УДК.681.142.001.4

О ЛОГИЧЕСКОМ КОНТРОЛЕ ВЫЧИСЛИТЕЛЬНЫХ УСТРОЙСТВ

А.К. Олефир
(Красноярск)

§ 1

Задачи автоматизации поиска неисправностей в электронных вычислительных устройствах (ВУ), к которым относятся специализированные цифровые устройства и машины, универсальные вычислительные машины и вычислительные системы, связаны с созданием алгоритмов анализа реакции контролируемого устройства на заданные входные процедуры. Известное внимание ряда авторов [1]-[16] к этому вопросу объясняется тем, что структура ВУ позволяет в известных пределах осуществлять локализацию неисправностей логическими методами с помощью программ, без затрат ручного труда.

В данной работе будем считать, что ВУ состоит из множества N элементов, для каждого из которых известны реализуемые функции в исправном состоянии и в состоянии устойчивого отказа элемента (ОЭ). Будем считать также, что при проведении тестовых процедур выделяется контролируемое устройство, состоящее из множества элементов $A' \subset N$, где

$$A' = \{a_1, a_2, \ldots, a_l\},$$

и контролирующие устройства, состоящие из множества элементов

$A'' \subset N$, для которых справедливо соотношение:

$$A' \cap A'' = \phi .$$

Входные процедуры по отношению к контролируемому устройству осуществляются по r входным двоичным каналам:

$$x_1, x_2, \ldots, x_r,$$

причем в реальных ВУ эти процедуры осуществляются в некотором интервале времени (t , $t + \Delta t$). Кроме того, каждая тестовая процедура предполагает наличие некоторой операции $\mathfrak{M}$, , которая разрушает в контролируемом устройстве информацию, оставшуюся от предыдущей тестовой процедуры. Таким образом, тестовая процедура m будет определена, если будут определены её составляющие:

$$m = f(x_1, x_2, \ldots, x_r, t, \mathfrak{M}) .$$

По установившейся терминологии, тестовую процедуру будем упрощенно называть набором входных переменных. Будем рассматривать устройства, для которых построена таблица неисправностей[15,16], строки которой соответствуют наборам, а столбцы – элементам контролируемого устройства. Элемент таблицы $c_{ij} = 1$ тогда и только тогда, если на наборе m_i проверяется элемент a_j. Пример таблицы неисправностей представлен в таблице I. Будем считать, что в ВУ имеется не более одного неисправного элемента одновременно.

§ 2.

Сформулируем две основные задачи логического контроля ВУ.

Установление работоспособности устройства проверкой его на всех наборах часто дает избыточную информацию. Необходимо найти минимальную совокупность наборов, которая является достаточной для проверки всего устройства. Эта минимальная совокупность наборов является минимальным контрольным тестом (МКТ) и нахождение её составляет первую задачу.

Вторая задача тестирования состоит в нахождении минимального теста для установления места неисправности. Такой тест составляется для заранее известных возможных неисправностей и называется диагностическим тестом.

Математический аппарат для построения тестов был предложен в работе [15]. В настоящей работе задачи составления минимальных контрольных и диагностических тестов формулируются и решаются с привлечением аппарата теории графов [18]. Заметим, что если объектом контроля является некоторое ВУ (контролируемое устройство), то объектом диагностики - вычислительные устройства, находящиеся на следующем, более низком по отношению к контролируемому, уровню сложности. Так, для вычислительных систем [17] объектом контроля могут быть элементарные ЭВМ, а объектом диагностики - блоки элементарной ЭВМ.

§ 3.

Обратимся к задаче нахождения минимального контрольного теста. Построим простой граф $L_1 = (M, A, \Gamma)$ (рис. I), где M есть множество наборов входных переменных $m_1, m_2, \ldots, m_n$, а Γ_{m_i} - множество всех элементов, проверяемых на наборе m_i.

