British Journal of Mathematics & Computer Science
21(6): 1-15, 2017; Article no.BJMCS.33161
ISSN: 2231-0851

SCIENCEDOMAIN *international*
www.sciencedomain.org

An Application of Reliability-analysis Techniques in Project Management

Ali Muhammad Ali Rushdi[1] and Alaa Mohammad Alturki[1*]

[1]King Abdulaziz University, P.O.Box 80204, Jeddah 21589, Saudi Arabia.

Authors' contributions

This work was carried out in collaboration between both authors. Author AMAR designed the study, managed the analysis, wrote the first draft of the manuscript and managed literature survey. Author AMA contributed to the analysis, implemented the algorithm, drew the figures, and contributed to literature survey. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/BJMCS/2017/33161
Editor(s):
(1) Mohamed Rabea Eid Said, Department of Science and Mathematics, Assiut University, Egypt.
Reviewers:
(1) Xingting Wang, Temple University, United States.
(2) Abdullah Sonmezoglu, Bozok University, Turkey.
(3) Harish Garg, School of Mathematics, Thapar University Patiala, India.
(4) Ruffin-Benoît M. Ngoie, Institut Supérieur Pédagogique de Mbanza-Ngungu, Congo.
Complete Peer review History: http://www.sciencedomain.org/review-history/18792

Original Research Article

Received: 31st March 2017
Accepted: 18th April 2017
Published: 25th April 2017

Abstract

This paper handles a prominent problem of project management, namely that of project scheduling under uncertainty. The paper models this problem as a vector-weighted voting system and expresses the indicator variable for the successful (on-time) completion of project activities as a vector-threshold Boolean function. The paper presents a solution of the problem in the Boolean domain using solely Boolean tools. The paper also presents another solution of the problem via recursive relations and boundary conditions. This solution is given as an insightful visual representation in terms of a signal-flow graph that resembles the Reduced-Ordered Binary Decision Diagram (ROBDD). The effect of the order of variables used in the recursive method is also explored.

Keywords: Project management; system reliability; uncertain activities; Vector-threshold Boolean model; ROBDD-like signal flow graph.

**Corresponding author: E-mail: amtamt666@gmail.com, amt666@hotmail.com;*

1 Introduction

A prominent problem of project management is the problem of project scheduling in which the pertinent project activities are not all known with certainty. The expected durations of the activities might be known only as best guesses, and the *actual* times of various activities can vary due to unexpected delays, worker illnesses and so on [1]. This problem of project scheduling is studied herein via techniques of system reliability [2], as it is modeled via a vector-weighted voting system [3-7], which is a useful generalization of a conventional weighted voting system. We note that a weighted voting system can be studied via a threshold Boolean function [8], or equivalently, as a weighted k-out-of-n system [9]. Hence, the current vector-weighted voting system can be studied via a vector-threshold Boolean function, or equivalently as a two-stage weighted k-out-of-n system [10].

The vector-threshold Boolean model has certain resemblance with (albeit, with subtle differences from) the double-threshold Boolean model of Rushdi and Bjaili [11]. Both models are useful and practical extensions of the (single-) threshold Boolean model [12,13], which is commonly (and inadvertently) referred to as the weighted k-out-of-n model [14]. However, typically, the vector-threshold model produces inequalities in the same direction, while the double-threshold model produces sandwiching or interval-type inequalities i.e., inequalities in the opposite lower and upper sides. Therefore, a vector-threshold Boolean model is typically a coherent model, while a double-threshold Boolean model is always a noncoherent one (except in some of its limiting cases). The vector-threshold model involves $k \geq 2$ inequalities (that might be reduced if some inequalities subsume or dominate others), while the double-threshold one has exactly two inequalities that cannot be reduced.

The organization of the rest of this paper is as follows. Section 2 introduces a well-known problem of project scheduling that appeared earlier in Winston [1] and Chen and Yang [10]. This section uses Karnaugh-map representations for pseudo-Boolean functions to obtain the indicator variable for the successful (on-time) completion of the project as a probability ready expression [2,11,13,15-21], which is immediately converted on a one-to-one basis into a probability expression. Section 3 interprets the current problem as one dealing with a vector-threshold Boolean function and introduces the pertinent recursive relations and boundary conditions. Section 4 uses ROBDD-like signal flow graphs to compute the probability of project prompt completion. This section also explores the effect of variable ordering on the complexity of computations. Section 5 concludes the paper.

2 A Typical Problem in Project Management

This example originally appeared in a classical text on Operations Research by Winston [1], and was later discussed in a prominent reliability paper by Chen and Yang [10]. The example deals with the management of a project dealing with the assembly of two products, labeled 1 and 2, to make a new composite product. The project consists of six activities (described in Table 1), which are sequenced according to the network of Fig. 1.

Table 1. Description of the activities comprising the project in Fig. 1

Activity **i**	**Immediate predecessors**	**Minimum duration D_i (days)**	**Possible delay (days) d_i (days)**
A = train workers	None	6	1
B = purchase raw materials	None	9	3
C = produce product 1	A, B	8	3
D = produce product 2	A, B	7	2
E = test product 2	D	10	2
F = assemble products 1 & 2	C, E	12	4

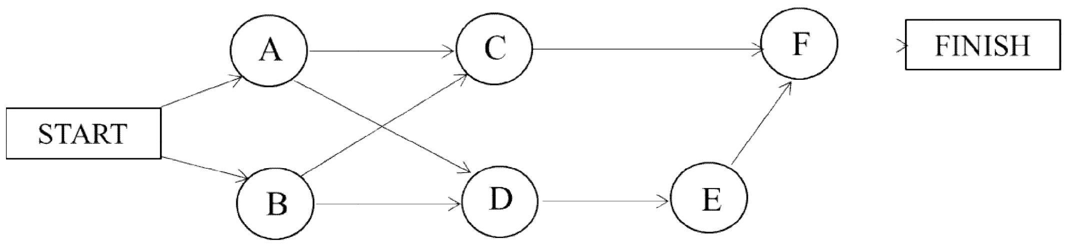


Fig. 1. Sequencing of activities for the project in Sec. 2

Let X_i (i = A, B, C, D, E, F) denote the indicator variable of completing activity i on (minimum) time, *i.e.,* X_i = 1 when activity i is not delayed and $X_i = 0$ when activity i is delayed. Also let $p_i = E\{X_i\}$ denote the probability of completing activity i without delay. Hence, $(1 - p_i) = E\ \{\overline{X}_i\}$ is the probability of experiencing a delay in implementing activity i. The indicators X_i and $\overline{X}_i$ are called, respectively, the success and failure for promptness of activity i. Now, suppose that the deadline of the whole project is 40 days, and use S to denote project success (its completion within the allowed period). Hence, R = E{S} is the probability of project success. This probability is analogous to system reliability, while the probabilities p_i(i = A, B, …, F) look quite similar to component reliabilities. The problem of identifying R as a function of the p_i's

