Yugoslav Journal of Operations Research
17 (2007), Number 2, 235-243
DOI: 10.2298/YUJOR0702235H

THE RELIABILITY OF SYSTEMS WITH STAIR-TYPE CONSECUTIVE MINIMAL CUTS

Y.-C. HSIEH

Department of Industrial Management, National Formosa University, Taiwan
yhsieh@nfu.edu.tw

T.-C. CHEN

Department of Information Management, National Formosa University, Taiwan
tchen@nfu.edu.tw

Received: April 2006 / Accepted: July 2007

Abstract: This paper considers the component system with stair-type consecutive minimal cuts. The system consists of *n* components and the set of minimal cuts can be linearly ordered. The proposed system generalizes the typical consecutive-*k*-out-of-*n*: F systems. By using integer linear programming, this paper shows that such a system can be converted into the consecutive-*k*-out-of-*n*: F systems with the insertion of artificial "broken-down" components. Then the system reliability can be obtained by the product form of component reliability matrices and the limit behavior of system could be easily analyzed. Additionally, we show that the integer constraints of the linear programming can be relaxed due to the total unimodularity. Thus, a general linear programming can be used to solve the problem. Numerical examples show the simple and effective new approach.

Keywords: Stair-type consecutive minimal cuts, linear programming, consecutive-*k*-out-of-*n*:F system.

1. INTRODUCTIONS

A system of components is coherent if (i) its structure function is increasing, and (ii) each component is relevant (Barlow and Proschan [1]). Therefore, a physical system is usually a coherent system. Both series systems and parallel systems are most well known engineering systems. It is known that the reliability of series system is not high, especially of a large series system, and the reliability of parallel system is high, but tends to be very expensive (Chao et al. [5]). Since 1980, a new system, namely consecutive-k-out-of-n: F system (or $C(k, n$: F) system), has caught much attention (Chao et al. [5]) because:

1. it usually has much higher reliability than the series systems, and
2. it is less expensive than the parallel systems.

The $C(k, n$: F) system consists of n linearly connected components and it fails if and only if at least consecutive k components are failed. There are numerous applications for such systems in practice, e.g., microwave transmission systems and oil pipe transportation systems etc. (Chao et al. [5]). In the early years, most of the proposed formulae were based upon the recursive equations and assumed that all the components are s-independent with the same probability. In 1984, Chao and Lin [6] first observed that the general $C(k, n$: F) system can be imbedded in a Markov chain with 2^k states. However, only systems with small k can be manipulated. In 1986, Fu [7] successfully reduced the Markov chain into k+1 states and considerably simplified the probability structure of $C(k, n$: F) system. Later, Fu and Hu [8], and Chao and Fu [3][4] developed the following simple and well known exact formula in the product form of matrices for the reliability of $C(k, n$: F) system.

$$R\left(k,n,p:\mathrm{F}\right)=\pi_0\left(\prod_{i=1}^{n}M_i\right)U^T \tag{1}$$

where

$$M_i=\begin{pmatrix} p_i & 1-p_i & 0 & \ldots & 0 & 0 \\ p_i & 0 & 1-p_i & \ldots & 0 & 0 \\ : & : & : & : & : & : \\ p_i & 0 & 0 & \ldots & 0 & 1-p_i \\ 0 & 0 & 0 & \ldots & 0 & 1 \end{pmatrix}\in R^{(k+1)\times(k+1)}$$

$\pi_0=(1,0,...,0)\in R^{1\times(k+1)}$, $U=(1,1,...,1,0)\in R^{1\times(k+1)}$, and p_i is the operational reliability of component i. With the use of equation (1), Hsieh [9] developed efficient lower bounds and upper bounds of reliability for general coherent systems.

