

# ( $m,n$ )중 연속( $r,s$ ):F시스템의 최적화 연구

## (A Study on Optimization for Connected- $(r,s)$ -out-of- $(m,n)$ :F System )

이상현, 강영태

국방대학교 운영분석학과, email : leesangh@kndu.ac.kr

### Abstract

This paper is about optimizing preventive maintenance period of connected  $(r,s)$  out of  $(m,n)$  : F lattice system that one of multi-component system,  $(m,n)$  matrix failure of whole system is occurrence when parts that belong in  $(r,s)$  matrix part procession of parts arranged with procession are break-down all.

The preventive maintenance about system is very important viewing from system reliability and operational expense viewpoint. Preventive maintenance that misses a time calls big loss by system failure and expense of frequent full equipment is paid excessively in preventive maintenance itself but expense is paid much in preventive maintenance itself and whole expense escalation can be achieved preferably.

Through this research, reliability model is constructed that do expense by smallest under full equipment policy chosen through comparison of each full equipment policy and preventive maintenance expense full equipment cycle and  $r, s$  value are made using simulated annealing algorithm and simulated annealing algorithm that converge fast in multi-component system certified most suitable to optimization decision

### 1.서론

근래에 고도의 신뢰도를 갖는 체계나 부품을 만들어야 할 중요성이 점차로 증대하고 있다. 경제적인 관점에서 보면 고도의 신뢰도는 총 비용을 줄이기 위해 필수적이다. 따라서 어떠한 시스템에서 그 시스템의 신뢰도를 높이기 위한 것은 당연하며 가장 많이 이용되는 방법이 부품을 중복으로 설계하는 것이다.

부품 중복 설계방법은 직렬 시스템과 병렬 시스템으로 나눌 수 있고 두 경우를 포괄하는 시스템이  $k$ -out-of- $n$  시스템이다.  $k$ -out-of- $n$  시스템은  $n$ 개의 부품 중에서  $k$ 개만 작동하면 시스템이 정상 작동하는 구조로 식(1)과 같이 신뢰도가 계산된다.  $k=n$ 인 경우 직렬 구조가 되고  $k=1$ 이면 병렬 구조가 된다.

$$R_s = \sum_{r=0}^{n-k} \binom{n}{r} P^{n-r} (1-P)^r \quad (1)$$

직렬 구조에 연속 고장의 개념을 추가하여 Consecutive  $k$ -out-of- $n$ :F 시스템이 Kontoleon[13]에 의해 처음 제시되었다. 이 시스템은  $n$ 개의

부품으로 이루어진 선형 혹은 원형 시스템으로 연속적인  $k$ 개의 부품이 고장나면 시스템이 고장나게 되는 구조를 말한다. 예를 들어 가로등 시스템을 들 수 있는데 가로등 하나가 비출 수 있는 거리가 양쪽에 인접한 가로등까지 인 경우 3개의 연속적인 가로등이 고장 나면 전체 시스템이 고장 나게 되는 것이다.

이러한 선형의 개념을 확장하여 평면의 개념에서 생각 할 수 있는데 이것이  $(m,n)$ 중-연속 $(r,s)$ :F시스템이다. 이 시스템은 Connected  $(r,s)$ -out-of- $(m,n)$ : F 시스템이라고 표현하기도 하며 Consecutive  $k$ -out-of- $n$ :F시스템의 2차원 형태의 시스템이다. 이 시스템은  $(m,n)$ 행렬과 같이 배열되어 있는 부품 중  $(r,s)$ 의 부분 행렬에 속한 부품이 모두 고장일 경우 시스템의 고장으로 정의 된다. 이러한 시스템은 근래 들어 많은 분야에서 활용되고 있는데 예를 들어 군사적 측면에서 일정 지역의 적의 침입을 감시하기 위한 감시 카메라 시스템과 각종 질환을 진단하는 X-ray와 같은 장비는 대표적인 사례라고 할 수 있다.

$(m,n)$ 중-연속 $(r,s)$ :F시스템의 신뢰도를 결정하고 단위 시간당 기대 비용을 최소로 하는 정비 기간을 결정하며 또한 각 부품의 적절한 용량을 결정하는 것은 정비 정책에 있어 매우 중요한 문제이다. 하지만 이러한 중복 설계 시스템의 신뢰도는 다양한 노력에도 불구하고 복잡한 순환식의 제시, 상한 및 하한 값의 제시 등으로 시스템의 최적화를 위한 적용에는 제한에 있으며 기존의 유전자 알고리즘을 이용한 정비모형 최적화는 알고리즘의 수행시간을 과도하게 요구하여 활용성이 제한되는 문제를 가지고 있다. 따라서 본 연구에서는 단위 시간당 기대 비용을 최소로 하는 정비주기와 기대비용을 결정하고 시스템을 최적화하여 기존의 유전자 알고리즘보다 빠른 탐색시간 내에 좀 더 효율적으로  $(m,n)$ 중-연속 $(r,s)$ :F시스템의 최적화 문제를 해결 할 수 있도록 하는 시뮬레이티드 어닐링(Simulated Annealing) 알고리즘을 제시한다. 이 방법은 초기해(정비주기)와 인접해를 비교하면서 Metropolis 기준에 의해 지역 최적해에 빠지지 않도록 하면서 목

적함수인 기대비용이 궁극적으로 최소값으로 수렴할 수 있도록 하는 것으로 본 논문의 문제와 같이 목적함수가 시뮬레이션에 의해 복잡하게 결정되는 경우 유전자 알고리즘이 적합도 평가과정에서 느린 탐색시간을 갖는 문제를 보완하여 해의 범위가 확대 될수록 효과적인 성능을 보임을 입증하였다.

## 2. 기존연구

선형/원형  $(m,n)$ 중-연속 $(r,s)$ :F시스템은 Salvia & Lsher[22]에 의해 처음으로 소개되었으며 Boehme *et al.*[8]는 이 시스템의 신뢰도 계산 문제를 연구하였다.

