



**Э. А. СВИРЩЁВА**

**СТРУКТУРНЫЙ  
СИНТЕЗ  
НЕИЗОМОРФНЫХ  
СИСТЕМ  
С ОДНОРОДНЫМИ  
КОМПОНЕНТАМИ**

**Э.А.СВИРЩЁВА**

**СТРУКТУРНЫЙ СИНТЕЗ  
НЕИЗОМОРФНЫХ СИСТЕМ  
С ОДНОРОДНЫМИ  
КОМПОНЕНТАМИ**

Харьков  
Харьковский государственный технический  
университет радиоэлектроники  
1998

ББК 22.174 я9  
С24  
УДК 005; 519.7

**Свищёва Э.А.**

С24 Структурный синтез неизоморфных систем с однородными компонентами. –Х. : ХТУРЭ, 1998. – 256 с.  
**ISBN 5-7763-8886-4**

В монографии изложены теоретические основы структурного синтеза неизоморфных систем с однородными компонентами. Применение предлагаемых средств (математических моделей, методов и алгоритмов) такого синтеза проиллюстрировано на задачах из области программирования, схемотехники, конструктивного перечисления абстрактных комбинаторных объектов, наиболее часто используемых для представления структур технических и других материальных систем.

Для научных работников, инженеров, аспирантов и студентов, занимающихся вопросами оптимизации разрабатываемых систем в пространстве структур, и математиков, интересующихся приложениями комбинаторных методов, теории групп, теории отношений.

Переработано в 2005 году.

1402050000-07  
С -----

## ОГЛАВЛЕНИЕ

Предисловие.....	5
Глава 1. Математическое моделирование структуры системы и процесса структурного синтеза систем.....	8
1.1. Основные понятия и определения .....	8
1.2. Взаимозаменяемость, однородность, симметричность .....	13
1.3. Симметрии $Q$ -структуры .....	18
1.4. Симметрии и изоморфизм $Q$ -структур.....	22
1.5. Синтез неизоморфных $Q$ -структур .....	28
1.5.1. Применение метода решета .....	28
1.5.2. Применение идеи канонического упорядочения комбинаторных объектов .....	30
1.5.3. Построение признаков каноничности .....	37
1.5.4. Применение матричного представления $Q$ -структуры.....	41
1.5.5. Алгоритм конструктивного перечисления неизоморфных $Q$ -структур .....	44
1.6. Выявление симметрий $Q$ -структуры и ее подструктур .....	48
1.6.1. Применение разбиений .....	50
1.6.2. Алгоритм выявления симметрий $Q$ -структуры .....	57
1.7. Классификация задач порождения $Q$ -структур по трудоемкости..	67
Глава 2. Конструктивное перечисление неизоморфных комбинаторных объектов .....	69
2.1. Комбинаторные объекты и структурный синтез материальных систем .....	69
2.2. Конструктивное перечисление неизоморфных векторов.....	69
2.2.1. Кортеж, вектор .....	69
2.2.2. Конструктивное перечисление неизоморфных векторов с неповторяющимися координатами .....	70
2.2.3. Конструктивное перечисление неизоморфных векторов с повторяющимися координатами .....	76
2.3. Конструктивное перечисление неизоморфных систем инцидентностей.....	79
2.3.1. Алгоритм конструктивного перечисления .....	79
2.3.2. Выявление симметрий системы инцидентностей .....	83
2.4. Конструктивное перечисление неизоморфных графов .....	89
2.4.1. Постановка задачи .....	89
2.4.2. Каноническое упорядочение множества $(p,q)$ -графов.....	90
2.4.3. Применение информации о симметриях графа .....	96
2.4.4. Алгоритм порождения $(p,q)$ -графов.....	105
2.5. Конструктивное перечисление неизоморфных деревьев.....	113
2.5.1. Постановка задачи и выбор способа представления	

дерева .....	113
2.5.2. Симметрии корневого дерева .....	116
2.5.3. Алгоритм разбиения множества вершин дерева на классы эквивалентности.....	126
2.5.4. Алгоритм построения неизоморфных корневых деревьев...	130
Глава 3. Симметрии и синтез программ .....	136
3.1. Симметрии $n$ -мерного логического пространства .....	136
3.2. Структура множества булевых функций .....	143
3.3. Симметрии булевых функций.....	145
3.4. Выявление симметрий булевых функций.....	149
3.4.1. Выявление симметрий функции, заданной в СДНФ.....	151
3.4.2. Выявление симметрий функции, заданной в ДНФ.....	156
3.5. Выявление симметрий булевых функций специальных видов....	161
3.5.1. Симметрические булевы функции.....	162
3.5.2. Ортогональные булевы функции.....	165
3.6. Симметрии системы булевых функций.....	171
3.7. Метод синтеза оптимальных бинарных программ .....	174
3.8. Применение информации о симметриях булевой функции при синтезе бинарных программ.....	181
3.9. Примеры синтеза оптимальных программных реализаций булевой функции .....	183
3.10. Приведение булевой функции к минимальной ортогональной форме .....	188
3.11. Применение информации о симметриях булевой функции в процессе приведении ее к МОДНФ.....	198
Глава 4. Симметрии и синтез схем.....	202
4.1. Синтез линейных избирательных устройств и проблема изоморфизма .....	202
4.2. Связь между структурой схемы и ее схемной функцией.....	204
4.3. Структура множества схемных реализаций заданной функции....	207
4.4. Эквивалентные преобразования и структурный синтез схем .....	213
4.5. Формальная постановка задачи структурного синтеза на этапе реализации .....	220
4.6. Реализация слагаемого полиномиального коэффициента передаточной функции .....	224
4.7. Реализация заданного спектра степеней.....	229
4.8. Отсечение изоморфных матриц $\tilde{Y}$ .....	233
4.9. Алгоритм синтеза схем с операционными усилителями.....	237
Список использованной литературы.....	251

## ПРЕДИСЛОВИЕ

Качество функционирования технической системы в значительной степени определяется выбором ее структуры в процессе проектирования. Однако в настоящее время именно на этом этапе разработки системы решение принимается преимущественно на основании опыта, интуиции, изобретательским путем, т.е. без достаточного обоснования. Развитие методов структурной оптимизации существенно отстает от развития методов численной оптимизации, используемой на этапе определения параметров проектируемой системы после выбора ее структуры. Но применение хороших методов параметрической оптимизации при плохой выбранной структуре бессмысленно. Поэтому диспропорция между темпами развития доступных инженеру средств структурной и параметрической оптимизации является тормозом технического прогресса. Таким образом, интенсификация развития методов структурного синтеза необходима, чтобы обеспечить доступность мощного резерва качества проектирования – оптимизации в пространстве структур, т.е. за счет выбора оптимальной структуры системы.

Отставание методов структурной оптимизации связано с чрезвычайной трудоемкостью обзора множества всех конкурентоспособных вариантов искомой структуры, поэтому для большинства задач проектирования он остается практически невыполнимым. Эта трудоемкость обусловлена комбинаторным характером решаемой задачи. Привлечение вычислительной техники к ее решению лишь незначительно расширяет пределы возможного.

Несмотря на указанные трудности, развитие методов структурного синтеза не останавливается, так как движущая сила этого процесса – жизненная необходимость постоянного совершенствования технических решений, требующая от инженеров и исследователей неустанный поиск средств их оптимизации. «Оптимальность является той жар-птицей, за которую стоит заплатить обожженными руками» (Растринин Л.А. Системы экстремального управления. М.: Наука, 1974. С. 9).

В большинстве работ, посвященных структурному синтезу технических систем, применен неявный подход к поиску оптимальной структуры. Он нацелен на однозначное определение оптимальных структуры и параметров разрабатываемой системы аналитическим путем, исходя из требований технического задания к ней, представленных совокупностью конструктивных критериев, ограничений и целевых функций. В отдельных приложениях удается решить эту задачу, но чаще всего такой подход обеспечивает лишь сокращение перебора допу-

стимых вариантов структуры, который в общем случае неизбежен в силу принципиальной неоднозначности решения задачи структурного синтеза.

Естественно, что конкретные методы синтеза, основанные на неявном подходе к определению оптимальной структуры, различны для различных приложений, так как построение эффективных конструктивных критериев, определяющих стратегию поиска, достигается путем глубокого проникновения в специфику решаемой задачи. Сосредоточенностью на специфике конкретных приложений обусловлена разобщенность исследований в области структурного синтеза, препятствующая распространению опыта, накопленного в одних приложениях, на другие. По этой причине разработчикам средств структурного синтеза, работающим в различных областях техники, трудно видеть, что они каждый в своей области, по существу, сталкиваются с общими для различных приложений проблемами.

Одна из таких проблем – исключение в процессе структурного синтеза изоморфных вариантов структуры, количество которых может существенно превышать количество неизоморфных вариантов, представляющих практический интерес. Математики столкнулись с этой проблемой при решении задачи конструктивного перечисления комбинаторных объектов: из 133 632 сгенерированных 8-вершинных графов только 12 346 оказались неизоморфными [1], из  $10^8$  10-вершинных корневых деревьев – только 106 [2] и т.д. Известно [3;4], что задача выявления изоморфных вариантов с целью их исключения относится к весьма трудоемким. Поэтому очевидно, что эффективность методов структурного синтеза систем в значительной степени определяется эффективностью решения указанной проблемы изоморфизма. Предлагаемая работа посвящена решению этой проблемы.

В монографии определен класс комбинаторных задач структурного синтеза систем, при решении которых эта проблема возникает; исследована природа появления изоморфных вариантов; введена математическая модель структуры системы, позволяющая отвлечься от специфических особенностей конкретных приложений и дать общие рекомендации по устранению изоморфных вариантов в процессе структурного синтеза. На базе этой модели разработан принцип синтеза неизоморфных вариантов искомой структуры. В связи с комбинаторным характером решаемых здесь задач основной упор сделан на регулярные методы их решения, которые легко автоматизируются. Процедура структурного синтеза представлена в виде управляемого вычислительного процесса с использованием в качестве управляющего фактора, не допускающего построения изоморфных вариантов, информации о симметриях подсистем синтезируемой системы. Рассмотрены различные приложения предлагаемого решения проблемы изоморфизма. По методологическим соображениям выбраны приложения, в которых эта проблема стоит наиболее остро. В приведенных решениях прикладных задач обращено внимание на использование специфики последних с целью максимального упрощения их комбинаторных подзадач, способствующего выбору оптимальной стратегии поиска искомой структуры.

В гл.1 показано, что появление изоморфных вариантов в процессе структурного синтеза систем обусловлено симметриями системы и ее подсистем. Построенная в ней математическая модель структуры технической системы отражает ее особенности как комбинаторного объекта. Эта модель позволила проследить механизм образования изоморфных вариантов в процессе структурного синтеза систем и разработать теоретические основы структурного синтеза неизоморфных систем с использованием информации об указанных симметриях. На основе этой модели дана классификация задач структурного синтеза по их трудоемкости. Уделено внимание повышению эффективности методов выявления симметрий системы и ее подсистем.

В гл. 2 продемонстрирована возможность повышения эффективности конструирования не только материальных, но и абстрактных систем с помощью предложенных средств устранения изоморфных вариантов. Рассмотрено конструктивное перечисление неизоморфных комбинаторных объектов, которые наиболее часто применяются для представления структуры материальной системы. Это кортежи, графы, в том числе деревья, системы инцидентностей.

Применение разработанных средств структурного синтеза в области программирования проиллюстрировано в гл. 3 на примере синтеза бинарных программ с минимальным объемом занимаемой памяти при максимальном быстродействии. Эти средства использованы при разработке метода построения бинарных программ, обеспечивающего лучшие результаты оптимизации по сравнению с известными методами.

В гл. 4 рассмотрено применение средств структурного синтеза в схемотехнике на примере синтеза линейных избирательных устройств с операционными усилителями. Разработанные в результате методы, алгоритмы и программы синтеза схем подтверждают реальность привлечения компьютера к поиску патентоспособных технических решений.

Предлагаемые средства структурного синтеза неизоморфных систем и рекомендации по их применению, проиллюстрированные на примерах конструирования систем, относящихся к различным областям техники, как надеется автор, послужат интенсификации разработки и других конкретных методов структурного синтеза, предназначенных для различных приложений. Тем самым они будут способствовать расширению круга технических задач, при решении которых становится реальной оптимизация проектируемой системы в пространстве структур.



МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ СТРУКТУРЫ СИСТЕМЫ  
И ПРОЦЕССА СТРУКТУРНОГО СИНТЕЗА СИСТЕМ

**1.1. Основные понятия и определения**

Под системой принято подразумевать множество элементов, которые, находясь в определенных связях и отношениях между собой, образуют целостность, характеризующуюся принципиальной несводимостью ее свойств к сумме свойств составляющих элементов [5]. Практически каждый объект может рассматриваться как система, т.е. это понятие имеет чрезвычайно широкую область применения. Ограничимся рассмотрением технических систем. Это материальные и абстрактные системы, отличающиеся от прочих тем, что являются продуктом творческой деятельности человека.

Элемент (или компонент) технической системы при общепринятом блочно-иерархическом [6] подходе к ее проектированию – это та ее часть, которая на данном уровне проектирования считается не расчленяемой. Из этого определения компонента следует, что он, в свою очередь, может рассматриваться как система, т.е. в нем могут быть также выделены составные части – подкомпоненты. Далее под компонентами будем подразумевать только такие, множества подкомпонентов которых не пересекаются.

Каждый компонент характеризуется множеством внешних связей (выводов, портов), через которые возможен обмен (функциональный, информационный, энергетический и т.д.) с другими компонентами. Из определения системы следует, что она характеризуется неким набором внутренних связей между компонентами, с помощью которых реализуется указанный обмен. Свойства системы проявляются в процессе ее взаимодействия со средой, реализуемого через посредство ее внешних связей.

Совокупность устойчивых связей системы (или объекта), обеспечивающих ее целостность и тождественность самой себе, т.е. инвариантность основных свойств при различных внешних и внутренних изменениях, называют структурой системы [7]. В общем случае структура системы может быть формально описана следующим образом.

Пусть  $I$  – произвольное множество, которое поставим в соответствие множеству компонентов системы. Это могут быть, например, числа натурального ряда, которые можно считать порядковыми номерами компонентов системы. В дальнейших рассуждениях множество  $I = \{i\}$  условимся называть компонентами системы. Аналогично каждому компоненту  $i \in I$  поставим в соответствие множество  $L_i = \{l_i\}$ , называя его множеством внешних связей этого компонента. По тому же принципу множеству внеш-

них и внутренних связей системы поставим в соответствие множество  $J$ , которое назовем множеством связей системы. Элементы множеств  $I, J, L_i, i=1, 2, \dots, |I|$ , для краткости условимся называть структурными элементами.

Для каждого  $i \in I$  зададим инъективное отображение

$$t_i: L_i \rightarrow J.$$

Будем говорить, что задана структура  $Q$  ( $Q$ -структура), если заданы множества  $I, J$  и отображения  $t_i$  для всех  $i \in I$ . Совокупностью отображений  $t_i, i=1, 2, \dots, |I|$ , определено отношение инцидентностей  $N$  между множествами  $I$  и  $J, INJ$ . Задание  $t_i$  равносильно заданию  $|L_i|$ -арного отношения

$$q_i = \left( \begin{array}{c} i \\ j_{i1}, j_{i2}, \mathbf{K}, j_{i l_i}, \mathbf{K}, j_{i |L_i|} \end{array} \right), \quad (1.1)$$

которое отображает расположение  $i$ -го компонента внутри системы. Задание  $Q$ -структуры соответствует заданию совокупности множества  $J$  и отношений  $q_i$  для всех  $i \in I, Q = \{J, q_1, q_2, \mathbf{K}, q_{|I|}\}$ .

Для наглядности  $Q$ -структура может быть представлена "взвешенной" матрицей инцидентностей  $\mathbf{X} = \|x_{ij}\|_{|I| \times |J|}$ , где  $x_{ij} = l_i$ , если элемент  $l_i \in L_i$  инцидентен  $j$ , и  $x_{ij} = 0$  – в противном случае.

**Пример 1.1.** Матрицей

$$\mathbf{X} = \left\| \begin{array}{cccccc} 1 & 0 & 2 & 0 & 3 & 0 \\ 0 & 1 & 2 & 3 & 4 & 0 \\ 1 & 0 & 0 & 2 & 3 & 4 \end{array} \right\|$$

представлена  $Q$ -структура  $Q = \{J, q_1, q_2, q_3\}$  системы из трех компонентов, где

$$J = \{1, 2, \dots, 6\}; \quad q_1 = \left( \begin{array}{c} 1 \\ 1, 3, 5 \end{array} \right) \quad q_2 = \left( \begin{array}{c} 2 \\ 2, 3, 4, 5 \end{array} \right) \quad q_3 = \left( \begin{array}{c} 3 \\ 1, 4, 5, 6 \end{array} \right)$$

Очевидно, что матрица инцидентностей представляет собой частный случай матрицы  $\mathbf{X}$  при  $l_{i1} = l_{i2} = \mathbf{L} = l_{i|L_i|} = 1, i = 1, 2, \dots, |I|$ . Таким образом,  $Q$ -структура является обобщением понятия системы инцидентностей, применение которой для описания различных абстрактных систем (графы, гиперграфы, блок-схемы и др.) широко распространено.

Указанное формальное описание структуры системы охватывает функциональные, организационные, электрические, механические, кинема-

тические и другие связи между компонентами различной природы. Исключение составляют системы, характеризующиеся взаимным расположением их компонентов в пространстве, при котором требуется учет формы этих составляющих и их размеров. Такие системы встречаются, например, при решении различных конструкторских задач, хотя и они на некоторых этапах идеализации могут быть описаны, как показано ранее. Введенное описание структуры целесообразно использовать при решении задач структурного синтеза не только материальных, но и абстрактных систем, какими являются, например, ориентированные графы, графы с раскрашенными вершинами и (или) ребрами.

Отображение  $t_i$  для каждого  $i \in I$  может быть задано не единственным способом. Поэтому можно определить множество  $M$   $Q$ -структур

с заданным набором структурных элементов. Остановимся на формализации конструктивного процесса построения всех элементов множества  $M$ .

Задание отображения

$$t_i = \left( \begin{array}{cccc|c} 1 & 2 & \dots & & |L_i| \\ j_{i1} & j_{i2} & \dots & & j_{i|L_i|} \end{array} \right) \quad (1.2)$$

символизирует отождествление внешних выводов  $i$ -го компонента,  $i \in I$ , с частью связей системы. Поэтому в дальнейшем, говоря о подключении  $i$ -го компонента к системе, будем подразумевать задание  $t_i$ . Аналогично условимся замену хотя бы части связей системы, с которыми отождествляются выводы  $i$ -го компонента, называть переключением этого компонента. Переход от одной структуры к другой осуществляется переключением одного или нескольких ее компонентов.

Множество  $J$  вместе с подключенными компонентами, имеющими порядковые номера  $1, 2, \dots, i$ , назовем  $i$ -подструктурой и обозначим  $Q_i$ ,  $Q_i = \{J, q_1, q_2, \mathbf{K}, q_i\}$ ,  $i \in I$ . Тогда множество  $J$  можно рассматривать как 0-подструктуру или  $Q$ -структуру в 0-состоянии.

Пусть имеется  $(i-1)$ -подструктура  $Q_{(i-1)}$ ,  $i \in I$ . Процедуру подключения  $i$ -го компонента к  $(i-1)$ -подструктуре представим с помощью операции объединения:

$$Q_i = Q_{(i-1)} \mathbf{U} q_i.$$

Для того чтобы получить множество  $M_i$  всех возможных вариантов  $i$ -подструктуры  $Q_i$ , необходимо рассмотреть все возможные варианты подключения  $i$ -го компонента к каждой фиксированной  $(i-1)$ -

подструктуре. Переход от одного варианта  $i$ -подструктуры к другому, соответствующий переключению  $i$ -го компонента, формализуется изменением выборки  $\mathbf{j}_i = (j_{i1}, j_{i2}, \mathbf{K}, j_{i|L_i|})$  из  $J$  в (1.1), (1.2). Очевидно, что задача конструктивного перечисления векторов  $\mathbf{j}_i$  распадается на две подзадачи, машинные алгоритмы решения которых приведены в работе [3] и др.:

построение всех неупорядоченных подмножеств множества  $J$ , содержащих  $|L_i|$  элементов (перебор сочетаний из  $J$  элементов по  $|L_i|$ );

упорядочение каждого такого подмножества всеми возможными способами (перебор перестановок элементов подмножества).

Таким образом, в основе конструктивного процесса построения множества вариантов  $i$ -подструктуры лежит перебор вариантов выборки  $\mathbf{j}_i$ . Пусть отношением

$$q_{i0} = \binom{i}{1, 2, \dots, |L_i|}$$

представлен исходный вариант расположения  $i$ -го компонента в  $i$ -подструктуре. Тогда любой другой вариант его расположения представим как

$$q_i = q_{i0} t_i(k) g_i(n),$$

где  $t_i(k)$  –  $k$ -й вариант отображения с лексикографически упорядоченной в алфавите  $(1, 2, \dots, |J|)$  выборкой  $\mathbf{j}_i(k)$ ,  $k = 1, 2, \mathbf{K}, C(|J|, |L_i|)$ ;  $g_i(n)$  –  $n$ -й вариант подстановки элементов выборки  $\mathbf{j}_i(k)$ ,  $n = 1, 2, \dots, P(|L_i|)$ . Всего насчитывается

$$N_i = C(|J|, |L_i|) P(|L_i|) = |L_i|! |J|! / [|L_i|! (|J| - |L_i|)!] = |J|! / (|J| - |L_i|)!$$

вариантов отношений  $q_i$ . Перемножение здесь и далее выполняется по правилам умножения преобразований [8]: сначала определяется образ  $l_i t_i(k)$  каждого элемента множества  $L_i$ , составляющей вторую строку отношения  $q_{i0}$ , под действием преобразования  $t_i(k)$ , а потом – образ каждого элемента  $l_i t_i(k)$  под действием преобразования  $g_i(n)$ .

Нам удобнее будет использовать равноценный конструктивный процесс, где подстановка  $g_i(n)$  элементов выборки  $\mathbf{j}_i$  заменена подстановкой  $g_{L_i}(n)$  элементов множества  $L_i$ :

$$q_i = q_{i0} g_{L_i}(n) t_i(k). \quad (1.3)$$

В этом случае в выражении (1.2) отображения  $t_i$  перестановка элементов второй строки заменена перестановкой элементов первой строки. Ясно, что второй вариант конструктивного процесса приведет к тому же множеству  $M$   $Q$ -структур, что и первый вариант. Различие может быть только в последовательности порождения элементов этого множества.

В дальнейшем, говоря о тривиальном алгоритме порождения  $Q$ -структур, будем подразумевать естественный для выбранной математической модели структуры системы конструктивный процесс, состоящий в переборе для каждой  $(i-1)$ -подструктуры,  $i \in I$ , вариантов  $q_i$  в соответствии с (1.3). Этот алгоритм интерпретируется обходом дерева  $T_1$  вариантов (рис.1.1). В дереве  $T_1$  отношения  $q_i$  представлены соответствующими выборками  $\mathbf{j}_i$ . Переход от одного варианта отношения  $q_i$  к другому при неизменной  $(i-1)$ -подструктуре формализуется с помощью отображения

$$t_i = g_{L_i}(n) t_i(k), \quad (1.4)$$

которое можно рассматривать как оператор, составляющий основу шага конструктивного процесса синтеза вариантов структуры системы. В частном случае, когда  $|L_i| = |J|$ , отображение  $t_i$  становится подстановкой.

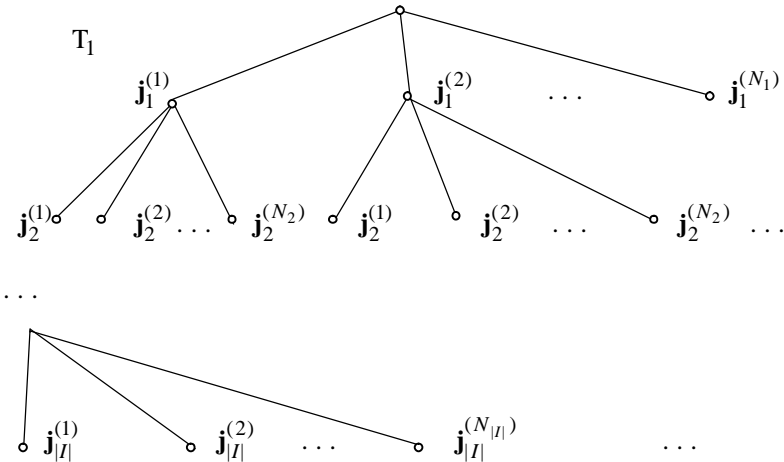


Рис. 1.1

## 1.2. Взаимозаменяемость, однородность, симметричность

Компоненты реальной технической системы (их внешние связи) могут быть как разнотипными, так и однотипными, т.е. взаимозаменяемыми. Их свойство "быть однотипными" формально отображается заданием на соответствующем множестве  $I$  ( $L_i, i = 1, 2, \dots, |I|$ ) отношения эквивалентности  $F_I$  ( $F_i$ ):

$$iF_I i' \Leftrightarrow i \sim_{F_I} i';$$

$$l_i F_i l'_i \Leftrightarrow l_i \sim_{F_i} l'_i.$$

Отношения эквивалентности разбивают множества  $I$  и  $L_i, i = 1, 2, \dots, |I|$ , на классы эквивалентности:

$$I = \bigcup_{\alpha=1}^A I_{\alpha}; \quad (1.5)$$

$$L_i = \bigcup_{\delta_i=1}^{D_i} L_{\delta_i}, i = 1, 2, \dots, |I|. \quad (1.6)$$

В (1.5)  $I_{\alpha}$  –  $F_I$ -класс,  $A$  – количество  $F_I$ -классов; в (1.6)  $L_{\delta_i}$  –  $F_i$ -класс,  $D_i$  – количество  $F_i$ -классов.

Для учета свойства элементов системы "быть разнотипными" введем на множествах  $F_I$ -классов и  $F_i$ -классов отношения строгого порядка

$$\alpha R_I \alpha' \Leftrightarrow \alpha \mathbf{f} \alpha', \quad (1.7)$$

$$\delta_i R_i \delta'_i \Leftrightarrow \delta_i \mathbf{f} \delta'_i, \quad (1.8)$$

условившись о принципе определения старшинства (или предпочтительности) одного типа перед другим, обозначаемого значком  $\mathbf{f}$ . Тем самым на элементах множества  $I$  ( $L_i, i = 1, 2, \dots, |I|$ ) задано отношение частичного порядка  $T_I$  ( $T_i, i = 1, 2, \dots, |I|$ ):

$$\forall i \in I_{\alpha}, \forall i' \in I_{\alpha'} \quad \alpha R_I \alpha' \Leftrightarrow iT_I i' \Leftrightarrow i \mathbf{f} i';$$

$$\forall l_i \in L_{\delta_i}, \forall l'_i \in L_{\delta'_i}, \quad \delta_i R_i \delta'_i \Leftrightarrow l_i T_i l'_i \Leftrightarrow l_i \mathbf{f} l'_i, \quad i = 1, 2, \dots, |I|.$$

Аналогично введем отношение эквивалентности  $F_J$  на множестве  $J$  – носителе связей системы

$$jF_J j' \Leftrightarrow j \stackrel{F_J}{\sim} j'.$$

Тогда множество  $J$  разбивается на классы эквивалентности –  $F_J$ -классы :

$$J = \bigcup_{\beta=1}^B J_\beta, \quad (1.9)$$

где  $J_\beta$  –  $F_J$ -класс,  $B$  – количество  $F_J$ -классов .

На множестве  $F_J$ -классов  $J_\beta$ ,  $\beta = 1, 2, \dots, B$ , введем отношение строгого порядка  $R_J$  :

$$\beta R_J \beta' \Leftrightarrow \beta \mathbf{f} \beta'. \quad (1.10)$$

Отношением  $R_J$  задается отношение частичного порядка  $T_J$  на множестве элементов  $J$  :

$$\forall j \in J_\beta, \quad \forall j' \in J_{\beta'}, \quad \beta R_J \beta' \Leftrightarrow jT_J j' \Leftrightarrow j \mathbf{f} j'.$$

Введенными отношениями эквивалентности  $F_J$  и порядка  $R_J$  учитывается априорное разделение связей системы (до их конкретизации) на однотипные и разнотипные. Например, в общем случае не являются взаимозаменяемыми внутренние и внешние связи системы.

Взаимозаменяемость компонентов системы и (или) их внешних связей не может не повлиять на свойства системы в целом: как правило, хотя бы некоторые из ее компонентов, их внешних связей, связей системы остаются взаимозаменяемыми и внутри ее. Остановимся на формальных средствах отображения такой взаимозаменяемости.

Отношения эквивалентности  $F_i$  и порядка  $T_i$  на  $L_i$ ,  $i \in I$ , через посредство отображения  $t_i$  индуцируют дополнительные эквивалентности и порядки на  $J$ . Следовательно, на множестве  $J$   $i$ -подструктуры,  $i \in I$ , возникает в общем случае  $(i+1)$  отношений эквивалентности и  $(i+1)$  отношений порядка. Новое отношение эквивалентности  $F$  на множестве  $J$

$$F = F_J \mathbf{I} \left( \bigcup_{i'=1}^i F_{i'} \right) \quad (1.11)$$

означает, что отношение  $jFj'$  выполняется тогда и только тогда, когда одновременно выполняются отношения  $F_J$  и  $F_{i'}$ ,  $i'=1, 2, \dots, i$ . Новое отношение порядка

$$T = T_J \mathbf{U} \left( \bigcup_{i'=1}^i T_{i'} \right) \quad (1.12)$$

означает, что отношение  $jTj'$  выполняется тогда и только тогда, когда выполняется хотя бы одно из отношений  $T_J$ ,  $T_{i'}$ ,  $i'=1, 2, \dots, i$ . При этом каждый класс  $J_\beta$ , в свою очередь, разбивается на классы эквивалентности  $J_{\beta_k}$  ( $F$ -классы):

$$J_\beta = \bigcup_{k=1}^K J_{\beta_k} .$$

В этом случае каждый новый  $F$ -класс  $Q_i$  оказывается пронумерованным  $(i+1)$  раз, т.е. имеет  $(i+1)$  порядковых номеров, каждый из которых соответствует одному из  $(i+1)$  отношений порядка на множестве  $J$ . При любом выборе нумерации множества порядков  $R_J, R_1, R_2, \dots, R_i$  порядок  $R$  на множестве  $F$ -классов  $Q_i$  задается лексикографическим порядком на множестве последовательностей из  $(i+1)$  их порядковых номеров.

