Computing Consecutive-Type Reliabilities Non-Recursively

Galit Shmueli

***Abstract*—The reliability of consecutive-type systems has been approached from various angles. A new method is presented for deriving exact expressions for the generating functions and the reliabilities of various consecutive-type systems. This method, based on Feller's run theory, is easy to implement, and leads to both recursive and nonrecursive formulas for the reliability. The nonrecursive expression is especially advantageous for systems with numerous components. In comparison to the n (number of components) computations that the recursive formulas require, the nonrecursive formula only requires the computation of the roots of a polynomial of order k. The method is extended for computing generating functions and reliabilities of systems with multi-state components as well as systems with s-dependent components.**

***Index Terms*—Consecutive system, nonrecursive reliability, partial fraction expansion, recurrence relation, system reliability.**

ACRONYMS[1]

iff	if and only if
s-	statistical(ly)

NOTATION

S	operative component
F	failed component
p	probability of an F
q	$1-p$
R_n	reliability of a system with n components
X_j	state of component j
$u_{t,\,i}$	$\Pr\{$pattern i is completed on component $\#t\}$
$U_i(s)$	generating function of $u_{t,\,i}$
$\mathcal{G}(s)$	reliability generating function
WT	waiting time
$\mathcal{G}_{WT}(s)$	WT generating function
l	number of possible states of a component
s_i	root $\#i$ of the polynomial in the generating function's denominator
λ	$\Pr\{$failed component|the previous component failed$\}$

Manuscript received September 7, 2001; revised February 2, and July 1, 2002. Responsible Editor: Sheng-Tsaing Tseng.

The author is with the Decision and Information Technologies Department, Robert H. Smith School of Business, University of Maryland, College Park, MD 20742 USA (e-mail: GShmuli@rhsmith.umd.edu).

Digital Object Identifier 10.1109/TR.2003.817846

[1]The singular and plural of an acronym are always spelled the same.

I. INTRODUCTION

CONSECUTIVE-k-out-of-n and similar systems usually have a higher reliability than series systems, and are less expensive than parallel systems [2].

Within the consecutive-type family, consider the reliability of a system with n linearly-arranged components, which are labeled $1, 2, \ldots, n$. Usually, the breakdown of the system is reached by progressive transitions through several levels of deterioration [9]; i.e., components break down successively, until a critical number, k, of failed components causes the system to fail. Three consecutive-type systems are considered:

1) Consecutive-k-out-of-n : F—the system fails when k consecutive components fail.
2) m-consecutive-k-out-of-n : F—the system fails when m nonoverlapping groups of k consecutive components fail.
3) r-within-consecutive-k-out-of-n : F—the system fails when r ($r < k$) components fail within a window of k consecutive components.

These definitions describe the relation between the component failures and the system failure. Methods for evaluating the reliability, its generating function, and long-run behavior in such systems, lead to either approximations or exact values. Two major approaches are: the combinatorial approach [1], [4], and the Markov chain embedding approach [3], [6], [9]. Reference [2] gives a comprehensive review. In all cases, the reliability expression is recursive in n.

This paper introduces a new method for computing the exact reliability (and its generating function) for systems with on/off or multiple-outcome components. This method leads to both a recursive and a nonrecursive expression for the reliability. This paper illustrates how to obtain a recursive expression for the reliability, but focuses on the method that leads to a nonrecursive expression, which is especially desirable for systems with a many components.

Section II describes the method, and uses it to derive expressions for the reliability of the 3 consecutive-type systems defined in this section.

Section III applies this method to systems that have components that do not exhibit a simple on/off behavior, but have multiple outcomes. Thus, each component can be in one of l states ($l \geq 2$). Consider the 2 cases where either the states are ordered (e.g., gradual degradation), or are mutually exclusive (e.g., different types of failure). When $l = 2$, the results reduce to the ordinary on/off component structure.

Some properties of consecutive-type systems with multi-state components exist in the literature: mean values and bounds for consecutive-k-out-of-n systems with ordered multi-state components [8]. In [7] the Markov-chain embedding method is used for computing probabilities that are associated with multi-state trials, which are related to consecutive-k-out-of-n : F and to m-consecutive-k-out-of-n : F systems.

Section IV addresses systems with s-dependent components.

Section V presents some computational aspects of the method in Section IV.

