Control and Cybernetics

vol. **24** (1995) No. 2

A method of reliability analysis of complex equipment[1]

by

J. Karpiński and D. Gątarek

Systems Research Institute,
Polish Academy of Sciences, ul. Newelska 6,
01-447 Warszawa
Poland

In the paper we present an algorithm for computing reliability of complex systems which is a junction of the top-down, bottom-up and Poincaré algorithms. Our algorithm consists of two steps: first we code the reliability structure of the system in a convenient form and then we compute reliability using this new form. The algorithm is especially oriented on systems whose structure is constant in time, but the reliability of components changes or is not exactly known. As the mathematical tool we use the relation theory and all obtained results are formally proved. The algorithms are written in a formal Pascal-like language. A simple example is presented.

Keywords: min-cuts, min-paths, reliability structure

1. Introduction

The problem of reliability evaluation of complex systems is one of the basic problems in reliability theory. In last few decades several important algorithms for evaluating system reliability have been developed. Various approaches are used but the most important ones are: cut set - path set technique and fault tree analysis (FTA). Constructing reliability structure for complex two-state system with two-state components is easier and more efficient if the system structure is presented with the help of fault trees. On the other hand, the cut set - path set approach is more general.

Even though much work has been done in developing methods to determine reliability of complex systems: Abraham (1979); Aggarwal, Misra, Gupta (1975); Barlow (1984); Beichelt, Spross (1987a;b); Bennetts (1975); Hariri, Raghavendra (1987); Jensen, Bellmore (1969); Jiang (1985); Locks (1985a;b;1978); Page, Perry (1986;1988); Schneeweiss (1985); Torrey (1988), there is a lack of

1 Part of this research was carried out when the first author was supported by the State Committee for Scientific Research, Poland, under grant number KBN 33118 91 02.

efficient, fast procedures. The focus of this paper is on the theoretical aspects of some new, very practical method of analyzing the system reliability structure. We present here an algorithm which is a junction of several known methods (the top-down, bottom-up and Poincaré algorithms) and some new approaches. The main idea of the algorithm consists in obtaining a convenient form of reliability structure of a system which enables quick evaluations of reliability factors and characteristics. Our algorithm consists of two steps: first we code the reliability structure of the system in a convenient form and then we compute reliability using this form. The main advantage of the developed method is the distinction between two problems: determining the structure and reliability evaluation. In a case of computing the reliability of a system for various number of values of component reliabilities the first step of the algorithm is executed only once. For second and next combinations of reliability data the first step is omitted, so computations are quick. The relation theory is a mathematical tool which we use to present our method. All obtained results are formally proved. The algorithm is written in a formal Pascal-like language - an extension of Pascal with the formal mathematical operations. A simple example illustrates the developed method. The algorithm presented in this paper has been found as a very efficient one and its computer realization supported by user friendly software solutions makes this method a useful analytical tool.

2. Mathematical foundations of structure analysis of reliability systems

Start with the following definitions: by the notion reliability system we understand the couple $\langle X, F\rangle$, where X is the set of system components and F is the structure function:

$F : 2^X \to \{0,1\}$, where 2^X is the set of all functions

$x : X \to \{0,1\}$. Obviously we assume that the set X is finite and denote $X = \{1,\ldots,N\}$. Denote by $\mathcal{P}(Y)$ a set of all subsets of some set Y. The function F is determined by any of the following sets:

$$\textit{cut set},\ \Phi = \{\emptyset = p \in \mathcal{P}(X) :\ F(\phi^{-1}(p)) = 0\}$$

and

$$\textit{path set},\ \Psi = \{\emptyset = p \in \mathcal{P}(X) :\ F(\psi^{-1}(p)) = 1\},$$

where $\phi(x) = \{k \in X :\ x(k) = 0\}$ and $\psi(x) = \{k \in X :\ x(k) = 1\}$ for any $x \in 2^X$ are so called cut and path, respectively. Functions ϕ and $\psi : 2^X \to \mathcal{P}(X)$ are invertible, so definitions of Φ and Ψ are correct.