$$R = R(p_A, p_B, p_C, p_D, p_E, p_F). \tag{1}$$

is exactly the problem of expressing system reliability in terms of component reliabilities. Fig. 1 indicates that there are four paths through the project network, namely paths ACF, ADEF, BCF, and BDEF. We now study the duration of each path as a pseudo-Boolean function of activity successes, wherein a pseudo-Boolean function has a Boolean-valued input or domain and a real-valued output or range [12,13,22-27]. Fig. 2(a) displays a Karnaugh-map for the pseudo-Boolean function T_{ACF} which depicts symbolically the time required by the path ACF as a function of X_A, X_C and X_F. Fig. 2(b) represents T_{ACF} numerically, while Fig. 2(c) represents the success S_{ACF} which is the indicator for the event $T_{ACF} \leq 40$. This figure shows that

$$S_{ACF} = 1, \tag{2a}$$

i.e., the path ACF is always successful. In typical reliability jargon, such a prefect success is usually a fictitious matter, and the function S_{ACF} is a noncoherent function. Figs. 3-5 display Karnaugh maps for T_i (numerically) and for S_i, where i denotes the three remaining paths ADEF, BCF, and BDEF. The successes of the three paths (written as disjoint sum-of-products expressions or probability-ready expressions) are

$$S_{ADEF} = X_F \vee X_D X_E \overline{X}_F, \tag{2b}$$

$$S_{BCF} = 1, \tag{2c}$$

$$S_{BDEF} = X_B X_F (X_E \vee X_D \overline{X}_E). \tag{2d}$$

The system success S is the conjunction (ANDing) of the four functions in (2a) - (2d), namely

$$\begin{aligned} S &= S_{ACF}\ S_{ADEF} S_{BCF}\ S_{BDEF} \\ &= (1)\ (X_F \vee X_D X_E \overline{X}_F)\ (1)\ X_B X_F\ (X_E \vee X_D \overline{X}_E)) \\ &= X_B X_F\ (X_E \vee X_D \overline{X}_E) \\ &= S_{BDEF}. \end{aligned} \tag{3}$$

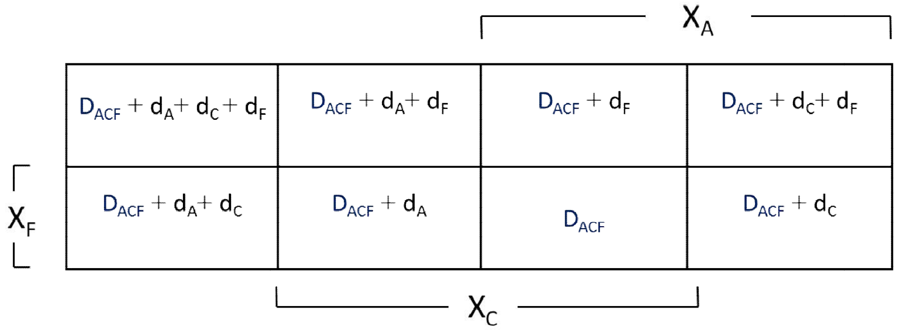
The result in (3) that $S = S_{BDEF}$ should have been anticipated since $S_{BDEF} \leq S_{ACF}$, $S_{BDEF} \leq S_{ADEF,}$ and $S_{BDEF} \leq S_{BCF}$. The success expression (3) is called a probability-ready expression (PRE), since it is one in which [2].

(a) All ORed terms (products) are disjoint, and
(b) All ANDed terms (sums) are statistically independent.

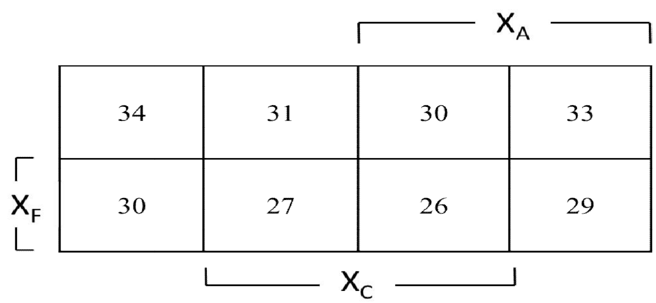
Therefore, this expression is converted into a reliability expression (in a one-to-one basis) by replacing Boolean variables by their expectations and Boolean operators by their arithmetic counterparts, namely

$$R = p_B\, p_F\, (p_E + p_D q_E). \tag{3a}$$

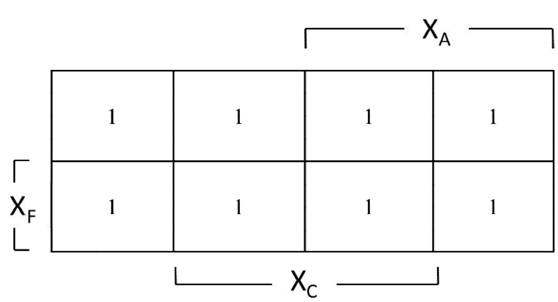
The answer in (3a) explicitly states a result of Chen and Yang [10], but it is obtained using more streamlined concepts and tools.