II. A New Method for Obtaining The Reliability of Consecutive-Type Systems

This method, based on Feller's theory for runs [5], has 5 steps, which lead to the reliability probability and generating functions:

Step 1. Specify all the component-patterns that cause the system to fail, e.g., in a simple series system, the pattern that causes the entire system to fail is failure of 1 component.

Step 2. Write a recursive relation for the probability that each pattern is completed on component t. To write this type of relation, place the failure-causing component-pattern so that it ends on component t. For example, let the failure-causing pattern be: $X_1 X_2 \cdots X_k$, then place it on components $t-k+1$ to t. The probability of this event is $\Pr\{X_1\} \cdot \Pr\{X_2\} \cdots \Pr\{X_k\}$. Given this placement, check which component could possibly complete a failure pattern; obviously, component t does. The other possibilities are:

- Sub-pattern $X_1 \cdots X_{k-1}$ completes a failure-causing pattern on component $t-1$, followed by X_k;
- Sub-pattern $X_1 \cdots X_{k-2}$ completes a failure-causing pattern on component $t-2$, followed by $X_{k-1} X_k$;

  $\ddots$
- Sub-pattern X_1 completes a failure causing pattern on component $t-k+1$, followed by $X_2 \cdots X_k$.

Thus the probabilities of all these exclusive events are added to obtain:

$$\begin{aligned}\Pr\{X_1, X_2, \ldots, X_k\} &= u_t + u_{t-1} \cdot \Pr\{X_k\} + u_{t-2} \cdot \Pr\{X_{k-1}, X_k\} \\ &\quad + \ldots + u_{t-k+1} \cdot \Pr\{X_2, \cdots, X_k\}. \end{aligned} \tag{1}$$

This type of equation can be written explicitly in the recursive form:

$$\begin{aligned}u_t = {} & \Pr\{X_1, X_2, \ldots, X_k) - u_{t-1} \cdot \Pr\{X_k\} - u_{t-2} \\ & \cdot \Pr\{X_{k-1}, X_k\} - \ldots - u_{t-k+1} \cdot \Pr\{X_2 \ldots X_k\}, \end{aligned} \tag{2}$$

and can be interpreted as:

The probability that the pattern $X_1, \ldots, X_k$ is completed on trial t equals the probability:

that it begins on component $t-k+1$ and ends on component t, minus the probabilities that the sub-pattern $X_1, \ldots, X_{k-1}$ completes a failure-causing pattern followed by X_k,

that the sub-pattern $X_1, \ldots, X_{k-2}$ completes a failure-causing pattern followed by $X_{k-1} X_k$,

etc.

Step 3. Multiply each equation by s^t and sum over t to ∞. $U_i(s) \equiv \sum_{t=0}^{\infty} s^t \cdot u_t$ (the generating function of u_t), for each pattern i ($i = 1, \ldots, a$).

Step 4. Solve the set of a linear equations to obtain $U_i(s)$; then combine the solutions to get $\mathcal{G}(s)$, by using (3)

$$\mathcal{G}(s) = \left[(1-s) \cdot \left(\sum_{i=1}^{a} U_i(s) - a + 1 \right) \right]^{-1}; \tag{3}$$

$a \equiv$ number of distinct failure-causing patterns.

In some cases the reliability is the probability that a specific pattern does not occur more than m times within n components.The generating function in such cases is closely related to that of the waiting time for occurrence $\#m$ of the pattern in an infinite series of Bernoulli trials. In particular, the reliability equals the probability that occurrence $\#m$ of the pattern is after n trials.

Let WT now be the waiting time for occurrence $\#m$ of the pattern, and let its generating function be $\mathcal{G}_{WT}$. The relation between the generating functions of the reliability and the waiting time is [5, p. 256]:

$$\mathcal{G}(s) = \frac{1 - \mathcal{G}_{WT}}{1-s}. \tag{4}$$

This relation is used for systems such as the m-consecutive-k-out-of-n system.

The procedure in steps 1–4 leads to generating functions of the special form: rational functions. In many cases this ratio of polynomials is very complicated, and the ordinary way of obtaining the reliability by differentiation is laborious. One alternative is to use a combinatorial method [11] that leads to a recursive formula for the reliability. However, a recursive expression means that computing the reliability of a system with n components involves computing all the reliabilities of similar systems with $1, 2, \ldots, n-1$ components. A different technique is suggested here, that is suitable for rational functions and leads to a nonrecursive expression for the system reliability:

Step 5. Expand the rational generating function into partial fractions [5].

