

# СПОСОБ ВЫДЕЛЕНИЯ ПОДСИСТЕМ ДОСТАТОЧНОГО РАЗМЕРА ДЛЯ ПАРАЛЛЕЛЬНОГО ДИАГНОСТИРОВАНИЯ БОЛЬШИХ ЦИФРОВЫХ СИСТЕМ С РЕГУЛЯРНОЙ СТРУКТУРОЙ

В. А. Ведешенков

*Институт проблем управления им. В. А. Трапезникова РАН, г. Москва*

Предложен способ выделения подсистем достаточного размера для параллельного диагностирования больших цифровых систем со структурой типа тороидальной решетки, в соответствии с которым контроль и диагностирование компонентов системы осуществляются в три этапа: на первом этапе параллельно проверяются компоненты всех подсистем, выделенных первичной раскладкой; на втором и третьем этапах также параллельно проверяются линии связи, примыкающие к узлам каждой подсистемы справа и снизу. Показано, что для сокращения общего времени контроля системы целесообразно выделять подсистемы, состоящие из 9-ти узлов и расположенные в виде квадрата ( $3 \times 3$ ).

## ВВЕДЕНИЕ

Задачи трехмерного моделирования во времени процессов гидроаэродинамики, уравнений теплопроводности и некоторые другие относятся к числу наиболее трудоемких в вычислительном отношении. Для них типичны требования к производительности в  $10^8$ – $10^{10}$  операций в секунду и к емкости памяти в  $10^8$ – $10^{10}$  слов [1]. Для решения таких задач широко применяются методы трехмерных сеток с локальными межузловыми связями, отличающиеся регулярностью получаемых алгоритмов и хорошими возможностями их широкого распараллеливания. Для реализации подобных алгоритмов разрабатывают проблемно-ориентированные вычислительные системы (ВС) с параллельной архитектурой, обеспечивающие сверхвысокую производительность на задачах своего класса при относительно невысокой стоимости. Примером такой ВС, разработанной для решения задач обтекания, может служить специализированная матричная система, архитектура которой показана на рис. 1 [2]. Решающее поле ВС содержит  $128 \times 128 = 16\,384$  процессорных элемента (ПЭ), образующих матричную двумерную конфигурацию со связями каждого ПЭ с восемью соседними (показаны на рис. 1 стрелками). Кроме матрицы ПЭ, система содержит управляющий процессор (УП), массовую память (МП) и 128 шин. Каждая из этих

шин соединяет УП и МП; при этом к ней подключены 128 ПЭ соответствующего столбца матрицы ПЭ. Каждый ПЭ состоит из процессора обработки данных, памяти и интерфейса, который осуществляет связь данного ПЭ с соседними ПЭ и через общую шину с УП и МП системы.

Каждый ПЭ системы (см. рис. 1) связан с восемью соседними ПЭ, что приводит к образованию так называемой решетки с октагонально связанными ПЭ [1]. Более известна двумерная решетка с ортогональными связями, в которой каждый ПЭ связан с четырьмя соседними (рис. 2). Как видно из этого рисунка, крайние элементы имеют меньшее число связей по сравнению с внутренними элементами, т. е. на краях решетки регулярность связей нарушается. Для исключения такого эффекта нужно соединить каждый крайний элемент с соответствующим ему крайним элементом на противоположной стороне решетки, чтобы образовать так называемую тороидальную решетку (рис. 3), в которой каждый элемент имеет локальные связи с четырьмя ближайшими соседями.

Процессор обработки данных каждого ПЭ рассчитан на производительность в 1 млн операций с плавающей запятой в секунду. Расчеты показывают, что при такой производительности отдельного ПЭ решение варианта задачи обтекания на сетке  $128 \times 128 \times 128$  с 50 000 временными шагами на приведенной ВС потребовало бы

