УДК 519.711

Логические методы расчета надежности систем. Часть II. Математическая модель надежности

Левин В. И.

Актуальность. *В последние годы все большее внимание ученых и проектировщиков технических систем приобретают вопросы совершенствования методов оценки надежности и безопасности этих систем, в связи с задачами повышения значений этих характеристик.* ***Цель статьи*** *заключается в разработке автоматно-логической модели надежности технических систем и соответствующих логических методов оценки надежности таких систем, которые в отличие от известных используют не традиционные – вероятностные показатели надежности, а детерминированные логические показатели.* ***Метод.*** *Для достижения поставленной цели в статье предложено использовать в качестве исходных данных наблюдаемые моменты последовательных отказов и восстановлений элементов технической системы, а в качестве характеристик надежности самой системы – моменты последовательных отказов и восстановлений этой системы. В этом случае задача оценки надежности системы сводится к построению ее математической модели в виде автоматных логических функций, выражающих моменты ее последовательных отказов и восстановлений через аналогичные моменты всех ее элементов. Данная статья представляет собой вторую часть работы, в которой детально разрабатывается автоматно-логическая модель, предназначенная для вычисления логической функции надежности технических систем.* ***Новизна*** *работы заключается в построении адекватной логической модели надежности системы, позволяющей свести оценку надежности технической системы к вычислению ее логических функций надежности.* ***Результат****. В статье детально разработаны логическая модель надежности и методы ее исследования, позволяющие вводить новые показатели надежности технических систем, не требующие для своей оценки использования вероятностных методов и исходных статистических данных об отказах элементов. На основе разработанной логической модели надежности и методах ее исследования решена задача построения автоматной модели надежности систем, которая позволит вести практические расчеты надежности реальных технических систем методами теории динамических автоматов.*

Ключевые слова: *переключательный процесс, надежностный процесс, динамический автомат, двоичный оператор, структура оператора, логическая теория надежности.*

Введение

В первой части работы [1] было показано, что традиционные вероятностные расчеты надежности различных систем [2–5] по своей природе не элементарны и потому применительно к сложным системам достаточно трудоемки. В связи с этим была высказана идея разработки методов расчета надежности систем, в которых оперируют элементарными (первичными) величинами, характеризующими надежность системы и ее элементов, и устанавливают связь между ними. В развитие этой идеи в первой части работы [1] был описан математический аппарат новой теории и новых методов расчета надежности систем, основанный на математической логике [6, 7]. В этой статье, являющейся второй частью работы, строится и детально описывается автоматная математическая модель для оценки надежности систем.

Библиографическая ссылка на статью:
Левин В. И. Логические методы расчета надежности систем. Часть II. Математическая модель надежности // Системы управления, связи и безопасности. 2017. № 3. С. 84-97. URL: http://sccs.intelgr.com/archive/2017-03/04-Levin.pdf
Reference for citation:
Levin V. I. Logical Methods of Computing of Systems Reliability. Part II. Mathematical Model of Reliability. *Systems of Control, Communication and Security*, 2017, no. 3, pp. 84-97. Available at: http://sccs.intelgr.com/archive/2017-03/04-Levin.pdf (in Russian).

Эта модель, в сочетании с разработанным ранее логико-математическим аппаратом [1], позволяет проводить конструктивные расчеты надежности различных систем.

1. Переключательные процессы

Рассмотрим произвольную двоичную функцию непрерывного времени t, генерируемую в некоторой системе (в частности, технической системе), т. е. функцию $x = x(t)$, значения которой принадлежат множеству $\{0,1\}$. Пусть эта функция удовлетворяет трем условиям:

1) значение функции x в момент ее изменения $t = a$ по определению совпадает со значением x при $t > a$;
2) значения x определены на интервале времени $(-\infty, \infty)$;
3) на любом конечном подынтервале указанного интервала имеется конечное число изменений значения функции. Введенная функция называется *переключательным процессом* в системе.

Обозначим: 1 – постоянный процесс, равный единице на некотором интервале времени; 0 – постоянный процесс, равный нулю на некотором интервале времени; $1'$ – изменение значения процесса $0 \to 1$; $0'$ – изменение значения процесса $1 \to 0$; $0'_a$ – изменение $0'$ в момент a; $1'_a$ – изменение $1'$ в момент a; $1(a,b)$ – импульс $1'_a 0'_b$; $0(a,b)$ – пауза $0'_a 1'_b$. По условию 1 в некоторой окрестности момента a изменения значения процесса

$$1'_a = \begin{cases} 0, & t < a; \\ 1, & t \ge a; \end{cases} \quad 0'_a = \begin{cases} 1, & t < a; \\ 0, & t \ge a. \end{cases} \tag{1}$$

Согласно (1) *импульс* – это интервал единичных значений процесса, включающий начало и не включающий конец, а *пауза* – интервал нулевых значений процесса с аналогичными включениями:

$$1(a,b) = \begin{cases} 1, & a \le t < b; \\ 0, & t < a \quad \text{или} \quad t \ge b; \end{cases} \quad 0(a,b) = \begin{cases} 0, & a \le t < b; \\ 1, & t < a \quad \text{или} \quad t \ge b. \end{cases} \tag{2}$$

Формулы (2) при $a = b$ принимают вид:

$$1(a,a) \equiv 0; \quad 0(a,a) \equiv 1. \tag{3}$$

Видим, что импульс (пауза) с совмещенными началом и концом фактически есть отсутствие импульса (паузы), т. е. вырожденный участок, и может быть исключен из рассмотрения. Однако из формулы (3) следует возможность формально рассматривать отсутствие импульса (паузы), т. е. тождественный 0 (тождественную 1), как импульс (паузу) с совмещенными началом и концом, что часто бывает полезно. Отметим также возможность рассматривать изменения процесса (1) как импульс (паузу) на бесконечном интервале:

$$1'_a = 1(a,\infty) = 0(-\infty,a); \quad 0'_a = 0(a,\infty) = 1(-\infty,a). \tag{4}$$

Введем необходимые определения. Пусть $x(t)$ – любой переключательный процесс, отличный от тождественного нуля или единицы; a_x – момент первого изменения (начала) и b_x – момент последнего изменения (окончания) $x(t)$, причем оба момента конечны. Значение x_0 процесса при $t < a_x$ назовем его *начальным значением.* При этом будем говорить, что $x(t)$ *начинается* импульсом (паузой), если $x_0 = 0$ $(x_0 = 1)$. Аналогично значение x_∞ процесса при $t > b_x$ назовем его *конечным значением,* говоря, что процесс $x(t)$ *оканчивается* импульсом (паузой), если $x_\infty = 0$ $(x_\infty = 1)$. Процессы $x(t), y(t)$ назовем *непересекающимися во времени,* если $b_x \le a_y$.