Задача нахождения минимального контрольного теста формулируется следующим образом: в множестве M необходимо найти подмножество T_k наименьшей мощности такое, что для любой вершины $a_j \in A$ найдется вершина $m_i \in T_k$, такая что $a_j \in \Gamma_{m_i}$. Эта задача может быть решена с помощью алгоритма, аналогичного предложенному в работе [19]. Но прежде, чем приступить к реализации данного алгоритма, целесообразно выполнить следующие преобразования:

I. Если a_j является висячей вершиной множества A, смежную с ней вершину m_i необходимо включить в множество T_k. Вершины a_k, смежные с вершиной m_i, необходимо исключить из графа вместе с ребрами; инцидентными с вершинами a_k.

2. Если m_p и m_q - две вершины из M такие, что $\Gamma_{m_p} \subset \Gamma_{m_q}$, то можно исключить из графа вершину m_p вместе с инцидентными с ней ребрами. Действительно, в множество T_k тогда не будут включены одновременно m_p и m_q, но свойство множества T_k не будет нарушено, если заменить m_p вершиной m_q.

3. Если a_p и a_q - две вершины из A такие, что $\Gamma^{-1}_{a_p} \subset \Gamma^{-1}_{a_q}$, то можно исключить из графа вершину a_q вместе с инцидентными с ней ребрами.

После применения указанных преобразования в нашем примере получаем $m_4 \in T_k$ и преобразованный граф L_2 (рис.2), к которому ни одно из указанных преобразований применить нельзя. Применим к нему несколько измененный алгоритм [I9], который заключается в следующем: для каждой вершины $a_j \in A$ составляется дизъюнкция вида:

$$m_{j1} \vee m_{j2} \vee \ldots \vee m_{jn},$$

где

$$\{ m_{j1}, m_{j2}, \ldots, m_{jn} \} = \Gamma^{-1}_{a_j}.$$

Составленные таким образом дизъюнкции соединяем знаком конъюнкции и раскрываем скобки по правилам алгебры логики. После преобразований берем конъюнкцию наименьшей длины. Пусть эта конъюнкция есть:

$$m^{(1)} \wedge m^{(2)} \wedge \ldots \wedge m^{(k)}.$$

Тогда множество $m^{(1)}, m^{(2)}, \ldots, m^{(k)}$ и является искомым. Для графа на рис.2 такими множествами будут:

$$\{ m_2, m_5 \}, \{ m_2, m_7 \}, \{ m_5, m_7 \},$$

а минимальными контрольными тестами:

$$\{m_2, m_4, m_7\}, \{m_2, m_4, m_7\}, \{m_4, m_5, m_7\}.$$

§ 4.

Рассмотрим задачу нахождения диагностического теста.

Из рассмотрения графа $L_1 = (M, A, \Gamma)$ видно, что неисправности элементов различимы на определенных наборах входных перемённых, что можно записать следующим образом: любая пара вершин a_i и a_j различима на множестве $\Gamma^{-1}_{a_{ij}}$, где:

$$\Gamma^{-1}_{a_{ij}} = (\Gamma^{-1}_{a_i} \cup \Gamma^{-1}_{a_j}) \setminus (\Gamma^{-1}_{a_i} \cap \Gamma^{-1}_{a_j}).$$

Однако для различимости неисправности элемента a_i от неисправности элемента a_j нет необходимости брать все множество $\Gamma^{-1}_{a_{ij}}$; достаточно для каждой пары вершин a_i и a_j иметь хотя бы один набор m_p такой, что:

$$m_p \in \Gamma^{-1}_{a_i} \quad \text{и} \quad m_p \notin \Gamma^{-1}_{a_j}. \tag{I}$$

