ABSTRACT

The computation of probabilities associated with combinations of events may be a laborious task due to a combinatorial complexity. This kind of difficulty can often be reduced by resorting to Boolean algebra and graph theory. The computation of the average number of disjoint available paths is considered in the paper as a particularly illustrative example.

1. INTRODUCTION

In some applications of probability theory, the sample points are combinations of a finite number of simple independent events, which are assigned complementary probabilities of occurrence and non occurrence. The calculation of probabilities associated with aggregates of such sample points is simple in principle but often unpractical due to the great combinatorial complexity involved. In the analysis of communication networks, the links of which are assigned probabilities of being operative or inoperative (idle or busy), this kind of complexity can arise even with networks having a few nodes. The calculation of the probability that at least i paths are available (operative or idle) is a typical example. The calculation of the average number of disjoint available paths may appear an even more cumbersome task; actually it can be reduced to a simpler one if resort is made to Boolean algebra and to some results of graph theory.

2. APPLICATION OF BOOLEAN ALGEBRA

A straightforward application of Boolean algebra to probability combinatorial problems can be founded on a simple correspondence between Boolean functions representing overall events and expressions of the ordinary algebra giving the probabilities of the same events.

Given a set of simple independent events \( e_1, e_2, \ldots, e_n \) we associate with each of them a binary variable that takes value 1 for occurrence and value 0 for non occurrence of the associated event. A sample point is then represented by one of the \( 2^n \) minterms:

\[
x_1 x_2 \cdots x_n
\]

\[
x_1 x_2 \cdots x_n
\]

\[
\cdots
\]

\[
x_1 x_2 \cdots x_n
\]

and an aggregate of sample points by a Boolean function \( X = X(x_1, x_2, \ldots, x_n) \).

The function \( X \) can be put in many forms, in particular, by applying well known theorems of Boolean algebra, \( X \) can be written in the form:

\[
X = \bigcup X_i
\]

where the \( X_i \) are minterms or more generally mutually exclusive functions, each written in such a way that

- every variable appears at most once
- intersections and complementations only are applied.

When the function \( X \) has this form, we can write

\[
Pr(X=1) = Pr \left( \bigcup_{i} X_i = 1 \right) = \sum_{i} Pr(X_i = 1)
\]

for the \( X_i \) represent mutually exclusive events.

Moreover, as the variables \( x_i \) represent by assumption independent events if we set

\[
Pr(x_i = 1) = Pr(x_i = 1) = Pr(x_j = 1)
\]

for any two variables \( x_i, x_j \) we have

\[
Pr(\bar{x_i} \bar{x_j} = 1) = Pr(\bar{x_i} \bar{x_j} = 1) = Pr(\bar{x_i} = 1)Pr(\bar{x_j} = 1)
\]

It is then possible to switch from a Boolean function \( X \), representing in the form (1) an overall event, to an ordinary expression of \( Pr(X=1) \) by substituting as follows:

- \( p_i = Pr(e_i) = Pr(x_i = 1) \rightarrow x_i \)
- \( \bar{p_i} = 1 - p_i \rightarrow \bar{x_i} \)
- arithmetic product \( \rightarrow \) intersection
- arithmetic sum \( \rightarrow \) union
- complementation

(In the sequel, when no confusion arises, the same symbols will still be used to denote arithmetic and logic product (sum).)

EXAMPLE

Fig. 1 shows a simple series-parallel network with four paths between nodes A and B. Let \( x_{ij} \) be the event "link \( ij \) is operative" and \( a_{ij} \) the probability of its occurrence. To calculate the probability of finding at least one operative path we write:

\[
Pr \left( x_{11} x_{21} + x_{11} x_{22} + x_{12} x_{21} + x_{12} x_{22} = 1 \right) =
\]

\[
= Pr \left[ \left( (x_{11} + x_{12}) (x_{21} + x_{22}) = 1 \right) \right] =
\]

\[
= Pr \left( \bar{x}_{11} \bar{x}_{21} \bar{x}_{22} = 1 \right)
\]

Fig. 1
As the function in parentheses is in the form (1) we substitute according to (5) and get the wanted probability

\[ \tilde{a}_{11} \tilde{a}_{12} \tilde{a}_{21} \tilde{a}_{22} \]

