IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 6, 2019, accepted March 28, 2019, date of publication April 4, 2019, date of current version April 16, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2908864

An Efficient Formula for Generalized Multi-State *k*-Out-of- *n*: G System Reliability

TAHA RADWAN[1,2], MUFDA J. ALRAWASHDEH[1], AND SEWELAM GHANEM[1]

[1]Department of Mathematics, College of Sciences and Arts in Ar Rass, Qassim University, Ar Rass 51921, Saudi Arabia
[2]Department of Mathematics and Statistics, Port Said University, Port Said 42511, Egypt

Corresponding author: Taha Radwan (taha_ali_2003@hotmail.com)

This work was supported by the Qassim University, during the academic year 1439AH/2018AD, under Grant 5046.

ABSTRACT The generalized multi-state-k-out-of-n: G system [named GMS(k,n,G)] model was suggested by Huang *et al.* This system contains n components and is very useful for description of the practical systems so that the components, as well as the system, take the states: 0, 1, 2, …, H. The GMS(k,n,G) is in state ≥ j, if k_l components or more are in state ≥ l, where l is an integer and j ≤ l ≤ H. This system has many practical applications; however, the existing methods of evaluation system-reliability with unequal components probabilities are suitable only for some special cases. In this paper, we will suggest efficient formulas to evaluate the exact reliability of GMS(k,n,G) with equal and unequal components probabilities. These formulas are based on the conditional probability and are suitable for all system types: increasing, constant, decreasing, and non-monotone k values system. Also, we will give the theoretical background, computer codes, and various numerical examples for the suggested formulas.

INDEX TERMS k-out-of-n: systems, multi-state systems, performance measures, reliability evaluation.

NOTATION

GMS(k, n, G)	generalized multi-state-k-out-of-n: G system.
GMS(k,n,F)	generalized multi-state-k-out-of-n: F system.
n	the total number of system components.
H	maximum state for the system, and its components.
k_j^g	required number of components for level j of a GMS(k,n,G).
k_j^f	required number of components for level j of a GMS(k,n,F), $k_j^f = n - k_j^g + 1$.
$k(k_1^g, k_2^g, \ldots, k_H^g)$ (k,n,G);	vector for a GMS or $(k_1^f, k_2^f, \ldots, k_H^f)$ vector for a GMS(k,n,F).
x_l	the state of the component l, $x_l \in \{0,1,2,\ldots,H\}$.
$x(x_1), x_2, \ldots, (x_n)$,	vector of the components states.
ϕ(x)	the structure function of the system state, $\phi(x) \in \{0, 1, 2, \ldots, H\}$.
$p_{l,j}$	$\Pr\{x_l = j\}, \sum_{a=0}^{H} p_{l,a} = 1$.
$Q_{l,j}$	$\Pr\{x_l < j\}, Q_{l,j} = 1 - \sum_{a=j}^{H} p_{l,a}$.
p_j	probability that the component state is j, when the components are i.i.d.
Q_j	probability that the component state is less than j, when the components are i.i.d.
Q	$(Q_1), Q_2, \ldots, (Q_n)$.
t_j	number of components that in state j, $0 \le t_j \le n$.
T_j	number of components that in state $\ge j$, $T_j = \sum_{i=j}^{H} t_i$.
y_j	number of components that in state $< j$.
$A_{i,j}$	event that y_j components in state less than j and n-y_j components in state $\ge j$ with condition $T_l < k_l^g$ for all $l > j$.
r_j	$\Pr\{\phi(x) = j\}$.
F_j	$\Pr\{\phi(x) < j\}, F_j = \sum_{a=0}^{j-1} r_a$.
R_j	$\Pr\{\phi(x) \ge j\}, R_j = 1 - F_j = \sum_{a=j}^{H} r_a$.
r	vector of r_j values.
F	vector of F_j values.
R	vector of R_j values.

The associate editor coordinating the review of this manuscript and approving it for publication was Jiajie Fan.

I. INTRODUCTION

The GMS(k, n, G) is an important and widespread model for redundancy of fault tolerant systems, especially in military, and industrial systems. There are many examples of these systems such as: multi display systems of the cockpit, multi-engine systems of the airplane, and multi-transmitter systems of the communication system [2]. Furthermore, there are several practical systems [3], [4], such as: computer system, electrical power generation system, fluid transmission system, communication network, coal transmission system, and the sensor network.

The states of binary systems, as well as components, are only the two states, functional state or failed state. Practically, many systems, as well as the components, take the two states and more. Such systems, are named multi-state systems. The properties of multi-state systems make the analysis of system reliability more difficult (than the properties of binary systems) [5]–[8], whereas in the first time the system state is completely in working state, after some time the system state decreases and becomes partially working state, after more time, the system state decreases again and becomes partially failed state, finally, the system state becomes entirely failed state. We can represent these states as: 0, 1, 2,..., H, where 0 represents completely failed state, and H represents perfect-functioning state. The multi-state systems models are more elastic tool (than the binary systems models) for representing systems. So that, in the last few years, there are many models are generalized from binary state to multi-state, such as, GMS(k, n, G) model [9]–[14], MS consecutive (k, n, G) model [15]–[18], MS consecutive k-out-of-r-from-n: model [19]–[22], and MS linear k-within-(r, s)-of-(m, n): lattice model [23], [24]. In this article, we will evaluate the reliability of GMS(k, n, G) model.

In [1], Huang et al. suggested a definition for the GMS(k, n, G) model, where the required k value depends on the system state. The GMS(k, n, G) is in state $\geq j$ if k_l components or more are in state $\geq l$, where l is an integer and $j \leq l \leq H$. This system is more flexible for describing the practical systems. The GMS(k, n, G) has many practical applications, but even so there is no any method for non-monotone GMS(k, n, G) reliability evaluation with non i.i.d. components, that is the general case for all the other cases. The existing methods are suitable for some special cases only. Many systems contain non i.i.d. components. Evaluating the reliability of such systems is more complex. All systems that contain i.i.d. components are special cases of these systems. Therefore, it is very important to evaluate the reliability of the systems that have non i.i.d. components. Some recent papers generalized many traditional models to the case where components are non i.i.d. For example, the paper [25] generalized the phased mission parallel systems to the case where each component has different capacity. Also, the paper [26] generalized the traditional linear consecutive system to the case where there are three types of components.

