

# 動的計劃을 利用한 最適信賴度 設計 (Optimum Redundant Design by Dynamic Programming)

朴 泰 堯\*

## ABSTRACT

In military applications of equipment, it is of the utmost importance that the equipment operates without breakdown for a specified length of time. We can increase the reliability of a system by adding redundant components to the basic nonredundant system. This study introduces a dynamic programming algorithm of finding optimum redundant design that maximizes the system reliability with constraints.

### 1. 序 論

一般的으로 시스템의 信賴度를 높이기 爲해서는 그림 1 과 같이 各 部品에 對해 追加 部品을 並列로 連結하는 方法이 使用된다.

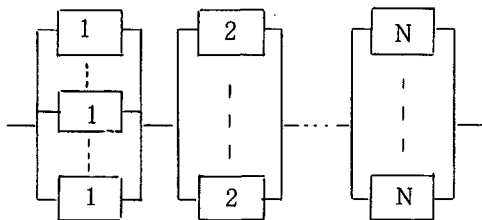


그림 1. 直列 — 並列시스템

그러나 實際問題에 있어서는 並列連結된 部品中 하나만 故障나면 그 並列連結 部分이 作動하지 않고 따라서 全体 시스템이 作動하지 않는 境遇가 많다. 이 때는 並列連結 部分에 스위치와 같은 裝置가 必要하게 되며 이 스위치

에 對한 故障率이 考慮되어야 한다. 이런 境遇 最大의 信賴度를 얻기 爲해서는 그림 2 의 例와 같은 直列 — 並列 — 直列 (series - parallel - series) 시스템이 되어야 한다.

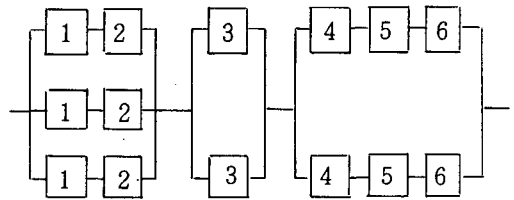


그림 2. 直列 — 並列 — 直列시스템

### 2. 動的 計劃 模型

#### 가. 最大信賴度 模型

그림 2에서 部品  $i$ 로부터  $j$ 까지를 直列로 連結한 서브시스템  $m$ 個를 並列로 連結한 것을  $(i, j, m)$ 이라 하면  $N$ 種類의 部品으로 構成된 시스템은

\* 國防管理 研究所

$$D = \{ (i_1, j_1; m_1), \dots, (i_k, j_k; m_k) \}$$

$$i_1 = 1, j_k = N, j_n + 1 = i_{n+1}$$

이 되고 이 시스템의 신뢰도는

$$R(D) = \prod_{(i,j;m) \in D} R(i, j; m)$$

이다. 여기서 최대 신뢰도 문제는  $R(D)$  를 최대로 하는  $D$ 를 찾는 것이다.

이제 부품 1에서  $j$ 까지로 구성된 서브시스템을  $D_j$ 라 하고 이 서브시스템의最適設計를  $D_j^*$ , 이때의 신뢰도를  $f_j$ 라 하면

$$f_0 = 1$$

$$f_j = \max_{1 \leq i \leq j} (\max_m R(i, j; m) \cdot f_{i-1}), 1 \leq j \leq N$$

이 된다.

#### 나. 制限條件下에서의最適化

實際적인問題에서 어떤 시스템을設計할 때는費用, 重量等 資源의制限이 따르게 된다.

$r$ 個 資源의制限이 있을 때 資源  $k$ 의限度를  $C_k$ ,  $i$  부품에 必要的 資源  $k$ 의 量을  $C_{ki}$ 라 하면, 서브시스템  $(i, j; m)$ 에 必要的 資源  $k$ 의 量은

$$C_k(i, j; m) = \sum_{h=i}^h C_{kh} \cdot m$$

이며, 全体시스템  $D$ 에 必要的 資源  $k$ 의 量은

$$C_k(D) = \sum_{(i,j;m) \in D} C_k(i, j; m)$$

이다. 따라서 制限條件下에서의最適化問題는

$$\text{maximize } \prod_{(i,j;m) \in D} R(i, j; m)$$

$$\text{subject to } C_k(D) \leq C_k, k=1, \dots, r$$

이 된다.