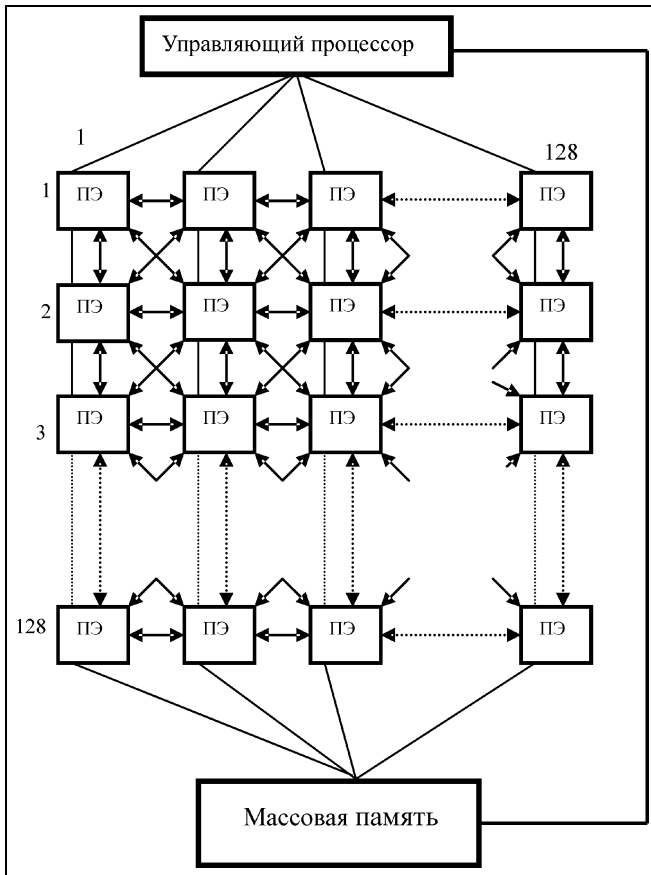


Рис. 1. Архитектура матричной системы

18 ч счета. Понятно, что отказ даже одного из 16 384 ПЭ матричной системы в течение времени счета может привести к искажению результатов решаемой задачи. Для обеспечения отказоустойчивости системы в матрицу ПЭ добавлена одна строка резервных ПЭ с обычными связями и введено несколько дополнительных связей между ПЭ одного столбца [3]. В системе организована система диагностирования с точностью до одного ПЭ. По результатам диагностирования отказавший ПЭ блокируется по всем его связям и логически исключается из соответствующего столбца, а связи резервного ПЭ в этом же столбце активизируются, благодаря чему он вводится в состав рабочей конфигурации. Для обеспечения восстановления того состояния системы, которое было до отказа, текущее состояние системы периодически запоминается, например, в конце каждого временного шага.

Цифровыми системами в системном диагностировании называются модели многомашинных и многопроцессорных вычислительных систем, состоящие из модулей и связей между ними, отражающих структуру и состав этих систем в деталях, необходимых для решения задач диагностирования. В частности, модули, на которые разбивается исходная ВС, обладают функциональными возможностями, достаточными для реализации диагностических алгоритмов, составными частями которых являются процедуры тестирования компонентов системы и сбора результатов этих проверок. Напомним,

что процедура тестирования заключается в следующем: проверяющий модуль подает на входы проверяемого модуля последовательность проверок (тест), получает ответные реакции проверяемого модуля, сравнивает их с эталонными и оценивает результат сравнения в двоичной форме: 0 — результаты правильные, 1 — неправильные. Для описания результатов тестирования модулей, зависящих от технических состояний проверяющего и проверяемого модулей, в данной работе используется модель Барси—Грандони—Маэстрини (сокращенно БГМ) [4]. В модели БГМ предполагается, что оценка результатов теста всегда достоверна и правильно идентифицирует состояние проверяемого модуля, если исправен проверяющий модуль; эта оценка равна 1, если неисправны оба модуля, и недостоверна (или 0, или 1) в том случае, когда неисправен проверяющий модуль, а проверяемый — исправен. Поэтому нулевая оценка, выдаваемая проверяющим модулем, с большой вероятностью подтверждает исправность проверенного им модуля. Такая предпосылка модели БГМ существенно упрощает процедуру диагностирования, так как для идентификации исправных модулей достаточно 0-путей единичной длины (в терминологии статьи [5]).

