Volume XXI 2018
ISSUE no.1
MBNA Publishing House Constanta 2018

Scientific Bulletin of Naval Academy

SBNA PAPER • OPEN ACCESS

Automatically determines the length of the dual system to a bivalent system

To cite this article: Paul Vasiliu, Scientific Bulletin of Naval Academy, Vol. XXI 2018, pg. 50-57.

Available online at www.anmb.ro

ISSN: 2392-8956; ISSN-L: 1454-864X

doi: 10.21279/1454-864X-18-I1-007

Automatically determines the length of the dual system to a bivalent system

Paul Vasiliu[1]

[1]**Naval Academy "Mircea cel Batran" Constantza, Romania**

Abstract. From the point of view of reliability theory, a system can have two stable states: functioning and defect (bivalent system). Any system is a set of elements. Each element in this set can be found in one of the following states: operating state and fault condition. A subset of elements in the running state is called a system link if they only ensure the system works. The length of a bivalent system is equal to the minimum number of elements that the system holds. In this paper we present an algorithm for automatic determination of dual system length to a bivalent system, a Matlab script, a case study and subsequent development directions.

Keywords: dual, reliability, script, structure, system

1. Introduction

The plurality of all elements of the assembly of elements that reflect how the state of an S system depends on the states of its components is called a structure.

In the reliability analysis, structures play an important role. In a first step, the structure of the equipment is analyzed, followed by establishing the algebraic expression of the structure function and then constructing a reliability network associated with that equipment.

Considerations regarding complex systems or equipment are based on the following assumptions: The equipment considered can only be in one of the following states: operating state or fault condition (bivalent system). The considered equipment can be decomposed into n components (elements) e_i, $i = 1,2,\cdots,n$. The set of all these elements is: $E = \{e_1, e_2, \cdots, e_n\}$.

Each component e_i, $i = 1,2,\cdots,n$, is associated with a state variable x_i defined by: $x_i = \begin{cases} 1 & \text{if } \ e_i \text{ is in operation} \\ 0 & \text{if } \ e_i \text{ does not work} \end{cases}$.

The set of states $X = \{x_1, x_2, \cdots, x_n\}$ characterizes the set of possible states of the assembly of elements. Obvious, $card(X) = 2^n$.

The system S is associated with a state variable y defined as follows: $y = \begin{cases} 1 & \text{if } S \text{ is in operation} \\ 0 & \text{if } S \text{ does not work} \end{cases}$.

Let $Y = \{y\} = \{0,1\}$. The y state variable depends on the set of X states.

We can define a function $\varphi: X \to Y$, defined by: $y = \varphi(x_1, x_2, \cdots, x_n)$. The function φ is called the function of the structure. Since the function φ depends on n independent variables, it is called function of the n -order structure. The S system can be identified with the pair (E, φ): $S = (E, \varphi)$.

Let the system $S = (E, \varphi)$. The subassembly $L = \{e_i \mid i \in I \subset \{1,2, \cdots, n\}\} \subset E$ is called a link if for any $x_i = 1$, $i \in I$ and for any $x_i = 0$, $i \notin I$ follows: $y = \varphi(x_1, x_2, \cdots, x_n) = 1$.
In other words, a connection of an operating system ($y = 1$) is a subset (subassembly) of elements in which all elements are in the running state, the rest of the elements being defective.

By reformulating, a link is a subset of elements that are in a state of operation and which ensure the system's operation if the other elements are defective.

Any system has at least one link.

To determine all system links, calculate the values of the structure function at all points $(x_1, x_2, \cdots, x_n)$. If $f(x_1, x_2, \cdots, x_n) = 1$ for $x_{i_1} = x_{i_2} = \cdots = x_{i_p} = 1$ and $x_i = 0$ for any $i \in \{1,2, \cdots, n\} \setminus \{i_1, i_2, \cdots, i_p\}$ then the subset of elements (subassembly of elements) $\{e_{i_1}, e_{i_2}, \cdots, e_{i_p}\}$ is a link.

