

## ПОДХОД К ДИАГНОСТИРОВАНИЮ НЕОДНОРОДНЫХ ЦИФРОВЫХ СИСТЕМ СО СТРУКТУРОЙ СИММЕТРИЧНОГО ДВУДОЛЬНОГО ГРАФА

В.А. Ведешенков

Рассмотрены неоднородные цифровые системы (ЦС) со структурой симметричного двудольного графа, состоящие из процессоров, блоков памяти и блоков коммутаторов, позволяющих связывать любую пару «процессор — блок памяти». Предложен подход к диагностированию ЦС с такой структурой с точностью до отдельного компонента (процессор, блок памяти, блок коммутатора, линия связи). На время проведения диагностирования разнотипные устройства объединяются в проверяемые подсистемы одинакового состава. Рассмотрен пример диагностирования неоднородной ЦС, состоящей из четырех процессоров, трех блоков памяти и семи блоков коммутаторов.

**Ключевые слова:** неоднородные цифровые системы, симметричный двудольный граф, блок коммутатора, диагностирование, проверяемые подсистемы.

### ВВЕДЕНИЕ

Поиск новых, более эффективных структур связей сложных информационных систем показал, что симметричные двудольные графы предпочтительнее полных графов и полнодоступных двудольных графов по ряду параметров [1].

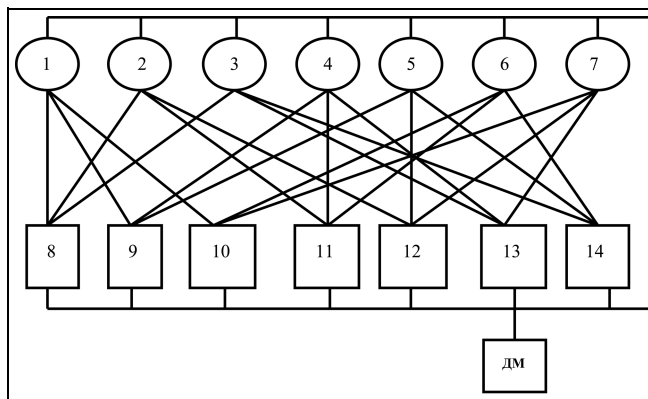
**Определение.** Двудольным графом называется граф, состоящий из двух непустых подмножеств вершин  $X$  и  $Y$  мощностью  $n$  и имеющий ребра, одна из концевых вершин которых принадлежит подмножеству  $X$  ( $Y$ ), а другая — подмножеству  $Y$  ( $X$ ). ♦

Пример такого графа для  $n = 7$  показан на рисунке, где вершины подмножества  $X$  обозначены кружками, а подмножества  $Y$  — прямоугольниками.

Среди возможных областей применения подобных графов в работе [1] названы сложные системы связи, в которых связи между источниками и приемниками информации, представленными вершинами подмножества  $X$ , осуществляются блоками локальных полнодоступных коммутаторов, представленных вершинами подмножества  $Y$ . Для сокращения в дальнейшем будем называть цифро-

вые системы (ЦС), включающие в себя блоки коммутаторов, неоднородными.

В работе [2] представлен подход к организации системного диагностирования ЦС со структурой симметричного двудольного графа, не содержа-



**Схема цифровой системы со структурой симметричного двудольного графа  $Q_{7,3}$ :**

1—7 — процессоры; 8—14 — блоки памяти; ДМ — диагностический монитор



щих блоков коммутаторов. Далее показаны особенности подхода к организации системного диагностирования неоднородных ЦС со структурой симметричного двудольного графа, включающих в себя блоки коммутаторов. В работе [3] рассматривается мультиагентная организация системного диагностирования подобных ЦС, но при других предположениях об использовании блоков памяти в диагностических процедурах.

Как и в большинстве алгоритмов системного диагностирования, будем предполагать, что результаты проверок собираются и обрабатываются (дешифрируются) внешним устройством — диагностическим монитором. По результатам дешифрации собранных результатов проверок обслуживающий персонал проводит замену отказавших компонентов и восстанавливает работоспособность ЦС.

## 1. ПОСТАНОВКА ЗАДАЧИ

Исходная неоднородная ЦС со структурой симметричного двудольного графа  $Q_{n,s}$  представлена графом, у которого  $n_1$  вершин подмножества  $X$  соответствуют процессорам,  $n_2$  вершин подмножества  $X$  — блокам памяти ( $n_1 + n_2 = n$ ), содержащим серверы памяти с необходимыми функциональными возможностями для выполнения диагностических процедур,  $n$  вершин подмножества  $Y$  — блокам коммутаторов;  $s$  — степень вершин двудольного графа, т. е. каждая вершина подмножества  $X$  ( $Y$ ) связана дугами с  $s$  вершинами подмножества  $Y$  ( $X$ ). (Для графа на рисунке  $s = 3$ .)

Допускаются устойчивые отказы ограниченного числа вершин и дуг, причем неисправности модулей таковы, что не влияют на работоспособность смежных модулей. Предполагается также, что во время выполнения процессов контроля и диагностирования (КиД) новые отказы не возникают. Запуском процессов КиД в различных подсистемах ЦС и обработкой полученных результатов занимается исправный диагностический монитор (ДМ), который находится в модуле, внешнем по отношению к анализируемой ЦС.

