2014 IEEE 44th International Symposium on Multiple-Valued Logic

Minimal Cut Vectors and Logical Differential Calculus

Miroslav Kvassay, Elena Zaitseva, Vitaly Levashenko, Jozef Kostolny
Faculty of Management Science and Informatics
University of Zilina
Zilina, Slovakia
{miroslav.kvassay, elena.zaitseva, vitaly.levashenko, jozef.kostolny}@fri.uniza.sk

***Abstract*—Reliability analysis of Multi-State System is one of fields, where the methods of the multiple-valued logic are used efficiently. Multi-State System is a mathematical model in reliability engineering, which allows considering some performance level in the system reliability behavior. The principal problem in the reliability engineering is identification of the system components that have the most influence on the system reliability. One of possible approaches, which can be used for this task, is to identify minimal cut sets (or minimal cut vectors (MCVs)). MCVs represent situations in which the repair of any damaged component causes the system improvement. In this paper, the new theoretical background for identification of MCVs is considered. The presented method is based on logical differential calculus.**

Keywords-reliability; structure function; minimal cut vector; logical differential calculus

I. INTRODUCTION

Reliability is one of the basic characteristics of many systems. There are a lot of methods for estimation of different aspects of the system reliability. Many of these methods are indicated as *Minimal Cut Set* (MCS) methods [1-4]. A MCS is a minimum set of system components where the fault condition for all components in the set results system outage. The MCS method is useful in evaluation of the reliability of series-parallel systems or small systems. In papers [3, 5-7] several methods for computation of MCSs have been proposed. For large systems, it is quite difficult and time consuming to identify MCS components by inspection. The principal difficulty of these methods is application for reliability analysis of complex systems, which contain a lot of components or components that are very different in their nature, e.g. hardware, software and human factor, which implies that those system contains different types of connections and not only series or parallel. Those reliability studies are generally off-line studies, but because of the combinatorial nature of the calculation, the calculation time is nonetheless an issue. Thus, it is important to find a better method to determine MCSs for large and complex systems.

The MCS method is used in evaluation of the system reliability and availability, which are computed based on the failure probability of each component of the set [1, 3]. Another application of the MCS method is importance analysis, which aim is to identify influence of individual components on the system reliability. *Importance Measures* (IMs) are used for quantification of this influence. One of the most commonly used IMs is the Fussell-Vesely IM [8-10] that is defined based on MCSs. There exist some mathematical backgrounds for the calculation of MCSs. One of them is based on the fault trees [5, 6, 11, 12] and another uses graph theory to find MCSs in different types of network systems [3, 4, 7].

The principal point in the development of the methods and algorithms for the calculation of MCSs is a mathematical model and interpretation of a system. There are two types of mathematical models in reliability analysis [13, 14]: *Binary-State System* (BSS) and *Multi-State System* (MSS). Only two states are possible for a BSS: system components and system can be in one of two states that are working and failure. These two states are represented by two numbers – 1 (functioning) and 0 (failure). MSS permits to consider more than two states in system/components performance, for example: failed, partially functional, fully operational, etc. These states are represented by numbers from 0 (failure) to m -1 (fully operational). The reliability of these systems is computed with regard to the system performance. The correlation between states of components and the system performance level is defined by the structure function [13, 15]:

$$\phi(\boldsymbol{x}): \{0, 1,\dots, m\text{ -}1\}^n \rightarrow \{0, 1,\dots, m\text{ -}1\}, \qquad (1)$$

where n is a number of system components, m is a number of states of every component and the system, and $\boldsymbol{x} = (x_1, x_2,\dots, x_n)$ is a vector of components states (state vector). For $m = 2$, definition (1) is a definition of the structure function of a BSS.

The MSS reliability and unreliability based on the conception of the structure function (1) are defined as [13]:

$$R(j) = \Pr\{\phi(\boldsymbol{x}) \geq j\}, j = 1,2,\dots, m\text{ -}1, \qquad (2)$$

$$U = \Pr\{\phi(\boldsymbol{x}) = 0\}. \qquad (3)$$

There is another definition of the MSS unreliability by MCSs [13]:

$$U = \Pr\left\{\bigcup_{v=1}^{t} \mathrm{MCS}_v\right\}, \qquad (4)$$

where t is a number of MCSs for system state 1 and MCS_v is the v-th MCS for system state 1.