Таким образом,  $Q$ -структура при наличии в ней как однотипных, так и разнотипных компонентов задается совокупностью отношений инцидентностей  $N$  (т.е. отображений  $t_i$ ,  $i = 1, 2, \dots, |I|$ ), эквивалентности  $F_I, F, F_i$ ,  $i = 1, 2, \dots, |I|$ , и строгого порядка  $R_I, R, R_i$ ,  $i = 1, 2, \dots, |I|$ , на классах эквивалентности (частичного порядка  $T_I, T, T_i$ ,  $i = 1, 2, \dots, |I|$ , на элементах множеств  $I, J, L_i$ ,  $i = 1, 2, \dots, |I|$ , соответственно).

Если в качестве структурных элементов  $Q$  выбраны порядковые номера соответствующих элементов системы, наиболее естественно лексикографическое упорядочение  $F_I, F, F_i$ -классов  $I_\alpha, L_{\delta_i}, J_\beta$ , рассматриваемых как векторы  $\mathbf{i}_\alpha, \mathbf{l}_{\delta_i}, \mathbf{j}_\beta$  соответственно в алфавитах  $(1, 2, \dots, |I|)$ ,  $(1, 2, \dots, |L_i|)$ ,  $(1, 2, \dots, |J|)$ . Тогда из (1.7), (1.8), (1.10) следуют соотношения:

$$\alpha R_I \alpha' \Leftrightarrow \mathbf{i}_\alpha \succ^{lex} \mathbf{i}_{\alpha'} ,$$

$$\delta_i R_i \delta'_i \Leftrightarrow \mathbf{l}_{\delta_i} \succ^{lex} \mathbf{l}_{\delta'_i} , \quad i = 1, 2, \dots, |I| ,$$



$$\beta R_J \beta' \Leftrightarrow \mathbf{j}_\beta \overset{lex}{\succ} \mathbf{j}_{\beta'}$$

В этом случае отношения инцидентностей, эквивалентности и порядка, определяющие  $Q$ -структуру, наиболее наглядно можно представить матрицей инцидентностей  $\mathbf{X}$ , в которой элементы множества  $L_i$ , относящиеся к общему  $F_i$ -классу, обозначены одинаковыми "весами", означающими порядковые номера  $F_i$ -классов. Такая взвешенная матрица инцидентностей отличается от рассмотренной ранее тем, что в ней иначе определяются ненулевые элементы:  $x_{ij} = \delta_i$ , если элемент  $l_i \in L_i$ , инцидентный  $j \in J$ , принадлежит  $F_i$ -классу  $L_{\delta_i}$ . Тогда в наличии одинаковых (различных) значений  $x_{ij}$  отражено отношение эквивалентности  $F_i$  (порядка  $R_i$ ). Разбиением множеств  $I$  и  $J$  на  $F_I$ - и  $F_J$ -классы индуцируется разбиение множеств строк и столбцов матрицы  $\mathbf{X}$  на подмножества. Назовем их также соответственно  $F_I$ - и  $F_J$ -классами. Отношения порядков  $R_I$  и  $R_J$  на множествах  $F_I$ - и  $F_J$ -классов взвешенной матрицы  $\mathbf{X}$  могут быть отражены в порядке следования последних. В общем случае матрица  $\mathbf{X}$  инвариантна к перестановке только части строк и (или) столбцов, относящихся к  $F_I$ - и (или)  $F_J$ -классу соответственно. Множество столбцов  $F_J$ -класса матрицы  $\mathbf{X}$  при условии, что изменения в ней, вызванные их перестановкой, компенсируются перестановкой ее строк, составляет  $F$ -класс. Отношение порядка на множестве  $F$ -классов внутри каждого  $F_J$ -класса, определяемое лексикографическим порядком на множестве последовательностей из  $(i+1)$  их порядковых номеров, отражается также посредством порядка их следования в матрице  $\mathbf{X}$ .

**Пример 1.2.** Взвешенная матрица инцидентностей

$$\mathbf{X} = \begin{array}{c} \begin{array}{ccccc} 6 & 4 & 7 & 4 & 8 \\ 1 & 2 & 3 & 4 & 5 \end{array} \\ \left[ \begin{array}{cc|cc} 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ \hline 1 & 1 & 0 & 0 & 2 \\ 1 & 0 & 1 & 2 & 0 \\ \hline 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \end{array} \right] \begin{array}{l} 1 \\ 2 \\ \hline 3 \\ 4 \\ \hline 5 \\ 6 \end{array} \end{array} \left. \begin{array}{l} \\ \\ \hline \\ \hline \\ \hline \end{array} \right\} \begin{array}{l} I_1 \\ I_2 \\ I_3 \end{array}$$

отображает  $Q$ -структуру, структурные элементы которой разбиты на классы эквивалентности следующим образом:

$$I = \bigcup_{\alpha=1}^3 I_{\alpha}, \quad I_1 = \{1, 2\}, \quad I_2 = \{3, 4\}, \quad I_3 = \{5, 6\};$$

$$J = \bigcup_{\beta=1}^2 J_{\beta}, \quad J_1 = \{1, 2, 3\}, \quad J_2 = \{4, 5\};$$

$$\text{при } i = 1, 2 \quad D_i = 1, \quad L_i = L_{I_i} = \{1, 2, 3\};$$

$$\text{при } i = 3, 4 \quad D_i = 2, \quad L_i = \bigcup_{\delta_i=1}^2 L_{\delta_i}, \quad L_{I_i} = \{1, 2\}, \quad L_{2_i} = \{3\},$$

$$\text{при } i = 5, 6 \quad D_i = 1, \quad L_i = L_{I_i} = \{1, 2\};$$

Из матрицы видно, что  $F_J$ -класс  $J_{\beta} = J_1$  содержит два  $F$ -класса –  $J_{\beta_1} = J_{1_1} = \{1\}$  и  $J_{\beta_2} = J_{1_2} = \{2, 3\}$ , а  $F_J$ -класс  $J_{\beta} = J_2$  – один  $F$ -класс  $J_{\beta_1} = J_{2_1} = \{4, 5\}$ .

При выбранном задании  $Q$ -структуры очевидно следующее

**У т в е р ж д е н и е 1.1.**  *$Q$ -структура инвариантна к изменениям в ней, не нарушающим заданных отношений инцидентностей  $N$  (т.е.  $t_i, i = 1, 2, \dots, |I|$ ), эквивалентности  $F_I, F, F_i, i = 1, 2, \dots, |I|$ , и порядка  $R_I, R, R_i, i = 1, 2, \dots, |I|$ .*

В частности, возможна взаимная замена элементов  $Q$ -структуры или ее подструктур, удовлетворяющая требованиям утверждения 1.1. Из утверждения 1.1 вытекает

**С л е д с т в и е.** *Для того чтобы элементы множества  $I, J$  или  $L_i, i \in I$ , можно было менять местами без изменения  $Q$ -структуры, необходимо, чтобы они принадлежали общему  $F_I$ -,  $F$ - или  $F_i$ -классу соответственно.*

Это следует из требования утверждения 1.1 о сохранении отношений эквивалентности при преобразованиях  $Q$ -структуры.

Инвариантность структуры системы относительно ее преобразования называется симметрией [9;10]. Определение совокупности преобразований, оставляющих без изменения все структурные отношения объекта, стало руководящим принципом современной математики и физики, позволяющим глубоко проникнуть вовнутреннее строение материи. Рассмотренное здесь применение информации о симметриях в структурном синтезе систем показывает, что и для техники это понятие имеет большое значение.

Термин "симметрия" принято использовать ([10] и др.) для обозначения не только указанного свойства системы, но и операции симметричного ее преобразования, т.е. отображения на себя, осуществимого только при условии, что система обладает свойством симметричности.

Из сказанного следует, что симметрии присущи только системам, содержащим однотипные компоненты. Систему из однотипных компонентов принято называть однородной. Будем называть систему абсолютно однородной (абсолютно неоднородной) по составу ее компонентов, если она характеризуется только однотипными (разнотипными) компонентами, однотипными (разнотипными) внутренними связями и каждый компонент имеет только однотипные (разнотипные) внешние связи. Очевидно, что частный случай  $Q$ -структуры, когда каждый из классов эквивалентности ее множеств  $I, J, L_i, i = 1, 2, \dots, |I|$ , состоит из единственного элемента, соответствует абсолютно неоднородной системе. И наоборот, если каждое из этих множеств составляет один общий класс эквивалентности,  $Q$ -структура представляет абсолютно однородную систему.

Большинство технических систем занимает промежуточное положение между абсолютно однородными и абсолютно неоднородными. Это дает основание говорить о степени их однородности. Чем выше степень однородности, тем больше симметрий может иметь система. Но без однородности не может быть симметрий.

### 1.3. Симметрии $Q$ -структуры

Инвариантные преобразования структуры системы носят групповой характер и описываются с помощью группы автоморфизмов.

На каждом из множеств  $I, J, L_i, i = 1, 2, \dots, |I|$ , определена симметрическая группа подстановок. Подгруппу каждой из этих групп, переводящих множество эквивалентных элементов в себя, обозначим соответственно  $H_I, H_J, H_{L_i}, i = 1, 2, \dots, |I|$ . Рассматривая множества  $I, J, L_i, i = 1, 2, \dots, |I|$ , с заданными на них отношениями эквивалентности и порядка как математические структуры, будем считать подгруппы  $H_I, H_J, H_{L_i}, i = 1, 2, \dots, |I|$ , их группами автоморфизмов. Эти группы представим с учетом разбиений (1.5), (1.6), (1.9) множеств  $I, J, L_i, i \in I$ , в виде

$$H_I = \bigotimes_{\alpha=1}^A H_{I_\alpha}; \quad H_J = \bigotimes_{\beta=1}^B H_{J_\beta}; \quad H_{L_i} = \bigotimes_{\delta_i=1}^{D_i} H_{\delta_i},$$

т.е. через прямые суммы групп автоморфизмов  $H_{I_\alpha}, H_{J_\beta}, H_{\delta_i}$ , определенных на  $F_I, F_J, F_i$ -классах соответственно.

Симметричное преобразование  $Q$ -структуры (любой ее подструктуры) представим упорядоченной парой подстановок-автоморфизмов  $h = (h_I, h_J)$  :

$$Qh = Q(h_I, h_J) = Q, \quad (1.13)$$

где  $h_I, h_J$  – составляющие подстановки-автоморфизмы, определенные на множествах  $I, J$  соответственно. Возможны частные случаи, когда  $h = (h_I, e_J)$  или  $h = (e_I, h_J)$ , где  $e_I, e_J$  – тождественные подстановки на множествах  $I, J$  соответственно.

Пусть  $Q_\alpha$  – подструктура  $Q$ -структуры с  $i \in I_\alpha, \alpha \in \{1, 2, \dots, A\}$ , т.е.

$$Q_\alpha = \{J, q_{i_1}, q_{i_2}, \mathbf{K}, q_{i_k}, \mathbf{K}, q_{i_{|I_\alpha|}}\}, \quad i_k \in I_\alpha. \quad (1.14)$$

В матрице  $\mathbf{X}$  подструктуре  $Q_\alpha$  (или  $\alpha$ -подструктуре) соответствует подматрица (блок) с  $i \in I_\alpha$ . Пусть  $H$  – такое множество автоморфизмов  $Q$ -структуры, что для любого автоморфизма  $h \in H$  справедливо равенство

$$Q_\alpha h = Q_\alpha. \quad (1.15)$$

**Пример 1.3.** Пусть  $Q$ -структура задана матрицей

$$\mathbf{X} = \left[ \begin{array}{cc|cc|cc} 1 & 1 & 1 & 0 & 0 & \\ \hline 1 & 0 & 0 & 1 & 1 & \\ \hline 1 & 1 & 0 & 2 & 0 & \\ 1 & 0 & 1 & 0 & 2 & \\ \hline 0 & 2 & 0 & 3 & 0 & \\ 0 & 0 & 2 & 0 & 3 & \end{array} \right] \left. \begin{array}{l} \\ \\ \\ \\ \\ \\ \end{array} \right\} \begin{array}{l} \mathbf{X}_1 \\ \mathbf{X}_2 \\ \mathbf{X}_3 \end{array}.$$

Ее подструктуры  $Q_1$  с  $I_1 = \{1, 2\}$ ,  $Q_2$  с  $I_2 = \{3, 4\}$ ,  $Q_3$  с  $I_3 = \{5, 6\}$  представлены подматрицами  $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$  соответственно, выделенными в матрице  $\mathbf{X}$ . Автоморфизмы  $(h_I, h_J) \in H$  определяются по матрице  $\mathbf{X}$  как подстановки множества  $I$  ее строк и множества  $J$  столбцов, к которым она инвариантна. Из матрицы  $\mathbf{X}$  видно, что множество  $H$  – это подстановки  $(e_I, e_J)$  и  $((3, 4)(5, 6), (2, 3)(4, 5))$ , оставляющие неизменными блоки  $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$ .

**Утверждение 1.2.** Множество  $H$  – группа.

**Доказательство.** Пусть  $h_1, h_2 \in H$ , тогда с учетом (1.15) можно записать

$$Q_\alpha h_1 h_2 = (Q_\alpha h_1) h_2 = Q_\alpha h_2 = Q_\alpha. \quad (1.16)$$

Из (1.16) следует, что композиция  $h_1$  и  $h_2$  принадлежит  $H$ . Выполним автоморфное отображение  $h^{-1}$  правой и левой частей (1.15):

$$(Q_\alpha h) h^{-1} = Q_\alpha h^{-1}. \quad (1.17)$$

Так как  $h h^{-1} = e = (e_i, e_j)$  – тождественная подстановка, из (1.17) получаем

$$Q_\alpha = Q_\alpha h^{-1},$$

следовательно,  $h^{-1} \in H$ .

Таким образом, в множестве  $H$  содержится композиция любых двух элементов из  $H$ . Кроме того, вместе со всяким своим элементом в  $H$  входит элемент, обратный ему. Значит,  $H$  – группа.

Структурные элементы  $i \in I, j \in J, l_i \in L_i, i = 1, 2, \dots, |I|$ , являясь компонентами  $Q$ -структуры, т.е. находясь в связанном состоянии, лишь частично сохраняют свою взаимозаменяемость. Поэтому можно утверждать, что множество автоморфизмов  $H = \{h\}$   $Q$ -структуры – это только часть множества элементов группы  $H_{IJ} = H_I \otimes H_J$ . Так как групповые операции групп  $H, H_{IJ}$  совпадают и множество элементов  $H$  является подмножеством множества элементов группы  $H_{IJ}$ ,  $H$  – подгруппа группы  $H_{IJ}$ .

Как видим из (1.14),  $\alpha$ -подструктуры  $Q_\alpha, \alpha \in \{1, 2, \dots, A\}$ , пересекаются на множестве  $J$ . Пусть  $H_\alpha$  – группа автоморфизмов  $\alpha$ -подструктуры, а  $h_\alpha = (h_{I_\alpha}, h_{J_\alpha})$  – элемент этой группы. Назовем элементы  $h'_\alpha = (h'_{I_\alpha}, h'_{J_\alpha}) \in H_{\alpha'}$  и  $h''_\alpha = (h''_{I_\alpha}, h''_{J_\alpha}) \in H_{\alpha''}$  согласованными на множестве  $J$ , если  $h'_{J_\alpha} = h''_{J_\alpha}$ .

**Пример 1.4.** В группах автоморфизмов  $H_1$  (группа представлена образующими),  $H_2, H_3$  соответственно подматриц  $\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3$   $Q$ -структуры из примера 1.3

$$H_1 = \langle (e_{i_1}, e_j), (e_{i_1}, (2, 3)), (e_{i_1}, (4, 5)), ((1, 2), (2, 4)(3, 5)) \rangle$$

$$H_2 = \{(e_{i_2}, e_j), ((3, 4), (2, 3)(4, 5))\},$$

$$H_3 = \{(e_{i_3}, e_j), ((5, 6), (2, 3)(4, 5))\},$$

которые определяются непосредственно по этим подматрицам, согласованными между собой являются две тройки автоморфизмов:

$$(e_{i_1}, (2, 3)(4, 5)) \in H_1, ((3, 4), (2, 3)(4, 5)) \in H_2, ((5, 6), (2, 3)(4, 5)) \in H_3;$$

$$(e_{i_1}, e_j) \in H_1, (e_{i_2}, e_j) \in H_2, (e_{i_3}, e_j) \in H_3.$$

Объединением согласованных подстановок  $h_\alpha = (h_{I_\alpha}, h_{J_\alpha}), \alpha = 1, 2, \dots, A$ , назовем подстановку  $h = (h_I, h_J)$ , у которой составляющая  $h_J = h_{J_1} = h_{J_2} = \mathbf{L} = h_{J_\alpha} = \mathbf{L} = h_{J_A}$ , а составляющая  $h_I$  определяется как произведение расширений  $\overset{\mathbf{L}}{h_{I_\alpha}}$  на множество  $I$  составляющих  $h_{I_\alpha}, \alpha = 1, 2, \dots, A$ ,  $h_I = \overset{\mathbf{L}}{h_{I_1}} \overset{\mathbf{L}}{h_{I_2}} \mathbf{K} \overset{\mathbf{L}}{h_{I_\alpha}} \mathbf{K} \overset{\mathbf{L}}{h_{I_A}}$ .

**У т в е р ж д е н и е 1.3.** *Любой элемент  $h \in H$   $Q$ -структуры равен объединению согласованных на множестве  $J$  автоморфизмов ее подструктур  $Q_\alpha, \alpha = 1, 2, \dots, A$ .*

Справедливость утверждения 1.3 следует из того, что подструктуры  $Q_\alpha, \alpha = 1, 2, \dots, A$ , – это составные части  $Q$ -структуры, причем множество  $J$   $Q$ -структуры является общим для всех  $Q_\alpha$ -подструктур, а множество  $I$ , как следует из (1.5), – объединением множеств  $I_\alpha, \alpha = 1, 2, \dots, A$ .

Утверждение 1.3 устанавливает связь между группой автоморфизмов  $Q$ -структуры и группами автоморфизмов ее  $\alpha$ -подструктур. Из него следует, что любой элемент группы  $H$  может быть определен путем объединения тех автоморфизмов  $h_\alpha \in H_\alpha, \alpha = 1, 2, \dots, A$ , которые согласованы между собой на множестве  $J$ .

**Пример 1.5.** Результаты выявления симметрий, полученные в примерах 1.3, 1.4, подтверждают, что автоморфизм  $(e_j, e_j) \in H$   $Q$ -структуры из примера 1.3 – это объединение согласованных на множестве  $J = \{1, 2, 3, 4, 5\}$  автоморфизмов  $(e_{i_1}, e_j) \in H_1, (e_{i_2}, e_j) \in H_2, (e_{i_3}, e_j) \in H_3$   $\alpha$ -подструктур  $Q_1, Q_2, Q_3$  соответственно,

$$(e_j, e_j) = (e_{i_1}, e_j) \mathbf{U} (e_{i_2}, e_j) \mathbf{U} (e_{i_3}, e_j),$$

а автоморфизм  $((3, 4)(5, 6), (2, 3)(4, 5)) \in H$  – объединение автоморфизмов  $(e_{i_1}, (2, 3)(4, 5)) \in H_1, ((3, 4), (2, 3)(4, 5)) \in H_2, ((5, 6), (2, 3)(4, 5)) \in H_3$ ,

$$((3, 4)(5, 6), (2, 3)(4, 5)) =$$

$$= (e_{i_1}, (2, 3)(4, 5)) \mathbf{U} ((3, 4), (2, 3)(4, 5)) \mathbf{U} ((5, 6), (2, 3)(4, 5)).$$

Как видим, результат совпадает с полученным непосредственно по матрице  $\mathbf{X}$  в примерах 1.3, 1.4.

Пусть  $H_i$  – группа автоморфизмов  $i$ -подструктуры. Из утверждения 1.3 следует, что наиболее богатой является группа автоморфизмов 0-подструктуры, так как  $H_i = H_0 = H_{II}$ .

#### 1.4. Симметрии и изоморфизм $Q$ -структур

К множеству  $Q$ -структур, если каждая из них рассматривается как система, применимо понятие изоморфизма [11]. Две  $Q$ -структуры  $Q', Q'' \hat{=} M$  называются изоморфными, если существуют такие взаимно однозначные отображения  $\varphi: I' \rightarrow I''$  и  $\psi: J' \rightarrow J''$ , что из наличия отношений инцидентностей  $N'$ , эквивалентности  $F'_I, F'$  и порядка  $T'_I, T'$ , определенных на множествах  $I', J'$   $Q$ -структуры  $Q'$ , вытекает отношение инцидентностей  $N''$

$$I' N' J' \Rightarrow \varphi(I') N'' \psi(J')$$

вместе с отношениями эквивалентности  $F''_I, F''$  и порядка  $T''_I, T''$

$$\forall i, i' \in I' \quad iF'_I i' \Rightarrow \varphi(i)F''_I \varphi(i'), \quad iT'_I i' \Rightarrow \varphi(i)T''_I \varphi(i');$$

$$\forall j, j' \in J' \quad jF' j' \Rightarrow \psi(j)F'' \psi(j'), \quad jT' j' \Rightarrow \psi(j)T'' \psi(j'),$$

которые определены на множествах  $I''$  и  $J''$   $Q$ -структуры  $Q''$ , и наоборот:

$$I'' N'' J'' \Rightarrow \varphi^{-1}(I'') N' \psi^{-1}(J'');$$

$$\forall i, i' \in I'' \quad iF''_I i' \Rightarrow \varphi^{-1}(i)F'_I \varphi^{-1}(i'), \quad iT''_I i' \Rightarrow \varphi^{-1}(i)T'_I \varphi^{-1}(i');$$

$$\forall j, j' \in J'' \quad jF'' j' \Rightarrow \psi^{-1}(j)F' \psi^{-1}(j'), \quad jT'' j' \Rightarrow \psi^{-1}(j)T' \psi^{-1}(j').$$

Взаимно однозначное соответствие между множествами  $I', I'', (J', J'')$   $Q$ -структур  $Q', Q''$  можно выразить подстановкой  $g_I (g_J)$ , связывающей между собой элементы этих двух множеств. Таким образом, изоморфное отображение  $Q'$  на  $Q''$  описывается с помощью пары подстановок  $g = (g_I, g_J)$ . Условимся считать, что запись

$$Q'' = Q' g = Q' (g_I, g_J) \quad (1.18)$$

означает изоморфное отображение структуры  $Q'$  на структуру  $Q''$ .

Из (1.18) и из определения автоморфизма следует, что автоморфизм  $Q$ -структуры – это частный случай изоморфизма, изоморфизм на себя:

$$Q'' = Q' h = Q' (h_I, h_J) = Q'.$$

**У т в е р ж д е н и е 1.4.** Множество  $M$   $Q$ -структур с фиксированным набором структурных элементов разбивается группой автоморфизмов  $H_i$  любой их общей  $i$ -подструктуры  $Q_i$ ,  $i = 0, 1, 2, \dots, |I|$ , на классы эквивалентности-изоморфизма.

**Д о к а з а т е л ь с т в о.** Пусть  $M_i \in M$  – множество  $Q$ -структур, полученных путем подключения к  $i$ -подструктуре всеми возможными способами компонентов с номерами  $(i+1), (i+2), \dots, |I|$ , а  $h = (h_I, h_J) \in H_i$  – один из автоморфизмов подструктуры  $Q_i$ . Тогда для каждой структуры  $Q$  с отношением инцидентностей  $N$ , представленным совокупностью отображений  $t_i$  (1.2),  $i = 1, 2, \dots, |I|$ , во множестве  $M_i$  найдется структура  $Q'$  с отношением инцидентностей  $N'$ , представленным отображениями

$$t_{i'} = (t_{ih_I})h_J = \left( \begin{array}{c} 1, 2, \dots, |L_i| \\ j_{i1}h_J, j_{i2}h_J, \dots, j_{i|L_i}h_J \end{array} \right), \quad i' = 1, 2, \dots, |I|.$$

Переход от  $i$  к  $i' = ih_I$  можно трактовать как изменение последовательности подключения компонентов  $Q$ . После такого изменения отношение эквивалентности (1.11)

$$F = (\dots((F_J \mathbf{I} F_1) \mathbf{I} F_2) \dots) \mathbf{I} F_{|I|}$$

будет заменено отношением эквивалентности

$$F' = F_J \mathbf{I} \left( \mathbf{I}_{i'=1}^{|I|} F_{i'} \right) = (\dots((F_J \mathbf{I} F_{1h_I}) \mathbf{I} F_{2h_I}) \dots) \mathbf{I} F_{|I|h_I}.$$

В силу ассоциативности операции пересечения отношений эквивалентности разбиения множества  $J$  на  $F$ - и  $F'$ -классы идентичны: каждому  $F$ -классу  $J_\beta$  ( $F'$ -классу  $J_{\beta'}$ ) с элементами  $j \in J_\beta$  ( $j' \in J_{\beta'}$ ) однозначно



соответствует  $F'$ -класс  $J_\beta$ , ( $F$ -класс  $J_\beta$ ) с элементами  $j' = jh_J \in J_\beta$ , ( $j = j'h_J^{-1} \in J_\beta$ ).

Если порядковые номера  $F$ -классов  $Q$  в соответствии с (1.12) определяются последовательностью отношений порядков  $(T_J, T_{i_1}, T_{i_2}, \dots, T_{i_{|I|}})$  на множестве  $J$ , то порядковые номера  $F'$ -классов – последовательностью  $(T_J, T_{i_1 h_I}, T_{i_2 h_I}, \mathbf{K}, T_{i_{|I|} h_I})$  на этом множестве. Замена первой последовательности на вторую приводит отношение порядка  $T$  структуры  $Q$  к отношению порядка  $T'$  структуры  $Q'$ . Очевидно, что обратный переход от  $T'$  к  $T$  осуществляется с помощью соотношения  $i = i' h_I^{-1}$ .

Таким образом,  $Q$  и  $Q'$  можно рассматривать как две системы идентичных объектов  $I, J, L_i, i = 1, 2, \dots, |I|$ , с заданными на них отношениями соответственно  $N, F_I, F, T$  и  $N', F_I', F', T'$ , которые преобразованием  $h \in H_i$  приводятся в такое взаимно однозначное соответствие, что из наличия отношений  $N, F_I, F, T$  вытекают отношения  $N', F_I', F', T'$  и наоборот. Отсюда следует [11], что  $Q$  и  $Q'$  изоморфны.

Подобное деление множества  $Q$ -структур на классы эквивалентности-изоморфизма осуществляется любым элементом  $h \in H_i$  подструктуры  $Q_i$ . Поэтому каждая  $Q_i$ -подструктура в множестве  $M$  образует класс изоморфизма из  $|H_i|$  элементов. Утверждение доказано.

В процессе генерирования  $Q$ -структур по тривиальному алгоритму, рассмотренному в п.1.1, просматриваются все возможные варианты выборки  $\mathbf{j}_i$ ; следовательно, вместе с  $Q$ -структурой  $Q'$  будет построена и структура  $Q'h$ , где  $h$  – автоморфизм одной из их общих подструктур. Изоморфизм означает подобие в строении. В технике практический интерес представляют лишь оригинальные, а значит, неизоморфные структуры. Поэтому возникает задача отсечения изоморфных в процессе генерирования  $Q$ -структур. Для решения этой задачи полезно следствие из утверждения 1.4:

*С л е д с т в и е .* Пусть  $M_i', M_i''$  – множества  $Q$ -структур, построенных путем добавления всеми возможными способами к изоморфным  $i$ -подструктурам соответственно  $Q_i'$  и  $Q_i''$  компонентов с номерами  $(i+1), (i+2), \dots, |I|$ . Элементы этих множеств попарно изоморфны.

Из следствия видно, что важнее не допускать построения изоморфных  $i$ -подструктур,  $i \in I$ , чем решать чрезвычайно трудоемкую задачу распознавания изоморфных с целью их исключения из множества  $M$  всех синтезированных  $Q$ -структур.

Утверждение 1.4 позволяет проследить механизм образования изоморфных  $Q$ -структур при использовании тривиального алгоритма. Пусть  $h = (h_I, h_J) \in H_{(i-1)}$  – автоморфизм  $(i-1)$ -подструктуры, а  $h_i \in H_{L_i}$  –  $i$ -го компонента. Тогда кроме  $Q_i = Q_{(i-1)} \mathbf{U} q_i = Q_{(i-1)} \mathbf{U} q_{i0} t_i$  появится изоморфная  $Q_i$   $i$ -подструктура  $Q_i'$ :

$$Q_i' = Q_{(i-1)} \mathbf{U} q_i' = Q_{(i-1)} \mathbf{U} q_{i0} h_i t_i h_J = Q_{(i-1)} \mathbf{U} q_{i0} t_i'. \quad (1.19)$$

Выражение (1.19) дает основание рассматривать отображение

$$t_i' = h_i t_i h_J \quad (1.20)$$

как шаг конструктивного процесса, эквивалентный встречавшемуся ранее шагу  $t_i$  по симметриям  $h_i$  и  $h_J$ , т.е. симметричный ему. Значит, исключение части изоморфных  $i$ -подструктур можно заменить исключением части шагов конструктивного процесса. Это та часть  $i$ -подструктур  $Q_i'$ , отображение которых на  $i$ -подструктуру  $Q_i$  сопровождается отображением  $i$ -го компонента первой на  $i$ -й же компонент второй при отображении  $j$ -го компонента на компонент с номером  $j' = j h_j$ .

