

## 論文

DOI: <http://dx.doi.org/10.5139/JKSAS.2011.39.12.1150>

## 인공위성 시스템의 신뢰도 최적 설계에 관한 연구

(아리랑위성 2호의 MSC 시스템 구조를 중심으로)

김홍섭\*, 전진욱\*\*

## A Study on Reliability Optimal Design of Satellite system

(Based on MSC System's structure of KOMPSAT-2)

Heung-Seob Kim\*, Geonwook Jeon\*\*

## ABSTRACT

Reliability is defined as a probability that a system will operate properly for a specified period of time under the design operating conditions without failure. Reliability-Redundancy Optimization Problem(RROP) involves selection of components with multiple choices, redundancy levels and redundancy strategy(Active or Standby) for maximizing system reliability with constraints such as cost, weight, etc. Based on the design configuration of Multi-Spectral Camera(MSC) system of KOMPSAT-2, the mathematical programming model for RROP is suggested in this study. Due to the nature of RROP, i.e. NP-hard problem, Parallel Particle Swarm Optimization(PPSO) algorithm is proposed to solve it. The result of the numerical experiment for RROP is presented as instance of recommended design configuration at some mission time.

## 초 록

신뢰도란 임의 시스템이 주어진 운용환경 하에서 의도한 기간 동안 의도된 기능을 정상적으로 수행할 확률로 정의된다. 신뢰도-중복 최적화 문제(RROP)는 비용, 무게 등의 제약 내에서 시스템의 신뢰도를 최대화할 수 있는 최적의 부품을 선택하고, 부품수와 중복전략(활성/대기중복)을 결정하는 문제이다. 본 연구에서는 아리랑위성 2호의 다채널광학카메라(MSC) 시스템의 설계 구조를 바탕으로 RROP의 수리모형을 제시하고, NP-hard인 RROP의 해법으로써 병렬 개체군집최적화(PPSO) 알고리즘을 제안하였다. RROP 예제의 수치실험 결과는 계획된 수명기간에서 신뢰도를 최대화하는 시스템의 설계 구조를 제시한다.

**Key Words** : Reliability(신뢰도), Satellite System(인공위성 시스템), Optimal Design(최적 설계), RROP(신뢰도-중복 최적화 문제), Standby Redundancy(대기 중복), PPSO Algorithm(병렬 개체군집최적화 알고리즘)

## 1. 서 론

† 2011년 9월 16일 접수 ~ 2011년 11월 28일 심사완료

\* 정회원, 국방대학교 운영분석전공

\*\* 비회원, 국방대학교 운영분석전공

교신저자 : 김홍섭, E-mail : afrotc02@naver.com

서울특별시 은평구 수색동 205번지

신뢰도(Reliability)란 임의 시스템이 주어진 운용환경 하에서 의도한 기간 동안 의도된 기능을 정상적으로 수행하여 본연의 임무를 성공적으로 완수할 확률로써 정의된다. 즉 임의 시스템이 임

무 말에 약 0.8의 신뢰도를 갖는다는 것은 임무 기간 중에 규격성능을 발휘할 확률이 80%이고, 그 반대의 상황은 20%임을 의미한다. 이러한 신뢰도를 과학적으로 접근하기 위한 신뢰도 공학(Reliability Engineering)의 중요성은 1986년 우주왕복선 Challenger호의 폭발사고, 2003년 Colombia호의 왼쪽날개 손상으로 인한 귀환 중 폭발사고에서도 찾아 볼 수 있다[1]. 오늘날에는 일상에서 사용되는 생활용품에서부터 각종 첨단 시스템에 이르기까지 다양한 산업분야의 시스템 설계의 중요한 척도로 고려되고 있다.

특히, 인공위성(Satellite)과 같이 고가이고 보수할 기회가 없는 시스템의 경우, 고유의 신뢰도는 두말 할 것도 없이 필수적이다. 따라서 우주공간에서 운용되는 시스템의 더 높은 신뢰도를 실현하기 위한 노력이 끊임없이 계속되고 있다[2]. 일반적으로 시스템의 신뢰도를 증진시키는 방법에는 시스템을 구성하는 부품의 신뢰도를 증진시키거나, 부품을 중복(Redundancy) 설계하는 2가지의 방법이 있다. 최근에는 활성중복(Active redundancy)과 대기중복(Standby redundancy)을 모두 고려하여 시스템의 신뢰도를 최대화하기 위한 최적의 설계 구조를 찾는 연구가 진행되고 있다[3,4].

인공위성 개발 시에는 주로 우주환경 시험 등의 고강도의 테스트를 거쳐 품질의 안정성이 검증된 부품을 사용하게 된다. 따라서 설계자는 요구하는 규격성능을 만족하는 여러 종류의 부품들 중 가장 적합한 부품을 선택하고, 목표하는 시스템의 신뢰도를 만족시키기 위해 중복설계 부품수를 고려하게 된다. 하지만 중복 부품수를 증가시키는 것은 시스템의 신뢰도를 향상시킬 수는 있지만 추가의 비용, 추가의 무게 등을 수반하게 되므로, 이에 대한 절충(Trade-off)을 통해 신뢰도를 최대화할 수 있는 설계 구조를 찾아야 한다.

따라서 본 연구에서는 이러한 문제를 해결하기 위해 비용, 무게 등의 규격 내에서 시스템의 신뢰도를 최대화하기 위한 최적의 부품을 선택하고, 중복설계 부품수와 중복형태(활성/대기중복)를 결정하는 신뢰도-중복 최적화 문제(RROP : Reliability-Redundancy Optimization Problem)를 다룬다. 인공위성 시스템의 예로는 아리랑위성 2호의 MSC(Multi-Spectral Camera) 시스템의 신뢰도 블록 다이어그램(RBD : Reliability Block Diagram)을 적용하였다[5]. 또한 NP-hard 조합최적화(Combinatorial Optimization) 문제[6]로 알려져 있는 RROP의 해법으로 병렬 개체군집최적화(PPSO) 알고리즘을 제안하였다.