Let $l_k > 0$ the number of links with k elements, $k \in N^*$. It is called the length of the system S number $\lambda(S) = min\{k \mid l_k > 0\}$.

Let $n \in N^*$, $E = \{e_1, e_2, \cdots, e_n\}$, $X = \{x_1, x_2, \cdots, x_n\}$, function $\varphi: X \to \{0,1\}$, $y = \varphi(x_1, x_2, \cdots, x_n)$, and the system $S = (E, \varphi)$.

It's called the dual system of the system $S = (E, \varphi)$ the system $S^d = (E, \varphi^d)$ where the structure function $\varphi^d: X \to \{0,1\}$ is defined by $\varphi^d(x_1, x_2, \cdots, x_n) = 1 - \varphi(1 - x_1, 1 - x_2, \cdots, 1 - x_n)$.

Through duality, series connections become parallel connections and parallel connections become serial connections.

The study of the links and length of a system brings important information about the operation of the system under the condition that some of the component elements are defective.

The difficulty of manually determining links and the length of a system increases as the number of elements in the system increases. There is a need to simplify and automate the determination of system links and system length. For this reason, the authors wrote the Matlab functions presented in Section 2.

2. Matlab implementation

The Matlab main function with the signature *function links(n)* receives the n-value of the input. The function generates all links to the reliability network. The function calculates the length of the dual system.

The function *function w=conversion(n,v)* converts the value of the input argument n from base 10 to base 2 and stores the result of the conversion in the variable w.

The function *function val=fstruct(x)* receives the input vector x of the structure function variables and returns the value of the structure function.

The function function val = fdstruct (x) receives the vector x and returns the value of the structure function of the dual system.

The display of links is made by the function *function write(n,w,tip)*.

The function *function w=inverse(v)* inverts the vector v and stores the result in the vector w.

The function *function printnl(nb)* displays the number of links with k elements, $k = 1,2, \cdots$

We present below the Matlab code for implementing the algorithm.

```
% The main function
function links(n)
clc;
k=2^n;
for i=1:n
v(i)=0;
end
```

```
ival=1;
for i=0:k-1
w=conversion(i,v);
val=fdstruct(w);
if val==1
vval(ival)=i;
fprintf(' The value %d :\n y = f ( ',i);
nb(ival)=0;
for j=1:length(w)-1
fprintf(' %d , ',w(j));
if w(j)==1
nb(ival)=nb(ival)+1;
end
end
if w(length(w))==1
nb(ival)=nb(ival)+1;
end
fprintf(' %d ',w(length(w)));
fprintf(' ) = %d \n',val);
write(i,w,'l');
ival=ival+1;
end
end
printnl(nb);
fprintf(' The length of the dual system is %d \n',min(nb));
end
% Conversion function
% from base 10 in base 2
function w=conversion(n,v)
i=1;
while n ~= 0
uc=mod(n,2);
v(i)=uc;
i=i+1;
n=floor(n/2);
end
w=inverse(v);
end
% Function for reversing vector v
function w=inverse(v)
n=length(v);
for i=1:n
w(i)=v(n-i+1);
end
end
% Function for displaying links
function write(n,w,tip)
p=length(w);
if tip=='l'
fprintf(' Subassembly L = { ');
for i=1:p
```

```
if w(i)==1
fprintf(' e%d ',i);
end
end
fprintf(' } is a link \n');
end
end
% Function to display the number of links
% with k elements
% k=1,2,...
function printnl(nb)
kmin=min(nb);
kmax=max(nb);
for i=kmin:1:kmax
s=0;
for j=1:length(nb)
if nb(j)==i & nb(j)~=0
s=s+1;
end
end
fprintf(' Number of links with %d elements = %d \n',i,s);
end
end
% Function of the dual system structure
function val=fdstruct(x)
y=1-x;
val=1-fstruct(y);
end
% Function of system structure
function val=fstruct(x)
val=1-(1-x(1)*x(2)*x(3))*(1-x(4)*x(5));
end
```

3. Case study

Consider the system $S=(E,\varphi)$ with the properties: $n=5$, $E=\{e_1,e_2,e_3,e_4,e_5\}$, $X=\{x_1,x_2,x_3,x_4,x_5\}$. The graph of the system is shown in figure 1. The graph of the dual system is shown in figure 2.