This method is especially advantageous for systems with many components, because it involves fewer computations.

The partial fraction expansion method is: The generating function is expressed as a ratio of polynomials $\mathcal{G}(s) = N(s)/D(s)$, that do not have common roots. Next, the ratio is expanded into partial fractions:

$$\mathcal{G}(s) = \sum_i \sum_{j=1}^{m_i} \frac{\rho_{i,j}}{(s_i - s)^j}, \tag{5}$$

s_i are the distinct roots of $D(s)$, each of multiplicity m_i, $\rho_{i,j}$ are functions of the roots.

Using an adequate infinite geometric series, then

$$\mathcal{G}(s) = \sum_{n=0}^{\infty} R_n \cdot s^n. \tag{6}$$

R_n is the required reliability, and is a function of s_i and $\rho_{i,j}$ (for more details, see [10]). The partial fraction expansion method, proposed by Feller for rational generating function, did not lead to exact probabilities because of computational difficulty.

However, now many standard software packages (e.g., Matlab, Maple, and Mathematica) have a partial fraction expansion procedure, which yields very accurate numerical reliabilities for the above roots and the constants. These advances are used to make Feller's theoretical method practical.

The following subsections apply this method to several consecutive-type systems and obtain an expression for their reliabilities and generating functions.

A. Consecutive-k-out-of-$n : \mathrm{F}$ Systems

In consecutive-k-out-of-$n : \mathrm{F}$ systems, the component-pattern that causes the system to fail is a sequence of k failures:

$$\underbrace{FF\ldots F}_{k \text{ times}}.$$

The probability that this pattern is completed on component t is: u_t ($t = k, \ldots, n$). A recurrence relation for u_t is [5]

$$u_t = p^k - p \cdot u_{t-1} - p^2 \cdot u_{t-2} - \ldots - p^{k-1} \cdot u_{t-k+1}; \quad p \equiv \Pr\{F\},\ u_0 = 1; \tag{7}$$

(7) can be interpreted as: the probability that the pattern of k consecutive failures is completed on component $\#t$ (it was not completed in the previous $k-1$ components) equals p^k minus the probabilities that the same pattern was completed on component $t-1$ followed by a failed component, or completed on component $t-2$ followed by 2 failed components, etc.

To obtain $U(s)$, the generating function of u_t, multiply by s^t and sum from 0 to ∞. The reliability generating function is then:

$$\mathcal{G}(s) = [(1-s) \cdot U(s)]^{-1} = \frac{1-(p \cdot s)^k}{1 - s + q \cdot p^k \cdot s^{k+1}}; \tag{8}$$

(8) can be manipulated slightly to reach the expression for the generating function in [9].

A recursive formula for the reliability (as a function of n) is then obtained directly, because the generating function is rational [11]. In this case:

$$R_n = \begin{cases} 1, & 0 \le n < k \\ 1 - p^k, & n = k \\ R_{n-1} - q \cdot p^k \cdot R_{n-k-1}, & n > k. \end{cases} \tag{9}$$

To obtain a nonrecursive formula, which is especially advantageous for systems with many components, use partial fraction expansion. Expanding (8) leads to:

$$R_n = \sum_{i=1}^{k+1} \frac{\rho_i}{s_i^{n+1}}, \tag{10}$$

$s_1, \ldots, s_k$ are the k distinct roots of the polynomial $1 - s + q \cdot p^k \cdot s^{k+1}$ with the additional root $s_{k+1} = 1/p$; and

$$\rho_i = \frac{(1-(ps_i)^k)/(1-ps_i)}{\prod\limits_{i \ne i'} (s_i - s_{i'})}; \tag{11}$$

the root $s_{k+1} = 1/p$ is common to the numerator & denominator in (8), and thus has a zero coefficient and does not contribute to R_n. In practice, R_n can be computed directly by using a partial fraction procedure that exists in many standard software packages.