## II. 본 론

### 2.1 신뢰도 확률모형

부품의 고장시간(TTF : Time To Failure) 분포 함수를  $f(t)$ , 이에 대한 누적분포함수를  $F(t)$ 라 할 때, 임의의 시점  $t$ 에 부품이 정상 작동할 확률인 신뢰도  $r(t)$ 는 식 (1)과 같이 표현된다.

$$r(t) = \int_t^{\infty} f(t)dt = 1 - \int_0^t f(t)dt = 1 - F(t) \quad (1)$$

우주부품(전자부품)의 고장률(Failure rate,  $\lambda$ ) 산정에 대한 지침 문서인 MIL-HDBK-217F[7]에 따라 고장시간 분포를 지수분포(이하  $\text{Exp}(\lambda)$ )로 고려하면, 부품의 고장시간 분포와 신뢰도는 식 (2), 식 (3)과 같다.

$$f(t) = \lambda \exp(-\lambda t) \quad (2)$$

$$r(t) = \exp(-\lambda t) \quad (3)$$

직렬 시스템의 신뢰도는 식 (4)와 같으며, 서브 시스템의 수가 증가할수록 시스템의 신뢰도는 저하된다. 따라서 시스템의 신뢰도를 향상시키기 위해 신뢰도가 낮은 서브시스템에 추가적인 예비 부품을 배치하고, 적절한 중복의 형태를 고려하여 시스템의 신뢰도를 향상시키게 된다.

$$R_s(t) = \prod_{i=1}^s R_i(t) , 0 < R_i(t) < 1.0 \quad (4)$$

예비 부품을 병렬적으로 추가하여 배치하는 서브시스템의 일반화된 형태는  $k$ -out-of- $n$  구조이며, 이는  $n$ 개의 부품 중  $k$ 개 이상이 작동되어야 함을 의미한다. 즉 배치된 예비(Secondary) 부품은  $(n-k)$ 개이다. 중복의 형태로는 활성중복과 대기중복이 고려된다. 활성중복은 모든 부품들이 동일한 가동 상태를 유지하면서 부하(Load)를 분담하고 있는 형태이며, 대기중복은 작동 중인 부품에 고장발생 시 대기 중인 예비 부품으로 부하를 전이(Switchover)하여 시스템의 기능을 유지하는 형태이다. 따라서 대기중복은 추가적으로 고장감지기(Fault detector)를 필요로 한다. 본 연구에서의 대기중복은 예비 부품들이 OFF 상태로 대기하고 있는 Cold-Standby이다.

#### 2.1.1 활성중복

활성중복으로 구성된  $k$ -out-of- $n$  서브시스템의 운용시간  $t$ 에서의 신뢰도는, 동시에 작동을 시작한  $n$ 개의 부품 중  $t$ 시점에  $k$ 개 이상의 부품이 작동하고 있을 확률로 식 (5)와 같다.

$$R(t) = \sum_{l=k}^n \binom{n}{l} [\exp(-\lambda t)]^l [1 - \exp(-\lambda t)]^{n-l} \quad (5)$$

2.1.2 대기중복(Cold-Standby)

대기중복으로 구성된  $k$ -out-of- $n$  서브시스템은  $n$ 개의 부품 중  $k$ 개가 작동을 시작하여 작동 중인 부품에 고장이 발생하면 이를 감지하고 대기 중인 부품으로 부하를 전이하는 과정이 예비 부품수  $(n-k)$ 번 만큼 반복되고,  $(n-k+1)$ 번째의 고장이 발생하면 서브시스템은 기능을 상실하게 된다. 이때, 연속적으로 작동하면서 부품의 고장을 탐지하는 고장감지기의 신뢰도  $\rho(t)$ 는 고장탐지 및 부하 전이에 성공할 확률이 된다.

또한 부품의 고장시간 분포로 고려되는  $\text{Exp}(\lambda)$ 의 무기억 속성(Memory-less property)으로 인해 작동 중인  $k$ 개의 부품은 고장시간 분포가  $\text{Exp}(k\lambda)$ 를 따르는 하나의 부품으로 고려할 수 있다. 즉 고장시간 분포가  $\text{Exp}(\lambda)$ 를 따르는 부품으로 구성된  $k$ -out-of- $n$  서브시스템은, 고장시간 분포와 신뢰도가 각각 식 (6), 식 (7)과 같은  $(n-k+1)$ 개의 부품으로 구성된 병렬 구조로 고려할 수 있다.

$$f_k(t) = k\lambda \exp(-k\lambda t) \quad (6)$$

$$r_k(t) = \exp(-k\lambda t) \quad (7)$$

따라서 Coit[4]이 제시한 식 (8)과 같은 대기중복 병렬 서브시스템의 근사 신뢰도 확률모형은 식 (9)와 같은  $k$ -out-of- $n$  서브시스템의 확률모형으로 확장할 수 있다.

$$\tilde{R}(t) = r(t) + \rho(t) \sum_{x=1}^{n-1} \int_0^t r(t-u) f_k^{(x)}(u) du \quad (8)$$

$$\tilde{R}(t) = r_k(t) + \rho(t) \sum_{x=1}^{n-k} \int_0^t r_k(t-u) f_k^{(x)}(u) du \quad (9)$$

여기서,  $f_k^{(x)}(u) : f_k(u)$ 의  $x$ -convolution

$(0, t]$  동안 고장이 발생한 부품 수를  $N(t)$ 로 정의하면,  $N(t) < n$ 인 경우  $N(t)$ 는  $\text{Exp}(k\lambda)$ 의 도착간격 시간을 갖는 재생과정으로 Poisson( $k\lambda$ ) 과정이 되어 식 (10)이 도출되며, 이를 식 (9)에 적용하면 신뢰도 확률모형을 식 (11)과 같이 도출할 수 있다.

$$\int_0^t r_k(t-u) f_k^{(x)}(u) du = \exp(-k\lambda t) \frac{(k\lambda t)^x}{x!} \quad (10)$$

$$\tilde{R}(t) = [1 - \rho(t)] \exp(-k\lambda t) + \rho(t) \left[ \exp(-k\lambda t) \sum_{x=0}^{n-k} \frac{(k\lambda t)^x}{x!} \right] \quad (11)$$

2.1.3 최적 중복전략 선택

최적 중복전략은 결정된 부품과 중복수( $n-k$ )에서 활성/대기중복 중 신뢰도가 높은 중복전략을 의미하며, 고장감지기의 신뢰도가 1.0 미만인 경우 그림 1과 같이 특정 중복수 이내에서는 대기중복이, 그 이상에서는 활성중복이 추천된다. 따라서 대기중복이 추천되는 최대 부품수를 임계 부품수( $n^c$ )로 정의한다.

