

# 계층구조 시스템에서의 최적 중복 구조 설계

김종운<sup>1</sup>, 윤원영<sup>1</sup>, 신주환<sup>2</sup>

## Abstract

Redundancy allocation problems have been considered at single-level systems and it may be the best policy in some specific situations, but not in general. With regards to reliability, it is most effective to allocate the lowest objects, because parallel-series systems are more reliable than series-parallel systems. However, the smaller and lower in the system an object is, the more time and accuracy are needed for duplicating it, and so, the cost can be decreased by using modular redundancy. Therefore, providing redundancy at high levels like as modules or subsystems, can be more economical than providing redundancy at low levels or duplicating components. In this paper, the problem in which redundancy is allocated at all level in a series system is addressed, a mixed integer nonlinear programming model is presented and genetic algorithm is proposed. An example illustrates the procedure.

## 1. 서론

품질과 신뢰성은 이제 가격이나 성능과 마찬가지로 상품 및 기업 이미지를 부각시키는 중요한 전략적인 경쟁력 변수가 되었다. 따라서 신뢰도 보증에 관한 기업의 노력도 더욱 적극적으로 되는 추세에서 품질보증을 단순히 최종 검사 선별로만 보는 기업은 줄어들고 품질과 신뢰성을 제조 절차의 중요한 요소로서 설계단계에서부터 이룰 고려하고자 하는 노력들이 이루어지고 있다. 설계 단계에서 고려할 수 있는 신뢰도 향상 방법 중 중복 부품의 사용은 시스템의 구조를 크게 변경시키지 않기 때문에 시스템의 기본 설계가 이루어지고 난 후 고려할 수 있는 신뢰도 향상 방법이다. 그러나 중복의 사용은 가격, 부피, 무게 등의 증가를 가져오기 때문에 이러한 상호관계를 고려하여 할당하여야 한다. 지금까지 다양한 모형의 정의에서부터 모형을 풀기 위한 많은 해법까지 많은 연구가 수행되어져 왔다[1-10]. 그러나 기존의 대부분의 연구는 최하위 부품만을 중복의 대상으로 고려해 왔다. 그 이유는 최하위 부품을 중복하는 것이 상위 부품을 중복하는 것보다 신뢰도 및 비용, 무게, 부피 측면에서 더 우수한 것으로 간주 되었기 때문이며, 이것은 특정 모형에서는 사실이다. 이것은 부품 단계에서의 중복이 시스템 단계에서의 중복보다 더 효과적이라는 원칙으로도 잘 알려져 있다. 그러나 Boland 와 EL-Newehi 는 동일하지 않은 예비품의 경우에는 그 원칙이 성립되지 않음을 보였다.

본 연구에서는 비용, 무게, 부피의 제한이 있는 경우에서의 최적 중복 구조 설계 문제를 다룬다. 이러한 문제에서 동일한 부품을 중복으로 사용한 경우에도 시스템 중복이

---

<sup>1</sup> 부산대학교 산업공학과

<sup>2</sup> 국방 과학 연구소

부품 중복보다 우수할 수 있다. 그 이유는 현대의 대부분의 시스템은 모듈로 구성되어 있기 때문에 몇 개의 부품으로 구성된 모듈을 중복하는 것이 각 부품을 중복하는 것보다 중복하기가 쉽고 적은 시간이 소요되기 때문이다. 일반적으로  $A_1$  과  $A_2$  두 부품으로 구성된 모듈의 가격은 각 부품의 가격에 상수의 추가 비용이 더해지는 것으로 간주되고 이 모듈의 중복 비용은 사용된 모듈 개수  $\times$  ( $A_1$  부품의 가격 +  $A_2$  부품의 가격 + 추가비) 같이 계산되는 것으로 간주되었다. 부피나 무게 측면에서의 소요량도 비용과 같은 방식으로 계산될 수 있으므로 이 경우는 모든 측면에서 부품 중복이 시스템 중복보다 우수한 것이다. 그러나 중복을 하는데 사용되는 기술적, 시간적 요소를 고려해보면 모듈의 중복이 부품 중복보다 더 경제적일 수 있다. 본 연구에서는 아래의 그림 1.B에서와 같이 모든 계층의 부품을 중복의 대상으로 고려하여 최적 중복 구조 설계 문제를 다룬다. 시스템의 구조는 직렬로 이루어진 계층구조를 대상으로 병렬 중복 설계 문제를 다루며 다음 장에서는 이에 대한 수리 모형과 유전 알고리즘이 제시된다.