Obviously, $\phi(x) = X \setminus \psi(x)$ for any $x \in 2^X$ and $\phi^{-1}(p) = 1 - \psi^{-1}(p)$ for any $p \in \mathcal{P}(X)$. The representation of the system structure as cut sets or path sets is more convenient then the matrix representation. This representation can be even simplified if we restrict to the case of *coherent structures*, i.e. when the

structure function is monotonic: $F(x) \leq F(y)$ if $x(n) \leq y(n)$ for any $n \in X$. In this case the structure can be represented as

minimal cut set $\Phi_{\min} = \{r \in \Phi : q \cap r = q$ for any $q \in \Phi$ and $r = q\}$ or

minimal path set $\Psi_{\min} = \{r \in \Psi : q \cap r = q$ for any $q \in \Psi$ and $r = q\}$.

For coherent structures the function F is determined by any of the sets $\Phi_{\min}$ and $\Psi_{\min}$, i.e.:

$F(x) = 0$ iff there exists a $q \in \Phi_{\min}$ such that $q \subseteq \phi(x)$,

$F(x) = 1$ iff there exists a $q \in \Psi_{\min}$ such that $q \subseteq \psi(x)$.

In this paper we accept the following assumptions:

ASSUMPTION 2.1 *The system $\langle X, F\rangle$ is coherent i.e. $F(x) \leq (y)$ if $x \leq y$,*

ASSUMPTION 2.2 *The system $\langle X, F\rangle$ is nondegenerated i.e. $X \in \phi$ and $X \in \Psi$ (the sets of paths and cuts are both nonempty).*

ASSUMPTION 2.3 *All elements are important, i.e. for any $x \in X$, there exists a $p \in \Phi_{\min}$ such that $F(\phi^{-1}(p \setminus \{x\})) = 1$.*

Let $\xi : \Omega \to 2^X$ be a multidimensional random variable describing the reliability state of all system components. Assume that all the components are independent, i.e. $\xi(1), \ldots, \xi(N)$ are independent random variables with values in $\{0, 1\}$.

Denote $Q(n) = \Pr(\xi(n) = 0)$ for any $n \in X$ and $Q(X)$ $\Pr(F(\xi) = 0)$. There exist simple formulas for computing system reliability, provided we know the set of minimal cuts (paths):

THEOREM 2.1 *(Poincaré formulas) The following formulas hold:*

$$Q(X) = -\sum_{T \subseteq \Phi} \{(-1)^{card(T)} \prod_{n \in \cup T} Q(n)\}, \tag{1}$$

$$1 - Q(X) = -\sum_{T \subseteq \Psi} \{(-1)^{card(T)} \prod_{n \in \cup T} (1 - Q(n))\}, \tag{2}$$

where $\cup T = \bigcup_{p \in T} p$.

These formulas are only apparently easy. For larger systems they are not to be computed even for big computers. We often use approximations of (1) and (2):

$$Q(X) \approx \sum_{p \in \Phi} [\prod_{n \in p} Q(n)] \tag{3}$$

and

$$1 - Q(X) \approx \sum_{p \in \Psi} [\prod_{n \in p} (1 - Q(n))]. \tag{4}$$

The following inequality holds:

$$1 - \sum_{p \in \Psi} [\prod_{n \in p} (1 - Q(n))] \leq Q(X) \leq \sum_{p \in \Phi} [\prod_{n \in p} Q(n)]. \tag{5}$$

There exist particular reliability structures for which reliability can be computed easily. Especially important are:
a. Parallel structure:

$$Q(X) = Q(1) \cdot Q(2) \cdot \ldots \cdot Q(N). \tag{6}$$

b. Series structure:

$$1 - Q(X) = [1 - Q(1)] \cdot [1 - Q(2)] \cdot \ldots \cdot [1 - Q(N)]. \tag{7}$$

The formulas (6) and (7) are obviously less complex than formulas (1) and (2). In fact there exist many parallel and series subsystems inside actual systems. The problem is to identify parallel and series subsystems, provided there is given the structure function, what is equivalent to the sets of minimal cuts and minimal paths. The algorithm, described below, is devoted to such a problem.