본 연구에서는 Hoeffding의 부등식[8]을 이용하여 식 (5)의 하한을 식 (12), 식 (11)의 상한을 식 (12)와 같이 도출하였다. 이를 통해 임계 부품수의 상한( $n^c$ )을 식 (14)와 같이 유도하고,  $[k, n^c]$  구간에서 실제 임계 부품수( $n^c$ )를 탐색하였다.

$$R_u = 1 - \frac{1}{2} \exp \left[ -2 \frac{\{nr(t) - k + 1\}^2}{n} \right] \quad (12)$$

$$\bar{R}_c(t) = [1 - \rho(t)] r_k(t) + \rho(t) \quad (13)$$

$$\bar{n}^c = \left\lfloor \left( \frac{\beta + \sqrt{\beta^2 + 4(k-1)}}{2r(t)} \right)^2 \right\rfloor \quad (14)$$

여기서,  $\beta = \sqrt{-\frac{1}{2} \ln [2(1 - \bar{R}_c)]}$

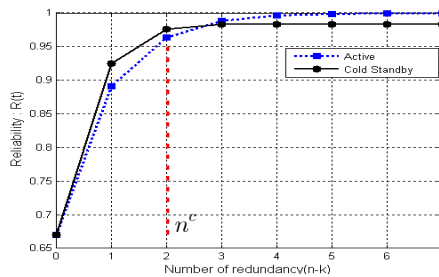


그림 1. 중복 부품수에 따른 신뢰도 변화

2.2 수리계획 모형

<표기(Notation)>

- $i$  : 서브시스템의 Index ( $i = 1, 2, \dots, s$ )
- $s$  : 서브시스템의 총 개수
- $z_i$  : 서브시스템  $i$ 에 선택된 부품의 Index ( $z_i = 1, 2, \dots, q_i$ )
- $q_i$  : 서브시스템  $i$ 의 대안 부품 수
- $R_s(t)$  : 시스템의 신뢰도
- $R_i(t)$  : 서브시스템  $i$ 의 신뢰도
- $C, W$  : 시스템 규격(비용, 무게)의 상한
- $n_{\max, i}$  : 서브시스템  $i$ 의 최대(계약) 부품 수
- $c_{i z_i}, w_{i z_i}$  : 부품  $z_i$ 의 단위 규격(비용, 무게)
- $N$  : 중복없음(No redundancy)

A : 활성중복(Active redundancy)  
 S : 대기중복(Standby redundancy)

수리계획 모형은 비선형의 신뢰도 확률모형을 대수(Logarithm) 변환하고, 새로운 결정변수( $y_{ijp}$ )를 도입하여 식 (15)부터 식 (24)까지의 혼합정수 계획 모형으로 변환하였다[4,9].

$$y_{ijp} = \begin{cases} 1, & \text{if } n_i = p \text{ and } z_i = j \text{ for subsystem } i \\ 0, & \text{otherwise} \end{cases}$$

$$\text{Max. } \ln \tilde{R}_s = \sum_{i=1}^s \ln \tilde{R}_i = \sum_{i=1}^s \sum_{j=1}^{q_i} \sum_{p=k_i}^{n_{\max,i}} \gamma_{ijp} y_{ijp} \quad (15)$$

Subject to

$$\sum_{i=1}^s \sum_{j=1}^{q_i} \sum_{p=k_i}^{n_{\max,i}} p c_{ij} y_{ijp} \leq C \quad (16)$$

$$\sum_{i=1}^s \sum_{j=1}^{q_i} \sum_{p=k_i}^{n_{\max,i}} p w_{ij} y_{ijp} \leq W \quad (17)$$

$$\sum_{j=1}^{q_i} \sum_{p=k_i}^{n_{\max,i}} y_{ijp} = 1, \forall i \quad (18)$$

$$1 \leq z_i \leq q_i, \quad z_i \in \text{Integer} \quad (19)$$

$$k_i \leq n_i \leq n_{\max,i}, \quad n_i \in \text{Integer} \quad (20)$$

$$y_{ijp} \in \{0, 1\} \quad (21)$$

For  $p = k_i, i \in N$  and

$$\cdot \gamma_{ijp} = \ln r_k(t) = k\lambda t \quad (22)$$

For  $k_i < p \leq n_{ij}^c, i \in S$  and

$$\cdot \gamma_{ijp} = \ln \left[ \{1 - \rho_i(t)\} \exp(-k_i \lambda_{iz_i} t) + \rho_i(t) \left\{ \exp(-k_i \lambda_{iz_i} t) \sum_{x=0}^{n_i - k_i} \frac{(k_i \lambda_{iz_i} t)^x}{x!} \right\} \right] \quad (23)$$

For  $n_{ij}^c < p \leq n_{\max,i}, i \in A$  and

$$\cdot \gamma_{ijp} = \ln \left[ \sum_{l=k_i}^{n_i} \binom{n_i}{l} [\exp(-\lambda_{iz_i} t)]^l \times [1 - \exp(-\lambda_{iz_i} t)]^{n_i - l} \right] \quad (24)$$

수리계획 모형의 목적함수 식 (15)는 대수변환된 시스템의 신뢰도를 최대화하는 것이며, 제약함수 식 (16)과 식 (17)은 시스템 규격의 제약(Constraint)을 의미한다. 본 연구에서는 비용과 무게로 고려하였으나, 소비전력, 부피 등의 추가적인 제약이 필요한 경우는 해당 규격에 맞는 제약식을 추가하게 된다. 식 (18)은 결정변수  $y_{ijp}$ 가 선택된 부품과 부품수에서는 1이 되고, 이외의

경우는 0이 되어 각 서브시스템에 선택할 수 있는 부품과 부품수의 조합은 유일함을 의미하고 있으며, 식 (19)와 식 (20)은 각 서브시스템에 선택할 수 있는 부품 종류와 부품수의 범위를 지정하고 있다. 식 (21)은 결정변수  $y_{ijp}$ 가 0-1(Binary) 변수임을 표현한다. 식 (22)부터 식 (24)는 결정된 부품수에 따른 중복전략과 서브시스템의 신뢰도 계산식이다. 식 (22)는 중복이 없는 경우, 식 (23)는 대기중복의 경우, 식 (24)는 활성중복이 선택되는 경우이다.

이러한 RRPOP는 NP-hard인 조합최적화 문제로 최적해 탐색에 상당한 계산시간을 필요로 한다. 따라서 최근에는 RRPOP의 해법으로 메타 휴리스틱(Meta-Heuristic) 기법을 활용한 연구가 주로 수행되고 있으며[3], 본 연구에서는 일부 Heuristic 기법이 포함된 병렬 개체군집최적화(PPSO : Parallel Particle Swarm Optimization) 알고리즘을 제안하였다.