(a) T_{ACF} (symbolically)

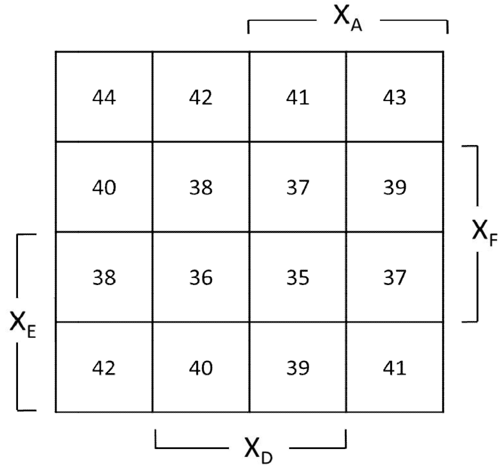


(b) T_{ACF} (symbolically)

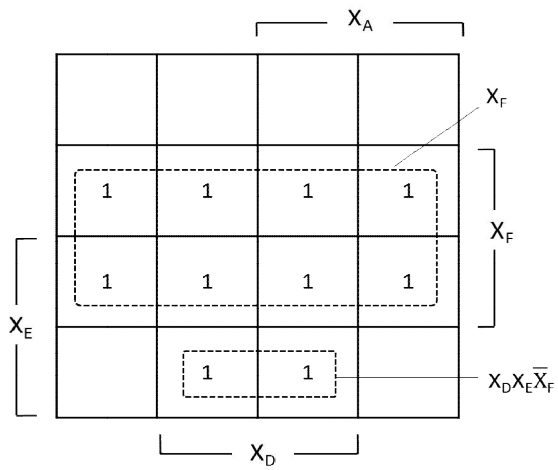


(c) S_{ACF} = Indicator $\{T_{ACF} \leq 40\}$

Fig. 2. Karnaugh maps for the pseudo-Boolean function T_{ACF}, and the Boolean function S_{ACF}. Here $D_{ACF} = D_A + D_C + D_F = 26$

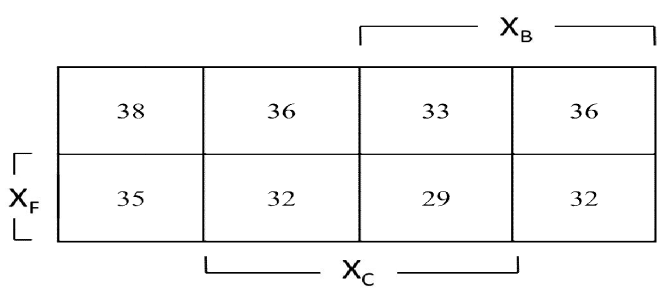


(a) T_{ADEF}

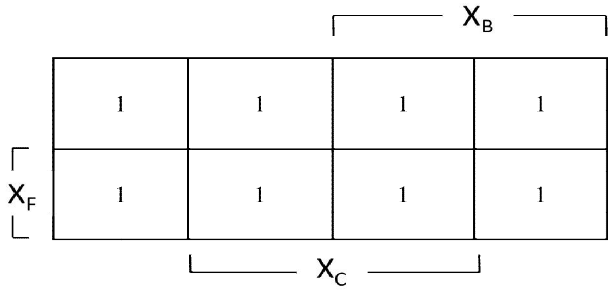


(b) S_{ADEF} = Indicator {$T_{ADEF} \leq 40$}

Fig. 3. Karnaugh maps for the pseudo-Boolean function T_{ADEF}, and the Boolean function S_{ADEF}. Here $D_{ADEF} = 35$

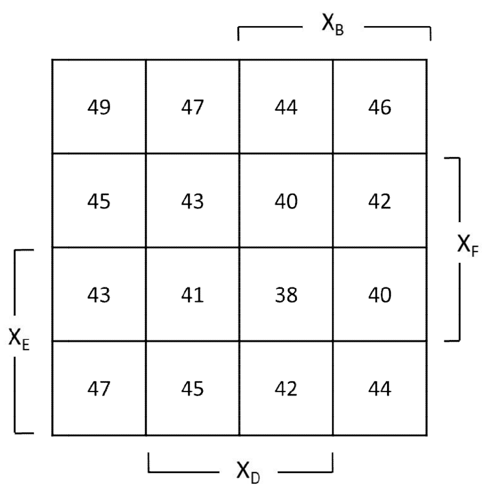


(a) T_{BCF}

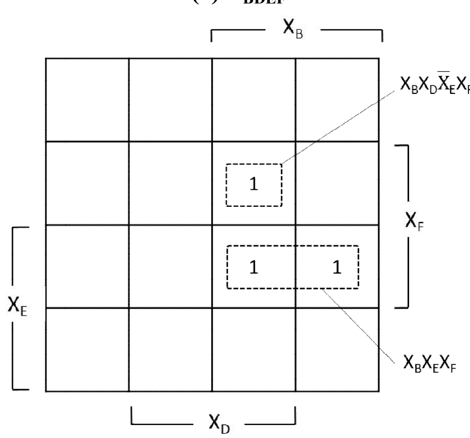


(b) S_{BCF} = Indicator {$T_{BCF} \leq 40$}

Fig. 4. Karnaugh maps for the pseudo-Boolean function T_{BCF}, and the Boolean function S_{BCF}. Here $D_{BCF} = 29$



(a) T_{BDEF}



(b) S_{BDEF} = Indicator {$T_{BDEF} \leq 40$}

Fig. 5. Karnaugh maps for the pseudo-Boolean function T_{BDEF}, and the Boolean function S_{BDEF}. Here $D_{BDEF} = 38$

Before closing this section, we note a remarkable advantage of the current solution via the pseudo-Boolean functions represented by the Karnaugh maps of Figs. 2(a)-5(a). It allows us to study the variation of the probability of project success R with the maximum allowable duration D_m of the project. Table 2 partially reports the result of such a study and shows that R is monotonically non-decreasing with D_m, and it goes down to its minimum value of 0 for $D_m \leq 37$, and attains its maximum value of 1 when $D_m \geq 50$.