3. CALCULATION OF \( \bar{N} \)

The particular problem considered in this paper is the calculation of the average number of the disjoint paths available between two nodes of a communication network, the links of which are assigned independent probabilities \( p_1, p_2, \ldots \) of being operative and probabilities \( \overline{p}_1, \overline{p}_2, \ldots \) of being inoperative. In other words, given the m possible paths between nodes A and B of a network, we want to calculate the expectation:

\[ \bar{N} = \frac{1}{m} \sum_{i=1}^{m} Pr(\{ E \}_{m} = s) = \frac{1}{m} \sum_{i=1}^{m} Pr(\{ E \}_{m} \geq s) \]  

where

\[ [E]_{m} \]  

is maximum number of operative disjoint paths that can be established between nodes A and B

When the paths between A and B are all mutually disjoint there is no calculation problem, as for that case we have the well known result:

\[ \bar{N} = \frac{1}{m} \sum_{i=1}^{m} Pr(E_i) \]  

where \( E_i \) is the event "path i is operative". In the sequel we shall assume that at least some of the paths are not disjoint.

It is then expedient to start from the following expression of \( \bar{N} \):

\[ \bar{N} = \frac{1}{m} \sum_{i=1}^{m} Pr(\{ E \}_{m} \geq s-1, \{ E \}_i = s) \]  

where

\[ [E] = s \text{ stands for } \{ E \}_{1}, \ldots, \{ E \}_m \text{ occur and the maximum number of disjoint operative paths that can be found among paths } 1,2,\ldots, \text{ is equal to } s \]

The calculation of \( \bar{N} \) according to (8) is illustrated by the following example.

EXAMPLE

Among the four paths of the network of Fig. 1 there are two couples of disjoint paths: 1, 11, 21, 12, 22 and 1, 11, 12, 11, 12, 21. Let \( E_1, E_2, E_3, E_4 \) be the event "path 1, 21, 11, 12, 21" is operative". We have:

\[ Pr(\{ E \}_1 = 0, \{ E \}_1 = 1) = Pr(E_1) \]
\[ Pr(\{ E \}_1 = 0, \{ E \}_2 = 1) = Pr(E_1 E_2) \]
\[ \ldots \]
\[ Pr(\{ E \}_2 = 1, \{ E \}_3 = 2) = 0 \]
\[ Pr(\{ E \}_3 = 1, \{ E \}_4 = 2) = Pr(E_1 E_2 E_3 E_4) \]
\[ \ldots \]

By rearranging the terms of (8) we get:

\[ \bar{N} = \frac{1}{m} \sum_{i=1}^{m} \sum_{r=1}^{m} Pr(\{ E \}_{m} \geq r-1, \{ E \}_i = r) \]  

In order to calculate \( \bar{N} \) according to (9), for given \( l \) and \( r \) we have to inspect the following combinations of events

\[ E_1 E_2 \ldots E_{l-1} E_l \]
\[ E_1 E_2 \ldots E_{l-1} E_l \]
\[ E_1 E_2 \ldots E_{l-1} E_l \]

and single out those implying a maximum number of disjoint operative paths equal to \( r \) when \( E_i \) is counted and to \( r-1 \) when \( E_i \) is not counted. If we denote with \( E_i^{r}\) the union of the combinations singled out according to the above criterium for given values of \( i \) and \( r \), we have:

\[ \bar{N} = \frac{1}{m} \sum_{i=1}^{m} Pr(\bigcup_{r=1}^{m} E_i^{r}) \]  

An inspection of all the combinations (10) in order to single out the wanted ones is not necessary; in fact it is possible to write compact expressions equivalent to the unions of the \( E_i^{r} \) by resorting to graph theory.