### 2.3 PPSO 알고리즘

PSO는 Kennedy and Eberhart[10]가 1995년 제안하였으며, 새나 물고기의 무리(Swarm)가 먹이를 찾아가는 과정에서 무리 전체가 정보를 공유한다는 가설과 무리 내부의 각 개체(Particle)가 자신의 경험이 무리 전체에 공유되어 있는 정보를 기초로 행동한다는 것을 기본개념으로 하는 진화형 계산 기법이다. 각 개체는 다차원(Multi-dimensional)의 해 공간에서, 자신이 경험했던 최적 위치정보와 이웃의 정보를 이용하여 자신의 위치를 갱신(Update)하면서 해를 탐색한다.

본 연구에서는 복수의 무리를 병렬적으로 운용하고, 각 무리들은 광범위한 공간에 분산되어 독립적으로 해를 탐색하면서 발견한 양질의 정보를 무리 간에 공유함으로써 협력하는 체계를 묘사하여 그림 2와 같은 PPSO 알고리즘을 제안하였다. 이는 최적해 탐색 확률을 증대하고, 임의 무리가 국부 최적해로 조기 수렴하는 것을 방지하는데 중점을 두었다.

#### 2.3.1 개체 위치벡터 표현

PPSO 알고리즘의 첫 번째 단계는 문제의 잠재해(Potential solution)를 개체의 위치벡터로 표현하는 것이다. 본 연구에서는 개체의 위치벡터를 각 서브시스템에 선택되는 대안 부품의 번호와 부품수를 정수 문자열(Integer string)의 형태로 그림 3과 같이 표현하였다. 즉 위치벡터의  $i$  열과  $s+i$ 열은 각각 서브시스템  $i$ 의 선택된 부품 번호와 부품수에 대한 정보를 담고 있다. 서브시

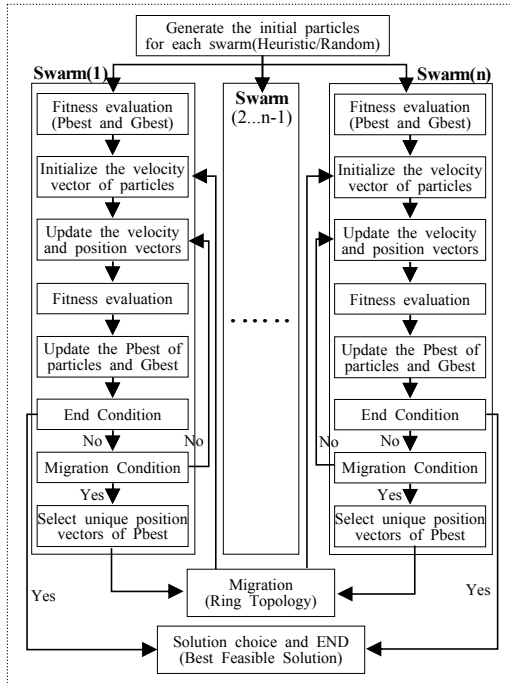


그림 2. PPSO 알고리즘 수행절차

Sub <sup>l</sup> (i) → x <sub>i</sub>	부품의 번호					부품수				
	1	2	...	s-1	s	1	2	...	s-1	s
	1 ≤ z <sub>i</sub> ≤ q <sub>i</sub>					k <sub>i</sub> ≤ n <sub>i</sub> ≤ n <sub>max,i</sub>				

그림 3. 개체의 위치벡터 표현

스텝별 부품 번호(z<sub>i</sub>)의 범위는 1부터 대안 부품 수(q<sub>i</sub>)까지, 부품수의 범위는 최소 부품수(k<sub>i</sub>)부터 최대 부품수(n<sub>max,i</sub>)까지이다.

2.3.2 초기 모집단 생성

초기 모집단 생성에서는 초기 개체들을 실행 가능영역(Feasible region) 내에서 생성하도록 하는 Heuristic 기법과 주로 규격의 제약을 초과하여 생성되는 임의생성 기법을 혼용하였다. 이는 개체들이 실행가능영역의 내·외에 분포되어 개체들 간의 교호작용을 통해 규격의 경계영역을 충분히 탐색하도록 하기 위함이다. 초기 개체의 생성 비율은 Heuristic 기법 70%, 임의생성 기법 30%를 적용하였다.

2.3.2.1 Heuristic 기법

Heuristic 기법은 가용한 규격을 각 서브시스템에 할당하고, 할당된 자원으로 서브시스템의 신뢰도를 최대화하는 부품과 중복수의 조합을 결

정하여 초기 개체로 반영하였다.

1단계는, 식 (25)와 같은 부품의 상대적 기여도를 계산하고, 식 (26)과 같이 정규화하여 해당 부품이 선택될 확률로 고려하였다. 즉 상대적으로 적은 비용과 무게가 소요되면서 신뢰도가 높은 부품이 선택될 확률이 높다. 이후 [0, 1] 구간의 Uniform 난수를 통해 각 서브시스템에 사용할 부품을 선택하고, 최소 부품수 배치에 필요한 비용, 무게를 식 (27)과 같이 할당하였다.

$$d_{iz_i} = \frac{r_{iz_i}}{(c_{iz_i}/C) + (w_{iz_i}/W)} \quad (25)$$

$$p_{iz_i} = d_{iz_i} / \sum_{z_i=1}^{q_i} d_{iz_i} \quad (26)$$

$$c_{a,i}^1 = c_{iz_i} \times k_i, \quad w_{a,i}^1 = w_{iz_i} \times k_i \quad (27)$$

2단계는, 1단계에서 할당 후 남은 비용과 무게를 각 서브시스템에 임의 할당하였다. 할당 방법은 [1, s]에서 정수형 Uniform 난수로 서브시스템을 선택하고, 잔여 비용을 c<sub>r</sub>, 잔여 무게를 w<sub>r</sub>이라 할 때, 각각 [1, c<sub>r</sub>], [1, w<sub>r</sub>]에서 정수형 Uniform 난수로 할당할 비용(c<sub>a,i</sub><sup>2</sup>)과 무게(w<sub>a,i</sub><sup>2</sup>)를 결정하였다. 이러한 절차는 잔여 비용과 무게가 소진될 때까지 반복하였다.

3단계에서는, 각 서브시스템에 할당된 비용과 무게로 신뢰도를 최대화하는 부품과 부품수의 조합을 초기 개체로 반영하였으며, 이에 대한 의사 코드(Pseudo code)는 그림 4와 같다.