The aim of this Section is to formulate some set-theoretical relations, being a mathematical basis of further formulated algorithms. We use the theory of equivalence relations. Recall that $R \subseteq X^2$ is an equivalence relation iff:

$$xRx \text{ for any } x \in X, \tag{8}$$

$$\text{if } xRy \text{ then } yRx \text{ for any } x, y \in X, \tag{9}$$

$$\text{if } xRy \text{ and } yRz \text{ then } xRz. \tag{10}$$

Define the following relations:

RELATION 2.1 $\approx$ – *"the same parallel block": $k \approx l$ iff, there exists a sequence $p_0, \ldots, p_n \in \Psi_{\min}$ such, that $p_i \in p_{i-1} = \emptyset$ for $i = 1, 2, \ldots, n$ and $k \in p_0$ and $l \in p_n$.*

RELATION 2.2 II – *"the same series block": $k \amalg l$ iff, there exists a sequence $p_0, \ldots, p_n \in \Phi_{\min}$ such, that $p_i \in p_{i-1} = \emptyset$ for $i = 1, 2, \ldots, n$ and $k \in p_0$ and $l \in p_n$.*

LEMMA 2.1 *The relations* II *and* $\approx$ *are equivalence relations.* ■

Recall that for any equivalence relation R we can define the following sets:

$$[n]_R = \{m \in X : mRn\} \tag{11}$$

and

$$X/R = \{p \in \mathcal{P}(X) : p = [n]_R \text{ for some } n \in X\}. \tag{12}$$

The following theorem holds:

THEOREM 2.2

$$F(\phi^{-1}(p)) = \max_{Z \in X/\approx} F(\phi^{-1}(p \cap Z)), \tag{13}$$

$$F(\psi^{-1}(p)) = \min_{Z \in X/\amalg} F(\psi^{-1}(p \cap Z)), \tag{14}$$

for any $\emptyset = p \in \mathcal{P}(X)$.

Proof of Theorem 2.2.

Notice that $k, l \in \Phi_{\min}$ implies that $k \amalg l$. Hence for any $p \in \Phi_{\min}$ there exists a $Z \in X/\amalg$ such that $p \subseteq Z$. $F(\phi^{-1}(r)) = 0$ iff there exists a $p \in \Phi_{\min}$ such that $p \subseteq r$. Therefore $F(\phi^{-1}(r)) = 0$ iff there exist both a $p \in \Phi_{\min}$ and $Z \in X/\amalg$ such that $p \subseteq Z \cap r$.

Hence $F(\phi^{-1}(r)) = \min_{Z \in X/\amalg} F(\phi^{-1}(r \cap Z))$.

Proof of the second statement of the theorem is identical by duality. ■

COROLLARY 2.1 *Let* $(\xi(1), \xi(2), \ldots, \xi(N))$ *be independent random variables with values in* $\{0, 1\}$. *Then*

$$\Pr(F(\xi) = 0) = \prod_{Z \in X/\approx} \Pr(F(\phi^{-1}(Z) \cdot \xi) = 0), \tag{15}$$

$$\Pr(F(\xi) = 1) = \prod_{Z \in X/\amalg} \Pr(F(\psi^{-1}(Z) \cdot \xi) = 1), \tag{16}$$

where $(x \cdot y)(n) = x(n) \cdot y(n)$ *for any* $x, y \in 2^X$ *and* $n \in X$. ■

In Theorem 2.2 we have proved that to compute reliability of the system $\langle X, F \rangle$ it suffices to divide this system into parallel (series) blocks, to compute reliability of any of these blocks and to use standard formula for reliability of parallel (series) system. Here the parallel (series) subblocks form new components.

We can also identify parallel and series components inside the actual block. Introduce notations:

RELATION 2.3 $\|$ – *"parallel components"*
$n \| m$ *iff* $\{p \in \Phi_{\min} : n \in p\} = \{p \in \Phi_{\min} : m \in p\}$ *for any* $n, m \in X$.

RELATION 2.4 $\perp$ – *"series components"*
$n \perp m$ *iff* $\{p \in \Psi_{\min} : n \in p\} = \{p \in \Psi_{\min} : m \in p\}$ *for any* $n, m \in X$.