В отличие от автоморфизмов  $h \in H_{(i-1)}$  автоморфизмы  $h_i \in H_{L_i}$  приводят не к изоморфным, а к повторяющимся  $Q$ -структурам, так как подстановки  $g_{L_i}'(n) = g_{L_i}(n) h_i$  в (1.4) меняют местами только элементы  $l_i \in L_i$ , принадлежащие общему классу эквивалентности  $L_{\delta_i}$ ,  $\delta_i \in \{1, 2, \dots, D_i\}$ . Исключив эти подстановки в (1.4), (1.3), получим

$$q_i = q_{i0} t_i(n, k), \quad (1.21)$$

где  $t_i(n, k)$  – отображение  $t_i$  с  $n$ -м вариантом мультимножества  $L_i$ ,  $n = /L_i/!(|L_{\delta_1}|!|L_{\delta_2}|!\dots|L_{D_i}|!)$ , в котором кратными считаются элементы каждого класса эквивалентности  $L_{\delta_i}$ ,  $\delta_i = 1, 2, \dots, D_i$ . Тогда вместо (1.20) имеем

$$t_i' = t_i h_J. \quad (1.22)$$

С учетом (1.21), (1.22) проследим появление изоморфных  $Q$ -струк-

тур при добавлении к  $(i-1)$ -подструктуре компонентов с номерами  $i, i+1, i+2, \dots, |I|$ . Кроме структуры  $Q = Q_{(i-1)} \prod_{s=i}^{|I|} q_s$  на одном из этапов порождения

элементов множества  $M$  появится структура  $Q'$ :

$$\begin{aligned} Q' &= Qh = \\ &= Q_{(i-1)} \prod_{s=i}^{|I|} q_s' = Q_{(i-1)} \prod_{s=i}^{|I|} q_{s0} t_s'(n, k) = Q_{(i-1)} \prod_{s=i}^{|I|} q_{s0} t_s(n, k) h = Q_{(i-1)} \prod_{s=i}^{|I|} q_s h, \end{aligned}$$

где

$$q_s' = q_s h = \left( \begin{array}{c} sh_I \\ j_{s1} h_J, j_{s2} h_J, \mathbf{K}, j_{s|L_i|} h_J \end{array} \right), \quad (1.23)$$

$$h = (h_I, h_J) \in H_{(i-1)}.$$

Таким же образом каждый автоморфизм  $h \in H_{(i-1)}$  каждой  $(i-1)$ -подструктуры,  $i \in I$ , порождает изоморфную  $Q$ -структуру. Из (1.23) следует, что в данном случае изоморфное отображение  $i$ -подструктуры  $Q_i$  на  $i$ -подструктуру  $Q_i'$  сопровождается отображением  $i$ -го компонента первой на компонент с номером  $i' = ih_I$  и одновременно  $\uparrow j$ -го компонента на компонент с номером  $j' = jh_j$ .

**Пример 1.6.** Структура  $Q$  с матрицей

$$\mathbf{X} = \left\| \begin{array}{ccccc} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{array} \right\|$$

приводится к структуре  $Q'$  с матрицей

$$\mathbf{X}' = \left\| \begin{array}{ccccc} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{array} \right\|$$

с помощью автоморфизма

$$h = (h_I, h_J) = \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 4 & 3 & 2 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 5 & 4 & 3 & 2 & 1 \end{pmatrix} \right) \in H_i = H_0 = H_{IJ}.$$

Из составляющей  $h_I$  этого автоморфизма видно, что при переходе под его действием от структуры  $Q$  к структуре  $Q'$  меняются местами первый и пятый, второй и четвертый компоненты множеств  $I$  и  $J$ . Нетрудно убедиться, что для такого преобразования необходима абсолютная однородность каждого из множеств  $I, J, L_i, i = 1, 2, \dots, 5$ . При нарушении однородности любого из них  $Q$ -структуры с матрицами  $X$  и  $X'$  перестают быть изоморфными.

Итак, изоморфные  $Q$ -структуры можно рассматривать как эквивалентные по симметриям их общей подструктуры. Осознание роли симметрий в порождении изоморфных структур позволяет сформулировать общий принцип построения множества неизоморфных структур: на каждом шаге конструктивного процесса необходимо отбирать только те подключения  $i$ -го компонента к  $(i-1)$ -подструктуре, при которых результирующая  $i$ -подструктура не является эквивалентной ни одной из полученных ранее  $i$ -подструктур ни по симметриям структуры  $Q_i$  или ее подструктур, ни по симметриям  $i$ -го компонента. Этим принципом мы пользуемся интуитивно, не задумываясь, при конструировании простейших структур. Например, нам не приходит в голову вслед за структурой, отображаемой графом рис.1.2, *а*, рассматривать структуру, соответствующую графу рис.1.2, *б* или рис 1.2, *в*, или менять между собой места подключения внешних выводов компонента, если они однотипны. Трудности с реализацией указанного принципа использования информации о симметриях возникают при решении более сложных задач структурного синтеза, а также в случае привлечения вычислительной

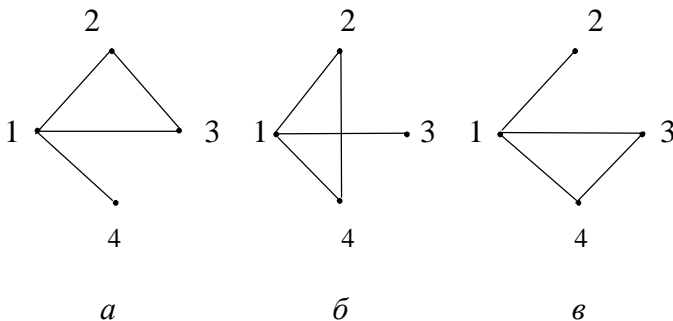


Рис. 1.2

техники к их решению, что неизбежно в связи с очень большой трудоемкостью этих задач.

Природа и проявления симметрий чрезвычайно разнообразны. Поэтому рассмотренный здесь принцип их использования для отсекаания изоморфных структур может быть сформулирован иначе: если в процессе структурного синтеза систем появляются изоморфные варианты, значит не все виды симметрий учтены и использованы для коррекции конструктивного процесса порождения их структур. Такая трактовка принципа отсекаания изоморфных структур с использованием информации о симметриях может быть полезна при учете специфики каждого конкретного приложения, проявляющейся, в частности, в превращении неважно-заменимых на первый взгляд структурных элементов во взаимозаменяемые в некотором смысле.

## 1.5. Синтез неизоморфных $Q$ -структур

### 1.5.1. Применение метода решета

Известная [3] процедура построения множества неизоморфных комбинаторных объектов (решето, отбраковывающее изоморфные объекты) состоит в следующем. Пусть  $O = \{ o_1, o_2, \dots, o_m \}$  – множество всех объектов. Сначала в нем находят объекты, которые изоморфны  $o_1$ , и исключают их из множества  $O$ . Затем находят изоморфные первому объекту среди оставшихся после исключения (т.е. второму после  $o_1$ ) и удаляют их из множества  $O$ . Продолжая в том же духе, получаем максимальное множество неизоморфных объектов, составленное из неисключенных элементов множества  $O$ .

Если применить указанную процедуру к  $Q$ -структурам, понадобилось бы каждую подвергнуть всем  $|I|!|J|!$  возможным преобразованиям  $g = (g_I, g_J)$  и после каждого преобразования сравнить с каждой из уже найденных неизоморфных  $Q$ -структур. Утверждение 1.4 и следствие из него позволяют разработать более эффективную процедуру построения неизоморфных  $Q$ -структур с использованием информации о симметриях структуры и ее подструктур. Из них видно, что наиболее целесообразно начать исключение изоморфных вариантов с  $i$ -подструктур,  $i = 1, 2, \dots, |I|$ . Исключение каждой изоморфной  $i$ -подструктуры интерпретируется на дереве вариантов (см. рис. 1.1) отсеканием соответствующей вершины  $i$ -го яруса вместе с висящим на нем поддеревом. Множество отсекаемых при

этом изоморфных  $Q$ -структур интерпретируется множеством концевых вершин такого поддерева.

Исключение части изоморфных  $i$ -подструктур, как было отмечено в п. 1.4, может быть заменено исключением части шагов конструктивного процесса, определяемых соотношением (1.20). Переходом от операторов  $t_i$ , определенных по (1.4), к операторам  $t_i(n, k)$  (см. п. 1.4) такое исключение частично реализовано. Из построения операторов  $t_i$  следует, что исключение их из конструктивного процесса позволяет отсечь ту часть изоморфных  $i$ -подструктур  $Q_i'$ , при отображении которых на  $Q_i$   $i$ -й компонент первой отображается на  $i$ -й компонент второй.

Теперь отсечению подлежат только такие подструктуры  $Q_i'$ , изоморфное отображение которых на  $Q_i$  сопровождается отображением  $i$ -го компонента на один из компонентов с номером  $i' < i$ . Если использовать для этой цели метод решета,  $i$ -подструктуру понадобилось бы подвергнуть в общем случае  $i! |J|!$  проверочным подстановкам  $g = (g_I, g_J)$ . Вместо таких подстановок достаточно ограничиться, как следует из п. 1.3, автоморфизмами  $i$ -подструктуры и ее составных частей, реализующими указанное отображение. Об этих автоморфизмах известно, что каждый из них 1) содержится в группе  $H_{IJ}$ , 2) не оставляет неизменным значение  $i$ , 3) переводит  $i$ -й компонент в  $i'$ -й,  $i' < i$ ,  $i, i' \in I_\alpha$ ,  $\alpha \in \{1, 2, \dots, A\}$ , 4) фиксирует  $[10]$  подмножество  $I' = \{(i+1), (i+2), \dots, |I|\}$  множества  $I$ . Пусть  $H^{(i)}$  – подмножество элементов  $H_{IJ}$ , удовлетворяющих условиям 2), 3), 4);  $H_{\{i\}}$  – подгруппа группы  $H_{IJ}$ , фиксирующая  $i$ ;  $H_{\{i'\}}$  – подгруппа группы  $H_{IJ}$ , фиксирующая подмножество  $I'$ . Тогда

$$H^{(i)} = H_{\{I'\}} \setminus H_{\{i\}}. \quad (1.24)$$

Очевидно, что  $|H^{(i)}| < i! |J|!$ . Таким образом, использование информации о симметриях  $i$ -подструктуры и ее частей позволяет повысить эффективность метода решета в применении к  $Q$ -структурам. Как будет ясно из дальнейших рассуждений, множество  $H^{(i)}$  проверочных подстановок-автоморфизмов можно дополнительно сократить.

### 1.5.2. Применение идеи канонического упорядочения комбинаторных объектов

Выделим два варианта постановки задачи структурного синтеза систем, представляющих практический интерес: полный перебор вариантов с заданным набором структурных элементов и выборочное построение структур множества  $M$ . К первому варианту обращаются при поиске глобально оптимального решения в пространстве структур. Но он зачастую бывает неприемлемым в технике при слишком больших множествах  $M$ . В этом случае остается доступным паллиативный вариант локальной оптимизации на некотором подмножестве множества  $M$ , которое определяется чаще всего случайным образом. Такой подход характерен для структурного синтеза с применением идей эволюции и адаптации ([12,13] и др.), для различных эвристических методов структурного синтеза. Порождение случайного подмножества множества  $M$  также сводится к задачам порождения случайных подмножеств множества  $J$ , для решения которых можно воспользоваться методами и алгоритмами, приведенными в [3] и других работах.

При полном переборе для отсеечения изоморфных структур воспользуемся идеей канонического упорядочения комбинаторных объектов [1]. Если применяется метод решета, каноническое упорядочение освобождает от необходимости сопоставления каждой вновь сгенерированной  $Q$ -структуры ( $i$ -подструктуры) с полученными ранее и, следовательно, от необходимости хранения множества ранее сгенерированных  $Q$ -структур ( $i$ -подструктур).

Прежде всего воспользуемся введенным порядком на множествах  $I, J, L_i, i \in I$ , для повышения направленности поиска неизоморфных  $Q$ -структур. Пусть  $M'$  – подмножество множества  $M$   $Q$ -структур, у которых структурные элементы, принадлежащие общему классу эквивалентности, пронумерованы подряд,  $F_I$ -классы  $I_\alpha$  упорядочены по возрастанию их порядковых номеров  $\alpha \in \{1, 2, \dots, A\}$ , а  $F_J$ -классы  $J_\beta$  – по возрастанию  $\beta \in \{1, 2, \dots, B\}$ . Из определения множества  $M'$  следует, что оно совпадает с множеством  $M$  только в случае абсолютно однородной  $Q$ -структуры.

**У т в е р ж д е н и е 1.5.** *Подмножество  $M' \subseteq M$  содержит по крайней мере по одному представителю от каждого класса эквивалентности-изоморфизма множества  $M$ .*

**Д о к а з а т е л ь с т в о.** Изоморфные варианты  $Q$ -структуры различаются между собой только нумерацией элементов  $i \in I$  и (или)  $j \in J$ . Каждый класс эквивалентности-изоморфизма в множестве  $M$  состоит из  $Q$ -структур, структурные элементы  $i \in I$  и  $j \in J$  которых

пронумерованы всеми возможными способами, в том числе и способами, определяющими принадлежность соответствующих  $Q$ -структур к множеству  $M'$ . Значит, элементы множества  $M'$  обязательно имеются в каждом классе эквивалентности-изоморфизма множества  $M$ .

Из утверждения 1.5 следует, что при поиске неизоморфных  $Q$ -структур, если они не являются абсолютно однородными, вместо множества  $M$  достаточно просмотреть только его подмножество  $M' \subset M$ . В процессе генерирования  $Q$ -структур просмотр только элементов множества  $M'$  достигается выбором соответствующей нумерации элементов  $j \in J$  и согласованием последовательности подключения элементов  $i \in I$  с нумерацией этих элементов во множестве  $M'$ . Для этого элементы каждого  $F_J$ -класса  $J_\beta$ ,  $\beta \in \{1, 2, \dots, B\}$ , множества  $J$  пронумеруем подряд и согласуем с этой нумерацией нумерацию  $F_J$ -классов так, чтобы выполнялось условие

$$\forall j' \in J_{\beta'}, \forall j'' \in J_{\beta''} \quad \beta' \langle \beta'' \Leftrightarrow j' \langle j'', \quad (1.25)$$

т.е. чтобы любой элемент  $j'$  старшего  $F_J$ -класса  $J_{\beta'}$  был старше любого элемента  $j''$  из нестаршего  $F_J$ -класса  $J_{\beta''}$ . Выберем такую последовательность подключения структурных элементов  $i \in I$ , чтобы сначала сформировать  $\alpha$ -подструктуру с  $\alpha = 1$ , затем – с  $\alpha = 2$  и т.д.

При таком выборе последовательности формирования  $Q$ -структуры появляется возможность сократить число проверочных подстановок, определенных соотношением (1.24). Пусть  $Q_{\alpha i}$  – подструктура с  $i \in I_\alpha$ , для ее проверки достаточно ограничиться теми автоморфизмами из  $H^{(i)}$ , которые представляют собой подстановки элементов  $i' \in I_\alpha$ ,  $i' \leq i$  в сочетании с подстановками элементов  $j \in J_\beta$ ,  $\beta \in \{1, 2, \dots, B\}$ , согласованными с автоморфизмами  $\alpha'$ -подструктур,  $\alpha' = 1, 2, \dots, \alpha-1$ . Из определений  $H^{(i)}$ ,  $H_\alpha$  следует, что множество таких подстановок составляет часть элементов группы автоморфизмов  $H_{\alpha 0}$   $\alpha$ -подструктуры в 0-состоянии. Обозначим это множество  $H_{\alpha i 0}$ . Оно совпадает с множеством  $H^{(i)}$  только при  $\alpha = 1$ . Элементы  $h \in H_{\alpha i 0}$  должны быть согласованы (см. п. 1.3) с автоморфизмами всех  $\alpha'$ -подструктур,  $\alpha' = 1, 2, \dots, \alpha-1$ , поэтому по мере увеличения  $\alpha'$  множество их составляющих  $h_j$  постепенно становится беднее.

Далее введем порядок на множестве вариантов оператора  $t_i$ , опре-



деляющих варианты  $i$ -подструктуры  $Q_i$  при неизменной их общей  $(i-1)$ -подструктуре. Для этого представим отношения эквивалентности  $F$  и порядка  $T$  на множестве  $J$   $i$ -подструктуры в виде

$$F = F_i \mathbf{I} F',$$

$$T = T_i \mathbf{U} T',$$

где  $F_i$  – отношение эквивалентности на множестве  $L_i$ , индуцированное с помощью отображения  $t_i$  на множество  $J$ ;  $F'$  – отношение эквивалентности, индуцированное разбиением множества  $J$   $(i-1)$ -подструктуры на  $F$ -классы,

$$F' = F_J \mathbf{I} \left( \mathbf{I}_{i'=1}^{i-1} F_{i'} \right),$$

$T_i$  – отношение порядка на множестве элементов  $L_i$ , индуцированное на элементы множества  $J$ ;  $T'$  – отношение порядка на элементах множества  $J$   $(i-1)$ -подструктуры,

$$T' = T_J \mathbf{U} \left( \mathbf{U}_{i'=1}^{i-1} T_{i'} \right).$$

Элементы выборки  $\mathbf{j}_i$ , будучи одновременно элементами множества  $J$ , как и последние, делятся на  $F$ -классы. Обозначим  $F$ -класс выборки  $\mathbf{j}_i$  символом  $\mathbf{j}_{i\beta}$ . В соответствии с определением, данным в п.1.2,  $F$ -класс выборки  $\mathbf{j}_i$  составляют элементы, для которых одновременно выполняются отношение эквивалентности  $F'$  и индуцированное отображением  $t_i$  отношение эквивалентности  $F_i$ . На  $F$ -классах выборки  $\mathbf{j}_i$  определены два отношения порядка  $T'$  и  $T_i$ . Отдавая предпочтение порядку  $T'$  (см. п.1.2), порядок на  $F$ -классах  $\mathbf{j}_{i\beta}$  будем считать заданным лексикографическим порядком на соответствующих упорядоченных парах  $(T', T_i)$ .

В общем случае элементы выборок  $\mathbf{j}'_i, \mathbf{j}''_i$  необходимо сравнивать между собой с учетом их лексикографического упорядочения в алфавите  $(1, 2, \dots, |J|)$  и порядков  $T', T_i$ , определенных соответственно на  $F'$  - и

$F_i$ -классов, к которым они относятся. Однако при выбранном способе согласования нумерации  $F_J$ -классов  $J_\beta$  с нумерацией элементов множества  $J$ , определенных соотношением (1.25), эта задача несколько упрощается. Учтем, что это соотношение сохраняется при дроблении  $F_J$ -классов на  $F$ -классы в силу упорядоченности элементов каждого  $F_J$ -класса. Значит, информация о порядке следования  $F$ -классов содержится в самом элементе  $j$  выборки  $\mathbf{j}_i$ . Это позволяет, сравнивая элемент  $j'$  из  $\delta_i'$ -го  $F_i$ -класса выборки  $\mathbf{j}_i'$  с элементом  $j''$  из  $\delta_i''$ -го  $F_i$ -класса выборки  $\mathbf{j}_i''$ , определить порядок на этих элементах соотношением

$$j' \mathbf{f} j'' \Leftrightarrow (j', \delta_i') \overset{lex}{\succ} (j'', \delta_i''). \quad (1.26)$$

Неравенство (1.26) означает, что элемент  $j' \in \mathbf{j}_i'$  старше элемента  $j'' \in \mathbf{j}_i''$ , если либо первый лексикографически старше второго в алфавите  $(1, 2, \dots, |J|)$ , либо при равенстве  $j' = j''$  элемент  $\delta_i'$  лексикографически старше элемента  $\delta_i''$  в алфавите  $(1, 2, \dots, D_i)$ .

Сравнивая выборки  $\mathbf{j}_i'$ ,  $\mathbf{j}_i''$ , будем считать ту из них старше, в которой раньше встречается старший элемент  $j$  при просмотре ее элементов слева направо. Введенным порядком на выборках  $\mathbf{j}_i$  однозначно определяется порядок на соответствующих им операторах  $t_i$ :

$$t_i' \mathbf{f} t_i'' \Leftrightarrow \mathbf{j}_i' \mathbf{f} \mathbf{j}_i''.$$

Упорядочив операторы  $t_i$ , введем понятие их каноничности. Объявим каноническим старший оператор  $t_i$  среди множества эквивалентных ему по  $h \in H_{(i-1)}$  и  $h_i \in H_{L_i}$ . Тогда для отбора единственного среди множества таких операторов можно использовать (1.20) в виде предиката

$$P(t_i) \equiv (\forall h_i \in H_{L_i}, \forall h = (h_I, h_J) \in H_{(i-1)}) \quad t_i \mathbf{f} h_i t_i h_J \vee t_i = h_i t_i h_J, \quad (1.27)$$

принимающего истинное значение только для канонического  $t_i$ .

Если перебор вариантов  $t_i$  выполняется по (1.21), предикат (1.27) с учетом (1.22) принимает вид

$$P(t_i) \equiv (\forall h = (h_I, h_J) \in H_{(i-1)}) \quad t_i \mathbf{f} t_i h_J \vee t_i = t_i h_J. \quad (1.28)$$

Введенным порядком на множестве операторов  $t_i$  задан порядок на множестве вариантов  $i$ -подструктуры  $Q_i$  при общей  $(i-1)$ -подструктуре: чем старше оператор  $t_i$ , тем старше соответствующая  $i$ -подструктура. На множестве  $\alpha$ -подструктур,  $\alpha \in \{1, 2, \dots, A\}$ , также уже установлен порядок. Упорядочим теперь  $Q_i$ -подструктуры с  $i \in I_\alpha$ ,  $\alpha \in \{1, 2, \dots, A\}$ . Этот порядок также индуцируется порядком на множестве операторов  $t_i$ :

$$Q_i \mathbf{f} Q_i'' \Leftrightarrow \exists i^* \in \{1, 2, \dots, i\}: \forall i \langle i^* \quad t_i' = t_i'', \quad t_{i^*}' \mathbf{f} t_{i^*}'' \rangle, \quad (1.29)$$

$$i \in I_\alpha, \quad \alpha \in \{1, 2, \mathbf{K}, A\}.$$

Объявим канонической старшую подструктуру  $Q_i$  среди множества эквивалентных ей по симметриям всех ее подструктур. Тогда для отбора неизоморфных  $i$ -подструктур построим предикат

$$P(Q_i) \equiv (\forall h \in H^{(i)} \quad Q_i \mathbf{f} Q_i h \vee Q_i = Q_i h),$$

принимающий истинное значение только для канонической  $i$ -подструктуры. Эту проверку каноничности  $Q_i$  можно упростить при выбранной последовательности подключения компонентов  $i \in I$  и при проверке каждой  $i$ -подструктуры на каноничность. Пусть  $\alpha'$ -подструктуры с  $\alpha' = 1, 2, \dots, (\alpha-1)$  являются каноническими. Тогда каноничность  $i$ -подструктуры  $Q_{\alpha i}$  с  $i \in I_\alpha$  достаточно проверить отдельно, заменив предикат  $P(Q_i)$  предикатом

$$P(Q_{\alpha i}) \equiv (\forall h \in H_{\alpha i 0} \quad Q_{\alpha i} \mathbf{f} Q_{\alpha i} h \vee Q_{\alpha i} = Q_{\alpha i} h). \quad (1.30)$$

Рассмотрим пример применения предикатов (1.28), (1.30) для отсечения изоморфных  $i$ -подструктур.

**Пример 1.7.** Пусть решается задача порождения вариантов  $Q$ -структуры, характеризующейся следующим набором структурных элементов:

$$I = \{I_1, I_2\}, \quad I_1 = \{1, 2, 3\}, \quad L_i = L_{I_i} = \{1, 2, 3\}, \quad i = 1, 2, 3;$$

$$I_2 = \{4, 5\}, \quad L_i = \{L_{I_i}, L_{2_i}\}, \quad L_{1_i} = \{1, 2\}, \quad L_{2_i} = \{3\}, \quad i = 4, 5;$$

$$J = \{J_1, J_2, J_3\}, \quad J_1 = \{1, 2, 3\}, \quad J_2 = \{4, 5\}, \quad J_3 = \{6\}.$$

Из конструктивного процесса решения поставленной задачи выделим этап перебора вариантов подключения пятого компонента к 4-подструктуре, характеризующейся следующими отображениями:

$$t_1 = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 6 \end{pmatrix}, \quad t_2 = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 4 & 6 \end{pmatrix}, \quad t_3 = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 5 & 6 \end{pmatrix}, \quad t_4 = \begin{pmatrix} 3 & 1 & 2 \\ 3 & 5 & 4 \end{pmatrix}.$$

Не перечисляя все варианты оператора  $t_5$ , определяющие возможные способы подключения пятого компонента к системе (всего  $\{ |J| / (|L_i| (|J| - |L_i|)) \} \times \{ (|L_{i_1}| + |L_{i_2}|) / (|L_{i_1}| |L_{i_2}|) \} = [6! / (3!3!)] \cdot [(3! / (2!1!))] = 60$  вариантов), рассмотрим отсечение вариантов

$$t'_{51} = \begin{pmatrix} 1 & 3 & 2 \\ 3 & 4 & 6 \end{pmatrix}; \quad t'_{52} = \begin{pmatrix} 3 & 1 & 2 \\ 3 & 5 & 6 \end{pmatrix}.$$

Группа симметрий 4-подструктуры

$$H_4 = \{ (e_I, e_J), ((2, 3), (1, 2)(4, 5)) \}.$$

Каноничность операторов  $t'_{51}$ ,  $t'_{52}$  проверяем по предикату (1.28). Разумеется, тождественное преобразование  $(e_I, e_J)$  опускаем. Тогда остается проверка с помощью подстановки  $h_J = (1, 2)(4, 5)$ :

$$t''_{51} = t'_{51} h_J = \begin{pmatrix} 1 & 3 & 2 \\ 3 & 4 & 6 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 5 & 4 & 6 \end{pmatrix} = \begin{pmatrix} 1 & 3 & 2 \\ 3 & 5 & 6 \end{pmatrix};$$

$$t''_{52} = t'_{52} h_J = \begin{pmatrix} 3 & 1 & 2 \\ 3 & 5 & 6 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 5 & 4 & 6 \end{pmatrix} = \begin{pmatrix} 3 & 1 & 2 \\ 3 & 4 & 6 \end{pmatrix}.$$

По отображениям  $t'_{51}$ ,  $t''_{51}$  строим последовательности упорядоченных двоек  $(j', \delta'_i)$  для элементов соответствующих выборок  $\mathbf{j}'_{51}$  и  $\mathbf{j}''_{51}$ :  $(3, 1), (4, 2), (6, 1)$  и  $(3, 1), (5, 2), (6, 1)$ . Сравниваем их попарно. Так как  $(4, 2) \overset{lex}{\succ} (5, 2)$ ,  $t'_{51} \mathbf{f} t''_{51}$ . Следовательно, оператор  $t'_{51}$  – канонический. Отображения  $t'_{52}$ ,  $t''_{52}$  сравниваем по аналогичным последовательностям:  $(3, 2), (5, 1), (6, 1)$  и  $(3, 2), (4, 1), (6, 1)$ . Так как  $(5, 1) \overset{lex}{\prec} (4, 1)$ ,  $t'_{52} \mathbf{p} t''_{52}$ . Следовательно, оператор  $t'_{52}$  – не канонический.

Теперь проверим на каноничность 5-подструктуру, соответствующую  $t_i = t'_{51}$ , по предикату (1.30), считая, что каноничность  $\alpha$ -подструктуры с  $\alpha = 1$  к этому моменту установлена. Множество  $H_{\alpha i 0} = H_{250}$ , как следует из его определения, составляют, кроме единичной подстановки, автоморфизмы  $h' = (h'_I, h'_J) = ((4, 5), (1, 2)(4, 5))$ ,  $h'' = (h''_I, h''_J) = ((4, 5), e_J)$ . Проверяем действие авто-

морфизма  $h' \in H_{250}$  на операторы  $t_4$  и  $t_5 = t_{51}'$ , относящиеся к  $\alpha$ -подструктуре с  $\alpha = 2$ :

$$t_4' h' = (t_4 h_I) h_J = \begin{pmatrix} 3 & 1 & 2 \\ 3 & 4 & 5 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 5 & 4 & 6 \end{pmatrix} = \begin{pmatrix} 3 & 1 & 2 \\ 3 & 5 & 4 \end{pmatrix} = t_5'';$$

$$t_5' h' = (t_4 h_I) h_J = \begin{pmatrix} 1 & 3 & 2 \\ 3 & 5 & 6 \end{pmatrix} = t_4''.$$

Положив  $i = 4$ , сравниваем  $t_4$  и  $t_4''$  по последовательностям упорядоченных пар  $(3, 2)$ ,  $(4, 1)$ ,  $(5, 1)$  и  $(3, 1)$ ,  $(5, 2)$ ,  $(6, 1)$ . Так как  $(3, 2) \stackrel{lex}{\prec} (3, 1)$ ,  $t_4' \mathbf{p} t_4''$ . Следовательно,  $i$ -подструктура  $Q_5$  не является канонической.

Идея упорядочения  $Q$ -структур полезна и при решении задач выборочного построения структур из множества  $M$ . На основе утверждения 1.5 достаточно генерировать только элементы множества  $M'$  и для проверки каноничности их использовать предикаты, аналогичные (1.27), (1.28), (1.30):

$$P(t_i) \equiv (\forall h_i \in H_{L_i}, \forall h = (h_I, h_J) \in H_{(i-1)} \quad h_i t_i h_J \in S_{t_i});$$

$$P(t_i) \equiv (\forall h = (h_I, h_J) \in H_{(i-1)} \quad t_i h_J \in S_{t_i});$$

$$P(Q_{\alpha i}) \equiv (\forall h \in H_{\alpha i 0} \quad Q_{\alpha i} h \in S_{Q_{\alpha i}}).$$

Здесь  $S_{t_i}$  ( $S_{Q_{\alpha i}}$ ) – множество операторов  $t_i$  ( $i$ -подструктур  $Q_{\alpha i}$ ), уже использованных ранее в конструктивном процессе.

Очевидно, что в этом случае хранение множеств  $S_{t_i}$  и  $S_{Q_{\alpha i}}$  неизбежно. Задача облегчается, если имеются способы компактного кодирования элементов этих множеств. В данном случае снова полезно представление оператора  $t_i$  вектором, а  $i$ -подструктуры  $Q_{\alpha i}$  – последовательностью векторов или матрицей. Тогда для хранения применимы методы их компактного кодирования [14].

Действие рассмотренных предикатов каноничности в конструктивном процессе порождения  $Q$ -структур интерпретируется на дереве вариантов (см. рис.1.1) дополнительным отсечением на каждом ярусе ряда вершин вместе с висящими на них поддеревьями.

### 1.5.3. Построение признаков каноничности

Проверки каноничности оператора  $t_i$  и  $i$ -подструктуры  $Q_i$  по предикатам, перечисленным в п. 1.5.2, являются переборными процедурами. Поэтому метод отсечения изоморфных  $Q$ -структур с применением (1.27), (1.28), (1.30), будучи экономичнее обычного метода решета, все-таки остается достаточно трудоемким. Однако эти предикаты допускают построение ряда беспереборных, а значит, и сравнительно нетрудоемких признаков каноничности. Для разработки наиболее эффективных признаков полезно использовать особенности как генерируемых  $Q$ -структур, так и их групп автоморфизмов. Примеры разновидностей  $Q$ -структур и использование их свойств для повышения эффективности отсечения изоморфных вариантов рассмотрены в последующих главах. Здесь остановимся на примерах построения признаков каноничности с использованием некоторых наиболее общих свойств симметрий  $Q$ -структуры.