In this article, we will suggest efficient formulae to evaluate the exact reliability of GMS(k, n, G), whether in the case of i.i.d. components or non i.i.d. components. These formulae are based on the conditional probability and are suitable for all system types: increasing, constant, decreasing, and non-monotone k values system. Furthermore, we will give the theoretical background and computer codes for the suggested formulae. And so, various numerical examples of the proposed formulae will be given.

II. THE DESCRIPTION OF GMS(k, n, G) AND GMS(k, n, F)

We consider the GMS(k, n, G) discussed in [1]. The generalized multi-state-($k_1^g, k_2^g, \ldots, k_H^g$)-out-of-$n$: G system equivalents to generalized multi-state-($k_1^f, k_2^f, \ldots, k_H^f$)-out-of-$n$: F system, such that $k_j^f = n - k_j^g + 1$ for all $1 \leq j \leq H$. The GMS(k, n, F) is in state $< j$ if at least k_l^f components in state $< l$, where l is an integer number and $j \leq l \leq H$.

The values of k vector categorized GMS(k, n, F) into 4 cases:

Case 1: when $k_1^f \geq k_2^f \geq \ldots \geq k_H^f$, the system is called a decreasing GMS(k, n, F).

Case 2: When $k_1^f \leq k_2^f \leq \ldots \leq k_H^f$, the system is called an increasing GMS(k, n, F).

Case 3: when $k_1^f = k_2^f = \ldots = k_H^f$, the system is called a constant GMS(k, n, F).

Case 4: when the values of k vector cannot be ordered in an ascending, constant, or descending order, the system is called a non-monotone GMS(k, n, F).

Note that, the decreasing GMS(k, n, F) and the increasing GMS(k, n, G) are equivalent. And so, the increasing GMS(k, n, F) and the decreasing GMS(k, n, G) are equivalent too.

In decreasing and constant GMS(k, n, F), if at least k_j^f components are in state $< j$, then the system is in state $< j$. In these cases, if the system cannot meet the requirements of level j, it also cannot meet the requirements of any higher level. So that, the methods used for evaluation the binary system reliability [2], [27] can be applied with consideration the system and its components function for state $\geq j$ and failed for state $< j$. For other cases, increasing and non-monotone GMS(k, n, F), we should use other methods those are suitable with their complex requirements.

III. THE PROPOSED FORMULAE OF GMS(k,n,G) RELIABILITY

Huang *et al.* [1] and Zuo and Tian [9] gave algorithms for evaluation the reliability of GMS(k, n, G) for some special cases: increasing, constant, and decreasing. Suprasad [28] and Mo [13] suggested an algorithm for evaluation of the reliability of GMS(k, n, G) when the components are i.i.d. There is no any algorithm for non-monotone system reliability when the components are not i.i.d. So that, in this section we will suggest efficient formulae to evaluate the exact reliability of GMS(k, n, G) for all system types: increasing, constant, decreasing, and non-monotone with equal and unequal components probabilities.

The state of GMS(k, n, F) can be redefined as follow: $\phi(x) < j$ if there is existed y_j ($k_j^f \leq y_j \leq n$) components in state less than j and $n - y_j$ components in state $\geq j$ with condition $T_l < k_l^g$ for all $l > j$.

$$F_j = \sum_{i=k_j^f}^{n} \Pr(A_{i,j}), \quad forall\ 1 \leq j \leq H, \tag{1}$$

where $\Pr(A_{i,j})$ is probability that y_j components in state less than j and $n - y_j$ components in state $\geq j$ with condition $T_l < k_l^g$ for all $l > j$. Therefore,

$$\Pr(A_{i,j}) = \Pr\{y_j = i, T_j = n - i | T_l \leq k_l^g - 1; \forall l > j\} \tag{2}$$

The value of T_j in eq. (2) depends on the value of y_j, so that:

$$\Pr(A_{i,j}) = \Pr(y_j = i) \times Pr(T_j = n - i | y_j = i, T_l \leq k_l^g - 1; \forall l > j) \tag{3}$$

The last condition in eq. (4) can be written as follows

$$\{T_l \leq k_l - 1; \forall l > j\} = \left\{\begin{array}{l} T_H \leq k_H^g - 1, \\ T_{H-1} \leq k_{H-1}^g - 1, \\ T_{H-2} \leq k_{H-2}^g - 1, \\ \vdots \\ T_{j+1} \leq k_{j+1}^g - 1 \end{array}\right\} \tag{4}$$

But, $t_H \leq t_H + t_{H-1} \leq \ldots \leq t_H + t_{H-1} + \ldots + t_{j+1}$, i.e. $T_H \leq T_{H-1} \leq \ldots \leq T_{j+2} \leq T_{j+1}$, then

$$\{T_l \leq k_l - 1; \forall l > j\} = \left\{\begin{array}{l} T_H \leq \min(k_H^g - 1, \ldots, k_{j+1}^g - 1), \\ T_{H-1} \leq \min(k_{H-1}^g - 1, \ldots, k_{j+1}^g - 1), \\ \vdots \\ T_{j+1} \leq k_{j+1}^g - 1 \end{array}\right\} \tag{5}$$

Using eq(5), the last probability in eq(3) becomes

$$\Pr\{T_j = n - i | y_j = i, T_l \leq k_l^g - 1; \forall l > j\} = \Pr\left\{\begin{array}{l} T_H \leq \min(k_H^g - 1, \ldots, k_{j+1}^g - 1, n - i), \\ T_{H-1} \leq \min(k_{H-1}^g - 1, \ldots, k_{j+1}^g - 1, n - i), \\ \vdots \\ T_{j+1} \leq \min(k_{j+1}^g - 1, n - i), \\ T_j = n - i \end{array}\right\} \tag{6}$$