Общее число изменений значения переключательного процесса называется *длиной* L *процесса.* При $L \le 1$ процесс считается *простым,* при $L \ge 2$ процесс считается *сложным.* Два переключательных процесса *равны,* если у них одинаковое число соответственно однотипных изменений, моменты которых совпадают. Два переключательных процесса с буквенными моментами изменений считаем *эквивалентными,* если при любой численной конкретизации указанных моментов оба процесса становятся равными.

Будем записывать переключательные процессы в виде последовательности изменений с указанием момента изменения или в виде последовательности импульсов и пауз. Во втором случае для простоты опускаем начальное и конечное постоянные значения, а моменты промежуточных изменений указываем один раз либо в импульсе, либо в соседней паузе. Например, один и тот же процесс можно записать либо так: $x(t) = 1'_a 0'_b 1'_c 0'_d 1'_e$, либо так:

$$x(t) = 1(a,b)0(-,c)1(-,d)0(-,e).$$

Этот процесс до момента a равен 0, в интервале $a \le t < b$ он равен 1, в интервале $b \le t < c$ равен 0, в интервале $c \le t < d$ – снова 1, в интервале $d \le t < e$ снова 0 и при $t \ge e$ принимает постоянное значение 1.

2. Двоичные операторы технических систем

Пусть имеется множество переключательных процессов $x_1(t),...,x_n(t)$, закон G, по которому это множество преобразуется в переключательный процесс $y(t)$, называется *двоичным оператором.* Таким образом,

$$y(t) = G[x_1(t),...,x_n(t)]. \quad (5)$$

В технических системах $x_1(t),...,x_n(t)$ – входные процессы, $y(t)$ – выходной процесс, а G – оператор системы. Оператор, реализующий преобразование (5), называется n-*местным,* по числу преобразуемых процессов. На операторном языке преобразуемые процессы $x_1(t),...,x_n(t)$ называются воздействиями на оператор G, а результирующий процесс $y(t)$ – *реакцией* оператора. Мы ограничимся рассмотрением операторов, удовлетворяющих следующему условию (*принцип физической осуществимости*): значение реакции $y(t)$ в любой момент времени t зависит только от значений воздействий $x_1(t_1),...x_n(t_n)$ в предшествующие $t_1,...,t_n$ или же текущий t моменты времени $(t_1 < t,...,t_n < t)$ и от значений самой реакций $y(t_*)$ в предшествующие моменты $t_*(t_* < t)$. Если зависимость $y(t)$ от $y_*(t)$ существенна, оператор называется *оператором с памятью,* если несущественна – *оператором без памяти.* Число моментов $t_{*1},...t_{*s}(t_{*i} < t)$, таких, что значение $y(t)$ существенно зависит от значений $y(t_{*1}),...,y(t_{*s})$, называется *глубиной памяти* оператора. Это число может быть как конечным, так и бесконечным. В первом случае имеем *оператор с конечной памятью,* во втором – с *бесконечной.* Оператор без памяти называется *временным,* если $y(t)$ существенно зависит от значения воздействий $x_i(t_i)$ в предшествующие моменты времени $t_i(t_i < t)$, *и логическим* – в противном случае, т. е. если $y(t)$ зависит только от значений воздействий $x_1(t),...x_n(t)$ в тот же текущий момент t. Для логического оператора зависимость (5) реакции от воздействий конкретизируется:

$$y = f(x_1,...,x_n), \quad (6)$$

где f – некоторая булева функция; $x_1,...,x_n, y$ – мгновенные значения воздействий и реакции в один и тот же произвольный момент времени t.

Двоичный оператор можно задать с помощью уравнения, связывающего значение $y(t)$ со значениями $x_1(t_1),...x_n(t_n), y(t_*)$, где $t_i \le t,\ t_* < t$, посредством алгоритма, позволяющего вычислить значения $y(t)$ для любого t, и т. д. Удобным способом задания произвольного оператора является его *структурное представление* в виде суперпозиции (схемы) из *элементарных операторов.* Элементарным считается оператор, который является простейшим и потому неделим, т. е. не представим суперпозицией более простых операторов. Удобство такого представления в том, что изучение произвольного оператора сводится к изучению существенно более простых элементарных операторов, число которых конечно.

Задачи изучения операторов технических систем можно условно разделить на три типа. *Задача анализа оператора* заключается в отыскании реакций $y(t)$ заданного оператора на заданные воздействия $x_1(t),...,x_n(t)$. *Задача синтеза оператора* состоит в построении оператора, преобразующего заданные воздействия $x_1(t),...,x_n(t)$ в требуемую реакцию $y(t)$. Под построением оператора понимается какое-нибудь конструктивное его задание – абстрактное или структурное (*абстрактный или структурный синтез*). *Задача синтеза воздействий* заключается в отыскании воздействий на оператор $x_1(t),...,x_n(t)$ по заданным оператору G и его реакции $y(t)$.

3. Элементарные операторы

Будем записывать любой переключательный процесс с неуточненным характером участков (импульсов и пауз) в виде

$$x(t)=u(a_1,a_2)\overline{u}(-,a_3)...u^{(-1)^m}(a_{m-1},a_m),\quad u\in\{0,1\},\tag{7}$$

где $\overline{u}$ – отрицание u, а

$$u^p=\begin{cases}u & \text{при } p=1;\\ \overline{u} & \text{при } p=-1.\end{cases}\tag{8}$$

Рассмотрим несколько элементарных временных операторов.

1. *Оператор* D_τ *задержки на* τ – это одноместный оператор, преобразующий воздействие $x(t)$ вида (7) в реакцию

$$y(t)=D_\tau[x(t)]=x(t-\tau)=u(a_1+\tau,a_2+\tau)\overline{u}(-,a_3+\tau)...u^{(-1)^m}(a_{m-1}+\tau,a_m+\tau),\tag{9}$$

т. е. сдвигающий входной процесс $x(t)$ на постоянное время задержки τ.