$Q$ -Структура может рассматриваться в общем случае как соединение более простых подструктур со своими группами автоморфизмов. Группа автоморфизмов  $Q$ -структуры формируется из групп автоморфизмов ее подструктур путем применения к ним операций, определенных на группах подстановок [2]: декартового произведения, сплетения и других. Поэтому множества  $H_{(i-1)}$ ,  $H_{\alpha i 0}$ ,  $i \in I$ ,  $\alpha \in \{1, 2, \dots, A\}$ , в общем случае имеют сложное строение. В частности, каждое из них может быть разбито на подмножества подстановок, одинаково воздействующих на оператор  $t_i$  (подструктуру  $Q_i$ ). Это значит, что наряду с подстановками  $h' = (h'_I, h'_J) \in H_{(i-1)}$  ( $h'' \in H_{\alpha i 0}$ ) в этих множествах имеются такие подстановки  $h'_3$  ( $h''_3$ ), что

$$t_i h'_J = t_i h'_{J\vartheta},$$

$$Q_i h'' = Q_i h''_3.$$

Очевидно, что это возможно при условии, что в множествах  $H_{(i-1)}$ ,  $H_{\alpha i 0}$ , имеются подмножества  $H_{(i-1)}^* = \{h^*\} = \{(h_I^*, h_J^*)\}$ ,  $H_{\alpha i 0}^{**} = \{h^{**}\}$  соответственно, элементы которых удовлетворяют следующим равенствам:

$$h^* = h'(h'_3)^{-1},$$

$$h^{**} = h''(h''_3)^{-1}.$$

Это свойство симметрий дает возможность сократить число проверочных подстановок в предикатах (1.28), (1.30): достаточно использовать по одной от каждого подмножества эквивалентных подстановок.

В ряде случаев применение в конструктивном процессе только операторов  $t_i$ , не эквивалентных между собой по симметриям синтезируемых объектов, позволяет совсем исключить проверку каноничности по предикатам (1.28), (1.30). Примеры читатель найдет в последующих главах.

Каждое из множеств  $H_{(i-1)}$ ,  $H_{\alpha i 0}$ , может содержать группы, по-разному воздействующие на  $F$ - и  $F_I$ -классы  $Q$ -структуры и их элементы, определяя свойства этих классов.

Выделим два типа  $F'$ -классов множества  $J$ . К первому отнесем каждый  $F'$ -класс, на элементах которого действует симметрическая группа автоморфизмов, входящая в состав группы  $H_{(i-1)}$ . Это значит, что элементы такого  $F'$ -класса можно менять местами произвольным образом, не внося изменений в  $(i-1)$ -подструктуру. Ко второму типу отнесем каждый  $F'$ -класс, на элементах которого действует циклическая подстановка  $h \in H_{(i-1)}$  и порожденная этой подстановкой циклическая группа. Элементы  $F'$ -класса второго типа под действием такой подстановки могут переходить друг в друга по цепочке, соответствующей циклу, не внося изменений в  $(i-1)$ -подструктуру. Множество  $I$   $Q$ -структуры содержит аналогичные типы  $F_I$ -классов.

В группе  $H_{(i-1)}$  может быть такая подстановка (не обязательно одна), действием которой меняется последовательность  $F'$ -классов  $i$ -подструктуры. Назовем такие  $F'$ -классы  $H$ -эквивалентными. Указанные подстановки также могут образовать симметрические или циклические подгруппы внутри группы  $H_{(i-1)}$ . Поэтому соответствующее множество  $H$ -эквивалентных  $F'$ -классов также относится к одному из двух типов. Определение типов не отличается от приведенного для элементов  $F$ -классов множества  $J$ .

Искомые признаки каноничности должны определить правила формирования операторов  $t_i$  и их последовательностей, обеспечивающих истинность предикатов (1.28), (1.30). Оператор  $t_i$  может быть однозначно представлен выборкой  $\mathbf{j}_i$  с заданными на ее элементах отношениями

эквивалентности  $F'$ ,  $F_i$  и порядка  $T'$ ,  $T_i$ . Поэтому искомые правила сформулируем применительно к выборке  $\mathbf{j}_i$ . Рассмотрим примеры построения признаков каноничности, определяющих правила формирования  $F$ -классов выборки  $\mathbf{j}_i$ , относящихся к  $F'$ -классам множества  $J$  первого типа, и правила выбора последовательностей выборок  $\mathbf{j}_i$  внутри каждого  $F_I$ -класса первого типа. Очевидно, что это могут быть только необходимые условия каноничности. Для сравнения между собой  $F$ -классов выборок  $\mathbf{j}_i$  снова обратимся к характеристикам их элементов в виде пар  $(j, \delta_i)$ .

Пусть под действием некоторого автоморфизма  $h \in H_{(i-1)}$  меняются местами  $H$ -эквивалентные  $F'$ -классы множества  $J$ . При этом поменяются местами соответствующие  $F$ -классы выборки  $\mathbf{j}_i$ . Такие  $F$ -классы также уместно назвать  $H$ -эквивалентными. Сопоставим  $F$ -класс  $\mathbf{j}'_{i\beta}$  с тем  $F$ -классом  $\mathbf{j}''_{i\beta}$ , который окажется на его месте в результате преобразования подструктуры  $Q_i$  под действием  $h \in H_{(i-1)}$ ,  $\mathbf{j}''_{i\beta} = \mathbf{j}'_{i\beta} h$ . Очевидно, что в последовательностях из упорядоченных векторов  $(j, \delta_i)$ , характеризующих элементы  $F$ -классов  $\mathbf{j}'_{i\beta}$ ,  $\mathbf{j}''_{i\beta}$ , первые координаты всегда попарно совпадают. Поэтому для сравнения  $H$ -эквивалентных  $F$ -классов выборок  $\mathbf{j}_i$  достаточно попарного сопоставления только координат  $\delta_i$ . Обозначим последовательность из координат  $\delta_i$  элементов  $F$ -класса выборки  $\mathbf{j}_i$  символом  $\mathbf{d}_{i\beta}$ . Тогда

$$\mathbf{j}'_{i\beta} \mathbf{f} \mathbf{j}''_{i\beta} \Leftrightarrow \mathbf{d}'_{i\beta} \overset{lex}{\succ} \mathbf{d}''_{i\beta}. \quad (1.31)$$

**У т в е р ж д е н и е 1.6.** Для того чтобы отображение  $t_i, i \in \hat{I}$  было каноническим, необходимо, чтобы в его выборке  $\mathbf{j}_i$

1) каждый  $F$ -класс  $\mathbf{j}_{i\beta}$  состоял из лексикографически старших в алфавите  $(1, 2, \dots, |J|)$  элементов соответствующего  $F'$ -класса множества  $J$ , а элементы каждого  $F$ -класса  $\mathbf{j}_{i\beta}$  были упорядочены по старшинству;

2) векторы  $\mathbf{d}_{i\beta}$   $H$ -эквивалентных  $F$ -классов  $\mathbf{j}_{i\beta}$  были упорядочены лексикографически в алфавите  $(1, 2, \dots, D_i)$ .



**Доказательство.** Доказываем от противного. Допустим, что отображение  $t_i$  является каноническим, однако при этом нарушено условие 1. Значит, некоторый  $F$ -класс  $\mathbf{j}_{i\beta}$  выборки  $\mathbf{j}_i$  оператора  $t_i$  содержит не только старшие в алфавите  $(1, 2, \dots, |J|)$  элементы соответствующего  $F'$ -класса множества  $J$  или (и) элементы  $\mathbf{j}_{i\beta}$  не упорядочены по старшинству. Но принадлежность  $F'$ -класса множества  $J$  к первому типу означает, что среди автоморфизмов  $H_{(i-1)}$  имеется такой, что, не изменив  $(i-1)$ -подструктуру и порядок следования элементов множества  $J$ , предшествующих  $F'$ -классу, изменит состав элементов  $\mathbf{j}_{i\beta}$  так, что туда войдут только старшие элементы  $F'$ -класса, и упорядочит эти элементы по старшинству. После такого преобразования выборка  $\mathbf{j}_i$  станет старше исходной в соответствии с определением порядка на множестве выборок. Следовательно, исходная выборка  $\mathbf{j}_i$ , как и соответствующий оператор  $t_i$ , не были старшими. Тогда исходный оператор  $t_i$  не был каноническим, что противоречит исходному допущению.

Предположим теперь, что отображение  $t_i$  является каноническим, но при этом не выполнено условие 2. Тогда найдется автоморфизм  $h \in H_{(i-1)}$ , который, не изменив  $(i-1)$ -подструктуру и последовательность элементов  $j \in J$ , предшествующих первому из  $H$ -эквивалентных  $F$ -классов, поменяет эти  $F$ -классы местами так, что векторы  $\mathbf{d}_{i\beta}$   $h$ -эквивалентных  $F$ -классов выборки  $\mathbf{j}_i$  окажутся упорядоченными лексикографически в алфавите  $(1, 2, \dots, D_i)$ . Из определения порядка на множестве операторов  $t_i$ , их выборок  $\mathbf{j}_i$  и  $F$ -классов выборки  $\mathbf{j}_i$  (1.31) следует, что после этого выборка  $\mathbf{j}_i$  и построенный по ней оператор  $t_i$  станут старше. Следовательно, исходное отображение  $t_i$  не было каноническим, что противоречит первоначальному предположению.

**Утверждение 1.7.** *Для того чтобы  $i$ -подструктура была канонической, необходимо, чтобы внутри каждого  $F_1$ -класса  $I_\alpha$ ,  $\alpha \in \hat{I} \{1, 2, \dots, A\}$ , выборки  $\mathbf{j}_i$  операторов  $t_i$ ,  $i \in I_\alpha$ , были упорядочены по старшинству.*

**Доказательство.** Доказываем, как и утверждение 1.6, от противного. Пусть  $i$ -подструктура  $Q_i$  является канонической, но при этом векторы  $\mathbf{j}_i', \mathbf{j}_i'', i', i'' \in I_\alpha$ , характеризующие подключение ее  $i'$ - и  $i''$ -го

компонентов, не упорядочены по старшинству, т.е. при  $i' < i''$   $\mathbf{j}_{i'} \mathbf{p} \mathbf{j}_{i''}$ , а значит, и  $t_{i'} \mathbf{p} t_{i''}$ . Но внутри одного  $F_i$ -класса  $i'$ - и  $i''$ -й компоненты являются взаимозаменяемыми, и мы имеем право их поменять местами, т.е. поменять номера  $i'$  и  $i''$ , ничего не изменив в подключении компонентов с номерами  $i < i'$ . Формально это осуществляется с помощью одного из автоморфизмов  $h = (h_I, h_J) \in H_{\alpha i 0}$ . В результате получим новую  $i$ -подструктуру  $Q_i'$  с такой же последовательностью выборок  $\mathbf{j}_i$  при  $i < i'$ , как и у подструктуры  $Q_i$ , но с выборкой  $\mathbf{j}_i' = \mathbf{j}_{i''} h_J$ , старшей, чем выборка  $\mathbf{j}_i$  подструктуры  $Q_i$ . Из определений старшинства операторов  $t_i$  и  $i$ -подструктур (1.29) следует, что  $Q_i' \mathbf{f} Q_i$ . Тогда подструктура  $Q_i$  не была старшей, а значит, и канонической. Это противоречит исходному допущению.

Утверждения 1.6, 1.7 указывают признаки, по которым отсекается подавляющее большинство изоморфных  $i$ -подструктур на каждом шаге конструктивного процесса без проверки предикатов (1.27), (1.28), (1.30). Это существенно повышает эффективность процесса генерирования неизоморфных  $Q$ -структур благодаря исключению основной части многократно повторяемых переборных процедур, какими являются эти проверки.

#### 1.5.4. Применение матричного представления $Q$ -структуры

Предикаты каноничности, рассмотренные в п. 1.5.2, как и признаки их истинности из п. 1.5.3, наиболее наглядно и просто интерпретируются, если представить  $Q$ -структуру из множества  $M'$  взвешенной матрицей инцидентностей. Рассмотрим эту интерпретацию.

Пусть  $\mathbf{X}_i$  ( $\mathbf{X}_{\alpha i}$ ) – подматрица (блок) матрицы  $\mathbf{X}$ , представляющая  $i$ -подструктуру  $Q_i$  ( $Q_{\alpha i}$ ),  $i \in I$ , а  $\mathbf{x}_i$  – вектор из элементов  $i$ -й строки этой матрицы.

Можно считать, что вектор-строка  $\mathbf{x}_i$  представляет в матрице  $\mathbf{X}_i$  оператор  $t_i$  и его выборку  $\mathbf{j}_i$ : элементы  $j \in \mathbf{j}_i$  отражены в нем порядковыми номерами ненулевых элементов вектора  $\mathbf{x}_i$ , а принадлежность элемента  $j \in \mathbf{j}_i$   $\delta_i$ -му  $F_i$ -классу – равенством  $\delta_i$  соответствующего ненулевого элемента. Нетрудно убедиться, что введенный порядок на множестве оператор-

операторов  $t_i$  и их выборки  $\mathbf{j}_i$ ,  $i \in I$ , интерпретируется лексикографическим порядком на множестве соответствующих векторов-строк  $\mathbf{x}_i$  матрицы  $\mathbf{X}$  в алфавите  $(1, 2, \dots, D_i, 0)$ :

$$\mathbf{x}'_i \succ^{lex} \mathbf{x}''_i \Leftrightarrow \exists j^* \in J: \forall j \langle j^* \quad \mathbf{x}'_{ij} = \mathbf{x}''_{ij}, \quad \mathbf{x}'_{ij^*} \succ^{lex} \mathbf{x}''_{ij^*}.$$

Условимся под  $\mathbf{x}_i h_J$  подразумевать перестановку элементов вектора  $\mathbf{x}_i$  в соответствии с подстановкой  $h_J$  множества порядковых номеров его элементов.

**Пример 1.8.** При  $\mathbf{x}_i = (1, 1, 0, 0, 2)$  и  $h_J = (2, 3)(4, 5)$

$$\mathbf{x}_i h_J = (1, 1, 0, 0, 2) \begin{pmatrix} 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 2 & 5 & 4 \end{pmatrix} = (1, 0, 1, 2, 0).$$

Назовем канонической вектор-строку  $\mathbf{x}_i$ , если его невозможно привести к лексикографически старшему в алфавите  $(1, 2, \dots, D_i, 0)$  никакими подстановками  $h_J$ , которые являются составляющими автоморфизмов  $h \in H_{(i-1)}$ . Ясно, что канонический вектор-строка  $\mathbf{x}_i$  представляет канонический оператор  $t_i$ .

Порядком на множестве векторов-строк  $\mathbf{x}_i$  индуцируется упорядочение матриц  $\mathbf{X}_i (\mathbf{X}_{\alpha i}, i \in I_\alpha, \alpha \in \{1, 2, \mathbf{K}, A\})$ :

$$\mathbf{X}'_i \mathbf{f} \mathbf{X}''_i \Leftrightarrow \exists i^* \in (1, 2, \mathbf{K}, i): \forall i \langle i^* \quad \mathbf{x}'_i = \mathbf{x}''_i, \quad \mathbf{x}'_{i^*} \succ^{lex} \mathbf{x}''_{i^*},$$

$$\mathbf{X}'_{\alpha i} \mathbf{f} \mathbf{X}''_{\alpha i} \Leftrightarrow \exists i^* \in I_\alpha: \forall i \langle i^* \quad \mathbf{x}'_i = \mathbf{x}''_i, \quad \mathbf{x}'_{i^*} \succ^{lex} \mathbf{x}''_{i^*}.$$

Матрица  $\mathbf{X}_i$ ,  $i \in I$ , как и  $i$ -подструктура  $Q_i$ , является канонической, если она не может стать старше под действием подстановок  $h = (h_I, h_J) \in H^{(i)}$  на множествах  $I, J$  номеров ее строк и столбцов соответственно. Очевидно, что каноническая матрица  $\mathbf{X}_i$  представляет каноническую  $i$ -подструктуру  $Q_i$ .

Из сказанного следует, что предикат (1.28) при матричном представлении  $Q$ -структуры интерпретируется предикатом

$$P(\mathbf{x}_i) \equiv (\forall h = (h_I, h_J) \in H_{(i-1)} \quad \mathbf{x}_i \geq^{lex} \mathbf{x}_i h_J), \quad (1.32)$$

принимающим истинное значение только для канонических векторов-строк  $\mathbf{x}_i$ , а предикат (1.30) – предикатом

$$P(\mathbf{X}_{\alpha i}) \equiv \\ \equiv (\forall h = (h_{I_\alpha}, h_{J_\alpha}) \in H_{\alpha i 0}: \mathbf{X}_{\alpha i} \mathbf{f} H_{I_\alpha}^{-1} \mathbf{X}_{\alpha i} H_{J_\alpha} \vee \mathbf{X}_{\alpha i} = H_{I_\alpha}^{-1} \mathbf{X}_{\alpha i} H_{J_\alpha}), \quad (1.33)$$

принимающим истинное значение только для канонических матриц  $\mathbf{X}_{\alpha i}$ . В (1.33)  $H_{I_\alpha}$  ( $H_{J_\alpha}$ ) – перестановочная единичная матрица, соответствующая подстановке  $h_{I_\alpha}$  ( $h_{J_\alpha}$ ), где  $h_{I_\alpha}$  – ограничение подстановки  $h_I$  на множество  $i \in I_\alpha$ .

Рассмотрим интерпретацию признаков каноничности из утверждений 1.6, 1.7 при представлении  $i$ -подструктуры  $Q_i$  матрицей  $\mathbf{X}_i$ . Вектор-строка  $\mathbf{x}_i$  разбиением множества  $J$  номеров столбцов подматрицы  $\mathbf{X}_{(i-1)}$  на классы эквивалентности также разбивается на  $F'$ -классы. Первое требование утверждения 1.6 интерпретируется лексикографической упорядоченностью в алфавите  $(1, 2, \dots, D_i, 0)$  элементов вектора  $\mathbf{x}_i$  внутри каждого  $F'$ -класса. Второе требование утверждения 1.6 интерпретируется лексикографической упорядоченностью в алфавите  $(1, 2, \dots, D_i, 0)$   $H$ -эквивалентных  $F$ -классов вектора  $\mathbf{x}_i$ . Требование утверждения 1.7 означает лексикографическую упорядоченность в том же алфавите векторов-строк  $\mathbf{x}_i$  внутри каждого  $F_I$ -класса. Из сказанного можно сделать вывод, что следующие два утверждения являются интерпретацией утверждений 1.6 и 1.7 при представлении  $i$ -подструктуры  $Q_i$  матрицей  $\mathbf{X}_i$ .

**Утверждение 1.8.** *Для того чтобы вектор  $\mathbf{x}_i, i \in I$ , был каноническим, необходимо, чтобы были упорядочены лексикографически в алфавите  $(1, 2, \dots, D_i, 0)$  элементы каждого его  $F'$ -класса, а также  $F$ -классы при условии, что последние  $H$ -эквивалентны.*

**Утверждение 1.9.** *Для того чтобы матрица  $\mathbf{X}_{\alpha i}, i \in I_\alpha, \alpha \in \{1, 2, \mathbf{K}, \mathbf{A}\}$ , была канонической, необходимо, чтобы были упорядочены лексикографически в алфавите  $(1, 2, \dots, D_i, 0)$  ее векторы-строки  $\mathbf{x}_i$ .*

Утверждения 1.8, 1.9 доказываются от противного по аналогии с утверждениями 1.6, 1.7, поэтому опускаем эти доказательства.

**С л е д с т в и е .** Пусть векторы-строки матрицы  $\mathbf{X}_{\alpha i}, i \in I_\alpha, \alpha \in \{1, 2, \mathbf{K}, \mathbf{A}\}$ , упорядочены лексикографически. Тогда в множестве  $H_{\alpha i 0}$  проверочных подстановок предиката (1.33) можно исключить подстановки с составляющей  $e_J$ .

Действительно, в силу упорядоченности векторов  $\mathbf{x}_i$  под действием таких подстановок матрица  $\mathbf{X}_{\alpha_i}$  не может стать старше.

Отметим, что некоторые частные случаи перечисленных признаков каноничности в приложении к графам и системам инцидентностей уже были применены [1].

### 1.5.5. Алгоритм конструктивного перечисления неизоморфных $Q$ -структур

Ниже приведен алгоритм порождения неизоморфных  $Q$ -структур с заданным набором структурных элементов. В нем применены рассмотренные в п. 1.5.2, 1.5.3 предикаты и признаки их истинности, а  $Q$ -структура и ее  $i$ -подструктуры  $Q_i$ ,  $i \in I$ , представлены взвешенными матрицами инцидентностей. При описании алгоритма подразумевается применение известных, например, из работы [3], алгоритмов генерирования сочетаний и перестановок в лексикографической последовательности.

Алгоритм 1.1

Шаг 1. Положить  $i = 1$ .

Шаг 2. На основании информации о симметриях подматрицы  $\mathbf{X}_{(i-1)}$  разбить множества  $I$  и  $J$  соответственно на  $F_I$ - и  $F'_I$ -классы, учитывая, что при  $i = 1$  симметрии 0-подструктуры отражены в группе  $H_{II}$ , а  $F'_I$ -классы совпадают с  $F_I$ -классами. Выполнить соответствующее разбиение на  $F_I$ - и  $F'_I$ -классы множества строк и столбцов матрицы  $\mathbf{X}_i$ .

Шаг 3. Выбрать очередную перестановку множества  $L_i$  с  $F_i$ -классами, упорядоченными лексикографически в алфавите  $(1, 2, \dots, |L_i|)$ . (При переходе от шага 2 к шагу 3 такой перестановкой становится лексикографически старшая в этом алфавите.) От этой перестановки перейти к мультимножеству, заменив каждый ее элемент порядковым номером  $\delta_i$   $F_i$ -класса  $L_{\delta_i}$ , которому этот элемент принадлежит. Если варианты перестановок  $L_i$  исчерпаны, идти к шагу 8.

Шаг 4. Построить очередной в порядке лексикографического убывания вариант вектора  $\mathbf{x}_i$  с учетом требований утверждений 1.8, 1.9. Для этого в качестве ненулевых элементов  $x_{ij}$  вектора  $\mathbf{x}_i$  использовать элементы полученного на шаге 3 мультимножества, не меняя их последовательности. (При переходе от шага 3 к шагу 4 очередным становится лексикографически старший в алфавите  $(1, 2, \dots, D_i, 0)$ . Если

параллельное включение идентичных компонентов не представляет практической ценности, вариант  $\mathbf{x}_i = \mathbf{x}_{(i-1)}$  опустить. Если при выбранной перестановке  $L_i$  варианты  $\mathbf{x}_i$  исчерпаны, идти к шагу 3.

Шаг 5. Проверить полученный вектор  $\mathbf{x}_i$  по предикату (1.32). Если предикат принимает ложное значение, перейти к шагу 4.

Шаг 6. Проверить матрицу  $\mathbf{X}_{\alpha i}$  по предикату (1.33). Если предикат принимает ложное значение, перейти к шагу 4.

Шаг 7. Зафиксировать каноническую матрицу  $\mathbf{X}_i$ .

Шаг 8. Если  $i < |I|$ , положить  $i := i + 1$  и идти к шагу 2. Если  $i = |I|$ , найти максимальное значение  $i$ , при котором не исчерпаны варианты вектора  $\mathbf{x}_i$  или перестановок множества  $L_i$ , и с этим значением  $i$  идти к шагу 4. Если таких нет, конец.

На шаге 3 вместо генератора перестановок целесообразно использовать генератор мультимножеств.

Предлагаемую процедуру синтеза неизоморфных  $Q$ -структур можно рассматривать как управляемый вычислительный процесс, где в качестве управляющего фактора на каждом шаге построения  $Q$ -структуры используется информация о симметриях ее подструктур. Условимся для краткости называть эту процедуру  $Q$ -процессом.

**Пример 1.9.** Вернемся к примеру 1.7 и рассмотрим работу алгоритма 1.1 при  $i = |I| = 5$ . Пусть матрица  $\mathbf{X}_4$  4-подструктуры, определенной к этому моменту (см. пример 1.7), имеет вид

$$\mathbf{X}_4 = \begin{vmatrix} 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 0 \end{vmatrix}.$$

Шаг 8.  $i = 5$ .

Шаг 2. Разбиваем множество строк и столбцов матрицы  $\mathbf{X}_5$  с незаполненной пятой строкой на классы эквивалентности с учетом симметрий  $\mathbf{X}_4$  (см. пример 1.7):

$$\mathbf{X}_5 = \begin{vmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ \hline & & & & & \end{vmatrix}.$$

Шаг 3.  $L_5 = (1, 2, 3) \rightarrow (1, 1, 2)$ .

Шаг 4.  $\mathbf{x}_5 = (0, 0, 0, 1, 1, 2)$ .

Шаг 5.  $(0, 0, 0, 1, 1, 2) \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 5 & 4 & 6 \end{pmatrix} = (0, 0, 0, 1, 1, 2) = \mathbf{x}_5$ .

Вектор  $\mathbf{x}_5$  – канонический.

Шаг 6.  $I_\alpha = I_2 = \{5, 6\}$ ,  $J_\alpha = \{1, 2, 3, 4, 5, 6\}$ ,

$$H_{I_2}^{-1} \mathbf{X}_{2i} H_{J_2} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 2 \\ 0 & 0 & 2 & 1 & 1 & 0 \end{pmatrix} \mathbf{p} \mathbf{X}_{\alpha i} = \mathbf{X}_{2i}.$$

Матрица  $\mathbf{X}_{2i}$  – каноническая.

Шаг 7. Фиксируем полученную каноническую матрицу  $\mathbf{X}_5$

$$\mathbf{X}_5 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 2 \end{pmatrix}.$$

Шаг 8.  $i = 5$ .

Шаг 4. Варианты допустимых векторов  $\mathbf{x}_5$  исчерпаны.

Шаг 3.  $L_5 = (1, 3, 2) \rightarrow (1, 2, 1)$ .

Шаг 4.  $\mathbf{x}_5 = (0, 0, 0, 1, 2, 1)$ .

Шаг 5.  $(0, 0, 0, 1, 2, 1) \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 5 & 4 & 6 \end{pmatrix} = (0, 0, 0, 2, 1, 1) \stackrel{lex}{\prec} \mathbf{x}_5$ .

Вектор  $\mathbf{x}_5$  – канонический.

Шаг 6.  $I_\alpha = I_2 = \{5, 6\}$ ,  $J_\alpha = \{1, 2, 3, 4, 5, 6\}$ ,

$$H_{I_2}^{-1} \mathbf{X}_{2i} H_{J_2} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 2 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 2 & 1 & 1 \\ 0 & 0 & 2 & 1 & 1 & 0 \end{pmatrix} \mathbf{p} \mathbf{X}_{\alpha i} = \mathbf{X}_{2i}.$$

Матрица  $\mathbf{X}_{2i}$  – каноническая.

Шаг 8.  $i = 5$ .

Шаг 4. Варианты допустимых векторов  $\mathbf{x}_5$  исчерпаны.

Шаг 3.  $L_5 = (3, 1, 2) \rightarrow (2, 1, 1)$ .

Шаг 4.  $\mathbf{x}_5 = (0, 0, 2, 1, 0, 1)$ . (Вариант  $\mathbf{x}_5 = (0, 0, 2, 1, 1, 0)$  опускаем.)

Шаг 5.  $(0, 0, 2, 1, 0, 1) \cdot \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 5 & 4 & 6 \end{pmatrix} = (0, 0, 2, 0, 1, 1) \stackrel{lex}{\prec} \mathbf{x}_5$ .

Вектор  $\mathbf{x}_5$  – канонический.

Шаг 6.  $I_\alpha = I_2 = \{5, 6\}$ ,  $J_\alpha = \{1, 2, 3, 4, 5, 6\}$ ,

$$H_{I_2}^{-1} \mathbf{X}_{2i} H_{J_2} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 2 & 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 2 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 1 & 0 \end{pmatrix} \mathbf{P} \mathbf{X}_{ai} = \mathbf{X}_{2i}.$$

Матрица  $\mathbf{X}_{2i}$  – каноническая.

Шаг 7. Фиксируем полученную каноническую матрицу  $\mathbf{X}_5$  и

$$\mathbf{X}_5 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 2 & 1 & 1 & 0 \\ 0 & 0 & 2 & 1 & 0 & 1 \end{pmatrix}$$

Шаг 8.  $i = 5$ .

Шаг 4. Варианты допустимых векторов  $\mathbf{x}_5$  исчерпаны.

Шаг 3. Варианты допустимых перестановок  $L_5$  исчерпаны.

Шаг 8.  $i = 4$ .

Продолжая действовать по алгоритму 1.1, читатель найдет еще три неизоморфных варианта Q-структуры: с  $\mathbf{x}_5 = (0, 0, 0, 1, 1, 2)$ ,  $\mathbf{x}_5 = (0, 0, 0, 1, 2, 1)$  и  $\mathbf{x}_5 = (0, 0, 0, 2, 1, 1)$  при  $\mathbf{x}_4 = (0, 0, 2, 1, 0, 1)$ .

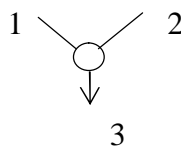
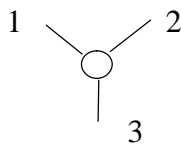
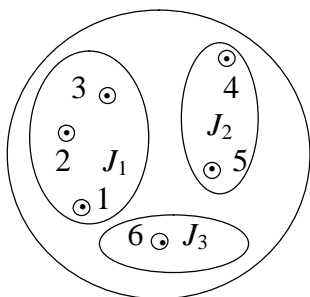


На рис. 1.3 приведена графическая интерпретация структурных элементов, использованных в примерах 1.7, 1.9 (рис. 1.3, а), а также интерпретации исходной 4-подструктуры (рис. 1.3, б) и найденных 5-подструктур  $Q_5'$ ,  $Q_5''$ ,  $Q_5'''$  (рис. 1.3, в, г, д). Элементы множеств  $J_1, J_2, J_3$  отображены в виде точек на плоскости, выделенных кружками. Элементы  $i \in I$  представлены кружками. Элементы множества  $L_i, i = 1, 2, \dots, 5$ , показаны как внешние связи элементов  $i \in I$ , помеченные стрелкой при  $\delta_i = 2$  и не помеченные при  $\delta_i = 1$ .