B. m-Consecutive-k-out-of-$n : \mathrm{F}$ Systems

The reliability is derived through the relation to the waiting time for the pattern $\#m$ of k consecutive failed components in an infinite series of Bernoulli trials; for more details, see the Appendix. The generating function is:

$$\mathcal{G}(s) = \frac{(1 - s + q \cdot p^k \cdot s^{k+1})^m - (p \cdot s)^{m \cdot k} \cdot (1 - p \cdot s)^m}{(1-s) \cdot (1 - s + q \cdot p^k \cdot s^{k+1})^m}. \tag{12}$$

Because it is still a rational function, the reliability can be obtained using 1 of the 2 methods in this Section II-B. In this case it is easier to obtain a nonrecursive formula, because the roots of the denominator are 1 and the same roots as the consecutive-k-out-of-$n : \mathrm{F}$ case (except now, they are of multiplicity m); and the generating function is expanded into an expression of the form in (5), and the reliability is then:

$$R_n = \sum_{j=1}^{m} (-1)^j \cdot \binom{n+j-1}{n} \cdot \sum_{i=1}^{k} \frac{\rho_{i,j}}{s_i^{n+j}}; \tag{13}$$

where $s_1, \ldots, s_k$ are the distinct roots described in Section II-A (disregarding the two roots $s = 1/p$ and $s = 1$ that are common to the numerator and denominator and thus have zero coefficients), and $\rho_{i,j}$ are functions of the roots. As in the consecutive-k-out-of-n case, a practical method for computing (13) is by using a built-in partial fraction expansion procedure in a standard software package.

C. r-Within-Consecutive-k-out-of-$n : \mathrm{F}$ Systems

The component-types that cause the system to fail include all the possibilities of r failed components within a *window* of k consecutive components. For simplicity, the method is illustrated with $r = 2$ and $k > 2$ (2 failed components within a window of k consecutive components). There are $k-1$ patterns that cause the system to fail: $FF, FSF, FSSF, \ldots, FS, \ldots, SF$. Recurrence relations for the probabilities are:

$$\begin{aligned} u_{t,1} &= p^2 - p \cdot \sum_{i=1}^{k-1} u_{t-1,i} \quad \text{for } n \ge 2; \\ u_{t,2} &= p^2 \cdot q - p \cdot q \cdot \sum_{i=1}^{k-1} u_{t-2,i} \quad \text{for } n \ge 3; \\ &\ddots \\ u_{t,k-1} &= p^2 \cdot q^{k-2} - p \cdot q^{k-1} \cdot \sum_{i=1}^{k-1} u_{t-2,i} \quad \text{for } n \ge k. \end{aligned} \tag{14}$$

The first of these equations (14) gives the probability that the pattern FF is completed on component $\#t$. This equals the probability of that the $t-1$ and t components are failed, minus the probability that the failed component $\#(t-1)$ completes any of the $k-1$ failure-causing patterns, followed by the failed component t. The other equations can be interpreted similarly.

To find the generating functions of $u_{t,i}$, defined as:

$$U_i(s) = \sum_{t=0}^{\infty} u_{t,i} \cdot s^t; \tag{15}$$

multiply the equations in (14) by s^t, and then sum to infinity. Summing the $k-1$ equations and equating the left and right sides leads to:

$$\sum_{i=1}^{k-1} U_i(s) - (k-1) = \frac{p^2 \cdot \sum_{j=1}^{k-2} (q \cdot s)^j}{1-s} - p \cdot s \cdot \sum_{j=1}^{k-2} (q \cdot s)^j \cdot \sum_{i=1}^{k-1} U_i(s). \tag{16}$$

Rearranging (16) and using (3) gives the reliability generating function:

$$\mathcal{G}(s) = \frac{1 + p \cdot s \cdot \sum_{j=0}^{k-2} (q \cdot s)^j}{1 - q \cdot s - p \cdot q^{k-1} \cdot s^k} = \frac{1 - s \cdot (q-p) - p \cdot q^{k-1} \cdot s^k}{1 - 2q \cdot s + q^2 \cdot s^2 - p \cdot q^{k-1} \cdot s^k + p \cdot q^k \cdot s^{k+1}}. \tag{17}$$

A nonrecursive expression can be computed using a partial-fraction expansion software procedure.

This general method can be applied to any value of r. For the popular case, $k = r+1$, it involves the solution of a linear equation system of rank $(k-1)$. For small to moderate values of k, this can be done symbolically, using a symbolic software package (e.g., Maple or Mathematica). For $k > r+1$, and for large values of r and k, the numerical value of p should be put into the equations, and the system solved numerically.