In order to do so, we refer to a graph \( G(V_i, A) \) whose vertices \( v_1, v_2, \ldots, v_i \) correspond to the given events \( E_1, E_2, \ldots, E_i \) and whose edges join vertices corresponding to dependent events. Finding a wanted combination of the given events is then the same as finding a subgraph of \( G(V_i, A) \) whose vertex \( v_i \) is "\( a \)-critical", i.e. whose vertex \( v_i \) belongs to all largest sets of non adjacent vertices of the subgraph (see Appendix). If the variables \( v_1, v_2, \ldots, v_i \) for the presence and the variables \( \overline{v}_1, \overline{v}_2, \ldots \) for the absence of vertices are used, we can express a complete set of conditions for the vertex \( v_i \) to be \( a \)-critical with a Boolean function \( V_i (v_1, \overline{v}_2, \ldots, v_i) \). When the function \( V_i \) has been determined, after setting

\[ Pr(V_i = 1) = Pr(E_i) \]

we can write

\[ Pr(\bigcup_{i=1}^{m} E_i^{r}) = Pr(V_i = 1) \]  

The Appendix deals with the problem of determining the functions \( V_i \). The following example illustrates the procedure with reference to a simple case in which the \( V_i \) can be determined by inspecting the graphs of reference.

EXAMPLE

To calculate \( \bar{N} \) for the network of Fig. 1 we have to consider events \( E_1, E_2, E_3, E_4 \). The graphs of reference are shown in Fig. 2.
The functions \( V_j \) are:

\[
\begin{align*}
V_1 &= v_1 \\
V_2 &= v_2 \\
V_3 &= v_1 v_2 v_3 \\
V_4 &= v_1 v_2 v_4 + v_1 v_2 v_3 v_4 + v_1 v_2 v_3 v_4
\end{align*}
\]

3. REMARKS

When the number of possible paths is substantial the procedure described in the paper may be impractical without the use of a computer. For certain applications, however, the approximation obtainable by neglecting certain terms may be acceptable. As an example, we may truncate at a fixed length the \( \sigma \)-sequences (see Appendix) constructed in order to determine the functions \( V_j \). Obviously the best approximate procedure depends on the values of the involved elementary probabilities.

In some reliability problems instead of paths we may have "modes of operation". (The realization of a mode depends on the occurrence or non-occurrence of certain simple independent events and the realization of any two modes may be a possible or an impossible event.) The procedure described above can still be applied when a list of modes of operation is given and it is required to calculate the average number of modes simultaneously realizable.

Whether we have to deal with paths or modes of operation, the definition of \( \mathcal{N} \) implies the assumption of a "rearrangeable system" where in any situation the best use is made of available links or parts, i.e., where the maximum number of paths or modes is always established. If we assume instead that the available links or parts are used so as to satisfy a given priority in the establishment of the simultaneously realizable paths or modes, the described procedure, with minor changes, can also be applied to the expectation thus defined. The changes consist in truncating the \( \sigma \)-sequences when a vertex \( a_i(b_j) \) corresponds to a path or mode that has no priority over the paths or modes corresponding to the preceding vertices \( b(a) \).

REFERENCES


/2/ Le Gall P., "Méthode de calcul de l'encoulement dans les systèmes téléphoniques à marques", Ann. Télécom, 12, 1957


APPENDIX

For a given simple graph \( G(V, A) \) we introduce the following definitions:

- a "stable" set is a set \( S \subseteq V \) that does not contain adjacent vertices, i.e., such that \( \mathcal{L}(S) \cap S = \emptyset \)

- a vertex \( v \in V \) that is included in every maximum stable set is said to be "\( \alpha \)-critical"

- a vertex \( v \in V \) that is included in no maximum stable set is said to be "\( \beta \)-critical"

- a "\( \sigma \)-critical" sequence is a sequence of distinct vertices of \( G \)

\[
\mathcal{L}(v) = \{a_1, b_1, a_2, b_2, \ldots\}
\]

that verifies:

1) \( \mathcal{L}(v) \cap \{a_1, a_2, \ldots, a_i\} \neq \emptyset \)

2) \( \mathcal{L}(v) \cap \{b_1, b_2, \ldots, b_i\} \neq \emptyset \)

3) no vertex satisfying 1) or 2) can be added to the sequence.