RELATION 2.5 $x/\|\in 2^{X/\|}$ – *parallel reduced reliability component,*
$x/\|: X/\| \rightarrow \{0,1\}$ *is defined by*

$$x/\| (p) = \max\{x(k) : k \in p\} \text{ for any } x \in 2^X \text{ and } p \in X/\| . \tag{17}$$

RELATION 2.6 $x/\perp\in 2^{X/\perp}$ – *series reduced reliability component,*
$x/\perp: X/\perp \rightarrow \{0,1\}$ *is defined by*

$$x/\perp (p) = \min\{x(k) : k \in p\} \text{ for any } x \in 2^X \text{ and } p \in X/\perp . \tag{18}$$

RELATION 2.7 $F_R : 2^{X/R} \rightarrow \{0,1\}$ – *reduced reliability structure function,*
$F_R(y) = F(y^R)$ *for any* $y \in 2^{X/R}$ *and* $R \in \{\|, \perp\}$, *where* $y^R(k) = y([k]_R)$ *for any* $y \in 2^{X/R}$, $k \in X$ *and* $R \in \{\|, \perp\}$,

The following theorem holds:

THEOREM 2.3

$$F_\perp(x/\perp) = F(x), \tag{19}$$

$$F_\|(x/\|) = F(x), \tag{20}$$

for any $x \in 2^X$.

Proof of Theorem 3.
By definition $F_\perp(x/\perp) = F((x/\perp)^\perp)$ and $F_\|(x/\|) = F((x/\|)^\|)$.
Notice that:

$$(x/\perp)^\perp \leq x \leq (x/\|)^\|,$$

$$\phi((x/\perp)^\perp) \supseteq \phi(x) \supseteq \phi((x/\|)^\|) \text{ and}$$

$$\psi((x/\perp)^\perp) \subseteq \psi(x) \subseteq \psi((x/\|)^\|),$$

$$\psi((x/\perp)^\perp) = \{k \in X : x(l) = 1 \text{ for any } l \in [k]_\perp\},$$

$$\phi((x/\|)^\|) = \{k \in X : x(l) = 0 \text{ for any } l \in [k]_\|\}.$$

$F((x/\perp)^\perp) = 1$ iff there exists a $p \in \Psi_{\min}$ such that $p \subseteq \phi((x/\perp)^\perp)$ and $F((x/\|)^\|) = 0$ iff there exists a $p \in \Phi_{\min}$ such that $p \subseteq \psi((x/\|)^\|)$.

By definition $[k]_{\|} \subseteq p$ if $k \in p \in \Phi_{\min}$ and analogously $[k]_{\perp} \subseteq p$ if $k \in p \in \Psi_{\min}$.

Therefore for any $p \in \Phi_{\min}$, if $k \in p \subseteq \psi(x)$ then $[k]_{\perp} \subseteq \psi(x)$ and $k \in \psi((x/\perp)^{\perp})$. Thus if $p \subseteq \psi(x)$ then $p \subseteq \psi((x/\perp)^{\perp})$. By the same argument for any $p \in \Phi_{\min}$, $p \subseteq \phi((x/\|)^{\|})$ if $p \subseteq \phi(x)$. Since $\phi((x/\|)^{\|}) \subseteq \phi(x)$ and $\psi((x/\perp)^{\perp}) \subseteq \psi(x)$,

$$F_{\perp}(x/\perp) = F(x) = F \parallel (x/\|).$$

■

In Theorem 2.3 we have proved that the system whose structure is modified by merging parallel or series components has the same reliability. Notice the following obvious properties:

PROPOSITION 2.1

1. $[k]_{\perp} \cap [k]_{\|} = \{k\}$ *for any* $k \in X$,
2. $[k]_{\approx} \cup [k]_{\amalg} = X$ *for any* $k \in X$. ■

To illustrate the defined relations consider a system described by the following minimal path and cut sets:
$X = \{1, 2, 3, 4, 5, 6\}$.
Path sets: $\{1, 3, 5, 6\}$; $\{1, 4, 5, 6\}$; $\{2, 3, 5, 6\}$; $\{2, 4, 5, 6\}$.
Cut sets: $\{1, 2\}$; $\{3, 4\}$; $\{5\}$; $\{6\}$.