Любой метод системного диагностирования основан на том, что вводится ограничение на максимальное число произвольно расположенных отказавших компонентов, которое позволяет оставшимся исправным модулям получить и сформировать информацию, достаточную

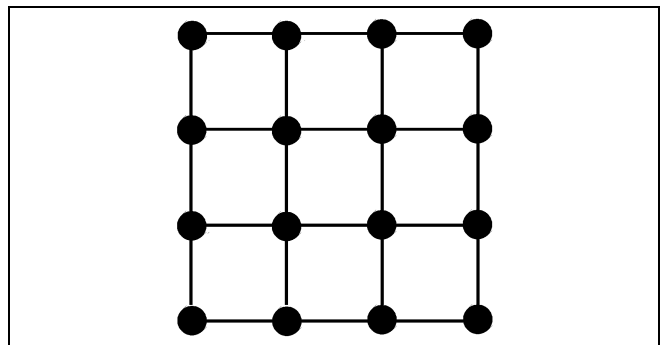


Рис. 2. Схема двумерной решетки с ортогональными связями

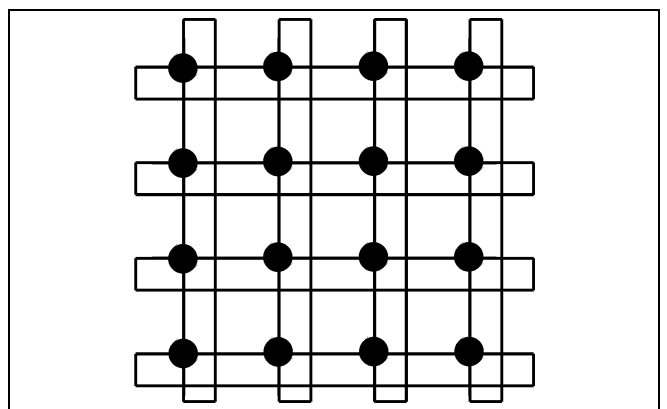


Рис. 3. Схема двумерной тороидальной решетки

для правильного диагностирования технического состояния компонентов системы, прежде всего, ее отказавших компонентов. Это ограничение называется мерой диагностируемости системы. В работе [4] доказано, что для  $t$ -однократно диагностируемых систем из  $N$  модулей мера диагностируемости  $p_{\text{БГМ}}$  отказавших модулей (при исправных линиях связи) для модели БГМ определяется соотношением

$$p_{\text{БГМ}} \leq \min[p, (N - 2)], \quad (1)$$

где  $p < (N - 1)$  — наименьшее число входящих дуг у одной из вершин графа. Возможные отказы линий связи понижают эту оценку и затрудняют процесс диагностирования.

При системном диагностировании больших цифровых систем — моделей многопроцессорных ВС с регулярной структурой, содержащих сотни и тысячи процессоров — необходимо организовать параллельное выполнение процедур системного диагностирования в различных частях (подсистемах) исходных анализируемых систем. Отметим, что к настоящему времени опубликовано небольшое число работ, в которых анализируются большие цифровые системы.

Один из возможных подходов к решению этой задачи заключается в разделении исходной большой цифровой системы на подсистемы небольшого размера, в параллельном выполнении необходимых диагностических процедур в каждой из этих подсистем и последующем объединении частных результатов, полученных в каждой из подсистем, в общий диагноз. В работе [6] представлен похожий подход к диагностированию процессорных решеток. Размер выделенной подсистемы зависит от требуемой точности диагностирования, свойств диагностического графа системы и степени сложности диагностических процедур, исполняемых в отдельной подсистеме. В данной работе предполагается использование процедур, подобных изложенным в статьях [7–9]. Их применение позволяет существенно повысить достоверность получаемых оценок результатов тестирования, так как функции проверяющих модулей исполняют исправные модули, и упростить объединение частных диагнозов в общий диагноз.

Далее представлен и обоснован способ выделения подсистем небольшого размера, достаточного для параллельного диагностирования больших цифровых систем с регулярной структурой типа тороидальной решетки, показано, что для сокращения общего времени контроля системы целесообразно выделять подсистемы, состоящие из девяти модулей.

## 1. ПОСТАНОВКА ЗАДАЧИ

Пусть большая анализируемая система с регулярной структурой представлена диагностическим графом вида тороидальной решетки размера  $(N \times N)$ , вершины которого отображают модули (процессоры) системы, причем каждый из них связан дугами с  $p = 4$  соседними вершинами, тогда  $L = pN^2/2 = 2N^2$  ненаправленных дуг между вершинами отображают линии связи между соответствующими модулями (процессорами).