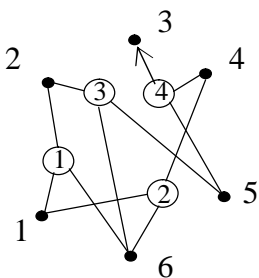
Назначение алгоритма 1.1 – показать перспективность применения информации о симметриях при решении задач структурного синтеза неизоморфных систем. Но основная работа по повышению его эффективности еще предстоит. Напомним, что в нем для сокращения перебора использованы только свойства одного типа  $F'$ - и  $F_I$ -классов. Пример 1.9 иллюстрирует полученный в результате эффект: вместо 60 вариантов подключения пятого компонента к 4-подструктуре (см. оценку в примере 1.7) по алгоритму 1.1 просматривается только 3 варианта, остальные 57 вариантов отсекаются на четвертом шаге как не выполняющие требований утверждений 1.8, 1.9. Но даже свойства указанных  $F'$ - и  $F_I$ -классов пока использованы не до конца. В этом нетрудно убедиться на примере 1.9. Так, после проверки каноничности векторов  $\mathbf{x}_i, i \in I_\alpha$ , по предикату (1.32) можно исключить проверку каноничности матрицы  $\mathbf{X}_i$  по предикату (1.33) на подстановках  $h \in H_{\alpha i 0}$ , фиксирующих множество  $I_\alpha$ ; упорядоченность векторов-строк  $\mathbf{x}_i$  в соответствии с утверждением 1.9 позволяет сократить число проверочных подстановок по предикату (1.33) путем исключения подстановок-транспозиций из множества  $H_{\alpha i 0}$  и так далее. Изучение условий, при которых утверждения 1.8, 1.9 дают не только необходимые, но и достаточные условия каноничности, позволит исключить эти проверки в алгоритме 1.1. Ряду других средств повышения эффективности алгоритма 1.1 уделено внимание в последующих главах.

## **1.6. Выявление симметрий $Q$ -структуры и ее подструктур**

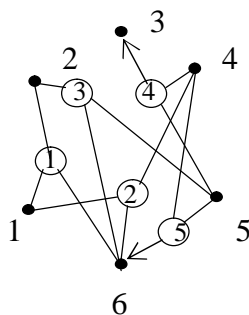
Роль информации о симметриях в решении задач отсекающего изоморфных структур в процессе их синтеза достаточно значительна, чтобы вызвать интерес к поиску эффективных средств выявления симметрий  $Q$ -структуры и ее подструктур.



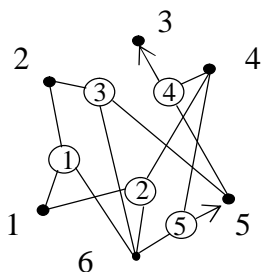
*a*



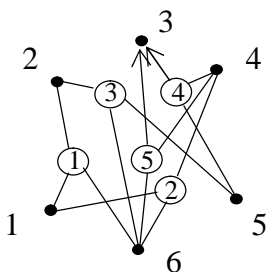
*б*



*в*



*г*



*д*

Рис. 1.3

При решении задачи выявления симметрий будем пользоваться для наглядности представлением  $Q$ -структуры в виде взвешенной матрицы инцидентностей  $X$ .

Очевидный алгоритм выявления симметрий основан на приведенном ранее определении симметрии, т.е. на поиске подстановок  $g = (g_I, g_J)$ , при воздействии которых на множества  $I$  и  $J$  соответственно строк и столбцов матрицы  $X$  последняя остается неизменной. Этот алгоритм включает в себя комбинаторную подзадачу, состоящую в конструктивном перечислении (генерировании) подстановок  $g = (g_I, g_J)$ . Пусть  $X'$  – обозначение преобразованной матрицы, т.е. со строками, переставленными в соответствии с подстановкой  $g_I$ , и столбцами, переставленными в соответствии с подстановками  $g_J$ . Те варианты преобразований  $g = (g_I, g_J)$ , для которых матрицы  $X$ ,  $X'$  совпадают, составляют группу автоморфизмов  $Q$ -структуры, отражающую ее симметрии.

Трудоемкость этого алгоритма в основном определяется трудоемкостью комбинаторной подзадачи. При больших значениях  $|I|$ ,  $|J|$  ее можно считать пропорциональной величине  $|I|!|J|!$ . Один из возможных подходов к уменьшению трудоемкости выявления симметрий  $Q$ -структуры состоит в сведении этой задачи к поиску симметрий частей, ее составляющих. Например, такой подход применен к определению группы автоморфизмов графа по группам автоморфизмов его подграфов [2]. В п. 1.3 он использован при определении группы автоморфизмов  $Q$ -структуры, которое сведено к поиску симметрий  $\alpha$ -подструктур,  $\alpha=1, 2, \dots, A$ . Недостаток такого подхода заключается в его нерегулярности: для разных способов соединения подструктур внутри структуры различны способы вычисления группы ее автоморфизмов по группам автоморфизмов, ее составляющих. Здесь будем рассматривать только регулярные методы, так как они удобны при автоматизации процесса поиска симметрий  $Q$ -структуры, которая неизбежна в связи с большой трудоемкостью этой задачи.

### 1.6.1. Применение разбиений

Основной известный принцип уменьшения трудоемкости указанного алгоритма состоит в сокращении его переборной части путем разбиения множеств, на которых ведется перебор, на подмножества на основании предварительного, сравнительно нетрудоемкого анализа каждого из этих множеств предпочтительно некомбинаторного характера. Здесь используется свойство факториала целого быть больше произведения факториалов его частей.

Для разбиения множеств  $I, J$  полезно использовать все имеющиеся структурные особенности матрицы  $\mathbf{X}$ . Так, при наличии одинаковых строк или (и) столбцов целесообразно выделить блок (блоки) из них. Поскольку матрица  $\mathbf{X}$  инвариантна к перестановке одинаковых строк (столбцов), подстановки на множестве их номеров образуют симметрическую группу, которая является подгруппой искомого группы автоморфизмов  $H$ . Остальные автоморфизмы матрицы  $\mathbf{X}$  – это автоморфизмы ее подматрицы, которая остается после удаления из матрицы  $\mathbf{X}$  блоков из одинаковых строк и столбцов. Понятно, что для выявления этих автоморфизмов потребуется уже меньшее число пробных подстановок  $g = (g_I, g_J)$ , так как они будут выполняться на меньших по мощности множествах строк или (и) столбцов. Если матрица  $\mathbf{X}$  является клеточно-диагональной, достаточно ограничиться просмотром подстановок  $g = (g_I, g_J)$ , не нарушающих такую структуру матрицы  $\mathbf{X}$ , и так далее.

Для сокращения перебора следует использовать и разбиения множеств  $I, J$  соответственно на  $F_I$ - и  $F_J$ -классы. Следствие из утверждения 1.1 говорит о том, что перебор может быть сокращен без риска потерять искомые симметрии, если просматривать только преобразования  $g = (g_I, g_J)$ , сохраняющие каждое из этих разбиений.

Обобщим на  $Q$ -структуры известный [15] способ дополнительного разбиения  $F_I$ - и  $F$ -классов на подклассы. Для этого введем понятие состава элементов строки (столбца) матрицы, который условимся характеризовать  $D$ -мерным вектором  $\mathbf{d}_i$  ( $\mathbf{d}_j$ ), в котором на  $\delta_i$ -м месте находится число элементов  $i$ -й строки ( $j$ -го столбца), равных  $\delta_i$ ,  $\delta_i \in \{1, 2, \dots, D_i\}$ . Очевидно, что  $F_I$ -класс – это множество одинаковых по составу строк, а  $F$ -класс – это множество одинаковых по составу столбцов. Множества строк и столбцов матрицы  $\mathbf{X}$ , не отличающихся по составу, разбивают матрицу  $\mathbf{X}$  на подматрицы (блоки). Ясно, что разбиение матриц  $\mathbf{X}$  и  $\mathbf{X}'$  на блоки совпадает, если  $g = (g_I, g_J)$  – автоморфизм.

**Утверждение 1.10.** *Для того чтобы перестановка строк и (или) столбцов не меняла матрицу  $\mathbf{X}$ , необходимо, чтобы их части, относящиеся к общему блоку, не отличались по составу элементов.*

Действительно, если хотя бы в одном блоке части переставляемых строк (столбцов) отличаются по составу элементов, то после перестановки этих строк (столбцов) в матрицу  $\mathbf{X}$  будет внесено изменение, которое не может быть компенсировано никакими перестановками столбцов (строк). Но тогда матрица  $\mathbf{X}$  изменится.

Утверждение 1.10 дает дополнительное необходимое условие, которому должны удовлетворять строки и столбцы матрицы  $\mathbf{X}$ , чтобы они могли

быть отнесены к одному классу эквивалентности, который в общем случае является подклассом  $F_I$ - или  $F_J$ -класса. Состав строки (столбца) матрицы  $\mathbf{X}$  определяется разбиением (1.6) множеств  $L_i$ ,  $i = 1, 2, \dots, |I|$ , на классы эквивалентности. Следовательно, разбиение (1.6) приводит к дополнительному разбиению  $F_I$ - и  $F_J$ -классов  $Q$ -структуры на подклассы.

Условимся называть множество строк (столбцов), удовлетворяющих условию утверждения 1.10,  $d_i$ -классами ( $d_j$ -классами). Из этого утверждения следует, что максимальное сокращение перебора проверочных преобразований  $g = (g_I, g_J)$  при поиске симметрий матрицы  $\mathbf{X}$  достигается, если не останавливаться на разбиении ее строк и столбцов на  $F_I$ - и  $F_J$ -классы, а продолжать разбиение этих классов до тех пор, пока не останется ни одного блока, в котором соответствующие ему части строк и столбцов не являются эквивалентными по составу элементов. В этом случае трудоемкость перебора проверяемых подстановок  $g = (g_I, g_J)$  при поиске симметрий матрицы  $\mathbf{X}$  сокращается до величины, пропорциональной  $\prod_n |I_n|! \prod_m |J_m|! \leq |I|! |J|!$ , где  $I_n$  –  $n$ -й  $d_i$ -класс,  $I = \sum_n I_n$ ;  $J_m$  –  $m$ -й  $d_j$ -класс,  $J = \sum_m J_m$ .

**Пример 1.10.** Пусть матрицей

$$\mathbf{X} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 2 & 0 & 0 & 0 & 2 \\ 0 & 0 & 1 & 0 & 2 & 2 \\ 0 & 0 & 0 & 1 & 2 & 2 \end{pmatrix}$$

задана  $Q$ -структура, у которой множество  $J$  образует один общий  $F_J$ -класс. Требуется выявить симметрии этой  $Q$ -структуры.

Состав элементов строк матрицы  $\mathbf{X}$  отражен в векторах  $\mathbf{d}_1 = (2, 0)$ ,  $\mathbf{d}_2 = (2, 0)$ ,  $\mathbf{d}_3 = (2, 0)$ ,  $\mathbf{d}_4 = (1, 2)$ ,  $\mathbf{d}_5 = (1, 2)$ ,  $\mathbf{d}_6 = (1, 2)$ , состав элементов столбцов – в векторах  $\mathbf{d}_1 = (4, 0)$ ,  $\mathbf{d}_2 = (0, 1)$ ,  $\mathbf{d}_3 = (2, 0)$ ,  $\mathbf{d}_4 = (2, 0)$ ,  $\mathbf{d}_5 = (0, 2)$ ,  $\mathbf{d}_6 = (1, 3)$ .

Покажем результат разбиения матрицы на соответствующие  $F_I$ - и  $F_J$ -классы

100001
101000
100100
120002
001022
000122

и, наконец, результат разбиения на  $d$ -классы ( $d_i$ -классы и  $d_j$ -классы)

1	0	0	0	0	1
1	0	1	0	0	0
1	0	0	1	0	0
1	2	0	0	0	2
0	0	1	0	2	2
0	0	0	1	2	2

Разбиение на  $d$ -классы показывает, что достаточно при поиске симметрий матрицы  $X$  проверить подстановки  $g = (g_I, g_J)$ , содержащие  $g_I = (2, 3)$ ,  $g_I = (5, 6)$ ,  $g_J = (3, 4)$ . В результате проверки находим, что группа автоморфизмов  $Q$ -структуры содержит подстановку  $h = ((2, 3)(5, 6), (3, 4))$  кроме тождественной подстановки  $(e_I, e_J)$ :

$$H = \{(e_I, e_J), ((2, 3)(5, 6), (3, 4))\}.$$

Эффект от разбиения множеств  $I$  и  $J$  на  $d$ -классы тем сильнее, чем меньше размеры  $d$ -классов. Эти размеры зависят от конкретной постановки задачи, в частности здесь большую роль играет степень однородности структуры  $Q$ : например, если все элементы каждого из множеств  $I$ ,  $J$  и  $L_i$ ,  $i = 1, 2, \dots, |I|$ , однотипны, разбиения может не произойти.

**Пример 1.11.** В приведенной ниже матрице ни одно из множеств  $I, J, L_i, i = 1, 2, \dots, |I|$ , не поддается разбиению, каждое из них – это один класс эквивалентности.

000111
001011
010101
101010
110100
111000

Остановимся на особенностях использования разбиений в процессе выявления симметрий  $i$ -подструктуры, которое в соответствии с алгоритмом 1.1 необходимо выполнять на каждом шаге  $Q$ -процесса при проверке каноничности  $\mathbf{x}_i$  и  $\mathbf{X}_i$  по предикатам (1.32), (1.33), и оценим выигрыш от этих разбиений.

Пусть множество  $I$  ( $J$ ) номеров строк (столбцов) матрицы  $\mathbf{X}$  разбито на подмножества  $I_1, I_2, \dots, I_\alpha, \dots, I_A$  ( $J_1, J_2, \dots, J_\beta, \dots, J_B$ ). И пусть  $i \in I_\alpha$ ,  $\alpha \in \{1, 2, \dots, A\}$ . В этом случае подматрица  $\mathbf{X}_i$ , как и матрица  $\mathbf{X}$ , разбита на блоки  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_\alpha$ . При этом блок  $\mathbf{X}_\alpha$  подматрицы  $\mathbf{X}_i$  состоит из  $\Delta_\alpha = i - \sum_{\alpha^*=1}^{\alpha-1} |I_{\alpha^*}| \leq |I_\alpha|$  строк. Тогда число проверочных преобразований  $g = (g_I, g_J)$ , необходимое для отбора неизоморфных  $i$ -подструктур методом решета (см. п. 1.5.1) и равное  $K_{\max} = i!|J|!$ , за счет применения разбиений сокращается до величины

$$K = \Delta_\alpha! \prod_{\alpha^*=1}^{\alpha-1} |I_{\alpha^*}|! \prod_{\beta=1}^B |J_\beta|! < K_{\max}. \quad (1.34)$$

Понятно, что оценка  $K$  дополнительно улучшается, если  $F_I$ - и  $F_J$ -классы матрицы  $\mathbf{X}$  удастся разбить соответственно на  $d_i$ - и  $d_j$ -классы. И наоборот, выигрыш от разбиений отсутствует, т.е.  $K = K_{\max}$ , если система абсолютно однородна.

Напомним, что для проверки каноничности вектора-строки  $\mathbf{x}_i$  по предикату (1.32) достаточно знать множество составляющих  $h_J$  подстановок  $h = (h_I, h_J) \in H_{(i-1)}$ . Из утверждения 1.3 следует, что это множество должно быть общим для всех  $\mathbf{X}_{\alpha^*}$ ,  $\alpha^* = 1, 2, \dots, \alpha$ , и оно имеет тенденцию к сокращению при переходе от  $\alpha'$  к  $\alpha'' = \alpha' + 1, \alpha' \in \{0, 1, 2, \dots, A-1\}$ . Пусть  $H_{J\alpha}$  – множество составляющих  $h_J$  матрицы  $\mathbf{X}_{(i-1)}$ , состоящей из блоков  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_\alpha$ , тогда

$$H_{J1} \supseteq H_{J2} \supseteq \dots \supseteq H_{J\alpha} \supseteq \dots \supseteq H_{JA}. \quad (1.35)$$

Из (1.35) следует, что элементы  $h_J \in H_{(i-1)}$ , необходимые для проверки каноничности вектора-строки  $\mathbf{x}_i$ ,  $i \in I_\alpha$ , по предикату (1.32),

могут находиться только среди элементов  $H_{J(\alpha-1)}$ . Множество  $H_{J(\alpha-1)}$  к моменту этой проверки определено: это множество составляющих  $h_J$  тех проверочных подстановок  $h = (h_I, h_J)$ , которым соответствует знак равенства в правой части предиката (1.32) при проверке вектора-строки  $\mathbf{x}_i$  с  $i = \sum_{\alpha^*=1}^{\alpha-1} |I_{\alpha^*}|$ . Значит, множество  $H_{(i-1)}$  определяется как подмножество множества подстановок  $h_J \in H_{J(\alpha-1)}$ , для каждой из которых найдется такая подстановка  $h_I$  первых  $(\Delta_\alpha - 1)$  элементов множества  $I_\alpha$ , что к подстановке  $h = (h_I, h_J)$  из этих составляющих блок  $\mathbf{X}_\alpha$  матрицы  $\mathbf{X}_{(i-1)}$  инвариантен. Таким образом, число проверочных подстановок, необходимых для определения множества  $H_{(i-1)}$ , сокращается до величины

$$K_{\mathbf{x}_i} = (\Delta_\alpha - 1)! |H_{J(\alpha-1)}|. \quad (1.36)$$

С помощью предиката (1.33) поиск неизоморфной  $i$ -подструктуры сведен к поиску канонической матрицы  $\mathbf{X}_i$ , т.е. старшей среди эквивалентных ей по симметриям  $i$ -подструктуры и ее частей. Можно утверждать, что подматрица  $\mathbf{X}_U$  матрицы  $\mathbf{X}$ , включающая в себя блоки  $\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_{\alpha-1}$  и, значит, строки с номерами  $i \in \bigcup_{\alpha^*=1}^{\alpha-1} I_{\alpha^*}$ , к моменту проверки матрицы  $\mathbf{X}_i$  на каноничность является канонической, так как в алгоритме 1.1 проверка каноничности предусмотрена для каждого  $i \leq \sum_{\alpha^*=1}^{\alpha-1} |I_{\alpha^*}|$ . Тогда матрица  $\mathbf{X}_i$  с  $i \in I_\alpha$  может стать старшей только за счет изменений в блоке  $\mathbf{X}_\alpha$ , не меняющих подматрицу  $\mathbf{X}_U$ . Такие изменения осуществляются только с помощью подстановок  $h = (h_I, h_J) \in H_{\alpha i 0}$ . Напомним, что по определению (см. п. 1.5.1, 1.5.2) в группе  $H_{\alpha i 0}$  исключены подстановки, оставляющие неподвижной  $i$ -ю строку. Исключив  $(\Delta_\alpha - 1)!$  таких подстановок, получим число  $K_{\mathbf{X}_i}$  элементов в группе  $H_{\alpha i 0}$ , применяемых в алгоритме 1.1 при проверке каноничности матрицы  $\mathbf{X}_i$  по предикату (1.33):

$$K_{\mathbf{X}_i} = [\Delta_\alpha! - (\Delta_\alpha - 1)!] |H_{J(\alpha-1)}| = (\Delta_\alpha - 1) (\Delta_\alpha - 1)! |H_{J(\alpha-1)}|. \quad (1.37)$$



Это число увеличивается до величины  $K_{\mathbf{X}_{i_{\max}}} = (i-1)(i-1)!|J| > K_{\mathbf{X}_i}$  в случае абсолютно однородной  $Q$ -структуры. Из выражений (1.36), (1.37) видно, что оценки  $K_{\mathbf{x}_i}$ ,  $K_{\mathbf{X}_i}$  дополнительно улучшаются, если множество из  $\Delta_\alpha$  строк подматрицы  $\mathbf{X}_\alpha$  удастся разбить на  $d_i$ -классы.

Таким образом, при использовании предикатов (1.32), (1.33) для установления каноничности вектора-строки  $\mathbf{x}_i$  и матрицы  $\mathbf{X}_i$  практически достаточно ограничиться проверочными подстановками-автоморфизмами, действующими на множествах строк и столбцов одного блока  $\mathbf{X}_\alpha$  матрицы  $\mathbf{X}_i$ , к которому относится  $i$ -я строка. Полученный при этом выигрыш по числу проверочных подстановок оценим, сопоставив величины  $K_{\mathbf{x}_i}$  (1.36) и  $K_{\mathbf{X}_i}$  (1.37) с величинами  $K_{\max} = (i-1)!|J|$  при проверке каноничности вектора  $\mathbf{x}_i$  и  $K_{\max} = i!|J|$  при проверке каноничности матрицы  $\mathbf{X}_i$ :

$$K_{\max} / K_{\mathbf{x}_i} = [(i-1)!|J|] / [(\Delta_\alpha - 1)! |H_{J(\alpha-1)}|]; \quad (1.38)$$

$$K_{\max} / K_{\mathbf{X}_i} = (i!|J|) / [(\Delta_\alpha - 1) (\Delta_\alpha - 1)! |H_{J(\alpha-1)}|]. \quad (1.39)$$

Ясно, что  $(\Delta_\alpha - 1) < i$ , а подстановки  $h_J \in H_{J(\alpha-1)}$  составляют только часть подстановок  $g_J$  на множестве элементов  $J$ . Поэтому выигрыш, отраженный в выражениях (1.38), (1.39), очень быстро ("комбинаторно") растет с увеличением числа строк  $i$  и столбцов  $|J|$  матрицы  $\mathbf{X}_i$ , так как множители в числителях этих выражений растут в факториальное число раз быстрее множителей их знаменателей.

Следует подчеркнуть, что при использовании алгоритма 1.1 определение элементов множества  $H_{(i-1)}$  ( $H_{J(\alpha-1)}$ ) не требует дополнительных временных затрат, так как его можно совместить с проверкой каноничности  $\mathbf{X}_{(i-1)}$  ( $\mathbf{X}_{(\alpha-1)}$ ) на одном из предшествующих шагов  $Q$ -процесса: это составляющие  $h_J$  тех проверочных подстановок  $h = (h_I, h_J)$ , которым соответствует знак равенства в правой части предиката (1.32) при проверке каноничности вектора  $\mathbf{x}_i$  (предиката (1.33) при проверке каноничности матрицы  $\mathbf{X}_{(\alpha-1)}$ ).

## 1.6.2. Алгоритм выявления симметрий $Q$ -структуры

Поскольку разбиения не полностью исключают перебор подстановок  $g = (g_I, g_J)$  при определении группы автоморфизмов  $Q$ -структуры и ее подструктур, остается актуальной задача рационального его выполнения.

Обобщим на  $Q$ -структуры предложенный в работе [16] подход к выявлению симметрий системы инцидентностей, основанный на представлении автоморфизма как отображения множества ненулевых элементов матрицы инцидентностей на себя. В случае  $Q$ -структуры к такому отображению предъявляется дополнительное требование: оно должно выполняться без нарушения разбиения множества  $X^*$  ненулевых элементов  $x_{ij}$  матрицы  $\mathbf{X}$  на подмножества  $X^{(1)}, X^{(2)}, \dots, X^{(\delta_i)}, \dots, X^{(D_i)}$  одинаковых между собой элементов:

$$X^* = \bigcup_{\delta_i=1}^{D_i} X^{(\delta_i)}. \quad (1.40)$$

Такой подход к выявлению симметрий  $Q$ -структуры правомочен только при условии, что ее матрица  $\mathbf{X}$  не содержит строк и (или) столбцов, заполненных только нулями. Если это условие не выполняется, целесообразно предлагаемый подход применить к подматрице заданной матрицы, которая остается после исключения из нее указанных строк и столбцов. При этом следует учесть, что симметрическая группа подстановок множества нулевых строк (столбцов), как уже говорилось в п.1.6.1, является подгруппой искомой группы симметрий всей матрицы  $\mathbf{X}$ .

На первый взгляд эффективность предлагаемого подхода представляется сомнительной, так как в его основе лежит перебор подстановок множества  $X^*$  всех ненулевых элементов матрицы  $\mathbf{X}$ , число которых, даже с учетом разбиения (1.40) их на подмножества  $X^{(\delta_i)}$ ,  $\delta_i = 1, 2, \dots, D_i$ , не всегда меньше числа  $|I|! |J|!$ . Однако ограничение на допустимые подстановки множества  $X^*$ , которое заключается в требовании реализации этих подстановок перемещениями строк и столбцов матрицы  $\mathbf{X}$ , существенно меняет картину. Условимся в дальнейшем для краткости называть такие подстановки и соответствующие перестановки множества  $X^*$  реализуемыми.

Остановимся на формализации предлагаемого подхода. Для этого пронумеруем ненулевые элементы  $x_{ij}$  матрицы  $\mathbf{X}$  и будем считать, что множество  $M$  составляют их порядковые номера  $m$ ,  $M = \{m\}$ . Пусть  $\mathbf{M} = (1, 2, \dots, \mu, \dots, |M|)$  – первоначальная последовательность этих

элементов в матрице  $\mathbf{X}$ . Перестановки  $\mathbf{M} = (m_1, m_2, \dots, m_\mu, \dots, m_{|M|})$  множества  $M$  будем просматривать в лексикографической последовательности, применив, например, известный [3] алгоритм их порождения. Такая организация просмотра вариантов перестановок  $\mathbf{M}$  обеспечивает его полноту.

Теперь займемся отсечением тех вариантов, которые соответствуют нереализуемым перестановкам множества  $X^*$ . С целью построения формальных признаков реализуемости введем ряд определений и обозначений.

Представим элемент  $x_{ij}$  упорядоченной парой его координат  $i$  и  $j$ , отражающих его расположение в матрице  $\mathbf{X}$ , т.е. вектором  $\mathbf{w} = (i, j)$ . Пусть  $W = \{\mathbf{w}\}$  – множество таких векторов. Тогда  $M = \{m\}$  можно рассматривать как множество порядковых номеров элементов  $\mathbf{w}$  в множестве  $W$ ,  $|M| = |W| = |X^*|$ . При этом условимся согласовывать нумерацию элементов множеств  $M$  и  $W$ , например, так:

$$m' \langle m'' \Leftrightarrow (i_{m'}, j_{m'}) \rangle^{lex} (i_{m''}, j_{m''}) . \quad (1.41)$$

Соотношением (1.41) между элементами множеств  $M$  и  $W$ , а значит, и  $X^*$  установлено взаимно однозначное соответствие, которое позволяет выбирать из трех указанных представлений ненулевых элементов матрицы  $\mathbf{X}$  наиболее удобное на каждом шаге решения поставленной задачи.

Для облегчения проверки реализуемости заменим элементы  $m$  порождаемых перестановок  $\mathbf{M}$  соответствующими элементами  $\mathbf{w} \in W$ . Пусть  $\mathbf{W}_s = (\mathbf{w}_1, \mathbf{w}_2, \dots, \mathbf{w}_\mu, \dots, \mathbf{w}_{|M|})$  – самая старшая перестановка множества  $W$ , соответствующая перестановке  $\mathbf{M} = (1, 2, \dots, \mu, \dots, |M|)$  множества  $M$ , а  $\mathbf{W} = (\mathbf{w}_{m_1}, \mathbf{w}_{m_2}, \dots, \mathbf{w}_{m_\mu}, \dots, \mathbf{w}_{m_{|M|}})$  – перестановка, соответствующая любой другой перестановке  $\mathbf{M} = (m_1, m_2, \dots, m_\mu, \dots, m_{|M|})$  множества  $M$ .

Назовем элементы  $\mathbf{w}_{m_\mu} = (i', j')$ ,  $\mathbf{w}_{m_\eta} = (i'', j'')$   $d$ -эквивалентными, если координаты  $i'$ ,  $i''$  принадлежат общему  $d_i$ -классу, а  $j'$ ,  $j''$  – общему  $d_j$ -классу, или  $\delta_i$ -эквивалентными, если равны между собой соответствующие им элементы  $x_{i'j'}$  и  $x_{i''j''}$  матрицы  $\mathbf{X}$ , т.е.

$$\mathbf{w}_{m_\mu} \overset{d}{\sim} \mathbf{w}_{m_\eta} \Leftrightarrow i' \overset{d_i}{\sim} i'' \wedge j' \overset{d_j}{\sim} j'' ,$$

$$\mathbf{w}_{m_\mu} \stackrel{\delta_i}{\sim} \mathbf{w}_{m_\eta} \Leftrightarrow x_{i'j'} = x_{i''j''}.$$

Поскольку разбиение множеств строк, столбцов и элементов матрицы  $\mathbf{X}$  должно сохраниться после преобразования ее под действием автоморфизма  $h \in H$ , очевидно следующее

**Утверждение 1.11.** *Для того чтобы подстановка элементов множества  $W$  представляла автоморфизм  $h \hat{I} H$ , необходимо, чтобы она осуществляла замену каждого элемент  $\mathbf{w}_{m_\mu}$  этого множества только  $d$ - и  $\delta_i$ -эквивалентным ему элементом  $\mathbf{w}_{m_\eta}$ ,  $m_\mu, m_\eta \hat{I} M$ .*

Утверждение 1.11 определяет одно из ограничений на просматриваемые перестановки множества  $W$ , следующее из разбиения матрицы  $\mathbf{X}$ . Другое ограничение связано с тем, что с перемещением строки (столбца) матрицы  $\mathbf{X}$  перемещаются все ее (его) ненулевые элементы одновременно. Рассмотрим это ограничение подробнее.

Элементы  $m', m'' \in M$ ,  $\mathbf{w}_{m'}, \mathbf{w}_{m''} \in W$ ,  $x_{i_m' j_m'}, x_{i_m'' j_m''} \in X^*$  назовем  $i$ -эквивалентными ( $j$ -эквивалентными), если  $i_{m'} = i_{m''}$  ( $j_{m'} = j_{m''}$ ), или независимыми, если  $i_{m'} \neq i_{m''}$ ,  $j_{m'} \neq j_{m''}$ .

Состояние матрицы  $\mathbf{X}$ , когда зафиксированы и не подлежат изменению номера строк и столбцов, на пересечении которых находятся первые  $\mu$  элементов множества  $X^*$ , назовем ее  $\mu$ -состоянием. Матрице  $\mathbf{X}$  в  $\mu$ -состоянии поставим в соответствие упорядоченное множество  $\mathbf{W}_\mu$ , в котором определены и зафиксированы обе координаты  $i, j$  первых  $\mu$  элементов, координата  $i$  ( $j$ ) каждого из элементов с номерами  $m_{(\mu+1)}, m_{(\mu+2)}, \dots, m_{|M|}$ ,  $i$ -эквивалентных ( $j$ -эквивалентных) хотя бы одному из первых  $\mu$  элементов, и заменены нулями координаты, допускающие изменение. Описанное множество  $\mathbf{W}_\mu$  по аналогии с  $\mu$ -состоянием матрицы  $\mathbf{X}$  логично назвать  $\mu$ -состоянием упорядоченного множества  $\mathbf{W}$ . Очевидно, что в 0-состоянии множество  $\mathbf{W}$  содержит только элементы  $(i, j) = (0, 0)$ , а в  $|M|$ -состоянии в нем отсутствуют элементы с нулевыми координатами. Последнему состоянию соответствует автоморфизм  $h \in H$ , который определяется поэлементным сопоставлением упорядоченных множеств  $\mathbf{W}_s$  и  $\mathbf{W}$ : если сравнить  $\mathbf{w}_\mu = (i, j) \in \mathbf{W}_s$  и  $\mathbf{w}_{m_\mu} = (i', j') \in \mathbf{W}$ ,  $\mu, m_\mu \in \{1, 2, \dots, |M|\}$ , видно, в какую координату  $i'(j')$  переходит координата  $i(j)$ . После исключения повторяющихся результатов получим

$$h = \left( \left( \begin{matrix} i_1 & i_2 & \dots & i_{|J|} \\ i'_1 & i'_2 & \dots & i'_{|J|} \end{matrix} \right), \left( \begin{matrix} j_1 & j_2 & \dots & j_{|J|} \\ j'_1 & j'_2 & \dots & j'_{|J|} \end{matrix} \right) \right).$$

Из определения  $\mu$ -состояния  $\mathbf{W}$  следует, что оно может быть установлено по его  $(\mu - 1)$ -состоянию, т.е. рекурсивно, с помощью следующего алгоритма.