2. *Оператор* D_τ^Φ *фильтрации на* τ – одноместный оператор, преобразующий каждый импульс и паузу $u(a_i,a_{i+1})$ воздействия (7) в реакцию

$$y(t)=D_\tau^\Phi[u(a_i,a_{i+1})]=\begin{cases}u(a_i+\tau,a_{i+1}+\tau), & a_{i+1}-a_i\ge\tau;\\ \overline{u}, & a_{i+1}-a_i<\tau,\end{cases}\tag{10}$$

иными словами, сдвигающий входной процесс $x(t)$ на время τ и, кроме того, не пропускающий (фильтрующий) изменения $x(t)$, отстоящие друг от друга ближе, чем на τ.

3. *Оператор достройки паузой до* c – одноместный оператор, преобразующий воздействие $x(t)$ вида (7) в реакцию (достройка справа, $c>a_m$)

$$y(t)\equiv x_c(t)=\begin{cases}x(t), & u^{(-1)^m}=0;\\ x(t)0(a_m,c), & u^{(-1)^m}=1,\end{cases}\tag{11}$$

или в реакцию (достройка слева, $c<a_1$)

$$y(t)\equiv {}_c x(t)=\begin{cases}x(t), & u=0;\\ 0(c,a_1)x(t) & u=1.\end{cases}\tag{12}$$

4. *Оператор достройки импульсом до* c – одноместный оператор, преобразующий воздействие $x(t)$ вида (7) в реакцию (достройка справа, $c>a_m$)

$$y(t)\equiv x^c(t)=\begin{cases}x(t), & u^{(-1)^m}=1;\\ x(t)1(a_m,c), & u^{(-1)^m}=0,\end{cases}\tag{13}$$

или в реакцию (достройка слева, $c<a_1$)

$$y(t)\equiv {}^c x(t)=\begin{cases}x(t), & u=1;\\ 1(c,a_1)x(t) & u=0.\end{cases}\tag{14}$$

Операторы достройки выполняются раньше других элементарных операторов.

5. *Оператор усечения до* b – одноместный оператор, преобразующий воздействие $x(t)$ (7) в реакцию

$$y(t) \equiv x(t \wedge b) = u(a_1 \wedge b, a_2 \wedge b)\overline{u}(-, a_3 \wedge b)...u^{(-1)^m}(a_{m-1} \wedge b, a_m \wedge b) \quad (15)$$

путем взятия конъюнкции НЛ моментов изменения $x(t)$ с данным моментом b (усечение справа) или в реакцию

$$y(t) \equiv x(t \vee b) = u(a_1 \vee b, a_2 \vee b)\overline{u}(-, a_3 \vee b)...u^{(-1)^m}(a_{m-1} \vee b, a_m \vee b) \quad (16)$$

взятием дизъюнкции НЛ данных моментов (усечение слева). При этом процесс $x(t \wedge b)$ отличается от процесса $x(t)$ заменой на интервале $b < t < \infty$ всех значений $x(t)$ конечным значением. Процесс $x(t \vee b)$ отличается от $x(t)$ заменой при $-\infty < t < b$ всех значений $x(t)$ начальным значением.

6. *Оператор умножения* – двухместный оператор, преобразующий пару воздействий $x_1(t), x_2(t)$, не пересекающихся во времени $(b_{x_1} \le a_{x_2})$ и таких, что конечное значение первого процесса $x_1(t)$ совпадает с начальным значением второго $x_2(t)$, в реакцию вида

$$y(t) = \begin{cases} x_1(t), & t < b_{x_1}; \\ x_2(t), & t \ge b_{x_1}. \end{cases} \quad (17)$$

Эта реакция называется *произведением процесса* $x_1(t)$ на $x_2(t)$ и обозначается таким образом

$$y(t) = x_1(t) \circ x_2(t). \quad (18)$$

Из (17) видно, что произведение процесса $x_1(t)$ на $x_2(t)$ до момента b_{x_1} окончания $x_1(t)$ совпадает с $x_1(t)$, с момента a_{x_2} начала $x_2(t)$ совпадает с $x_2(t)$, в интервале $[b_{x_1}, a_{x_2}]$ равно конечному значению $x_1(t)$ (начальному значению $x_2(t)$). Оператор умножения подчиняется ассоциативному закону, т. е. при $b_{x_1} \le a_{x_2}$ $b_{x_2} \le a_{x_3}$

$$[x_1(t) \circ x_2(t)] \circ x_3(t) = x_1(t) \circ [x_2(t) \circ x_3(t)] = x_1(t) \circ x_2(t) \circ x_3(t), \quad (19)$$

но не подчиняется коммутативному закону, т. е. в общем случае $x_1(t) \circ x_2(t)$ не совпадает с $x_2(t) \circ x_1(t)$.

7. Оператор разбиения – одноместный оператор, который разбивает процесс $x(t)$ вида (7) на два последовательных подпроцесса:

$$\left.\begin{aligned} x_1(t) &= u(a_1, a_2)\overline{u}(-, a_3)...\widetilde{u}(a_{i-1}, a_i), \quad \text{где } \widetilde{u} = u \text{ или } \overline{u} \\ x_2(t) &= \overline{\widetilde{u}}(a_i, a_{i+1})\widetilde{u}(-, a_{i+2})...u^{(-1)^m}(a_{m-1}, a_m) \end{aligned}\right\}, \quad (20)$$

так что

$$x(t) = \begin{cases} x_1(t), & t < a_i; \\ x_2(t), & t \ge a_i. \end{cases} \quad (21)$$

Сравнение (21) с (17) показывает, что

$$x(t) = x_1(t) \circ x_2(t), \quad (22)$$

т. е. перемножение подпроцессов $x_1(t)$ и $x_2(t)$ снова дает исходный процесс $x(t)$. Потому операторы умножения и разбиения взаимно обратны. Заключительное изменение в первом подпроцессе $x_1(t)$ разбиения (20) назовем точкой деления процесса $x(t)$. Точка деления имеет вид $1'_{a_i}$ или $0'_{a_i}$.