$(m,n)$ 중-연속 $(r,s)$ :F 시스템의 신뢰도를 계산하는 문제는 Boehme *et al.*[8]에 의해 연구되기 시작하여 특별한 경우에만 신뢰도에 대한 정확한 식을 구할 수 있음을 보였다. Zuo[25]는 일반적인 경우에도 적용할 수 있는 SDP(Sum of Disjoint Products) 방법을 제안하였으나 그 절차가 너무 복잡하여 활용에 제한이 된다. 그 후 Yamamoto와 Miyakawa[23]는 계산 시간을  $O(s^{m-r} \cdot m^2 \cdot r \cdot n)$ 로 표현한 시스템 신뢰도에 대한 순환식인 YM 알고리즘을 식(2)와 같이 제안하였다.

$$R(r,s,(m,j);h_i, i \in I(m);[p_{i,j}]) \quad (2)$$

$$\left\{ \begin{array}{l} R(r,s,m[p_{i,j}]) \cdot R(r,s,(m,j-1);s,s,\dots,s[p_{i,j}]) \\ + \sum_{\delta \in \Theta(r,m)} R(r,s,(m,j-1);s[p_{i,j}]) \cdot F(r,m,\delta), \\ \text{for } j \geq h \geq 1 \\ 1, \text{ for } h > j \geq 0 \\ 0, \text{ for } h = 0 \end{array} \right.$$

$$\varepsilon_i = \begin{cases} h_i - 1, & \text{if } \delta_i = 1 \\ s, & \text{if } \delta_i = 0 \end{cases}$$

$$R(r,s,(m,j);[p_{i,j}]) = R(r,s,(m,j);s,s,\dots,s[p_{i,j}]) \quad \text{for } n \geq j \geq 1$$

또한 Malinowski와 Preuss[17]는  $(m,n)$ 중-연속 $(r,s)$ : F 시스템 중 부품간 고장이 상호 독

립적인 시스템에서 신뢰도 상한과 하한 한계선을 이용한 신뢰도 계산 방법을 식(3)과 같이 제안하였다.

$$\max \left[ \begin{array}{l} R_c(s, n; 1-(1-p)^r)^{m-r+1}, \\ R_L(r, m; 1-(1-p)^s)^n \end{array} \right] \quad (3)$$

$$\leq R_c((r, s); (m, n); p) \leq$$

$$\min \left[ \begin{array}{l} R_c(s, n; 1-(1-p)^r)^{gilb(m/r)}, \\ R_L(r, m; 1-(1-p)^s)^{gilb(n/s)} \end{array} \right]$$

위와 같이 이 시스템의 신뢰도는 계산식으로 구하기가 매우 어려우므로 윤원영[3]은 이 시스템의 정비모형을 최적화하기 위해 몬테카를로 시뮬레이션과 유전자 알고리즘을 적용하여 최적 정비주기를 제안하였다.

또한 Consecutive  $k$ -out-of- $n$ :F 시스템에 있어서 Derman *et al.*[10]는 조합방법(Combinatorial Method)을 이용하여 신뢰도를 식(4)와 같이 구하였으며 Salvia[21]는 시스템 신뢰도의 상한치와 하한치를 식(5)과 같이 구하였다.

$$R(n, k) = \sum_{j=0}^n M(j, n-j+1, k-1) p^{n-j} q^j \quad (4)$$

$$\begin{aligned} L(p, k, n) &= 1 - (n-k+1)q^k \\ U(p, k, n) &= 1 - (n-k+1)p^{n-k}q^k \end{aligned} \quad (5)$$

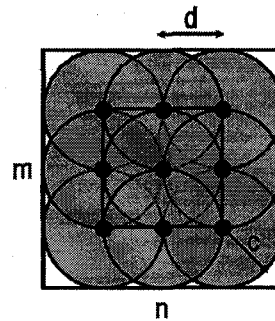
Fu와 Hu[11]는 한 부품의 고장이  $k-1$ 개의 직전의 부품 상태에 의존하는 경우에 Markov chain을 이용하여 정적 시스템 신뢰도를 계산하였고 Papastavridis와 Lambiris[19]는 한 부품의 고장이 확률이 바로 앞의 부품 상태에만 의존하는 경우에 대한 연구로 확장하였다.

### 3. 시스템 모형 및 정비 정책

#### 3.1 $(m, n)$ 중-연속 $(r, s)$ :F 시스템

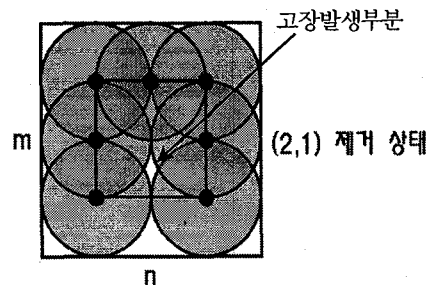
선형  $(m, n)$ 중-연속 $(r, s)$ :F 시스템은 각 요소들이 <그림 1>처럼  $m$ 개의 행이 각각  $n$ 개의 요소들을 포함하고 있는 시스템으로 연속된  $r$

개 행과  $s$ 개 열의 고장에 의해 전체 시스템이 고장나는 시스템이다. 이러한 시스템의 설명을 위해 군사 분야에서 일정지역의 적의 침입을 감시하기 위한 감시 카메라 시스템을 예로 들어 설명하면 <그림 1>에서 각각의 작은 원은 감시 카메라를 나타낸다. 각각의 감시 카메라는 일정거리  $d$ 만큼 떨어져 있으며 반지름이  $c$ 인 원형의 지역을 감시 할 수 있는 능력을 가지고 있다(즉,  $d=c$ 인 경우). 위에서 정의한 것처럼 이 감시 카메라 시스템의 고장은 각각의 카메라들이 고장으로 인하여 전 영역을 감시하지 못할 때 발생하며 각각의 감시 카메라 용량이 크면 그만큼 이 시스템의 신뢰도는 증가하는 것이다.



<그림 1>  $(m, n)$ 중-연속 $(r, s)$ :F 시스템

$d=c$ 인 경우 시스템은 아래 <그림 2>처럼 선형(3,3)중-연속(2,1):F 시스템으로 정의 할 수 있으며 한 대의 카메라가 고장났을 경우는 시스템이 고장나지 않으며 두 대의 연속적인 카메라가 고장 났을 경우 시스템 전체의 고장이 발생한다.