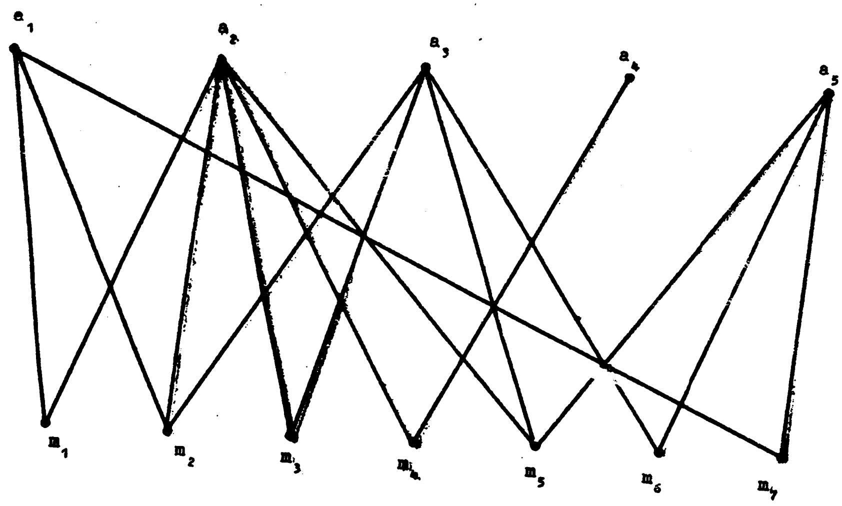

Рис. I. Граф L_1

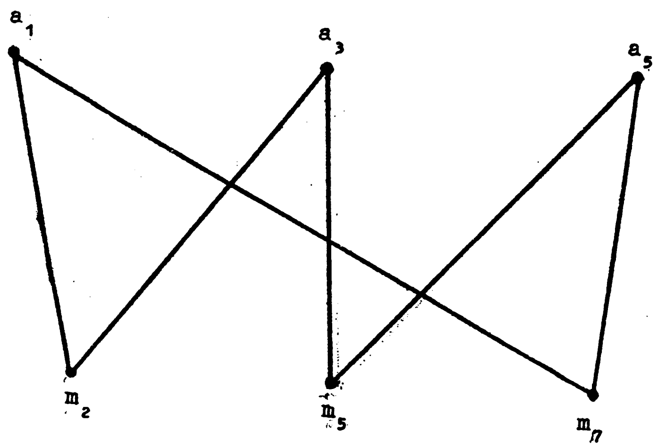

Рис. 2. Граф L_2

Задача составления минимального диагностического теста состоит в том, что из $\bigcup\limits_{i,j=1}^{l} Г^{-1}_{a_{ij}}$, где l - количество элементов, нужно выбрать множество $Т_Д$ минимальной мощности и такое, что:

$$Г^{-1}_{a_{ij}} \cap Т_Д \neq \Phi , \qquad (i = j, \; i = 1,2,\dots ;l, \; j = 1,2,\dots ,l).$$

Для решения этой задачи преобразуем исходный граф L_1 в граф $L_3 = (M^*, A^*, Г)$ следующим образом. Вместо вершин A возьмем все их парные сочетания: т.е. a_1 и a_2, a_2, и a_3 и т.д. Каждую вершину a_{ij}, полученную в результате сочетания a_i и a_j, соединим с вершинами

$$m_p^{(1)}, m_p^{(2)}, \dots , m_p^{(k)},$$

каждая из которых обладает свойством (I). Теперь задача нахождения минимального диагностического теста может формулироваться как задача нахождения в множестве M^* подмножества $Т_Д$ наименьшей мощности такого, что для любой вершины $a_{ij} \in A^*$ найдется вершина $m_k \in Т_Д$ такая, что $a_{ij} \in Г_{m_k}$. К графу L_3 можно применить преобразования и алгоритм, предложенные для нахождения минимального контрольного теста. В результате будет определена одна или несколько совокупностей наборов . Каждая совокупность будет представлять минимальный диагностический тест. Поиск отказавшего элемента будет производиться комбинационным способом [15] .

§ 5.

Метод, изложенный в предыдущих параграфах, позволяет получить минимальный диагностический тест, однако,для устройств с большим количеством элементов,не всегда нужен абсолютно минимальный тест.