Рассмотрим несколько элементарных логических операторов. Как сказано выше, согласно (6) такой оператор можно задать с помощью булевой функции, преобразующей мгновенное значение воздействий в любой момент t в мгновенное значение реакции, относящееся к тому же моменту.

1. Конъюнктор – это двухместный оператор, преобразующий воздействия $x_1(t)$, $x_2(t)$ в реакцию $y(t)$ согласно конъюнкции в выражении (1) в первой части работы, опубликованной в статье [1]

$$y = x_1 \wedge x_2. \tag{23}$$

2. Дизъюнктор – также двухместный оператор, преобразующий воздействия $x_1(t)$, $x_2(t)$ в реакцию $y(t)$ согласно дизъюнкции в выражении (2) в первой части работы, опубликованной в статье [1]

$$y = x_1 \vee x_2. \tag{24}$$

3. Инвертор – одноместный оператор, преобразующий воздействие $x(t)$ в реакцию $y(t)$ согласно булевой функции отрицания в выражении (3) в первой части работы, опубликованной в статье [1]

$$y = \bar{x}. \tag{25}$$

4. Дизъюнктивный инвертор (оператор Вебба) – двухместный оператор, преобразующий воздействия $x_1(t), x_2(t)$ в реакцию $y(t)$ согласно булевой функции «отрицание дизъюнкции»:

$$y = \overline{x_1 \vee x_2}. \tag{26}$$

5. Конъюнктивный инвертор (оператор Шеффера) – двухместный оператор, преобразующий воздействия $x_1(t), x_2(t)$ в реакцию $y(t)$ согласно булевой функции «отрицание конъюнкции»:

$$y = \overline{x_1 \wedge x_2}. \tag{27}$$

Дизъюнктивный и конъюнктивный инверторы, строго говоря, не могут считаться элементарными операторами, так как они являются суперпозицией операторов (23)–(25). Однако на практике оба используются часто как элементарные операторы.

4. Структурное представление операторов без памяти

Удобство структурного представления операторов (см. п. 2) делает целесообразной разработку специальной методики перехода от произвольного содержательного описания оператора к его структурному представлению, т. е. к схеме, реализующей оператор в виде суперпозиции конечного числа элементарных операторов. Такой переход включает два этапа:

1) от содержательного описания оператора к его математическому описанию;
2) от математического описания оператора к реализующей его схеме.

Первый этап неалгоритмичен и выполняется неформально, мы рассмотрим второй этап.

Реакция $y(t)$ оператора без памяти в любой момент t зависит от значений воздействий $x_1(t),..., x_n(t)$ в тот же самый момент t, а также от их значений $x_1(t_1),...,x_n(t_n)$ в некоторые предшествующие моменты $t_1,...,t_n$ (п. 2). Будем считать, что число таких предшествующих моментов для каждого воздействия конечно. Тогда зависимость реакции оператора без памяти от этих воздействий принимает вид:

$$y(t) = f[x_1(t), x_1(t_{11}),..., x_1(t_{1m_1});...; x_n(t), x_n(t_{n1}),..., x_n(t_{nm_n})], \tag{28}$$

где f – некоторая булева функция; t – текущий момент; $t_{ij}(t_{ij} < t)$ – предшествующие t моменты, значения воздействий в которых влияют на значение реакции в текущий момент t.

В частном случае, когда значения воздействий в предшествующие моменты несущественны, т. е. когда оператор логический, зависимость реакции от воздействий приобретает известный вид (6):

$$y = f(x_1,..., x_n). \tag{29}$$

Здесь $x_1,\dots,x_n,y$ – мгновенные значения воздействий и реакции, взятые в один и тот же произвольный момент времени. Формулы (28), (29) дают математическое описание двух типов оператора без памяти: временного и логического.

Начнем с задачи структурного представления логического оператора, т. е. построения схемы, реализующей булеву функцию f (29) в виде суперпозиции элементарных операторов f_i. При этом достаточно ограничиться только логическими операторами f_i. Набор f_i, позволяющий реализовать любую функцию f, называется *функционально полным* или *базисом.* Образуют базис, например, следующие наборы операторов:

1) конъюнктор и инвертор;
2) дизъюнктор и инвертор;
3) конъюнктор, дизъюнктор и инвертор;
4) оператор Вебба;
5) оператор Шеффера.

Будем реализовывать логический оператор в базисе 3. Для такой реализации нужно выполнить четыре этапа:

1) перейти от имеющегося представления оператора к соответствующей булевой функции f;
2) привести полученную функцию f к эквивалентному выражению в ДНФ или КНФ [1];
3) полученное выражение разложить по элементарным операциям – двухместным конъюнкции и дизъюнкции, используя для этого сочетательный закон, в соответствии с выражением (10) первой части работы, опубликованной в статье [1];
4) сопоставить каждой из элементарных операций – конъюнкции, дизъюнкции и отрицанию – соответствующий элементарный логический оператор.

При необходимости между этапами 2 и 3 можно выполнить этап упрощения функции f. Для этого выражение f подвергается подходящим эквивалентным преобразованиям [1].

Пример 1. Реализуем логический оператор, для которого зависимость реакции y от воздействий x_1,x_2,x_3 такова, что $y=1$ на следующих наборах воздействий: 000 и 111. Выполнение этапов расчета.

1) Функция $y=f(x_1,x_2,x_3)$ задана перечислением единичных наборов.
2) Согласно процедуре [1] ДНФ функции такова: $y=\bar{x}_1\bar{x}_2\bar{x}_3 \vee x_1x_2x_3$.
3) По сочетательному закону [1] получаем выражение $y=(\bar{x}_1\bar{x}_2)\bar{x}_3 \vee (x_1x_2)x_3$.
4) Сопоставляем отрицаниям инверторы, конъюнкциям – конъюнкторы, дизъюнкции – дизъюнктор.

В результате находим схему, реализующую оператор (рис. 1).

Итак, любой логический оператор можно представить структурно в виде логической схемы, построенной из элементарных логических операторов.