Table 2. Variation of probability of project success with the maximum allowed project duration D_m

D_m	S_{ACF}	S_{ADEF}	S_{BCF}	S_{BDEF}	S	R	R(iid)
37	1	-----	-----	0	0	0	0
38	1	$X_E X_F \vee X_D \overline{X}_E X_F$	1	$X_B X_D X_E X_F$	$X_B X_D X_E X_F$	$p_B p_D p_E p_F$	p^4
39	1	$X_E X_F \vee X_D \overline{X}_E X_F \vee$ $X_A \overline{X}_D \overline{X}_E X_F \vee$ $X_A X_D X_E \overline{X}_F$	1	$X_B X_D X_E X_F$	$X_B X_D X_E X_F$	$p_B p_D p_E p_F$	p^4
40	1	$X_F \vee X_D X_E \overline{X}_F$	1	$X_B X_E X_F \vee$ $X_B X_D \overline{X}_E X_F$	$X_B X_F (X_E \vee X_D \overline{X}_E)$	$p_B p_F (p_E + p_D q_E)$	$2p^3 - p^4$
----	----	----	----	----	----	----	----
50	1	1	1	1	1	1	1

3 Problem Interpretation in terms of Vector-threshold Boolean Functions

A Threshold Boolean function S(**X**) of weights $\mathbf{W} = [W_1\ W_2\ \ldots\ W_n]^T$ and a threshold T is specified by [11-13]

$$\{S(\mathbf{X}) = 1\} \Leftrightarrow \{\mathbf{W}^T\mathbf{X} \equiv \sum_{i=1}^{n} W_i\ X_i \geq T\}. \quad (4)$$

Every certain instance **A** of $\mathbf{X} \in \{0, 1\}^n$ represents a truth-table line or a Karnaugh-map cell and is described by the minterm

$$\mathbf{X}^{\mathbf{A}} = X_1^{a_1} X_2^{a_2} \ldots X_{\mathbf{n}}^{a_{\mathbf{n}}}, \quad (5)$$

Where

$$X_i^{a_i} = \begin{cases} \overline{X_i} & \text{if } a_i = 0 \\ X_i & \text{if } a_i = 1. \end{cases} \quad (6)$$

The general functional condition (4) is equivalent to 2^ninequalities written for every instance of $\mathbf{X} \in \{0, 1\}^n$.

$$\sum_{i=1}^{n} W_i^{(a_i)} \geq T, \qquad \mathbf{A} \in \{0, 1\}^n. \quad (7)$$

Where

$$W_i^{(a_i)} = \begin{cases} 0 & \text{if } a_i = 0 \\ W_i & \text{if } a_i = 1. \end{cases} \quad (8)$$

For comparison, path success for any of the paths in Sec. 2 can be written in the cellwise form

$$D_p + \sum_{i=1}^{n} f(d_i, a_i) \leq D_m, \qquad \mathbf{A} \in \{0, 1\}^n \quad (9)$$

Where D_p = expected duration of the path and D_m = maximum duration of the project, and f (d_i) is given by

$$\sum_{i=1}^{n} f(d_i, a_i) = d_i - d_i^{(a_i)} = \begin{cases} d_i & \text{if} \quad a_i = 0 \\ 0 & \text{if} \quad a_i = 1. \end{cases} \tag{10}$$

We rearrange terms in (9), and then multiply both its sides by (–1) {taking care to reverse the direction of the inequality} to obtain

$$D_p + \sum_{i=1}^{n} (d_i - d_i^{(a_i)}) \leq D_m, \quad \mathbf{A} \in \{0, 1\}^n$$
$$\sum_{i=1}^{n} d_i^{(a_i)} \geq (D_p + \sum_{i=1}^{n} d_i - D_m), \qquad \mathbf{A} \in \{0, 1\}^n. \tag{11}$$

Comparing (11) to (7), we conclude that any path success function in our example is a threshold function of a component weight d_i and a threshold equal to ($D_p + \sum_{i=1}^{n} d_i - D_m$). Applying this result to the four paths in our running example, we obtain

$$\{S_{ACF} = 1\} \Leftrightarrow \{X_A + 3\, X_C + 4\, X_F \geq -6\}, \tag{12a}$$

$$\{S_{ADEF} = 1\} \Leftrightarrow \{X_A + 2\, X_D + 2\, X_E + 4\, X_F \geq 4\}, \tag{12b}$$

$$\{S_{BCF} = 1\} \Leftrightarrow \{3\, X_B + 3\, X_C + 4\, X_F \geq -1\}, \tag{12c}$$

$$\{S_{BDEF} = 1\} \Leftrightarrow \{3\, X_B + 2\, X_D + 2\, X_E + 4\, X_F \geq 9\}. \tag{12d}$$

Equations (12) can be considered as a single equation representing a vector-threshold voting system [3-7], or a vector-threshold Boolean function

$$\begin{matrix} & [X_A & X_B & X_C & X_D & X_E & X_F] & & \end{matrix}$$
$$\{\mathbf{S} = 1\} \Leftrightarrow \left\{ \begin{bmatrix} 1 & 0 & 3 & 0 & 0 & 4 \\ 1 & 0 & 0 & 2 & 2 & 4 \\ 0 & 3 & 3 & 0 & 0 & 4 \\ 0 & 3 & 0 & 2 & 2 & 4 \end{bmatrix} \geq \begin{bmatrix} -6 \\ 4 \\ -1 \\ 9 \end{bmatrix} \right\}. \tag{13}$$

In (13), we wrote the component-success vector as a row above the weight matrix rather than as a column after it. This practice facilitates reading matrix equations considerably [28,29]. Note that the weight matrix has the same nonzero entries in every column (representing the delay for the corresponding component). The inequalities in rows 1 and 3 are identically satisfied and could be omitted to reduce (13) to

$$\begin{matrix} & [X_A & X_B & X_C & X_D & X_E & X_F] & & \end{matrix}$$
$$\{\mathbf{S} = 1\} \Leftrightarrow \left\{ \begin{bmatrix} 1 & 0 & 0 & 2 & 2 & 4 \\ 0 & 3 & 0 & 2 & 2 & 4 \end{bmatrix} \geq \begin{bmatrix} 4 \\ 9 \end{bmatrix} \right\}. \tag{14}$$

The vector equation (14) consists of two equations, the second of which implies the first, *viz*.

$$\{3\, X_B + 2\, X_D + 2\, X_E + 4\, X_F \geq 9\}$$

$$=> \{2\, X_D + 2\, X_E + 4\, X_F \geq 9 - 3 = 6\}$$

$$=> \{X_A + 2\, X_D + 2\, X_E + 4\, X_F \geq 6\}$$

$$=> \{X_A + 2\, X_D + 2\, X_E + 4\, X_F \geq 4\}, \tag{15}$$

which means that Equation (14) can be reduced to the form

$$\{\mathbf{S}=1\} \Leftrightarrow \left\{ \begin{matrix} [X_A & X_B & X_C & X_D & X_E & X_F] \\ [0 & 3 & 0 & 2 & 2 & 4] \end{matrix} \geq [9] \right\} \quad (16)$$

