

УДК 004.724.2+004.272.43

О.И. Виноградов, Л.С. Ломакина, Н.В. Сильянов

**АЛГОРИТМ ОБЕСПЕЧЕНИЯ ОТКАЗОУСТОЙЧИВОСТИ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ НА ГРАФ-МОДЕЛЯХ**

Нижегородский государственный технический университет им. П. Е. Алексеева

Работа посвящена проблеме отказов многопроцессорных вычислительных систем из-за блокировки пакетов данных между узлами. Исследованы варианты построения неблокирующих сетей. В качестве неблокирующей структуры выбран расширенный обобщенный гиперкуб. Узлы гиперкуба связаны простейшими сетями, имеющими структуру квазиполного однородного двудольного графа. Разработан алгоритм построения отказоустойчивой многопроцессорной системы со структурой расширенного обобщенного гиперкуба. Приведен пример программной реализации алгоритма в виде консольного приложения на языке Java.

*Ключевые слова:* отказоустойчивость, неблокирующая сеть, квазиполный граф, расширенный обобщенный гиперкуб.

В настоящее время одним из актуальных вопросов построения многопроцессорных систем является эффективность использования необходимого количества портов для коммуникаций между узлами. Наиболее распространенным отказом многопроцессорной сети является блокировка пакетов данных между узлами, что приводит к простоям процессоров и негативно сказывается на времени выполнения задачи. Также при этом повышаются затраты на обслуживание системы. В связи с изложенным целесообразно рассматривать проблему еще на стадии проектирования системы.

Разработаны алгоритмы построения отказоустойчивой многопроцессорной системы со структурой расширенного обобщенного гиперкуба. Преимущество этого метода заключается в использовании «квазиполных» графов, которые позволяют построить отказоустойчивую и масштабируемую архитектуру, сохранив при этом маршрутные свойства.

**Основные понятия**

Базовым понятием в данной работе является сеть Клоза. Сеть Клоза – один из видов многокаскадной неблокирующей коммутационной сети. Это означает, что в данной сети любой входной порт может быть подключен к любому свободному выходу без влияния на уже существующие соединения. Пример сети Клоза показан на рис. 1.

Сеть Клоза имеет три каскада: входной, промежуточный и выходной. Каждый каскад состоит из перекрестных коммутаторов, носящих название кроссбар. Каждый сигнал попадает на входной кроссбар, после чего направляется в любой кроссбар промежуточного каскада, который имеет свободную линию связи с входными и выходными кроссбарами, после чего направляется на кроссбар выходного яруса [1].

Моделью межсоединений в сети Клоза является полный однородный двудольный граф. Однородный двудольный граф  $D(X, Y, m, r)$  – граф, состоящий из двух непустых множеств вершин с одинаковой степенью, где  $X$  – множество кроссбаров первого и третьего каскадов;  $Y$  – множество кроссбаров второго каскада, а  $m$  и  $r$  – степень вершин  $X$  и  $Y$  соответственно. В данной работе рассматривается случай, когда в каждой доле имеется одинаковое количество вершин ( $X = Y = m = r$ ) – такие графы называют симметричными [2].

Как можно заметить из рис. 2, а, основным недостатком полного графа является большое количество связей. Поэтому возникает вопрос уменьшения количества связей между узлами. И его можно решить при помощи квазиполных графов. Однородный двудольный граф называется «минимально квазиполным», если каждый элемент множества  $X$  связан со всеми

остальными элементами своего множества единственным путем длины 2, который проходит через один из элементов множества  $Y$  [2]. Пример такого графа показан на рис. 2, б.