Рассмотрим построение диагностического теста с использо -

ванием минимальных контрольных тестов. Основной принцип построения этого теста состоит в следующем.

На множестве M строится $T_k \subset M$, где

$$T_k = \{m^{(1)}, m^{(2)}, \ldots, m^{(k)}\}.$$

Для каждого $m^{(i)}$, где $i = 1,2,\ldots,k$, строится граф $L^{(i)}$ преобразованием исходного графа L_1 следующим образом: вычеркиваются все вершины a_j, для которых справедливо $a_j \in \Gamma_{m^{(i)}}$, и инцидентные с ними ребра, а также те вершины m_j, для которых справедливо: $\Gamma_{m_j} \cap \Gamma_{m^{(i)}} = \phi$. Каждый полученный граф $L^{(i)}$ для условного устройства с элементами $\Gamma_{m^{(i)}}$ отражает те же зависимости, что и граф L_1 для всего устройства. Для каждого условного устройства строим минимальный контрольный тест, после чего с каждым графом $L^{(i)}$ производим описанные выше операции. Эти операции повторяются до тех пор, пока при построении некоторого МКТ получим [18]:

$$\left|\Gamma_{m^{(i)}}\right| = 1 \quad \text{и} \quad a_j = \Gamma_{m^{(i)}}. \qquad (2)$$

В отличие от предыдущего диагностического теста (комбинационного), можно построить условный тест: выбор последующей тестовой процедуры будет зависеть от исхода предыдущей тестовой процедуры. Условный тест построим в виде дерева, вершинами которого будут наборы переменных, образующие минимальные контрольные тесты. Направление стрелок указывает следующую процедуру: вправо – в случае правильного результата, вниз – в случае неправильного результата. В висячих вершинах поместим элементы a_j, обладающие свойством 2. Прямоугольниками отметим входные наборы, а кружками – отказавшие элементы. Символ a_0 соответствует отсутствию отказавшего элемента.

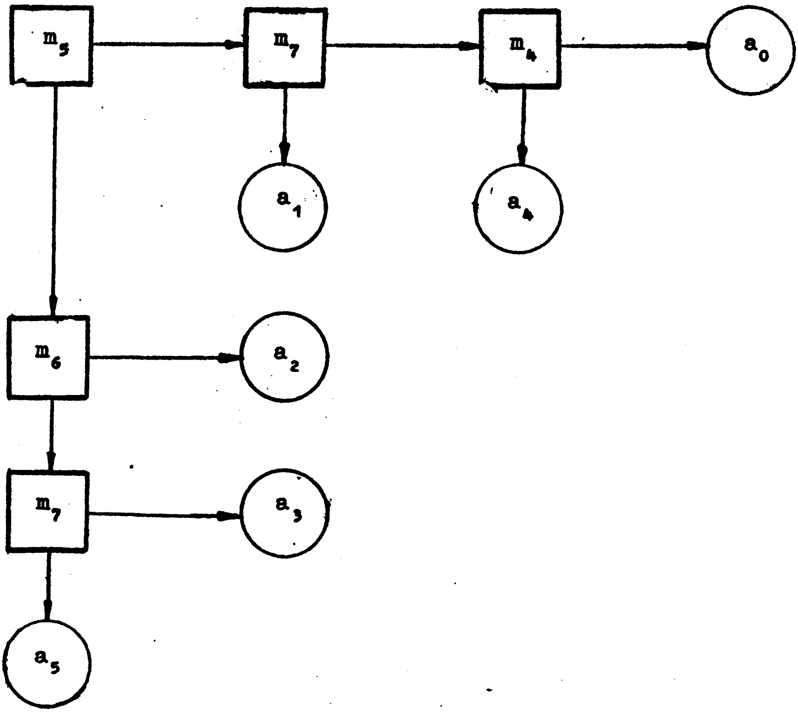

Рис. 3. Диагностический тест

Таблица

Наборы переменных \ Проверяемые элементы	a_1	a_2	a_3	a_4	a_5
m_1	I	I			
m_2	I	I	I		
m_3		I	I		
m_4		I		I	
m_5		I	I		I
m_6			I		I
m_7	I				I

§ 6.