Для описания результатов тестирования компонентов используется модель Барси — Грандони — Маестрини [4]. При проведении КиД в каждой из формируемых проверяемых подсистем будут использованы процедуры, подобные процедурам диагностирования из работы [2].

Требуется разработать подход к организации системного диагностирования неоднородных ЦС, обеспечивающий ДМ достоверной информацией для диагностирования состояния компонентов (процессоров, блоков памяти, блоков коммутаторов, линий связи) системы.

## 2. ПОДХОД К ДИАГНОСТИРОВАНИЮ НЕОДНОРОДНЫХ ЦИФРОВЫХ СИСТЕМ СО СТРУКТУРОЙ СИММЕТРИЧНОГО ДВУДОЛЬНОГО ГРАФА

Процесс диагностирования компонентов однородных цифровых систем включает в себя [2]:

- исполнение исправным модулем функций проверяющего модуля;
- нахождение пути от одного исправного модуля к другому исправному для передачи ему функций проверяющего модуля;
- дешифрацию результатов выполненных проверок.

Как видно из данного перечня, для реализации этих процедур от проверяющего модуля не требуется выполнения сложных вычислительных и логических функций. Поэтому будем предполагать, что сервер памяти обладает функциональными возможностями, достаточными для реализации перечисленных процедур. Таким образом, можно считать, что процессор и блок памяти с сервером памяти эквивалентны в смысле реализации диагностических процедур и могут в полном объеме исполнять функции проверяющего модуля. (Поэтому в дальнейшем основное функциональное назначение этих блоков (процессора или блока памяти) отмечать не будем.)

Контроль работоспособности цифровой системы заключается в том, что каждый проверяющий модуль должен проверить техническое состояние своих приемников и соответствующих линий связи. Простейший способ выполнения проверок одного модуля разными проверяющими модулями — последовательный. Для оптимальной реализации этого способа на диагностическом графе ЦС нужно построить гамильтонов цикл, который обеспечит однократный обход всех модулей исправной системы.

В цикле, выбранном для последующего примера (см. рисунок), модули подмножества  $X$  чередуются с модулями подмножества  $Y$  (блоками коммутаторов):

$$1 \rightarrow 8 \rightarrow 2 \rightarrow 11 \rightarrow 6 \rightarrow 10 \rightarrow 7 \rightarrow 12 \rightarrow 5 \rightarrow \\ \rightarrow 14 \rightarrow 3 \rightarrow 13 \rightarrow 4 \rightarrow 9 \rightarrow 1.$$

Проверяющим модулем  $m_j$  назначим один из модулей подмножества  $X$  (вершины  $j = 1, \dots, 7$ ), а в проверяемую подсистему  $U_{j,g,k}$  включим следующие компоненты: блок коммутатора  $c_g$  ( $g = 8, \dots, 14$ ), модуль  $m_k$  подмножества  $X$  ( $k = 1, \dots, 7$ ), линию связи  $l_{j-g}$  между проверяющим модулем  $m_j$  и блоком коммутатора  $c_g$  и линию связи  $l_{g-k}$  между блоком коммутатора  $c_g$  и проверяемым модулем  $m_k$ .

(Таким образом, индекс  $(j, g, k)$  подсистемы  $U_{j,g,k}$  состоит из номера  $j$  проверяющего модуля, номера  $g$  блока коммутатора, номера  $k$  проверяемого модуля).

В таблице  $T_S$  (табл. 1), построенной для системы  $Q_{7,3}$  (см. рисунок), показана информация о составе проверяемых подсистем и номерах проверяющих модулей  $m_j$ . В табл. 1 семь строк  $T_j$ , в каждой из которых для проверяющего модуля  $m_j$  ( $j = 1, 2, \dots, 3, 4$ ) записаны номера компонентов, входящих в состав трех проверяемых им подсистем  $U_{j,g1,k1}$ ,  $U_{j,g2,k2}$  и  $U_{j,g3,k3}$ , в ней также есть поля для записи оценок результатов проверки  $r_{j,g1,k1}$ ,  $r_{j,g2,k2}$  и  $r_{j,g3,k3}$  этих подсистем модулем  $m_j$ .

Для отображения местонахождения проверяющих модулей  $m_j$  в цикле контроля в строке  $T_j$  табл. 1 проверяемые подсистемы записаны в таком порядке, что именно модуль  $m_{k1}$ , входящий в проверяемую подсистему  $U_{j,g1,k1}$ , оказывается в построенном цикле соседним справа для проверяющего модуля  $m_j$  (после блока коммутатора  $c_{g1}$ ). Например, в строке  $T_1$  табл. 1 модуль  $m_1$  проверяющий для подсистем  $U_{1,8,2}$ ,  $U_{1,9,5}$  и  $U_{1,10,6}$ , и в следующей строке  $T_2$  в качестве проверяющего записан модуль  $m_2$ , являющийся в цикле контроля соседом справа для модуля  $m_1$  (после блока коммутатора  $c_8$ ).