어떤 서브시스템  $D_j$ 에 대해 制限條件을 滿足시키는 두가지設計를  $D_j^1, D_j^2$ 라 할 때

$$C_k(D_j^1) \leq C_k(D_j^2), k=1, \dots, r$$

$$R(D_j^1) \geq R(D_j^2)$$

일 때  $D_j^1$ 은  $D_j^2$ 를 支配(dominate)하며, 式에서 하나 以上の 不等關係가 完全(strict)할 때 完全支配(strictly dominate)한다.

이제 各 段階別로  $D_j (j=1, \dots, N)$ 의 支配系列(dominating sequence)  $S_j$ 를 만들면 이 系列内の設計는 다른設計에 完全支配되지 않으며 다른 모든設計는 支配系列  $S_j$ 의

設計에 支配된다. 이때 支配系列  $S_N$ 中에서 가장 信賴도가 높은設計가 最適設計가 된다.

#### 다. 計算量의 減縮

萬一 우리가 實現可能한設計의 信賴도  $R_f$ 를 찾을 수 있다면,  $R(D_j) < R_f$ 인設計  $D_j$ 는 支配系列에서 除外시킬 수 있다.

이런  $R_f$ 를 求하기 爲해 逆動的計劃(reverse dynamic programming) 模型을 만든다.

$$f'_{N+1} = 1$$

$$f'_h = \max_{h \leq k \leq N} (\max_m R(h, k; m) \cdot f'_{k+1}), 1 \leq h \leq N$$

여기서  $f'_h$ 는 部品  $h$ 로부터  $N$ 까지로 구성된 서브시스템의 最大 信賴도이며 이때의設計를  $D'_h$ 라 한다.

$$R'(D_j) = R(D_j) \cdot f'_{j+1}$$

이라면,  $R'(D_j) < R_f$ 인設計  $D_j$ 는 支配系列에서 除外시킬 수 있다.

이제  $C'_k(h) = \sum_{i=h}^i C_{ki}$ 라 하면 各設計  $D_j$ 에 대해  $C_k(D_j) + C'_k(j+1)$ 는  $D_j$ 를 包含하는 全体 시스템의  $k$ 資源 最少所要量이 된다. 따라서

$$C_k(D_j) + C'_k(j+1) > C_k$$

인設計  $D_j$ 는 除外시킬 수 있다.

萬一 모든 資源에 대해

$$C_k(D_j) + C'_k(j+1) \leq C_k$$

이라면  $D_j$ 를 包含하는 全体 시스템設計는 實現可能한設計가 되며, 따라서  $R(D_j) \cdot f'_{j+1}$ 은 實現可能한 信賴도가 되며, 이때

$$R(D_j) \cdot f'_{j+1} > R_f$$

이면 이 값으로  $R_f$ 를 代置할 수 있다.

#### 라. 演算法

制限條件下에서의最適化問題

$$\text{maximize } \prod_{(i,j;m) \in D} R(i, j; m)$$

$$\text{subject to } C_k(D) \leq C_k, k=1, \dots, r$$

를 풀기 爲한 演算法은 다음과 같다.

(1) 初期條件;

$S_0$ 의 構成要素는  $D_0 = \phi$ 이며,  $C_k(D_0) = 0, k = 1, \dots, r, R(D_0) = 1, R_f = 0$ 이다.

(2)  $j = 1$

(3)  $i = j$

(4)  $m = m^0$  ( $m^0$ 는  $R(i, j; m)$ 을 最大로 하는  $m$ 의 값)

(5)  $(i-1)$  段階의 支配系列  $S_{i-1}$ 을 構成하는 모든 設計  $D_{i-1}$ 에 對해  $D_j = \{D_{i-1}, (i, j; m)\}$ 을 求한다.

(6)  $D_j$ 가 制限條件을 滿足시키지 않거나  $R(D_j) \cdot f'_{j+1} < R_f$ 이면 (11)로, 그렇지 않으면 (7)로 간다.

(7)  $C_k(D_j) + C_k(D'_{j+1}) \leq C_k$ 이고  $R(D_j) \cdot f'_{j+1} > R_f$ 이면  $R_f = R(D_j) \cdot f'_{j+1}$ 로 놓고 (8)로, 그렇지 않으면 (9)로 간다.

(8) 모든 支配系列에서  $R(D_h) \cdot f'_{h+1} < R_f$ 인 모든  $D_h$ 를 除外시킨다.

(9)  $D_j$ 가  $S_j$ 의 다른 設計에 支配되지 않으면  $D_j$ 를 配列하고 支配되면 (11)로 간다.