Алгоритм 1.2.

Шаг 1. Сравнить между собой элементы  $\mathbf{w}_{m_\mu} \in \mathbf{W}$  и  $\mathbf{w}_\mu \in \mathbf{W}_s$ . Если  $j_{m_\mu} = j_\mu$ , идти к шагу 3.

Шаг 2. Найти в подмножестве  $\{ \mathbf{w}_{(\mu+1)}, \mathbf{w}_{(\mu+2)}, \dots, \mathbf{w}_{|M|} \}$  множества  $\mathbf{W}_s$  элементы, которые  $j$ -эквивалентны  $\mathbf{w}_\mu$ . В соответствующих, т.е. с теми же порядковыми номерами  $\mu$ , элементах множества  $\mathbf{W}$  в  $(\mu - 1)$ -состоянии заменить вторую нулевую координату координатой  $j_{m_\mu}$ . Если  $i_{m_\mu} = i_\mu$ , идти к шагу 4.

Шаг 3. Найти в множестве  $\{ \mathbf{w}_{(\mu+1)}, \mathbf{w}_{(\mu+2)}, \dots, \mathbf{w}_{|M|} \} \in \mathbf{W}_s$  элементы, которые  $i$ -эквивалентны  $\mathbf{w}_\mu$ . В соответствующих элементах множества  $\mathbf{W}$  в  $(\mu - 1)$ -состоянии заменить первую нулевую координату координатой  $i_{m_\mu}$ .

Шаг 4. Конец.

Пусть  $W'_\mu (W''_\mu)$  – множество элементов  $\mathbf{W}(W_\mu)$ , за исключением первых  $\mu$ .

**У т в е р ж д е н и е 1.12.** *Для того чтобы элемент  $\mathbf{w}_{m_\mu} \hat{I} \mathbf{W}$  можно было заменить элементом  $\mathbf{w}_{m_\eta} \hat{I} W'_\mu$  без изменения  $(m-1)$ -состояния  $\mathbf{W}$ , необходимо, чтобы элемент  $\mathbf{w}_{m_\eta}$  был*

1) *независимым от элементов  $\mathbf{w}_{m_\nu} \hat{I} \mathbf{W}$ ,  $\nu = 1, 2, \dots, (m-1)$ , если  $\mathbf{w}_{m_\mu}$  не зависит от них;*

2)  *$i$ -эквивалентным ( $j$ -эквивалентным) элементу  $\mathbf{w}_{m_\mu}$ , если последний  $i$ -эквивалентен ( $j$ -эквивалентен) хотя бы одному из элементов  $\mathbf{w}_{m_\nu} \hat{I} \mathbf{W}$ ,  $\nu = 1, 2, \dots, (m-1)$ .*

**Д о к а з а т е л ь с т в о.** Из определения  $(\mu - 1)$ -состояния матрицы  $\mathbf{X}$  следует, что оно не нарушается, если меняются между собой местами ее строки и (или) столбцы, не относящиеся к зафиксированным. Из независимости  $\mathbf{w}_{m_\mu}$  от элементов с номерами  $m_1, m_2, \dots, m_{(\mu-1)}$  следует,

что соответствующий элемент  $x_{i'j'} \in X^*$  находится как раз на пересечении таких строки и столбца. Следовательно, если заменить их также незафиксированными соответственно  $i''$ -й строкой и  $j''$ -м столбцом,  $(\mu - 1)$ -состояние матрицы  $\mathbf{X}$  сохранится. Мы рассматриваем только такие перемещения строк и столбцов матрицы  $\mathbf{X}$ , которые переставляют между собой ненулевые элементы. Поэтому  $i''$ ,  $j''$  – одновременно координаты такого элемента,  $x_{i''j''} \in X^*$ . Эти координаты не совпадают ни с одной из соответствующих координат элементов  $x_{ij} \in X^*$  с номерами  $m_1, m_2, \dots, m_{(\mu-1)}$ , так как номера зафиксированных и незафиксированных строк (столбцов) относятся к двум различным и непересекающимся множествам. Тогда по определению элемент  $x_{i''j''} \in X^*$ , как и соответствующий ему элемент  $\mathbf{w}_{m_\eta} \hat{\mathbf{I}} W'_\mu$ , является независимым от элементов  $\mathbf{w}_{m_\nu}$ ,  $\nu = 1, 2, \dots, \mu - 1$ .

Пусть теперь элемент  $\mathbf{w}_{m_\mu} i(j)$ -эквивалентен хотя бы одному из элементов с номерами  $m_1, m_2, \dots, m_{(\mu-1)}$ . Тогда представляемый им элемент  $x_{i'j'} \in X^*$  находится в строке (столбце) с зафиксированным номером  $i'(j')$  и незафиксированным номером  $j'(i')$  столбца (строки). Из определения  $(\mu - 1)$ -состояния матрицы  $\mathbf{X}$  следует, что оно не будет нарушено, если столбец (строку) с незафиксированным номером  $j'(i')$  заменить столбцом (строкой) с незафиксированным номером  $j''(i'')$ . При такой перестановке место элемента  $x_{i'j'} \in X^*$  займет элемент  $x_{i''j''} \in X^*$  ( $x_{i''j''} \in X^*$ ). Так как первые (вторые) координаты замещающего и замещаемого элементов совпадают, они  $i(j)$ -эквивалентны. Тогда  $i(j)$ -эквивалентны и представляющие их элементы  $\mathbf{w}_{m_\mu}, \mathbf{w}_{m_\eta} \hat{\mathbf{I}} W'_\mu$ .

Утверждение 1.12 позволяет сократить количество просматриваемых вариантов  $\mathbf{W}_\mu$  без риска потерять искомые перестановки  $W$ , но не гарантирует, что каждый из просматриваемых вариантов приведет к перестановке, представляющей автоморфизм  $h \in H$ . Каждый раз после перехода от  $\mathbf{W}_{(\mu-1)}$  к  $\mathbf{W}_\mu$ ,  $\mu \in \{1, 2, \dots, |M|\}$ , среди элементов множества  $W''_\mu$  может оказаться такой (не обязательно один), у которого обе координаты  $i$  и  $j$  определились автоматически в силу его  $i$ -эквивалентности одним элементам

$w_{m_\nu}$ ,  $\nu = 1, 2, \dots, \mu - 1$ , и  $j$ -эквивалентности другим. Из определения автоморфизма следует требование к такому элементу, чтобы он принадлежал множеству  $W'_\mu$ . Если это требование удовлетворяется для каждого автоматически определившегося элемента множества  $W''_\mu$ , условимся называть соответствующее  $\mu$ -состояние корректным. Некорректность  $W_\mu$  является признаком того, что выбранная последовательность первых  $\mu$  элементов множества  $W$  не приведет к искомому автоморфизму.

Построенный по рекурсивной схеме алгоритм 1.3 поиска перестановок  $W$ , определяющих автоморфизмы матрицы  $X$ , отражает правила перехода от одной перестановки  $W$  к последующей в порядке лексикографического убывания соответствующих перестановок  $M$  в алфавите  $(1, 2, \dots, |M|)$ . При этом построение первой самой старшей перестановки  $W_s$  начинается с задания  $\mu = 1$ ,  $(\mu - 1)$ -состояния  $W_0$  и списка допустимых вариантов элемента  $w_{m_1} = w_1$ , куда входит он сам вместе с множеством  $d$ - и  $\delta_i$ -эквивалентных ему элементов множества  $W_s$ . В алгоритме используются признаки независимости,  $i$ -эквивалентности или  $j$ -эквивалентности элемента  $w_{m_\mu}$ ,  $\mu \in M$ , подлежащего замене, которые состоят в том, что на месте последнего в  $W_{(\mu-1)}$  находятся соответственно элементы  $(0,0)$ ,  $(i,0)$  или  $(0, j)$ .

Алгоритм 1.3.

Шаг 1. Элементы множества  $W$  с номерами  $m_1, m_2, \dots, m_{(\mu-1)}$  оставить без изменения, а элемент  $w_{m_\mu}$  заменить лексикографически старшим из списка допустимых вариантов его замены, исключив последний из списка. Определить  $\mu$ -состояние по алгоритму 1.2 и проверить его корректность. Если  $\mu$ -состояние некорректно, идти к шагу 3.

Шаг 2. В множестве  $W''_\mu$  найти элемент с минимальным номером  $\mu$ , у которого хотя бы одна координата была равна нулю. Если есть такой, присвоить  $m_\mu$  порядковый номер этого элемента. В соответствии с утверждениями 1.11, 1.12 определить список допустимых вариантов замены  $w_{m_\mu}$  с учетом  $(\mu - 1)$ -состояния  $W$  и идти к шагу 1. Если в множестве  $W_\mu$  не осталось элементов с нулевыми координатами, зафиксировать соответствующий ему автоморфизм  $h \in H$ .

Шаг 3. Просматривая поочередно списки допустимых вариантов  $\mathbf{w}_{m_\eta}$  замены элементов  $\mathbf{W}_\mu$  с номерами  $m_\mu, m_{(\mu-1)}, \dots, m_2, m_1$ , найти первый, у которого этот список не исчерпан. Присвоить  $m_\mu$  порядковый номер этого элемента и идти к шагу 1.

Шаги 1, 2, 3 повторять до тех пор, пока при  $\mu = 1$  список допустимых вариантов замены этого элемента не окажется пустым.

Работу алгоритма проиллюстрируем примером.

**Пример 1.12.** Пусть требуется определить группу автоморфизмов матрицы

$$\mathbf{X} = \left\| \begin{array}{cc|cc|cccc} 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 2 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 2 & 2 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 2 & 2 \end{array} \right\|.$$

На матрице  $\mathbf{X}$  показано ее разбиение на  $d$ -классы.

Исходные данные для работы алгоритма:  $\mu = 1$ ,  $\mathbf{W}_{(\mu-1)} = \mathbf{W}_0 = ( (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0) )$ ; допустимые варианты  $\mathbf{w}_1 : (1, 1), (2, 2)$ .

Шаг 1. Выберем  $\mathbf{w}_1 : (1, 1)$ . Остается вариант  $\mathbf{w}_1 = (2, 2)$ .  $\mathbf{W}_\mu = \mathbf{W}_1 = ( (1, 1), (1, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 2$ , варианты  $\mathbf{w}_2 : (1, 3)$ .

Шаг 1. Выбираем  $\mathbf{w}_2 = (1, 3)$ . Варианты исчерпаны.  $\mathbf{W}_2 = ( (1, 1), (1, 3), (0, 0), (0, 0), (0, 3), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 3$ , варианты  $\mathbf{w}_2 : (2, 2)$ .

Шаг 1. Выбираем  $\mathbf{w}_3 = (2, 2)$ . Варианты исчерпаны.  $\mathbf{W}_3 = ( (1, 1), (1, 3), (2, 2), (2, 0), (0, 3), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 4$ , варианты  $\mathbf{w}_4 : (2, 4)$ .

Шаг 1. Выбираем  $\mathbf{w}_4 = (2, 4)$ . Варианты исчерпаны.  $\mathbf{W}_4 = ( (1, 1), (1, 3), (2, 2), (2, 4), (0, 3), (0, 0), (0, 0), (0, 4), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 5$ , варианты  $\mathbf{w}_5 : (3, 3)$ .

Шаг 1. Выбираем  $\mathbf{w}_5 = (3, 3)$ . Варианты исчерпаны.  $\mathbf{W}_5 = ( (1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 0), (3, 0), (0, 4), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 6$ , варианты  $\mathbf{w}_6 : (3, 5), (3, 6)$ .

Шаг 1. Выбираем  $\mathbf{w}_6 = (3, 5)$ . Остается вариант  $(3, 6)$ .  $\mathbf{W}_6 = ( (1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 5), (3, 0), (0, 4), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 7$ , варианты  $\mathbf{w}_7 : (3, 6)$ .



Шаг 1. Выбираем  $w_7 = (3, 6)$ . Варианты исчерпаны.  $W_7 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 5), (3, 6), (0, 4), (0, 0), (0, 0))$ .

Шаг 2.  $\mu = 8$ , варианты  $w_8 : (4, 4)$ .

Шаг 1. Выбираем  $w_8 = (4, 4)$ . Варианты исчерпаны.  $W_8 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 5), (3, 6), (4, 4), (4, 0), (4, 0))$ .

Шаг 2.  $\mu = 9$ , варианты  $w_9 : (4, 7), (4, 8)$ .

Шаг 1. Выбираем  $w_9 = (4, 7)$ . Остаются варианты:  $(4, 8)$ .  $W_9 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 5), (3, 6), (4, 4), (4, 7), (4, 0))$ .

Шаг 2.  $\mu = 10$ , варианты  $w_{10} : (4, 8)$ .

Шаг 1. Выбираем  $w_{10} = (4, 8)$ . Варианты исчерпаны.  $W_{10} = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 5), (3, 6), (4, 4), (4, 7), (4, 8))$ .

Шаг 2.  $h_1 = (e_i, e_j)$ .

Шаг 3.  $\mu = 9$ .

Шаг 1. Выбираем  $w_9 = (4, 8)$ . Варианты исчерпаны.  $W_9 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 5), (3, 6), (4, 4), (4, 8), (4, 0))$ .

Шаг 2.  $\mu = 10$ , варианты  $w_{10} : (4, 7)$ .

Шаг 1. Выбираем  $w_{10} = (4, 7)$ . Варианты исчерпаны.  $W_{10} = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 5), (3, 6), (4, 4), (4, 8), (4, 7))$ .

$$\text{Шаг 2. } h_2 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 5 & 6 & 8 & 7 \end{pmatrix} \right) \right).$$

Шаг 3.  $\mu = 6$ , варианты  $w_6 : (3, 6)$ .

Шаг 1. Выбираем  $w_6 = (3, 6)$ . Варианты исчерпаны.  $W_6 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 6), (3, 0), (0, 4), (0, 0), (0, 0))$ .

Шаг 2.  $\mu = 7$ , варианты  $w_7 : (3, 5)$ .

Шаг 1. Выбираем  $w_7 = (3, 5)$ . Варианты исчерпаны.  $W_7 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 6), (3, 5), (0, 4), (0, 0), (0, 0))$ .

Шаг 2.  $\mu = 8$ , варианты  $w_8 : (4, 4)$ .

Шаг 1. Выбираем  $w_8 = (4, 4)$ . Варианты исчерпаны.  $W_8 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 6), (3, 5), (4, 4), (4, 0), (4, 0))$ .

Шаг 2.  $\mu = 9$ , варианты  $w_9 : (4, 7), (4, 8)$ .

Шаг 1. Выбираем  $w_9 = (4, 7)$ . Остаются варианты:  $(4, 8)$ .  $W_9 = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 6), (3, 5), (4, 4), (4, 7), (4, 0))$ .

Шаг 2.  $\mu = 10$ , варианты  $w_{10} : (4, 8)$ .

Шаг 1. Выбираем  $w_{10} = (4, 8)$ . Варианты исчерпаны.  $W_{10} = ((1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 6), (3, 5), (4, 4), (4, 7), (4, 8))$ .

$$\text{Шаг 2. } h_3 = \left( \left( \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{array} \right), \left( \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 6 & 5 & 7 & 8 \end{array} \right) \right).$$

Шаг 3.  $\mu = 9$ , варианты  $\mathbf{w}_9 : (4, 8)$ .

Шаг 1. Выбираем  $\mathbf{w}_9 = (4, 8)$ . Варианты исчерпаны.  $\mathbf{W}_9 = ( (1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 6), (3, 5), (4, 4), (4, 8), (4, 0) )$ .

Шаг 2.  $\mu = 10$ , варианты  $\mathbf{w}_{10} : (4, 7)$ .

Шаг 1. Выбираем  $\mathbf{w}_{10} = (4, 7)$ . Варианты исчерпаны.  $\mathbf{W}_{10} = ( (1, 1), (1, 3), (2, 2), (2, 4), (3, 3), (3, 6), (3, 5), (4, 4), (4, 8), (4, 7) )$ .

$$\text{Шаг 2. } h_4 = \left( \left( \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 1 & 2 & 3 & 4 \end{array} \right), \left( \begin{array}{cccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 2 & 3 & 4 & 6 & 5 & 8 & 7 \end{array} \right) \right).$$

Шаг 3.  $\mu = 1$ , варианты  $\mathbf{w}_1 : (2, 2)$ .

Шаг 1. Выбираем  $\mathbf{w}_1 = (2, 2)$ . Варианты исчерпаны.  $\mathbf{W}_1 = ( (2, 2), (2, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 2$ , варианты  $\mathbf{w}_2 : (2, 4)$ .

Шаг 1. Выбираем  $\mathbf{w}_2 = (2, 4)$ . Варианты исчерпаны.  $\mathbf{W}_2 = ( (2, 2), (2, 4), (0, 0), (0, 0), (0, 4), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 3$ , варианты  $\mathbf{w}_3 : (1, 1)$ .

Шаг 1. Выбираем  $\mathbf{w}_3 = (1, 1)$ . Варианты исчерпаны.  $\mathbf{W}_3 = ( (2, 2), (2, 4), (1, 1), (1, 0), (0, 4), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 4$ , варианты  $\mathbf{w}_4 : (1, 3)$ .

Шаг 1. Выбираем  $\mathbf{w}_4 = (1, 3)$ . Варианты исчерпаны.  $\mathbf{W}_4 = ( (2, 2), (2, 4), (1, 1), (1, 3), (0, 4), (0, 0), (0, 0), (0, 3), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 5$ , варианты  $\mathbf{w}_5 : (4, 4)$ .

Шаг 1. Выбираем  $\mathbf{w}_5 = (4, 4)$ . Варианты исчерпаны.  $\mathbf{W}_5 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 0), (4, 0), (0, 3), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 6$ , варианты  $\mathbf{w}_6 : (4, 7), (4, 8)$ .

Шаг 1. Выбираем  $\mathbf{w}_6 = (4, 7)$ . Остаются варианты :  $(4, 8)$ .  $\mathbf{W}_6 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 7), (4, 0), (0, 3), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 7$ , варианты  $\mathbf{w}_7 : (4, 8)$ .

Шаг 1. Выбираем  $\mathbf{w}_7 = (4, 8)$ . Варианты исчерпаны.  $\mathbf{W}_7 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 7), (4, 8), (0, 3), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 8$ , варианты  $\mathbf{w}_8 : (3, 3)$ .

Шаг 1. Выбираем  $\mathbf{w}_8 = (3, 3)$ . Варианты исчерпаны.  $\mathbf{W}_8 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 7), (4, 8), (3, 3), (3, 0), (3, 0) )$ .

Шаг 2.  $\mu = 9$ , варианты  $\mathbf{w}_9 : (3, 5), (3, 6)$ .

Шаг 1. Выбираем  $w_9 = (3, 5)$ . Остаются варианты:  $(3, 6)$ .  $W_9 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 7), (4, 8), (3, 3), (3, 5), (3, 0) )$ .

Шаг 2.  $\mu = 10$ , варианты  $w_{10} : (3, 6)$ .

Шаг 1. Выбираем  $w_{10} = (3, 6)$ . Варианты исчерпаны.  $W_{10} = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 7), (4, 8), (3, 3), (3, 5), (3, 6) )$ .

$$\text{Шаг 2. } h_5 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 1 & 4 & 3 & 7 & 8 & 5 & 6 \end{pmatrix} \right) \right),$$

Шаг 3.  $\mu = 9$ , варианты  $w_9 : (3, 6)$ .

Шаг 1. Выбираем  $w_9 = (3, 6)$ . Варианты исчерпаны.  $W_9 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 7), (4, 8), (3, 3), (3, 6), (3, 0) )$ .

Шаг 2.  $\mu = 10$ , варианты  $w_{10} : (3, 5)$ .

Шаг 1. Выбираем  $w_{10} = (3, 5)$ . Варианты исчерпаны.  $W_{10} = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 7), (4, 8), (3, 3), (3, 6), (3, 5) )$ .

$$\text{Шаг 2. } h_6 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 1 & 4 & 3 & 7 & 8 & 6 & 5 \end{pmatrix} \right) \right).$$

Шаг 3.  $\mu = 6$ , варианты  $w_6 : (4, 8)$ .

Шаг 1. Выбираем  $w_6 = (4, 8)$ . Варианты исчерпаны.  $W_6 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 8), (4, 0), (0, 3), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 7$ , варианты  $w_7 : (4, 7)$ .

Шаг 1. Выбираем  $w_7 = (4, 7)$ . Варианты исчерпаны.  $W_7 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 8), (4, 7), (0, 3), (0, 0), (0, 0) )$ .

Шаг 2.  $\mu = 8$ , варианты  $w_8 : (3, 3)$ .

Шаг 1. Выбираем  $w_8 = (3, 3)$ . Варианты исчерпаны.  $W_8 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 8), (4, 7), (3, 3), (3, 0), (3, 0) )$ .

Шаг 2.  $\mu = 9$ , варианты  $w_9 : (3, 5), (3, 6)$ .

Шаг 1. Выбираем  $w_9 = (3, 5)$ . Остаются варианты:  $(3, 6)$ .  $W_9 = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 8), (4, 7), (3, 3), (3, 5), (3, 0) )$ .

Шаг 2.  $\mu = 10$ , варианты  $w_{10} : (3, 6)$ .

Шаг 1. Выбираем  $w_{10} = (3, 6)$ . Варианты исчерпаны.  $W_{10} = ( (2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 8), (4, 7), (3, 3), (3, 5), (3, 6) )$ .

$$\text{Шаг 2. } h_7 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 1 & 4 & 3 & 8 & 7 & 5 & 6 \end{pmatrix} \right) \right).$$

Шаг 3.  $\mu = 9$ , варианты  $w_9 : (3, 6)$ .

Шаг 1. Выбираем  $w_9 = (3, 6)$ . Варианты исчерпаны.  $W_9 = ((2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 8), (4, 7), (3, 3), (3, 6), (3, 0))$ .

Шаг 2.  $\mu = 10$ , варианты  $w_{10} : (3, 5)$ .

Шаг 1. Выбираем  $w_{10} = (3, 5)$ . Варианты исчерпаны.  $W_{10} = ((2, 2), (2, 4), (1, 1), (1, 3), (4, 4), (4, 8), (4, 7), (3, 3), (3, 6), (3, 5))$ .

$$\text{Шаг 2. } h_8 = \left( \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 1 & 4 & 3 & 8 & 7 & 6 & 5 \end{pmatrix} \right).$$

Шаг 3.  $\mu = 1$ , варианты исчерпаны, конец.

Для сопоставления по трудоемкости двух альтернативных подходов к выявлению симметрий  $Q$ -структуры – известного и предлагаемого – целесообразно исключить разбиения, которые используются при каждом из них и влияние которых на трудоемкости алгоритмов хорошо известно. Это можно сделать, выбрав наиболее трудоемкий случай абсолютно однородной  $Q$ -структуры, когда разбиения невозможны. Такой случай рассмотрен в п. 2.3.2. Забегая вперед, приведем здесь основной вывод из п. 2.3.2: эффект от примененного в алгоритме 1.3 подхода можно получить не хуже, чем от дополнительного разбиения  $d_i$ - и  $d_j$ -классов на подклассы. Учитывая более высокую эффективность такого подхода по сравнению с известным, его можно рекомендовать для  $Q$ -структур с высокой степенью однородности и при больших мощностях множеств  $I, J$ . В то же время известный подход полезен при использовании предикатов (1.32), (1.33), когда множество проверочных подстановок ограничено множеством автоморфизмов  $(i-1)$ -подструктуры, найденных на предыдущем шаге алгоритма 1.1, т.е. сравнительно невелико.

### 1.7. Классификация задач порождения $Q$ -структур по трудоемкости

Процедура конструктивного перечисления неизоморфных  $Q$ -структур, как было показано ранее, включает в себя решение двух комбинаторных задач: порождение вариантов  $Q$ -структуры и перебор подстановок элементов множеств  $I, J$  в процессе проверки каноничности этих вариантов с целью отсекаания изоморфных. Трудоемкость решения каждой из этих задач в основном определяется размерами множеств  $I, J$ . Поэтому классификация задач порождения неизоморфных  $Q$ -структур по трудоемкости – это практически классификация их по размерам множеств  $I, J$ .

$Q$ -Структуры с  $|I| > 1$  и  $|J| > 1$  назовем двумерными и отнесем задачу синтеза таких структур к задачам второго порядка сложности.

Частный случай, когда множество  $I$  или  $J$  состоит из одного элемента, сводится к задачам генерирования неизоморфных векторов. Назовем такие  $Q$ -структуры одномерными и отнесем их генерирование к задачам первого порядка сложности (векторным).

И наконец, если  $Q$ -структура характеризуется параметрами  $|I| = |J| = 1$ , назовем ее точечной, а задачу порождения таких  $Q$ -структур отнесем к задачам единичного порядка сложности. Очевидно, что число шагов решения такой задачи линейно (а не комбинаторно) зависит от числа элементов множества  $L_i$ ,  $i = |I| = 1$ .

Примеры задач первого и второго порядка сложности читатель найдет в гл. 2 и 4, а задачи единичного порядка сложности – в гл. 3.

При разработке методов структурного синтеза неизоморфных систем естественно стремление уменьшить трудоемкость решаемой при этом комбинаторной подзадачи. Обычно этого удается достичь путем максимального использования специфики решаемой задачи, что требует в каждом конкретном случае дополнительных теоретических исследований, направленных на поиск адекватной математической модели структуры системы, а также изобретательности исследователя. Особенной удачей следует считать понижение порядка сложности задачи, так как в этом случае трудоемкость ее решения уменьшается в комбинаторное число раз.

## КОНСТРУКТИВНОЕ ПЕРЕЧИСЛЕНИЕ НЕИЗОМОРФНЫХ КОМБИНАТОРНЫХ ОБЪЕКТОВ

### 2.1. Комбинаторные объекты и структурный синтез материальных систем

Рассмотрим задачи конструктивного перечисления неизоморфных абстрактных систем, которые широко применяются в качестве математических моделей структур материальных систем: это кортежи, системы инцидентностей, графы, в том числе деревья. Каждый из этих комбинаторных объектов можно считать частным случаем  $Q$ -структуры, и, следовательно, к решению задач их конструктивного перечисления применимы общие подходы, методы и алгоритмы, рассмотренные в гл. 1.

Здесь основное внимание уделено разработке приемов понижения трудоемкости указанных методов и алгоритмов с использованием особенностей перечисленных комбинаторных объектов, в том числе приемов понижения порядка сложности решаемой задачи или сведения ее к задачам, для которых такие приемы уже разработаны.

В прикладных задачах структурного синтеза перечисленные математические модели необязательно применяются в чистом виде, иногда структура системы представляется их комбинацией (в гл. 4 рассмотрен пример такого приложения). Но предлагаемые здесь приемы понижения трудоемкости применимы для решения и этих прикладных задач.

### 2.2. Конструктивное перечисление неизоморфных векторов

#### 2.2.1. Кортеж, вектор

Рассмотрим простейший, но достаточно общий комбинаторный объект, который в силу этого обстоятельства играет важную роль как в конструктивной математике, так и в ее приложениях.

Кортеж – это конечная последовательность  $p$  элементов какого-либо множества  $M$ , допускающая повторения [17].

В качестве синонимов термина "кортеж" используются такие выражения [17]: слово в конечном алфавите  $M$ , элемент некоторой декартовой системы множества  $M$ ; элемент свободной подгруппы с множеством образующих  $M$ ; функция, определенная на  $p$  натуральных числах со значениями в множестве  $M$ .

Термин "вектор" принято применять в качестве синонима термина "кортеж" в связи с наиболее естественной интерпретацией кортежей как точек  $p$ -мерного векторного пространства или упорядоченных совокупностей их координат. Вектор представляет собой частный случай более общего алгебраического понятия – матрицы.

Таким образом, понятие "кортеж" в различных вариантах применяется во многих разделах математики: математической логике, математической лингвистике, теории множеств, топологии, алгебре, функциональном анализе, теории автоматов и др. Особенно легко и естественно с помощью кортежей вводятся основные понятия и формулы комбинаторики.

Для вектора,  $\mathbf{a}$  значит, и для кортежа выберем обозначение

$$\mathbf{a} = (\alpha_1, \alpha_2, \mathbf{K}, \alpha_j, \mathbf{K}, \alpha_p),$$

где  $p$  – длина вектора,  $p > 0$ ;  $\alpha_j$  – его  $j$ -я координата,  $1 \leq j \leq p$  ( $j$ -й член или компонент кортежа,  $j$ -я буква слова и т.д., т. е. элемент рассматриваемого конструктивного объекта, частями которого мы не интересуемся).

Конструктивный процесс, результатом которого является вектор  $\mathbf{a}$ , заключается в выборе из множества  $M$  координат этого вектора в порядке нарастания  $j$ .

Конструктивный процесс порождения векторов, как следует из сказанного в гл.1, – это часть конструктивного процесса генерирования взвешенных матриц  $\mathbf{X}$ , представляющих искомые  $Q$ -структуры. Вектор-строка  $\mathbf{x}_i$  используется для представления  $Q$ -структуры в частном случае, когда ее множество  $I$  состоит из одного элемента.

Задача конструктивного перечисления неизоморфных векторов решается здесь при условии, что некоторые координаты эквивалентны в некотором смысле, т.е. взаимозаменяемы.