Checking properties of the relations $\amalg$, $\approx$, $\|$ and $\perp$, we get that:

$$n \approx m \text{ for any } n, m \in X; \tag{21}$$

$$1 \amalg 2,\ 3 \amalg 4,\ 5 \amalg 5,\ 6 \amalg 6; \tag{22}$$

$$1 \parallel 2,\ 3 \parallel 4; \tag{23}$$

$$5 \perp 6. \tag{24}$$

From (21)-(24) we conclude that the system $\langle X, F\rangle$ has the following structure:

3. Two algorithms

In this Section two algorithms are presented which enable the analysis of reliability structure of complex systems and computing its reliability. The first algorithm is used for simplifying the system reliability structure by dividing the system components into separate blocks and/or merging parallel and series components inside nondecomposable subsystems. The second one computes reliability of the system on the basis of results obtained from the first algorithm. It should be strongly marked that all discussions in both algorithms are based

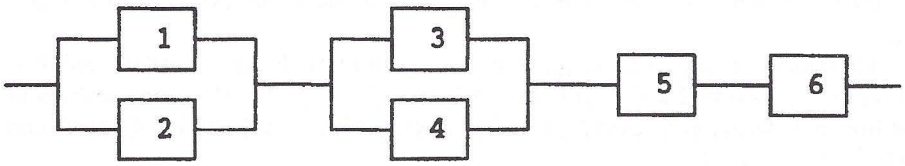

Figure 1. Structure of the system $\langle X, F\rangle$.

on the principal assumption that the reliability structure of the system is defined only by set of minimal cuts or/and set of minimal paths, without any additional information.

These algorithms are formulated here in a Pascal-like language enlarged by finite operation as sums and intersections. All array elements are written A_n instead of $A[n]$. The notation

$$for\ q \in V\ do\ operation1;$$

means: make operation *operation*1 for any element q of the set V.

To simplify and condense the algorithms let us introduce the following additional notations:

$\mathcal{L}_0(Y,T) = \{p \in T : p \subseteq Y\}$ for any $Y \in \mathcal{P}(X)$ and $T \subseteq \mathcal{P}(X)$

$\mathcal{L}_1(Y,T) = \{p \in \mathcal{P}(Y)$: there exists a $q \in T$ such that $p = q \cap Y\}$ for any $Y \in \mathcal{P}(X)$ and $T \subseteq P(X)$

$\mathcal{K}(T,U) = \{\emptyset = p \in \mathcal{P}(X)$: there exist an $r \in T$ and a $q \in U$ such that $p = q \cap r\}$ for any $T, U \subseteq \mathcal{P}(X)$

$\mathcal{B}(Y,T) = \{(k,l) \in Y^2 : \{p \in T : k \in p\} = \{p \in T : l \in p\}\}$ for any $T \subseteq \mathcal{P}(Y)$

$\mathcal{W}(T) = \{(k,l) \in X^2$: there exist two sequences $k = k_0, \ldots, k_n = l \in X$ and $p_1, \ldots, p_n \in T$ such that $\{k_{n-1}, k_n\} \subseteq p_i$ for $i = 1, 2, \ldots, n\}$, for any $T \subseteq \mathcal{P}(X)$

$H(k)$ probability that the component k is in state of functioning

$H(X,F)$ probability that the system $\langle X, F\rangle$ is in state of functioning

ALGORITHM1 – simplifying the system structure

```
Input data:  X, Φ_min, Ψ_min
Output data:  V, J, K, M
{Step 1 - dividing the system into parallel and series subsystems}
begin
      Φ_0 := Φ_min;
      Ψ_0 := Ψ_min;
      k := 1;
      IND := .true.;
      W_1 := X/W(Ψ_min);
      Φ_1 := K(W_1, Φ_0)_min;
      Ψ_1 := K(W_1, Ψ_0)_min;
      repeat
            k := k + 1;
            IND := .not.IND
            if IND then W_k := X/W(Φ_{k-1})
                  else W_k := X/W(Ψ_{k-1});
            endif
            Φ_k := K(W_k, Φ_{k-1})_min;
            Ψ_k := K(W_k, Ψ_{k-1})_min;
      until W_k = W_{k-1};
      K := k - 1;
      J := .not.IND
      V := W_K
end ;
{Step 2 - reducing the system by merging parallel and series
elements}
begin
      k := 1;
      IND := .false.;
      B_1 := B(X, Ψ_min);
      X_1 := (X/B_1)*;
      Φ_1 := L_1(X_1, Φ_K)_min;
      Ψ_1 := L_0(X_1, Ψ_K)_min;
      repeat
            k := k + 1;
            IND := .not.IND;
            if IND then
                  begin
                        B_k := B(X_{k-1}, Φ_{k-1});
                        i := 1;  j := 0
                  end
```

```
            else
                begin
                    B_k := B(X_{k-1}, Ψ_{k-1});
                    i := 0;  j := 1
                end
            endif
            X_k := (X_{k-1}/B_k)*;
            Φ_k := L_i(X_k, Φ_{k-1})_min;
            Ψ_k := L_j(X_k, Ψ_{k-1})_min;
        until X_{k-1} = X_k;
        M := k - 1;
end
```