(10)  $S_j$ 의 設計中  $D_j$ 에 支配되는 設計를 除外시킨다.

(11)  $S_{i-1}$ 에 構成要素가 더 있을 境遇 (5)로 간다.

(12)  $R(D_j) \cdot f'_{j+1} = R_f$ 이면 (15)로, 그렇지 않으면  $m = m - 1$ 로 한다.<sup>1)</sup>

(13)  $m = 1$ 이고  $i \neq j$ 이면 (15)로 간다.<sup>2)</sup>

(14)  $m = 0$ 이면 (5)로 간다.

(15)  $i = j - 1$ 로 하고  $i > 0$ 이면 (4)로 간다.

(16)  $j = j + 1$ 로 하고  $j \leq N$ 이면 (3)으로 간다.

(17)  $f_N = R(D_N^*)$ 가 最大 信賴도가 된다.

### 3. 事例 研究

그림 3과 같이 5개의 部品으로 構成된 시스템이 있다. 各 部品  $i$ 의 故障率은  $q_i$ 이고 部品  $i$ 에서의 連結스위치의 故障率은  $\alpha_i$ 이다.



그림 3. 事例研究의 基本시스템

서비스시스템  $(i, j; m)$ 에서 連結部分  $j$ 의 故障率을  $\alpha_j(m-1)$ 이라면 서비스시스템  $(i, j; m)$ 의 信賴도는

$$R(i, j; m) = \left[ 1 - \left\{ 1 - \prod_{k=i}^{k=j} (1 - q_k) \right\}^m \right] \cdot \{ 1 - \alpha_j(m-1) \}$$

이 된다.

시스템 設計에 있어서 費用과 重量의 制限이 있다. 部品  $i$ 의 費用과 重量은 各各  $C_{i1}, C_{2i}$ 이며 스위치의 費用과 重量은 各各  $CS_{1i}, CS_{2i}$ 이다. 서비스시스템  $(i, j; m)$ 의  $j$ 連結部分의 費用 및 重量을  $CS_{kj} \cdot (m-1)$ 이라 하면

$$C_k(i, j; m) = \sum_{h=i}^{h=j} C_{kh} \cdot m + CS_{kj} \cdot (m-1)$$

이 된다.

事例研究의 最適化問題는

$$\begin{aligned} &\text{maximize } \Pi(i, j; m) \in D \quad R(i, j; m) \\ &\text{subject to } \sum_{(i, j; m) \in D} C_k(i, j; m) \leq C_k, \\ & \quad \quad \quad k = 1, 2 \end{aligned}$$

이며, 人力資料는 表1과 같다.

問題 解決을 爲해 첫째로 表2와 같이 最大 信賴도를 求한다. 둘째로 表4는 制限條件下의 最適設計를 計算量의 減縮없이 求한 것이 며 이 때 80개의 構成要素가 있다. 다음 이 構成要素의 數를 줄이기 爲해 表3과 같이 逆動的 計劃法에 依한 最大 信賴도를 求한다. 表5는 앞에서 說明한 演算法에 依한 結果이며 9개의 構成要素로 되어 있다. 여기서 Jensen의 方法과 本 研究의 方法을 比較해 보면 表6과 같이 支配系列에 配列된 要素의 數는 28個에서 18個로, 配列後 除外된 要素의 數는 19個에서 9個로, 最大 配列數는 10個에서 9個로,  $R_f$ 交替數는 10個에서 4個로 減少되었다.

註: 1) Jensen의 方法은  $m$ 을 1부터 증가시킴으로써 (12)번 順序가 없으며 支配系列의 構成要素가 많아진다.

2)  $m = 1$ 이고  $i \neq j$ 인 設計는 實際로  $m = 1, i = j$ 인 設計와 同一하다. Jensen의 方法에는 考慮되지 않음.