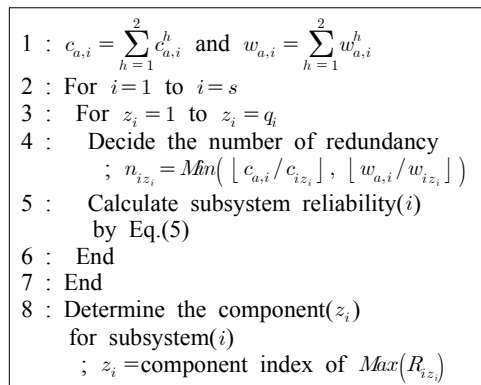


그림 4. Heuristic 기법 3단계 수행절차

2.3.2.2 임의생성 기법

임의생성 기법은 각 서브시스템에 사용할 부품 번호와 부품수를 각각 [1, q<sub>i</sub>]와 [1, n<sub>max,i</sub>]에서 정수형 Uniform 난수로 생성하였다. 개체들

은 주로 비용과 무게의 규격을 초과하여 적합도 (Fitness)는 좋지 않으나, Heuristic 기법으로 생성된 개체들에 비해 다양성이 우수하다.

2.3.3 적합도 평가

적합도 평가는 개체 위치벡터의 정보(부품 번호, 부품수)를 바탕으로 시스템의 신뢰도를 계산하고, 제약의 충족 여부에 따라 벌금(Penalty)을 부여하여 해의 품질을 평가하는 단계이다.

시스템의 신뢰도는 가용한 규격을 충실히 활용하도록 향상될 수 있으므로, 규격의 인근에 최적해가 존재할 확률이 높다. 따라서 규격을 초과하더라도 규격의 인근에 위치한 개체들은 최적해에 근사한 정보를 포함하고 있을 가능성이 높다. 따라서 본 연구에서는 규격의 초과에 대한 허용치(NFT : Near Feasible Threshold)를 부여하여, 유용한 정보를 포함한 개체들이 배제되는 것을 방지하고, 식 (28)과 같이 계산의 진행에 따라 허용치를 감소시켜 규격을 엄격하게 적용하는 식 (29)와 같은 적응형 벌금함수를 적용하였다[11].

<표기(Notation)>

- $R_p(\vec{x}_i)$  : Particle  $i$ 의 적합도 평가 값
- $R(x_i)$  : Particle  $i$ 의 신뢰도
- $R_{all}$  : 전체  $R(x_i)$  중 최대값
- $R_{fea}$  : 규격을 만족하는  $R(x_i)$  중 최대값
- $d_j(\vec{x}, B_j)$  : Particle  $i$ 가 초과한 규격  $j$ 의 양
- $NFT_j$  :  $g$ 회 반복시 규격  $j$ 의 초과 허용치
- $NFT_{o,j}$  : 규격  $j$ 의 초기 초과 허용치
- $\theta_j$  : 규격  $j$ 에 대한 모수 ;  $j = c, w$
- $\alpha, g$  : 감쇄 상수, 반복(Iteration) 횟수

$$NFT_j = \frac{NFT_{o,j}}{1 + \alpha g} \quad (28)$$

$$R_p(\vec{x}_i) = R(\vec{x}_i) - (R_{all} - R_{fea}) \sum_{j=1}^m \left[ \frac{d_j(\vec{x}, B)}{NFT_j} \right]^{\theta_j} \quad (29)$$

2.3.4 속도/위치벡터 갱신

개체의 운동방정식은 식 (30)과 같은 Shi and Eberhart의 모형을 사용하였으며[12], RROP는 이산적인(Discrete) 해 공간을 갖게 되므로 식 (31)과 같이 위치벡터를 반올림하여 정수화하였다. 그림 5는 속도/위치벡터의 갱신 과정을 보여준다.

<표기(Notation)>

- $i$  : 개체의 index
- $\vec{v}_i, \vec{x}_i$  : 개체  $i$ 의 속도, 위치벡터

$\vec{U}(0, \phi_j)$  :  $0 \sim \phi_j$  구간의 Uniform분포의 난수

$w$  : Inertia weight ;  $\vec{U}(0, 1)$

$\vec{p}_i$  : 개체  $i$ 가 경험한 최적 위치벡터(Pbest)

$\vec{p}_g$  : 무리 전체에서의 최적 위치벡터(Gbest)

$\otimes$  : Component-wise multiplication

$$\vec{v}_{i,t+1} = \omega \vec{v}_{i,t} + \left[ \vec{U}(0, \phi_1) \otimes (\vec{p}_i - \vec{x}_{i,t}) + \vec{U}(0, \phi_2) \otimes (\vec{p}_g - \vec{x}_{i,t}) \right] \quad (30)$$

$$\vec{x}_i(t+1) = \text{round} \{ \vec{x}_i(t) + \vec{v}_i(t+1) \} \quad (31)$$

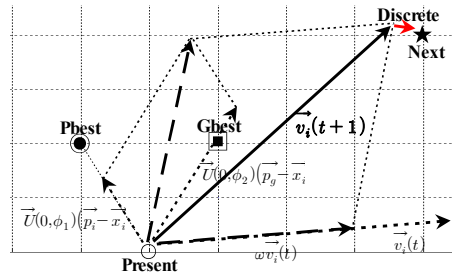


그림 5. 속도/위치벡터 갱신 과정

PSO에서 개체가 급격하게 발산하는 현상[13]을 방지하기 위해 속도의 상·하한( $\pm V_{max}$ )을 설정하였으며, 계산이 진행될수록 식 (32)와 같이  $V_{max}$ 를 감소시켜 이동 속도를 통제함으로써 국부 영역에서는 해를 세밀하게 탐색하도록 하였다. 또한 위치벡터는 대안 부품수( $q_i$ )와 최대 부품수( $n_{max,i}$ ) 제약을 초과하지 못하도록 제한하였다.

$$V_{max,t+1} = (V_{max,t} - 0.5) \exp \left[ -\frac{0.1t}{T} V_{max,t} \right] + 0.5 \quad (32)$$

2.3.5 이주(Migration)

이주는 무리 간에 주기적으로 적합도가 우수한 개체들을 교환하여 양질의 정보를 공유하는 과정이다. 무리들이 분산된 공간에서 독립적으로 해를 탐색한 후 정보를 교환함으로써 최적해 탐색 확률을 높이는데 목적이 있다. 이주 구조는 Fig. 6과 같은 Ring Topology를 적용하였다.