<그림 2> 선형(3,3)중-연속(2,1):F 시스템의 고장 상태

각각의 카메라의 용량이 크다는 것은 비록 몇 개의 카메라가 고장 나더라도 남은 카메라가 전 지역을 감시 할 수 있으며 이것은 선형  $(m,n)$ 중-연속 $(r,s):F$  시스템에서 시스템 모수  $r, s$ 가 크다는 것을 의미한다. 즉 시스템의 신뢰도는  $r, s$ 의 크기에 비례한다고 할 수 있다. 그러나 각각의 카메라 용량의 크기는 투자비용과 비례하기 때문에 무작정 카메라 용량을 늘릴 수는 없다. 따라서 적절한 정비 주기와  $r, s$ 의 크기를 결정하는 것은 비용의 측면에서 볼 때 매우 중요한 문제이다.

### 3. 2 고장 발생 및 정비정책

시스템의 모형구축을 위하여 선형  $(m,n)$ 중-연속 $(r,s):F$  시스템의 각 시스템을 구성하는 부품들은 지수분포(exponential distribution)를 따라 고장을 발생한다고 가정하며 지수분포의 확률밀도 함수(pdf)는 식(6)과 같다.

$$f(t) = \lambda e^{-\lambda t} \quad (t > 0) \quad (6)$$

지수분포는 일반적으로 전자 제품과 같이 하나의 장비가 여러 형태의 고장률을 갖고 쇠약해지는 즉, 수없이 많은 부품으로 이루어져 있을 때 나타난다. 이 분포의 가장 큰 특징은  $t$ 라는 시간 동안 사용 중이던 부품이 그 다음 순간에도 가동할 확률은 새 부품이 방금 가동을 시작하여 그 다음 순간에도 가동할 확률과 같다는 것이다.

시스템을 운용함에 따라 각 부품들은 고장이 나게 마련이며 고장 난 부품들을 언제 어떻게 정비 하느냐는 시스템의 신뢰도 및 운용 비용에 크게 영향을 미친다. 시스템 정비 정책은 시기와 방법에 따라 여러 가지로 나뉠 수 있다.

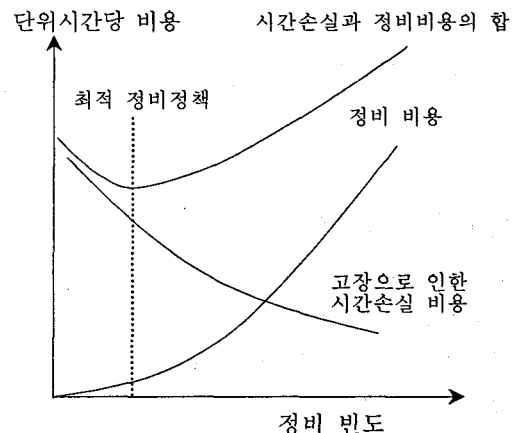
본 연구에서는 가장 많이 사용하는 시스템이 고장 나거나 혹은 일정 주기 때 정비를 실시하는 주기적 정비 정책(periodic replacement)을 사용한다. 또한 정비 방법은 시스템 내의 고장 난 부품을 찾아 고장 난 부품들만 새것으로 교체하는 것으로 가정하고 각 구성 부품

들 간의 고장은 각 부품들의 상태와 고장 확률이 다른 부품에 영향을 끼치지 않고 독립적으로 발생한다고 가정하였다.

## 4. 수리모형

정비는 크게 예방정비와 사후정비로 나눌 수 있는데 예방정비 횟수가 많아지면 시스템 신뢰도가 높아지지만 그만큼 예방정비 비용이 늘어나게 된다. 따라서 적절한 교체 주기를 결정하는 것은 정비비용과 시스템 신뢰도 문제에서 매우 중요한 문제이다. 왜냐하면 시스템이 고장 난 후의 정비 비용은 예방정비 시의 정비 비용보다 훨씬 크게 적용되기 때문이다. 따라서 사후정비 비용이 예방정비 비용보다 큰 것으로 가정하는 것이 타당하다고 여겨진다.

또한 <그림 3>과 같이 정비 주기를 너무 작게 설정하여 정비 빈도를 늘리면 예방정비의 횟수가 많아지므로 장기적으로는 총 교체 비용이 커질 가능성이 있다. 반면 정비 주기를 너무 크게 설정하여 정비 빈도를 줄이면 고장 발생의 빈도가 높아져 사후 정비 비용이 많이 들기 때문에 이 또한 경제적이지 않다.



<그림 3> 정비 빈도에 따른 최적 정비정책

따라서 시스템의 최적설계를 위해 구하고자 하는 비용함수는 Jardine[12]에 의해 아래와 같이

나타낼 수 있다.

$$\alpha(T_r) = \quad (10)$$

$\alpha(T_r)$  = 한 주기  $(0, T_r)$ 에서의 총 기대비용 / 한 주기의 기대길이

$$\begin{aligned} \lim_{t \rightarrow \infty} \frac{C_2 R(T_r)(t/L(T_r)) + C_0 F(T_r)(t/L(T_r))}{t} \\ = \frac{C_2 R(T_r) + C_0 F(T_r)}{L(T_r)} \end{aligned}$$

여기서 한 주기는 두 가지의 경우로 볼 수 있으며 하나는 예방정비 주기( $T_r$ ) 이전에 시스템이 고장 나는 경우이며 다른 하나는  $T_r$ 시점에서 예방정비를 하는 것이다. 그러므로 한 주기  $(0, T_r)$ 에서의 총 기대 비용은 식 (7)과 같이 쓸 수 있다.

$$\text{총기대비용} = C_2 R(T_r) + C_0(1 - R(T_r)) \quad (7)$$

또한 비슷한 방법으로  $T_r$ 에서의 시스템 기대 수명은 예방정비의 주기  $\times$  예방정비를 취할 확률 + 고장 주기의 기대길이  $\times$  고장 주기의 기대길이를 나타낼 수 있으므로 식 (8)과 같이 쓸 수 있다.