Putting $m_l = \min(k_l^g - 1, \ldots, k_{j+1}^g - 1); \forall l > j$, then eq.(6) can be reduced to

$$\begin{aligned} &\Pr\{T_j = n - i | y_j = i, T_l \leq k_l^g - 1; \forall l > j\} \\ &= \Pr\left\{\begin{array}{l} T_H \leq \min(n - i, m_H), \\ T_{H-1} \leq \min(n - i, m_{H-1}), \\ \vdots \\ T_{j+1} \leq \min(n - i, m_{j+1}), \\ T_j = n - i \end{array}\right\} \\ &= \Pr\left\{\begin{array}{l} t_H \leq \min(n - i, m_H), \\ t_{H-1} + T_H \leq \min(n - i, m_{H-1}), \\ \vdots \\ t_{j+1} + T_{j+2} \leq \min(n - i, m_{j+1}), \\ t_j + T_{j+1} = n - i \end{array}\right\} \\ &= \Pr\left\{\begin{array}{l} t_H \leq \min(n - i, m_H), \\ t_{H-1} \leq \min(n - i, m_{H-1}) - T_H, \\ \vdots \\ t_{j+1} \leq \min(n - i, m_{j+1}) - T_{j+2}, \\ t_j = n - i - T_{j+1} \end{array}\right\} \\ &= \Pr(t_1 \leq \min(n - i, m_H) | y_j) \\ &\quad \times \Pr(t_2 \leq \min(n - i, m_{H-1}) - T_H | y_j, t_1) \\ &\quad \times \Pr(t_3 \leq \min(n - i, m_{H-2}) - T_{H-1} | y_j, t_1, t_2) \\ &\quad \vdots \\ &\quad \times \Pr(t_{j+1} \leq \min(n - i, m_{j+1}) - T_{j+2} | y_j, t_1, \ldots, t_{j+2}) \\ &\quad \times \Pr(t_j = n - y_j - T_{j+1} | y_j, t_1, \ldots, t_{j+1}) \end{aligned} \tag{7}$$

By substituting form eq7 in eq(3), then

$$\begin{aligned} &\Pr(A_{i,j}) \\ &= \Pr(y_j = i) \times \Pr(t_1 \leq \min(n - i, m_H) | y_j) \\ &\quad \times \Pr(t_2 \leq \min(n - i, m_{H-1}) - T_H | y_j, t_1) \\ &\quad \times \Pr(t_3 \leq \min(n - i, m_{H-2}) - T_{H-1} | y_j, t_1, t_2) \\ &\quad \vdots \\ &\quad \times \Pr(t_{j+1} \leq \min(n - i, m_{j+1}) - T_{j+2} | y_j, t_1, \ldots, t_{j+2}) \\ &\quad \times \Pr(t_j = n - y_j - T_{j+1} | y_j, t_1, \ldots, t_{j+1}) \\ &= \Pr(y_j = i) \times \Pr(0 \leq t_1 \leq \min(n - i, m_H) | y_j) \\ &\quad \times \Pr(0 \leq t_2 \leq \min(n - i, m_{H-1}) - T_H | y_j, t_1) \\ &\quad \times \Pr(0 \leq t_3 \leq \min(n - i, m_{H-2}) - T_{H-1} | y_j, t_1, t_2) \\ &\quad \vdots \\ &\quad \times \Pr(0 \leq t_{j+1} < \min(n - i, m_{j+1}) - T_{j+2} | y_j, t_1, \ldots, t_{j+2}) \\ &\quad \times \Pr(t_j = n - y_j - T_{j+1} | y_j, t_1, \ldots, t_{j+1}) \end{aligned} \tag{8}$$

Now after determination numbers of components those are in state $< j$ ($k_j^f \leq y_j \leq n$), numbers of components that in state $> j$ ($0 \leq t_l < \min(n - y_j, m_l) - T_{l+1}; \forall j + 1 \leq l \leq H$), and numbers of components that in state j ($t_j = n - y_j - T_{j+1}$), we can find easily F_j as follows:

Firstly, when the components have equal probabilities:

$$\begin{aligned} F_j &= \sum_{y=k_j^f}^{n} \binom{n}{y} Q_j^y \beta_{y,j}, \quad forall\ j < H, \\ \beta_{y,j} &= \sum_{t_H=0}^{\min(n-y,m_H)-T_{H+1}} \binom{n - y - T_{H+1}}{t_H} p_H^{t_H} \\ &\cdots \sum_{t_{j+1}=0}^{\min(n-y,m_{j+1})-T_{j+2}} \binom{n - y - T_{j+2}}{t_{j+1}} p_{j+1}^{t_{j+1}} \cdot p_j^{n-y-T_{j+1}} \\ m_l &= \min(k_l^g - 1, \ldots, k_{j+1}^g - 1); \quad \forall l > j \end{aligned}$$

$$T_l = \sum_{a=l}^{H} t_a; \quad \forall j < l \le H, \; T_{H+1} = 0. \tag{9}$$

For $j = H$, eq. (9) is reduced to:

$$F_H = \sum_{y=k_H^f}^{n} \binom{n}{y} Q_H^y . p_H^{n-y} \tag{10}$$

Secondly, when the components have unequal probabilities:

$$F_j = \prod_{i=1}^{n} \sum_{x_i=v_i}^{1} Q_{i,j}^{x_i} . C_{i,j}, \quad \text{for all } j \; H,$$
$$v_i = \max(0, k_j^f - (n-i) - \sum_{a=1}^{i-1} x_a),$$
$$C_{i,j} = C_{i,j}(x_i, T_{i,j+1}, \ldots, T_{i,H})$$
$$= \sum_{t_{i,H}=0}^{\min(T^*_{i,H}, T_{i,H}, \ldots, T_{i,j+1})} p_{i,H}^{t_{i,H}} \cdots \sum_{t_{i,j+1}=0}^{\min(T^*_{i,j+1}, T_{i,j+1})} p_{i,j+1}^{t_{i,j+1}} . p_{i,j}^{T^*_{i,j}}$$
$$T^*_{i,l} = 1 - x_i - \sum_{b=l+1}^{H} t_{i,b},$$
$$T_{i,l} = k_l^g - 1 - \sum_{a=1}^{i-1} \sum_{b=l}^{H} t_{a,b}. \tag{11}$$