In this paper, we define a special class of systems with consecutive minimal cuts, namely stair-type consecutive minimal cuts system (or ST-CMC system). For the ST-CMC system, the set of minimal cuts is linearly ordered so that the i-th minimal cut $C_i=\{n_{i,1}, n_{i,2},\ldots, n_{i,Ii}\}$, the $(i+1)$-th minimal cut $C_{i+1}=\{n_{i+1,1}, n_{i+1,2},\ldots, n_{i+1,Ii+1}\}$ with $n_{i,1}< n_{i+1,1}$, $n_{i,Ii} < n_{i+1,Ii+1}$ and $C_i\cap C_{i+1}\neq\phi\ \forall i$, where $n_{i,j}=n_{i,1}+j-1$. With the use of integer linear programming, we will convert the ST-CMC system into a typical $C(k, n$: F) system by inserting several artificial "broken-down" components in the appropriate locations of sequence of minimal cuts. Therefore, the reliability of ST-CMC system can be computed by the product form of component reliability matrices. Note that the limit behavior of systems could be easily analyzed if their system reliabilities can be represented by the product form of matrices. In addition, we will show that the integer constraints of the integer linear programming can be relaxed due to the total unimodularity of constraints. Thus, a general linear programming can be used to solve the problem. Numerical examples are provided to show the new linear programming approach.

This paper is organized as follows. Section 2 briefly introduces the ST-CMC systems and its relationship with $C(k, n$: F) systems. Section 3 demonstrates the formulation of integer linear programming for converting the ST-CMC systems into typical $C(k, n$: F) systems by inserting several artificial broken-down components. Additionally, we will show that the integer constraints of the proposed linear programming can be relaxed due to the total unimodularity of constraints. Numerical examples are provided to show the new linear programming approach in this section. Brief conclusions are summarized in Section 4.

2. THE ST-CMC SYSTEMS

Assume that a coherent system consists of c minimal cuts $C=\{C_1, C_2,\ldots, C_c\}$. We define the ST-CMC system below.

Definition 1. *A coherent system of components* $\{1,2,\ldots,n\}$ *is said to be a ST-CMC system if the set of minimal cuts can be linearly ordered so that the i-th minimal cut* $C_i=\{n_{i,1}, n_{i,2},\ldots, n_{i,Ii}\}$, *the* $(i+1)$*-th minimal cut* $C_{i+1}=\{n_{i+1,1}, n_{i+1,2},\ldots, n_{i+1,Ii+1}\}$ *with* $n_{i,1}< n_{i+1,1}$, $n_{i,Ii} < n_{i+1,Ii+1}$ *and* $C_i \cap C_{i+1} \neq \phi \ \forall \ i \in \{1,2,\ldots,c-1\}$, *where* $n_{i,j}=n_{i,1}+j-1$.

Property 1. *The C(k, n: F) system is a special case of ST-CMC systems.*

Proof: Let $C_i=\{i, i+1,\ldots, i+k-1\}$ $\forall i$, the ST-CMC system will lead to the $C(k, n$: F) system. ■

Property 2. The minimal cuts of ST-CMC systems are consecutive minimal cuts.

Proof: It is straightforward and omitted by the definition. ■

3. LINEAR PROGRAMMING AND EXAMPLES

3.1. Concept of the New Approach

Before introducing the new method, we consider the following example for the main idea of the new linear programming approach.

Example 1. Assume that there are ten various pumps between *A* and *B* for transporting oil. The carrying capacity for each pump is different, so that each pump can carry oil to various far stations. For example, Figure 1 shows that: Pump 1 can transport oil to station1 and station 2, Pump 2 can transport oil to stations 1, 2, 3, and Pump 10 can transport oil to stations 8, 9, 10 etc. If any pump fails, then it cannot carry any oil. Let p_i be the operational reliability of pump *i*. Figure 1 shows that when both pumps 1 and 2 are failed, there is no connection between *A* and *B*. Therefore, $C_1=\{1,2\}$ is a minimal cut for the system. In addition, $C_2=\{2,3,4\}$, $C_3=\{4,5\}$, $C_4=\{5,6,7\}$, $C_5=\{7,8\}$, $C_6=\{8,9,10\}$ are the other minimal cuts for the system. Figure 2 shows these six minimal cuts, and one may verify that these minimal cuts are connected in stair-type shape.