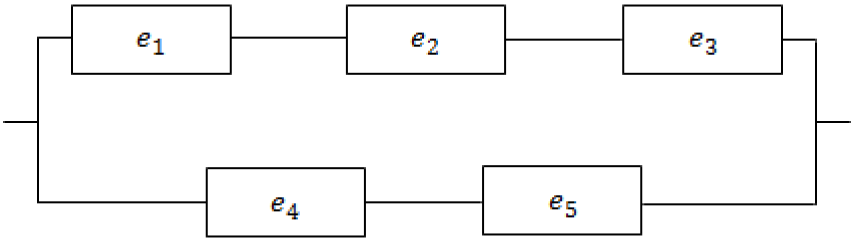

Figure 1: The system $S=(E,\varphi)$.

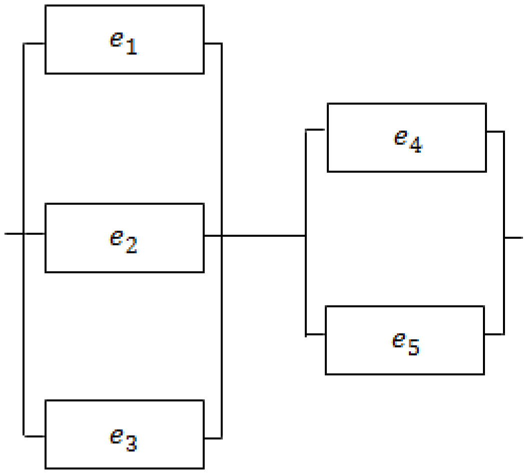

Figure 2: The dual system $S^d = (E, \varphi^d)$.

Obviously, the structure function of the system is the function: $\varphi: X \to \{0,1\}$, $\varphi(x_1, x_2, x_3, x_4, x_5) = 1 - (1 - x_1 \cdot x_2 \cdot x_3) \cdot (1 - x_4 \cdot x_5)$.

The function of the dual system structure is the function:
$\varphi^d: X \to \{0,1\}$ defined by:
$\varphi^d(x_1, x_2, x_3, x_4, x_5) = 1 - \varphi(1 - x_1, 1 - x_2, 1 - x_3, 1 - x_4, 1 - x_5)$,
$\varphi^d(x_1, x_2, x_3, x_4, x_5) = \big(1 - (1 - x_1) \cdot (1 - x_2) \cdot (1 - x_3)\big) \cdot \big(1 - (1 - x_4) \cdot (1 - x_5)\big)$.

For manual determination of all system links, the values of the structure function are calculated at all points $(x_1, x_2, x_3, x_4, x_5)$. If for $x_{i_1} = x_{i_2} = \cdots = x_{i_p} = 1$ and $x_i = 0$ for any $i \in \{1,2,\cdots,n\} \setminus \{i_1, i_2, \cdots, i_p\}$ and $\varphi(x_1, x_2, \cdots, x_n) = 1$ then the subassembly of elements $\{e_{i_1}, e_{i_2}, \cdots, e_{i_p}\}$ is a link.

Table 1 is completed and all system links are obtained.

Table 1. Dual system links $S^d = (E, \varphi^d)$.