In the sequel, we will retain the vector threshold form (14). Since component C is irrelevant (the column under X_C is all 0), we omit the variable X_C and rewrite (14) as

$$[X_A \ X_B \ X_D \ X_E \ X_F]$$

$$\{\mathbf{S}=1\} \Leftrightarrow \left\{ \begin{bmatrix} 1 & 0 & 2 & 2 & 4 \\ 0 & 3 & 2 & 2 & 4 \end{bmatrix} \geq \begin{bmatrix} 4 \\ 9 \end{bmatrix} \right\}. \quad (14a)$$

4 Probability of Project Prompt Completion

In this section, we use techniques of system reliability [2,11-21,30,31] to evaluate the probability R of successful or prompt completion of the project. As shown in Eq. (3), the system success is given by the probability- ready expression

$$S_{PRE} = X_B X_F (X_E \vee \overline{X}_E X_D), \quad (17)$$

which corresponds to a value of R given by (3a) conveniently rewritten here as

$$R = p_B \, p_F \, (p_E + q_E p_D), \quad (18)$$

where $p_i = 1 - q_i = E\{X_i\}$ is the probability of completing activity i on time.

Now, we describe a more general technique to compute S_{PRE} (and hence R) for any vector-threshold system. The success of such a system is denoted by S(n; **X**; **W**; **T**) where **X** and **T** are vectors of component successes and thresholds of lengths m and n, respectively, while **W** is an m x n matrix of weights. The success satisfies the following recursive relation, which is valid for n > 0

$$S(n; \mathbf{X}; \mathbf{W}; \mathbf{T}) = \overline{X}_i \, S(n-1; \mathbf{X}/X_i; \mathbf{W}/\mathbf{W_i}; \mathbf{T}) \ \vee \ X_i \, S(n-1; \mathbf{X}/X_i; \mathbf{W}/\mathbf{W_i}; \mathbf{T} - \mathbf{W_i}). \quad (19)$$

This recursive relation is simply an expression of the Boole-Shannon expansion [16] and is a straightforward extension of the recursion used for (scalar-) threshold systems [12,13]. Here $\mathbf{W_i}$ denotes the i^{th} column of the matrix **W** which consists solely of (scalar) W_i and 0 elements, $\mathbf{X}/X_i$ is a vector of length (n–1) obtained by deleting the i^{th} element of **X**, while $\mathbf{W}/\mathbf{W_i}$ is an m x (n–1) matrix obtained by deleting the ith column $\mathbf{W_i}$ of the matrix **W**. The recursive relation (19) must be augmented by the non-recursive boundary conditions

$$S = (0; ; ; \mathbf{T}) = I\,\{\mathbf{0} \geq \mathbf{T}\} = \begin{cases} 1 & \text{if} \quad 0 \geq T_j \quad \text{for all j} \quad (20a) \\ 0 & \text{if} \quad 0 < T_j \quad \text{for any j} \quad (20b) \end{cases}$$

Here, the bold symbol **0** is a vector of m components, each of which is 0, and the notation **0** ≥ **T** means that **0** is greater than or equal to **T** componentwise. Its indicator variable, I(**0** ≥ **T**) is 1 unless some component of **T** is strictly greater than 0. For the current problem of project management the vector-threshold system is coherent and possesses nonnegative weights, and hence it is more convenient to let it have the boundary conditions.

$$S(n; \mathbf{X}; \mathbf{W}; \mathbf{T}) = 1 \quad \text{if } \mathbf{0} \geq \mathbf{T} \quad (21a)$$

$$S(n; \mathbf{X}; \mathbf{W}; \mathbf{T}) = 0 \quad \text{if } \left(\textstyle\sum_{i=1}^{n} \mathbf{W_i} \right) < \mathbf{T} \quad (21b)$$

Equations (21) might allow an early termination of the recursion while n is still strictly greater than 0. If none of the conditions (21) is true, recursion is performed *via* (19).

Repeated application of the recursion relations (19) together with the boundary conditions (20) or (21) yields a probability-ready expression for S. This fact is demonstrated by Fig. 6 which displays a signal-flow-graph representation of (19)–(21) for our current problem. Here, a shaded circle is a node characterized by a matrix **W** (indexing its vertical coordinate) and a vector **T**(indexing its horizontal coordinate) at which (19) is applicable. By contrast, a square black node is a source node of value 1 as dictated by (20a) or (21a), while a square white node is a source node of value 0 as required by (20b) or (21b). These white nodes of 0 values might be omitted, but they are retained to clearly bound the region of validity of the recursion (19). Fig. 6 uses the order (F, B, E, D, A) for the pertinent components which is called the best order [13] since it handles a component of a larger weight earlier than ones of smaller weights. If we apply Mason gain formula [32] to the signal flow graph of Fig. 6, we immediately recover the expression in (17) for S_{PRE}. For comparison, we use Fig. 7 to replicate the work of Fig. 6 using the reverse alphabetical order (F, E, D, B, A). The resulting graph is more involved and yields the expression

$$S_{PRE} = (X_B X_D \vee X_B \overline{X}_D) X_E X_F \vee X_B\, X_D \overline{X}_E\, X_F. \tag{22}$$