Перейдем к структурному представлению временного оператора, построив схему, реализующую зависимость (28). Введем замену:

$$\left.\begin{array}{r} x_1(t_{11})=x_{n+1}(t),\dots,x_1(t_{1m_1})=x_{n+m_1}(t) \\ x_2(t_{21})=x_{n+m_1+1}(t),\dots,x_2(t_{2m_2})=x_{n+m_1+m_2}(t) \\ \hdashline x_n(t_{n1})=x_{n+\sum_{i=1}^{n-1}m_i+1}(t),\dots,x_n(t_{nm_n})=x_{n+\sum_{i=1}^{n}m_i}(t) \end{array}\right\}. \qquad (30)$$

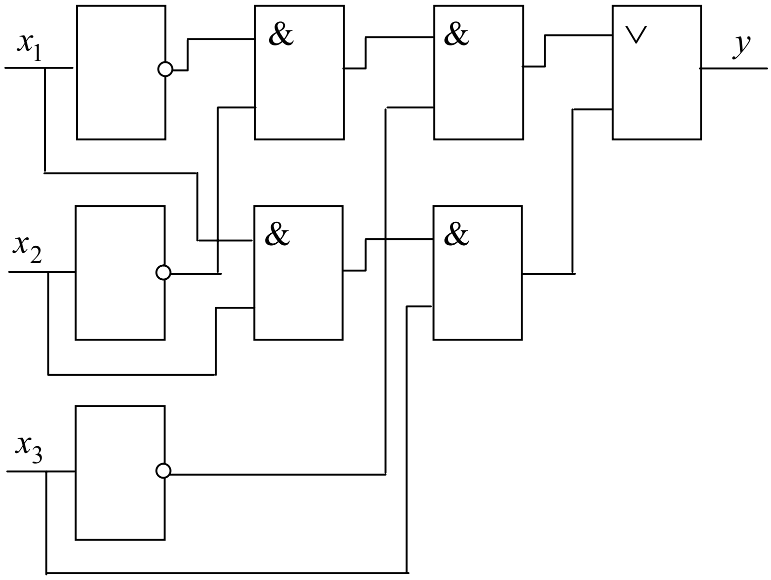

Рис. 1. Схема к примеру 1

Тогда зависимость (28) примет вид булевой функции

$$y = f\left(x_1, \ldots, x_n, x_{n+1}, \ldots, x_{n+\sum_{i=1}^{n} m_i} \right) \tag{31}$$

от расширенного множества аргументов $x_1, \ldots, x_{n+\sum_{i=1}^{n} m_i}$, в которой x_i, y – мгновенные значения воздействий и реакции, взятые в один и тот же, произвольный момент времени. Функция (31) имеет тип (29), т. е. задает некоторый логический оператор. Таким образом, структурное представление временного оператора распадается на задачу нахождения структурного представления логического оператора и задачу нахождения соотношений (30). Первая задача рассмотрена выше. Рассмотрим вторую задачу.

Обратимся, например, к первому соотношению из (30). Учитывая, что $t_{11} < t$, т. е. $t_{11} = t - \tau_{11}$, где $\tau_{11} > 0$, запишем его так: $x_{n+1}(t) = x_1(t - \tau_{11})$ или, используя оператор задержки D_τ,

$$x_{n+1}(t) = D_{\tau_{11}}[x_1(t)]. \tag{32}$$

Видим, что любое соотношение (30) реализуется при помощи оператора задержки $D_{\tau_{ij}}$ с нужным временем задержки τ_{ij}. При этом для реализации всех соотношений (30) нет необходимости использовать соответствующее число операторов задержки. Действительно, соединяя последовательно несколько операторов задержки $D_{\tau_1}, \ldots, D_{\tau_p}$, мы получаем новый оператор D_τ с суммарным временем задержки $\tau = \sum_{i=1}^{p} \tau_i$. Поэтому достаточно выбрать в качестве элементарного оператор D_τ с временем задержки τ, которое является общим делителем всех времен $\tau_{ij} = t - t_{ij}$ в (30). Тогда реализация любого соотношения (30) сведется к последовательному соединению нужного числа элементарных операторов D_τ.

Таким образом, любой временной оператор можно представить структурно в виде логической схемы, построенной из элементарных логических операторов и элементарного оператора задержки.

5. Математическая модель надежности системы

Рассмотрим произвольную *систему* (техническую, экономическую, биологическую и т. д.), состоящую из N взаимодействующих подсистем, которые назовем *блоками.* В системе имеется n *входов* и r *выходов.* По входам система получает предусмотренные условиями ее работы полезные воздействия (физические входные сигналы, задачи, подлежащие решению, управляющие команды и т. д.) или вредные воздействия (помехи, вибрация, повышенная температура, влажность и т. д.), влияющие на ее надежность, причем каждый вход предназначен для воздействий одного типа.

С выходов системы снимаются различные результаты ее работы (обработанные сигналы, решенные задачи, выполненные команды и т. д.), причем каждый выход характеризует какую-то одну функцию (один результат работы) системы.

Зададим *надежностное состояние* (НС) системы двоичным вектором

$$\boldsymbol{y} = (y_1, \ldots, y_r), \quad y_i \in \{0,1\}, \tag{33}$$

где i-я компонента y_i характеризует НС i-го выхода системы

$$y_i = \begin{cases} 1, & \text{если система } \textit{работоспособна} \text{ по } i\text{-й функции;} \\ 0, & \text{если система } \textit{неработоспособна} \text{ по } i\text{-й функции} \\ & \text{(частичный отказ } i\text{-го типа).} \end{cases} \tag{34}$$

Аналогично зададим НС совокупности блоков двоичным вектором

$$\boldsymbol{a} = (a_1, \ldots, a_N), \quad a_i \in \{0,1\}, \tag{35}$$

причем i-я компонента a_i характеризует НС i-го блока:

$$a_i = \begin{cases} 1, & \text{если } i\text{-й блок работоспособен;} \\ 0, & \text{если } i\text{-й блок отказал.} \end{cases} \tag{36}$$

Опишем НС совокупности входов системы вектором

$$\boldsymbol{x} = (x_1, \ldots, x_n), \quad x_i \in \{0,1\}, \tag{37}$$

у которого i-я компонента x_i $(i = 1, \ldots, n)$ характеризует НС i-го входа:

$$a_i = \begin{cases} 1, & \text{если система воспринимает воздействие } i\text{-го типа;} \\ 0 & \text{в противном случае.} \end{cases} \tag{38}$$

Описание входов системы при помощи двоичного вектора (37) пригодно и в более общем случае, когда существен не только факт наличия (отсутствия) воздействия каждого типа, но и значения воздействия. При этом множество возможных значений воздействия каждого типа i дискретизуется (если воздействия непрерывны) и кодируется двоичным кодом $x_{i_1}, \ldots, x_{im_i}$; последний заменяет x_i в основном коде (37).