$$L(T_r) = T_r R(T_r) + \int_0^{T_r} t f(t) dt = \int_0^{T_r} R(t) dt \quad (8)$$

따라서 단위 시간당 기대 비용은 식 (9)와 같다.

$$\alpha(T_r) = \frac{C_2 R(T_r) + C_0 F(T_r)}{\int_0^{T_r} R(t) dt} \quad (9)$$

본 연구의 정비정책은 시스템이 기간  $T_r$ 까지 고장 나지 않으면 그 시점에서 예방정비를 하고 기간  $T_r$ 이 되기 전에 고장이 나면 즉각 사후 정비를 하는 정비 정책이다. 고장 난 부품은 모두 새 것으로 교체해 주며 임의의 시간  $t$ 까지의 총 교체 수의 기대치는  $t/L(T_r)$ 으로 표현 할 수 있다.

따라서 장기간에서의 단위 시간당 기대 비용은 식 (10)과 같이 구할 수 있다.

위 식에서 단위 부품을 교환하는 비용( $C_1$ )을 추가하면 결정변수가  $T_r$ 일 경우의 단위 시간당 기대비용은 식 (11)과 같이 표현 할 수 있다.

$$\begin{aligned} \alpha(T_r) &= \frac{C_1 M(T_r) + C_2 R(T_r) + C_0 F(T_r)}{L(T_r)} \quad (11) \\ &= \frac{C_1 M(T_r) + C_2 R(T_r) + C_0(1 - R(T_r))}{L(T_r)} \end{aligned}$$

결정 변수가  $T_r, r, s$  세 가지인 경우에는 단위 부품을 교체하는데 드는 비용  $C_1$ 은  $r, s$ 의 크기에 종속적이다. 왜냐하면 시스템 모수  $(m, n, r, s)$  중 고장 모수  $(r, s)$ 가 (2, 2)인 경우와 (3, 3)인 경우의 단위 부품을 교체하는데 드는 비용은 다르며 고장 모수가 커질수록 더 많은 비용이 들게 된다. 앞의 경우에서 살펴본 것처럼  $r, s$ 가 커질수록 한 개의 부품이 커버할 수 있는 범위가 큰(좋은 성능) 것이고 그만큼 비용이 비싸기 때문이다. 따라서 위 내용을 정리하면 식 (12)와 같은 최종 목적함수를 표현 할 수 있다.

$$\begin{aligned} \alpha(T_r, r, s) \\ = \frac{C_1 M(T_r, r, s) + C_2 R(T_r, r, s) + C_0(1 - R(T_r, r, s))}{L(T_r, r, s)} \quad (12) \end{aligned}$$

## 5. 시뮬레이티드 어닐링 알고리즘

본 연구에서 목적함수인 단위시간당 기대비용  $\alpha(T_r)$ 을 최소로 하는 최적의 교체주기를

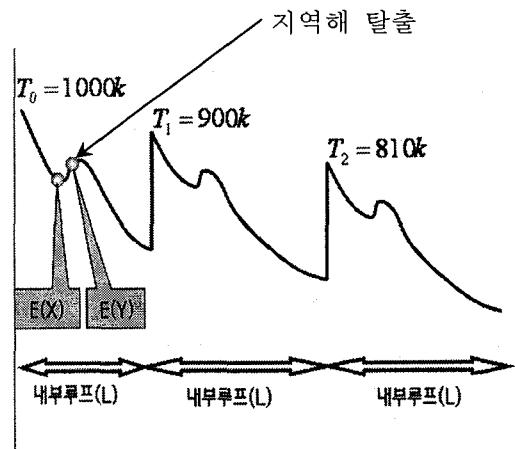
구하기 위해서는 단위시간당 기대비용의 함수식을 수리적으로 구하여야 하나 그 식이  $T$ ,  $r$ ,  $s$ 에 따라 달라지면서 신뢰도 함수  $R(t)$ 를 구하는 것이 기존의 연구처럼 복잡한 순환식의 제시, 상한 및 하한 값의 제시 등으로 시스템의 최적화를 위한 적용에는 제한에 있으므로 본 연구에서는 시뮬레이티드 어닐링(Simulated Annealing) 알고리즘을 이용하여 단위 시간당 기대 비용을 최소화 하는 정비주기와 기대비용을 결정하고 시스템을 최적화하여 기존의 유전자 알고리즘보다 빠른 탐색시간 내에 좀 더 효율적으로  $(m, n)$ 중-연속  $(r,s)$ F시스템의 최적화 문제를 해결 할 수 있도록 하는 알고리즘을 제시한다.

### 5. 1 알고리즘 개요

고체물리학에서 에너지 수준이 가장 낮은 상태인 결정을 얻기 위한 어닐링 과정을 최적화 문제에 묘사한 SA알고리즘의 기본형태는 <그림 4>와 같다. 초기해  $X$ 를 선택하고,  $X$ 의 이웃해  $Y$ 를 얻어 이웃해가 목적함수를 개선시킬 수 있는 경우에 이웃해를 초기해와 교체하고, 이 과정을 반복하는데 이때 지역최적해에 빠지는 문제를 해결하기 위해 온도에 대한 에너지변화에 따른 확률  $\exp(-\Delta/T)$ (metropolis criterion)에 의해 비용이 큰 이웃해로의 이동도 허용함으로써 지역 최적해에 빠졌을 경우, 지역해 탈출(hill climbing) 방법을 통해 이를 탈피할 수 있도록 하여준다. 시뮬레이티드 어닐링 알고리즘은 아래와 같은 요소에 의해 성능이 결정된다[1].

- INITIALIZE

알고리즘이 시작하는데 필요한 초기해  $X$ 와 컨트롤 파라미터  $T$  내부루프의 반복수  $L$ 를 결정한다.  $T$ 는 결정 상태를 얻기 위한 어닐링 과정의 온도와 같은 역할을 하고  $L$ 은 주어진 온도에서 평형상태에 이를 때까지 무작위적인 입자들의 이동을 계속하게 하는 어닐링 과정에 해당하는 내부루프의 반복수다.