Так как процесс преобразования графов (или таблиц) происходит по строго определенным правилам, оказалось возможным запрограммировать этот процесс на ЭВМ Урал-4.

Полученная программа составляет минимальные контрольные тесты для блока и условных блоков и печатает результаты на выходном устройстве АЦПУ. Все тесты печатаются в строку. Для указания последовательности наборов введена сквозная нумерация испытаний (в восьмеричной системе). Номер испытания x_i заключается в угловые скобки, за ним следует номер входного набора m_j, номер элемента заключается в прямоугольные скобки. Вместо стрелки вниз используется знак "НЕ". Ниже приводится пример записи диагностического теста, построенного по таблице.

$\langle x_1 \rangle : m_5 \to \langle x_2 \rangle : m_7 \to \langle x_3 \rangle : m_4 \to [a_0]$

НЕ $\langle x_1 \rangle \to \langle x_4 \rangle : m_6 \to [a_2]$

НЕ $\langle x_4 \rangle \to \langle x_5 \rangle : m_7 \to [a_3]$

НЕ $\langle x_5 \rangle \to [a_5]$

НЕ $\langle x_2 \rangle \to [a_1]$

НЕ $\langle x_3 \rangle \to [a_4]$

Приведенная запись означает, что если все испытания над наборами, записанными в первой строке, выполнились, то неисправен элемент a_0 . Если какое-либо из испытаний не выполнилось, то в левой колонне отыскивается номер этого испытания и выполняется соответствующая ему строка. Процесс продолжается до тех пор, пока какая-либо из строк не выполнится полностью.

Описываемый метод составления диагностического теста на основе минимальных контрольных тестов был опробован для устройств ЭВМ и дал положительные результаты. Основная трудность при этом заключалась в составлении таблиц неисправностей.

§ 7.

Оценим среднюю длину $s_{ср}$ диагностического теста, построенного на основе минимальных контрольных тестов. Для рассмотренного случая, когда выходы элементов из строя являются независимыми случайными событиями равной вероятности, $s_{ср}$ можно вычислить по формуле:

$$s_{ср} = \frac{1}{l+1} \sum_{i=1}^{l+1} s_i ,$$

где $l + 1$ - количество состояний (исправных и неисправных) блока;

s_i - количество испытаний до установления состояния i ;

Покажем преимущества предлагаемого метода составления тестов.

1. При составлении диагностического теста [15] необходимо преобразовать выражение в виде булевого произведения, состоящего из C^2_{l+1} членов. При составлении диагностического теста на основе минимальных контрольных тестов преобразуется булево произведение, состоящее из l членов, что сокращает затраты труда в $\frac{l+1}{2}$ раз.

2. Сравним среднюю длину диагностического теста, построенного на основе МКТ, со средней длиной диагностического теста, построенного Ченгом [12]*. Для этого построим диагностические тесты двумя методами для различных таблиц неисправностей, отличающихся друг от друга коэффициентом заполнения. Под коэффициентом заполнения таблицы $k_з$ будем понимать отноше -

