An Algorithm for Computing the Reliability of Weighted-k-out-of-n Systems

Jer-Shyan Wu
National Chiao Tung University, Hsinchu
Rong-Jaye Chen, Member IEEE
National Chiao Tung University, Hsinchu

Key Words — **Weighted-k-out-of-n system, System reliability, Algorithm.**

Reader Aids —
General purpose: Report a new algorithm
Special math needed for explanations: Probability theory
Special math needed to use results: Same
Result useful to: Reliability analysts & theoreticians

Summary & Conclusions — **This paper constructs a new k-out-of-n model, *viz*, a weighted-k-out-of-n system, and proposes an $O(n \cdot k)$ algorithm for computing its reliability.**

1. INTRODUCTION

The k-out-of-n systems were extensively studied in [1-7]. This paper proposes a new, more general model: a weighted-k-out-of-n:G(F) system which has n components, each with its own positive integer weight (total system weight $= w$), such that the system is good (failed) iff the total weight of good (failed) components is at least k. The reliability of the weighted-k-out-of-n:G system is the complement of the unreliability of a weighted-$(w-k+1)$-out-of-n:F system. Without loss of generality, we discuss the weighted-k-out-of-n:G system only. The k-out-of-n:G system is a special case of the weighted-k-out-of-n:G system wherein the weight of each component is 1.

An efficient algorithm is given to evaluate the reliability of the weighted-k-out-of-n:G system. The time complexity of this algorithm is $O(n \cdot \text{k})$.

2. MODEL

Assumptions

1. Each component and the system is either good or failed.
2. All n component states are mutually s-independent.
3. Each component has its own positive integer weight.
4. The system is good iff the total weight of good components is at least k.

Notation

n	number of components in a system
k	minimum total weight of all good components which makes the system good
w_i	weight of component i
p_i, q_i	Pr{component i is [good, failed]}; $p_i+q_i=1$
$R(i,j)$	reliability of the weighted-j-out-of-i:G system.

Other, standard notation is given in "Information for Readers & Authors" at the rear of each issue.

3. ALGORITHM

To derive $R(n,k)$, we need to construct the table with $R(i,j)$, for $i=0,1,\ldots,n$, and $j=0,1,2,\ldots,k$. Initially,

$$R(i,0) = 1, \text{ for } i=0,1,\ldots,n; \tag{1}$$

$$R(0,j) = 0, \text{ for } j=1,2,\ldots,k. \tag{2}$$

Considering $R(i,j)$ for $i=0,1,\ldots,n$, and $j<0$, it is obvious that $R(i,j)=1$. So, we propose a recursive formula to generate each $R(i,j)$, for $i=1,\ldots,n$, and $j=1,2,\ldots,k$:

$$R(i,j) = \begin{cases} p_i \cdot R(i-1,j-w_i) + q_i \cdot R(i-1,j), \text{ if } j-w_i \geq 0; \\ p_i + q_i \cdot R(i-1,\text{j}), \text{ otherwise.} \end{cases} \tag{3}$$

Now, the final algorithm for $R(n,k)$ is:

1. By (1) & (2), construct column #1 and row #1 in the table.
2. By (3), construct row #2, row #3, ... , row #$(n+1)$ in that order; the $R(n,k)$ is eventually derived. ◄

Because the size of the table is $(n+1)\cdot(k+1)$, the time complexity and space complexity are $O(n \cdot k)$.

4. EXAMPLE

Consider a weighted-5-out-of-3:G system with weights: 2, 6, 4.

By (1), get column #1 wherein,

$$R(0,0) = R(1,0) = R(2,0) = R(3,0) = 1;$$

and by (2), get row #1 wherein,

$$R(0,1) = R(0,2) = R(0,3) = R(0,4) = R(0,5) = 0.$$

Therefore, by (3), derive rows #2, #3, #4 in table 1 as follows:

TABLE 1
Weighted-5-out-of-3:G system
[Columns 1&2 are the same; columns 3&4 are the same]

i\j	0	1 & 2	3 & 4	5
0	1	0	0	0
1	1	p_1	0	0
2	1	$p_2+q_2p_1$	p_2	p_2
3	1	ζ_A	$p_3+q_3p_2$	ζ_B