表 1. 人力資料

OPTIMIZATION OF SERIES - PARALLEL - SERIES NETWORKS

TABLE(I) INPUT DATA

I	Q	A	COST (1)		COST (2)	
			C (1)	CS (1)	C (2)	CS (2)
1	0.050	0.010	10.00	1.00	5.00	1.00
2	0.100	0.015	7.00	1.00	8.00	1.00
3	0.150	0.010	6.00	1.00	7.00	1.00
4	0.100	0.015	8.00	1.00	6.00	1.00
5	0.200	0.020	5.00	1.00	10.00	1.00
C MAX			100.00		100.00	

表 2. 最大 信賴度

TABLE (II) MAXIMUM RELIABILITY

J	I	M	F(J)	FO(J)	I (J)	M(J)
1	1	2	0.98752			
	OPT. DESIGN AT STAGE (1)			0.98752	1	2
2	1	3	0.96704			
2	2	2	0.96298			
	OPT. DESIGN AT STAGE (2)			0.96704	1	3
3	1	4	0.96459			
3	2	3	0.95521			
3	3	3	0.94450			
	OPT. DESIGN AT STAGE (3)			0.96459	1	4
4	1	4	0.94132			
4	2	4	0.93421			
4	3	3	0.92586			
4	4	2	0.94062			
	OPT. DESIGN AT STAGE (4)			0.94132	1	4
5	1	5	0.89734			
5	2	5	0.89191			
5	3	4	0.88842			
5	4	3	0.90568			
5	5	3	0.89644			
	OPT. DESIGN AT STAGE (5)			0.90568	4	3

OPTIMUM DESIGN IS  
(1, 3, 4) (4, 5, 3)

表 3. 逆動的 計劃法

TABLE (III) REVERSE DYNAMIC PROGRAMMING

I	J	M	FP (I)
5	5	3	0.95232
4	5	3	0.93893
3	5	4	0.91870
2	3	3	0.90821
1	3	4	0.90568

表 4. 制限條件下에서의 最適設計

TABLE (IV) OPTIMIZATION WITH CONSTRAINTS

STAGE	D (J)	D (I - 1)	I	J	M	R (DJ)	C1 (DJ)	C2 (DJ)
1	1	0	1	1	1	0.95000	10.00	5.00
	2	0	1	1	2	0.98752	21.00	11.00
2	3	1	2	2	1	0.85500	17.00	13.00
	4	2	2	2	1	0.88877	28.00	19.00
	5	1	2	2	2	0.92639	25.00	22.00
	6	0	1	2	2	0.96429	35.00	27.00
	7	0	1	1	2	3	0.96704	53.00
3	8	3	3	3	1	0.72675	23.00	20.00
	9	4	3	3	1	0.75546	34.00	26.00
	10	3	3	3	2	0.82740	30.00	28.00
	11	4	3	3	2	0.86009	41.00	34.00
	12	1	2	3	2	0.88856	37.00	36.00
	13	5	3	3	2	0.89649	38.00	37.00
	14	5	3	3	3	0.90480	45.00	45.00
	15	0	1	3	2	0.91608	47.00	41.00
	16	6	3	3	2	0.93317	48.00	42.00
	17	6	3	3	3	0.94182	55.00	50.00
	18	2	2	3	3	0.95521	62.00	58.00
	19	0	1	3	3	0.96001	71.00	62.00
20	0	1	1	3	4	0.96459	95.00	83.00

STAGE	D (J)	D(I - 1)	I	J	M	R (DJ)	C1(DJ)	C 2(DJ)
4	21	8	4	4	1	0.65407	31.00	26.00
	22	9	4	4	1	0.67991	42.00	32.00
	23	8	4	4	2	0.70869	40.00	33.00
	24	10	4	4	1	0.74466	38.00	34.00
	25	3	3	4	2	0.79567	46.00	40.00
	26	12	4	4	1	0.79970	45.00	42.00
	27	10	4	4	2	0.80684	47.00	41.00
	28	13	4	4	1	0.80684	46.00	43.00
	29	15	4	4	1	0.82447	55.00	47.00
	30	4	3	4	2	0.82709	57.00	46.00
	31	11	4	4	2	0.83871	58.00	47.00
	32	1	2	4	2	0.84495	53.00	48.00
	33	12	4	4	2	0.86648	54.00	49.00
	34	13	4	4	2	0.87422	55.00	50.00
	35	14	4	4	2	0.88232	62.00	58.00
	36	6	3	4	2	0.89737	64.00	54.00
	37	16	4	4	2	0.90998	65.00	55.00
	38	17	4	4	2	0.91841	72.00	63.00
	39	6	3	4	3	0.92322	79.00	68.00
	40	18	4	4	2	0.93148	79.00	71.00
	41	19	4	4	2	0.93615	88.00	75.00
5	42	21	5	5	1	0.52326	36.00	36.00
	43	22	5	5	1	0.54393	47.00	42.00
	44	23	5	5	1	0.56695	45.00	43.00
	45	24	5	5	1	0.59573	43.00	44.00
	46	21	5	5	2	0.61535	42.00	47.00
	47	21	5	5	3	0.62289	48.00	58.00
	48	25	5	5	1	0.63653	51.00	50.00
	49	26	5	5	1	0.63976	50.00	52.00
	50	27	5	5	1	0.64547	52.00	51.00
	51	8	4	5	2	0.65638	50.00	53.00
	52	23	5	5	2	0.66674	51.00	54.00
	53	24	5	5	2	0.70058	49.00	55.00
	54	24	5	5	3	0.70916	55.00	66.00
	55	3	3	5	2	0.71176	56.00	60.00
	56	25	5	5	2	0.74856	57.00	61.00
	57	26	5	5	2	0.75236	56.00	63.00
	58	27	5	5	2	0.75908	58.00	62.00
	59	28	5	5	2	0.75908	57.00	64.00
	60	26	5	5	3	0.76157	62.00	74.00
	61	28	5	5	3	0.76837	63.00	75.00
	62	29	5	5	2	0.77566	66.00	68.00
63	30	5	5	2	0.77813	68.00	67.00	
64	31	5	5	2	0.78906	69.00	68.00	
65	12	4	4	2	0.80252	64.00	69.00	