### 2.2.2. Конструктивное перечисление неизоморфных векторов с неповторяющимися координатами

Пусть  $\mathbf{a} = (\alpha_1, \alpha_2, \mathbf{K}, \alpha_j, \mathbf{K}, \alpha_p)$  –  $p$ -мерный вектор, координаты которого не повторяются,  $\alpha_j \in \{1, 2, \mathbf{K}, p\}$ ;  $A$  – множество векторов  $\mathbf{a}$ ;  $S_p$  – симметрическая группа подстановок координат вектора  $\mathbf{a}$ . Переход от одного вектора к другому осуществляется подстановкой  $g \in S_p$ :

$$\forall \mathbf{a}', \mathbf{a}'' \in A \quad \exists g \in S_p : \mathbf{a}'g = \mathbf{a}''.$$

Пусть  $H$  – некоторая подгруппа группы  $S_p$ . Назовем векторы  $\mathbf{a}'$  и  $\mathbf{a}''$  эквивалентными по  $H$ , если один приводится к другому подстановкой  $h \in H$ .

$$\mathbf{a}' \stackrel{h}{\sim} \mathbf{a}'' \Leftrightarrow \exists h \in H: \mathbf{a}'h = \mathbf{a}''.$$

Таким образом, множество  $A$  разбивается подгруппой  $H$  на классы эквивалентности. В соответствии с теоремой Лагранжа [21, 22] количество таких классов равно  $|S_p|/|H|$ . Пусть стоит задача построения векторов  $\mathbf{a}$ , каждый из которых представляет один и только один класс эквивалентности. Для ее решения упорядочим векторы  $\mathbf{a}$  лексикографически:

$$\mathbf{a}' \stackrel{lex}{\succ} \mathbf{a}'' \Leftrightarrow \exists l \in \{1, 2, \mathbf{K}, p\}: \forall j \langle l \quad \alpha'_j = \alpha''_j, \quad \alpha'_l \langle \alpha''_l.$$

Тогда из каждого класса эквивалентности можно выбрать, например, старший вектор, для которого следующий предикат принимает истинное значение:

$$P(\mathbf{a}) \equiv (\forall h \in H \quad \mathbf{a} \stackrel{lex}{\geq} \mathbf{a}h).$$

Старший вектор  $\mathbf{a}$  объявим каноническим. Очевидно, что в каждом классе эквивалентности множества  $A$  имеется единственный такой вектор. Поэтому построение канонических векторов  $\mathbf{a}$  можно считать решением задачи построения множества представителей от каждого из указанных классов эквивалентности.

Координаты  $\alpha', \alpha''$  вектора  $\mathbf{a}$  назовем  $h$ -эквивалентными, если в группе  $H$  имеется подстановка  $h$ , приводящая одну координату к другой:

$$\alpha' \stackrel{h}{\sim} \alpha'' \Leftrightarrow \exists h \in H: \alpha' h = \alpha''.$$

Множество  $h$ -эквивалентных координат условимся называть  $h$ -классом. Будучи частью вектора,  $h$ -класс также является вектором. Обозначим его  $\mathbf{a}_h$ . Если в группе  $H$  имеются подстановки, меняющие местами  $h$ -классы, назовем такие  $h$ -классы  $H$ -эквивалентными.

Пусть на множестве элементов каждого  $h$ -класса  $\mathbf{a}_h$  вектора  $\mathbf{a}$  (каждом множестве  $H$ -эквивалентных  $h$ -классов) действует симметрическая группа  $H'$ .

**У т в е р ж д е н и е 2.1.** Для того чтобы вектор  $\mathbf{a}$  был каноническим, необходимо и достаточно, чтобы

1) координаты каждого его  $h$ -класса  $\mathbf{a}_h$  располагались в алфавитном порядке;



2) *H*-эквивалентные *h*'-классы были упорядочены лексикографически.

**Доказательство.**

Доказываем от противного. Допустим, что вектор **a** – канонический, но при этом координаты его *h*'-класса **a<sub>h</sub>** расположены не в алфавитном порядке. Тогда в силу симметричности группы *H*', действующей на множестве координат *h*'-класса **a<sub>h</sub>**, в ней найдутся такие подстановки, что переведут его старшую координату на первое место в этом *h*'-классе, следующую по старшинству координату – на второе место в нем, и т. д. В результате такого упорядочения координат вектора **a<sub>h</sub>** он станет старше. Следовательно, и вектор **a** станет старше, так как порядок на множестве его координат, не принадлежащих *h*'-классу, при этом не меняется. Такое заключение противоречит предположению каноничности вектора **a**. Первая часть утверждения 2.1 доказана.

Доказательство второй части утверждения 2.1 ничем не отличается от приведенного доказательства.

Для построения других признаков каноничности вектора **a** требуется изучение действия на него группы подстановок  $h \in H$ , зависящего от ее строения, которое, в свою очередь, определяется особенностями конкретного приложения. Простейшими, но в то же время важными для многих приложений являются циклические группы, действие которых на векторы  $\mathbf{a} \in A$  мы и рассмотрим.

Пусть группа *H* состоит только из циклических подстановок. Представим подстановку  $h_k \in H$  в виде произведения независимых [8] циклов:

$$h_k = h^{(1)}h^{(2)}\mathbf{L}h^{(n)}\mathbf{L}h^{(N)}. \quad (2.1)$$

Пусть  $H_k$  – циклическая группа, порожденная подстановкой  $h_k \in H$ . Назовем координаты  $\alpha', \alpha'' \in \mathbf{a}$  эквивалентными относительно  $h_k$ , если они принадлежат орбите группы  $H_k$ :

$$\alpha' \sim^{h_k} \alpha'' \Leftrightarrow \exists h_k \in H_k : \alpha' h_k^v = \alpha'', \quad (2.2)$$

где *v* – степень подстановки  $h_k$ .

Отношение эквивалентности (2.2) разбивает множество координат вектора **a** на непересекающиеся классы эквивалентности. Выделим множество координат этого вектора, относящихся к общему классу эквивалентности (*h*-классу), не меняя последовательности, в которой они в векторе встречаются. Вектор из выделенных координат также обозначим **a<sub>h</sub>**. Для указания конкретного *h*-класса, соответствующего *n*-му циклу  $h_k \in H$ , условимся

применять обозначение  $\mathbf{a}_h(k, n)$ ,  $k \in \{1, 2, \dots, |H|\}$ ,  $n \in \{1, 2, \dots, N_k\}$ . При этом договоримся элементы группы  $H$  нумеровать в произвольном порядке, а нумерацию циклов согласовывать с очередностью появления в векторе  $\mathbf{a}$  координат-представителей соответствующих  $h$ -классов. Последнее означает, что для каждого конкретного вектора  $\mathbf{a}$  нумерация  $h$ -классов должна предшествовать перенумерации циклов в  $h_k$  (2.1).

**Пример 2.1.** Для вектора  $\mathbf{a} = (8, 7, 5, 3, 1, 9, 10, 4, 6, 2)$   $h$ -классы по  $h_1 = (8, 9) (7, 5) (3, 4)$  – это  $\mathbf{a}_h(1, 1) = (8, 9)$ ,  $\mathbf{a}_h(1, 2) = (7, 5)$ ,  $\mathbf{a}_h(1, 3) = (3, 4)$ , а по  $h_2 = (7, 1, 4) (5, 3, 6)$  –  $\mathbf{a}_h(2, 1) = (7, 1, 4)$ ,  $\mathbf{a}_h(2, 2) = (5, 3, 6)$ .

Если группа  $H_k$  фиксирует координату  $\mathbf{a}_j \in \mathbf{a}$ , то ее также можно рассматривать как отдельный  $h$ -класс, содержащий единственную координату. Обозначение  $\mathbf{a}_h(k, n)$  будем применять только к  $h$ -классам, содержащим не менее двух координат.

Число  $r_n$  координат в  $h$ -классе определяется длиной соответствующего цикла. Числа  $r_{n'}$ ,  $r_{n''}$  могут быть между собой взаимно простыми или иметь общие множители. Среди  $h$ -классов с одинаковыми длинами  $r_n$ ,  $n \in \{1, 2, \dots, N_k\}$ , выделим старший, т.е. с минимальным номером  $n$ . Будем называть его  $r_n$ -старшим, а остальные  $h$ -классы, содержащие  $r_n$  координат, –  $r_n$ -младшими. Наименьшее общее кратное чисел  $r_1, r_2, \mathbf{K}, r_n$  обозначим символом  $R_n$ .

Пусть  $h_{k/n}$  – ограничение подстановки  $h_k \in H_k$  на координаты вектора  $\mathbf{a}$ , относящиеся к  $h$ -классу  $\mathbf{a}_h(k, n)$ . Перестановки координат вектора  $\mathbf{a}$  внутри  $h$ -класса  $\mathbf{a}_h(k, n)$  осуществляются с помощью подстановки  $h_{k/n}^v$ ,  $v \in \{1, 2, \dots, r_n\}$ .

**У т в е р ж д е н и е 2.2.** Для того чтобы вектор  $\mathbf{a}$  был каноническим, необходимо и достаточно, чтобы для каждого его  $r_n$ -старшего  $h$ -класса  $\mathbf{a}_h(k, n)$ ,  $k=1, 2, \dots, |H|$ ,  $n \in \{1, 2, \dots, N_k\}$ , предикат

$$P(\mathbf{a}_h(k, n)) \equiv (\forall v \in N_n \quad \mathbf{a}_h(k, n) h_{k/n}^v \stackrel{lex}{\leq} \mathbf{a}_h(k, n)), \quad (2.3)$$

где

$$N_n = \begin{cases} 0, 1, 2, \mathbf{K}, r_1 - 1 & \text{при } n = 1; \\ 0, R_{(n-1)}, 2R_{(n-1)}, \mathbf{K}, R_n - 1 & \text{при } n \neq 1, \end{cases}$$

принимал истинное значение.

До к а з а т е л ь с т в о. Доказательство ведем от противного.

*Необходимость.* Пусть  $\mathbf{a}$  – старший вектор и при этом для некоторых  $r_n$ -старших  $h$ -классов предикат (2.3) принимает ложное значение. Пусть  $n'$  – минимальный номер такого класса. Тогда можно найти такое  $\nu \in N_n$ , что под действием  $h_k^\nu$   $h$ -класс  $\mathbf{a}_h(k, n')$  станет старше при неизменности той части вектора  $\mathbf{a}$ , которая находится левее этого  $h$ -класса. В результате вектор  $\mathbf{a}$  станет старше, что противоречит первоначальному утверждению. Рассуждая аналогично по отношению к каждому  $r_n$ -старшему  $h$ -классу в порядке нарастания  $n$ , получим, что для каждого из них предикат (2.3) принимает истинное значение.

*Достаточность.* Допустим, что для любого  $r_n$ -старшего  $h$ -класса вектора  $\mathbf{a}$   $P(\mathbf{a}_h(k, n))$  истинно, но при этом вектор  $\mathbf{a}$  не является старшим.

Увеличение\* вектора  $\mathbf{a}$  может быть осуществлено только путем перемещения его координат влево на места, занимаемые младшими координатами. Перемещение осуществляется только с помощью подстановок  $h_k \in H_k$ , следовательно, только внутри некоторого  $h$ -класса. Из истинности предиката (2.3) для  $\mathbf{a}_h(k, n)$  следует, что этот  $h$ -класс не может быть увеличен подстановками  $h_{k/n}^\nu, \nu \in N_n$ . При этом ни один из  $h$ -классов, находящихся левее его, не может быть увеличен подстановками  $h_k^\nu, \nu \in N_n$ . А другие подстановки, которые не обеспечивают неизменности части вектора  $\mathbf{a}$ , предшествующей  $\mathbf{a}_h(k, n)$ , исключены. Перемещение координат внутри любого  $r_n$ -младшего  $h$ -класса под действием  $h_k^\nu, \nu \in N_n$ , происходит только одновременно с перемещением координат внутри соответствующего  $r_n$ -старшего  $h$ -класса. Поэтому невозможно увеличение  $r_n$ -младшего  $h$ -класса без уменьшения соответствующего  $r_n$ -старшего  $h$ -класса. Таким образом, вектор  $\mathbf{a}$  при истинном значении  $P(\mathbf{a}_h(k, n))$  не может быть не старшим.

Утверждение доказано.

С л е д с т в и е 2.2.1. Для того чтобы привести вектор  $\mathbf{a}$  к канонической форме, необходимо и достаточно каждый его старший  $h$ -класс в порядке нарастания  $n$ , начиная с  $n=1$ , увеличить с помощью подстановок  $h_k^\nu, \nu \in N_n$ , насколько это возможно.

---

\* Здесь и далее имеется в виду лексикографическое увеличение или уменьшение, если речь идет о векторе.

**Д о к а з а т е л ь с т в о.** Указанные подстановки позволяют это сделать, не меняя части вектора  $\mathbf{a}$ , предшествующей каждому из увеличиваемых  $r_n$ -младших  $h$ -классов,  $n = 1, 2, \dots, N_k$ . При этом вместе с  $r_n$ -старшим  $h$ -классом меняется каждый из соответствующих  $r_n$ -младших  $h$ -классов. Существенное различие между ними состоит в том, что хотя бы одна из координат первого всегда находится в векторе  $\mathbf{a}$  левее старшей координаты второго. Поэтому после увеличения первой координаты в с  $r_n$ -старшем  $h$ -классе под действием  $h_k^v$ ,  $v \in N_n$ , вектор  $\mathbf{a}$  становится старше независимо от изменений в соответствующих  $r_n$ -младших  $h$ -классах. Максимальное лексикографическое увеличение вектора  $\mathbf{a}$  наступает в результате перемещения на первое место старшей координаты в каждом из  $r_n$ -старших  $h$ -классов  $\mathbf{a}_h(k, n)$ ,  $n = 1, 2, \dots, N_k$ . Вектор  $\mathbf{a}$  в таком случае становится каноническим по определению.

**С л е д с т в и е 2.2.2.** Пусть циклы подстановки  $h_k \in H_k$ , действующей на вектор  $\mathbf{a}$ , являются взаимно простыми. Тогда для приведения его к канонической форме необходимо и достаточно установить в каждом  $r_n$ -старшем  $h$ -классе  $\mathbf{a}_h(k, n)$ ,  $n = 1, 2, \dots, N_k$ , старшую координату на первое место, сдвигая ее внутри своего цикла и внося соответствующие изменения в  $r_n$ -младшие  $h$ -классы с тем же значением  $r_n$ .

Действительно, в этом случае  $R_n$  является наименьшим общим кратным чисел  $R_n$  и  $R_{(n-1)}$ , из чего следует, что среди подстановок  $h_k^v$ ,  $v \in N_n$ , всегда найдется такая, что обеспечит перемещение старшей координаты  $r_n$ -старшего  $h$ -класса на первое место в нем.

Из утверждений 2.1, 2.2 и следствий к ним видно, что каноничность вектора  $\mathbf{a}$  можно проверять по частям, какими являются его  $h$ -классы по различным  $h \in H$ , применяя к каждому  $h$ -классу свой набор проверочных подстановок. Этот набор представляет собой ограничения на просматриваемый  $h$ -класс только тех подстановок  $h \in H$ , которые не меняют части вектора  $\mathbf{a}$ , предшествующей этому  $h$ -классу.

Рассмотрим примеры применения перечисленных признаков каноничности векторов  $\mathbf{a}$  для отбора по одному представителю от каждого класса эквивалентности множества  $A = \{\mathbf{a}\}$ .

**Пример 2.2.** Пусть множество  $A$  состоит из следующих элементов:

$$\begin{aligned} \mathbf{a}_1 = (1, 2, 3, 4); & \quad \mathbf{a}_2 = (1, 2, 4, 3); & \quad \mathbf{a}_3 = (1, 3, 2, 4); & \quad \mathbf{a}_4 = (1, 3, 4, 2); \\ \mathbf{a}_5 = (1, 4, 2, 3); & \quad \mathbf{a}_6 = (1, 4, 3, 2); & \quad \mathbf{a}_7 = (2, 1, 3, 4); & \quad \mathbf{a}_8 = (2, 1, 4, 3); \\ \mathbf{a}_9 = (2, 3, 1, 4); & \quad \mathbf{a}_{10} = (2, 3, 4, 1); & \quad \mathbf{a}_{11} = (2, 4, 1, 3); & \quad \mathbf{a}_{12} = (2, 4, 3, 1); \end{aligned}$$

$$\begin{aligned} \mathbf{a}_{13} &= (3, 1, 2, 4); & \mathbf{a}_{14} &= (3, 1, 4, 2); & \mathbf{a}_{15} &= (3, 2, 1, 4); & \mathbf{a}_{16} &= (3, 2, 4, 1); \\ \mathbf{a}_{17} &= (3, 4, 1, 2); & \mathbf{a}_{18} &= (3, 4, 2, 1); & \mathbf{a}_{19} &= (4, 1, 2, 3); & \mathbf{a}_{20} &= (4, 1, 3, 2); \\ \mathbf{a}_{21} &= (4, 2, 1, 3); & \mathbf{a}_{22} &= (4, 2, 3, 1); & \mathbf{a}_{23} &= (4, 3, 1, 2); & \mathbf{a}_{24} &= (4, 3, 2, 1). \end{aligned}$$

И пусть  $H_1 = (e, (1, 2))$ . В соответствии с условием 1 утверждения 2.1 в этом случае каноническими являются векторы  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3, \mathbf{a}_4, \mathbf{a}_5, \mathbf{a}_6, \mathbf{a}_{13}, \mathbf{a}_{14}, \mathbf{a}_{17}, \mathbf{a}_{19}, \mathbf{a}_{20}, \mathbf{a}_{23}$ . Пусть  $H_2 = \langle e, (1, 3), (2, 4), (1, 2)(3, 4), (1, 4)(2, 3) \rangle$ . Тогда каноническими в соответствии с условиями утверждений 2.1 и 2.2 являются векторы  $\mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3$ .

**Пример 2.3.** Убедимся, что вектор

$$\mathbf{a} = (5, 8, 3, 6, 11, 7, 4, 9, 12, 1, 10, 2)$$

при условии действия на него циклической группы подстановок с образующей

$$h_k = (8, 3), (6, 11, 4), (7, 12), (9, 1, 10, 2)$$

не является каноническим, и приведем его к канонической форме. Для этого вычисляем набор степеней  $v$  проверочных подстановок для каждого  $r_n$ -старшего  $h$ -класса:

$$n = 1, r_1 = R_1 = 2, v = 0, 1;$$

$$n = 2, r_2 = 3, R_2 = 6, v = 0, 2, 4;$$

$$n = 4, r_4 = 4, R_4 = 12, v = 0, 6.$$

Для того чтобы убедиться, что заданный вектор  $\mathbf{a}$  каноническим не является, достаточно применить к  $r_1$ -му  $h$ -классу соответствующие проверочные подстановки.

Преобразовав каждый из  $r_n$ -старших  $h$ -классов,  $n = 1, 2, 4$ , вместе с правой от него частью вектора  $\mathbf{a}$  найденными проверочными подстановками, получаем канонический вектор

$$\mathbf{a} = (5, 3, 8, 4, 6, 12, 11, 1, 7, 10, 2, 9).$$

### 2.2.3. Конструктивное перечисление неизоморфных векторов с повторяющимися координатами

Рассмотрим теперь множество  $X$   $p$ -мерных векторов  $\mathbf{x}$ , отличающихся от векторов  $\mathbf{a}$  тем, что они содержат повторяющиеся координаты наряду с неповторяющимися. Пусть требуется построить множество векторов  $\mathbf{x}$ , не эквивалентных по  $H$ .

На множестве векторов  $\mathbf{x}$  введем лексикографический порядок, предварительно упорядочив типы координат, из которых они строятся, т.е. выбрав алфавит. Координаты  $p$ -мерного вектора  $\mathbf{a}$  можно рассматривать как

порядковые номера координат вектора  $\mathbf{x}$ . В этом случае  $h$ - и  $H$ -разбиение вектора  $\mathbf{a}$  индуцирует соответствующее разбиение вектора  $\mathbf{x}$ . Обозначим  $\mathbf{x}_h(k, n)$   $h$ -класс вектора  $\mathbf{x}$ , соответствующий  $h$ -классу  $\mathbf{a}_h(k, n)$ .

Пусть на множестве координат каждого  $h$ -класса  $\mathbf{x}_h(k, n)$  вектора  $\mathbf{x}$  ( $H$ -эквивалентных  $h$ -классов этого вектора) действует симметрическая группа подстановок. Тогда к нему, как и к вектору  $\mathbf{a}$ , применимо утверждение 2.1. Действие на вектор  $\mathbf{x}$  циклических подстановок требует отдельного рассмотрения.

В отличие от  $h$ -класса вектора  $\mathbf{a}$   $h$ -класс вектора  $\mathbf{x}$  дополнительно разбивается на классы эквивалентности другой природы: две одинаковые координаты можно поменять местами, и при этом вектор  $\mathbf{x}$  не изменится. Следовательно, среди подстановок  $h_k \in H_k$  могут быть такие, к которым  $h$ -класс  $\mathbf{x}_h(k, n)$  инвариантен. Такие подстановки можно рассматривать как отображающие внутренние симметрии этого  $h$ -класса. В частности это могут быть повторяющиеся части  $h$ -класса  $\mathbf{x}_h(k, n)$ , к циклической подстановке которых он инвариантен. Пусть  $t_n$  – длина такой повторяющейся части  $h$ -класса  $\mathbf{x}_h(k, n)$ . В случае, если в  $h$ -классе  $\mathbf{x}_h(k, n)$  отсутствуют повторяющиеся части,  $t_n = r_n$ . Наименьшее общее кратное чисел  $t_1, t_2, \dots, t_n$  обозначим символом  $T_n$ .

**У т в е р ж д е н и е 2.3.** *Для того чтобы вектор  $\mathbf{x}$  при действии на него подстановки  $h_k \in H_k$  был каноническим, необходимо и достаточно, чтобы для каждого его  $h$ -класса  $\mathbf{x}_h(k, n)$ ,  $k = 1, 2, \dots, |H|$ ,  $n = 1, 2, \dots, N_k$ , предикат*

$$P[\mathbf{x}_h(k, n)] \equiv \{ \forall \mathbf{n} \in N_n \quad [\mathbf{x}_h(k, n)] h_{k/n}^{\mathbf{n}} \leq \mathbf{x}_h(k, n) \},$$

где

$$N_n = \begin{cases} 0, 1, 2, \mathbf{K}, t_1 - 1 & \text{при } n = 1; \\ 0, T_{(n-1)}, 2T_{(n-1)}, \mathbf{K}, T_n - 1 & \text{при } n \neq 1, \end{cases}$$

принимал истинное значение.

Д о к а з а т е л ь с т в о утверждения 2.3 аналогично доказательству утверждения 2.2. Только в нем следует учесть появление дополнительной возможности увеличения каждого вектора  $\mathbf{x}_h(k, n)$  за счет внутренних симметрий старших по отношению к нему  $h$ -классов. Это сделано путем замены элементов  $r_1, R_{(n-1)}, 2R_{(n-1)}, \mathbf{K}, R_n$  множества  $N_n$  элементами

$t_1, T_{(n-1)}, 2T_{(n-1)}, \mathbf{K}, T_n$ , обеспечивающими неизменность части вектора  $\mathbf{x}$ , предшествующей  $h$ -классу  $\mathbf{x}_h(k, n)$  с учетом указанных симметрий. Этой заменой учтена при наличии указанных симметрий также возможность увеличения  $r_n$ - младших  $h$ -классов без изменения соответствующих  $r_n$ -старших  $h$ -классов.

Из утверждения 2.3 следует, что каноничность вектора  $\mathbf{x}$  также можно проверять по частям.

**С л е д с т в и е 2.3.** Для того чтобы привести вектор  $\mathbf{x}$  к канонической форме, необходимо и достаточно каждый его  $h$ -класс в порядке нарастания  $n$ , начиная с  $n = 1$ , увеличить с помощью подстановок  $h_k^v, v \in N_n$ , насколько это возможно.

Доказывается аналогично следствию 2.2.1. с учетом того, что неизменность части вектора  $\mathbf{x}$ , предшествующей каждому  $t_n$ -старшему  $h$ -классу, и возможность увеличения  $r_n$ - младших  $h$ -классов обеспечиваются подстановками  $h_k^v, v \in N_n$ , определяемыми согласно утверждению 2.3.

**Пример 2.4.** Пусть требуется привести к канонической форме вектор

$$\mathbf{x} = (1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0)$$

при условии действия на него циклической группы подстановок с образующей

$$h_k = (1,2,3,4), (5,6,7,8,9,10), (11,12,13,14,15,16), (17,18,19,20), (21,22,23,24,25,26).$$

Вычисляем набор степеней  $v$  проверочных подстановок для каждого  $t_n$ -старшего  $h$ -класса:

$$n = 1, t_1 = T_1 = 2, v = 0, 1;$$

$$n = 2, t_2 = 2, T_2 = 2, v = 0;$$

$$n = 3, t_3 = 3, T_3 = 6, v = 0, 2, 4;$$

$$n = 4, t_4 = 4, T_4 = 12, v = 0, 4, 8;$$

$$n = 5, t_4 = 6, T_4 = 12, v = 0, 6.$$

Преобразовав каждый из  $h$ -классов  $\mathbf{x}_h(k, n)$ ,  $n = 1, 2, 3, 4, 5$ , вместе с правой от него частью вектора  $\mathbf{x}$  найденными проверочными подстановками, получаем канонический вектор

$$\mathbf{x} = (1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0, 1).$$

### 2.3. Конструктивное перечисление неизоморфных систем инцидентностей

#### 2.3.1. Алгоритм конструктивного перечисления

Здесь пойдет речь о системе инцидентностей  $S = (I, J, \Phi)$ , где  $I, J$  – конечные множества, а  $\Phi$  – отношение инцидентностей между их элементами. Такую систему принято представлять матрицей инцидентностей  $\mathbf{X}_{|I| \times |J|}$ , где  $x_{ij} = 1$ , если  $i$ -й элемент из  $I$  инцидентен  $j$ -му элементу из  $J$ , или  $x_{ij} = 0$  в противном случае.

Система инцидентностей – это частный случай  $Q$ -структуры с абсолютно однородными множествами  $L_i$ ,  $i = 1, 2, \dots, |I|$ . При этом множества  $I, J$  системы инцидентностей могут быть разбиты на подмножества, например, по числу инцидентностей элементов  $i \in I$  элементам  $j \in J$  и наоборот, – элементов  $j \in J$  элементам  $i \in I$ . В этом случае можно рассматривать систему инцидентностей как  $Q$ -структуру с заданными отношениями эквивалентности  $F_I, F$  и порядка  $T_I, T$  только на множествах  $I, J$  соответственно. Но возможны и абсолютно однородные системы инцидентностей, как, например, некоторые виды блок-схем, которые получаются, если потребовать, чтобы: 1) каждый элемент  $i \in I$  был инцидентен в точности  $r$  элементам  $J$ ; 2) каждый элемент  $j \in J$  был инцидентен в точности  $s$  элементам  $I$ . В этом случае система инцидентностей – это абсолютно однородная  $Q$ -структура.

Из сказанного следует, что к конструктивному перечислению неизоморфных систем инцидентностей применимы средства, предложенные в п. 1.5 для  $Q$ -структур. При этом алгоритм 1.1, предназначенный для генерирования неизоморфных  $Q$ -структур, может быть упрощен с учетом перечисленных особенностей системы инцидентностей. Алгоритм 2.1 генерирования неизоморфных систем инцидентностей представлен как модификация алгоритма 1.1 из п. 1.5.5.

Алгоритм 2.1.

Шаг 1. Положить  $i = 1$ .

Шаг 2. Пользуясь информацией о симметриях подматрицы  $\mathbf{X}_{(i-1)}$ , разбить множества строк и столбцов матрицы  $\mathbf{X}_i$  на  $F_I$ - и  $F'$ -классы.

Шаг 3. С учетом полученного разбиения и в соответствии с утверждениями 1.8, 1.9 построить очередной вектор  $\mathbf{x}_i$  в порядке лексико-



графического невозрастания в алфавите (1, 0). (При переходе от шага 2 к шагу 3 очередным вектором  $\mathbf{x}_i$  считать самый старший в этом алфавите.)

Шаг 4. Проверить полученный вектор  $\mathbf{x}_i$  по предикату (1.32). Если предикат принимает ложное значение, перейти к шагу 3.

Шаг 5. Проверить подматрицу  $\mathbf{X}_{\alpha i}$  по предикату (1.33). Если предикат принимает ложное значение, перейти к шагу 3.

Шаг 6. Зафиксировать канонический вариант подматрицы  $\mathbf{X}_i$ .

Шаг 7. Если  $i < |I|$ , положить  $i := i + 1$  и идти к шагу 2. Если  $i = |I|$ , вернуться к максимальному значению  $i$ , при котором не исчерпаны варианты вектора  $\mathbf{x}_i$ , и с этим значением  $i$  идти к шагу 3. Если таких нет, конец.

Рассмотрим работу алгоритма 2.1 на примере.

**Пример 2.5.** Пусть имеется 5-подматрица  $\mathbf{X}_5$  искомой матрицы инцидентностей  $\mathbf{X}$ , отражающая инцидентности пяти элементов множества  $I$  элементам множества  $J$ :

$$\mathbf{X}_5 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}.$$

Требуется перечислить неизоморфные варианты подматрицы  $\mathbf{X}_6$ , в которой отражена инцидентность шестого ( $i = 6$ ) элемента множества  $I$  двум элементам множества  $J$ . Повторение уже имеющихся строк в матрице  $\mathbf{X}_6$  исключить.

Заметим, что автоморфизмы подматрицы  $\mathbf{X}_5$ , которые необходимы для проверки каноничности вектора  $\mathbf{x}_6$  на шаге 4 алгоритма 2.1, составляют группу  $H_5 = \{(e_I, e_J), ((2, 3)(4, 5), (1, 2)(5, 6))\}$ . Множество  $H_{\alpha i 0} = H_{360}$  автоморфизмов, необходимых для проверки каноничности подматрицы  $\mathbf{X}_{\alpha i} = \mathbf{X}_{36}$  на шаге 5 алгоритма, состоит из подстановок:  $((6, 5), (1, 2)(5, 6)), ((5, 6, 4), (1, 2)(5, 6)), ((6, 4, 5), (1, 2)(5, 6)), ((6, 4), (1, 2)(5, 6))$ .

Шаг 7.  $i = 6$ .

Шаг 2. Разбиваем множества  $I$  и  $J$  матрицы  $\mathbf{X}_6$ , в которой пока не заполнена шестая строка, на классы эквивалентности с учетом симметрий  $\mathbf{X}_5$ :

$$\mathbf{X}_6 = \left\| \begin{array}{|c|c|c|c|c|} \hline 1 & 1 & 1 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 1 & 1 & 0 \\ \hline 0 & 1 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline \end{array} \right\|.$$

Шаг 3  $\mathbf{x}_6 = (0, 0, 0, 1, 1, 0)$ .

Шаг 4.

$$(0, 0, 0, 1, 1, 0) \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 4 & 6 & 5 \end{pmatrix} = (0, 0, 0, 1, 0, 1),$$

вектор  $\mathbf{x}_6$  – канонический.