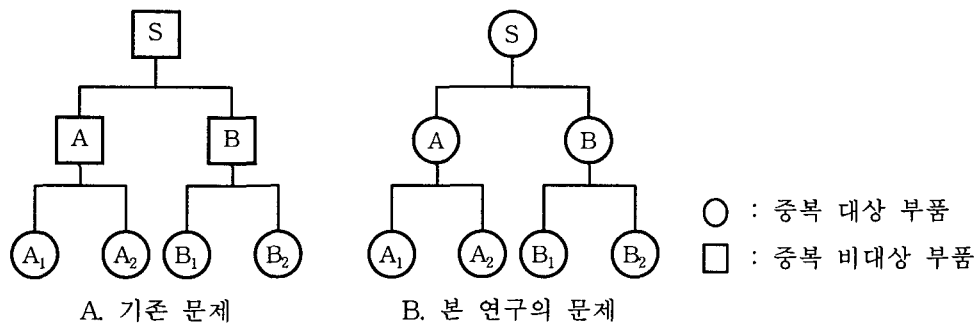


그림1. 기존 문제와 본 연구 문제의 계층구조

### 기호

- $j_f$ : 부품  $j$  의 부모 부품의 집합
- $R_s$ : 시스템 신뢰도
- $b_r$ : 자원  $r$  에 대한 최대 허용량
- $N$ : 부품의 총 수
- $n_r$ : 자원의 수
- $x_j$ : 부품  $j$  에 할당된 수
- $y_j$ : 0 또는 1
- $g_i(x_j)$ :  $x_j$  개의  $j$  부품이 사용되었을 때의 사용되는 자원  $i$  의 양

## 2. 최적 중복 구조 설계 모형

본 연구에서는 그림 1.B에서와 같이 모든 계층의 부품이 중복 대상으로 고려된다. 하지만 하나의 직계 라인에서는 하나의 부품만이 사용될 수 있다고 가정한다. 즉 <그림 1.B>에서는 직계라인이  $(A_1-A-S)$ ,  $(A_2-A-S)$ ,  $(B_1-B-S)$ ,  $(B_2-B-S)$  모두 4개가 존재하는데 각

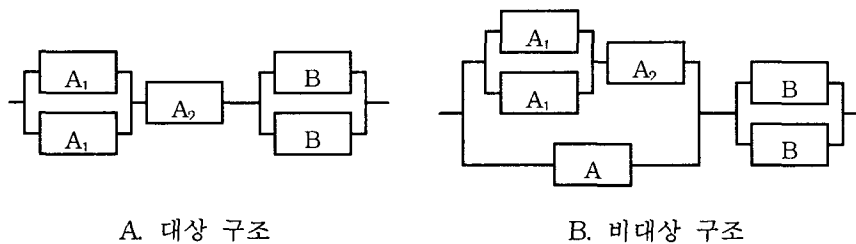
라인에는 하나의 부품만이 사용될 수 있다고 가정한다. <그림2>에서 <그림2.A>의 구조는 본 연구에서 사용 가능한 중복 구조의 대상이 되나 <그림2.B>와 같은 구조는 하나의 직계라인 (A<sub>1</sub>-A-S)안에 있는 A1 과 A 두 부품이 사용되었기 때문에 본 연구의 가정에 위배된다.

이러한 가정하에서 제한조건 하에서 최적 중복 설계 문제는 다음과 같이 모형화된다.

$$\text{최대화 } R_s = \prod_{j=1}^n (1 - y_j (1 - R_j)^{x_j})$$

$$\begin{aligned} \text{제약조건 } & \sum_{j=1}^n y_j g_{rj}(x_j) \leq b_r, \quad r=1,2,\dots,n_r \\ & y_l + \sum_{k \in \{j_j\}} y_k = 1, \quad l = \text{최하위 부품} \\ & y_j = 0 \text{ or } 1 \text{ for all } j \\ & \text{All } x \geq 1 \text{ 그리고 정수} \end{aligned}$$