For $j = H$, eq. (11) is reduced to:

$$F_j = \prod_{i=1}^{n} \sum_{x_i=v_i}^{1} Q_{i,j}^{x_i} . p_{i,j}^{1-x_i},$$
$$v_i = \max(0, k_j^f - (n-i) - \sum_{a=1}^{i-1} x_a). \tag{12}$$

Further, we can find r_j, and R_j, for all $0 \le j \le H$, from the following equations:

$$r_j = F_{j+1} - F_j; F_0 = 0, \quad F_{H+1} = 1 \tag{13}$$
$$R_j = 1 - F_j = \sum\nolimits_{a=j}^{H} r_a. \tag{14}$$

IV. NUMERICAL EXAMPLES

We obtained the numerical results for evaluation the reliability of GMS(k,n,G) using MATLAB codes which are given in the appendix. These codes were executed on Core i5 CPU with 2.3 GHz under Windows 10 operating system. We calculated CPU times per seconds using *cputime* function. In order to demonstrate the validity of the proposed equations and MATLAB codes, we calculated the reliability of GMS(k,n,G) for some published examples as shown in examples 1-4. All obtained results in these examples using our method match the published results. The proposed equations are illustrated by examples 1 and 2. The examples 5 and 6 are given for evaluation the reliability of non-monotone GMS(k,n,G) type with unequal components probabilities, that are given by table 1.

Illustrated Example 1. Consider example 4 from [28]. Given $n = 4$, $H = 4$, $k^f = (2, 3, 3, 1)$, i.e. $k^g = (3, 2, 2, 4)$, and $p = (p_0, p_1, p_2, p_3, p_4) = (0.1, 0.2, 0.1, 0.4, 0.2)$. Then $Q = (0.1, 0.3, 0.4, 0.8)$. Using equations (9) and eq(10) we have

For state 4.

$$F_4 = \sum_{y=1}^{4} \binom{4}{y} Q_4^y . p_4^{4-y}$$
$$= \binom{4}{1}(0.8)(0.2)^3 + \binom{4}{2}(0.8)^2(0.2)^2 + \binom{4}{3}(0.8)^3(0.2) + \binom{4}{4}(0.8)^4$$
$$= 0.9984$$

For state 3.

$$F_3 = \sum_{y=3}^{4} \binom{4}{y} Q_3^y \beta_{y,3} = \binom{4}{3}(0.4)^3\beta_{3,3} + \binom{4}{4}(0.4)^4\beta_{4,3}$$
$$F_3 = (4)(0.4)^3(p_3 + p_4) + (0.4)^4$$
$$= (4)(0.4)^3(0.4 + 0.2) + (0.4)^4$$
$$= 0.1792$$

where,

$$\beta_{3,3} = \sum_{t_4=0}^{1} \binom{1}{t_4} p_4^{t_4} p_3^{1-t_4} = p_3 + p_4,$$
$$\beta_{4,3} = 1.$$

For state 2.

$$F_2 = \sum_{y=3}^{4} \binom{4}{y} Q_2^y \beta_{y,2} = \binom{4}{3}(0.3)^3\beta_{3,2} + \binom{4}{4}(0.3)^4\beta_{4,2}$$
$$= (4)(0.3)^3(p_2 + p_3 + p_4) + (0.3)^4$$
$$= (4)(0.3)^3(0.1 + 0.4 + 0.2) + (0.3)^4$$
$$= 0.0837$$

where,

$$\beta_{3,2} = \sum_{t_4=0}^{1} \sum_{t_3=0}^{1-t_4} \binom{1}{t_4} \binom{1-t_4}{t_3} p_4^{t_4} p_3^{1-t_4} p_2^{1-t_4-t_3}$$
$$= p_2 + p_3 + p_4$$
$$\beta_{4,2} = 1.$$

For state 1.

$$F_1 = \sum_{y=2}^{4} \binom{4}{y} Q_1^y \beta_{y,1} = (6)(0.1)^2\beta_{2,1} + (4)(0.1)^3\beta_{3,1} + (1)(0.1)^4\beta_{4,1}$$
$$= (6)(0.1)^2(p_1^2 + 2p_2p_1 + 2p_3p_1 + 2p_4p_1) + (6)(0.1)^3(p_1 + p_2 + p_3 + p_4) + (0.1)^4$$

$$= (6)(0.1)^2((0.2)^2 + 2(0.1)(0.2) + 2(0.4)(0.2) + 2(0.2)(0.2)) + (4)(0.1)^3(0.2 + 0.1 + 0.4 + 0.2) + (0.1)^4 = 0.0229$$

where,

$$\beta_{2,1} = \sum_{t_4=0}^{1}\sum_{t_3=0}^{1-t_4}\sum_{t_2=0}^{1-t_4-t_3}\binom{2}{t_4}\binom{2-t_4}{t_3}\binom{2-t_4-t_3}{t_2}p_4^{t_4} \times p_3^{t_3}p_2^{t_2}p_1^{2-t_4-t_3-t_2} = p_1^2 + 2p_2p_1 + 2p_3p_1 + 2p_4p_1$$

$$\beta_{3,1} = \sum_{t_4=0}^{1}\sum_{t_3=0}^{1-t_4}\sum_{t_2=0}^{1-t_4-t_3}\binom{1}{t_4}\binom{1-t_4}{t_3}\binom{1-t_4-t_3}{t_2}p_4^{t_4} \times p_3^{t_3}p_2^{t_2}p_1^{1-t_4-t_3-t_2} = p_1 + p_2 + p_3 + p_4$$

$$\beta_{4,1} = 1$$

Then

$$F = (0.0229, 0.0837, 0.1792)$$
$$r = (0.0299, 0.0608, 0.0955, 0.8192, 0.0016)$$
$$R = (0.9771, 0.9163, 0.8208, 0.0016)$$
$$CPU = 5.8462 \times 10^{-5} seconds$$