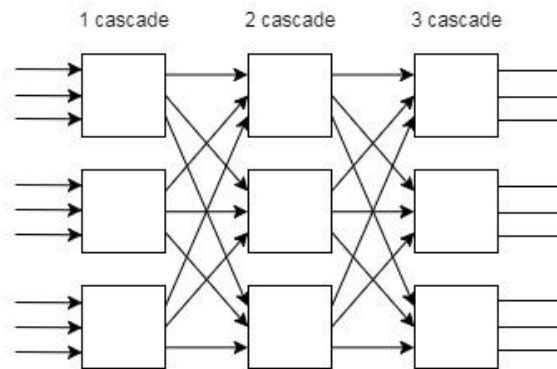


Рис. 1. Сеть Клоза

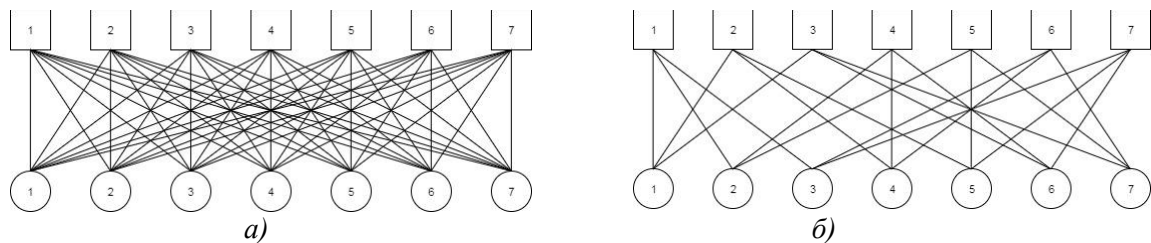


Рис. 2. Однородные двудольные графы:

а – полный с параметрами  $X = Y = m = r = 7$ ;

б – минимально квазиполный с параметрами  $X = Y = 7, m = r = 3$

Как видно из рис. 2, минимально квазиполный граф имеет намного меньше ребер. Количество вершин равно  $N = X = Y = \frac{m(m-1)}{\sigma} + 1$ , где  $\sigma < m$  и задает число путей длины 2 между двумя различными узлами. В данном случае  $\sigma = 1$ .

Описанные структуры используются при построении связей в обобщенном гиперкубе.

Обобщенный гиперкуб представляет собой  $n$ -мерный  $s$ -ный гиперкуб, который имеет  $V = s^d$  узлов степени  $d(s - 1)$ , размещенных в строках или столбцах по  $s$  узлов, задающих геометрические ребра  $d$ -мерного простого куба. Узлы данного обобщенного гиперкуба по столбцам и строкам будут иметь топологию полного однородного двудольного графа. Пример обобщенного гиперкуба показан на рис. 3.

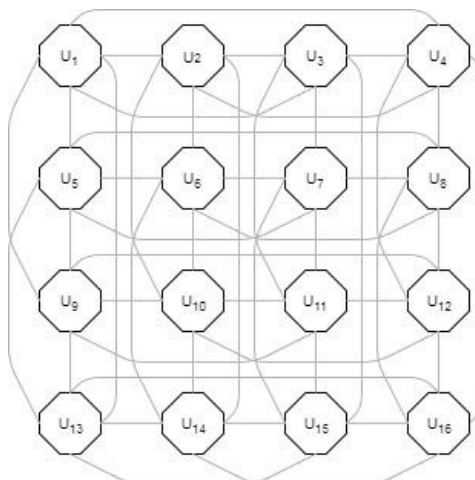


Рис. 3. Обобщенный гиперкуб с параметрами  $V = 16, d = 2, s = 4$

Для реализации системной сети с топологией обобщенного гиперкуба каждый из узлов должен иметь следующую схемную реализацию: узел  $U_i$  содержит процессорный узел  $A_i$ , «коммутатор измерений»  $g \times g$  ( $g = d + \pi$  и  $d$  коммутаторов каналов  $1 \times m$ , где  $m = s - 1$ ) для реализации ребер полного графа в каждой строке или столбце [3]. Пример такого узла показан на рис. 4.

Как отмечалось ранее, полный граф имеет слишком большое количество связей и для решения этой проблемы необходимо перейти на структуру квазиполного графа. Расширенный обобщенный гиперкуб является модификацией обобщенного гиперкуба. Для его получения необходимо заменить топологию связей с полного однородного двудольного графа на топологию простейшей сети, которая имеет структуру квазиполного однородного двудольного графа.

Простейшая сеть имеет набор параметров  $N, m, \sigma$ , где  $N$  – число элементов и блоков;  $m$  – число блоков, в которые входит каждый элемент, и число элементов, входящих в каждый блок;  $\sigma$  – задает число блоков, в которые входит каждая пара элементов.

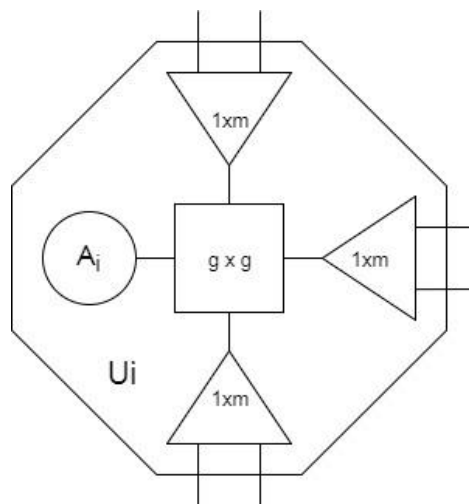


Рис. 4. Общая схемная реализация узла обобщенного гиперкуба

### Алгоритм построения системы

В общем виде алгоритм построения системы можно описать следующим образом:

1. Построение обобщенного гиперкуба;
2. Разработка простейших сетей;
  - 2.1 Проверка возможности существования минимального квазиполного графа;
  - 2.2 Построение минимального квазиполного графа;
3. Построение расширенного обобщенного гиперкуба.