Будем считать, что допускаются устойчивые отказы как вершин, так и дуг. Суммарное число отказавших

компонентов ограничено. Для описания результатов тестирования модулей, зависящих от технических состояний проверяющего и проверяемого модулей, в работе используется модель БГМ. Предполагается, что для проведения контроля и диагностирования в каждой выделенной подсистеме будут применены процедуры самодиагностирования, подобные изложенным в статьях [7, 8]. Предполагается также, что запуском и организацией параллельного диагностирования в различных подсистемах занимается диагностический монитор (ДМ), который находится в процессоре, внешнем по отношению к решетке ПЭ (например, в УП для матричной системы, см. рис. 1).

Требуется разработать способ выделения на графе большой цифровой системы с регулярной структурой подсистем такого размера  $(n \times n)$ , который достаточен для выполнения процедур контроля и диагностирования состояния компонентов этих подсистем.

## 2. СПОСОБ ВЫДЕЛЕНИЯ ПОДСИСТЕМ ДОСТАТОЧНОГО РАЗМЕРА НА ГРАФЕ БОЛЬШОЙ ЦИФРОВОЙ СИСТЕМЫ СО СТРУКТУРОЙ ТОРОИДАЛЬНОЙ РЕШЕТКИ

Как отмечалось во Введении, достаточный размер подсистемы зависит от требуемой точности диагностирования, свойств диагностического графа системы и степени сложности диагностических процедур, исполняемых в отдельной подсистеме. Остановимся подробнее на этих положениях.

Точность диагностирования — это число подозреваемых компонентов, идентифицированных в результате выполнения диагностических процедур как одно неразличимое подмножество. Точность диагностирования зависит от числа неисправных компонентов, находящихся в системе во время диагностирования, ее меры диагностируемости  $p_{\text{БГМ}}$  и выбранного метода диагностирования. По соотношению (1) нетрудно подсчитать, что  $p_{\text{БГМ}} = 1$  при  $p = 1$  и  $N = 3$ , т. е. в этом случае диагностический граф системы будет иметь вид кольца из трех вершин, каждая из которых связана одной дугой с двумя соседними вершинами. Но такой кольцевой граф с любым числом вершин не позволяет различить отказ вершины от отказа входящей дуги; нужен граф, в котором  $p \geq 2$ .

Наименьшим фрагментом тороидальной решетки (см. рис. 3) является квадрат  $(2 \times 2)$ , т. е. вновь кольцевой граф, в котором нельзя достоверно различить отказы вершины и входящей дуги. Выход — в объединении смежных квадратов для увеличения числа входных дуг у общих вершин этих квадратов и обеспечения различимости отказов этих вершин и их входных дуг. Такое объединение нужно сделать до начала процесса диагностирования, а именно, выделить в качестве начального фрагмента для организации диагностирования квадрат  $(3 \times 3)$ , включающий 9 вершин и 12 дуг. Подсчет числа входных дуг в таком графе показывает, что для центральной вершины  $p = 4$ , четыре смежные с ней вершины имеют  $p = 3$ , а для четырех угловых вершин  $p = 2$ . Такие характеристики числа входных дуг обеспечивают потенциальные возможности для различимости отказов вершин и их входных дуг.



Следующая особенность, которую нужно учесть при разделении большой цифровой системы на подсистемы достаточного объема, состоит в требовании обеспечения независимости исполнения диагностических процедур в разных подсистемах. Для обеспечения такой независимости нужно, чтобы различные подсистемы не имели общих компонентов (ни вершин, ни дуг). В этом случае исполнение процедур в одной подсистеме не будет мешать исполнению процедур в другой подсистеме и наоборот, т. е. возможно их независимое параллельное исполнение. Этому требованию об отсутствии общих компонентов в разных подсистемах, в том числе в смежных подсистемах, можно удовлетворить следующим образом: будем раскладывать на вершины графа исходной системы вершины графов (квадратов) подсистем достаточного размера вплотную один к другому. Назовем такое размещение подсистем первичной раскладкой. При таком размещении вершины графов смежных подсистем будут разделены дугами, выходящими из одной подсистемы и входящими в смежную подсистему. Эти дуги не входят ни в одну из смежных подсистем, и таким образом смежные подсистемы не будут иметь общих компонентов.