Illustrated Example 2. Consider example 8 from [1]. Given $n = 3$, $H = 3$, $k^f = (1, 2, 2)$, i.e. $k^g = (3, 2, 2)$, and $p_{i,j} = \begin{bmatrix} 0.1 & 0.2 & 0.3 & 0.4 \\ 0.1 & 0.1 & 0.2 & 0.6 \\ 0.1 & 0.2 & 0.4 & 0.3 \end{bmatrix}$, therefore, $Q_{i,j} = \begin{bmatrix} 0.1 & 0.3 & 0.6 \\ 0.1 & 0.2 & 0.4 \\ 0.1 & 0.3 & 0.7 \end{bmatrix}$. Using equations (11) and (12) we have

For state 3:

$$F_3 = \sum_{x_1=0}^{1}\sum_{x_2=1-x_1}^{1}\sum_{x_3=2-x_1-x_2}^{1} Q_{1,3}^{x_1}.Q_{2,3}^{x_2}.Q_{3,3}^{x_3} \times p_{1,3}^{1-x_1}.p_{2,3}^{1-x_2}.p_{3,3}^{1-x_3}$$
$$= p_{1,3}.Q_{2,3}.Q_{3,3} + Q_{1,3}.p_{2,3}.Q_{3,3} + Q_{1,3}.Q_{2,3}.p_{3,3} + Q_{1,3}.Q_{2,3}.Q_{3,3}$$
$$= (0.4)(0.4)(0.7) + (0.6)(0.6)(0.7) + (0.6)(0.4)(0.3) + (0.6)(0.4)(0.7) = 0.604$$

For state 2:

$$F_2 = \sum_{x_1=0}^{1}\sum_{x_2=1-x_1}^{1}\sum_{x_3=2-x_1-x_2}^{1} Q_{1,2}^{x_1}.Q_{2,2}^{x_2}.Q_{3,2}^{x_3} \times C_{1,2}.C_{2,2}.C_{3,2}$$
$$= p_{1,2}.Q_{2,2}.Q_{3,2} + p_{1,3}.Q_{2,2}.Q_{3,2} + Q_{1,2}.p_{2,2}.Q_{3,2} + Q_{1,2}.p_{2,3}.Q_{3,2} + Q_{1,2}.Q_{2,2}.p_{3,2} + Q_{1,2}.Q_{2,2}.p_{3,3} + Q_{1,2}.Q_{2,2}.Q_{3,2}$$
$$= (0.3)(0.2)(0.3) + (0.4)(0.2)(0.3) + (0.3)(0.2)(0.3) + (0.3)(0.6)(0.3) + (0.3)(0.2)(0.4) + (0.3)(0.2)(0.3) + (0.3)(0.2)(0.3)$$
$$= 0.174$$

where,

$$C_{1,2} = \sum_{t_{1,3}=0}^{\min(1-x_1,1)} p_{1,3}^{t_{1,3}}.p_{1.2}^{1-x_1-t_{1,2}}$$
$$C_{2,2} = \sum_{t_{2,3}=0}^{\min(1-x_2,1-t_{1,3})} p_{2,3}^{t_{2,3}}.p_{2,2}^{1-x_2-t_{2,3}}$$
$$C_{3,2} = \sum_{t_{3,3}=0}^{\min(1-x_3,1-t_{1,3}-t_{2,3})} p_{3,3}^{t_{3,3}}.p_{3,2}^{1-x_3-t_{3,3}}$$

For state 1:

$$F_1 = \sum_{x_1=0}^{1}\sum_{x_2=0}^{1}\sum_{x_3=\max(0,1-x_1-x_2)}^{1} \times Q_{1,1}^{x_1}.Q_{2,1}^{x_2}.Q_{3,1}^{x_3}.C_{1,1}.C_{2,1}.C_{3,1},$$
$$= p_{1,1}.p_{2,1}.Q_{3,1} + p_{1,1}.p_{2,2}.Q_{3,1} + p_{1,1}.p_{2,3}.Q_{3,1} + p_{1,1}.Q_{2,1}.p_{3,1} + p_{1,1}.Q_{2,1}.p_{3,2} + p_{1,1}.Q_{2,1}.p_{3,3} + p_{1,1}.Q_{2,1}.Q_{3,1} + p_{1,2}.p_{2,1}.Q_{3,1} + p_{1,2}.Q_{2,1}.p_{3,1} + p_{1,2}.Q_{2,1}.Q_{3,1} + p_{1,3}.p_{2,1}.Q_{3,1} + p_{1,3}.Q_{2,1}.p_{3,1} + p_{1,3}.Q_{2,1}.Q_{3,1} + Q_{1,1}.p_{2,1}.p_{3,1} + Q_{1,1}.p_{2,1}.p_{3,2} + Q_{1,1}.p_{2,1}.p_{3,3} + Q_{1,1}.p_{2,1}.Q_{3,1} + Q_{1,1}.p_{2,2}.p_{3,1} + Q_{1,1}.p_{2,2}.Q_{3,1} + Q_{1,1}.p_{2,3}.p_{3,1} + Q_{1,1}.p_{2,3}.Q_{3,1} + Q_{1,1}.Q_{2,1}.p_{3,1} + Q_{1,1}.Q_{2,1}.p_{3,2} + Q_{1,1}.Q_{2,1}.p_{3,3} + Q_{1,1}.Q_{2,1}.Q_{3,1}$$
$$= (0.2)(0.1)(0.1) + (0.2)(0.2)(0.1) + (0.2)(0.6)(0.1) + (0.2)(0.1)(0.2) + (0.2)(0.1)(0.4) + (0.2)(0.1)(0.3) + (0.2)(0.1)(0.1) + (0.3)(0.1)(0.1) + (0.3)(0.1)(0.2) + (0.3)(0.1)(0.1) + (0.4)(0.1)(0.1) + (0.4)(0.1)(0.2) + (0.4)(0.1)(0.1) + (0.1)(0.1)(0.2) + (0.1)(0.1)(0.4) + (0.1)(0.1)(0.3) + (0.1)(0.1)(0.1) + (0.1)(0.2)(0.2) + (0.1)(0.2)(0.1) + (0.1)(0.6)(0.2) + (0.1)(0.6)(0.1) + (0.1)(0.1)(0.2) + (0.1)(0.1)(0.4) + (0.1)(0.1)(0.3) + (0.1)(0.1)(0.1) = 0.11$$