На первом этапе строится базовая модель обобщенного гиперкуба. Пример выполнения этого этапа показан на рис. 3.

На втором этапе происходит разработка минимально квазиполных графов. На этапе 2.1 решается вопрос существования минимальных квазиполных графов. Доказать существование минимального квазиполного графа можно при помощи теоремы Брука, Райзера и Човла о существовании симметричных блок-схем.

Теорема Брука, Райзера и Човла [4]: Если существует симметричная блок-схема с параметрами  $N, m, \sigma$ , то, полагая  $\eta = m - \sigma$ , получаем:

- 1) если  $N$  четно, то  $\eta$  есть квадратцелого числа;
- 2) если  $N$  нечетно, то уравнение  $z^2 = \eta x^2 + (-1)^{\frac{N-1}{2}} \sigma y^2$  имеет решение в целых числах  $x, y, z$ , не равных одновременно нулю.

В работе [5] была выведено уравнение, связывающее параметры  $n$  и  $s$ . Оно имеет следующий вид:  $n = s(s - 1)/\sigma + 1$ .

В работе [6] приведены возможные параметры для симметричных блок-схем, представленные в табл. 1 и табл. 2.

Таблица 1

Параметры  $N$  и  $m$  при  $\sigma = 1$ 

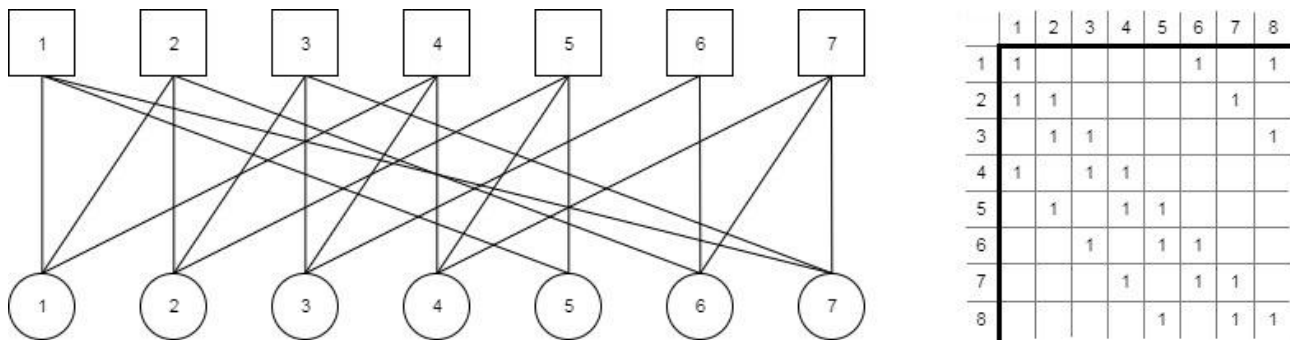
$B(N, m, 1)$						
$m$	2	3	4	5	6	8
$N$	3	7	13	21	31	57

Таблица 2

Параметры  $N$  и  $m$  при  $\sigma = 2$ 

$B(N, m, 2)$						
$m$	2	3	4	5	6	9
$N$	2	4	7	11	16	37

На этапе 2.2 выполняется построение минимального квазиполного графа. Имеется несколько методов построения [2]. На рис. 5 изображен пример выполнения этого этапа.

Рис. 5. Вариант с параметрами  $N = 7, m = 3, \sigma = 1$ 

В качестве примера выбрана симметричная блок-схема с параметрами  $N = 7, m = 3, \sigma = 1$ . В результате были получены следующие данные:

- минимальный квазиполный граф, построенный по имеющейся матрице. Квадратами на рис. 5 будут коммутаторы размерностью  $m \times x$ , а кругами узлы  $U_i$ ;
- матрица, описывающая связи между коммутаторами (строки) и узлами (столбцы) (1-существование ребра).

Получившиеся симметричные блок-схемы можно использовать для описания простейшей сети, которая является неблокируемой самомаршрутизируемой сетью. Это означает, что перестановка пакетов данных между абонентами может осуществляться в нем бесконфликтно по прямым каналам (без промежуточной буферизации пакетов), причем каждый абонент прокладывает свой канал самостоятельно [2].

В свою очередь, простейшие сети используются для построения расширенного обобщенного гиперкуба.