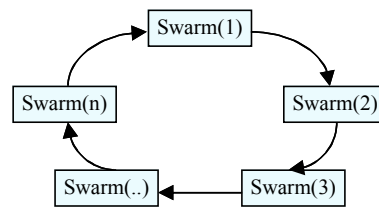


Fig. 6. Ring Topology

### 2.4 최적화 예제(MSC 시스템)

아리랑위성 2호의 MSC(Multi-Spectral Camera) 시스템은 1m급 흑백영상과 4m급 컬러영상을 제공하고 있으며, 획득된 영상은 지도제작, 지리정보시스템 구축 등에 활용되고 있다. 서브시스템으로는 광학서브시스템 EOS(Electro-Optical Subsystem), 위성본체와의 통신 및 탑재체 동작제어를 수행하는 PMU(Payload Management Unit)와 EOS가 획득한 영상정보를 지상으로 송신하는 PDTs (Payload Data Transmission Unit)로 구성되어 있다[14]. MSC 시스템의 신뢰도 블록 다이어그램은 그림 7과 같다[5].

#### 2.4.1 수치실험 조건

수치실험에 적용한 PPSO 알고리즘과 적응형 벌금함수의 파라미터는 표 1과 같다.

표 1. PPSO 및 적응형 벌금함수 파라미터

PSO		Parallel		Fitness	
Para.	Value	Para.	Value	Para.	Value
Particles	200	Swarm	5	$\psi$	1
Iteration	50	Mig_Freq	10	$\theta_c$	2
$\phi_1/\phi_2$	2/2	Mig_Frac	0.3	$\theta_w$	1

RROP에서 고려하는 Element별 부품수의 범위는 표 2와 같다. 최소 부품수( $k_i$ )는 그림 7의 현재 설계를 기준으로 하였으며, 최대 부품수( $n_{max,i}$ )는 OM을 제외한 나머지 Element들은 현재 설계 부품수의 2배까지 고려하였다.

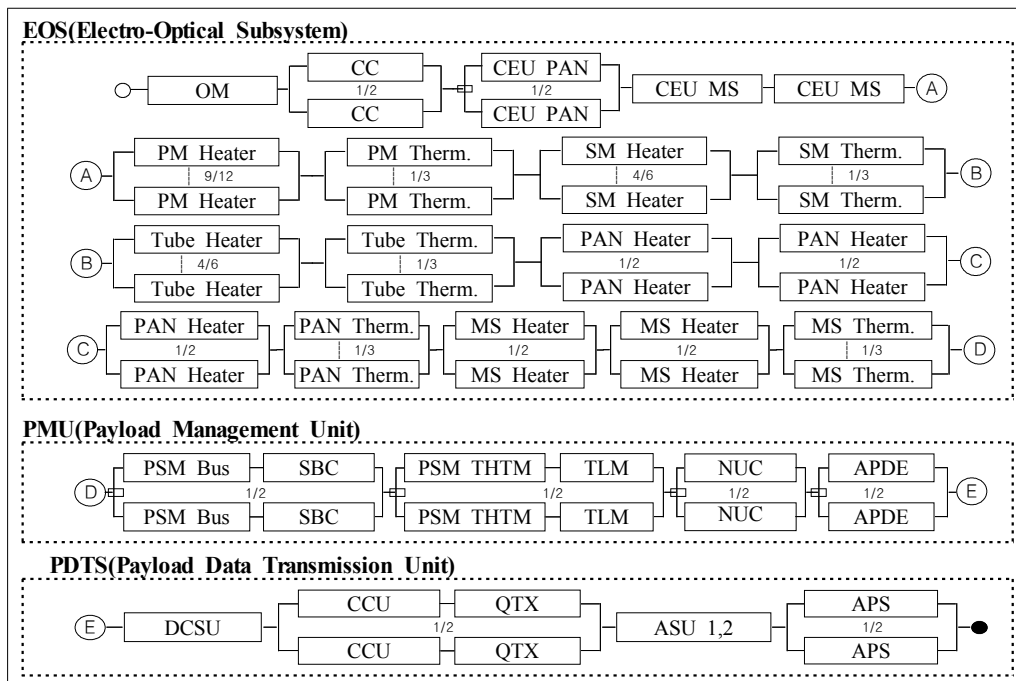
대안 부품의 데이터는 Fyffe et al.[15]의 예제를 사용하였으며, 부품의 신뢰도는 아리랑위성 2호의 설계수명(3년)을 고려하여 평균 고장률로 환산한 표 3을 적용하였다. 단, 그림 7에서 PSM Bus와 SBC, PSM THTM과 TLM, CCU와 QTX Element는 각 1개의 Set 부품으로 고려하였으며, 또한 CEU MS(2개), HEATER류(8개), THERM류(5개) Element들은 동일한 대안 부품을 갖는다. Element별 최소/최대 부품수, 대안 부품종류와

표 2. Element별  $k_i$  및  $n_{max,i}$

No.(i)	1	2	3	4	5	6	7	8	9	10	11	12	13
$k_i$	1	1	1	1	1	9	1	4	1	4	1	1	1
$n_{max,i}$	1	4	4	2	2	24	6	12	6	12	6	4	4

No.(i)	14	15	16	17	18	19	20	21	22	23	24	25	26
$k_i$	1	1	1	1	1	1	1	1	1	1	1	1	1
$n_{max,i}$	4	6	4	4	6	4	4	4	4	2	4	2	4



※ Note 각 Element 중앙  $k/n$  :  $k$ -out-of- $n$  구조  
연결 노드의 □ 기호 : 현재 Cold-Standby로 설계

그림 7. MSC 시스템의 신뢰도 블록 다이어그램

표 3. MSC 시스템 최적화 예제 수치실험 데이터

Subsystem( <i>i</i> )	Choice 1			Choice 2			Choice 3			Choice 4		
	$\lambda_{i,z_i}$	$c_{i,z_i}$	$w_{i,z_i}$	$\lambda_{i,z_i}$	$c_{i,z_i}$	$w_{i,z_i}$	$\lambda_{i,z_i}$	$c_{i,z_i}$	$w_{i,z_i}$	$\lambda_{i,z_i}$	$c_{i,z_i}$	$w_{i,z_i}$
OM	4,010	1	3	2,762	1	4	3,589	2	2	1,952	2	5
CC	1,952	2	8	2,355	1	10	2,762	1	9	-	-	-
CEU PAN	6,185	2	7	4,010	3	5	5,300	1	6	3,173	4	4
CEU MS(2)	7,091	3	5	5,300	4	6	6,185	5	4	-	-	-
HEATER(8)	2,355	2	4	2,762	2	3	1,952	3	5	-	-	-
THERM.(5)	383.0	3	5	769.0	3	4	1,160	2	5	1,554	2	4
PSM Bus+SBC	3,589	4	7	3,173	4	8	2,355	5	9	-	-	-
PSM THTM+TLM	8,019	3	4	4,010	5	7	3,589	6	6	-	-	-
NUC	1,160	2	8	383.0	3	9	1,554	4	7	3,589	3	8
APDE	7,091	4	6	6,185	4	5	4,010	5	6	-	-	-
DCSU	2,355	3	5	1,952	4	6	1,554	5	6	-	-	-
CCU+QTX	8,970	2	4	7,552	3	5	6,185	4	6	4,010	5	7
ASU	769.0	2	5	383.0	3	5	1,160	2	6	-	-	-
APS	4,010	4	6	3,173	4	7	1,952	5	6	382.5	6	9