Тогда отдельная проверка (тест  $t_{j,g,k}$ ) будет выполняться следующим образом. Получив управление (в процессе КиД), проверяющий модуль  $m_j$  через блок коммутатора  $c_g$  и линии  $l_{j-g}$ ,  $l_{g-k}$  передает в модуль  $m_k$  команду на проведение его самоконтроля. Результаты выполненного теста  $t_{j,g,k}$  из мо-

дуля  $m_k$  передаются через блок коммутатора  $c_g$  и линии  $l_{j-g}$ ,  $l_{g-k}$  в процессор  $m_j$ , который сравнивает их с эталонами из блока собственной сверхоперативной памяти и формирует оценку  $r_{j,g,k}$ : равную 0, если результаты совпали с эталонами, равную 1 — в противном случае. На основании полученных оценок процессор  $m_j$  выбирает среди проверенных им исправную подсистему  $U_{j,g,k^*}$ , модулю  $m_{k^*}$  из состава которой передает управление и необходимую информацию для продолжения процесса КиД. Такая процедура возможна, если в системе ограниченное число отказавших компонентов. Модуль  $m_{k^*}$ , получив разрешающую команду, перейдет в режим продолжения проводимого процесса, а после завершения своего этапа проверок передаст управление следующему модулю одной из исправных подсистем, проверенных им.

Известно, что одиночные отказы вершины и прилегающей к ней линии связи неразличимы. Обычно считают, что отказ модуля, представленного вершиной графа, более вероятен, чем отказ линии связи, представленной дугой, и восстановление нужно начинать с замены модуля, подозреваемого в отказе, на исправный. Если при повторной проверке для этой пары компонентов будет получен тот же результат, то нужно восстанавливать линию связи.

Таким образом, для достоверного восстановления кратных отказов в неоднородных ЦС нужно соблюдать такой порядок диагностирования:

- запуск КиД, дешифрация результатов проверки: если обнаружены подозреваемые в отказе модули подмножеств  $X$  и  $Y$  — их диагностирование и восстановление;
- запуск КиД, дешифрация результатов проверки: если обнаружены подозреваемые в отказе

Таблица 1

 Таблица  $T_S$  состава проверяемых подсистем

Строка $T_j$	Проверяющий модуль	Проверяемые подсистемы $U_{j,g,k}$ , оценки результатов их проверок					
	$m_j$	$U_{j,g1,k1}$	$r_{j,g1,k1}$	$U_{j,g2,k2}$	$r_{j,g2,k2}$	$U_{j,g3,k3}$	$r_{j,g3,k3}$
$T_1$	1	(1-8), 8, (8-2), 2		(1-9), 9, (9-5), 5		(1-10), 10, (10-6), 6	
$T_2$	2	(2-11), 11, (11-6), 6		(2-8), 8, (8-3), 3		(2-12), 12, (12-7), 7	
$T_6$	6	(6-10), 10, (10-7), 7		(6-11), 11, (11-4), 4		(6-14), 14, (14-5), 5	
$T_7$	7	(7-12), 12, (12-5), 5		(7-10), 10, (10-1), 1		(7-13), 13, (13-3), 3	
$T_5$	5	(5-14), 14, (14-3), 3		(5-12), 12, (12-2), 2		(5-9), 9, (9-4), 4	
$T_3$	3	(3-13), 13, (13-4), 4		(3-14), 14, (14-6), 6		(3-8), 8, (8-1), 1	
$T_4$	4	(4-9), 9, (9-1), 1		(4-13), 13, (13-7), 7		(4-11), 11, (11-2), 2	



линии связи — их диагностирование и восстановление;

— если ранее были обнаружены и восстановлены отказавшие компоненты, то запуск КиД, дешифрация результатов проверок — для подтверждения исправности ЦС.

Поскольку задачи, возлагаемые на системы диагностирования рассматриваемых неоднородных ЦС и однородных ЦС из работы [2], во многом сходны, то в организации системы диагностирования рассматриваемых ЦС можно использовать большинство диагностических процедур из работы [2]. Поэтому в последующем описании организации диагностирования рассматриваемых ЦС будем опускать некоторые подробности, заменяя их необходимыми ссылками на работу [2].

Традиционно процесс контроля запускается диагностическим монитором (ДМ) через сравнительно небольшие интервалы времени, что препятствует накоплению отказавших компонентов в системе.

Назначим в качестве начального модуля процесса контроля один из модулей подмножества  $X$ , обозначим его  $m_n$ .

По сигналу ДМ о начале контроля модуль  $m_n$  выполняет следующие действия:

— из таблицы  $T_S$  выделяет строку  $T_n$ , переписывает ее в таблицу контроля  $PT_0$ , находит в ней индексы  $(n, g1, k1)$ ,  $(n, g2, k2)$ ,  $(n, g3, k3)$  проверяемых подсистем  $U_{n,g1,k1}$ ,  $U_{n,g2,k2}$ ,  $U_{n,g3,k3}$ , передает модулю  $m_{k1}$  ( $m_{k2}$ ,  $m_{k3}$ ) команду «Выполнить самоконтроль модуля  $m_{k1}$  ( $m_{k2}$ ,  $m_{k3}$ ) и передать результаты проверок в модуль  $m_n$ ».

Получив эту команду, модуль  $m_{k1}$  ( $m_{k2}$ ,  $m_{k3}$ ) проводит самоконтроль и передает результаты проверок в модуль  $m_n$ .