Value	x_1	x_2	x_3	x_4	x_5	φ	φ^d	Links in $S^d = (E, \varphi^d)$
0	0	0	0	0	0	1	0	
1	0	0	0	0	1	1	0	
2	0	0	0	1	0	1	0	
3	0	0	0	1	1	1	0	
4	0	0	1	0	0	1	0	
5	0	0	1	0	1	0	1	$\{e_3, e_5\}$
6	0	0	1	1	0	0	1	$\{e_3, e_4\}$
7	0	0	1	1	1	0	1	$\{e_3, e_4, e_5\}$
8	0	1	0	0	0	1	0	
9	0	1	0	0	1	0	1	$\{e_2, e_5\}$
10	0	1	0	1	0	0	1	$\{e_2, e_4\}$
11	0	1	0	1	1	0	1	$\{e_2, e_4, e_5\}$
12	0	1	1	0	0	1	0	
13	0	1	1	0	1	0	1	$\{e_2, e_3, e_5\}$
14	0	1	1	1	0	0	1	$\{e_2, e_3, e_4\}$

15	0	1	1	1	1	0	1	$\{e_2, e_3, e_4, e_5\}$
16	1	0	0	0	0	1	0	
17	1	0	0	0	1	0	1	$\{e_1, e_5\}$
18	1	0	0	1	0	0	1	$\{e_1, e_4\}$
19	1	0	0	1	1	0	1	$\{e_1, e_4, e_5\}$
20	1	0	1	0	0	1	0	
21	1	0	1	0	1	0	1	$\{e_1, e_3, e_5\}$
22	1	0	1	1	0	0	1	$\{e_1, e_3, e_4\}$
23	1	0	1	1	1	0	1	$\{e_1, e_3, e_4, e_5\}$
24	1	1	0	0	0	1	0	
25	1	1	0	0	1	0	1	$\{e_1, e_2, e_5\}$
26	1	1	0	1	0	0	1	$\{e_1, e_2, e_4\}$
27	1	1	0	1	1	0	1	$\{e_1, e_2, e_4, e_5\}$
28	1	1	1	0	0	1	0	
29	1	1	1	0	1	0	1	$\{e_1, e_2, e_3, e_5\}$
30	1	1	1	1	0	0	1	$\{e_1, e_2, e_3, e_4\}$
31	1	1	1	1	1	0	1	$\{e_1, e_2, e_3, e_4, e_5\}$

To determine the length of the dual system, the numbers are determined:
$l_2 = 6$ because there are six links each with two elements (links: $\{e_3, e_5\}$, $\{e_3, e_4\}$, $\{e_2, e_5\}$, $\{e_2, e_4\}$, $\{e_1, e_5\}$ and $\{e_1, e_4\}$);
$l_3 = 9$ because there are nine links each with three elements (links: $\{e_3, e_4, e_5\}$, $\{e_2, e_4, e_5\}$, $\{e_2, e_3, e_5\}$, $\{e_2, e_3, e_4\}$, $\{e_1, e_4, e_5\}$, $\{e_1, e_3, e_5\}$, $\{e_1, e_3, e_4\}$, $\{e_1, e_2, e_5\}$ and $\{e_1, e_2, e_4\}$,);
$l_4 = 5$ because there are five cuts each with four elements (links: $\{e_1, e_2, e_3, e_4\}$, $\{e_1, e_2, e_3, e_5\}$, $\{e_1, e_2, e_4, e_5\}$, $\{e_1, e_3, e_4, e_5\}$ and $\{e_2, e_3, e_4, e_5\}$);
$l_5 = 1$ because there is only one link with five elements (cut: $\{e_1, e_2, e_3, e_4, e_5\}$).

The S^d dual system length is equal to: $\lambda(S^d) = min\{k \mid c_k > 0\} = min\{2,3,4,5\} = 2$.

Using the Matlab function above, we will determine all the links of this system and the length of the system.

The structure function denoted by φ coincides with the function denoted by f in the Matlab code.