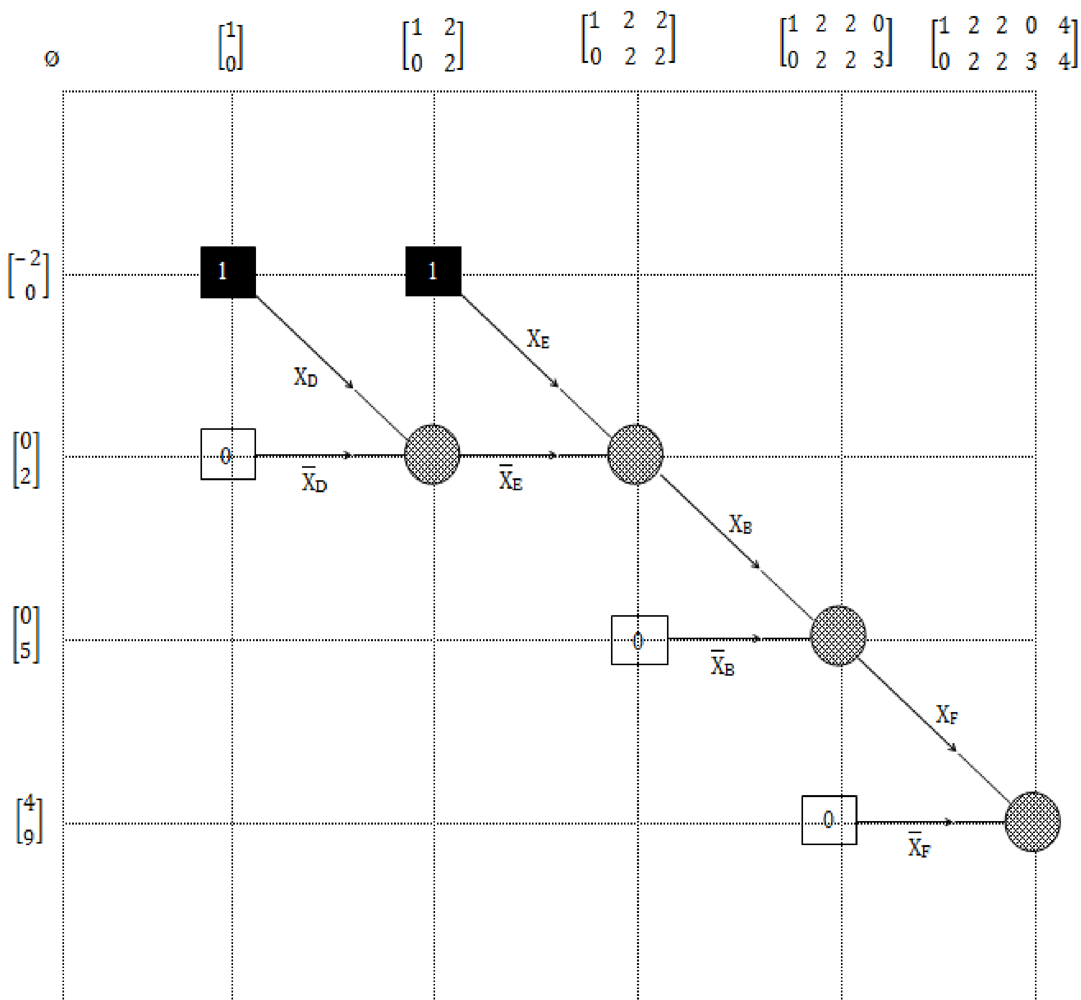


Fig. 6. A signal flow graph representing a recursive expression of the vector-threshold function S (Best ordering (F, B, E, D, A) used)

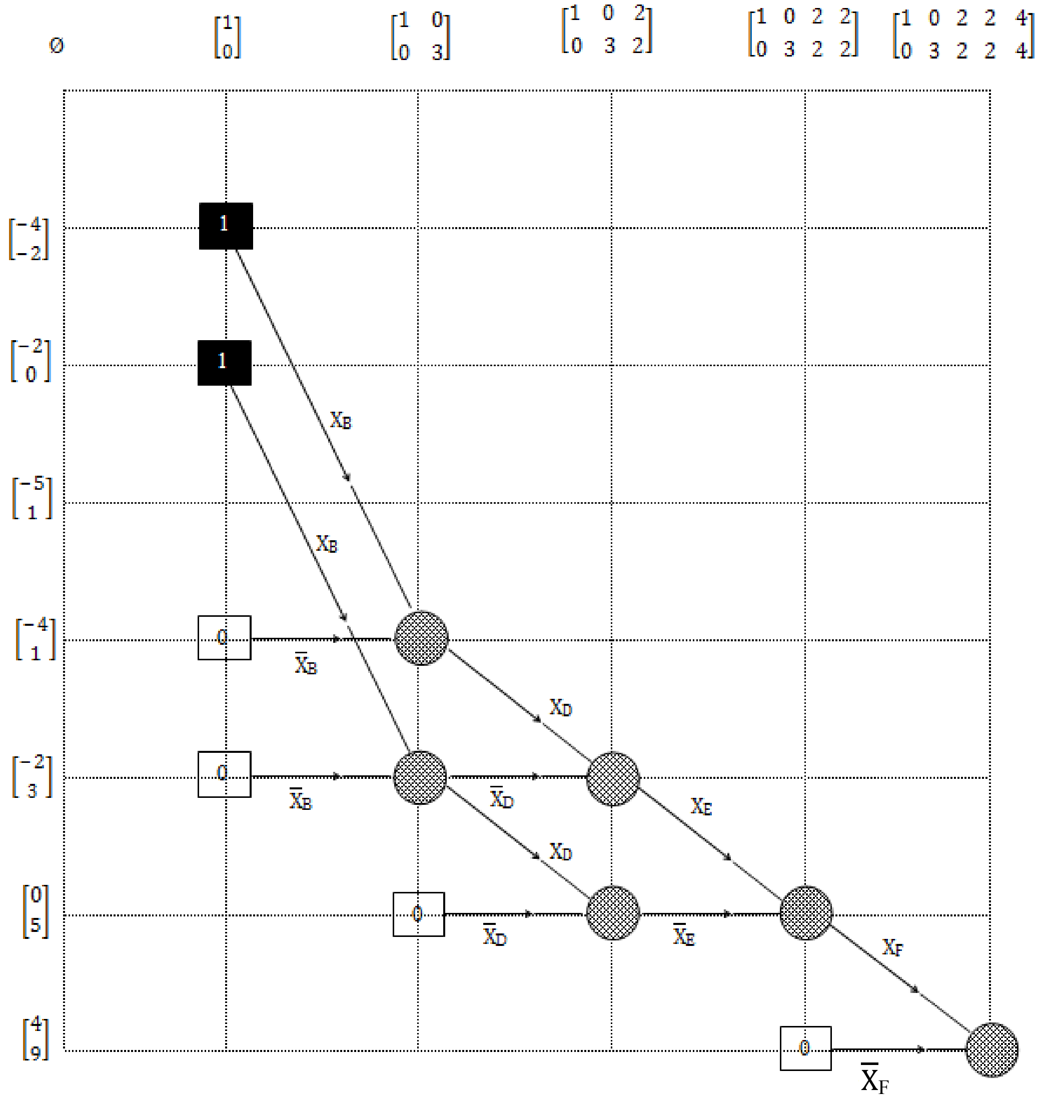


Fig. 7. A signal flow graph representing a recursive expression of the vector-threshold function S (following a reverse alphabetical ordering (F, E, D, B, A))