Модуль  $m_n$  формирует оценку  $r_{n,g1,k1}$  ( $r_{n,g2,k2}$ , ...), равную 0, если полученные результаты проверок совпали с эталонами, и равную 1 — при несовпадении; записывает ее в соответствующие клетки строки  $T_n$  в табл.  $PT_0$ , затем анализирует полученные оценки:

— если среди полученных оценок есть как единичные, так и нулевые, то передает модулю одной из подсистем с нулевой оценкой управление для продолжения процесса контроля;

— если проверки всех проверявшихся подсистем дали единичные оценки, то ДМ должен запустить процесс контроля с другого проверяющего модуля.

Модуль  $m_{k1}$  ( $m_{k2}$ , ...) и каждый из последующих «соседей» по циклу контроля выполняют аналогичные действия в случае получения нулевых оце-

нок. Этот процесс проверок и «хождения» по вершинам выбранного цикла заканчивается, когда управление возвращается к начальному модулю  $m_n$ . Если же на одном из этапов контроля (кроме начального) получена хотя бы одна единичная оценка, свидетельствующая об обнаружении одного или нескольких отказавших компонентов, то процесс контроля заканчивается, и система переходит к диагностированию отказавших компонентов.

Процесс диагностирования начинает тот модуль  $m_j$ , который первым получил хотя бы одну единичную оценку при проверке состояний связанных с ним подсистем  $U_{j,g1,k1}$ ,  $U_{j,g2,k2}$ , ...; естественно, модуль  $m_j$  должен быть исправным и проверенным в данном «сеансе» контроля, в частности, он не может быть начальным модулем  $m_n$ .

На основе информации, полученной в процессе контроля, начальный модуль процесса диагностирования  $m_j$ :

— формирует промежуточную таблицу  $PT_j$ , переписывая в нее из таблицы  $PT_0$  строку  $T_j$  с оценками, полученными в процессе контроля (в индексе символа данной таблицы указан индекс проверяющего модуля  $m_j$ );

— по нулевым оценкам в строке  $T_j$  выделяет подмножество индексов исправных модулей  $\{q_j\}$  и одному из этих модулей (например, первому в строке  $T_j$ , если он исправен) передает таблицу  $PT_j$  и управление процессом диагностирования.

Если среди модулей, проверенных модулем  $m_j$ , нет ни одного исправного, то для получения диагностических результатов ДМ должен повторить запуск процесса контроля с другого начального модуля.

Очередной исправный модуль  $m_v$ , получив управление процессом диагностирования, выполняет следующие действия:

1) из таблицы состава  $T_S$  выделяет строку  $T_v$ , переписывает ее в таблицу  $PT_j$ , полученную от предшественника — модуля  $m_j$  ( $m_{v-1}$ );

2) в строке  $T_v$  выделяет индексы  $(v, g1, k1)$ ,  $(v, g2, k2)$ ,  $(v, g3, k3)$  проверяемых подсистем  $U_{v,g1,k1}$ ,  $U_{v,g2,k2}$ ,  $U_{v,g3,k3}$ , выдает команды на самоконтроль модулей, входящих в эти подсистемы, получает результаты проведенных проверок и в клетки  $r_{v,g1,k1}$ ,  $(r_{v,g2,k2}, \dots)$ , ... строки  $T_v$  записывает двоичные оценки этих результатов;

3) по нулевым оценкам проверок в строке  $T_v$  выделяет подмножество  $\{q_v\}$  индексов исправных «преемников» — проверенных модулей, сравнива-

ет индексы из подмножества  $\{q_v\}$  с индексами строк в таблице  $PT_j$ :

— если один индекс (или больше) из подмножества  $\{q_v\}$  отсутствует в списке индексов строк в таблице  $PT_j$ , то передает модулю с этим индексом таблицу  $PT_j$  и управление процессом диагностирования;

— если список  $\{q_v\}$  пуст или все индексы  $\{q_v\}$  — в списке индексов строк в таблице  $PT_j$ , то переход к п. 4;

4) находит один из исправных модулей, еще не исполнявших функции проверяющего, и передает ему таблицу  $PT_j$  и управление диагностированием.

Для нахождения одного из исправных модулей, еще не исполнявших функций проверяющего модуля, модуль  $m_v$  находит индексы  $\{q_{v*}\}$  исправных модулей строки  $T_{v*}$  и сравнивает их с индексами строк в таблице  $PT_j$ . Если при сравнении списков индексов окажется, что:

— один индекс (или больше) из подмножества  $\{q_{v*}\}$  отсутствует в списке индексов строк в таблице  $PT_j$ , то модуль  $m_v$  передает модулю с этим индексом  $(v + 1)$  таблицу  $PT_j$  и управление диагностированием;

— список  $\{q_{v*}\}$  пуст или все индексы  $\{q_{v*}\}$  — в списке индексов строк в таблице  $PT_j$ , то модуль  $m_v$  переходит к анализу предыдущей строки  $T_{v*-1}$ .

Эти действия проводятся поочередно снизу вверх, начиная с нижней строки  $T_v$ . При нахождении исправного модуля  $m_{v+1}$  в строке  $T_{v*}$  управление диагностированием и таблица  $PT_j$  передаются этому модулю снизу вверх через проверяющие модули  $m_v, \dots, m_{v*}$ . Если при достижении верхней строки  $T_j$  окажется, что и в этой строке нет исправного модуля, не исполнявшего функций проверяющего модуля, то поиск прекращается, и модуль  $m_v$  пересылает в ДМ таблицу  $PT_j$  для обработки полученных результатов проверок.