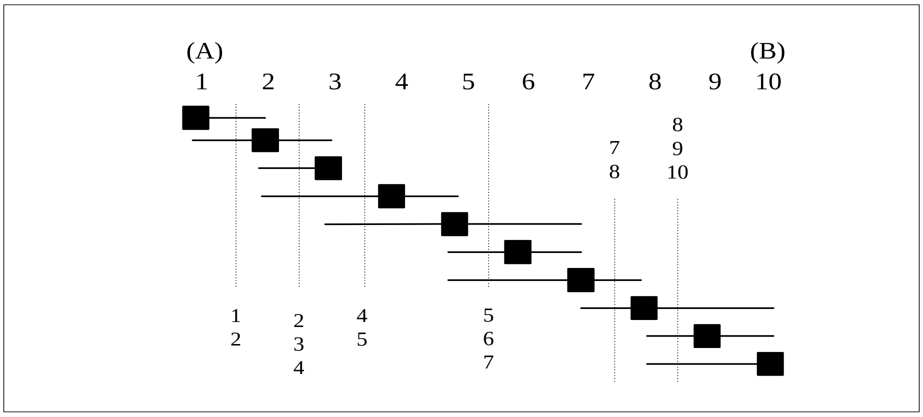

Figure 1: The pump system of Example 1

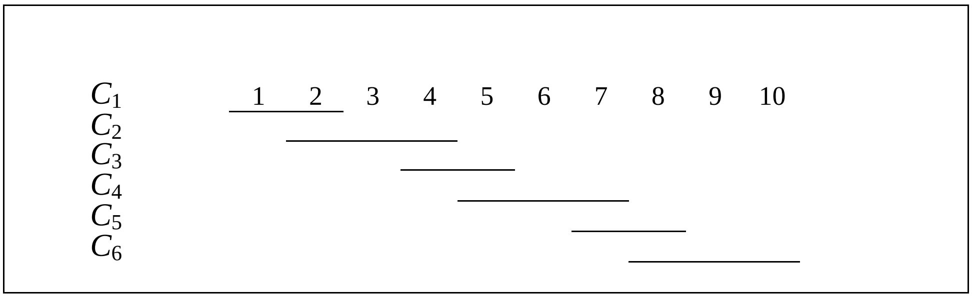

Figure 2: The six minimal cuts for Example 1

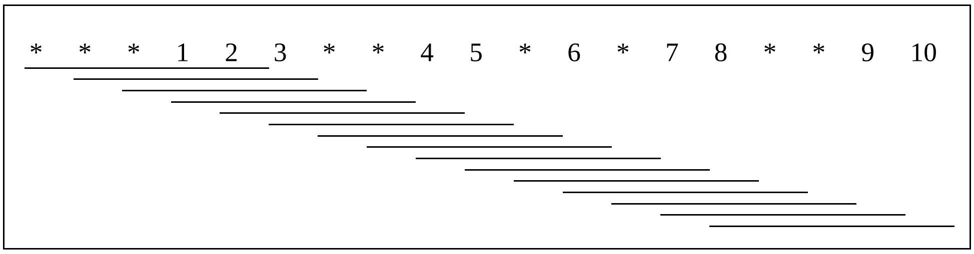

Figure 3: $C(5,19\text{: F})$ system. Alternative representation of minimal cuts with the insertions of 9 artificial "broken-down" components (*)

Suppose that we insert 9 artificial “broken-down” components into the system of Figure 2 as that of Figure 3. Note that (*) denotes an artificial “broken-down” component, and there are 15 cuts in Figure 3, namely:

$\tilde{C}_1$={*,*,*,1,2}	⇒ {1,2}	($\tilde{C}_1$ is a minimal cut)
$\tilde{C}_2$={*,*,1,2,3}	⇒ {1,2,3}	($\tilde{C}_1 \subseteq \tilde{C}_2$, $\tilde{C}_2$ is redundant)
$\tilde{C}_3$={*,1,2,3,*}	⇒ {1,2,3}	($\tilde{C}_3 = \tilde{C}_2$, $\tilde{C}_3$ is redundant)
$\tilde{C}_4$={1,2,3,*,*}	⇒ {1,2,3}	($\tilde{C}_4 = \tilde{C}_2$, $\tilde{C}_4$ is redundant)
$\tilde{C}_5$={2,3,*,*,4}	⇒ {2,3,4}	($\tilde{C}_5$ is a minimal cut)
$\tilde{C}_6$={3,*,*,4,5}	⇒ {3,4,5}	($\tilde{C}_7 \subseteq \tilde{C}_6$, $\tilde{C}_6$ is redundant)
$\tilde{C}_7$={*,*,4,5,*}	⇒ {4,5}	($\tilde{C}_7$ is a minimal cut)
$\tilde{C}_8$={*,4,5,*,6}	⇒ {4,5,6}	($\tilde{C}_7 \subseteq \tilde{C}_8$, $\tilde{C}_8$ is redundant)
$\tilde{C}_9$={4,5,*,6,*}	⇒ {4,5,6}	($\tilde{C}_9 = \tilde{C}_8$, $\tilde{C}_9$ is redundant)
$\tilde{C}_{10}$={5,*,6,*,7}	⇒ {5,6,7}	($\tilde{C}_{10}$ is a minimal cut)
$\tilde{C}_{11}$={*,6,*,7,8}	⇒ {6,7,8}	($\tilde{C}_{13} \subseteq \tilde{C}_{11}$, $\tilde{C}_{11}$ is redundant)
$\tilde{C}_{12}$={6,*,7,8,*}	⇒ {6,7,8}	($\tilde{C}_{13} \subseteq \tilde{C}_{12}$, $\tilde{C}_{12}$ is redundant)
$\tilde{C}_{13}$={*,7,8,*,*}	⇒ {7,8}	($\tilde{C}_{13}$ is a minimal cut)
$\tilde{C}_{14}$={7,8,*,*,9}	⇒ {7,8,9}	($\tilde{C}_{13} \subseteq \tilde{C}_{14}$, $\tilde{C}_{14}$ is redundant)
$\tilde{C}_{15}$={8,*,*,9,10}	⇒ {8,9,10}	($\tilde{C}_{15}$ is a minimal cut)

Thus, $\tilde{C}_1 = C_1$, $\tilde{C}_5 = C_2$, $\tilde{C}_7 = C_3$, $\tilde{C}_{10} = C_4$, $\tilde{C}_{13} = C_5$, $\tilde{C}_{15} = C_6$, which further imply that minimal cuts of Figure 2 and Figure 3 are the same. Therefore, the system of Figure 2 and the system of Figure 3 will have the same system reliability. Since Figure 3 is a C(5,19: F) system, equation (1) can be used for computing the system reliability. More specifically, the system reliability for Figure 3 (or Figure 2) is：

$$R(5,19,\tilde{p}:\mathrm{F}) = \pi_0 \left(\prod_{i=1}^{19} M_i \right) U^T \tag{2}$$

where

$$M_i = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{for } i = 1,2,3,7,8,11,13,16,17$$

and

$$M_i = \begin{pmatrix} \tilde{p}_i & 1-\tilde{p}_i & 0 & 0 & 0 & 0 \\ \tilde{p}_i & 0 & 1-\tilde{p}_i & 0 & 0 & 0 \\ \tilde{p}_i & 0 & 0 & 1-\tilde{p}_i & 0 & 0 \\ \tilde{p}_i & 0 & 0 & 0 & 1-\tilde{p}_i & 0 \\ \tilde{p}_i & 0 & 0 & 0 & 0 & 1-\tilde{p}_i \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \text{ for } i = 4,5,6,9,10,12,14,15,18,19$$

where $\pi_0 = (1,0,0,0,0,0) \in R^{1\times 6}$, $U = (1,1,1,1,1,0) \in R^{1\times 6}$, $\tilde{p}_4 = p_1$, $\tilde{p}_5 = p_2$, $\tilde{p}_6 = p_3$, $\tilde{p}_9 = p_4$, $\tilde{p}_{10} = p_5$, $\tilde{p}_{12} = p_6$, $\tilde{p}_{14} = p_7$, $\tilde{p}_{15} = p_8$, $\tilde{p}_{18} = p_9$, $\tilde{p}_{19} = p_{10}$, and p_i is the operational reliability of pump i.