<그림 4> SA 알고리즘 절차

- PERTURB

현재해로부터 인접해를 만들어주는 과정으로 문제의 특성에 따라 다르게 정해지는데 매우 많은 반복이 이루어지고 알고리즘의 성능에 직접적인 영향을 끼친다.

- 수락기준

해의 변동결과 목적함수 값이 커졌을 경우에 나쁜 해를 받아들이는 기준으로 식 (13)과 같이 초기온도가 높을 경우에는 확률적으로 많이 받아들이고 온도가 낮아지면서 적게 받아들여 궁극적으로는 hill climbing하면서 가장 비용이 낮은 목적함수 값으로 수렴 할 수 있도록 하는 기준이다.

$$\exp(-\Delta/T) > \text{random}(0, 1) \quad (13)$$

- 쿨링스케줄

온도를 내리는 함수는 감소함수로 시간의 경과에 따라 0으로 수렴하며 대표적인 쿨링스케줄은 기하스케줄, 로그스케줄 등의 방법이 있다.

### 5. 2 실험설계

본 연구에서 제안한 시뮬레이티드 어닐링 알고리즘을 이용하여 단위시간당 기대비용을 최소화 시키는 정비주기와 기대비용을 실험하

고 기존해법에 의한 결과와 비교 분석하였다.

본 연구에서는 기존의 알고리즘이 Visual Basic으로 코딩한데 반하여 MFC(VC++)로 구현을 하여 성능을 훨씬 향상시켰으며 컴퓨터는 Pentium-IV CPU 2.66GHz RAM : 512M을 사용하였다.

시스템에 있는 모든 부품의 고장은 서로 독립이고 각 부품의 고장 분포는  $\lambda=0.02$ 인 지수분포를 따른다. 입력모수는 실험횟수,  $m, n, r, s, C_0, C_1, C_2$ 와 시뮬레이티드 어닐링 알고리즘 모수인 초기온도, 냉각률, 현재해 변경비율, 반복수를 변화시켜 최적의 값을 도출한다.

실험의 반복수는 반복수가 많아질수록 실험 결과의 정확도는 높으나 시간이 많이 소요되므로 결과에 많은 영향을 끼치지만 실제로는 반복수가 일정 수 이상 증가하면 그 결과치에 많은 영향을 끼치지 않기 때문에 실험 반복수는 유전자 알고리즘과 비교하여 시뮬레이티드 알고리즘의 효과를 분석하기 위하여 윤원영[3]과 동일하게 100회를 사용하여 실험을 실시하였다.

• 초기해 표현

어떠한 초기치가 선정되느냐에 따라 해의 성능향상에 중요한 역할을 하나 설정에 대한 규칙이 확립되지 않아 본 논문에서는 문제의 특성상 난수를 사용하였다. 초기해인 정비주기 ( $T_r$ )의 표현은 아래와 같이 정수부분 2자리 소수부분 3자리를 사용하였고 난수를 사용하여 5자리의 수가 0에서 9까지 random하게(경우의 수  $10^5$ ) 변화도록 하였다.

예)  $T_r=34.374$ (각각의 수가 0-9까지 변함)

• 초기온도 / 내부루프 길이 결정

초기온도는 내부루프의 길이 보다 전역 최적해의 수렴에 더 큰 영향을 주므로 적절한 초기온도를 구하기 위하여 <표 1>과 같이 선형 (5,5)중-연속(3,3):고장 격자 시스템에서 다른 조건을 동일하게 하면서 50℃, 70℃, 100℃, 150℃, 200℃로 다르게 적용하여 실험한 결과 50℃에서 0.433으로 시작하여 70℃에서 0.406

으로 100℃에서는 0.379로 해가 개선되다가 150℃이상에서는 개의 개선 편차가 0.01이하로 적어져, 알고리즘의 시간성능을 고려 100℃로 설정하였다.

또한 내부루프의 길이는 주어진 온도에서 안정상태가 될 때까지의 반복횟수로 수렴성의 보장과 해의 질 개선을 위해서는 길이를 과도하게 주는 경향이 있으나 오히려 길이를 늘이는 것보다 온도의 떨어지는 폭을 줄이는 것이 계산 시간을 줄이면서 동질의 해를 얻을 수 있기 때문에 길이를 과도하게 늘려서 실험하지 않고 <표 1>과 같이 50, 100, 150, 200, 250으로 실험하여 해의 개선 편차가 0.01이하로 적어지는 100으로 설정하였다.

<표 1> 온도와 루프 길이에 따른 실험결과

$(m,n)(r,s) : (5,5)(3,3)$					
$C_0:100, C_1:1, C_2:0.1, \text{반복수 } 100\text{회}$					
온도(℃)	50	70	100	150	200
$\alpha(T_r)$	0.433	0.406	0.379	0.368	0.362
내부루프길이	50	70	100	150	200
$\alpha(T_r)$	0.454	0.410	0.367	0.364	0.358

• 쿨링스케줄

본 연구에서는 로그스케줄이 이론적인 수렴성은 보장되나 수렴속도가 너무 늦어져 가장 일반적으로 사용하는 식 (14)와 같은 기하스케줄에 의한 방법을 적용하였으며 전역최적해로의 수렴성에 직접적인 영향을 주기 때문에 <표 2>와 같이 냉각률  $a$ 값을 구하기 위해 0.85, 0.9, 0.95, 0.99로 실험하여 해의 개선 편차가 0.01이하로 적어지는 0.95로 설정하였다.