Но эти вход-выходные дуги смежных подсистем не будут и проверяться при выбранной раскладке подсистем. А их проверка необходима, так как в постановке задачи допускается возможность отказов не только модулей (вершин графа), но и линий связи (дуг графа). Для обеспечения проверки и диагностирования дуг, соединяющих вершины смежных подсистем первичной раскладки, сделаем следующее. Сдвинем границы первичной раскладки подсистем на графе исходной системы на один столбец вправо. Наличие дуг, соединяющих вершины правого столбца (см. рис. 3) тороидальной решетки с вершинами ее левого столбца, делает такой сдвиг реализуемым для всех подсистем. Тогда все дуги, соединявшие правые вершины любой подсистемы с соответствующими левыми вершинами правой смежной подсистемы, окажутся в составе первой подсистемы после сдвига ее границ вправо на один столбец. Проверки этих дуг вместе с вершинами, которые они соединяют, определяют содержание второго этапа контроля и диагностирования исходной системы. После его выполнения еще раз изменим раскладку подсистем, а именно, сдвинем границы первичной раскладки подсистем на графе исходной системы на одну строку вниз. Наличие дуг, соединяющих вершины нижней строки (см. рис. 3) тороидальной решетки с вершинами ее верхней строки, делает такой сдвиг также реализуемым. Теперь все дуги, соединявшие нижние вершины любой подсистемы с соответствующими верхними вершинами смежной нижней подсистемы, попадут в состав первой подсистемы после сдвига ее границ вниз на одну строку. Проверки этих дуг вместе с вершинами, которые они соединяют, определяют содержание третьего этапа контроля и диагностирования исходной системы.

Для иллюстрации сказанного на рис. 4 изображен фрагмент тороидальной решетки, где вершины показаны зачерненными кружками, а соединяющие их дуги — линиями различной толщины. Числами от 0 до 14 обозначены вершины, расположенные в правом верхнем углу того квадрата, где показано одно из этих чисел. Квадрат,

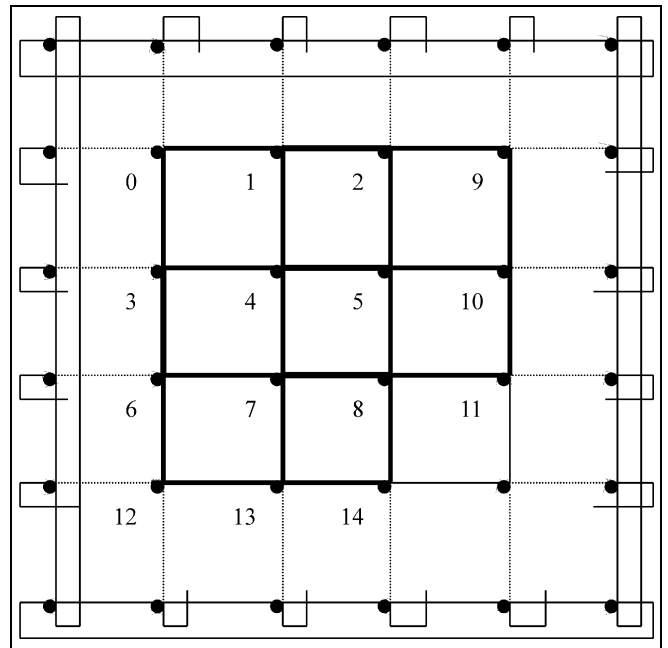


Рис. 4. Схема фрагмента тороидальной решетки

включающий в себя вершины с номерами 0, 1, 2, ..., 8, определяет расположение одной из подсистем  $S_0$ , определенных первичной раскладкой. После сдвига его границ вправо на один столбец получим подсистему  $S_{01}$ , включающую в себя вершины 1, 2, 4, 5, 7 и 8 из подсистемы  $S_0$ , вершины 9, 10 и 11 из подсистемы  $S_1$ , смежной справа с подсистемой  $S_0$ , и дуги 2—9, 5—10 и 8—11, соединяющие подсистемы  $S_0$  и  $S_1$ . Действуя подобным образом, после сдвига границ  $S_0$  вниз на одну строку получим подсистему  $S_{02}$ , включающую в себя вершины 3, 4, ..., 8 из подсистемы  $S_0$ , вершины 12, 13 и 14 из подсистемы  $S_2$ , смежной снизу с подсистемой  $S_0$ , и дуги 6—12, 7—13 и 8—14, соединяющие подсистемы  $S_0$  и  $S_2$ . (Дуги, входящие в рассматриваемые подсистемы  $S_0$ ,  $S_{01}$ ,  $S_1$ ,  $S_{02}$  и  $S_2$ , показаны утолщенными линиями, а все остальные дуги этого фрагмента — более тонкими.)