Процедура обработки полученных результатов проверок изложена в работе [2]. В данном тексте эта процедура будет показана в примере.

### 3. ПРИМЕР

Особенности предлагаемой организации диагностирования покажем на примере ЦС со структурой симметричного двудольного графа (система  $Q_{7,3}$ , см. рисунок).

Как отмечалось, для организации контроля выбран следующий цикл, в котором модули подмножества  $X$

чередуются с блоками коммутатора:  $1 \rightarrow 8 \rightarrow 2 \rightarrow 11 \rightarrow 6 \rightarrow 10 \rightarrow 7 \rightarrow 12 \rightarrow 5 \rightarrow 14 \rightarrow 3 \rightarrow 13 \rightarrow 4 \rightarrow 9 \rightarrow 1$ .

Таблица состава  $T_S$ , построенная с учетом этого цикла, приведена в табл. 1.

Предположим, что из-за большого интервала между циклами контроля в рассматриваемой ЦС накопились отказы модулей 5, 10, 11 и линии связи (7–13).

По сигналу ДМ о начале контроля начальный модуль 1 выполнит следующие действия: из таблицы  $T_S$  (см. табл. 1) выделит строку  $T_1$ , переписет ее в таблицу  $PT_0$ , найдет в ней индексы (1, 8, 2) (1, 9, 5), (1, 10, 6) проверяемых подсистем  $U_{1,8,2}$ ,  $U_{1,9,5}$ ,  $U_{1,10,6}$ , передаст модулю 2 (5, 6) команду «Выполнить самоконтроль модуля 2 (5, 6) и передать результаты проверок в модуль 1».

Получив эту команду, модуль 2 (5, 6) проводит самоконтроль и передает полученные результаты проверки модулю 1.

Модуль 1 формирует оценки  $r_{1,8,2} = 0$ ,  $r_{1,9,5} = 1$ ,  $r_{1,10,6} = 1$ , записывает их в соответствующие клетки строки  $T_1$  в таблице  $PT_0$  (для сокращения числа таблиц будем записывать результаты проверок, находящиеся в разных модулях, в общую табл. 2), анализирует полученные оценки: так как среди полученных оценок есть две единичные и одна нулевая, то передает модулю 2 управление для продолжения процесса контроля. (Отметим, что в строке  $T_1$  и в последующих полужирным шрифтом в результатах проверки выделен 0 того исправного модуля, которому передается управление для продолжения процесса контроля или диагностирования.)

Модуль 2 выполнит аналогичные действия:

— передаст в подсистемы  $U_{2,11,6}$ ,  $U_{2,8,3}$ ,  $U_{2,12,7}$  команду «Выполнить самоконтроль и передать результаты проверок в модуль 2»;

— получит оценки результатов контроля  $r_{2,11,6} = 1$ ,  $r_{2,8,3} = 0$ ,  $r_{2,12,7} = 0$ , проанализирует их: так как среди полученных оценок есть единичная ( $r_{2,11,6} = 1$ ), то заведомо исправный модуль 2 перейдет к процессу диагностирования отказавших компонентов и выполнит следующие действия:

— образует промежуточную таблицу  $PT_2$  и переписет в нее строку  $T_2$  с полученными оценками результатов проверок;

— по нулевым оценкам в строке  $T_2$  выделит индекс исправной подсистемы  $U_{2,8,3}$ , найдет номер исправного модуля  $\{q_4 = 3\}$ , которому передаст таблицу  $PT_2$  и управление процессом диагностирования.

В свою очередь, модуль 3 выполнит следующие действия:

— из таблицы  $T_S$  (см. табл. 1) выделит строку  $T_3$ , переписет ее в таблицу  $PT_2$ , в строке  $T_3$  найдет индексы проверяемых подсистем  $U_{3,13,4}$ ,  $U_{3,14,6}$ ,  $U_{3,8,1}$ , и в модули 4, 6, 1 передаст команду «Выполнить самоконтроль и вернуть результаты проверок в модуль 3»;

— получит оценки результатов контроля  $r_{3,13,4} = 0$ ,  $r_{3,14,6} = 0$ ,  $r_{3,8,1} = 0$ ; по нулевым оценкам в строке  $T_3$





найдет номера исправных модулей  $\{q_3 = 4, 6, 1\}$  и первому из них (модулю 4) передаст таблицу  $PT_2$  и управление процессом диагностирования.

Модуль 4 выполнит аналогичные действия, запишет оценки результатов проверки в строку  $T_4$  и передаст таблицу  $PT_2$  и управление процессом диагностирования модулю 1.

Модуль 1 вместо проверки подсистем  $U_{1,8,2}$ ,  $U_{1,9,5}$ ,  $U_{1,10,6}$  выберет из строки  $T_1$  результаты их проверок (на этапе контроля)  $r_{1,8,2} = 0$ ,  $r_{1,9,5} = 1$ ,  $r_{1,10,6} = 1$  и определит, что исправен модуль 2. Но так как модуль 2 уже есть в числе проверяющих модулей на этапе диагностирования, то модуль 1 должен искать исправный модуль для исполнения функций проверяющего модуля среди ранее проверенных. В данном случае в строке  $T_3$  есть нулевые оценки проверок модулей 6 и 1. Таким образом, модуль 1 через модули 9, 4, 13, 3 и 14 передаст модулю 6 таблицу  $PT_2$  и управление дальнейшим процессом диагностирования.