D. Deriving the Reliability for General Failure Patterns

The method described in the beginning of this section can be applied to any type of failure pattern/s, which cause a system to fail. For example, for a system that fails if the pattern $FSFS$ occurs, the probability that the pattern is completed on component t (u_t) equals the probability that components $t-3$ and $t-1$ are failed and components $t-2$ and t are operative ($p^2 \cdot q^2$) minus the probability that the sub-pattern FS on components $t-3$, $t-2$ completes an $FSFS$ pattern, followed by FS on components $t-1$ and t. This can all be written as:

$$u_t = p^2 \cdot q^2 - p \cdot q \cdot u_{t-2}. \tag{18}$$

Using the generating function of u_t, the reliability generating function can be obtained from (3), and an expression for the reliability thus derived.

III. Reliability of Systems With Multi-State Components

The method in this paper can be used to obtain the reliability of systems with multi-state components. The only difference from the binary case (each component is either on or off) is in specifying the component-patterns that lead to the system failure (step 1).

This section deals with 2 types of relations between the possible component states:

1) Exclusive states, where each component can be in 1 state only.
2) Inclusive, gradual states; each state is included in the next state.

For simplicity, the application of this method is illustrated with a system with 3-state components.

- S the component is on
- F_{I} the component fails by type I failure
- F_{II} the component fails by type II failure.

A. Exclusive Types of Failures

A simple example is a system with 3-state components, where the temperature of the components can vary. Within a certain temperature range, the component is operational (state 1), while above or below that range, the component fails. In this case a component can either fail because it is below the required temperature (state 2), or because it is above the required temperature (state 3). The states are therefore mutually exclusive. Consider a consecutive-k-out-of-n system with mutually exclusive 3-state components. Here the system fails when there exist k_1 consecutive failed components of type I, or k_2 failed components of type II. Each component can either be operational (S), failed by type I (F_{I}) *or* failed by type II (F_{II}). The probabilities of type I and II failures are p_1 and p_2, respectively. The probability of an operational component is then $1 - p_1 - p_2$.

For this system, the 2 distinct component-patterns that cause system failure are

$$\underbrace{F_{\mathrm{I}} F_{\mathrm{I}} \ldots F_{\mathrm{I}}}_{k_1 \text{ times}} \quad \text{and} \quad \underbrace{F_{\mathrm{II}} F_{\mathrm{II}} \ldots F_{\mathrm{II}}}_{k_2 \text{ times}}.$$

Following steps 2–4, the reliability generating function is:

$$\mathcal{G}(s) = \frac{\left[1 - (p_1 \cdot s)^{k_1}\right] \cdot \left[1 - (p_2 \cdot s)^{k_2}\right]}{\Psi},$$
$$\Psi \equiv 1 - s + q_1 \cdot p_1^{k_1} \cdot s^{k_1+1} + q_2 \cdot p_2^{k_2} \cdot s^{k_2+1} - p_1^{k_1} \cdot p_2^{k_2} \cdot s^{k_1+k_2} - p_1^{k_1} \cdot p_2^{k_2} \cdot (1 - p_1 - p_2) \cdot s^{k_1+k_2+1}. \tag{19}$$

A nonrecursive calculation of the system reliability, based on expanding (19) into partial fractions can be obtained via a simple computer program.

This method is general and can be applied to various consecutive-type systems, with components that can have any fixed number of mutually exclusive states.

B. Gradual Degradation

A simple example for this type of system is: each component can be perfectly operative (S), have a minor failure (F_{I}), or a major failure (F_{II}). Each of these 3 states is considered to be inclusive of the previous state, i.e., the degree of the failure is gradual, from nonexistent to totally failed with a medium in-between state.

Let the system fail, following a run of k_2 type II (major) failed components, or a run of k_1 failed components of either type I or II (major/minor). In many cases it makes sense to assume that

$k_1 > k_2$. For example, a system with $k_1 = 3$ and $k_2 = 2$ fails iff it consists of 2 successive major-failed components ($F_{\rm II}F_{\rm II}$), or 3 consecutive failed components of any type:

$\cdot F_{\rm I}F_{\rm I}F_{\rm I}$, $F_{\rm I}$, $F_{\rm I}$, $F_{\rm II}$, $F_{\rm I}$, $F_{\rm II}$, $F_{\rm I}$, or $F_{\rm II}F_{\rm I}F_{\rm I}$.