Если бы исходная матрица содержала диагональные связи, как в системе (см. рис. 1), то для их проверки и диагностирования потребовался бы еще один этап, перед началом которого границы первичной раскладки подсистем нужно было бы сдвинуть на один шаг по диагонали, например, вверх.

Сравнивая предложенный способ выделения подсистем достаточного размера со способом их выделения в работе [6], нужно отметить следующее:

- число вершин в отдельной подсистеме — одинаково (3 × 3);
- в работе [6] допускаются только отказы вершин, а дуги считаются безотказными; поэтому для диагностирования используется неизменная раскладка подсистем, называемых «розетками»;
- в работе [6] для описания оценок результатов тестирования используется модель Препарата—Метце—

Чжена (сокращенно модель ПМЧ) [10], отличающаяся от модели БГМ большей неопределенностью; поэтому процесс диагностирования состоит из трех этапов, которые существенно отличаются по содержанию от процедур, проводимых в данной работе.

Для получения оценки выигрыша  $A$  во времени предлагаемого способа параллельного диагностирования по сравнению с чисто последовательным сравним затраты времени на проведение контроля работоспособности исходной большой системы двумя указанными способами. Для простоты будем считать, что затраты времени пропорциональны числу проводимых последовательно проверок (тестов).

Контроль работоспособности цифровой системы заключается в том, что каждый проверяющий модуль проверяет техническое состояние своих приемников и соответствующих линий связи. Контроль цифровой системы будет полным, если проверены все модули системы и линии связи в обоих направлениях. Поскольку линии связи проверяются одновременно (в одном тесте) с модулями, которые они соединяют, то минимально необходимое число тестов определяется числом линий связи. Возвращаясь к нашей задаче, мы видим, что подсистема  $(3 \times 3)$  содержит 9 модулей и 12 линий связи. Таким образом, для полной проверки этой подсистемы потребуется не меньше, чем  $24 = 12 \times 2$  теста. На втором и третьем этапах потребуются тесты для проверки 6-ти смежных линий связи, т. е. еще 12 тестов. Таким образом, для проверки одной подсистемы и 6-ти прилегающих линий связи за три этапа потребуется не меньше 36-ти последовательно выполняемых тестов.

Для упрощения процесса получения оценки  $A$  будем считать, что  $N = 15a$ , т. е. что по каждой стороне квадрата исходной системы с тороидальной структурой раскладывается  $5a$  ( $a \geq 1$ ) подсистем размера  $(3 \times 3)$ , тогда на графе исходной системы можно разложить  $25a^2$  таких подсистем. Нетрудно подсчитать, что подобный граф включает в себя  $N^2 = 225a^2$  вершин и  $L = 2N^2 = 450a^2$  дуг. Пользуясь тем же подходом, что и для отдельной подсистемы  $(3 \times 3)$ , заключаем, что для контроля такой системы потребуется не меньше, чем  $T_S = 2 \times 450a^2 = 900a^2$  тестов. Если все эти тесты будут выполняться последовательно, то общее время контроля будет пропорционально  $900a^2$ , в то время как для контроля одной подсистемы нужно 36 тестов. Таким образом, параллельное исполнение процедур контроля в  $25a^2$  выделенных подсистемах размера  $(3 \times 3)$  может дать выигрыш по времени  $A_3 = 900a^2/36 = 25a^2$ .

Оценим теперь выигрыш  $A_5$  в ускорении процесса контроля исходной системы при использовании подсистем размера  $(5 \times 5)$ . Разделив  $N = 15a$  на 5, получим, что по каждой стороне графа исходной системы раскладывается  $3a$  квадратов размера  $(5 \times 5)$ . Каждая подсистема размера  $(5 \times 5)$  содержит 25 модулей и 40 линий связи. Таким образом, для полной проверки этой подсистемы потребуется не меньше  $80 = 40 \times 2$  тестов. На втором и третьем этапах потребуются тесты для проверки 10-ти прилегающих линий связи, т. е. еще 20 тестов. Таким образом, для проверки одной подсистемы  $(5 \times 5)$  и 10-ти прилегающих линий связи потребуется не мень-