where,

$$C_{1,1} = \sum_{t_{1,3}=0}^{\min(1-x_1,1,1)}\sum_{t_{1,2}=0}^{\min(1-x_1-t_{1,3},1)} p_{1,3}^{t_{1,3}}.p_{1,2}^{t_{1,2}}.p_{1,1}^{1-x_1-t_{1,2}-t_{1,3}}$$
$$C_{2,1} = \sum_{t_{2,3}=0}^{\min(1-x_2,1-t_{1,3},1-t_{1,2}-t_{1,3})}\sum_{t_{2,2}=0}^{\min(1-x_2-t_{2,3},1-t_{1,2}-t_{1,3})} p_{2,3}^{t_{2,3}} \times p_{2,2}^{t_{2,2}}.p_{2,1}^{1-x_2-t_{2,2}-t_{2,3}}$$

TABLE 1. $p_{i,j}$ **for the components.**

i \ j	0	1	2	3
1	0.12	0.14	0.28	0.46
2	0.09	0.19	0.33	0.39
3	0.08	0.19	0.5	0.23
4	0.1	0.2	0.24	0.46
5	0.06	0.35	0.41	0.18
6	0.12	0.14	0.25	0.49
7	0.15	0.23	0.33	0.29
8	0.18	0.24	0.37	0.21
9	0.12	0.23	0.28	0.37
10	0.11	0.16	0.34	0.39
11	0.19	0.21	0.33	0.27
12	0.06	0.27	0.39	0.28
13	0.09	0.31	0.39	0.21
14	0.13	0.15	0.26	0.46
15	0.06	0.3	0.3	0.34
16	0.17	0.22	0.29	0.32
17	0.16	0.14	0.3	0.4
18	0.2	0.3	0.25	0.25
19	0.12	0.22	0.44	0.22
20	0.09	0.15	0.28	0.48

$$C_{3,1} = \sum_{t_{3,3}=0}^{\min(1-x_3,1-t_{1,3}-t_{2,3},1-t_{1,2}-t_{1,3}-t_{2,2}-t_{2,3})} \sum_{t_{3,2}=0}^{\min(1-x_3-t_{3,3},1-t_{1,2}-t_{1,3}-t_{2,2}-t_{2,3})} \times p_{3,3}^{t_{3,3}} . p_{3,2}^{t_{3,2}} . p_{3,1}^{1-x_3-t_{3,2}-t_{3,3}}$$

Then

$$\begin{aligned} F &= (0.11, 0.174, 0.604) \\ r &= (0.11, 0.064, 0.43, 0.396) \\ R &= (0.89, 0.826, 0.396) \\ CPU &= 3.5784 \times 10^{-4} seconds \end{aligned}$$

Example 3. Consider example 2 from [9]. Given $n = 10$, $H = 3$, $k^f = (3, 6, 8)$, i.e. $k^g = (8, 5, 3)$, and $p = (p_0, p_1, p_2, p_3, p_4) = (0.1, 0.3, 0.4, 0.2)$. Then $Q = (0.1, 0.4, 0.8)$. Using the suggested method, we get

$$\begin{aligned} F &= [0.0308,\ 0.1523,\ 0.6778] \\ r &= [0.0308,\ 0.1214,\ 0.5255,\ 0.3222] \\ R &= [0.9692,\ 0.8477,\ 0.3222] \\ CPU &= 6.1387 \times 10^{-5} seconds \end{aligned}$$

Example 4. Consider example 6 from [10]. Given $n = 100$, $H = 7$, $k^f = (10, 15, 20, 25, 30, 35, 40)$ and $p_j = 0.125$ for all $j = 0, 1, 2, \ldots, 7$. Then $k^g = (91, 86, 81, 76, 71, 66, 61)$ and $Q = (0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875)$. Using the suggested method, we get

$$\begin{aligned} F &= [0.81596, 0.99457, 0.99995, 1, 1, 1, 1] \\ r &= [0.81596, 0.17861, 0.00538, 0.00005, 0, 0, 0, 0] \\ R &= [0.18404, 0.00543, 0.00005, 0, 0, 0, 0] \\ CPU &= 3.0542 seconds \end{aligned}$$

Example 5. Consider $n = 20$, $H = 3$, $k^f = (10, 15, 20)$ and $p_{i,j}$ given in table 1. Using the suggested method, we get

$$\begin{aligned} F &= [0, 0.012982, 0.471272], \\ r &= [0, 0.012982, 0.45829, 0.528728], \\ R &= [1, 0.987018, 0.528728]. \\ CPU &= 0.0324 seconds \end{aligned}$$

Example 6. Consider $n = 10$, $H = 3$, $k^f = (3, 8, 6)$ and $p_{i,j}$ given in table 1. Using the suggested method, we get

$$\begin{aligned} F &= [0.001402, 0.00233, 0.763398] \\ r &= [0.001402, 0.000928, 0.761067, 0.236602] \\ R &= [0.998598, 0.99767, 0.236602] \\ CPU &= 7.5342 \times 10^{-4} seconds \end{aligned}$$

V. CONCLUSION AND FUTURE WORK

In this article, we suggested a general method to calculate the exact reliability of GMS(k, n, G). Also, we given the theoretical background for this method. The proposed method is suitable for all system cases: increasing, constant, decreasing, and non-monotone k values system, whether in the case of equal or unequal components probabilities. The main result in this paper is evaluation the reliability of non-monotone GMS(k, n, G) with unequal components probabilities. This type is the general case for all the other cases. The reliability of this type is not found previously, and is more complex. So that, all the published results are considered as special cases by our results. Furthermore, complete MATLAB codes for evaluating of system reliability are given. The validity of the proposed method and MATLAB codes can be examined by published examples, when available.

As an extension of our results, we will generalize these results for evaluation the reliability of other models that are more general. Such as multi-state consecutive k-out-of-r-from-n: G system model [19]–[22] and multi-state linear k-within-(m, s)-of-(m, n):G lattice system model [23], [24].