На рис. 6 представлен обобщенный гиперкуб с параметрами  $V = 16, d = 2, s = 4$ , построенный на этапе 3. Для построения расширенного обобщенного гиперкуба заменим топологию полного графа на получившиеся на предыдущем этапе простейшие сети. Поскольку число узлов в простейшей сети превосходит число имеющихся узлов, то увеличим число узлов с 4 до 7, тем самым «расширив» обобщенный гиперкуб с  $s$  до  $N$ . В результате получает-

ся расширенный обобщенный гиперкуб, в котором узлы всех строк и столбцов связаны в  $\sigma$  прямыми каналами.

Для формирования отказоустойчивой конфигурации простейшая сеть может быть заменена на 1-расширенную простейшую сеть. В такой сети малая часть узлов одной доли связана  $\sigma + 1$  путями длины 2, а остальные –  $\sigma$  путями длины 2. Получившиеся дополнительные независимые пути позволяют формировать  $(\sigma - 1)$ -отказоустойчивость по каналам и коммутаторам  $m \times m$  каждой простейшей сети. Следует заметить, что при  $\sigma = 1$ , связи не обеспечивают отказоустойчивость по связным узлам.

Также для обеспечения отказоустойчивости можно использовать избыточность расширенного обобщенного гиперкуба. Для этого необходимо разбить узлы каждой строки на пары четный-нечетный. Каждая такая пара должна входить в состав строки каждого измерения. Это приведет к уменьшению числа строк всех измерений, кроме первого, в два раза, но обеспечит два варианта перехода между строками разных измерений [3].

Аналогичным образом можно строить и отказоустойчивые расширенные обобщенные гиперкубы. Для этого строки должны разбиваться на группы по  $(\mu + 1)$  узлов. Однако в этом случае будет уменьшаться пропускная способность между измерениями.

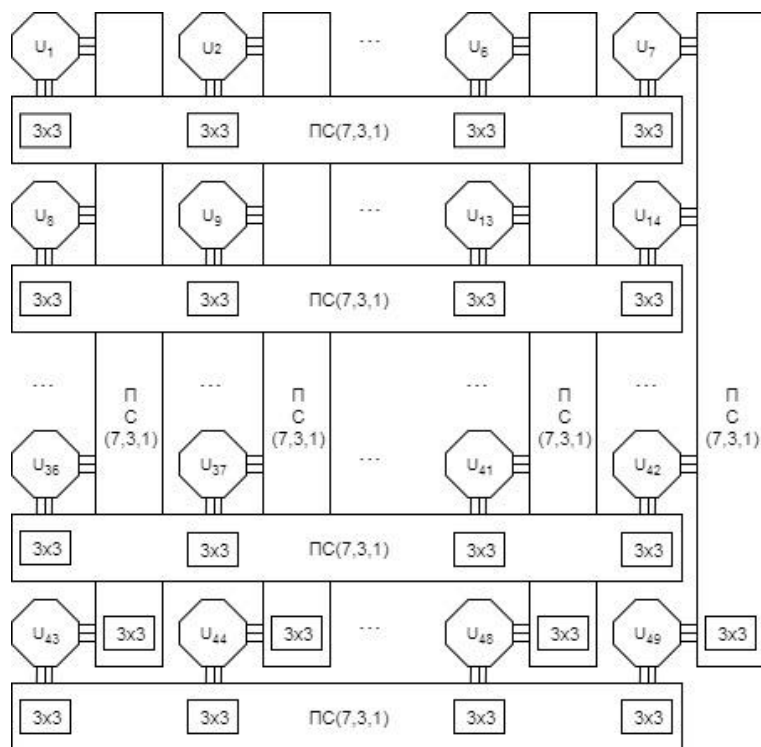


Рис. 6. Расширенный обобщенный гиперкуб с параметрами  $V = 49, d = 2, s = 7$ , узлы которого связаны (по столбцам и строкам) простейшими сетями с параметрами  $N = 7, m = 3, \sigma = 1$

### Программная реализация

Разработанный алгоритм реализован в виде консольного приложения на языке Java. Для хранения параметров симметричных блок-схем используется система управления базами данных MySQL. Структура проекта показана на рис. 7.

В модуле **algorithms** хранятся алгоритмы построения симметричных блок-схем. На данный момент реализован один комбинаторный алгоритм для построения симметричных блок-схем с параметром  $\sigma=1$ .

В модуле **infrastructure** хранятся вспомогательные классы. Основным классом является класс Database.java, реализующий соединение и работу с базой данных MySQL. Осуществляется данный функционал при помощи драйвера MySQL-connector.