ше 100 тестов. Поскольку число тестов  $T_S$  для контроля исходной системы осталось неизменным (равным  $900a^2$ ), то выигрыш  $A_5 = 900a^2/100 = 9a^2$ , в то время как для подсистем размера  $(3 \times 3)$  выигрыш  $A_3 = 25a^2$ . Причина уменьшения выигрыша с увеличением размера одной подсистемы понятна: уменьшение числа выделенных подсистем, определяющего число параллельно проводимых процессов (в данном случае, тестов), увеличивает время выполнения неизменного общего числа  $T_S$  процессов (тестов). Таким образом, для увеличения выигрыша благодаря параллельному выполнению диагностических процедур в разных подсистемах исходной цифровой системы целесообразно уменьшать размер (объем) выделенных подсистем. Но, как показано выше, с уменьшением размера подсистемы уменьшаются потенциальные возможности качественного выполнения диагностических процедур, в частности, не обеспечивается требуемая точность диагностирования — до одного компонента: модуля или линии связи. Таким образом, для организации эффективного распараллеливания диагностических процедур в цифровых системах со структурой типа тороидальной решетки целесообразно использовать подсистемы размера  $(3 \times 3)$ . В этом случае кратность размера  $N$  стороны системы и размера 3 стороны подсистемы можно записать в более простом виде:

$$N = 3k, \quad (2)$$

где  $k \gg 1$ .

Приведенные ранее оценки выигрыша  $A_3$  и  $A_5$  получены в предположении, что размер  $N$  стороны исходной системы кратен размеру стороны выделенных подсистем ( $N = 15a$ ). При невыполнении этого равенства для проверки модулей и линий связи исходной системы, не попавших в состав сформированных подсистем размера  $(3 \times 3)$  или  $(5 \times 5)$ , потребуются дополнительные раскладки таких подсистем и затраты времени на их контроль. Но эти дополнительные операции не могут изменить соотношение  $A_3 > A_5$  и потому не влияют на справедливость вывода о целесообразности использования подсистем размера  $(3 \times 3)$ .

Сформулируем основные этапы предлагаемого подхода к организации параллельного диагностирования больших цифровых систем со структурой типа тороидальной решетки.

1. Проверить выполнение равенства (2):

а) если оно выполняется, то провести первичную раскладку квадратов  $(3 \times 3)$  на графе исходной системы;

б) если — нет, то после проведения первичной раскладки в оставшихся столбцах и строках исходного графа выполнить раскладку квадратов  $(3 \times 3)$ , добавив недостающие один (два) столбца и одну (две) строки из числа смежных столбцов и строк, ранее включенных в подсистемы первичной раскладки; назовем эти действия вторичной раскладкой.

2. Диагностический монитор выдает команду на выполнение контроля работоспособности и, в случае необходимости, диагностирования подсистем, определенных первичной раскладкой.

3. По истечении определенного времени ДМ проводит реконфигурацию состава проверяемых подсистем,



сдвигая границы первичной раскладки вправо на один столбец, в результате такого сдвига в состав полученных подсистем входят дуги, расположенные справа от подсистем первичной раскладки, и вершины, которые они связывают.

4. Диагностический монитор выдает команду на выполнение контроля работоспособности и, в случае необходимости, диагностирования подсистем, образованных сдвигом вправо на один столбец границ первичной раскладки.

5. По истечении определенного времени ДМ проводит реконфигурацию состава проверяемых подсистем, сдвигая границы первичной раскладки вниз на одну строку, в результате такого сдвига в состав полученных подсистем входят дуги, расположенные снизу от подсистем первичной раскладки, и вершины, которые они связывают.

6. Диагностический монитор выдает команду на выполнение контроля работоспособности и, в случае необходимости, диагностирования подсистем, образованных сдвигом вниз на одну строку границ первичной раскладки.

7. По истечении определенного времени ДМ собирает диагностическую информацию из всех модулей, вошедших в подсистемы первичной раскладки, обрабатывает ее и формирует диагноз состояния проанализированных компонентов.

8. Если равенство (2) не выполняется, то этапы 2—6 повторяются для подсистем, образованных вторичной раскладкой.