Модуль 6 выполнит необходимые проверки подсистем  $U_{6,10,7}$ ,  $U_{6,11,4}$ ,  $U_{6,14,5}$ , определит, что среди его «преемников» нет исправных модулей, и аналогично модулю 1 проведет поиск проверяющего модуля среди ранее проверенных исправных модулей. Таким окажется модуль 7 в строке  $T_2$ , и модуль 6 через цепочку модулей 11, 2 и 12 передаст модулю 7 таблицу  $PT_2$  и управление дальнейшим процессом диагностирования.

В свою очередь, модуль 7 после проверок своих «преемников» — подсистем  $U_{7,12,5}$ ,  $U_{7,10,1}$ ,  $U_{7,13,3}$  выяснит, что эти системы подозреваются в отказе. Поиск исправного модуля для исполнения функций проверяющего модуля не даст положительного результата: все исправные модули уже исполняли подобные функции. Поэтому модуль 7 закончит процесс диагностирования и перешлет в ДМ таблицу  $PT_2$  для обработки полученных оценок результатов проверок.

Для обработки и дешифрации оценок результатов ДМ переписывает их из таблицы  $PT_2$  в заранее заготовленную диагностическую таблицу  $T_D$  [2]. В табл. 3 и 4 представлены две части таблицы  $T_D$  после обработки результатов из таблицы  $PT_2$  (табл. 2). Поясним структуру этих таблиц. В клетках левого столбца  $T_D$  перечислены компоненты проверяемых подсистем  $U_{j,g,k}$ , взятые из соответствующих клеток таблицы состава  $T_S$  (см. табл. 1), а клетки следующего столбца предназначены для записи оценок  $r_{j,g,k}$  проверок этих подсистем, полученных в процессе диагностирования. Последующие столбцы  $T_D$  обозначены символами одиночных компонентов системы  $Q_{7,3}$ : модули подмножества  $X$  символами  $u$  с номерами 1–7, блоки коммутатора символами  $c$  с номерами 8–14, линии связи — символами  $l$  с номерами начальной и конечной вершин. Нетрудно заметить, что в столбце  $U_{j,g,k}$  проверяемых подсистем каждая линия показана дважды: в прямом и инверсном направлении, например  $(1-8)$  и  $(8-1)$ ,  $(1-10)$  и  $(10-1)$  и т. д. Однако в списке отдельных компонентов каждая из линий записана один раз для сокращения ширины табл. 3 и 4.

Вхождение компонентов  $(l_{j-g})$ ,  $c_g$ ,  $(l_{g-k})$ ,  $u_k$  в состав подсистемы  $U_{j,g,k}$ , приведенной в левом столбце заготовленной таблицы  $T_D$ , отмечается знаком \*, поставленным в клетку на пересечении столбца с символом одного из компонентов  $(l_{j-g})$ ,  $c_g$ ,  $(l_{g-k})$ ,  $u_k$  и строки  $U_{j,g,k}$ . (Так как в табл. 3 и 4 показаны результаты обработки информации из таблицы  $PT_2$ , то знаки \* остались только в тех трех строках, для которых нет оценок проверки этих подсистем.)

В нижней части таблицы  $T_D$  находятся три строки, озаглавленные (в левом столбце) символами  $A_0$ ,  $A_1$ ,  $A_Z$  и предназначенные для записи результатов обработки полученной информации.

Таблица 2

Информация к Примеру

Строка $T_j$	Проверяющий модуль	Проверяемые подсистемы $U_{j,g,k}$ , оценки их проверок					
	$m_j$	$U_{j,g1,k1}$	$r_{j,g1,k1}$	$U_{j,g2,k2}$	$r_{j,g2,k2}$	$U_{j,g3,k3}$	$r_{j,g3,k3}$
Таблица $PT_0$							
$T_1$	1	(1-8), 8, (8-2), 2	0	(1-9), 9, (9-5), 5	1	(1-10), 10, (10-6), 6	1
$T_2$	2	(2-11), 11, (11-6), 6	1	(2-8), 8, (8-3), 3	0	(2-12), 12, (12-7), 7	0
Таблица $PT_2$							
$T_2$	2	(2-11), 11, (11-6), 6	1	(2-8), 8, (8-3), 3	0	(2-12), 12, (12-7), 7	0
$T_3$	3	(3-13), 13, (13-4), 4	0	(3-14), 14, (14-6), 6	0	(3-8), 8, (8-1), 1	0
$T_4$	4	(4-9), 9, (9-1), 1	0	(4-13), 13, (13-7), 7	1	(4-11), 11, (11-2), 2	1
$T_1$	1	(1-8), 8, (8-2), 2	0	(1-9), 9, (9-5), 5	1	(1-10), 10, (10-6), 6	1
$T_6$	6	(6-10), 10, (10-7), 7	1	(6-11), 11, (11-4), 4	1	(6-14), 14, (14-5), 5	1
$T_7$	7	(7-12), 12, (12-5), 5	1	(7-10), 10, (10-1), 1	1	(7-13), 13, (13-3), 3	1

Обработка записей из таблицы  $PT_2$  заключается в следующем:

1) из строк  $T_2, T_3, \dots, T_7$  таблицы  $PT_2$  выделены оценки результатов проверок  $r_{2,11,6} = 1, r_{2,8,3} = 0, r_{2,12,7} = 0, \dots, r_{7,12,5} = 1, r_{7,10,1} = 1, r_{7,13,3} = 1$  и записаны в соответствующие клетки столбца  $r_{j,g,k}$  табл. 3 и 4, три клетки этого столбца заполнены прочерками (—); в этих строках и сохранились знаки \*;

2) значения 0 и 1, записанные в столбце  $r_{j,g,k}$ , переписаны в клетки той же строки  $U_{j,g,k}$ , отмеченные знаком \* (заменили эти знаки);

3) заполнены прочерками (—) клетки строк  $A_0, A_1, A_2$ ;

4) после этого в клетки этих строк записаны результаты диагностирования в соответствии со следующими правилами:

а) в  $k$ -ю клетку строки  $A_0$  записан 0, если в вышележащих клетках этого столбца есть хотя бы один нуль;

Таблица 3

 Диагностическая таблица  $T_D$  (часть 1)

Проверяемые подсистемы $U_{j,g,k}$	$r_{j,g,k}$	Компоненты системы																	
		$u5$	$l5-14$	$c14$	$l14-6$	$u6$	$l14-3$	$l9-4$	$l5-9$	$c9$	$u3$	$c12$	$l9-1$	$u1$	$u4$	$l5-12$	$l12-2$	$l12-7$	$u2$
(1-8), 8, (8-2), 2	0																		0
(1-9), 9, (9-5), 5	1	1							1	1			1						
(1-10), 10, (10-6), 6	1					1													
(2-11), 11, (11-6), 6	1					1													
(2-8), 8, (8-3), 3	0										0								
(2-12), 12, (12-7), 7	0											0					0	0	
(6-10), 10, (10-7), 7	1																		
(6-11), 11, (11-4), 4	1														1				
(6-14), 14, (14-5), 5	1	1	1	1	1														
(7-12), 12, (12-5), 5	1	1										1				1		1	
(7-10), 10, (10-1), 1	1													1					
(7-13), 13, (13-3), 3	1										1								
(5-14), 14, (14-3), 3	—		*	*				*			*								
(5-12), 12, (12-2), 2	—											*				*	*		*
(5-9), 9, (9-4), 4	—							*	*	*					*				
(3-13), 13, (13-4), 4	0														0				
(3-14), 14, (14-6), 6	0			0	0	0	0												
(3-8), 8, (8-1), 1	0													0					
(4-9), 9, (9-1), 1	0							0		0			0	0					
(4-13), 13, (13-7), 7	1																		
(4-11), 11, (11-2), 2	1																		1
$A_0$	—	—	—	0	0	0	0	0	—	0	0	0	0	0	—	0	0	0	0
$A_1$	1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
$A_2$	—	Z	—	—	—	—	—	—	Z	—	—	—	—	—	Z	—	—	—	—



б) в клетки на пересечении строки  $A_1$  и столбцов, озаглавленных символами вершин  $u5, c10, c11$  записаны единицы, так как в вышележащих клетках этих столбцов есть хотя бы одна единица и нет ни одного нуля;

в) в клетку на пересечении строки  $A_1$  и столбца, озаглавленного символом дуги  $l(13-7)$ , записана единица, так как в этом столбце есть две единицы, нет ни

одного нуля и вершины 13 и 7, которые связывает эта линия, исправны (в соответствующих клетках строки  $A_0$  записаны нули);

г) в 9 клеток строки  $A_Z$ , озаглавленных символами дуг  $(l_{j-g}), (l_{g-k})$ , записан символ  $Z$ , так как хотя бы одна из вершин ( $j$  или  $k$ ), которые связывает каждая из этих линий, подозревается в отказе (в соответствующей клетке строки  $A_1$  записана 1).

Таблица 4

Диагностическая таблица  $T_D$  (часть 2)

Проверяемые подсистемы $U_{j,g,k}$	$r_{j,g,k}$	Компоненты системы																
		$l6-10$	$l1-8$	$u7$	$l1-10$	$l10-7$	$l6-11$	$l11-2$	$l11-4$	$l3-13$	$l8-3$	$c11$	$l8-2$	$c8$	$c13$	$l13-7$	$c10$	$l13-4$
(1-8), 8, (8-2), 2	0		0										0	0				
(1-9), 9, (9-5), 5	1																	
(1-10), 10, (10-6), 6	1	1			1												1	
(2-11), 11, (11-6), 6	1					1	1						1					
(2-8), 8, (8-3), 3	0											0	0	0				
(2-12), 12, (12-7), 7	0			0														
(6-10), 10, (10-7), 7	1	1		1		1											1	
(6-11), 11, (11-4), 4	1					1		1					1					
(6-14), 14, (14-5), 5	1																	
(7-12), 12, (12-5), 5	1																	
(7-10), 10, (10-1), 1	1				1	1											1	
(7-13), 13, (13-3), 3	1									1					1	1		
(5-14), 14, (14-3), 3	—																	
(5-12), 12, (12-2), 2	—																	
(5-9), 9, (9-4), 4	—																	
(3-13), 13, (13-4), 4	0									0					0			0
(3-14), 14, (14-6), 6	0																	
(3-8), 8, (8-1), 1	0		0									0		0				
(4-9), 9, (9-1), 1	0																	
(4-13), 13, (13-7), 7	1			1											1	1		1
(4-11), 11, (11-2), 2	1						1	1					1					
$A_0$		—	0	0	—	—	—	—	—	0	0	—	0	0	0	—	—	0
$A_1$		—	—	—	—	—	—	—	—	—	—	1	—	—	—	1	1	—
$A_Z$		Z	—	—	Z	Z	Z	Z	Z	—	—	—	—	—	—	—	—	—