$$T_k = aT_{k-1}, k=1, 2, \dots, 0 < a < 1 \quad (14)$$

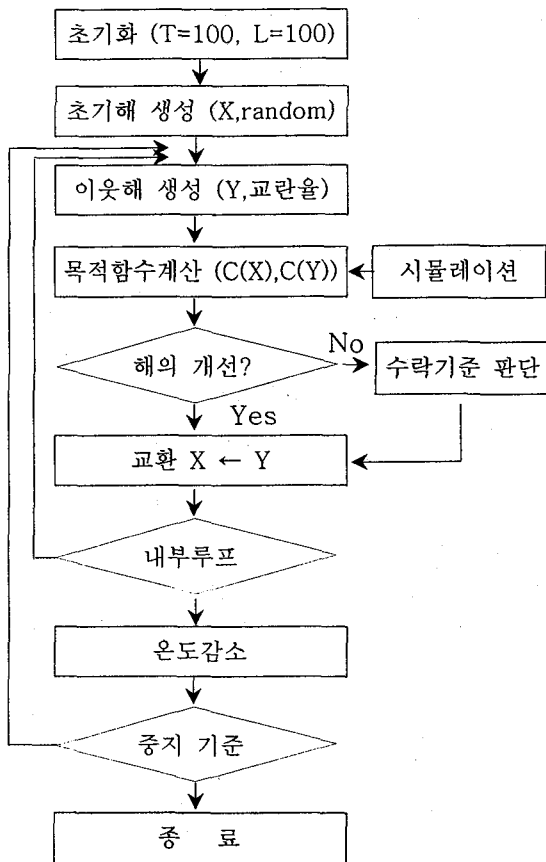
<표 2> 쿨링스케줄 변화에 따른 실험결과

$(m,n)(r,s) : (5,5)(3,3)$				
$C_0:100, C_1:1, C_2:0.1, \text{반복수 } 100\text{회}$				
냉각률	0.85	0.9	0.95	0.99
$\alpha(T_r)$	0.435	0.405	0.375	0.368

• 목적함수 계산

시뮬레이티드 어닐링 알고리즘에서 정비주기( $T$ )인 초기해와 이웃해를 결정후에 해의 개선을 위해서는 목적함수인  $C(T)$  값을 구하여야하나 수리적으로 해결되지 않으므로 윤원영[3]이 사용한 몬테칼로 시뮬레이션을 동일하게 적용하여 시스템에 작동중인 부품들의 고장을  $\lambda=0.02$ 인 지수분포를 적용하여 발생시키고 시스템의 고장을 확인하여 고장유무에 따른 예방정비를 함에 따라 반복 시행을 통하여 신뢰도 및 평균고장개수, 평균고장시간 등을 구하여 목적함수인  $C(T)$  값을 구할 수 있도록 하여 유전자 알고리즘과 시뮬레이티드 어닐링 알고리즘의 결과가 비교될 수 있도록 하였다.

위에서 설계한 시뮬레이티드 알고리즘의 절차를 요약하면 <그림 5>와 같다.



<그림 5> SA 알고리즘 절차

• 수치예제

Initialize :  $T=100, L=100$

(1 단계)  $L=1$

$X1 = 75.885$  (random으로 발생)

$Y1 = Perturb(X1) = 75.894$

$C(X1) = 0.712$   $C(Y1) = 0.615$

$C(X1) = 0.712 > C(Y1) = 0.615$  이므로

$X1 = Y1 = 75.894$

(2단계)  $L=2$

$X2 = 75.894$

$Y2 = Perturb(X2) = 71.894$

$C(X2) = 0.615$   $C(Y2) = 0.746$

$C(X2) = 0.615 < C(Y2) = 0.746$  이나  $\exp((C(X2)-C(Y2))/T) > Random(0,1)$

$X2 = Y2 = 71.894$

수치예제의 1단계에서는  $C(X1) = 0.712 > C(Y1) = 0.615$ 으로 해의 개선이 이루어져 인접해  $Y1$ 이 선택되고 2단계에서는  $C(X2) = 0.615 < C(Y2) = 0.746$ 으로 해의 개선이 이루어지지 않았으나 metropolis 기준에 의해  $\exp((C(X2)-C(Y2))/T) > Random(0,1)$ 이므로 hill climbing을 위해 나쁜 해  $Y2 = 71.894$ 을 받아들이는 것을 볼 수 있다. 이러한 과정을 초기온도  $T=100$ 일 때  $L=1$ 에서부터 100까지 실행하여 내부루프를 반복하고 다시 외부루프로 돌아와 쿨링 스케줄에 의해 온도를  $T=95$ 로 낮추어  $L=1$ 에서부터 100까지 실행한다. 이러한 반복을  $T$ 가 0에 근접할 때 까지 반복하면 최종적으로 해가 구해진다.

6. 실험 및 결과분석

실험은 두 가지 유형으로 구분하여 실행하였다. 먼저 축소실험에서는 윤원영[3]이 사용한 선형(5,5)중-연속(3,3):고장 격자 시스템을 실험하였고 확대실험에서는 선형(10,10)중-연속(3,3):고장 격자 시스템을 실험하고  $m, n$ 을 증가시키면서 목적함수의 값 및 계산시간을 비교하여 알고리즘의 우수성을 확인하며 기존



해법들과의 비교를 통해 효율성을 판단하였다.

### 6. 1 축소실험

축소실험에서는 선형(5,5)중-연속(3,3):F 시스템에 대하여 기존의 유전자 알고리즘과 시뮬레이티드 어닐링 알고리즘을 사용한 결과를 비교하였다.

실험결과 <표 3>과 같이 시뮬레이티드 어닐링 알고리즘의 기대비용  $\alpha(T)$ 값이 기존의 값과 근사한 결과를 나타내고 있으며 반면에 수행시간 면에서는 유전자 알고리즘이 5분 10초인데 반하여 시뮬레이티드 어닐링 알고리즘은 1분 50초로 월등히 우수하게 나타나고 있다. 이렇게 수행시간이 우수하게 나타나는 원인은 알고리즘의 수행 간 목적함수  $\alpha(T)$ 값의 결정이 시뮬레이션을 통해서 구해지므로 시간이 많이 소비되는 부분인데 유전자 알고리즘은 다수의 염색체에 대한 적합도 과정을 거쳐야 하므로 염색체의 적합도 평가 과정에서 과도하게 시간이 소비되는 것으로 판단된다.