In this paper, we consider approach (2) for estimation of the MSS reliability. In this case, the probability of the system component state is used and defined as:

$$p_{i,s} = \Pr\{x_i = s\}, s = 0,1,\dots, m\text{ -}1. \qquad (5)$$

0195-623X/14 $31.00 © 2014 IEEE
DOI 10.1109/ISMVL.2014.37

IEEE computer society

We propose new algorithm for calculation of MCSs for a MSS based on the structure function (1). The structure function definition (1) corresponds with the formal definition of the Multiple-Valued Logic (MVL) function. This fact allows using some methods and techniques of the MVL for reliability analysis of MSSs. One of those methods is logical differential calculus. The application of logical differential calculus in MSS reliability analysis has been introduced in papers [16, 17]. In those papers authors proposed a new method for calculation of IMs based on the *Direct Partial Logic Derivatives* (DPLD). In this paper, another application of this mathematical tool is considered: the definition and computation of MCSs of a coherent MSS, which meets the following assumptions [13, 14]:

(a) every component is relevant to the system performance,

(b) the system structure function (1) is non-decreasing, i.e. degradation of any system component cannot cause improvement of system performance level.

There are different methods for the reliability estimation of a coherent MSS. Most of these methods are based on the investigation of boundary states of the system, which represent situations in which the change of one system component state causes the transformation of the system performance level. A MCS is one type of boundary states. Therefore, the calculation of MCSs is a principal problem in reliability analysis. A new algorithm for computation of MCSs is proposed in this paper.

II. Minimal Cut Vectors

A. Minimal Cut Set and Minimal Cut Vector of Binary-State System

MCS methods have been proposed for BSS reliability analysis in the first. The quantification of the BSS reliability based on the conception of MCS has been considered and used in many papers [1-7].

According to [1] a cut set is defined as a set of components of a system whose simultaneous failure leads into the failure of the system (if the system has been operational). A cut set is minimal, if no component can be removed from it without losing its status as a cut set.

In the terms of the structure function, a (minimal) cut set can be interpreted by a special state vector, which is known as a (*Minimal*) *Cut Vector* (MCV). According to the definition of cut set, the system state for a state vector covered by a cut set is zero. Therefore *state vector* $\boldsymbol{x}$ *is a cut vector if* $\phi(\boldsymbol{x}) = 0$.

So, if all components of a cut set are failed and components out of the cut set are functioning then the system is failed. A state vector, which coincides with a MCS, is known as a MCV. Using the convention that $\boldsymbol{y} > \boldsymbol{x}$, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are two states vector, for which $y_i \geq x_i$ (for $i = 1,2,\ldots,n$) and there exists at least one i such that $y_i > x_i$, we say that *a cut vector* $\boldsymbol{x}$ *is minimal if* $\phi(\boldsymbol{y}) = 1$ *for any* $\boldsymbol{y} > \boldsymbol{x}$.

There is one-to-one correspondence between MCSs and MCVs. However, the terms MCS and MCV are slightly different. A MCS is a minimal set of components, whose simultaneous failure causes the system failure, while MCV represents situation in which the repair of any failed component results into the repair of the whole system.

B. Minimal Cut Set and Minimal Cut Vector of Multi-State System

The definition of MCS has been generalized for MSSs in paper [18]. The development of this conception for MCVs of a MSS has been proposed in [13, 19, 20]. In papers [21-23], there have been developed and analyzed some algorithms that can be used to find all MCVs of a network system, which is modelled as a MSS. However, those algorithms assume that cut sets of source-sink cuts (in the sense of graph theory) of a network are known and therefore they cannot be used to find MCVs in other types of systems.

The generalization of MCV definition for a MSS takes into account that components of a MSS have more than two states. This extension is based on the assumption that MCV is defined for every relevant system state, i.e. for states $\{1, 2,\ldots, m-1\}$. The definition of a MCV of a MSS has been proposed in [13, 19] as follows: *a state vector* $\boldsymbol{x}$ *is a cut vector for demand state* j *of the system if* $\phi(\boldsymbol{x}) < j$. *A cut vector* $\boldsymbol{x}$ *is minimal if* $\phi(\boldsymbol{y}) \geq j$ *for any* $\boldsymbol{y} > \boldsymbol{x}$.