ALGORITHM 2 – computing system reliability

```
Input data:  V, J, K, M -- output data of ALGORITHM1 - and H(k).
Output data:  H(X).
{Step 1 - computing reliability of parallel and series blocks in
nondecomposable subsystems}
begin
    IND := .false.
    for k := 1 to M do
        for n ∈ X_k do
            begin
                if IND then H(n) := 1 - ∏_{l∈[n]_{B_{k-1}}} (1 - H(l))
                else H(n) := ∏_{l∈[[n]_{B_{k-1}}} H(l);
                endif;
            end;
        endfor;
        IND = .not.IND
    endfor;
end ;
{Step 2 - computing reliability of all nondecomposable subsystems
by Poincare formula}
begin
    for k := K downto 1 do
        begin
            for q ∈ W_k do
                begin
                    Δ := L_1(q, W_k)_min;
                    H(q) := ∑_{⊆Δ}{(-1)^{card(T)+1} ∑_{p∈T}[∏_{n∈p} H(n)]};
                end;
            endfor;
        end;
```

```
    endfor;
end;
{Step 3 - computing the system reliability}
begin
    IND := J;
    for k := K - 1 downto 1 do
        begin
            IND := .not.IND;
            for p ∈ W_k do
                begin
                    if IND then
                        H(p) := 1 - ∏_{q ∈ W_{k+1}, q ⊆ p} (1 - H(q))
                    else
                        H(p) := ∏_{q ∈ W_{k+1}, q ⊆ p} (H(q))
                    endif;
                end
            endfor;
        end
    endfor;
    H(X) := H(W_1)
end.
```

4. Example

Consider the system with structure as in Figure 2:

Min-cuts:	Min-paths:
{1,3}	{1,2,4,5,6,7}
{2,3}	{3,4,5,6,7}
{4}	{1,2,4,5,6,8}
{5,7}	{1,2,4,5,6,9}
{6,7}	{3,4,5,6,8}
{5,8,9}	{3,4,5,6,9}
{6,8,9}	{1,2,4,7,8}
{7,8,9}	{1,2,4,7,9}
	{3,4,7,8}
	{3,4,7,9}

Results of ALGORITHM1

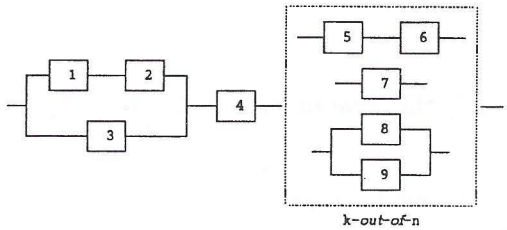

Figure 2. Reliability structure

Step 1

*Level*1

$1 \amalg 2 \amalg 3$	$\rightarrow$	Block $A = \{1, 2, 3\}$,
$4 \amalg 4$	$\rightarrow$	Block $B = \{4\}$– 1-component, nondecomposable,
$5 \amalg 6 \amalg 7 \amalg 8 \amalg 9$	$\rightarrow$	Block $C = \{5, 6, 7, 8, 9\}$.

*Level*2

$1 \approx 2$	$\rightarrow$	Block $A1 = \{1, 2\}$,
$3 \approx 3$	$\rightarrow$	Block $A2 = \{3\}$ – 1-component, nondecomposable,
$5 \approx 6 \approx 7 \approx 8 \approx 9$	$\rightarrow$	Block $C1 = \{5, 6, 7, 8, 9\}$ – 5-component, nondecomposable.

Step 2-splining parallel and series components in the subsystem C.

$5 \perp 6 \rightarrow 5/\perp$, $\qquad 8 \parallel 9 \rightarrow 8/\parallel$.