※  $\lambda_{i,z_i}$  : FIT( $10^{-9}$ failure/hour), Failure rate of fault detector=383 FIT( $\forall i$ )

표 4. 설계수명에 따라 추천되는 설계 구조(PPSO 알고리즘의 최선해)

No.	Life	Redundancy Strategy												Used		Reli-ability
		Choced component( $z_i$ )												C	W	
Number of components( $n_i$ )																
1	3-year	N A A S S A S A S A S A A A S A A S S S A S A	199	400	0.91474											
		4 3 3 2 2 2 3 1 3 2 3 1 1 1 3 2 1 3 2 1 1 2 1 1 1 4														
		1 3 4 2 2 13 2 7 2 7 2 3 3 3 2 3 3 2 3 4 2 3 2 4 2 2														
2	5-year	N A A S S A S A S A S A A A S A A S S S S S S A	200	399	0.81910											
		4 2 3 2 2 2 3 1 3 2 3 1 2 1 3 2 2 3 2 1 1 2 1 1 1 4														
		1 3 4 2 2 14 2 7 2 8 2 3 3 3 2 3 3 2 3 3 2 3 2 4 2 2														
3	7-year	N A S S S A S A S A S A A A S A S S S S S A S	200	397	0.69985											
		4 2 3 2 2 2 3 1 3 2 3 1 1 1 3 1 1 3 3 1 1 2 1 1 1 3														
		1 3 3 2 2 15 2 7 2 8 2 3 3 3 2 3 3 2 2 4 2 3 2 4 2 2														
4	10-year	N S S S S A S A S A S A S A S S A S S S S S S A N	200	400	0.51211											
		4 3 3 2 2 2 3 1 3 1 3 1 2 1 3 2 1 3 3 1 1 2 1 1 1 4														
		1 3 3 2 2 16 2 8 2 8 2 3 3 3 2 3 3 2 2 4 2 3 2 4 2 1														

중복전략의 선택을 고려할 때, 예제의 해 탐색 공간은  $3.28 \times 10^{37}$ 개를 초과한다.

2.4.2 결과 분석

수치실험에서 시스템의 규격은 비용 규격 200, 무게 규격은 400으로 고정하고, 인공위성의 설계 수명을 3년, 5년, 7년, 10년으로 변경하면서 실험 하였다.

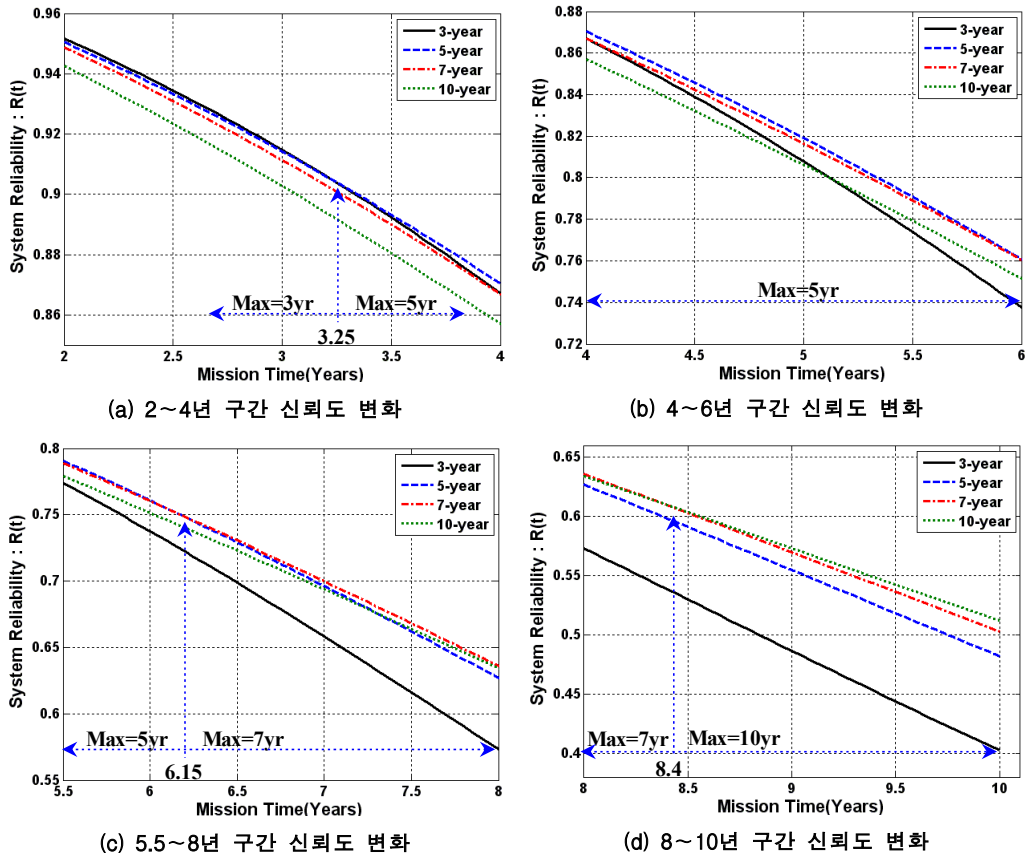
설계수명에 따라 신뢰도가 최대화되는 설계 구조는 표 4(PPSO 최선해)와 같이 나타났다. 표 4의 해 표현에서 첫 번째 줄은 각 Element에 추천되는 중복전략을 나타내며, 두 번째 줄은 표 3의 대안 부품 중 선택된 부품의 Index를 표시하고 있다. 세 번째 줄은 결정된 부품수이다. 표 4에서는 동일한 규격에서도 설계수명에 따라 추천되는 설계 구조가 상이함을 알 수 있다.