<표 4>  $C_0, C_2$ 변화에 따른  $T_r, \alpha(T)$ 의 변화

구 분	$T_r$	$\alpha(T)$
$C_0$ (시스템 고장시 고정비용)의 증가	감소	증가
$C_2$ (예방정비시 고정비용)의 증가	증가	증가

또한 <표 4>와 같이 목적식이 비용함수이므로  $C_0$ (시스템 고장 시 고정비용),  $C_2$ (예방정비시 고정비용)값이 커짐에 따라 기대비용  $\alpha(T)$ 은 커짐을 알 수 있다. 또한 시스템 고장 시 고정비용  $C_0$ 값이 커짐에 따라 정비주기  $T_r$ 은 시스템 고장 시 고정비용을 줄이기 위해 고장 전에 정비를 더 자주 해야 하므로 결국 정비주기  $T_r$  값이 작아짐을 알 수 있으며 반면에 예방 정비 시 고정비용  $C_2$ 값이 커짐에 따라 정비주기  $T_r$ 은 고정비용을 줄이기 위해  $T_r$ 값이 커짐을 알 수 있다.

<표 3> GA 알고리즘과 SA 알고리즘의 실험결과 비교

구 분		실험횟수 100회, 염색체수 50, 세대수 100			실험횟수 100회, 초기온도 100℃, 내부루프 반복수 100		
$C_2$	$C_0$	GA 알고리즘			SA 알고리즘		
		정비주기( $T_r$ )	기대비용 $\alpha(T)$	Time	정비주기( $T_r$ )	기대비용 $\alpha(T)$	Time
0.1	10	61.043	0.2851	5분 10초	48.443	0.318	1분 50초
	50	36.524	0.3001		32.568	0.336	
	100	33.921	0.3084		26.709	0.379	
	1000	29.486	0.3206		23.512	0.397	
0.5	10	68.836	0.2893		53.752	0.357	
	50	36.756	0.3219		33.406	0.382	
	100	34.472	0.3223		31.756	0.402	
	1000	30.475	0.3452		26.478	0.426	
1	10	80.093	0.2899		73.607	0.376	
	50	37.876	0.3376		35.426	0.395	
	100	34.822	0.3426		33.733	0.428	
	1000	30.277	0.3640		25.930	0.442	

## 6. 2 확대실험

확대실험은 선형(10,10)중-연속(3,3):F 시스템을 실험하고  $m, n$ 을 증가시키면서 목적함수의 값 및 계산시간을 확인하여 기존해법들과의 비교를 통해 알고리즘의 우수성을 입증하였다. 또한 각 부품이  $\lambda=0.02$ 인 지수분포로 고장 나는 상황을 확대하여  $\lambda$ 값의 증가에 따른 변화를 살펴보았다.

실험결과 <표 5>와 같이  $m, n$ 이 증가 할수록 기대비용과 알고리즘 계산시간은 증가하는 것을 볼 수 있다. 또한 기존의 GA 알고리즘과 비교할 때 기대비용은 비슷한 값을 보이는 반면 알고리즘 계산시간은  $m, n$ 의 크기가 커질수록 훨씬 우수하게 나타나고 있는 것을 볼 수 있다. 이러한 현상은 앞의 축소 실험에서도 설명하였듯이 GA 알고리즘이 유전자 적합도 평가 과정에서 과도하게 시간을 소비하기 때문이다.

<표 5> 확대 실험 결과 비교

$(m,n)$ $(r,s)$	$C_2$	$C_0$	GA 알고리즘		SA 알고리즘	
			기대비용 $C(T)$	Time	기대비용 $C(T)$	Time
(10,10) (3,3)	0.1	100	1.5212	51분	1.665	25분
(11,11) (3,3)	0.1	100	1.8594	75분	1.944	29분
(12,12) (3,3)	0.1	100	2.1384	118분	2.254	39분
(13,13) (3,3)	0.1	100	2.5268	174분	2.749	58분
(14,14) (3,3)	0.1	100	3.0528	240분	3.250	75분
(15,15) (3,3)	0.1	100	3.4823	318분	3.604	98분

또한 각 부품이  $\lambda=0.02$ 인 지수분포로 고장 나는 상황을 확대하여  $\lambda$ 를 증가시킬 때는 <표 6>과 같이 기대비용이 증가하는 것을 알 수 있다. 즉 각 부품의 고장률이 증가할 때 전체 시스템의 정비주기는 짧아지며 기대비용은 증가하는 것이다.

<표 6> 고장률  $\lambda$ 값의 변화에 따른 결과

$\lambda$	$(m,n)(r,s) : (5,5)(3,3)$				
	0.02	0.05	0.1	0.3	0.5
$C(T)$	0.379	0.919	1.796	5.443	8.741
$T$	26.709	11.529	6.191	1.920	0.725

## 7. 결론

본 논문은 시뮬레이티드 어닐링 알고리즘을 이용하여 부품 중복 설계의 한 종류인 선형  $(m,n)$ 중-연속  $(r,s):F$  시스템의 정비 주기와 최적의 단위시간당 기대비용을 구하였다.

기존의 유전자 알고리즘을 이용한 정비모형 최적화는 알고리즘의 수행시간을 과도하게 요구하여 활용성이 제한되는 문제를 가지고 있어 본 연구에서는 기존의 유전자 알고리즘보다 빠른 탐색시간 내에 좀 더 효율적으로  $(m,n)$ 중-연속  $(r,s):F$  시스템의 최적화 문제를 해결할 수 있도록 하는 시뮬레이티드 어닐링 알고리즘을 제시하였다. 그 결과 목적함수가 시뮬레이션에 의해 복잡하게 결정되는 경우 유전자 알고리즘이 적합도 평가과정에서 느린 탐색시간을 갖는 문제를 보완하여 확대시험 결과에서 본 것처럼  $(m,n)$ 이 (10,10)일 경우 25분, (15,15)일 경우 210분 이상 탐색시간이 줄어들어 해의 범위가 확대 될수록 효과적인 성능을 보임을 입증하였다.