Equation (22) is a correct PRE expression, albeit not as compact as (17). It produces the reliability expression

$$R = (p_B p_D + p_B q_D)\, p_E\, p_F + p_B p_D q_E p_F. \tag{23}$$

Fig. 8 shows the case of the worst ordering, and yields the result

$$S_{PRE} = X_F\, X_B\, X_D\, X_E\, X_A \vee X_F\, X_B\, X_D\, X_E\, \bar{X}_A \vee X_F\, X_B\, (\bar{X}_D\, X_E \vee X_D\, \bar{X}_E)\, X_A$$

$$\vee X_F\, X_B\, (\bar{X}_D\, X_E \vee X_D\, \bar{X}_E)\, \bar{X}_A \tag{24}$$

which is computationally intensive, indeed (though obviously reducible to (17)). It produces the reliability expression

$$R = p_F p_B p_D p_E p_A + p_F p_B p_D p_E q_A + p_F p_B (q_D p_E + p_D q_E) p_A + p_F p_B (q_D p_E + p_D q_E) q_A. \quad (25)$$

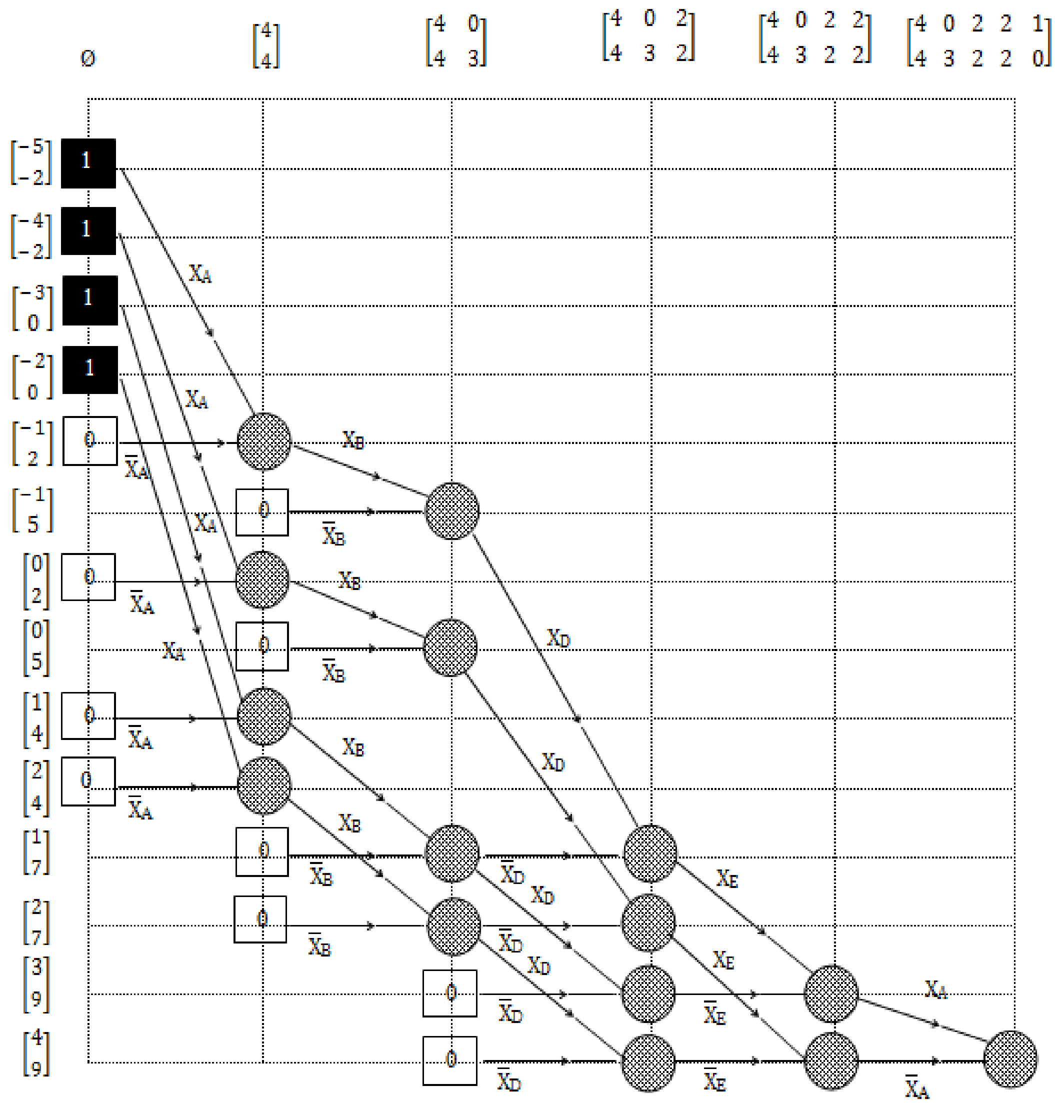


Fig. 8. A signal flow graph representing a recursive expression of the vector-threshold function S (worst ordering (A, E, D, B, F) used)

In passing, we emphasize that the signal flow graphs in Figs. 6-8 have nice interpretations as Reduced Ordered Binary Decision Diagrams [2,11,31,33-37].

5 Conclusions

This paper formulates the problem of project scheduling under uncertainty as a problem of a vector-weighted voting system. It offers two distinct methods (borrowed from the field of system reliability) for handling the problem. The first method uses Karnaugh maps to represent pseudo-Boolean functions and to derive a probability-ready expression for the indicator of prompt completion of the project. The second method uses an ROBDD-like signal flow graph to represent the underlying recursive relations and boundary conditions. Efficiency of computation is very sensitive to the order of traversing the variables in this method. The two methods obtain the same answer, both in a more intuitive and straightforward way than obtained earlier in the literature. The first method might be conveniently be used for evaluating not just a single value of the probability R of prompt project completion, but for computing R repeatedly for varying values of the maximum allowed project duration D_m. Knowledge of R as a function of D_m constitutes invaluable information for both contractor and contractee when negotiating budget and associated penalties for delays or incentives for promptness or early accomplishment.

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Winston WL. Operations research: Applications and Algorithms. Duxbury Press, Boston, MA, USA; 1994.