The meaning of MCVs for a MSS is similar as in the case of a BSS, i.e. MCVs for system state j identify those situations in which the repair of any damaged component causes the improvement of the system at least to state j.

III. Logical Differential Calculus

Logical differential calculus is a special tool, which is used for analyzing of dynamic properties of MVL functions. In papers [16, 17], applications of logical differential calculus in reliability analysis have been introduced. Those applications are based on DPLDs.

There exist several types of logic derivatives in MVL. One of them is a *Direct Partial Logic Derivative* (DPLD). A DPLD can be used for the analysis of the dynamic properties of a MVL function or a MSS structure function, in the case of reliability analysis. These derivatives reflect the change in the value of the underlying function when the value of given variable changes [16, 24]. DPLD $\partial\phi(l \to \tilde{l})/\partial x_i(s \to \tilde{s})$ of a MVL function $\phi(\boldsymbol{x})$ of n variables with respect to variable x_i reflects the fact of changing of the function from l to $\tilde{l}$ when the value of variable x_i changes from s to $\tilde{s}$ [24]. In terms of reliability analysis, a DPLD with respect to variable x_i for the MSS structure function (1) permits to analyze the system performance level change from value l to $\tilde{l}$ when the i-th component state changes from s to $\tilde{s}$ [16].

A DPLD for the MSS structure function has some specific properties for a coherent MSS. According to the assumption (b), a DPLD of the structure function is nonzero if $l > \tilde{l}$ and $s > \tilde{s}$ or $l < \tilde{l}$ and $s < \tilde{s}$. The condition $l > \tilde{l}$ and $s > \tilde{s}$ allows analyzing the system degradation and failure, and the condition $l < \tilde{l}$ and $s < \tilde{s}$ is used to investigate the system performance level improving. Assumptions (a) and (b) cause gradual changes of the function value and the same variable. Below, we consider only the improving of the system performance level, because the mathematical backgrounds of the evaluation of the system performance level degradation and improvement are similar. Therefore, a DPLD for analysis of a coherent MSS is defined for the system improvement as [16]:

$$\frac{\partial\phi(l \to \tilde{l})}{\partial x_i(s \to s+1)} = \begin{cases} 1 & \text{if } \phi(s_i, \boldsymbol{x}) = l \text{ and } \phi((s+1)_i, \boldsymbol{x}) = \tilde{l} \\ 0 & \text{other} \end{cases}, \quad (6)$$

where $\phi(s_i, \boldsymbol{x}) = \phi(x_1,\ldots,x_{i-1}, s, x_{i+1},\ldots,x_n)$; $s \in \{0,\ldots, m-2\}$; $l, \tilde{l} \in \{0,\ldots, m-1\}$ and $l < \tilde{l}$.

DPLD (6) allows discovering boundary system states for which the change of component i state from s to $s+1$ causes the change of MSS performance level from l to $\tilde{l}$.

IV. Minimal Cut Vectors and Logical Differential Calculus

A MCV corresponds to system components states for which an improving of one of component state causes the improvement of the system performance level. Consider a MCV from the point of view of a DPLD with respect to the i-th variable. Let state vector $\boldsymbol{x}$ be a MCV for system state j. It means that the value of the structure function for it is l, $l < j$. Let $\boldsymbol{x} = ((m-1)_1,\ldots,c_{r_1},\ldots,c_{r_k},\ldots,(m-1)_n)$, where k is a number of components which are in state less than m -1 and r_q (for $q = 1, 2,\ldots, k$) is the index of the q-th component, which state is less than m -1, and c_{r_q} is a state of component r_q. According to the definition of a MCV of a MSS, the change of any component r_q from c_{r_q} to $c_{r_q}+1$ causes the improvement of the structure function value from l to $\tilde{l}_q$, where $\tilde{l}_q \geq j$. Therefore, DPLDs $\partial\phi(l \to \tilde{l})/\partial x_{r_q}(c_{r_q} \to c_{r_q}+1)$ in regard to variable x_{r_q}, for $q = 1, 2,\ldots, k$, have the nonzero value for the considered state vector (Fig. 1). This derivative is calculated according to (6). According to the definition of a MCV, every system component state change causes the system performance level improving, therefore every DPLD with respect to variable x_{r_q}, for $q = 1, 2,\ldots, k$, has the nonzero value for the state vector that agrees with the MCV.