Combinations such as $F_{\rm II}$, $F_{\rm II}$, $F_{\rm I}$ are excluded from the second set. Because these 5 patterns are mutually exclusive, step 1 is completed. Following steps 2, 3, 4 lead to the generating function and reliability of this system.

IV. Dependent Failures of Components

The 5-step method in Section II can be adapted for cases where failures of components are s-dependent on the state of preceding components. Except for step 2 (creating the recurrence relations), the rest of the steps remain unchanged. To illustrate how the dependence is incorporated into the recurrence relations, consider a 1-step Markovian dependence: the failure of component t depends on the state of component $t-1$.

Let X_t be the state of component $\#t$.

Specify the conditional probability of a failure by

$$\Pr\{X_t = F \mid X_{t-1} = F\} = \lambda. \tag{20}$$

Specify the unconditional probability of a failure by

$$\Pr\{X_t = F) = p.$$

For a consecutive-k-out-of-n : F system, it was shown that the failure-causing pattern is a sequence of k successive failures. In this case, the recurrence relation for u_t is:

$$u_t = p\cdot\lambda^{k-1} - \lambda\cdot u_{t-1} - \lambda^2\cdot u_{t-2} - \ldots - \lambda^{k-1}\cdot u_{t-k+1}. \tag{21}$$

Following steps 3 and 4 leads to the generating function:

$$\mathcal{G}(s) = \frac{1-(\lambda\cdot s)^k}{1-s+(p-\lambda)\cdot\lambda^{k-1}\cdot s^k + q\cdot\lambda^k\cdot s^{k+1}} \tag{22}$$

which can then be expanded to obtain the probability function. When $\lambda = p$ this reduces to the s-independent case.

The same method can be used for dependence of higher order, by defining the relevant conditional probabilities and incorporating them into the recurrence relations.

V. Computational Aspects

The 5-step method can lead to both recursive and nonrecursive formulas for the reliability. In comparison with the combinatorial methods which are very inefficient for computation, even for small systems and short runs, the method in this paper is easy to implement and computationally efficient. In comparison with the Markov-chain method, [7], [8], using a nonrecursive formula which is based on partial-fraction expansion allows one to apply it to very large systems at a negligible cost. The only computational problem that might arise is that of accuracy of the partial fraction expansion procedure. In such cases, it is possible to approximate R_n very closely by using only the terms in the partial fraction expansion that are based on the smallest root in absolute value [5, p. 276]. As n increases, the contribution of the smallest root in absolute value dominates the reliability. Another possibility is to use the recursive formulas that can be obtained directly from the generating function [11].

The method in this paper, like the Markov chain embedding method, involves 2 major parts: 1) the problem setup, and 2) obtaining a solution. The Markov chain setup requires specifying the imbedding of the random variable. From [7]: "Often there are several different ways to imbed a random variable. To find the best imbedded Markov chain it requires experience and understanding the structure of the counting process associated with the random variable." The setup in this paper (steps 1–2) requires specifying the failure-causing patterns and writing recursive equations. When multi-state components are involved, the setup can be laborious if done manually. Therefore, they were incorporated into a computer program which yields the required equations automatically and efficiently. The time that this part takes does not depend on n (system size).

Computationally, once the transition matrix has been set up for the Markov chain embedding, obtaining the reliability of a system with n components involves multiplying the matrix with itself n times. The size of the matrix depends on the length of the longest failure-causing pattern. Whe both the order of the matrix and n are large this can cause memory problems. Fu [7] suggests, in such cases, using a recursive equation or a Poisson approximation if the probability of the failure-pattern is very small.

In this paper, the major computational effort is solving a set of linear equations symbolically. This set of equations is always well determined, and its size is a function of the number of failure-causing patterns (e.g., k and r and the number of states the components can be in). It does not, however, depend on n, the number of components in the system. In cases such as the consecutive-k-out-of-n : F and m-consecutive-k-out-of-n : F systems, the linear equations are independent and thus the solution is very easy to obtain. For the r-within-consecutive-k -out-of-n : F systems if r and k are very large, there will be many failure-patterns, and thus a large set of linear equations to be solved.