[2] Rushdi AM, Rushdi MA. Switching-algebraic analysis of system reliability. Chapter 6 in M. Ram and P. Davim (Editors), Advances in Reliability and System Engineering, Management and Industrial Engineering Series, Springer International Publishing, Cham, Switzerland. 2016;139-161.

[3] Taylor A, Zwicker W. A characterization of weighted voting. Proceedings of the American Mathematical Society. 1992;115(4):1089-1094.

[4] Taylor A, Zwicker W. Weighted voting, multicameral representation, and power. Games and Economic Behavior.1993;5(1):170-181.

[5] Taylor AD, Pacelli AM. Mathematics and politics: Strategy, voting, power, and proof, second edition, Springer Science+Business Media, New York, NY, USA; 2008.

[6] Elkind E, Goldberg LA, Goldberg PW, Wooldridge M. On the computational complexity of weighted voting games. Annals of Mathematics and Artificial Intelligence. 2009;56(2):109-131.

[7] Freixas J, Marciniak D. On the notion of dimension and codimension of simple games. Contributions to Game Theory and Management. 2010;3:67-81.

[8] Alturki AM, Rushdi AM. Weighted voting systems: A threshold-Boolean perspective. Journal of Engineering Research. 2016;4(1):125-143.

[9] Eryilmaz S. Capacity loss and residual capacity in weighted k-out-of-n: G systems. Reliability Engineering and Systems Safety.2015;136:140-144.

[10] Chen Y, Yang Q. Reliability of two-stage weighted-k-out-of-n systems with components in common. IEEE Transactions on Reliability. 2005;54(3):431-440.

[11] Rushdi AMA, Bjaili HA. An ROBDD algorithm for the reliability of double-threshold systems. British Journal of Mathematics and Computer Science. 2016;19(6):1-17.

[12] Rushdi AM. Threshold systems and their reliability. Microelectronics and Reliability. 1990;30(2): 299-312.

[13] Rushdi AM, Alturki AM. Reliability of coherent threshold systems. Journal of Applied Science. 2015;15(3):431-443.

[14] Kuo W, Zuo MJ. Optimal reliability modeling: Principles and applications. Wiley, New York, NY, USA; 2003.

[15] Rushdi AM. Symbolic reliability analysis with the aid of variable-entered Karnaugh maps. IEEE Transactions on Reliability. 1983;R-32(2):134-139.

[16] Rushdi AM, Goda AS. Symbolic reliability analysis via Shannon's expansion and statistical independence, Microelectronics and Reliability. 1985;25(6):1041-1053.

[17] Rushdi AM. Utilization of symmetric switching functions in the computation of k-out-of-n system reliability. Microelectronics and Reliability. 1986;26(5):973-987.

[18] Rushdi AM. Reliability of k-out-of-n systems, Chapter 5 in Misra, K. B. (Editor), New Trends in System Reliability Evaluation, Fundamental Studies in Engineering, Elsevier Science Publishers, Amsterdam, The Netherlands. 1993;16:185-227.

[19] Rushdi AM, Abdul Ghani AA. A comparison between reliability analyses based primarily on disjointness or statistical independence: The case of the generalized INDRA network. Microelectronics and Reliability. 1993;33(7):965-978.

[20] Rushdi AMA, Hassan AK. Reliability of migration between habitat patches with heterogeneous ecological corridors. Ecological Modelling. 2015;304:1-10.

[21] Rushdi AMA, Hassan AK. An exposition of system reliability analysis with an ecological perspective. Ecological Indicators. 2016;63:282-295.

[22] Hammer PL, Rudeanu S. Boolean methods in operations research and related areas. Springer Science & Business Media, Berlin-Heidelberg, Germany; 1968.

[23] Rushdi AM. Probabilistic performance indices of power generation and transmission systems. In Proceedings of the First Symposium on Electric Power Systems in Fast Developing Countries, Riyadh, Saudi Arabia.1987;247-254.

[24] Rushdi AM. Capacity function-preserving star-delta transformations in flow networks. Reliability Engineering.1987;19(1):49-58

[25] Rushdi AM. Performance indexes of a telecommunication network. IEEE Transactions on Reliability.1988;R-37(1):57-64.

[26] Rushdi AM. Star-delta transformations of bidirectional branches in probabilistic flow networks. Microelectronics and Reliability. 1990;30(3):525-535.

[27] Rushdi AMA, Ghaleb FAM. The Walsh spectrum and the real transform of a switching function: A review with a Karnaugh-map perspective. Journal of Qassim University: Engineering and Computer Sciences. 2015;7(2):73-112.

[28] Rushdi AM. Development of modified nodal analysis into a pedagogical tool. IEEE Transactions on Education. 1985;E-28(1):17-25.

[29] Rushdi MA, Rushdi AM. On the fundamental masses derivable by dimensional analysis. Journal of King Abdulaziz University: Engineering Sciences. 2016;27(1):35-42.

[30] Rushdi AM, Al-Khateeb DL. A review of methods for system reliability analysis: A Karnaugh-map perspective, Proceedings of the First Saudi Engineering Conference, Jeddah, Saudi Arabia. 1983;1: 57-95.

[31] Rushdi AMA, Alturki AM. Unification of mathematical concepts and algorithms of k-out-of-n system reliability: A perspective of improved disjoint products. Journal of Engineering Research. 2017;5.

[32] Golnarghi F, Kuo BC. Automatic control systems. 9th Edition. Wiley, New York, NY, USA; 2009.

[33] Bryant R. Graph-based algorithms for Boolean function manipulation. IEEE Transactions on Computers. 1986;35(8):677-691.

[34] Rauzy A. A brief introduction to binary decision diagrams. Journal Européen des Systèmes Automatisés.1996;30(8):1033-1050.

[35] Rauzy A. Binary decision diagrams for reliability studies, Chapter 25 in Misra, K. B. (Editor), Handbook of Performability Engineering, Springer, London, UK. 2008;381-396.

[36] Mo Y. A multiple-valued decision-based approach to solve dynamic fault tree. IEEE Transactions on Reliability. 2014;63(1):81-93.

[37] Mo Y, Xing L, Amari SV, Dugan JB. Efficient analysis of multi-state k-out-of-n system. Reliability Engineering and System Safety. 2015;133:95-105.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
http://sciencedomain.org/review-history/18792