THEOREM

A vertex \( v \) of \( G(V, A) \) is \( \alpha \)-critical if and only if there is an uneven critical sequence

\[
\mathcal{L}(v) = \{a_1, b_1, a_2, b_2, \ldots, b_{n-1}, a_n\}
\]

with \( v = b_1 \)

1) If there is an uneven critical sequence with \( v = b_1 \), for every stable set \( S \) such that

\[
\{b_1, b_2, \ldots, b_j\} \subseteq S \quad 1 \leq j \leq n-2
\]

\[
b_{j+1} \notin S
\]

there is a set

\[
S' = (S - \{b_1, b_2, \ldots, b_j\}) \cup \{a_1, a_2, \ldots, a_{j+1}\}
\]

that is stable as

\[
y \in \{a_1, a_2, \ldots, a_{j+1}\} \Rightarrow y \cap (S - \{b_1, b_2, \ldots, b_j\}) = \emptyset
\]

By the same token for every stable set

\[
S = \{b_1, b_2, \ldots, b_{n-1}\}
\]

there is a stable set

\[
S' = (S - \{b_1, b_2, \ldots, b_{n-1}\}) \cup \{a_1, a_2, \ldots, a_n\}
\]

As in all cases \( |S'| = |S| + 1, v \) is \( \beta \)-critical.

2) If \( v \) is \( \beta \)-critical and \( S_{\text{max}} \) a maximum stable set there is no even critical sequence

\[
a_1, b_1, \ldots, a_n, b_n
\]

with \( b_1 = v \)

\[
\mathcal{L}(v) = \{a_1, a_2, \ldots, a_n\} \subseteq S_{\text{max}}
\]

\[
\{b_1, b_2, \ldots, b_n\} \subseteq C - S_{\text{max}}
\]

and otherwise there would be a maximum stable set

\[
S'_{\text{max}} = (S_{\text{max}} - \{a_1, a_2, \ldots, a_n\}) \cup \{b_1, b_2, \ldots, b_n\}
\]

including \( v \). As \( v \) has at least one neighbour \( v \in S_{\text{max}} \) there is at least one uneven critical sequence with

\[
b_1 = v \quad \text{and} \quad a_i \in \mathcal{L}(v),
\]
COROLLARY

A vertex \( v \) is \( \alpha \)-critical if \( \Gamma_G(v) = \emptyset \) or if \( \forall \in \Gamma_G(v) \Rightarrow v \) is \( \beta \)-critical.

The above results permit to solve the problem of determining all the subgraphs of \( G(V,A) \) generated by subsets of \( V \) including an \( \alpha \)-critical vertex \( v \). In order to do so, we find all the sequences \( \sigma = a_1, b_1, a_2, b_2, \ldots \) we can construct according to the following rules:

1. \( a_1 = v \)
2. \( b_i \in \Gamma_G(a_i) \)
   \[ b_i \cap \{ b_1, b_2, \ldots, b_{i-1} \} = \emptyset \quad i > 1 \]
3. \( a_{i+1} \in \Gamma_G(b_i) \)
   \[ a_{i+1} \cap \{ a_1, a_2, \ldots, a_i \} = \emptyset \quad i > 1 \]

When just one of these sequences can be constructed, no critical even sequence with \( b_1 = v \) exists if one of the following conditions is verified:

\[ a_1 \not\in \{ a_2, b_2 \} \]
\[ a_1 \not\in \{ a_2, b_2 \} \]
\[ \ldots \]
\[ a_1 \not\in \{ a_2, b_2 \} \]
\[ (or \quad a_1 \not\in \{ a_2, b_2 \}) \]

It is also easily seen (by referring to the definition of critical sequence) that:

- when we have any two sequences
  \[ \sigma_1 = a_{11}, b_{11}, a_{12}, \ldots \]
  \[ \sigma_2 = a_{21}, b_{21}, a_{22}, \ldots \]
  with \( a_{11} = a_{21}, b_{11} = b_{21}, \ldots, a_{1i} = a_{2i} \) but \( b_{1j} \neq b_{2j} \), conditions for \( \sigma_1 \) and \( \sigma_2 \) must both be verified.

- when we have any two sequences with
  \[ a_{11} = a_{21}, b_{11} = b_{21}, \ldots, b_{1i} = b_{2i} \] but
  \[ a_{1(i+1)} \neq a_{2(i+1)} \]
  conditions for \( \sigma_1 \) or \( \sigma_2 \) must be verified.

By combining the conditions for the single sequences according to the above criterion, we get the Boolean functions the minterms of which denote the wanted subgraphs.
ERRATA

Paper No 317, G. Timperi

page 3, column 2, THEOREM,

first line
"A vertex v of..." should read "A vertex of a tree..."

point 1), fourth line
"b_{i+1}" should read "b_{i+j}"

point 2, first line
"S_{max}" should read "S_{max}"