9. По истечении определенного времени ДМ собирает диагностическую информацию из всех модулей, вошедших в подсистемы вторичной раскладки, суммирует ее с информацией, полученной для компонентов подсистем первичной раскладки, обрабатывает суммарную информацию и формирует общий диагноз о состоянии компонентов исходной цифровой системы.

#### ЗАКЛЮЧЕНИЕ

Для организации параллельного диагностирования больших цифровых систем со структурой типа тороидальной решетки предложен следующий подход. На графе исходной системы выделяются подсистемы в виде квадрата размера  $(n \times n)$ , т. е. проводится так называемая первичная раскладка подсистем. Затем в каждой из выделенных подсистем проводятся процедуры контроля и диагностирования, формируется частный диагноз состояния их компонентов. По истечении определенного времени диагностический монитор осуществляет реконфигурацию состава выделенных подсистем, сдвигая границы первичной раскладки вправо на один столбец, и выдает команду на выполнение контроля работоспо-

собности и диагностирования состояния компонентов образованных подсистем. Точно так же, по истечении определенного времени, ДМ осуществляет еще одну реконфигурацию состава выделенных подсистем, сдвигая границы первичной раскладки вниз на одну строку, и выдает команду на выполнение контроля и диагностирования компонентов вновь образованных подсистем. Наконец, по истечении определенного времени, диагностический монитор собирает диагностическую информацию из всех модулей, входивших в подсистемы первичной раскладки, обрабатывает ее и формирует общий диагноз состояния компонентов исходной системы.

Показано, что для сокращения общего времени контроля работоспособности системы целесообразно выделять подсистемы, состоящие из 9-ти узлов (вершин графа) и расположенные в виде квадрата  $(3 \times 3)$ .

#### ЛИТЕРАТУРА

1. Головкин Б. А. Вычислительные системы с большим числом процессоров. — М.: Радио и связь, 1995. — 320 с.
2. Андрианов А. Н. и др. О структуре вычислителя для решения задач обтекания. Комплексный подход к проектированию // Вычислительные процессы и системы. — 1985. — Вып. 2. — С. 13—62.
3. Об одном методе повышения надежности матричных многопроцессорных систем / А. Н. Мямлин, Л. А. Поздняков, Е. И. Котов, И. Б. Задыхайло // Электронная вычислительная техника. — 1988. — Вып. 2. — С. 26—37.
4. Barsi F., Grandoni F., Maestrini P. A theory of diagnosability of digital systems // IEEE Trans. Computers. — 1976. — Vol. C-25. — № 6. — P. 585—593.
5. Пархоменко П. П. Определение технического состояния многопроцессорных вычислительных систем путем анализа графа синдромов // Автоматика и телемеханика. — 1999. — № 5. — С. 126—135.
6. Baldelli L., Maestrini P. Diagnosis of processor arrays // Proc. of the 24-th Int. Sympos. on Fault Tolerant Computers. — 1994. — P. 48—54.
7. Ведешенков В. А. О диагностировании отказавших модулей и связей в цифровых системах с использованием модели БГМ // Автоматика и телемеханика. — 2002. — № 2. — С. 159—171.
8. Ведешенков В. А. Самодиагностирование цифровых систем с реконфигурацией // Проблемы управления. — 2003. — № 4. — С. 39—51.
9. Ведешенков В. А. Организация самодиагностирования технического состояния цифровых систем // Автоматика и телемеханика. — 2003. — № 11. — С. 165—182.
10. Preparata F. P., Metzger G., Chien R. T. On the connection assignment problem of diagnosable systems // IEEE Trans. Electr. Computers. — 1967. — Vol. 16. — № 6. — P. 848—854.

☎ (495) 334-75-90

e-mail: vva@i9.ipu.ru



### Для вас, аспиранты и докторанты!

В целях сокращения сроков публикаций в нашем журнале появилась новая рубрика — "Краткие сообщения", в которой публикуются конкретные результаты исследований, новые идеи и подходы к решению задач управления, изложенные в сжатой, но доступной для понимания большинства читателей форме.

Объем краткого сообщения — не более пяти страниц формата А4 вместе со списком литературы и иллюстрациями, текст набирается через полтора интервала, шрифт Times New Roman, кегль 12 пунктов. Обращаем внимание авторов на необходимость оформления статей в полном соответствии с правилами, которые можно найти на нашей Интернет-страничке [www.ipu.ru/period/pu](http://www.ipu.ru/period/pu).