$\zeta_A \equiv p_3+q_3p_2+q_3q_2p_1$; $\zeta_B \equiv p_3p_2+p_3q_2p_1+q_3p_2$

Row #2

$$R(1,1) = p_1 \cdot R(0,-1) + q_1 \cdot R(0,1) = p_1$$

$$R(1,2) = p_1 \cdot R(0,0) + q_1 \cdot R(0,2) = p_1$$

$$R(1,3) = p_1 \cdot R(0,1) + q_1 \cdot R(0,3) = 0$$

$$R(1,4) = p_1 \cdot R(0,2) + q_1 \cdot R(0,4) = 0$$

$$R(1,5) = p_1 \cdot R(0,3) + q_1 \cdot R(0,5) = 0$$

Row #3

$$R(2,1) = p_2 \cdot R(1,-5) + q_2 \cdot R(1,1) = p_2 + q_2\, p_1$$

$$R(2,2) = p_2 \cdot R(1,-4) + q_2 \cdot R(1,2) = p_2 + q_2\, p_1$$

$$R(2,3) = p_2 \cdot R(1,-3) + q_2 \cdot R(1,3) = p_2$$

$$R(2,4) = p_2 \cdot R(1,-2) + q_2 \cdot R(1,4) = p_2$$

$$R(2,5) = p_2 \cdot R(1,-1) + q_2 \cdot R(1,5) = p_2$$

Row #4

$$R(3,1) = p_3 \cdot R(2,-3) + q_3 \cdot R(2,1) = p_3 + q_3p_2 + q_3q_2p_1$$

$$R(3,2) = p_3 \cdot R(2,-2) + q_3 \cdot R(2,2) = p_3 + q_3p_2 + q_3q_2p_1$$

$$R(3,3) = p_3 \cdot R(2,-1) + q_3 \cdot R(2,3) = p_3 + q_3\, p_2$$

$$R(3,4) = p_3 \cdot R(2,0) + q_3 \cdot R(2,4) = p_3 + q_3\, p_2$$

$$R(3,5) = p_3 \cdot R(2,1) + q_3 \cdot R(2,5)$$

$$= p_3p_2 + p_3q_2p_1 + q_3p_2$$

$R(3,5)$ is the final result.

REFERENCES

[1] R.E. Barlow, K.D. Heidtmann, "Computing *k*-out-of-*n* system reliability", *IEEE Trans. Reliability*, vol R-33, 1984 Oct, pp 322-323.

[2] S.P. Jain, K. Gopal, "Recursive algorithm for reliability evaluation of *k*-out-of-*n*:G system", *IEEE Trans. Reliability*, vol R-34, 1985 Jun, pp 144-146.

[3] P.W. McGrady, "The availability of a *k*-out-of-*n*:G network", *IEEE Trans. Reliability*, vol R-34, 1985 Dec, pp 451-452.

[4] H. Pham, S.J. Upadhyaya, "The efficiency of computing the reliability of *k*-out-of-*n* systems", *IEEE Trans. Reliability*, vol 37, 1988 Dec, pp 521-523.

[5] S. Rai, A.K. Sarje, E.V. Prasad, A. Kumar, "Two recursive algorithm for computing the reliability of *k*-out-of-*n* systems", *IEEE Trans. Reliability*, vol R-36, 1987 Jun, pp 261-265.

[6] T. Risse, "On the evaluation of the reliability of *k*-out-of-*n* systems", *IEEE Trans. Reliability*, vol R-36, 1987 Oct, pp 433-435.

[7] A.K. Sarje, E.V. Prasad, "An efficient non-recursive algorithm for computing the reliability of *k*-out-of-*n* systems", *IEEE Trans. Reliability*, vol 38, 1989 Jun, pp 234-235.

AUTHORS

Jer-Shyan Wu; Dept. of Computer Science and Information Engineering; National Chiao Tung University; Hsinchu 30050 TAIWAN - R.O. CHINA. Internet (e-mail): jswu@algo1.csie.nctu.edu.tw

Jer-Shyan Wu was born in Taipei, Taiwan in 1967. He received his BS (1989) in Computer Science from National Taiwan University, and MS (1991) in Computer Science from National Chiao Tung University. His research interests include reliability analysis, queueing theory, parallel computing, interconnected network, tower of Hanoi problem and graph algorithms.

Dr. Rong-Jaye Chen, Professor; Dept. of Computer Science and Information Engineering; National Chiao Tung University; Hsinchu 30050 TAIWAN - R.O. CHINA.

Rong-Jaye Chen (M'90) was born in Taiwan in 1952. He received his BS (1977) in Mathematics from National Tsing-Hua University, Taiwan and PhD (1987) in Computer Science from University of Wisconsin-Madison. Dr. Chen is Associate Professor in the Department of Computer Science and Information Engineering in National Chiao Tung University. He is a member of IEEE. His research interests include reliability theory, algorithms, mathematical programming, computer networking.

Manuscript TR92-172 received 1992 September 17; revised 1993 April 5.

IEEE Log Number 92-10951 ◄TR►