3.2. Linear Programming Formulation

The example in Section 3.1 shows that the ST-CMC system can be converted into a typical $C(k, n$: F) system by inserting some artificial broken-down components. There are several issues for the insertion of artificial broken-down components, including the number of artificial broken-down components and their locations. The next integer programming formulation will provide a simple but effective approach for solving both issues simultaneously. Assume that a ST-CMC system has n components $\{1,2,\ldots,n\}$ and c minimal cuts $C=\{C_1, C_2,\ldots, C_c\}$ and the i-th minimal cut $C_i=\{n_{i,1}, n_{i,2},\ldots, n_{i,Ii}\}$, the $(i+1)$-th minimal cut $C_{i+1}=\{n_{i+1,1}, n_{i+1,2},\ldots, n_{i+1,Ii+1}\}$, and $U_{n_{i,2}}=\{n_{i,2}, n_{i,3},\ldots, n_{i+1,Ii+1} \mid n_{i+1,1}-n_{i,1}\geq 2, i=1, 2,\ldots, c-1\}$ where $n_{i,j}= n_{i,1}+j-1$. Let $|C_i|=I_i$ and $|U_j|=J_j$.

Integer Linear Programming

$$\text{Min } k \tag{3}$$

s.t.

$$I_i + \sum_{j=n_{i,1}}^{I_i+1} Y_j = k, \ \forall i \in \{1,2,\ldots,c\} \tag{4}$$

$$J_i + \sum_{j=n_{i,2}}^{n_{i,2}+J_i-1} Y_j \leq k, \ \forall i \in \{j \mid U_j \neq \phi\} \tag{5}$$

$Y_j \in$ non-negative integers and k is a positive integer. (6)

Remarks.

1. $Y_j \equiv$ the number of artificial broken-down components *before* component j.
   $Y_{n+1} \equiv$ the number of artificial broken-down components *after* the last component n.
   $k \equiv$ the length of consecutive components in $C(k, n$: F) system.
2. The objective in (3) is to minimize the length of consecutive components in $C(k, n$: F) system.

3. Constraint (4) is to assure that each minimal cut appears in the $C(k, n\text{: F})$ system.
4. Constraint (5) is to assure that each redundant cut (see Example 1) appears in the cuts of $C(k, n\text{: F})$ system.
5. Constrain (6) denotes the ranges for all variables.
6. There are n+2 integer variables, $Y_1,\ldots, Y_{n+1}$, k, and $c+|\{j \mid U_j \neq \phi\}|$ constraints.

Property 3. *Constraint (6) of integer linear programming in (3)-(6) can be relaxed to*