목적함수는 시스템 신뢰도의 최대화이다. 두개의 주요 변수  $x_j$  와  $y_j$ 가 사용된다.  $x_j$ 는 부품  $j$  에 할당된 수를 나타내고  $y_j$ 는 부품  $j$  의 사용 여부를 나타낸다. 따라서 실제 사용된 부품  $j$  의 수는  $x_j \times y_j$ 가 된다.  $\sum_{j=1}^n y_j g_{rj}(x_j) \leq b_r$ 는 비용, 무게, 부피 등의 자원의 제한을 나타내기 위한 식이다. 비용 요소를 제외한  $g_{rj}(x_j)$ 들은 선형 함수로 가정될 수 있다. 본 연구에서는 중복 비용은 부품이 작을수록 그 단계가 하위일수록 증가하는 것을 가정하고 있다. 이러한 경우를 표현할 수 있는 비용함수로 본 연구에서는  $C(x) = cx + \lambda^x$ 이 사용된다. 즉 부품의 사용비용은 부품의 비용,  $cx$  와 추가 비용,  $\lambda^x$ 의 합이 된다. 추가 비용은 부품의 수가 많을수록 기하 급수적으로 늘어나고  $\lambda$ 는 부품이 하위에 있을수록 커지게 된다.



<그림 2> 대상 구조와 비 대상 구조의 신뢰도 블록 그림

### 3. 유전 알고리즘

#### 3.1 염색체 표현

하나의 유전자는 할당된 부품의 수,  $m_{ki}$  와 사용여부,  $\alpha_{kj}$  의 쌍으로 표현된다. 첨자  $k$ 는 유전자가 포함되는 염색체에 대한 인덱스이고 첨자  $i$  부품에 대한 인덱스이다. 따라서  $n$  개의 부품으로 이루어진 시스템에 대한 염색체는 아래와 같이 표현된다.

$$v_k = [(m_{k1}, \alpha_{k1}) (m_{k2}, \alpha_{k2}) \dots (m_{kn}, \alpha_{kn})]$$

### 3.2 적합도 평가

해의 적합도 평가하기 위해 목적함수인 시스템의 신뢰도와 제한조건의 위배성의 합으로 평가한다. 적합도 함수(fitness function)는 아래와 같다.

$$eval(v_k) = \prod_{j=1}^N (1 - \alpha_{kj} (1 - R_j)^{m_{kj}}) + \sum_{r=1}^{n_r} \text{Min} \left\{ \left( \sum_{j=1}^N \alpha_{kj} g_{rj} (m_{kj}) - b_r \right) / b_r, 0 \right\}$$

### 3.3 교배 연산자

교배 연산자는 두 부모해의 임의의 위치에서 유전자를 상호 교환하여 새로운 자식 유전자를 생성해 나가는 과정이다. 본 문제에서는 적합도 평가에서 한 직계라인에서는 하나의 부품만이 사용되어야 한다는 제약조건이 포함되지 않았다. 따라서 교배를 할 때 이러한 원칙을 지키면서 교배를 한다. 즉 두 염색체에서 선택된 두개의 유전자 정보, 즉 부품의 정보를 서로 일대일로 교환하는 것이 아니라 상위 단계에 있는 부품에서 그 하위 부품의 정보를 모두 교환하는 것이다. 이렇게 함으로써 하나의 직계라인에서 한 부품만이 사용되어야 한다는 제약조건은 만족된다.

### 3.4 돌연변이 연산자

알고리즘이 국부해로 빠지는 것을 방지하기 위해 한 염색체의 임의의 유전자를 선택하여 그 정보를 변경한다. 이 때 교배 연산자와 마찬가지로 한 직계라인에서는 하나의 부품만 사용되어야 한다는 제약조건은 만족시키면서 변이시킨다. 즉 부품의 수,  $m_{ki}$ 의 변이는 이러한 제약과 아무런 영향이 없으나 사용여부,  $\alpha_{ki}$ 의 변이를 할 때 그 직계라인도 함께 변화해야 하고 직계라인에서의 변화는 또 다른 직계라인에서의 사용 부품 단일성을 보장해 주어야 한다. 이러한 보장의 절차는 발견적 기법이 연산자안에 포함되어 조정한다.

## 4. 수치 예제

직렬 시스템에 대한 정보가 아래의 표1과 같이 주어졌다. 비용함수는  $C(x) = cx + \lambda^x$  이고 비용의 제한은 \$150.00인 경우 모형은 아래와 같다.