An example of execution is:

```
>> links(5)
The value 5 :
y = f (  0 ,  0 ,  1 ,  0 ,  1  ) = 1
Subassembly L = {  e3  e5  } is a link
The value 6 :
y = f (  0 ,  0 ,  1 ,  1 ,  0  ) = 1
Subassembly L = {  e3  e4  } is a link
The value 7 :
y = f (  0 ,  0 ,  1 ,  1 ,  1  ) = 1
Subassembly L = {  e3  e4  e5  } is a link
The value 9 :
y = f (  0 ,  1 ,  0 ,  0 ,  1  ) = 1
```

Subassembly L = { e2 e5 } is a link
The value 10 :
y = f (0 , 1 , 0 , 1 , 0) = 1
Subassembly L = { e2 e4 } is a link
The value 11 :
y = f (0 , 1 , 0 , 1 , 1) = 1
Subassembly L = { e2 e4 e5 } is a link
The value 13 :
y = f (0 , 1 , 1 , 0 , 1) = 1
Subassembly L = { e2 e3 e5 } is a link
The value 14 :
y = f (0 , 1 , 1 , 1 , 0) = 1
Subassembly L = { e2 e3 e4 } is a link
The value 15 :
y = f (0 , 1 , 1 , 1 , 1) = 1
Subassembly L = { e2 e3 e4 e5 } is a link
The value 17 :
y = f (1 , 0 , 0 , 0 , 1) = 1
Subassembly L = { e1 e5 } is a link
The value 18 :
y = f (1 , 0 , 0 , 1 , 0) = 1
Subassembly L = { e1 e4 } is a link
The value 19 :
y = f (1 , 0 , 0 , 1 , 1) = 1
Subassembly L = { e1 e4 e5 } is a link
The value 21 :
y = f (1 , 0 , 1 , 0 , 1) = 1
Subassembly L = { e1 e3 e5 } is a link
The value 22 :
y = f (1 , 0 , 1 , 1 , 0) = 1
Subassembly L = { e1 e3 e4 } is a link
The value 23 :
y = f (1 , 0 , 1 , 1 , 1) = 1
Subassembly L = { e1 e3 e4 e5 } is a link
The value 25 :
y = f (1 , 1 , 0 , 0 , 1) = 1
Subassembly L = { e1 e2 e5 } is a link
The value 26 :
y = f (1 , 1 , 0 , 1 , 0) = 1
subassembly L = { e1 e2 e4 } is a link
The value 27 :
y = f (1 , 1 , 0 , 1 , 1) = 1
subassembly L = { e1 e2 e4 e5 } is a link
The value 29 :
y = f (1 , 1 , 1 , 0 , 1) = 1
subassembly L = { e1 e2 e3 e5 } is a link
The value 30 :
y = f (1 , 1 , 1 , 1 , 0) = 1
subassembly L = { e1 e2 e3 e4 } is a link
The value 31 :
y = f (1 , 1 , 1 , 1 , 1) = 1

Subassembly L = { e1 e2 e3 e4 e5 } is a link
Number of links with 2 elements = 6
Number of links with 3 elements = 9
Number of links with 4 elements = 5
Number of links with 5 elements = 1
The length of the dual system is 2

It is easy to see that the linkages generated by the Matlab function coincide with the links determined manually in Table 1.

Also, the length of the system determined with the Matlab function coincides with the one determined manually.

4. Conclusions

One of the future development directions is to determine the minimal links of the $S = (E, \varphi)$ system and its dual system $S^d = \left(E, \varphi^d\right)$.

Acknowledgements

The author thanks this way to all those who contributed in one way or another to the elaboration of this work.

5. References

[1] Cătuneanu V M and Mihalache A 1983 *Bazele teoretice ale fiabilităţii*, Ed. Academiei, Bucureşti, România
[2] Panaite V and Popescu M O 2003 *Calitatea produselor şi fiabilitate*, Ed. Matrix Rom, Bucureşti, România
[3] Ţârcolea C, Filipoiu A and Bontaş S 1989 *Tehnici actuale în teoria fiabilităţii*, Ed. Ştiinţifică şi Enciclopedică, Bucureşti, România
[4] Vasiliu P and Deliu F 2017 *Funcţie Matlab pentru determinarea legăturilor unei reţele de fiabilitate*, Buletinul AGIR nr. 4/2017, pag. 24-27
[5] Vasiliu P 2015 *Programare în Matlab,* Ed. ANMB, Constanţa, România