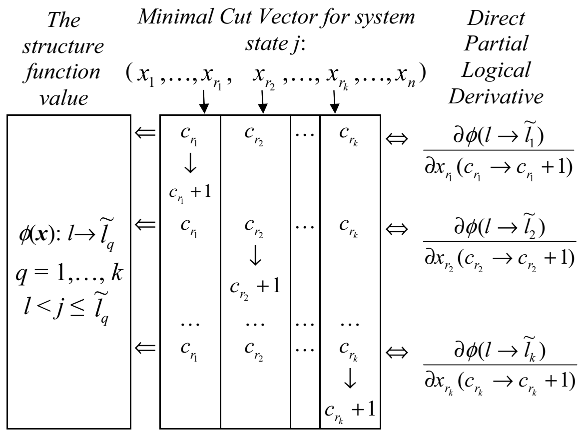

Figure 1. Minimal Cut Vector and Direct Partial Logical Derivatives of the structure function.

The union of all DPLDs $\partial\phi(l \to \tilde{l})/\partial x_{r_q}(c_{r_q} \to c_{r_q}+1)$ that meet the property $l < j$ and $\tilde{l} \geq j$ has to be computed to identify all situations in which the improvement of the fixed component state results the transition of the system from state less than j to state greater than or equal to j:

$$\frac{\partial\phi(\uparrow j \uparrow)}{\partial x_i(s \to s+1)} = \bigcup_{l=0}^{j-1}\left(\bigcup_{\tilde{l}=j}^{m-1}\frac{\partial\phi(l \to \tilde{l})}{\partial x_i(s \to s+1)}\right), \quad (7)$$

for $s = 0, 1,\ldots, m-2$ and $j = 1, 2,\ldots, m-1$.

DPLD union (7) can be defined by the next equation too:

$$\frac{\partial\phi(\uparrow j \uparrow)}{\partial x_i(s \to s+1)} = \begin{cases} 1 & \text{if } \phi(s_i, \boldsymbol{x}) < j \text{ and } \phi((s+1)_i, \boldsymbol{x}) \geq j \\ 0 & \text{other} \end{cases}, \quad (8)$$

for $s = 0, 1,\ldots, m-2$ and $j = 1, 2,\ldots, m-1$.

DPLD (6) and union (7), (8) of DPLDs can be calculated for state vectors $\boldsymbol{x} = (x_1,\ldots,x_{i-1}, s, x_{i+1},\ldots,x_n)$. For state vectors $\boldsymbol{x} = (x_1,\ldots,x_{i-1}, a, x_{i+1},\ldots,x_n)$, in which $a \neq s$, DPLD (6) and union (7), (8) do not exist. Therefore the dimension of a DPLD with respect to state s of variable x_i is m^{n-1}. It implies that DPLDs union (7), (8) for the i-th variable has dimension m^{n-1} too. But the calculation of the MCVs supposes analysis of the all possible system states. Therefore DPLDs union (7), (8) has to be transformed into the extended union, in which the non-existing values of DPLDs union are designated by special symbol "*" (Fig. 2). The dimension of the extended union is m^n and it can be defined as follows:

$$\frac{\partial\phi(\uparrow j \uparrow)}{\partial x_i(s \to s+1)} = \begin{cases} 1 & \text{if } x_i = s \text{ and } \phi(s_i, \boldsymbol{x}) < j \text{ and } \phi((s+1)_i, \boldsymbol{x}) \geq j \\ 0 & \text{if } x_i = s \text{ and } \left(\phi(s_i, \boldsymbol{x}) \geq j \text{ or } \phi((s+1)_i, \boldsymbol{x}) < j\right) \\ * & \text{if } x_i \neq s \end{cases}, \quad (9)$$

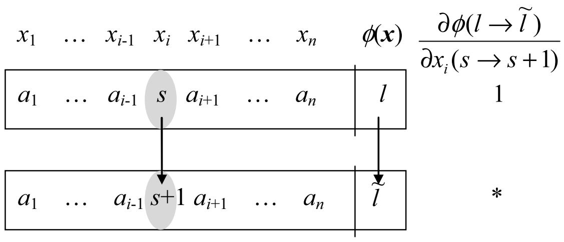

Figure 2. Direct Partial Logic Derivative of the structure function $\phi(\boldsymbol{x})$ with respect to variable x_i.