또한 각 설계수명에서 추천된 설계 구조를 바

탕으로 운용기간에 따른 신뢰도 변화를 해석한 결과는 그림 8과 같다. 각 설계수명에서 추천된 구조는 해당 운용기간에서는 최선이지만, 설계수명의 이전 또는 이후에서는 최선이 아님을 확인할 수 있다. 즉 설계수명 10년에 추천된 구조는 약 5년까지 최저 신뢰도를 보이며, 5년 이후에는 설계수명 3년에 추천된 구조가 최저 신뢰도를 보이고 있다. 세부적으로 살펴보면, 설계수명별 추천된 4가지 구조 중 최대 신뢰도를 보이는 운용 기간은 설계수명 3년의 구조는 3.25년까지, 설계수명 5년의 구조는 3.25년~6.15년, 설계수명 7년의 구조는 6.15년~8.4년, 설계수명 10년의 구조는 8.4년 이상에서 최대 신뢰도를 보이고 있다. 즉 최대 신뢰도를 보이는 구조는 특정 운용기간에서 변경된다.

또한 아리랑위성 2호처럼 임무기간의 연장가능성이 있어 설계수명이 3년~7년의 구간으로





설계수명 운용기간	3년	5년	7년	10년
3년	<b>0.91473</b>	0.91422	0.91132	<u>0.90270</u>
5년	0.80786	<b>0.81910</b>	0.81639	<u>0.80620</u>
7년	<u>0.65823</u>	0.69629	<b>0.69985</b>	0.69376
10년	<u>0.40236</u>	0.48197	0.50291	<b>0.51212</b>

그림 8. 운용기간에 따른 신뢰도 변화와 임계점

설정된다면, 8과 같이 각 설계수명 구조에 대해 신뢰도를 해석하고 절충(Trade-off)을 통해 최적 구조를 결정할 필요가 있다. 그림 8을 본다면 설계수명 5년 또는 7년의 설계구조가 추천될 수 있을 것으로 판단된다. 따라서 신뢰도는 인공위성의 수명이 결정되는 기획 또는 설계단계의 초기부터 충분히 고려되어야 한다.

### III. 결 론

본 연구에서는 동일한 기능을 수행하지만, 평

균 고장률 등의 특성이 상이한 대안 부품들이 존재할 때, 시스템의 신뢰도를 최대화하기 위해 최적 부품을 선택하고, 부품수와 중복전략(Active or Cold-Standby)을 결정하는 RROP를 다루었다. 또한 NP-hard인 조합최적화 문제로 분류되는 RROP의 해법으로 PPSO 알고리즘을 제안하였다.

또한 아리랑위성 2호의 MSC 시스템의 설계 구조를 바탕으로 RROP를 구성하고, 동일한 규격 내에서 설계수명을 달리하여 수치실험을 수행하였다. 수치실험의 결과로부터 설계수명에 따라 최적의 설계 구조가 변경됨을 확인할 수 있었으며, 운용기간에 따른 신뢰도 변화의 해석을 통해

수명이 구간으로 지정되는 경우에서의 최적 설계 선정에 대해 검토하였다.

결론적으로, 인공위성의 임무에 필수적인 최소의 부품 배치에 대한 설계 후에는 인공위성의 신뢰도를 향상시키기 위한 중복설계를 고려하게 되며, 이때 RROP와 PPSO 알고리즘과 같은 메타휴리스틱(Meta-Heuristic) 기법이 적용될 수 있다. 따라서 본 연구가 복잡한 인공위성 시스템 설계자에게 중요한 가이드라인(Guideline)을 제공하는 도구가 될 수 있기를 기대하며, 신뢰도는 인공위성의 수명이 결정되는 시기부터 충분히 고려되어야 함을 강조하고자 한다.

### 참고문헌

- 1) Elsayed, E. A., "Reliability Engineering", Addison Wesley Longman, Inc., 1996.
- 2) 장영근, 이동호, "인공위성 시스템 설계공학", 경문사, 1999.
- 3) Kuo, W. and Wan, R., "Recent Advances in Optimal Reliability Allocation", IEEE Transactions on Systems, Man, and Cybernetics Part A : Systems and Humans, Vol. 37(2), 2007, pp143~156
- 4) Coit, D. W., "Maximization of system reliability with a choice of redundancy strategies", IIE Transaction, Vol. 35, 2003, pp535~543.
- 5) Kim, Y. S., Lee, D. K. Lee, C. H., and Woo, S. H., "RELIABILITY ANALYSIS OF THE MSC SYSTEM", J. Astron. Space Sci., Vol. 20(3), 2003, pp217~226.
- 6) Chern, M. S., "On the Computational Complexity of Reliability Redundancy Allocation in a Series System", Operation Research Letters, Vol. 11(5), 1992, pp309~315.
- 7) Lockheed Martin, "MIL-HDBK-217F : Reliability Prediction of Electronic Equipment", Department of Defense, USA, 1991
- 8) Hoeffding, W., "Probability inequalities for sums of bounded random variables", Journal of the American Statistical Association, Vol. 58(301), 1963, pp13-30
- 9) Misra, K. B., and Sharma, U., "Reliability optimization of a system by zero-one programming", Microelectronics and Reliability, Vol. 12(3), 1973, pp229-233
- 10) Kennedy, J., and Eberhart, R. C., "Particle swarm optimization", Proceedings of IEEE international Conference on Neural Networks, 1995, pp1942-1948
- 11) Coit, D. W. and Smith, A. E., "Adaptive Penalty Methods for Genetic Optimization of Constrained Combinatorial Problems", INFORMS Journal on Computing, Vol. 8, No. 2, 1996, pp173~182.
- 12) Shi, Y., and Eberhart, R. C., "A modified particle swarm optimizer", In Proceedings of the IEEE international conference on evolutionary computation, 1998, pp69-73
- 13) Poli, R., Kennedy, J., and Blackwell, T., "Particle swarm optimization(An overview)", Swarm Intell, 2007, pp33~57.
- 14) 공종필, 허행필, 김영선, 박종익, 장영준, "MSC(Multi-Spectral Camera) 열제어 시스템 소개", 항공우주기술, Vol. 4(2), 2005, pp107~116.
- 15) Fyffe, D. E., Hines, W. W., and Lee, N. K., "System Reliability Allocation and a Computational Algorithm", Operation Research, Vol. 17(2), 1968, pp64~69.