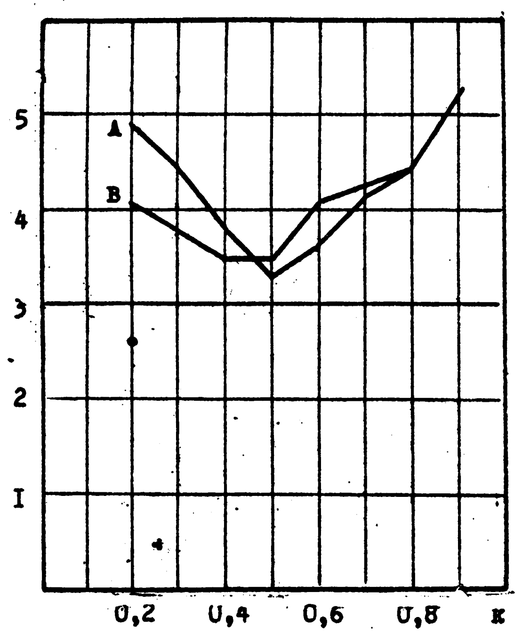

Рис. 4. Сравнительные характеристики диагностических тестов: А – по методу Ченга, Б – на основе МКТ

ние количества элементов таблицы c_{ij} = I ко всему количеству элементов таблицы. Как видно из рис.4, на котором эта зависимость показана графически, наилучшие результаты по методу Ченга получаются при k_3 = 0,5. В то же время при $k_3 \leq 0,45$ лучшие результаты получаются при построении диагностического теста с помощью МКТ, что позволяет применять его для реальных цифровых устройств.

ЛИТЕРАТУРА

I. Г.А.Миронов. Испытательные программы для контроля электронных цифровых машин. М.,"Наука",1964.

2. T.R.Bashkow, I.Friets, A.Karson. A Programming System for Detection and Diagnosis of Machine Malfunctions.-IEEE Transactions,1963,vol.EC-12,N 1.

3. A.D.Woolner. Test programs for HEC.- Computer J.,1959,vol.2, p.44-47.

4. Р.С.Гольдман. О программном контроле устройств управления цифровых вычислительных машин.- известия высших учебных заведений.Приборостроение,1966,т.9,№ 5,с.77-8I.

5. I.Cohen, L.Whitaker. Improved Techniques in Diagnostic Programming.- Sylvania Technologist,1960,N 3.

6. D.W.Liddell. Integration and automatic fault location techniques in large digital data systems. - AFIPS Proc. Spring ICC,1962,vol.21. Palo Alto,Calif.,1962,p.213-224.

7. W.R.McCormack, C.Michel. Diagnostic maiutenauce: a technique using a computer.- IEEE Trans.Aerospace,1963,vol. 1,N 2,p.931-941.

8. S.Seshu, Freeman. The Diagnosis of asynchronous sequential switching systems.- IRE Trans.,1962,vol.EC-11,N 4.

9. S.Seshu. On an Improved Diagnosis program.- IEEE Trans., 1965,vol.EC-14, N 1.

IO. А.М.Сидоров. Методы контроля электронных цифровых машин. М., "Советское радио",1966.

II. А.В.Мозгалевский, Д.В.Гаскаров, Л.П.Глазунов, В.Д.Ерастов. Автоматический поиск неисправностей. Л.,"Машиностроение",I967.

I2. H.Y.Chang. An algorithm for selecting an optimum set of diagnostic tests.- IEEE Trans.,1965,vol.EC-14, N 5.

I3. Э.И.Клямко. Схемный и тестовый контроль автоматических цифровых вычислиетльных машин.М.,"Советское радио", I963.

I4. Brule, Johnson, Kletsky. Diagnosis of equpment failures.- Trans.IRE on Rel.and Quality Control,1960,April.

I5. И.Д.Чегис, С.В.Яблонский. Логические способы контроля работы электрических схем.- Труды математического института им. Стеклова. М.,I958,т.5I.

I6. Диагностика неисправностей вычислительных машин.Сборник статей под.ред. Н.В.Паутина.М.,"Наука",I965.

I7. Э.В.Евреинов, Ю.Г.Косарев. Однородные универсальные вычислительные системы высокой производительности. Новосибирск,"Наука",Сибирское отделение,I966.

I8. К.Берж. Теория графов и её применения. М.,Иностр.лит-ра, I962.

19. M.Khaled. Sur la determination des nombres de stabilité et du nombre chromatique d'un graphe.- Comptes rendus de Academie,1959,vol.248,N 25,p.3522-3523.