The merge (union) of unions (9) of DPLDs has to be computed to identify all situations in which any minor improvement of a given component causes the transition of the system from state less than j to state greater than or equal to j:

$$\frac{\partial\phi(\uparrow j\uparrow)}{\partial x_i} = \bigcup_{s=0}^{m-2} \frac{\partial\phi(\uparrow j\uparrow)}{\partial x_i(s\to s+1)} \quad \text{for } j=1,2,\ldots,m-1. \quad (10)$$

Using definition (9) of the extended union, the merge (10) can also be expressed in the following manner:

$$\frac{\partial\phi(\uparrow j\uparrow)}{\partial x_i} = \\ = \begin{cases} 1 & \text{if } x_i < m-1 \text{ and } \phi(s_i,\boldsymbol{x}) < j \text{ and } \phi((s+1)_i,\boldsymbol{x}) \ge j \\ 0 & \text{if } x_i < m-1 \text{ and } (\phi(s_i,\boldsymbol{x}) \ge j \text{ or } \phi((s+1)_i,\boldsymbol{x}) < j) \\ * & \text{if } x_i = m-1 \end{cases}, \quad (11)$$

The merge (10), (11) of extended unions (9) of DPLDs has to be computed for all components of the system and then their "intersection" has to be computed to identify MCVs for system state *j*. The intersection of two merges (10), (11) of two different variables (components) is defined in Table I. This intersection identifies state vectors in which the change of states of both components (if component state can be changed) results in the change of the system state from value less than *j* to value greater than or equal to *j*. In Table I, symbol "1" means that at least state of one component of components i_1 and i_2 can be changed and all those changes result in the required change of the system state (from state less than *j* to state greater than or equal to *j*). Symbol "0" identifies those state vectors in which at least one component change does not cause the required change of the system state. Finally, symbol "*" correlates with those situations, when neither component i_1 nor component i_2 can be changed from state *s* to state *s* +1, because both components are in state *m* -1.

TABLE I. THE INTERSECTION OF TWO MERGES OF UNIONS OF DPLDS

		$\partial\phi(\uparrow j\uparrow)/\partial x_{i_2}$		
		*	0	1
$\partial\phi(\uparrow j\uparrow)/\partial x_{i_1}$	*	*	0	1
	0	0	0	0
	1	1	0	1

The intersection of two merges of extended unions of DPLDs identifies state vectors in which the improvement of both components (if component can be repaired, i.e. if component is not in state *m* -1) results in the improvement of the system from state less than *j* to state greater than or equal to *j*. So, the intersection of all merges of extended unions of DPLDs identifies state vectors in which improvement of any component (if component can be repaired) results in the required repair of the system, and this correlates with the definition of a MCV.

V. ALGORITHM FOR COMPUTATION OF MINIMAL CUT VECTORS

According to correlations between MCVs and DPLDs, which have been formulated in the previous part, an algorithm for calculation of MCVs for state *j* of a MSS of *n* components can be defined. This algorithm has next steps:

1. For every component, i.e. for $i = 1, 2,\ldots, n$, and for relevant component states, i.e. for $s = 0, 1,\ldots, m-2$, compute all DPLDs $\partial\phi(l\to\tilde{l})/\partial x_i(s\to s+1)$ for $l<j\le\tilde{l}$.
2. Using DPLDs from step 1, compute the extended unions $\partial\phi(\uparrow j\uparrow)/\partial x_i(s\to s+1)$ for $i = 1, 2,\ldots, n$ and $s = 0, 1,\ldots, m-2$.
3. Calculate the merge $\partial\phi(\uparrow j\uparrow)/\partial x_i$ of extended unions for $i = 1, 2,\ldots, n$, i.e. for every system component.
4. Compute the intersection of all merges $\partial\phi(\uparrow j\uparrow)/\partial x_i$ of extended unions according to Table I.
5. Define MCVs for state *j* that agree to the value 1 in the intersection, calculated in the previous step.