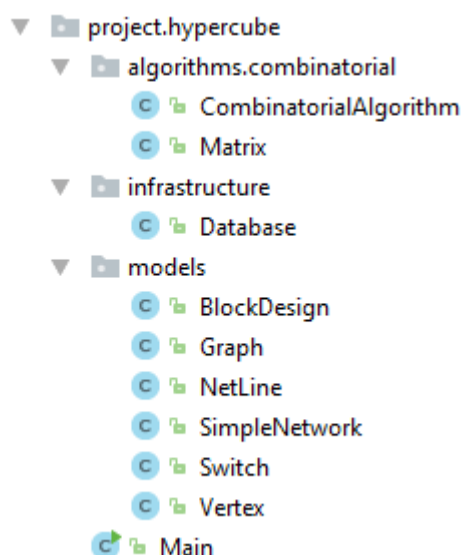


Рис. 7. Структура проекта

Модуль **models** содержит основные модели, используемые для моделирования выполнения алгоритма: **BlockDesign** для квазиполного графа, **Graph** для графа гиперкуба, **NetLine** для каналов связи, **SimpleNetwork** для системных сетей, **Switch** для коммутатора, **Vertex** для узла гиперкуба.

После ввода параметров программа построит гиперкуб, обобщенный гиперкуб и выведет параметры сети. После каждого шага пользователю будет предложено просмотреть список построенных каналов, как показано на рис. 8.

```
Гиперкуб:  
Данная модель содержит: 5 узл. и 4 кан.  
Получить список каналов ? (Д/Н) Д  
Switch №1 [id: 1639705018] <-> Switch №1 [id: 1627674070]  
Switch №1 [id: 1627674070] <-> Switch №1 [id: 1360875712]  
Switch №1 [id: 1360875712] <-> Switch №1 [id: 1625635731]  
Switch №1 [id: 1625635731] <-> Switch №1 [id: 1580066828]
```

Рис. 8. Список построенных каналов

## Выводы

Модель построения расширенного обобщенного гиперкуба позволяет проектировать системы с увеличивающимся количеством узлов и увеличивающимся числом независимых каналов между любыми парами узлов при сохранении числа портов. Это дает возможность обеспечивать масштабируемость и решить проблему блокировок пакетов данных сети.

## Библиографический список

1. **Назаров, А.Н.** АТМ: Технические решения создания сетей: справочное издание / А.Н. Назаров, И.А. Разживин, М.В. Симонов. – М.: Горячая Линия - Телеком, 2001. – С. 376.
2. **Каравай, М.Ф.** Комбинаторные методы построения двудольных однородных минимальных квазиполных графов (симметричных блок-схем) / М.Ф. Каравай, П.П. Пархоменко, В.С. Подлазов // Автоматика и телемеханика. – 2009. – Вып. 2. – С. 153–170.
3. **Каравай, М.Ф.** Расширенный обобщенный гиперкуб как отказоустойчивая системная сеть для многопроцессорных систем / М.Ф. Каравай, В.С. Подлазов // Управление большими системами. – 2013. – Вып. 45. – С. 344–371.
4. **Холл, М.** Комбинаторика / М. Холл. – М.: Мир, 1970.

5. Подлазов, В.С. Обобщенные сети Клоза / В.С. Подлазов, В.В. Соколов // Автоматика и телемеханика. – 2009. – Вып. 10. – С. 158–171.
6. Каравай, М.Ф. Топологические резервы суперкомпьютерного интерконнекта / М.Ф. Каравай, В.С. Подлазов // Управление большими системами. – 2013. – Вып. 41. – С. 395–423.

*Дата поступления  
в редакцию 10.06.2018*

**O.I. Vinogradov, L.S. Lomakina, N.V. Sil'yanov**

## **ALGORITHM OF ENSURING FAULT-TOLERANCE OF COMPUTING SYSTEMS BASED ON GRAPH-MODELS**

Nizhni Novgorod state technical university n.a. R.E. Alekseev

**Purpose:** The article is devoted to multiprocessor systems fault-tolerance problem, when data packets from one node can block another node.

**Methodology:** Variants of designing non-blocking networks are researched. The extended generalized hypercube is chosen as a non-blocking structure. Nodes of extended generalized hypercube are connected with an elemental networks. Structure of elemental network can be shown as quasi-complete homogeneous bipartite graph.

**Findings:** The algorithm of designing fault-tolerant multiprocessor system based on extended generalized hypercube is developed.

**Research implications:** This model allows to scale network structure without any essential modifications of algorithm.

**Originality:** The example of Java console application realizing the algorithm is shown. Parameters of the model stored with MySQL database.

*Key words:* fault-tolerance, non-blocking network, quasi-complete graph, extended generalized hypercube.