Итак, надежностную ситуацию в системе в произвольный момент времени t можно полностью описать тройкой векторов

$$\boldsymbol{z} = (\boldsymbol{x}, \boldsymbol{a}, \boldsymbol{y}), \tag{39}$$

где $\boldsymbol{x}$ – НС входов; $\boldsymbol{a}$ – НС блоков; $\boldsymbol{y}$ – НС выходов системы в момент t. Это описание – статическое, относящееся к выбранному моменту времени. Реально все три вектора зависят от времени и надежностную эволюцию системы можно описать вектор-функцией

$$\boldsymbol{z}(t) = [\boldsymbol{x}(t), \boldsymbol{a}(t), \boldsymbol{y}(t)]. \tag{40}$$

Это описание динамическое, оно охватывает необходимый интервал времени функционирования системы.

Первая компонента (40) – вектор-функция $\boldsymbol{x}(t) = [x_1(t), \ldots, x_n(t)]$ – задает эволюцию НС входов системы, т. е. воздействия на входах рассматриваемой системы. Здесь $x_i(t)$ – двоичная функция непрерывного времени t, описывающая эволюцию НС i-го входа, т. е. воздействие на i-м входе системы, $x_i(t)$ имеет вид последовательности интервалов наличия и отсутствия i-го внешнего фактора, влияющего на надежность системы. Из физического смысла функции $x_i(t)$ следует, что она определена в любой момент бесконечного временного интервала $t\ (-\infty < t < \infty)$, причем на любом конечном подынтервале этого интервала $x_i(t)$ меняется конечное число раз. Условимся, что значение функции $x_i(t)$ в момент ее изменения $t = a$ совпадает с ее значением при $t > a$. Так, воздействия на входы системы $x_1(t),...,x_n(t)$ – некоторые переключательные процессы.

Вторая компонента в (40) – вектор-функция $\boldsymbol{a}(t) = [a_1(t),...,a_N(t)]$ – задает эволюцию НС блоков системы, причем $a_i(t)$ – двоичная функция времени, задающая эволюцию НС i-го блока в виде последовательности интервалов наличия и отсутствия работоспособности блока. Аналогично убеждаемся, что процессы надежностной эволюции блоков $a_1(t),...,a_N(t)$ – переключательные процессы. Назовем их *надежностными процессами в блоках.*

Третья компонента в (40) – вектор-функция $\boldsymbol{y}(t) = [y_1(t),..., y_r(t)]$ – описывает эволюцию НС выходов системы, т. е. эволюцию работоспособности в отношении функций системы. Здесь $y_i(t)$ – двоичный процесс, задающий эволюцию НС i-го выхода, т. е. эволюцию работоспособности системы в отношении ее i-й функции; $y_i(t)$ имеет вид последовательности интервалов выполнения и невыполнения функции. Как и выше, устанавливаем, что процессы надежностной эволюции выходов системы $y_1(t),..., y_r(t)$ – переключательные процессы. Назовем их *надежностными процессами* (*НП*) *на выходах системы.*

Итак, надежностную эволюцию в некоторой системе можно полностью описать тремя группами процессов:

1) воздействия $x_1(t),...,x_n(t)$ на n входов системы, влияющие на ее надежность;
2) НП $a_1(t),...,a_N(t)$ в N блоках системы;
3) НП $y_1(t),..., y_r(t)$ на r выходах системы, характеризующие эволюцию работоспособности в отношении r разных функций системы.

Эти группы процессов зависимы. Действительно, выполнение системой возложенных на нее функций определяется НП в блоках системы и входными воздействиями на систему. Из физических соображений следует, что выполнение системой любой i-й функции в любой момент времени t зависит только от значений НП в блоках и значений входных воздействий в тот же момент t и предшествующие моменты (и, возможно, от выполнения системой ее функций в предшествующие моменты времени). Таким образом,

$$\left.\begin{array}{l} y_1(t) = G_1[x_1(t),...,x_n(t);\ a_1(t),...,a_N(t)] \\ \text{-------------------------------------} \\ y_r(t) = G_r[x_1(t),...,x_n(t);\ a_1(t),...,a_N(t)] \end{array}\right\}, \qquad (41)$$

где $G_i\ (i = 1,...,r)$ – это некоторые двоичные операторы, удовлетворяющие принципу физической осуществимости. Эти операторы назовем собственными *надежностными операторами* (*НО*) системы. Совокупность собственных НО системы

$$\boldsymbol{G} = (G_1,..., G_r) \qquad (42)$$

является наиболее полной *надежностной* характеристикой нашей системы. Зная эту характеристику, можно из (41) вычислить НП $y_1(t),..., y_r(t)$ на выходах системы при любых заданных входных воздействиях $x_1(t),...,x_n(t)$ и НП $a_1(t),...,a_N(t)$ в блоках системы. Получаемые НП $y_1(t),..., y_r(t)$ полностью характеризуют надежность результатов работы системы. В частности, по ним можно вычислить любой *показатель надежности* (ПН) системы. Действительно, это связано с тем, что каждый ПН R представляет собой некоторый функционал F от $y_1(t),..., y_r(t)$:

$$R = F[y_1(t),..., y_r(t)]. \qquad (43)$$

Рассмотрим этот вопрос подробнее. Выбор того или иного ПН системы зависит от назначения системы и *надежностного режима* ее работы – *без восстановления* или *с восста-*

новлением отказавших блоков. Для системы без восстановления основным ПН является *наработка* T *до отказа*, определяемая как интервал времени от момента t_0 начала эксплуатации системы до ее первого отказа. Другим ПН этих систем может служить *функция готовности* $K_г(t)$, определяемая как

$$K_г(t) = \begin{cases} 1, & \text{если система в момент } t \text{ работоспособна;} \\ 0, & \text{если система в момент } t \text{ неработоспособна,} \end{cases} \tag{44}$$