APPENDIX

A. MATLAB CODES FOR GMS(k,n,G) RELIABILITY WITH i.i.d. COMPONENTS

```
%%%%%%% Inputs for example 4 from [28]
n = 4; H = 4; kf = [2 3 3 1];
p = [0.1 0.2 0.1 0.4 0.2];
%%%%%%% Derived inputs & Boundary values
kg = n-flip(kf); Q = cumsum(p); F(H+1) = 1;
%%%%%%% System Reliability Evaluation
for j = H: − 1: 1
F(j) = 0; T = 0; T1 = 0;
for L = 1: H−j
T1(L) = min(kg(L:H−j));
end
```

```
12) for y = kf(j): n
13) Pr(1:H−j+1) = nchoosek(n,y) ∗ (Q(j)^y);
14) F(j) = F(j) + Pr(1) ∗ (p(j+1)^(n-y));
15) t(1:H−j+1) = 0;
16) T(2:H−j+1) = min(n-y,T1)- cumsum(t(1:H−j));
17) while isequal(T,t) == 0
18) for g = H−j+1: −1: 2
19) if t(g) < T(g)
20) t(g) = t(g)+1;
21) t(g+1:H−j+1) = 0;
22) Pr(g:H−j+1) = Pr(g−1) ∗ (nchoosek(n-y-sum
    (t(1:g−1)),t(g))) ∗(p(H−g+3)^t(g));
23) F(j) = F(j) + Pr(g) ∗ (p(j+1)^(n-y-sum(t(2:H-j+1))));
24) break
25) end
26) end
27) T(2:H−j+1) = min(n−y,T1)- cumsum(t(1:H−j));
28) end
29) end
30) r(j) = F(j+1)-F(j);
31) end
32) %%%%%%% the Results
33) r = [F(1),r], F = F(1:H), R(1:H) = 1-F(1:H)
```

B. MATLB CODES FOR GMS(k,n,G) RELIABILITY EVALUATION WITH NOT i.i.d. COMPONENTS

```
1) %%%%%%% Inputs for example 8 from [1]
2) n = 3; H = 3; kf = [1 2 2];
3) p = [0.1 0.2 0.3 0.4; 0.1 0.1 0.2 0.6; 0.1 0.2 0.4 0.3];
4) %%%%%%% Derived inputs & Boundary values
5) kg = n-flip(kf); F(H+1)= 1; y = 0; Pr(1)= 1;
6) for i = 2: n+1
7) Q(i,1:H) = cumsum(p(i,1:H));
8) end
9) %%%%%%% System Reliability Evaluation
10) for j = H: −1: 1
11) for i = 2: n+1
12) y(i) = max(0, kf(j)−n+(i−1)-sum(y(2:i−1)));
13) t(i, H−j+2)= 1-y(i);
14) Pr(i) = Pr(i-1) ∗ (Q(i,j)^y(i)) ∗ (p(i,j+1)^t(i,H−j+2));
15) end
16) t(1:n+1,1:H−j+1) = 0;
17) F(j) = F(j) + Pr(n+1);
18) while sum(y) < n
19) for e = n+1: −1: 2
20) if y(e) == 1
21) continue
22) end
23) Cht = 0;
24) C = cumsum(t(1:e−1, 1:H−j+1));
25) D(e-1,2:H−j+1)= kg(1:H-j) - cumsum(C(e-1,
    2:H−j+1));
26) for g = H−j+1: −1: 2
27) if (1-y(e)-sum(t(e,2:g))) > 0 & min(D(e-1,g:H−j+1))
    > 0
28) Cht = 1;
29) t(e,H−j+2) = 0;
30) t(e:n+1,1:H−j+1) = 0;
31) t(e,g) = 1;
32) Pr(e) = Pr(e-1) ∗ p(e, H−g+3 );
33) for i = e+1: n+1
34) y(i) = max(0, kf(j)−n+(i−1)-sum(y(2:i−1)));
35) t(i,H−j+2) = 1−y(i);
36) Pr(i) = Pr(i-1) ∗ (Q(i,j)^y(i)) ∗ (p(i,j+1)^t(i,H−j+2));
37) end
38) break
39) end
40) end
41) if Cht ==1
42) break
43) end
44) y(e) = 1;
45) Pr(e) = Pr(e-1) ∗ Q(e, j );
46) t(e,H−j+2) = 0;
47) for i = e+1: n+1
48) y(i) = max(0, kf(j)−n+(i−1)-sum(y(2:i−1)));
49) t(i,H−j+2) = 1-y(i);
50) Pr(i) = Pr(i-1) ∗ (Q(i,j)^y(i)) ∗ (p(i,j+1)^t(i,H−j+2));
51) end
52) t(e:n+1,1:H−j+1) = 0;
53) break
54) end
55) F(j) = F(j) + Pr(n+1);
56) end
57) r(j) = F(j+1)−F(j);
58) end
59) %%%%%%% The Results
60) r = [F(1),r], F = F(1:H), R(1:H) = 1−F(1:H)
```


ACKNOWLEDGMENT

The authors gratefully acknowledge Qassim University, represented by the Deanship of Scientific on the material support for this research under the number (5046) during the academic year 1439 AH/2018 AD.

REFERENCES

[1] J. Huang, M. J. Zuo, and Y. Wu, "Generalized multi-state k-out-of-n: G systems," *IEEE Trans. Rel.*, vol. 49, no. 1, pp. 105–111, Mar. 2000.

[2] W. Kuo and M. J. Zuo, *Optimal Reliability Modeling: Principles and Applications*. Hoboken, NJ, USA: Wiley, 2003.

[3] T. Aven and U. Jensen, *Stochastic Models in Reliability*. New York, NY, USA: Springer, 1999.

[4] A. Lisnianski and G. Levitin, *Multi-State System Reliability* (Assessment, Optimization and Applications). Singapore: World Scientific, 2003.

[5] J. Xue and K. Yang, "Dynamic reliability analysis of coherent multistate systems," *IEEE Trans. Rel.*, vol. 44, no. 4, pp. 683–688, Dec. 1995.