STAGE	D (J)	D (I-1)	I	J	M	R (DJ)	C1 (DJ)	C2(DJ)
	66	33	5	5	2	0.81518	65.00	70.00
	67	34	5	5	2	0.82246	66.00	71.00
	68	33	5	5	3	0.82517	71.00	81.00
	69	15	4	5	2	0.82738	74.00	74.00
	70	35	5	5	2	0.83008	73.00	79.00
	71	34	5	5	3	0.83253	72.00	82.00
	72	36	5	5	2	0.84425	75.00	75.00
	73	37	5	5	2	0.85611	76.00	76.00
	74	38	5	5	2	0.86404	83.00	84.00
	75	37	5	5	3	0.86659	82.00	87.00
	76	39	5	5	2	0.86857	90.00	89.00
	77	16	4	5	3	0.87618	89.00	92.00
	78	40	5	5	2	0.87633	90.00	92.00
	79	41	5	5	2	0.88073	99.00	96.00
	80	17	4	5	3	0.88429	96.00	100.00

OPTIMUM DESIGN WITH CONSTRAINTS IS

(1, 2, 2) (3, 3, 3) (4, 5, 3)

表 5. 制限条件下에서의 最適設計(計算量 減縮)

TABLE (IV) OPTIMIZATION WITH CONSTRAINTS

STAGE	D (J)	D (I - 1)	I	J	M	R (DJ)	C1(DJ)	C2 (DJ)
1	1	0	1	1	2	0.98752	21.00	11.00
2	2	0	1	2	2	0.96429	35.00	27.00
	3	0	1	2	3	0.96704	53.00	41.00
3	4	2	3	3	3	0.94182	55.00	50.00
	5	1	2	3	3	0.95521	62.00	58.00
	6	0	1	3	3	0.96001	71.00	62.00
4	7	5	4	4	2	0.93148	79.00	71.00
	8	6	4	4	2	0.93615	88.00	75.00
5	9	4	4	5	3	0.88429	96.00	100.00

OPTIMUM DESIGN WITH CONSTRAINTS IS

(1, 2, 2) (3, 3, 3) (4, 5, 3)

表 6 - 1. Jensen의 方法

NO. OF PLACE	=	28
NO. OF DELETE	=	19
MAX. LENGTH	=	10
NO. RF REPLACE	=	10

表 6 - 2. 本 研究의 方法

NO. OF PLACE	=	18
NO. OF DELETE	=	9
MAX. LENGTH	=	9
NO. RF REPLACE	=	4

#### 4. 結 論

本 研究에서는 最適 信賴度 設計를 為한 Jensen의 演算法中에서 높은 信賴度를 갖는 實現 可能한 設計를 빨리 發見함으로써 對案設計의 數를 減少시켰다. 事例 研究 中 信賴度 函數 및 費用函數는 利用者의 必要에 따라 修正될 수 있다.

#### 參 考 文 獻

- (1) Jensen, A. P., " Optimization of Series - Parallel - Series Networks, " Operations Research, 18, 471 - 82 (1970) .
- (2) Proschan F., and T. A. Bray, " Optimum Redundancy Under Multiple Constraints, " Operations Research, 13, 800 - 814 (1965) .