и функция надежности $P(t)$:

$$P(t) = \begin{cases} 1 & \text{при отсутствии отказов на интервале } [t_0, t); \\ 0 & \text{в противном случае.} \end{cases} \tag{45}$$

Функции $P(t), K_г(t)$ невосстанавливаемой системы являются переключательными процессами одинакового вида

$$K_г(t) = P(t) = 0'_T, \tag{46}$$

так что ПН $K_г(t), P(t), T$ оказываются зависимыми. Для систем с восстановлением основными ПН служат функция готовности $K_г(t)$ (имеющая в отличие от (46) вид переключательного процесса с несколькими изменениями) и ресурс V, определяемый как интервал времени от момента t_0 начала эксплуатации системы до момента ее окончательного (невосстанавливаемого) отказа. Используется и функция надежности $P(t)$, в виде (46), а также *коэффициент готовности* $K_г$ – доля времени, в течение которого система работоспособна. Он равен

$$K_г = \frac{1}{V} \int_{t_0}^{t_0+V} K_г(t) dt, \tag{47}$$

т. е. является средним значением функции готовности $K_г(t)$ на некотором интервале $(t_0, t_0 + V)$. Часто надежность восстанавливаемой системы характеризуют *наработкой* T_i *между отказами*, определяемой как интервал времени от момента очередного i-го восстановления системы до момента следующего после него отказа, и временем i-го восстановления $T_{вi}$. Как видно из (45), (47), готовность $K_г(t)$ является первичным ПН системы, через который выражаются другие ее ПН. Отметим, что при $T_i = T$, $T_{вi} = T_в$

$$K_г = T/(T + T_в). \tag{48}$$

Вычисление ПН по соотношению (43) требует знания *критерия отказа* системы. Этот критерий зависит от назначения системы, режима эксплуатации и т. д. Если по условиям работы система должна выполнять одновременно все r своих функций, то критерием отказа системы является невыполнение хотя бы одной из этих функций (случай 1). Если система должна выполнять, по крайней мере, одну из возможных функций, то критерий отказа – невыполнение всех r функций, (случай 2). Если система должна выполнять не менее p $(1 < p < r)$ функций, безразлично каких, то критерий отказа системы – невыполнение не менее $r - p$ каких-либо функций (случай 3). Возможны и более сложные критерии отказа системы, например, учитывающие еще неравноценность различных функций системы. Знание критерия отказа системы позволяет выразить ее ПН $K_г(t)$ и $P(t)$ через НП на выходах системы $y_1(t), ..., y_r(t)$.

$$K_г(t) = y_{экв}(t) = \begin{cases} y(t) & \text{(однофункцио-нальная система)} \\ \bigwedge_{i=1}^{r} y_i(t) \quad \text{в случае 1,} & \\ \bigvee_{i=1}^{r} y_i(t) \quad \text{в случае 2,} & \text{(многофункцио-нальная система)} \\ \bigvee_{s=p}^{r} \ \bigvee_{i_1 \neq \ldots \neq i_s} [y_{i_1}(t) \ldots y_{i_s}(t)] \quad \text{в случае 3.} & \end{cases} \tag{49}$$

Здесь $\wedge$, $\vee$ – конъюнкция и дизъюнкция двузначной логики, $y_{экв}(t)$ – эквивалентный НП в системе, полученный объединением всех НП на выходах;

$$P(t) = \begin{cases} 1, & \text{если } y_{экв}(\tau) = 1 \text{ при } 0 \le \tau \le t, \\ 0 & \text{в противном случае.} \end{cases} \tag{50}$$

Таким образом вычисление различных ПН системы сводится к одной более общей задаче – определению НП на выходах системы.

Введенные выше операторные зависимости (41) НП на выходах произвольной системы от НП на ее входах и в блоках задают *надежностную модель системы*. Эта модель имеет две важные особенности:

1) работоспособность системы определяется не только работоспособностью ее блоков, но и воздействиями на ее входах;
2) работоспособность системы в любой текущий момент может зависеть от работоспособности блоков и входных воздействий не только в этот, но и в предшествующие моменты (и, возможно, от предшествующих значений работоспособности системы).

Заключение

Работа публикуется в двух частях. В первой части работы [1] был описан математический аппарат создаваемой логической теории надежности. Также были изложены дискретная и непрерывная логики и описаны вероятностные расчеты с логическими функциями от случайных аргументов.

В настоящей статье – второй части работы, на основе разработанного в [1] математического аппарата логической теории надежности решена задача создания автоматной модели надежности систем, которая дает возможность выполнять практические расчеты надежности реальных систем.

С математической точки зрения введенная надежностная модель системы в виде операторной зависимости (41) замечательна тем, что ее структурным воплощением является некоторый динамический автомат (рис. 1), входные процессы которого связаны с его выходными процессами указанной зависимостью. Таким образом, вычисление НП на выходах системы по известным НП в ее блоках и на входах сводится к хорошо известным и детально разработанным в теории динамических автоматов методам вычисления выходных процессов динамических автоматов по их входным процессам [8, 9]. Поскольку в статике в любой фиксированный момент времени выходные значения автомата связаны с его входными значениями суперпозицией операций двузначной логики, а в динамике выходные процессы автомата связаны с его входными процессами суперпозицией операций НЛ, то можно говорить, что предложенная модель и вытекающие из нее теория и методы расчета надежности систем являются логическими.

Настоящая работа продолжает цикл исследований автора по разработке математического аппарата и математических моделей логической теории надежности. От опубликованных ранее работ автора [10, 11] она отличается усовершенствованием математической модели и расчетных формул, что облегчит практическое приложение разработанных ранее методов и уменьшит вычислительную сложность расчета надежности технических систем.

Литература

1. Левин В. И. Логические методы расчета надежности систем. Часть I. Математический аппарат // Системы управления, связи и безопасности. 2017. № 2. С. 182-195.

2. Ллойд Д. К., Липов М. Надежность: организация, исследования, методы, математический аппарат. – М.: Советское радио, 1964. – 350 с.

3. Барлоу Р., Прошан Ф. Математическая теория надежности. – М.: Советское радио, 1969. – 410 с.

4. Райншке К. Модели надежности и чувствительности систем. – М.: Мир, 1979. – 454 с.