선형(10,10)중-연속(3,3):F 시스템과 같은 다 부품 중복설계 시스템의 신뢰도와 최적화 모형 연구는 지금까지 많은 연구에도 불구하고 수치적 해법의 복잡성으로 인해 국내에서는 연구 성과가 거의 없으며 지속적인 연구가 요구되는 분야이다. 때문에 차후 타부서치 등 다양한 방법을 이용한 연구를 통해 이 분야의 연구 활성화와 발전이 필요하며 각종 제약조건이 없는 상태에 대한 연구와 확대하여 3차원 형태인  $(m,n,o)$ 중-연속  $(r,s,t):F$  시스템의 문제 분야로의 추가적인 연구가 필요할 것으로 생각된다.

## 참고문헌

- [1] 김여근, 윤복식, 이상복, 「메타휴리스틱」, 영지문화사, 2000, pp.185-258.
- [2] 김호균, 배창욱, 백천현, “다중부품선택이 존재하는 직렬구조 시스템의 최적 신뢰성설계를 위한 시뮬레이티드 어닐링 알고리즘”, 「IE Interfaces」, 제17권, Special Edition(2004), pp.69-78.
- [3] 윤원영, 김귀래, 정철훈, “유전자 알고리즘을 이용한  $(m,n)$ 중-연속  $(r,s)$ 고장격자 시스템의 정비모형”, 「대한산업공학회지」, 제30권, 제3호(2004), pp.250-260.
- [4] 조계연, 윤복식, 이창훈, “Simulated Annealing의 효과적인 변형과 실험”, '94 한국경영과학회 추계학술대회(1994).
- [5] Aarts, E.H.L. and Korst, J.H.J., *Simulated Annealing and Boltzmann Machines*, John Wiley & Sons, New York, 1989.
- [6] Angus, J. E. and Ames, K., "A Simulated annealing algorithm for system cost minimization subject to reliability constraints", *Communications in Statistics; Simulation and Computation*, Vol.26(1997), pp.783-790.
- [7] Blanks, H. S. and Tordan, M. J. "Optimum replacement of deteriorating and inadequate equipment", *Quality and Reliability Engineering*, Vol.2(1986), pp.183-197.
- [8] Boehme, T. K., Kossow, A. and Preuss, W., "A generalization of consecutive- $k$ -out-of- $n:F$  systems", *IEEE Trans. Reliability*, Vol.41, No.3(1992), pp.451-457.
- [9] Chao, M. T., Fu, J. C., Koutras, M. V., "Survey of Reliability Studies of Consecutive- $k$ -out-of- $n:F$  & Related Systems", *IEEE Trans. Reliability*, Vol.44, No.1(1995), pp.120-127.
- [10] Derman, C., Lieberman, G. J., and Ross, S. M., "On the Consecutive- $k$ -out-of- $n:F$  system", *IEEE Trans. Reliability*, Vol.R-31, No.1(1982), pp. 57-63.
- [11] Fu, J. C. and Hu., "On reliability of a large consecutive- $k$ -out-of- $n:F$  systems with  $(k-1)$ step Markov dependence", *IEEE Trans. Reliability*, Vol.R-36(1987), pp. 602-606.
- [12] Jardine, A. K. S., *Maintenance, replacement, and reliability Theory and Applications*, New York, John Wiley, 2005.
- [13] Kontoleon, J. M., "Reliability determination of a  $r$ -successive-out-of- $n:F$  system", *IEEE Trans. Reliability*, Vol.R-29(1980), pp. 437-438.
- [14] Koutras, M. V., Papadopoulos, G. K. and Papastavridis, S. G., "Reliability of 2-dimensional consecutive- $k$ -out-of- $n:F$  systems", *IEEE Trans. Reliability*, Vol.42, No.4(1993), pp.658-661.
- [15] Ksir, B., "Comment on: 2-dimensional consecutive  $k$ -out-of- $n:F$  models", *IEEE Trans. Reliability*, Vol.41, No.4(1992), p. 575.
- [16] Locks, M. O., "A minimizing algorithm for sum of disjoint products", *IEEE Trans. Reliability*, R-36(1987), pp.120-127.
- [17] Malinowski, J. and Preuss, W., "Lower & upper bounds for the reliability of connected- $(r,s)$ -out-of- $(m,n):F$  lattice systems", *IEEE Trans. Reliability*, Vol.45, No.1(1996), pp.156-160.
- [18] Noguchi, K., Sasaki, M., Yanagi, S. and Yuge, T., "Reliability of connected- $(r,s)$ -out-of- $(m,n):F$  lattice system", *IEICE Transactions Fundamentals*, VolJ79-A, No.4(1996), pp.954 -963.
- [19] Papastavridis, S. and Lambiris, M., "Reliability of a consecutive- $k$ -out-of- $n:F$  system for Markov dependent components", *IEEE Trans. Reliability*, Vol.R-36(1987), pp.78-79.
- [20] Ramirez-Marque, J.E. and Coit, D.W., "A heuristic for solving the redundancy allocation problem for multiple-state series-parallel systems", *Reliability Engineering and System Safety*, Vol.83(2004), pp341-349.
- [21] Salvia, A. A., "Simple Inequalities of Consecutive- $k$ -out-of- $n:F$  System", *IEEE*

- Trans. Reliability*, Vol.R-31(1982), p.450.
- [22] Salvia, A. A. and Lasher, W. C., "2-Dimensional consecutive- $k$ -out-of- $n$ :F models", *IEEE Trans. Reliability*, Vol.39, No.3(1990), pp.333-336.
- [23] Yamamoto, H. and Miyakawa, M., "Reliability of a linear connected- $(r,s)$ -out-of- $(m,n)$ :F lattice systems", *IEEE Trans. Reliability* Vol.44, No.2(1995), pp.333-336.
- [24] Yamamoto, H. and Miyakawa, M., "Reliability of circular connected- $(r,s)$ -out-of- $(m,n)$ :F lattice systems", *Journal of the Operations Research Society of Japan*, Vol.39, No. 3(1996), pp.389-406.
- [25] Zuo, M. G., "Reliability & design of 2-dimensional consecutive  $k$ -out-of- $n$ :F systems", *IEEE Trans. Reliability*, Vol.42, No.4(1993), pp. 488-490.