[6] R. D. Brunelle and K. C. Kapur, "Review and classification of reliability measures for multistate and continuum models," *IIE Trans.*, vol. 31, no. 12, pp. 1171–1180, 1999.

[7] A. Lisnianski, "Extended block diagram method for a multi-state system reliability assessment," *Rel. Eng. Syst. Saf.*, vol. 92, no. 12, pp. 1601–1607, 2007.

[8] J. D. Murchland, "Fundamental concepts and relations for reliability analysis of multi-state systems," *Rel. Fault Tree Anal., Theor. Appl. Aspects Syst. Rel.*, pp. 581–618, 1975.

[9] M. J. Zuo and Z. Tian, "Performance evaluation of generalized multi-state k-out-of-n systems," *IEEE Trans. Rel.*, vol. 55, no. 2, pp. 319–327, Jun. 2006.
[10] Z. Tian, R. C. M. Yam, M. J. Zuo, and H.-Z. Huang, "Reliability bounds for multi-state k-out-of-n systems," *IEEE Trans. Rel.*, vol. 57, no. 1, pp. 53–58, Mar. 2008.
[11] Z. Tian, M. J. Zuo, and R. C. M. Yam, "Multi-state k-out-of-n systems and their performance evaluation," *IIE Trans.*, vol. 41, no. 1, pp. 32–44, 2008.
[12] X. Zhao and L. Cui, "Reliability evaluation of generalised multi-state k-out-of-n systems based on FMCI approach," *Int. J. Syst. Sci.*, vol. 41, no. 12, pp. 1437–1443, 2010.
[13] Y. Mo, L. Xing, S. V. Amari, and J. B. Dugan, "Efficient analysis of multi-state k-out-of-n systems," *Rel. Eng. Syst. Saf.*, vol. 133, pp. 95–105, Jan. 2015.
[14] X. Zhao, C. Wu, S. Wang, and X. Wang, "Reliability analysis of multi-state k-out-of-n: G system with common bus performance sharing," *Comput. Ind. Eng.*, vol. 124, pp. 359–369, Oct. 2018.
[15] J. Huang, M. J. Zuo, and Z. Fang, "Multi-State Consecutive- k -out-of- n Systems," *IIE Trans.*, vol. 35, no. 6, pp. 527–534, 2003.
[16] M. J. Zuo, Z. Fang, J. Huang, and X. Xu, "Performance evaluation of decreasing multi-state consecutive k-out-of-n: G systems," *Int. J. Rel., Qual. Saf. Eng.*, vol. 10, no. 3, pp. 345–358, 2003.
[17] H. Yamamoto, M. J. Zuo, T. Akiba, and Z. Tian, "Recursive formulas for the reliability of multi-state consecutive-k-out-of-n: G systems," *IEEE Trans. Rel.*, vol. 55, no. 1, pp. 98–104, Mar. 2006.
[18] S. Belaloui and B. Ksir, "reliability of a multi-state consecutive k-out-of-n: G system," *Int. J. Rel., Qual. Saf. Eng.*, vol. 14, no. 4, pp. 361–377, 2007.
[19] A. Habib, R. Al-Seedy, and T. Radwan, "Reliability evaluation of multi-state consecutive k-out-of-r-from-n: G system," *Appl. Math. Model.*, vol. 31, pp. 2412–2423, Nov. 2007.
[20] T. Radwan, A. Habib, R. Alseedy, and A. Elsherbeny, "Bounds for increasing multi-state consecutive k-out-of-r-from-n: F system with equal components probabilities," *Appl. Math. Model.*, vol. 35, pp. 2366–2373, May 2011.
[21] G. Levitin, "Consecutive k-out-of-r-from-n system with multiple failure criteria," *IEEE Trans. Rel.*, vol. 53, no. 3, pp. 394–400, Sep. 2004.
[22] G. Levitin, "Reliability of linear multistate multiple sliding window systems," *Nav. Res. Logistics*, vol. 52, no. 3, pp. 212–223, 2005.
[23] A. Habib, R. Alseedy, A. Elsherbeny, and T. Radwan, "Generalization for lattice system with equal components," *Appl. Math. Sci.*, vol. 3, no. 30, pp. 1479–1486, 2009.
[24] R. Taha, "On system reliability of increasing multi-state linear k-within-(m, s)-of-(m, n):F lattice system," *Eksploatacja I Niezawodnosc-Maintenance Reliability*, vol. 20, no. 1, pp. 73–82, 2017.
[25] R. Peng, Q. Zhai, L. Xing, and JunYang, "Reliability of demand-based phased-mission systems subject to fault level coverage," *Rel. Eng. Syst. Saf.*, vol. 121, no. 4, pp. 18–25, 2014.
[26] R. Peng and H. Xiao, "Reliability of linear consecutive-k-out-of-n systems with two change points," *IEEE Trans. Rel.*, vol. 67, no. 3, pp. 1019–1029, Sep. 2018.
[27] S. V. Amari, M. J. Zuo, and G. Dill, "O(kn) algorithms for analyzing repairable and non-repairable k-out-of-n: G systems," in *Handbook of Performability Engineering*, K. B. Misra, Ed. London, U.K.: Springer, 2008, pp. 309–320.
[28] S. V. Amari, M. J. Zuo, and G. Dill, "A fast and robust reliability evaluation algorithm for generalized multi-state k-out-of-n systems," *IEEE Trans. Rel.*, vol. 58, no. 1, pp. 88–97, Mar. 2009.

TAHA RADWAN received the B.S., M.S., and Ph.D. degrees in pure mathematics from Menofia University, in 1998, 2006, and 2010, respectively. His research interests include system reliability, probability, applied statistics, and pure mathematics.

MUFDA J. ALRAWASHDEH received the B.S. degree from Zarqa National University, in 2004, the M.S. degree from Malaysian National University, in 2009, and the Ph.D. degree in applied statistics from the Science of Malaysia, in 2014. His research interest is applied statistics.

SEWELAM GHANEM received the B.S., M.S., and Ph.D. degrees in applied mathematics from Tanta University, in 1992, 1997, and 2002, respectively. His research interest includes applied mathematics.