5. Левин Б. Р. Теория надежности радиотехнических систем. – М.: Советское радио, 1978 – 264 с.

6. Поспелов Д. А. Логические методы анализа и синтеза схем. – М.: Энергия, 1974. – 350 с.

7. Левин В. И. Бесконечнозначная логика в задачах кибернетики. – М.: Советское радио, 1982. – 176 с.

8. Левин В. И. Введение в динамическую теорию конечных автоматов. – Рига: Зинатне, 1975. – 376 с.

9. Левин В. И. Теория динамических автоматов. – Пенза: Изд-во Пензенского государственного технического университета, 1995. – 407 с.

10. Левин В. И. Логические методы в теории надежности. II. Математическая модель надежности // Вестник Тамбовского государственного технического университета. 2010. Т. 16. № 1. С. 119-132.

11. Левин В. И. Логическое моделирование надежности систем управления II // Известия Пензенского государственного педагогического университета имени В.Г. Белинского. 2011. № 26. С. 578-588.

References

1. Levin V. I. Evaluation of Reliability of Systems by Logical Methods. *Systems of Control, Communication and Security*, 2017, no. 2, pp. 182-195 (in Russian).

2. Lloyd D. K., Lipov M. *Nadezhnost: organizaciya, issledovaniya, metody, matematicheskiy apparat.* [Reliability: Organization, Research, Methods, Mathematical Means]. Moscow, Sovetskoe radio Publ., 1964. 350 p. (in Russian).

3. Barlow R. E., Proshan F. *Mathematical Theory of Reliability*. N.-Y., John Wiley and Sons, 1965.

4. Rainshke K. *Modeli nadezhnosti i chuvstvitelnosti system* [Models of Reliability and Sensitivity of Systems]. Moscow, Mir Publ., 1979. 454 p. (in Russian).

5. Levin B. R. *Teoriya nadezhnosti radiotehnicheskih system* [Theory of Reliability of Radiotechnical Systems]. Moscow, Sovetskoe radio Publ., 1978. 264 p. (in Russian).

6. Pospelov D. A. *Logicheskie metody analiza i sinteza skhem* [Logical Methods of Analysis snd synthesis of Schemes]. Moscow, Energiya Publ., 1974. 350 p. (in Russian).

7. Levin V. I. *Beskonechnoznachnaya logika v zadachah kibernetiki* [Continuous Logic in Problems of Cybernetics]. Moscow, Sovetskoe radio Publ., 1982. 176 p. (in Russian).

8. Levin V. I. *Vvedenie v dinamicheskuyu teoriyu kohechnyh avtomatov* [Introduction in Dynamical Theory of Finite Automata]. Riga, Zinatne Publ., 1975. 376 p. (in Russian).

9. Levin V. I. *Teoriya dinamicheskih avtomatov* [Theory of Dynamical Automata]. Penza, Penza State Technical Univ. Publ.,, 1995. 407 p. (in Russian).

10. Levin V. I. Logicheskie metody v teorii nadezhnosti. II. Matematicheskaya model nadezhnosti [Logical Methods in Theory of Reliability. II. Mathematical model of Reliability]. *Transaction of the TSTU*, 2010, vol. 16, no. 1, pp. 119-132 (in Russian).

11. Levin V. I. Logicheskoe modelirovanie nadezhnosti system upravleniya [Logical Modeling of Reliability of Control Systems]. *Izvestiya Penzenskogo gos. ped. un-ta im. V.G. Belinskogo*, 2011, no. 26, pp. 578-588. (in Russian).

Статья поступила 21 декабря 2017 г.

Информация об авторе

Левин Виталий Ильич – доктор технических наук, профессор, PhD, Full Professor. Заслуженный деятель науки РФ. Пензенский государственный технологический университет. Область научных интересов: логика; математическое моделирование в технике, экономике, социологии, истории; принятие решений; оптимизация; теория автоматов; теория надежности; распознавание; история науки; проблемы образования. E-mail: vilevin@mail.ru

Адрес: 440039, Россия, г. Пенза, пр. Байдукова/ул. Гагарина, д. 1а/11.

Logical Methods of Computing of Systems Reliability. Part II. Mathematical Model of Reliability

V. I. Levin

Relevance. *In recent years, the increasing attention of scientists and designers of communication systems has been acquiring the issues of improving methods for assessing the reliability and safety of technical systems, in connection with the tasks put forward to increase the values of these characteristics.* ***Purpose*** *of the article. is to develop an automata-logical model of the reliability of technical systems and the corresponding logical methods for assessing the reliability of such systems, using not traditional probabilistic but deterministic logical reliability indicators.* ***Method.*** *To achieve this goal, it was suggested to use as the initial data the observed moments of sequential failures and recoveries of the elements of the technical system, and as the reliability characteristics of the system itself, the moments of sequential failures and recoveries of the system. In this case, the problem of estimating the reliability of a system is reduced to constructing its mathematical model in the form of automata-logical functions expressing the moments of its sequential failures and restores through analogous moments of all its elements. In this part of article the automata-logical model is developed in detail, useful for calculate the logical function of the reliability of technical systems.* ***Novelty*** *of the work is the construction of an adequate logical model of the system's reliability, which makes it possible to reduce the reliability estimate of a technical system to the calculation of its logical reliability functions.* ***Result.*** *In this part of article the logical model of reliability and methods of its investigation are developed in detail, allowing to introduce new indicators of reliability of technical systems that do not require for their evaluation the use of probability methods and initial statistical data on element failures. In the article, based on the developed logical reliability model and methods of its investigation, the problem of constructing an automaton model of system reliability that will allow practical calculations of the reliability of real systems by methods of the theory of dynamic automata is solved.*

Keywords: *switching process, reliability process, dynamical automata, binary operator, structure of operator, logical theory of reliability.*

Information about Author

Vitaly Ilich Levin – Doctor of Technical Sciences, Full Professor. Honoured Scientist of Russia. Penza State Technological University. Field of Research: logic; mathematical modeling in technics, economics, sociology, history; decision making, optimization, recognition, automata theory, reliability theory, history of science, problems of education. E-mail: vilevin@mail.ru

Address: Russia, 440039, Penza, pr. Baydukova / Gagarin st., 1a/11.