5. Conclusions

We have described theoretical aspects of a method of analyzing the system reliability structure. The most important advantage of this method is its generality which enables to analyze all coherent structure although there is no significant gain for the systems with simple reliability structures. The reliability structure of an analyzed system is coded only once and next evaluations for the system with different data are essentially accelerated. In the paper we use the Poincaré formula for explaining the principles of the algorithms for evaluating reliability of the nondecomposable subsystems. In practical implementation one can use any method, for example, the pivotal decomposition.

Practically this method has been implemented in the computer program RACE (Reliability Analysis of Complex Equipment) which enables determining various reliability factors of complex systems. In RACE the system structure is entered into the computer with the help of fault trees and then paths and cuts are determined by using the improved standard methods.

RACE is a language independent (one can create an own native language version), menu driven, user friendly professional program, very useful for reliability researchers, constructors and scientists. It has also significant didactic values. This program was used for estimating reliability of nuclear power plant, aircrafts and some telecommunication equipment. There are very good experiences with practical application of algorithms presented in this paper.

The demo version of RACE is available from the first author.

References

ABRAHAM J.A. (1979) An improved method for network reliability. *IEEE Transactions on Reliability*, R-28:58-61.

AGGARWAL K.K., MISRA K.B., GUPTA J.S. (1975) A fast algorithm for reliability evaluation. *IEEE Transactions on Reliability*, R-24:83-85.

BARLOW R.E. (1984) Reliability calculation and inductive inference: An overview. In: *Reliability Theory and Models*, M.S. Abdel-Hammed, E. Cinlar, J. Quinn (Eds). Academic Press: Orlando: 221-229.

BARLOW R.E., PROSCHAN F. (1975) *Statistical Theory of Reliability and Life Testing. Probabilistic Models.* Holt, Rinehart and Winston: New York.

BEICHELT F., SPROSS L. (1987A) An improved Abraham-method for generating disjoint sums. *IEEE Transactions on Reliability*, R-36:70-74.

BEICHELT F., SPROSS L. (1987B) An effective method for reliability analysis of complex systems. *EIK, Journal of Information Processing and Cybernetics*, 23:227-235.

BENNETTS R.G. (1975) On the analysis of fault trees. *IEEE Transactions on Reliability*, R-24:175-185.

BIRKHOFF G., BARTEE T.C. (1970) *Modern applied algebra*, McGraw-Hill, New York.

BUTTERWORTH R.W. (1972) A set theoretic treatment of coherent systems. *SIAM Journal of Applied Mathematics*, 22:590-598.

DHILLON B.S., SINGH C. (1981) *Engineering Reliability, New Techniques and Applications.* John Wiley & Sons: New York.

HARIRI S., RAGHAVENDRA C.S. (1987) SYREL: A symbolic reliability algorithm based on path and cutset methods. *IEEE Transactions on Computers*, C-36:1223-1232.

HENLEY E.J., KUMAMOTO H. (1981) *Reliability Engineering and Risk Assessment.* Prentice-Hall, New York.

JENSEN P.A., BELLMORE M. (1969) An algorithm to determine the reliability of a complex system. *IEEE Transactions on Reliability*, R-18:169-174.

JIANG MINGXIANG (1985) Theory and algorithm for the quantitative analysis of fault trees. *Reliability Engineering*, 12:241-257.

LOCKS M.O. (1978) System reliability analysis: A tutorial. *Microelectronics and Reliability*, 18:335-345.

LOCKS M.O. (1982) Recursive disjoint products: A review of three algorithms.

IEEE Transactions on Reliability, R-31:33-35.

LOCKS M.O. (1985) Recent development in computing of system reliability. *IEEE Transactions on Reliability*, R-34:425-435 No. 5, December 1985, pp.425-435.

LOCKS M.O. (1987) A minimizing algorithm for sum of disjoint products. *IEEE Transactions on Reliability*, R-36:445-453.

PAGE L.B., PERRY J.E. (1986) An algorithm for exact fault-tree probabilities without cut sets. *IEEE Transactions on Reliability*, R-35:544-558.

PAGE L.B., PERRY J.E. (1988) An algorithm for fault-tree probabilities using the factoring theorem. *Microelectronics and Reliability*, 28:273-286.

SCHNEEWEISS W.G. (1985) Fault-tree analysis using a binary decision tree. *IEEE Transactions on Reliability*, R-34:453-457.

TORREY J.A. (1983) A pruned tree approach to reliability computation. *IEEE Transactions on Reliability*, R-32:170-174.