Steps 1 – 3 in the algorithm can be replaced by only one step that is based on (11), which defines simple scheme for computation of merge of extended unions. Therefore, we get the following simple algorithm for calculation of MCVs for state *j* of a MSS:

1. According to (11), for every system component, i.e. for $i = 1, 2,\ldots, n$, find the merge $\partial\phi(\uparrow j\uparrow)/\partial x_i$ of extended unions.
2. Compute the intersection of all merges $\partial\phi(\uparrow j\uparrow)/\partial x_i$ of extended unions according to Table I.
3. Define MCVs for state *j* that agrees to the value 1 in the intersection, calculated in the previous step.

VI. EXAMPLE

Consider the system that is investigated in paper [25]. It is a simple power grid, which consist of three generation plants – coal (component 1), hydro (component 2) and wind (component 3) plants. This system has three performance levels: 0 – non-operational, 1 – partially operational, 2 – fully operational. The structure function of this system is defined in Table II.

TABLE II. THE STRUCTURE FUNCTION OF THE SIMPLE POWER GRID

		x_3		
x_1	x_2	0	1	2
0	0	0	0	0
0	1	0	1	1
0	2	1	1	1
1	0	0	0	0
1	1	1	1	1
1	2	2	2	2
2	0	0	0	0
2	1	2	2	2
2	2	2	2	2

Consider as example the calculation of MCVs for the performance level 1 of this system by the approach based on DPLDs above mentioned.

According to the mathematical definition (8) of this approach, two DPLDs $\partial\phi(0\to 1)/\partial x_i(s\to s+1)$ and $\partial\phi(0\to 2)/\partial x_i(s\to s+1)$, for $i = 1, 2, 3$ and $s = 0, 1$, have to be computed. These derivatives are combined into the extended union $\partial\phi(\uparrow 1\uparrow)/\partial x_i(s\to s+1)$, for $s = 0, 1$, based on the rule (9). These extended unions for the first variable are presented in Table III.

The merge (10) of two extended unions of component 1 is presented in Table III too. The merge $\partial\phi(\uparrow 1\uparrow)/\partial x_1$ identifies states vector for which any minor improvement of the 1-th component causes the transition of the system from state 0 to states 1 or 2. The intersection of merges (10) detects state vectors for which any minor improvement of any component causes the transition of the system from state 0 to states 1 or 2, i.e. this intersection finds state vectors, which are MCVs for system state 1. The merge of extended unions for every component of the power grid system and their intersection are calculated in Table IV. From this table, we can see that there are 2 MCVs for state 1 of the studied system, i.e. (0,1,0) and (2,0,2). These MCVs represents boundary states for system unreliability and therefore they can be used to calculate system unreliability (4):

$$U = \Pr\{\phi(\boldsymbol{x}) = 0\} = \Pr\{\phi(\boldsymbol{x}) < 1\} \\ = \Pr\{\boldsymbol{x} \le (0,1,0) \text{ OR } \boldsymbol{x} \le (2,0,2)\}. \tag{12}$$

TABLE III. EXTENDED UNIONS OF DPLDS AND THEIR MERGE

x_1 x_2 x_3	$\phi(\boldsymbol{x})$	$\frac{\partial\phi(\uparrow 1\uparrow)}{\partial x_1(0\to 1)}$	$\frac{\partial\phi(\uparrow 1\uparrow)}{\partial x_1(1\to 2)}$	$\frac{\partial\phi(\uparrow 1\uparrow)}{\partial x_1}$
0 0 0	0	0	*	0
0 0 1	0	0	*	0
0 0 2	0	0	*	0
0 1 0	0	1	*	1
0 1 1	1	0	*	0
0 1 2	1	0	*	0
0 2 0	1	0	*	0
0 2 1	1	0	*	0
0 2 2	1	0	*	0
1 0 0	0	*	0	0
1 0 1	0	*	0	0
1 0 2	0	*	0	0
1 1 0	1	*	0	0
1 1 1	1	*	0	0
1 1 2	1	*	0	0
1 2 0	2	*	0	0
1 2 1	2	*	0	0
1 2 2	2	*	0	0
2 0 0	0	*	*	*
2 0 1	0	*	*	*
2 0 2	0	*	*	*
2 1 0	2	*	*	*
2 1 1	2	*	*	*
2 1 2	2	*	*	*
2 2 0	2	*	*	*
2 2 1	2	*	*	*
2 2 2	2	*	*	*