Расшифровка полученных записей в строках  $A_0$ ,  $A_1$  и  $A_2$  дает следующие диагностические заключения:

а) исправны 22 компонента, номера которых определяются местоположением нулевых записей в строке  $A_0$ ;

б) в соответствии с местоположением единиц в строке  $A_1$  подозреваются в отказе модуль 5, блоки коммутатора 10, 11, линия связи (13–7), что совпадает с номерами компонентов в исходной отказовой ситуации;

в) в соответствии с местоположением символов Z в строке  $A_2$  остались неопределенными состояния линий (5–14), (5–9), (5–12), (6–10), (1–10), (10–7), (6–11), (11–2), (11–4), так как их проявления «маскируются» неисправным состоянием одной из вершин (5, 10, 11), к которым они примыкают.

Таким образом, в данном примере идентифицированы состояния 26 из 35 компонентов системы  $Q_{7,3}$ .

### ЗАКЛЮЧЕНИЕ

Представлен подход к диагностированию неоднородных цифровых систем со структурой симметричного двудольного графа, состоящих из процессоров, блоков памяти и блоков коммутаторов, позволяющих связывать любую пару «процессор — блок памяти». На время проведения диагностирования разнотипные устройства объединяются в проверяемые подсистемы, состоящие из блока коммутатора, двух прилегающих линий связи и процессора или блока памяти. Предложен подход к диагностированию цифровых систем с такой структурой с точностью до отдельного компонента (процессор, блок памяти, блок коммутатора, линия связи).

Предполагается, что в системе есть диагностический монитор, который инициирует процессы контроля и диагностирования в анализируемой системе.

Рассмотрен пример организации диагностирования отказавших компонентов в системе, состоящей из 7 модулей подмножества X (процессоров и блоков памяти) и 7 блоков коммутаторов, с точностью до отдельного компонента ЦС: процессор, блок памяти, блок коммутатора, линия связи между ними. В приведенном примере получен правильный результат: диагноз совпадает с исходной

отказовой ситуацией, включающей 4 компонента. Но можно подобрать такие сочетания неисправных компонентов, диагноз которых даст неправильный результат. Так, при диагностировании любой пары отказавших модулей подмножества X в число подозреваемых в отказе войдет блок коммутатора, соединяющий эту пару отказавших модулей. Структура симметричного двудольного графа характерна тем, что для любой пары модулей подмножества X (Y) обязательно найдется модуль подмножества Y (X), их соединяющий. Таким образом, нетрудно сформировать 21 пару отказавших модулей из подмножества X, диагностирование которых даст заведомо неправильный результат. Этот факт говорит о том, что рассматриваемая структура не гарантирует правильного диагностирования двух любых отказавших компонентов, т. е. не является 2-однократно диагностируемой. Причина — в «слабой» разветвленности выходов блоков коммутаторов: с каждым выходом связаны только две линии.

### ЛИТЕРАТУРА

1. *Каравай М.Ф., Пархоменко П.П., Подлазов В.С.* Комбинаторные методы построения двудольных однородных избыточных квазиполносвязных графов (симметричных блок-схем) // Автоматика и телемеханика. — 2009. — № 2. — С. 153–170.
2. *Ведешенков В.А.* Организация диагностирования цифровых систем со структурой симметричного двудольного графа // Проблемы управления. — 2009. — № 6. — С. 59–67.
3. *Ведешенков В.А.* Подход к мультиагентной организации системного диагностирования цифровых систем со структурой симметричного двудольного графа // Автоматика и телемеханика. — 2009. — № 11. — С. 161–171.
4. *Barsi F., Grandoni F., Maestrini P.* A theory of diagnosability of digital systems // IEEE Trans. Comput. — 1976. — Vol. C-25. — N 6. — P. 585–593.

*Статья представлена к публикации членом редколлегии чл.-корр. РАН П.П. Пархоменко.*

**Ведешенков Виктор Алексеевич** — д-р техн. наук, гл. науч. сотрудник, ст. науч. сотрудник, Институт проблем управления им. В.А. Трапезникова РАН, г. Москва, ☎(495) 334-75-90, ✉vva@ipu.ru.

### Читайте в следующем номере

- ✓ **Задорожный В.Н.** Случайные графы с нелинейным правилом предпочтительного связывания
- ✓ **Акопов А.С.** К вопросу проектирования интеллектуальных систем управления сложными организационными структурами
  - ✓ **Угольницкий Г.А., Усов А.Б.** Управление устойчивым развитием иерархических систем в условиях корпорации
  - ✓ **Токмачев М.С.** Здоровье населения региона: модели и управление
  - ✓ **Фархадов М.П., Петухова Н.В., Ефросинин Д.В., Семенова О.В.** Двухфазная модель с неограниченными очередями для расчета характеристик и оптимизации речевых порталов самообслуживания