$$\text{Max} \quad (1 - y_1(1 - R_1)^{x_1}) \times (1 - y_{11}(1 - R_{11})^{x_{11}}) \times (1 - y_{12}(1 - R_{12})^{x_{12}}) \times (1 - y_{111}(1 - R_{111})^{x_{111}}) \times (1 - y_{112}(1 - R_{112})^{x_{112}}) \\ \times (1 - y_{121}(1 - R_{121})^{x_{121}}) \times (1 - y_{121}(1 - R_{121})^{x_{121}})$$

$$\text{Subject to} \quad y_1(31x_1 + 2^{x_1}) + y_{11}(16x_{11} + 2^{x_{11}}) + y_{12}(13x_{12} + 2^{x_{12}}) + y_{111}(6x_{111} + 3^{x_{111}}) \\ + y_{112}(5x_{112} + 4^{x_{112}}) + y_{121}(5x_{121} + 3^{x_{121}}) + y_{122}(4x_{122} + 3^{x_{122}}) \leq 150$$

$$y_1+y_{11}+y_{111}=1 \quad y_1+y_{11}+y_{112}=1 \quad y_1+y_{12}+y_{121}=1 \quad y_1+y_{12}+y_{122}=1$$

$$y_j = 0 \text{ or } 1 \text{ for all } j$$

$$\text{All } x \geq 1 \text{ and integer}$$

Table 1. 예제 데이터

부품	부모부품	신뢰도	가격모수	추가비용모수
$x$		0.45	31	2
$x_1$	$x$	0.72	16	2
$x_2$	$x$	0.63	13	2
$x_{11}$	$x_1$	0.9	6	3
$x_{12}$	$x_1$	0.8	5	4
$x_{21}$	$x_2$	0.9	5	3
$x_{22}$	$x_2$	0.7	4	3

3장에서 제시된 유전 알고리즘에 의한 최종해는  $[(4,0)(4,1)(4,1)(5,0)(1,0)(3,0)(3,0)]$ 으로 시스템 신뢰도가 0.97523 이고 총 비용은 \$148.00이 된다. 최하위 부품만을 고려했을 경우는 시스템 신뢰도가 0.9253 이 되고 비용은 \$128.00이 된다. 이 비용에 맞추어 비용 제한을 \$128.00으로 두고 계층 구조 문제를 다시 풀어보면, 시스템 신뢰도가 0.9597 , 총 비용은 \$124.00가 된다. 따라서 본 예제에서는 모듈 중복을 고려하는 것이 최하위 부품만을 고려하는 경우보다 우수하다.

## 5. 결론

본 연구에서는 계층 구조로 이루어진 직렬 시스템의 최적 중복 구조 설계 문제를 다루었다. 중복 비용이 부품이 작을수록 그리고 그 단계가 하위일수록 많은 추가 비용이 드는 경우에 시스템 중복이 부품 중복보다 우수할 수 있음을 보였고 이러한 문제를 혼합 비 선형 정수 계획법으로 모델링하였다. 또한 이 문제를 풀기위해 발견적 기법이 첨가된 유전 알고리즘을 제시하였다. 추후로 하나의 직계 라인에는 한 부품만 사용 가능하다는 제약이 없는 문제에 대한 연구가 필요할 것으로 생각된다.

### 참고문헌

1. Aggawal, K.K. Redundancy optimization in general systems. IEEE Trans. Reliab., 1976, 25 (5): 330-332
2. Aggawal, K.K, J.S. Gupta, and K.B. Misra. A new heuristic criterion for solving a redundancy optimization. IEEE Trans. Reliab., 1975, 24 (apr): 86-87
3. Boland P and EL-Neweihi E, Component redundancy vs system redundancy in the hazard rate ordering, IEEE Trans. Reliab., 1995, 44 (4):614-619
4. Coit D.W and Smith A.E., Redundancy allocation to maximize a lower percentile of the system time to failure distribution, IEEE Trans. Reliab., 47 (1): 79-87

5. Gopal K, K.K. Aggarwal and J.S. Gupta. An improved algorithm for reliability optimization. IEEE Trans. Reliab., 1978, 27 (5): 325-328
6. Kuo, W, C.L. Hwang, and F.A. Tillman, A note on heuristic methods in optimal system reliability. IEEE Trans. Reliab., 1978, 27 (5): 320-324.
7. Li J, A bound heuristic algorithm for solving reliability redundancy optimization, Microelectron. Reliab., 1996, 36 (3),: 335-339
8. Misra, K.B. Reliability optimization of a series-parallel system. IEEE Trans. Reliab., 1972, 21: 230-238
9. Misra, K.B. A simple approach for constrained redundancy optimization problems. IEEE Trans. Reliab., 1972, 21 (1): 30-34
10. Woodhouse, C.F. Optimal redundancy allocation by dynamic programming. IEEE Trans. Reliab., 1972, 21 (1): 60-62