VII. CONCLUSION

MCSs and MCVs are the basic for qualitative analyses of systems because they represent minimal scenarios of the system performance level change. Therefore, the development of algorithms for computation of MCVs is an important problem in reliability engineering. In this paper, theoretical background for new algorithm, which can be used to solve this problem, is considered. This mathematical background is based on the correlation between MCVs and DPLDs. We have shown that MCVs can be derived from DPLDs. This fact has allowed us to proposed simple algorithm for computation of MCVs of a MSS which computational complexity does not depend on the number of MCVs and can be used for analysis of any coherent MSS.

TABLE IV. MERGE OF UNIONS OF DPLDS AND THEIR INTERSECTION

x_1 x_2 x_3	$\phi(\boldsymbol{x})$	$\frac{\partial\phi(\uparrow 1\uparrow)}{\partial x_1}$	$\frac{\partial\phi(\uparrow 1\uparrow)}{\partial x_2}$	$\frac{\partial\phi(\uparrow 1\uparrow)}{\partial x_3}$	**The intersection of $\partial\phi(\uparrow 1\uparrow)/\partial x_i$**
0 0 0	0	0	0	0	0
0 0 1	0	0	1	0	0
0 0 2	0	0	1	*	0
0 1 0	0	1	1	1	1
0 1 1	1	0	0	0	0
0 1 2	1	0	0	*	0
0 2 0	1	0	*	0	0
0 2 1	1	0	*	0	0
0 2 2	1	0	*	*	0
1 0 0	0	0	1	0	0
1 0 1	0	0	1	0	0
1 0 2	0	0	1	*	0
1 1 0	1	0	0	0	0
1 1 1	1	0	0	0	0
1 1 2	1	0	0	*	0
1 2 0	2	0	*	0	0
1 2 1	2	0	*	0	0
1 2 2	2	0	*	*	0
2 0 0	0	*	1	0	0
2 0 1	0	*	1	0	0
2 0 2	0	*	1	*	1
2 1 0	2	*	0	0	0
2 1 1	2	*	0	0	0
2 1 2	2	*	0	*	0
2 2 0	2	*	*	0	0
2 2 1	2	*	*	0	0
2 2 2	2	*	*	*	*

ACKNOWLEDGMENT

This work was supported by the grant FP7 "Regional Anaesthesia Simulator and Assistant" (RASimAs) and grant of Slovak Research and Development Agency SK-PL-0023-12.

REFERENCES

[1] M. L. Shooman, Probabilistic Reliability: An Engineering Approach. New York, NY: McGraw-Hill, 1968.

[2] Z. Ling, Y. Shan-shui, C. Lin, and W. Li, "Reliability analysis of DC power distribution network based on minimal cut sets," Proc. of the 2011-14th European Conference on Power Electronics and Applications (EPE 2011), August-September 2011, pp. 1-7.

[3] M. Al-Muhaini and G. T. Heydt, "Minimal cut sets, Petri nets, and prime number encoding in distribution system reliability evaluation," Proc. Transmission and Distribution Conference and Exposition (T&D), 2012 IEEE PES, May 2012, pp. 1-8, doi: 10.1109/TDC.2012.6281495.

[4] J. M. Nahman, "Minimal paths and cuts of networks exposed to common-cause failures," IEEE Transactions on Reliability, vol. 41, no. 1, pp. 76-80, 84, March 1992, doi: 10.1109/24.126675.

[5] L. Rosenberg, "Algorithm for finding minimal cut sets in a fault tree," Reliability Engineering & System Safety, vol. 53, no. 1, pp. 67-71, July 1996, doi: 10.1016/0951-8320(96)00034-8.

[6] J. S. Choi and N. Z. Cho, "A practical method for accurate quantification of large fault trees," Reliability Engineering & System Safety, vol. 92, no. 7, pp. 971-982, July 2007, doi: 10.1016/j.ress.2006.07.005.

[7] W.-C. Yeh, "An improved algorithm for searching all minimal cuts in modified networks," Reliability Engineering & System Safety, vol. 93, no. 7, pp. 1018-1024, July 2008, doi: 10.1016/j.ress.2007.05.003.