Appendix
Relation Between Waiting Times and The m-Consecutive-k-out-of-n System Reliability

To obtain the reliability generating function for the m-consecutive-k-out-of-n : F system, use a shortcut that is based on the relations between 2 sets of generating functions.

1. Find the generating function for the waiting time for the first sequence of k consecutive failures in a Bernoulli series ($\mathcal{G}_{WT}(s)$), applying relation (4) to the consecutive-k-out-of-n reliability generating function ($\mathcal{G}(s)$):

$$\mathcal{G}_{WT}(s) = 1-(1-s)\cdot\mathcal{G}(s) = \frac{(p\cdot s)^k\cdot(1-p\cdot s)}{1-s+q\cdot p^k\cdot s^{k+1}}. \tag{A-1}$$

Next, relate this generating function to that for the waiting time for sequence $\#m$ of k consecutive failures in an infinite series of Bernoulli trials, $\mathcal{G}_{WT}^m(s)$. Because the waiting time for sequence $\#m$ is a convolution of m waiting times for the first such sequence, the relation between the 2 generating functions is:

$$\mathcal{G}_{WT}^m(s) = [\mathcal{G}_{WT}(s)]^m. \tag{A-2}$$

Finally, the m-consecutive-k-out-of-n : F reliability is related to the waiting time for the $\#m$ sequence of k consecutive failures in Bernoulli trials through (4), thus yielding the expression in (12):

$$\mathcal{G}(s) = \frac{1-\mathcal{G}_{WT}^{m}(s)}{1-s} = \frac{(1-s+q\cdot p^k\cdot s^{k+1})^m-(p\cdot s)^{m\cdot k}\cdot(1-p\cdot s)^m}{(1-s)(1-s+q\cdot p^k\cdot s^{k+1})^m}. \tag{A-3}$$

REFERENCES

[1] S. Aki and K. Hirano, "Lifetime distribution and estimation problems of consecutive-k-out-of-n : F systems," *Ann. Inst. Statist. Math.*, vol. 48, no. 1, pp. 185–199, 1996.

[2] M. T. Chao, J. C. Fu, and M. V. Koutras, "Survey of reliability studies of consecutive-k-out-of-n : F & related systems," *IEEE Trans. Rel.*, vol. 48, no. 1, pp. 120–127, 1995.

[3] M. T. Chao and G. D. Lin, "Econimical design of large consecutive-k-out-of n : F systems," *IEEE Trans. Rel.*, vol. R-33, pp. 411–413, 1984.

[4] C. Derman, G. J. Liberman, and S. Ross, "On the consecutive-k-out-of-n : F system," *IEEE Trans. Rel.*, vol. R-31, pp. 57–63, 1982.

[5] W. Feller, *An Introduction to Probability Theory and Its Applications*: John Wiley & Sons, 1968, vol. I.

[6] J. C. Fu, "On reliability of a large consecutive-k-out-of-n : F system with $(k-1)$-step Markov dependence," *IEEE Trans. Rel.*, vol. R-36, no. 1, pp. 75–77, 1987.

[7] ——, "Distribution theory of runs and patterns with a sequence of multistate trials," *Statistica Sinica*, vol. 6, pp. 957–974, 1996.

[8] M. Haim and Z. Porat, "Bayes reliability modeling of a multistate consecutive k-out-of-n : F system," in *Proc. Annual Reliability and Maintainability Symp.*, 1991, pp. 582–586.

[9] M. V. Koutras, "On a Markov chain approach for the study of reliability structures," *J. Appl. Prob.*, vol. 33, pp. 357–367, 1996.

[10] G. Shmueli and A. Cohen, "Run-related probability functions applied to sampling inspection," *Technometrics*, vol. 42, no. 2, pp. 188–202, 2000.

[11] R. P. Stanley, *Enumerative Combinatorics*: Wadsworth & Brooks/Cole Advanced Books & Software, 1994, vol. I.

Galit Shmueli is an Assistant Professor at the Department of Decision and Information Technologies, Robert H. Smith School of Business, University of Maryland, College Park. She received her Ph.D. in 2000 in Statistics from the Israel Institute of Technology. Her research interests include industrial applications based on runs and scans, and creating Web applications for statistical methods that are used in industry.