. W. 2017. "Multi-level redundancy allocation optimization problems." *Journal of the Korean Institute of Industrial Engineers* 43(2):135-146.