[8] W. E. Vesely, “A time-dependent methodology for fault tree evaluation,” Nuclear Engineering and Design, vol. 13, no. 2, pp. 337-360, August 1970, doi: 10.1016/0029-5493(70)90167-6.

[9] J. B. Fussell, “How to hand-calculate system reliability and safety characteristics,” IEEE Transactions on Reliability, vol. R-24, no. 3, pp. 169-174, August 1975, doi: 10.1109/TR.1975.5215142.

[10] W. Kuo and X. Zhu, Importance Measures in Reliability, Risk, and Optimization: Principles and Applications. Chichester, UK: Wiley, 2012.

[11] J. Vatn, “Finding minimal cut sets in a fault tree,” Reliability Engineering & System Safety, vol. 36, no. 1, pp. 59-62, 1992, doi: 10.1016/0951-8320(92)90152-B.

[12] N. Fard, “Determination of minimal cut sets of a complex fault tree,” Computers & Industrial Engineering, vol. 33, no. 97, pp. 59-62, October 1997, doi: 10.1016/S0360-8352(97)00041-7.

[13] A. Lisnianski, G. Levitin, Multi-state System Reliability. Assessment, Optimization and Applications. Singapore, SG: World Scientific, 2003.

[14] B. Natvig, Multistate systems reliability theory with applications. New York, NY: Wiley, 2011.

[15] E. Zaitseva and V. Levashenko, “Multiple-valued logic mathematical approaches for multi-state system reliability analysis,” Journal of Applied Logic, vol. 11, no. 3, pp. 350-362, September 2013, doi: 10.1016/j.jal.2013.05.005.

[16] E. Zaitseva, “Importance analysis of a multi-state system based on multiple-valued logic methods,” Recent advances in system reliability: signatures, multi-state systems and statistical inference, A. Lisnianski and I. Frenkel, Eds. London, UK: Springer, 2012, pp. 113-134, doi: 10.1007/978-1-4471-2207-4_8.

[17] E. Zaitseva, V. Levashenko, K. Matiasko, S. Puuronen, “Dynamic Reliability Analysis of the k-out-of-n Multi-State System”. Proc. of the IEEE 35th Int. Symp. on Multiple-Valued Logic, May 2005, pp. 264-269, doi: 10.1109/ISMVL.2005.16.

[18] D. A. Butler, “A complete importance ranking for components of binary coherent systems with extensions to multi-state systems,” Naval Research Logistics, vol. 4, pp. 565-578, 1979.

[19] T. Aven, “Reliability evaluation of multistate systems with multistate components,” IEEE Transactions on Reliability, vol. R-34, no. 5, pp. 473-479, December 1985, doi: 10.1109/TR.1985.5222235.

[20] S. Soh and S. Rai, “An efficient cutset approach for evaluating communication-network reliability with heterogeneous link-capacities,” IEEE Transactions on Reliability, vol. 54, no. 1, pp. 133-144, March 2005, doi: 10.1109/TR.2004.842530.

[21] W. Yeh, “A simple approach to search for all d-MCs of a limited-flow network,” Reliability Engineering & System Safety, vol. 71, no. 1, pp. 15-19, January 2001, doi: 10.1016/S0951-8320(00)00070-3.

[22] W. Yeh, “A new approach to the d-MC problem,” Reliability Engineering & System Safety, vol. 77, no. 2, pp. 201-206, August 2002, doi: 10.1016/S0951-8320(02)00038-8.

[23] W. Yeh, “A fast algorithm for searching all multi-state minimal cuts,” IEEE Transactions on Reliability, vol. 57, no. 4, pp. 581-588, December 2008, doi: 10.1109/TR.2008.2006293.

[24] M. A. Tapia, T. A. Guima, A. Katbab, “Calculus for a multivalued-logic algebraic system,” Applied Mathematics and Computation, vol. 42, no. 3, pp. 255-285, April 1991, doi: 10.1016/0096-3003(91)90004-7.

[25] T. W. Manikas, M. A. Thornton, and D. Y. Feinstein, “Using multiple-valued logic decision diagrams to model system threat probabilities,” in 2011 41st IEEE International Symposium on Multiple-Valued Logic, May 2011, pp. 263-267, doi: 10.1109/ISMVL.2011.12.