$$Y_j \geq 0 \text{ and } k > 0. \tag{7}$$

Proof: For the simplicity of proof, we (i) replace $U_{n_{i,2}}=\{n_{i,2},\ldots, n_{i+1,Ii+1} \mid n_{i+1,1}-n_{i,2}\geq 2, i=1, 2,\ldots, c-1\}$ with $U_{n_{i,2}}$, $U_{n_{i,3}}$,…, $U_{n_{i+1,1}-1}$ where $U_{n_{i,2}}=\{n_{i,2},\ldots, n_{i+1,Ii+1}\}$, $U_{n_{i,3}}=\{n_{i,3},\ldots, n_{i+1,Ii+1}\}$,…, $U_{n_{i+1,1}-1}=\{n_{i+1,1}-1,\ldots, n_{i+1,Ii+1}\}$ if $n_{i+1,1}-n_{i,1}\geq 2$ for i=1, 2,…, c-1, and (ii) add slack variables X_i to constraints (5) such that the equalities will hold. Because $U_{n_{i,3}}\subseteq U_{n_{i,2}}$,…, $U_{n_{i+1,1}-1}\subseteq U_{n_{i,2}}$, process (i) will increase some redundant constraints but it will not affect the optimal solution for the integer linear programming. Note that after the processes of (i) and (ii), the number of constraints is exactly $s=n-|C_c|+1$. Denote by (A_1), (A_2), …, (A_s) the s constraints. Now let (B_i)←(A_i)-(A_{i+1}), i.e., subtracting constraint (A_{i+1}) from (A_i) to be a new constraint (B_i) for i=1, 2, …, $n-|C_c|$ and let (B_s)←(A_s)-(A_1). Thus the new constraints are now (B_1), (B_2), …, (B_s). Note that the left-hand-side matrix of constraints (B_1), (B_2), …, (B_s), namely M, has elements either 0, -1, or 1 and matrix M is a node-arc incidence matrix. Since every node-arc incidence is totally unimodular and the values for the right-hand-side of constraints (B_1), (B_2), …, (B_s) are all integers, the linear programming will have only integer-valued basic solutions (Bazarra et al. [2]). It implies that the integer constraints can be relaxed, i.e., constraint (6) could be replaced with constraint (7). ■

Example 2. Consider Example 1 again for showing the proposed linear programming formulation. Recall that in Example 1 we have $C_1=\{1,2\}$, $C_2=\{2,3,4\}$, $C_3=\{4,5\}$, $C_4=\{5,6,7\}$, $C_5=\{7,8\}$, $C_6=\{8,9,10\}$. (i) $U_1=U_2=U_4=U_5=U_7=U_8=U_9=U_{10}=\phi$, $U_3=\{3,4,5\}$, $U_6=\{6,7,8\}$, and (ii) adding slack variables X_5 and X_8 to constraints (5), we have:

Min k

s.t.	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	Y_{10}	Y_{11}	k	X_5	X_8	RHS	
	1	1	1									-1			=-2	(A_1)
		1	1	1	1							-1			=-3	(A_2)
			1	1	1							-1	1		=-3	(A_3)
				1	1	1						-1			=-2	(A_4)
					1	1	1	1				-1			=-3	(A_5)
						1	1	1				-1		1	=-3	(A_6)
							1	1	1			-1			=-2	(A_7)
								1	1	1	1	-1			=-3	(A_8)

Y_1, Y_2, …, Y_{11}, X_5, X_8 are non-negative integers and k is a positive integer.

Now let $(B_i) \leftarrow (A_i)-(A_{i+1})$, i.e., subtracting constraint (A_{i+1}) from (A_i) to be a new constraint (B_i), i=1, 2, …, n-| C_c | (=10-3) and let $(B_8) \leftarrow (A_8)-(A_1)$, we have:

Min k

s.t.	Y_1	Y_2	Y_3	Y_4	Y_5	Y_6	Y_7	Y_8	Y_9	Y_{10}	Y_{11}	X_5	X_8	RHS	
	1			-1	-1									=1	(B_1)
		1										-1		=0	(B_2)
			1			-1						1		=-1	(B_3)
				1			-1	-1						=1	(B_4)
					1								-1	=0	(B_5)
						1			-1				1	=-1	(B_6)
							1			-1	-1			=1	(B_7)
	-1	-1	-1					1	1	1	1			=-1	(B_8)

$Y_1, Y_2, \ldots, Y_{11}, X_5, X_8$ are non-negative integer and k is positive integer.

Thus the left-hand-side matrix of constraints $(B_1), (B_2), \ldots, (B_8)$ has elements either 0, -1, or 1 and it is a node-arc incidence matrix (Bazarra et al. [2]). Since the values for the right-hand-side of constraints $(B_1), (B_2), \ldots, (B_8)$ are all integers, the linear programming will have only integer-valued basic solutions. It implies that the integer constraints could be relaxed by constraint (7). Therefore the integer linear programming can be rewritten as:

Min k

s.t. $2+Y_1+Y_2+Y_3=k$

$3+Y_2+Y_3+Y_4+Y_5=k$

$2+Y_4+Y_5+Y_6=k$

$3+Y_5+Y_6+Y_7+Y_8=k$

$2+Y_7+Y_8+Y_9=k$

$3+Y_8+Y_9+Y_{10}+Y_{11}=k$

$3+Y_3+Y_4+Y_5 \leq k$

$3+Y_6+Y_7+Y_8 \leq k$

$Y_1, Y_2, \ldots, Y_{11} \geq 0$ and $k > 0$

Several software systems can be used for solving this general linear programming, for example, LINDO, LINGO, GINO, and AMPL etc. The optimal solution for this linear programming is: $k^* = 5, Y_1^* = 3, Y_4^* = Y_9^* = 2, Y_6^* = Y_7^* = 1, Y_i = 0$ for the other i. It implies that we have to insert 3 artificial broken-down components before component 1; insert 2 artificial broken-down components before components 4 and 9; insert 1 artificial broken-down components before components 6 and 7, which is exactly the C(5, 19: F) system in Figure 3.

4. CONCLUSIONS

In this paper, we have presented a special class of consecutive minimal cut systems whose consecutive minimal cuts are in stair-type shape. The stair-type consecutive minimal cuts can be linearly ordered so that the *i*-th minimal cut $C_i=\{n_{i,1}, n_{i,2},\dots, n_{i,Ii}\}$, the $(i+1)$-th minimal cut $C_{i+1}=\{n_{i+1,1}, n_{i+1,2},\dots, n_{i+1,Ii+1}\}$ with $n_{i,1}< n_{i+1,1}$, $n_{i,Ii} < n_{i+1,Ii+1}$ and $C_i \cap C_{i+1} \neq \phi \ \forall i$, where $n_{i,j}=n_{i,1}+j-1$. The ST-CMC system generalizes the typical $C(k, n$: F) system. In this paper, we have shown that such a ST-CMC system can be converted into the well known $C(k, n$: F) system with the insertion of artificial "broken-down" components. Then the system reliability can be obtained by the product form of component reliability matrices and the limit behavior of system could be easily analyzed. A simple integer linear programming is developed for the optimal insertion of artificial broken-down components. Additionally, we have shown that the integer constraints of the integer linear programming can be relaxed due to the total unimodularity. Thus, a general linear programming can be used to solve the problem. Numerical examples show the simplicity and effectiveness of the proposed approach.

Acknowledgements: This research is partially supported by National Science Council, Taiwan, under grant No. NSC 92-2213-E-150-012 and NSC 92-2213-E-150-013.

REFERENCES

[1] Barlow, R., and Proschan, F., *Statistical Theory of Reliability and Life Testing Probability Models*, Silver Spring, NY, 1981.

[2] Bazarra, M.S., Jarvis, J.J., and Sherali, H.D., *Linear Programming and Network Flows*, John Wiley & Sons, NY, 1990.

[3] Chao, M.T., and Fu, J.C., "A limit of theorem of certain repairable system", *Annals of the Institute of Statistical Mathematics*, 41 (1989) 809-818.

[4] Chao, M.T., and Fu, J.C., "The reliability of large series systems under Markov structure", *Advances in Applied Probability*, 23 (1991) 894-908.

[5] Chao, M.T., Fu, J.C., and Koutras, M.V., "Survey of reliability studies of consecutive-k-out-of-n: F related systems", *IEEE Transactions on Reliability*, 44 (1995) 120-127.

[6] Chao, M.T., and Lin, G.D., "Economical design of large consecutive-k-out-of-n: F system", *IEEE Transactions on Reliability*, 33 (1984) 411-413.

[7] Fu, J.C., "Reliability of consecutive-k-out-of-n: F systems with (k-1) step Markov dependence", *IEEE Transactions on Reliability*, 35 (1986) 602-606.

[8] Fu, J.C., and Hu, B. "On reliability of large consecutive-k-out-of-n: F system with (k-1) step Markov dependence", *IEEE Transactions on Reliability*, 36 (1987) 75-77.

[9] Hsieh, Y.C., "New reliability bounds for coherent systems", *Journal of the Operational Research Society*, 54 (2003) 995-1001.