Шаг 5.  $I_\alpha = I_2 = \{4, 5, 6\}$ ,  $J_\alpha = \{1, 2, 3, 4, 5, 6\}$ ,

$$\begin{aligned} & H_{I_2}^{-1} \mathbf{X}_{2i} H_{J_2} = \\ & \left\| \begin{array}{|c|c|} \hline 1 & 0 & 0 \\ \hline 0 & 0 & 1 \\ \hline 0 & 1 & 0 \\ \hline \end{array} \right\| \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 & 1 & 0 \\ \hline \end{array} \right\| \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 \\ \hline \end{array} \right\| = \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 & 1 & 0 \\ \hline \end{array} \right\| \mathbf{p} \mathbf{X}_{ai} = \mathbf{X}_{2i}, \end{aligned}$$

$$\left\| \begin{array}{|c|c|} \hline 0 & 1 & 0 \\ \hline 0 & 0 & 1 \\ \hline 1 & 0 & 0 \\ \hline \end{array} \right\| \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 & 1 & 0 \\ \hline \end{array} \right\| \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 \\ \hline \end{array} \right\| = \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline \end{array} \right\| \mathbf{p} \mathbf{X}_{ai} = \mathbf{X}_{2i},$$

$$\left\| \begin{array}{|c|c|} \hline 0 & 0 & 1 \\ \hline 1 & 0 & 0 \\ \hline 0 & 1 & 0 \\ \hline \end{array} \right\| \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 1 & 0 & 1 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 & 1 & 0 \\ \hline \end{array} \right\| \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 & 0 \\ \hline \end{array} \right\| = \left\| \begin{array}{|c|c|c|c|c|c|} \hline 0 & 0 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 & 1 & 0 \\ \hline \end{array} \right\| \mathbf{p} \mathbf{X}_{ai} = \mathbf{X}_{2i},$$

$$\left\| \begin{array}{c} 1 \ 0 \ 0 \\ 0 \ 1 \ 0 \\ 0 \ 0 \ 1 \end{array} \right\| \left\| \begin{array}{c} 0 \ 0 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \end{array} \right\| \left\| \begin{array}{c} 0 \ 1 \ 0 \ 0 \ 0 \ 0 \\ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \end{array} \right\| = \left\| \begin{array}{c} 0 \ 0 \ 0 \ 1 \ 0 \ 1 \\ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \end{array} \right\| \mathbf{p} \mathbf{X}_{ai} = \mathbf{X}_{2i},$$

Матрица  $\mathbf{X}_6$  – каноническая.

Шаг 6. Фиксируем каноническую матрицу  $\mathbf{X}_6$ .

Шаг 7.  $i = 6$ .

Шаг 3.  $\mathbf{x}_6 = (0, 0, 0, 0, 1, 1)$ .

Шаг 4.

$$(0, 0, 0, 0, 1, 1) \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 3 & 4 & 6 & 5 \end{pmatrix} = (0, 0, 0, 0, 1, 1) = \mathbf{x}_6,$$

вектор  $\mathbf{x}_6$  – канонический.

Шаг 5.

$$\left\| \begin{array}{c} 1 \ 0 \ 0 \\ 0 \ 0 \ 1 \\ 0 \ 1 \ 0 \end{array} \right\| \left\| \begin{array}{c} 0 \ 0 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \end{array} \right\| \left\| \begin{array}{c} 0 \ 1 \ 0 \ 0 \ 0 \ 0 \\ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \end{array} \right\| = \left\| \begin{array}{c} 0 \ 0 \ 1 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \\ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \end{array} \right\| \mathbf{p} \mathbf{X}_{ai} = \mathbf{X}_{2i},$$

$$\left\| \begin{array}{c} 0 \ 1 \ 0 \\ 0 \ 0 \ 1 \\ 1 \ 0 \ 0 \end{array} \right\| \left\| \begin{array}{c} 0 \ 0 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \end{array} \right\| \left\| \begin{array}{c} 0 \ 1 \ 0 \ 0 \ 0 \ 0 \\ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \end{array} \right\| = \left\| \begin{array}{c} 0 \ 0 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 1 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \end{array} \right\| \mathbf{p} \mathbf{X}_{ai} = \mathbf{X}_{2i},$$

$$\left\| \begin{array}{c} 0 \ 0 \ 1 \\ 1 \ 0 \ 0 \\ 0 \ 1 \ 0 \end{array} \right\| \left\| \begin{array}{c} 0 \ 0 \ 1 \ 0 \ 1 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 1 \ 1 \ 0 \end{array} \right\| \left\| \begin{array}{c} 0 \ 1 \ 0 \ 0 \ 0 \ 0 \\ 1 \ 0 \ 0 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \\ 0 \ 0 \ 0 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \end{array} \right\| = \left\| \begin{array}{c} 0 \ 0 \ 0 \ 0 \ 1 \ 1 \\ 0 \ 0 \ 1 \ 0 \ 0 \ 1 \\ 0 \ 0 \ 1 \ 0 \ 1 \ 0 \end{array} \right\| \mathbf{p} \mathbf{X}_{ai} = \mathbf{X}_{2i},$$

$$\left\| \begin{array}{ccc} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{array} \right\| \left\| \begin{array}{cccccc} 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{array} \right\| \left\| \begin{array}{cccccc} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{array} \right\| = \left\| \begin{array}{cccccc} 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{array} \right\| \mathbf{p} \quad \mathbf{X}_{\alpha i} = \mathbf{X}_{2i}.$$

Матрица  $\mathbf{X}_6$  – каноническая.

Шаг 6 Фиксируем канонический вариант матрицы  $\mathbf{X}_6$ .

Шаг 7.  $i = 5$ .

Вектор-строка матрицы инцидентностей – это булев  $|J|$ -мерный вектор  $\mathbf{x}_i$ , т.е. вектор с повторяющимися координатами. Конструктивное перечисление неизоморфных вариантов таких векторов рассмотрено в п. 2.2.3. С учетом рекомендаций, следующих из п. 2.2.3, может быть повышена эффективность выполнения шагов 3 – 5 алгоритма 2.1.

### 2.3.2. Выявление симметрий системы инцидентностей

Исходя из сказанного в п. 2.3.1, задачу выявления симметрий системы инцидентностей будем рассматривать как частный случай аналогичной задачи для  $Q$ -структуры. В этом частном случае все ненулевые элементы матрицы  $\mathbf{X}$   $Q$ -структуры  $\delta_i$ -эквивалентны между собой. В основу алгоритма 1.3 решения этой задачи, как следует из п. 1.6.2, положены утверждения 1.11, 1.12. Очевидно, что оба утверждения справедливы и для системы инцидентностей, только в утверждении 1.11 в этом случае теряет смысл проверка  $\delta_i$ -эквивалентности, так как она заведомо даст положительный результат. С этой поправкой алгоритм 1.3 применим для выявления симметрий системы инцидентностей. Рассмотрим пример такого применения.

**Пример 2.6.** Определить группу автоморфизмов матрицы

$$\mathbf{X} = \left\| \begin{array}{cccccc} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{array} \right\|.$$

Исходные данные для работы алгоритма:  $\mu = 1$ ,  $\mathbf{W}_{(\mu-1)} = \mathbf{W}_0 = ((0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0))$ .

Допустимые варианты  $\mathbf{w}_1$ : (1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 4), (3, 1), (3, 3), (3, 5), (4, 2), (4, 4), (4, 6), (5, 3), (5, 5), (5, 6), (6, 4), (6, 5), (6, 6).

Приведем часть результатов работы алгоритма 1.3 в виде первых пяти последовательно сгенерированных множеств  $\mathbf{W}_\mu$  :

1. ((1, 1), (1, 2), (1, 3), (2, 1), (2, 2), (2, 4), (3, 1), (3, 3), (3, 5), (4, 2), (4, 4), (4, 6), (5, 3), (5, 5), (5, 6), (6, 4), (6, 5), (6, 6)).

2. ((1, 1), (1, 2), (1, 3), (3, 1), (3, 2), (3, 0), (0, 1), (0, 3), (0, 0), (0, 2), (0, 0), (0, 0), (0, 3), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0)).

3. ((1, 1), (1, 3), (1, 2), (2, 1), (2, 3), (2, 0), (0, 1), (0, 2), (0, 0), (0, 3), (0, 0), (0, 0), (0, 2), (0, 0), (0, 0), (0, 0), (0, 0), (0, 0)).

4. ((1, 1), (1, 3), (1, 2), (3, 1), (3, 3), (3, 5), (2, 1), (2, 2), (2, 4), (5, 3), (5, 5), (5, 6), (4, 2), (4, 4), (4, 6), (6, 5), (6, 4), (6, 6)).

5. ((1, 2), (1, 1), (1, 3), (2, 2), (2, 1), (2, 4), (4, 2), (4, 3), (4, 0), (0, 1), (0, 4), (0, 0), (0, 3), (0, 0), (0, 0), (0, 4), (0, 0), (0, 0)).

Множества с номерами 1 и 4 представляют автоморфизмы матрицы

$$h_1 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 3 & 4 & 5 & 6 \end{pmatrix} \right); \quad h_2 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 3 & 2 & 5 & 4 & 6 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 3 & 2 & 5 & 4 & 6 \end{pmatrix} \right)$$

соответственно.

Построение множества  $\mathbf{W}_\mu$  с номерами 2, 3, 5 обрывается после проверки их корректности, предусмотренной шагом 1 алгоритма 1.3. Далее приведены остальные автоморфизмы матрицы  $\mathbf{X}$ , соответствующие множествам  $\mathbf{W}_\mu$  с порядковыми номерами 17, 20, 29, 32, 41, 44, 53, 56, 69, 72 (читателю предоставляется проверить самостоятельно):

$$h_3 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 4 & 3 & 6 & 5 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 1 & 4 & 3 & 6 & 5 \end{pmatrix} \right); \quad h_4 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 4 & 1 & 6 & 3 & 5 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 4 & 1 & 6 & 3 & 5 \end{pmatrix} \right);$$

$$h_5 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 1 & 5 & 2 & 6 & 4 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 1 & 5 & 2 & 6 & 4 \end{pmatrix} \right); \quad h_6 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 5 & 1 & 6 & 2 & 4 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 3 & 5 & 1 & 6 & 2 & 4 \end{pmatrix} \right);$$

$$h_7 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 4 & 2 & 6 & 1 & 5 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 4 & 2 & 6 & 1 & 5 & 3 \end{pmatrix} \right); \quad h_8 = \left( \left( \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 4 & 6 & 2 & 5 & 1 & 3 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 4 & 6 & 2 & 5 & 1 & 3 \end{pmatrix} \right);$$

$$h_9 = \left( \binom{123456}{536142}, \binom{123456}{536142} \right); \quad h_{10} = \left( \binom{123456}{563412}, \binom{123456}{563412} \right);$$

$$h_{11} = \left( \binom{123456}{645231}, \binom{123456}{645231} \right); \quad h_{12} = \left( \binom{123456}{654321}, \binom{123456}{654321} \right).$$

Остановимся на оценке трудоемкости процедуры, реализующей выявления группы автоморфизмов системы инцидентностей по алгоритму 1.3.

Отмеченное выше изменение в алгоритме 1.3 означает увеличение его трудоемкости. Худшим по трудоемкости является случай абсолютно однородной системы инцидентностей, когда не только элементы всех множеств  $L_i$ ,  $i = 1, 2, \dots, |I|$ ,  $\delta_i$ -эквивалентны, но и каждое из множеств  $I$  и  $J$  составляет один общий класс эквивалентности. Таким образом, это случай, когда ни к одному из этих множеств нельзя применить разбиения на классы эквивалентности для сокращения числа просматриваемых вариантов упорядоченных множеств  $\mathbf{W}_\mu$ , и число последних резко возрастает. Например, варианты замены  $\mathbf{w}_\mu = \mathbf{w}_1$  составляют, как следует из утверждения 1.11, все множество  $\mathbf{W}_s$ ; следовательно, их число принимает максимальное из возможных значений, равное  $|\mathbf{W}_s|$  (см. пример 2.6). Правда, с ростом  $\mu$  число вариантов замены быстро сокращается. Выполним оценку алгоритма 1.3 именно для такого худшего случая. Пусть  $q_i$  – число ненулевых элементов в строке матрицы  $\mathbf{X}$ , а  $q_j$  – в столбце.

Трудоемкость алгоритма 1.3 оценивается числом  $N$  просматриваемых вариантов  $\mathbf{W}_\mu$ . Оно равно произведению чисел допустимых вариантов каждого элемента  $\mathbf{w} \in \mathbf{W}_\mu$ , так как алгоритм предусматривает перебор этих вариантов по принципу "каждый с каждым".

Покажем, что трудоемкость алгоритма зависит от формы записи матрицы  $\mathbf{X}$ , которая определяется выбором последовательности ее строк и столбцов (всего  $|I|!|J|!$  вариантов форм записи). Для этого удобнее представить число  $N$  в виде произведения чисел  $n_i$ , представляющих собой варианты последовательностей ненулевых элементов  $i$ -й строки,  $i = 1, 2, \dots, |I|$ , которые вычисляются по-разному в зависимости от того, независим ли старший ненулевой элемент строки или  $j$ -эквивалентен по отношению к ненулевым элементам предшествующих строк.

В первом случае в соответствии с утверждением 1.12 число  $n_i$  вариантов старшего ненулевого элемента  $i$ -й строки вычисляется так:

$$n'_{i1} = q_j \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right], \quad (2.4)$$

где  $\lambda'_k$  – число не старших элементов  $i$ -й строки, которые  $i$ -эквивалентны старшему элементу этой строки и не являются  $j$ -эквивалентными другим предшествующим элементам матрицы. В этом случае общее число последовательностей вариантов элементов  $i$ -й строки в соответствии с утверждением 1.12 равно

$$n'_i = q_j \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right] (\lambda'_i)!, \quad (2.5)$$

Для получения аналогичной оценки  $n''_{i1}$  во втором случае введем некоторые определения и обозначения.

Назовем две строки  $j$ -эквивалентными, если их первые ненулевые элементы  $j$ -эквивалентны между собой. Эта эквивалентность разбивает на классы эквивалентности множество строк матрицы  $\mathbf{X}$ , за исключением первой, старший элемент которой обязательно независим. Упорядоченность классов, как и упорядоченность строк, индуцируется лексикографической упорядоченностью элементов матрицы. Обозначим буквой  $K$  количество классов, а символом  $L_k$  – максимальное количество строк в  $k$ -м классе,  $k = 1, 2, \dots, K$ . Пусть  $i$ -я строка принадлежит  $k$ -му классу. Введем обозначение  $l_{ik}$  для количества строк в этом классе с  $i' \geq i$ ,  $l_{ik} \in \{1, 2, \dots, L_k\}$ .

Тогда в соответствии с утверждением 1.12 для  $i \neq 1$  имеем

$$n''_{i1} = l_{ik}.$$

В этом случае общее число  $n''_i$  вариантов  $i$ -й строки,  $i = 2, 3, \dots, |I|$ , вычисляем так

$$n''_i = l_{ik} \lambda''_i!, \quad (2.6)$$

где  $\lambda''_i$  определяется как  $\lambda'_i$  в (2.4). Значения  $\lambda'_i$  и  $\lambda''_i$  соответственно в первом и во втором из рассматриваемых случаев при одинаковых значениях  $q_i$  отличаются на единицу

$$\lambda'_i = \lambda''_i - 1, \quad (2.7)$$

если  $i \in \{2, 3, \dots, |I|\}$ , так как первый случай от второго отличается наличием одного независимого элемента в строке. Только  $\lambda'_1 = \lambda''_1$ .

Сопоставим  $n'_i$  и  $n''_i$  при одинаковых значениях  $q_i$  и одинаковом числе полностью зависимых элементов в  $i$ -й строке, т.е. не влияющих на исковое число вариантов:

$$\frac{n'_i}{n''_i} = \frac{q_j \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right] \lambda'_i!}{l_{ik} \lambda''_i!} = \frac{q_j \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right]}{l_{ik} (1 + \lambda'_i)}. \quad (2.8)$$

Первая дробь в правой части (2.8) всегда больше единицы, так как  $l_{ik} < q_i$ . Вторая дробь принимает максимальное значение, равное единице, только для одной строки, где выполняется равенство  $|J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) = (1 + \lambda'_i)$ . Поэтому  $n'_i / n''_i > 1$ . Отсюда можно сделать вывод, что следует отдавать предпочтение записи матрицы  $\mathbf{X}$ , в которой минимизировано число строк, начинающихся с независимого ненулевого элемента. Наилучшим образом этому требованию удовлетворяет каноническая форма записи матрицы  $\mathbf{X}$ , последовательность  $\mathbf{W}$  которой является самой старшей по сравнению с последовательностями  $\mathbf{W}$  других записей матрицы  $\mathbf{X}$ . В этом случае только число вариантов последовательностей ненулевых элементов первой строки определяется по (2.5), а остальных строк – по (2.6). Тогда общее число  $N''$  просматриваемых вариантов матрицы  $\mathbf{X}$  с учетом (2.7) равно

$$N'' = |J| \cdot q_j \cdot \lambda'_1! \cdot \prod_{k=1}^K L_k! \cdot \prod_{i=2}^{|I|} \lambda''_i! = |I| \cdot q_i! \cdot \prod_{i=2}^{|I|} \lambda''_i! \cdot \prod_{k=1}^K L_k!. \quad (2.9)$$

Сравним полученную оценку (2.9) с оценкой известного метода выявления автоморфизмов  $N_0 = |I|! |J|!$ :

$$\frac{N''}{N_0} = \frac{|I| \cdot q_i! \cdot \prod_{i=2}^{|I|} \lambda''_i! \cdot \prod_{k=1}^K L_k!}{|I|! |J|!} = \frac{q_i! \cdot \prod_{i=2}^{|I|} \lambda''_i! \cdot \prod_{k=1}^K L_k!}{|J|! (|I|-1)!}. \quad (2.10)$$



Поскольку  $q_i + \sum_{i=2}^{|I|} \lambda_i'' = |J|$ ,  $\sum_{k=1}^K L_k = |I| - 1$ , из (2.10) следует, что эффект от применения алгоритма 1.3 не меньше, чем от разбиения множеств  $I$  и  $J$  на подмножества.

Покажем, что предложенный алгоритм эффективнее известного и в худшем случае, т.е. когда число строк матрицы  $\mathbf{X}$ , начинающихся независимыми элементами  $x_{ij} \in X_1$ , максимально и при этом все ее  $j$ -эквивалентные строки принадлежат одному классу эквивалентности. В этом случае для  $L_k = L_1 = q_j - 1$   $j$ -эквивалентных строк применимо соотношение (2.6), а для остальных  $t = |I| - L_1 = |I| - q_j + 1$  строк – соотношение (2.5). Общее число просматриваемых множеств в этом случае

$$N' = \left[ (q_j - 1)! \prod_{i=t}^{|I|} \lambda_i'' \right] \cdot \left\{ q_j^t \prod_{i=1}^t \left[ \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right] \lambda_i' \right] \right\}. \quad (2.11)$$

Нетрудно убедиться, что в строках с номерами  $|I| - q_i < i \leq |I|$ , т.е. относящихся к  $j$ -эквивалентным, любой элемент  $x_{ij} \in X_1$   $j$ -эквивалентен одному из предшествующих элементов множества  $X_1$ . Следовательно, для этих строк  $\lambda_i'' = 0$ . Поэтому (2.11) можно представить как

$$N' = q_j^{(t-1)} \cdot q_j \cdot \prod_{i=1}^t \left\{ \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right] \lambda_i' \right\}.$$

Сопоставим полученный результат с трудоемкостью известного алгоритма:

$$\frac{N'}{N_0} = \frac{q_j^{(|I|-q_j)}}{|I|(|I|-1)(|I|-2)\mathbf{L}(q_j+1)} \cdot \frac{\prod_{i=1}^t \left\{ \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right] \lambda_i' \right\}}{|J|!}. \quad (2.12)$$

Первая дробь в правой части (2.12) всегда меньше единицы, так как числа сомножителей ее числителя и знаменателя совпадают, но при этом каждый сомножитель знаменателя больше любого сомножителя числителя (2.12). Для анализа второй дроби (2.12) удобно сократить ее числитель и

знаменатель на величину  $\prod_{i=1}^t \left[ |J| - \sum_{k=1}^{i-1} (1 + \lambda'_k) \right]$ . Тогда в числителе останется

$\prod_{i=1}^t \lambda'_i !$ , а в знаменателе –  $|J| - t = \sum_{i=1}^t \lambda'_i$  сомножителей, произведение  $P$  ко-

торых больше величины  $(|J| - t)!$  Так как  $(\sum_{i=1}^t \lambda'_i) ! > \prod_{i=1}^t (\lambda'_i) !$ , то тем более

$P > \prod_{i=1}^t (\lambda'_i) !$ . Следовательно, вторая дробь также меньше единицы. Отсюда

следует вывод о более высокой эффективности алгоритма 1.3 и в худшем случае.

**Пример 2.7.** Приведем оценки трудоемкости для примера 2.6. В нем матрица  $X$  представлена в канонической форме, поэтому для определения числа  $N''$  просматриваемых вариантов  $W_{\mu}$  используется (2.9). Оно равно 72. В худшем случае, который получаем, если поменять последовательность столбцов заданной матрицы  $X$  на противоположную, т.е. перейти к матрице  $X'$ , симметричной заданной относительно вертикальной оси, число просматриваемых вариантов следует определять по (2.11). Оно равно 11 664. Для сравнения приведем аналогичную оценку известного метода:  $N_0 = |I|! |J|! = 518\,400$ .

## 2.4. Конструктивное перечисление неизоморфных графов

### 2.4.1. Постановка задачи

Граф – особенно важный для приложений комбинаторный объект. Благодаря наглядности графового представления структур различных материальных систем и доступности языка теории графов они, применяются во многих областях техники: в электротехнике, технике связи, машиностроении, архитектуре, при проектировании вычислительных машин и других устройств дискретного действия, автоматических систем управления и т.д. Понятие "граф" широко используется при создании математических моделей в физике, химии, экономике, генетике, психологии, социологии, антропологии, лингвистике. В математике графы часто используются для наглядного представления различных абстрактных систем, описываемых бинарными отношениями.

Конструктивному перечислению попарно неизоморфных графов с использованием вычислительной техники принадлежит важная роль в развитии как самой теории графов, так и ее приложений, поскольку дает обшир-

ный фактический материал для проверки гипотез и решения не решенных аналитическими методами задач. Далекое не всегда применительно к математике и приложениям интересуются графами общего вида с заданными числами вершин и ребер. Чаще для представления структурных отношений применяются графы более узких классов. Однако эффективность конструктивного перечисления графов специального вида в большинстве случаев зависит от эффективности методов порождения графов общего вида.

Среди известных методов конструктивного перечисления неизоморфных графов наиболее эффективны те, в основе которых лежит каноническое упорядочение графов, облегчающее выделение единственного представителя из множества изоморфных графов ([18] и др.). Каноническим представителем класса эквивалентности-изоморфизма объявляется в этом случае экстремальный элемент множества, т.е. такой, который невозможно увеличить (уменьшить) никакой перенумерацией вершин графа. Такой подход, как уже говорилось, не требует запоминания и хранения уже построенных графов с целью сопоставления с ними каждого нового: введенный признак каноничности позволяет для каждого графа, рассматриваемого в отдельности, определить, является ли он каноническим представителем своего класса эквивалентности, и таким образом решить, включать ли его в список неизоморфных искомого графов.

Однако процедура порождения неизоморфных графов остается по-прежнему достаточно трудоемкой. Ей характерны тупики, заключающиеся в том, что частично построенный граф невозможно достроить до канонического варианта. В этом случае приходится оставлять недостроенный граф и переходить к построению нового варианта. Поэтому число шагов перебора может существенно превосходить число искомого неизоморфных графов. Следовательно, решаемая здесь задача сокращения переборных элементов в процедуре порождения графов по-прежнему остается остро актуальной.

#### 2.4.2. Каноническое упорядочение множества $(p, q)$ -графов

Граф с  $p$  вершинами и  $q$  ребрами ( $(p, q)$ -граф) может быть представлен матрицей инцидентностей  $\mathbf{X} = \|x_{ij}\|_{q \times p}$ ,  $x_{ij} \in \{0, 1\}$ , в каждой строке которой два элемента являются единичными, остальные – нулевыми. Число таких матриц  $[C(p, 2)]^q$ . Задача порождения  $(p, q)$ -графов, как и задача порождения систем инцидентностей, имеет второй порядок сложности. Оче-

видно, что при таком представлении  $(p, q)$ -графов для решения этой задачи может быть применен алгоритм 2.1.

Вместо матрицы инцидентностей при решении задачи порождения  $(p, q)$ -графов принято применять матрицу смежности ([18] и др.). Тогда эту задачу можно трактовать как задачу перебора вариантов размещения в наддиагональной (поддиагональной) части матрицы смежности  $q$  единиц всеми возможными способами. Число таких вариантов равно  $C[p(p-1)/2, q]$ . При данном представлении задача порождения  $(p, q)$ -графов по-прежнему остается задачей второго порядка сложности. Однако нетрудно убедиться, что трудоемкость ее решения уменьшается. Действительно, отношение числа просматриваемых вариантов матриц смежности  $\mathbf{Y}$  к числу просматриваемых вариантов матриц  $\mathbf{X}$  при одинаковых значениях  $p$  и  $q$

$$\begin{aligned} & C[p(p-1)/2, q]/[C(p, 2)]^q = \\ &= \frac{1}{q!} \left( \frac{2}{p(p-1)} \right)^q \frac{p(p-1)}{2} \left( \frac{p(p-1)}{2} - 1 \right) \left( \frac{p(p-1)}{2} - 2 \right) \mathbf{L} \left( \frac{p(p-1)}{2} - q + 1 \right) = \\ &= \frac{1}{q!} \left( \frac{2}{p(p-1)} \right)^{(q-1)} \left( \frac{p(p-1)}{2} - 1 \right) \left( \frac{p(p-1)}{2} - 2 \right) \mathbf{L} \left( \frac{p(p-1)}{2} - q + 1 \right), \quad (2.13) \end{aligned}$$

как видно из (2.13), существенно меньше единицы. Поэтому остановимся также на матрице смежности для представления  $(p, q)$ -графа.

Выбрав матрицу смежности, применим к задаче порождения  $(p, q)$ -графов подход, рассмотренный в гл.1. Пусть  $M$  – множество из  $C[p(p-1)/2, q]$  графов, построенных путем перебора вариантов матрицы смежности с  $q$  ненулевыми элементами в ее наддиагональной части. Начнем с упорядочения элементов множества  $M$ , облегчающего отбор по одному представителю от каждого класса эквивалентности (изоморфизма). Из двух известных [18; 19] способов канонического упорядочения графов выберем второй как более эффективный для графов общего вида.

Пусть  $d_i$  – степень  $i$ -й вершины  $v_i$  графа  $\Gamma$ ,  $i=1, 2, \dots, p$ . Множество  $V = (v_1, v_2, \dots, v_p)$  вершин графа разбивается на подмножества  $V_k$  вершин с одинаковыми степенями  $d_k, k \in \{1, 2, \dots, K\}$ , исключение составляют только однородные графы. Этим разбиением индуцируется разбиение множества  $I$  строк ( $J$  столбцов) матрицы  $\mathbf{Y}$  на классы эквивалентности  $I_k (J_k), k \in \{1,$

2, ..., K}: каждая строка с  $i \in I_k$  (каждый столбец с  $j \in J_k$ ) содержит  $d_k$  ненулевых (единичных) элементов.

Рассмотрим подмножество  $M'$  множества  $M$  графов, вершины которых упорядочены по невозрастанию их степеней:

$$v_1 \mathbf{f} v_2 \Leftrightarrow d_1 \succ d_2,$$

т.е. вершину  $v_1$  будем считать старше вершины  $v_2$ , если степень первой больше степени второй.

**Утверждение 2.4.** *Множество  $M'$  ( $p, q$ )-графов с вершинами, упорядоченными по невозрастанию их степеней, содержит, по крайней мере, по одному представителю от каждого класса изоморфных графов.*

Справедливость этого утверждения следует из того, что каждый класс изоморфных ( $p, q$ )-графов состоит из графов с вершинами, пронумерованными всеми  $p!$  возможными способами.

Из утверждения 2.4 следует, что при поиске множества всех неизоморфных ( $p, q$ )-графов достаточно ограничиться только рассмотрением элементов множества  $M'$ . Пусть  $M'_Y$  – множество матриц смежности, представляющих эти графы.

Матрице смежности графа, учитывая ее симметричность относительно главной диагонали, поставим в соответствие  $(p(p-1)/2)$ -мерный булев вектор,

$$\mathbf{y} = (y_{12}, y_{13}, \mathbf{K}, y_{1p}, y_{23}, y_{24}, \mathbf{K}, y_{2p}, \mathbf{K}, y_{(p-1)p}),$$

построенный из наддиагональных элементов матрицы. Очевидно, что отображение  $\mathbf{Y} \rightarrow \mathbf{y}$  взаимно однозначно.

Упорядочим линейно множество матриц  $\mathbf{Y} \in M'_Y$ , а значит, и графов  $\Gamma \in M'$ , полагая для любых  $\mathbf{Y}_1, \mathbf{Y}_2$

$$\mathbf{Y}_1 \mathbf{f} \mathbf{Y}_2 \Leftrightarrow \mathbf{y}_1 \overset{lex}{\succ} \mathbf{y}_2,$$

где векторы  $\mathbf{y}_1, \mathbf{y}_2$  упорядочены в алфавите  $(1, 0)$ .

Объявим каноническим элемент  $\mathbf{Y} \in M'_Y$  ( $\Gamma \in M'$ ), вектор  $\mathbf{y}$  которого нельзя увеличить никакими подстановками множества  $I$  номеров строк или ( $\mathbf{y}$ ) множества  $J$  номеров столбцов, сохраняющими разбиение этого множества на подмножества  $I_k (J_k), k \in \{1, 2, \dots, K\}$ .

Пусть  $S_k$  – симметрическая группа подстановок множества  $I_k (J_k)$ . Обозначим буквой  $\mathbf{G}$  матричную подстановку, соответствующую подста-

новке  $g \in S_p = \bigotimes_{k=1}^K S_k$ . Предлагаемое определение каноничности позволяет ввести предикат

$$P(\mathbf{Y}) \equiv (\forall g \in S_p \quad \mathbf{G}^{-1} \mathbf{Y} \mathbf{G} \mathbf{p} \mathbf{Y} \vee \mathbf{G}^{-1} \mathbf{Y} \mathbf{G} = \mathbf{Y}), \quad (2.14)$$

принимающий истинное значение только для канонических матриц  $\mathbf{Y} \in M_{\mathbf{Y}}$ .

Для однородных графов, у которых  $d_1 = d_2 = \mathbf{L} = d_p$ , данное в [19] определение каноничности  $\mathbf{Y}$  совпадает с определением из работы [18]. Это означает, что для однородных графов выбранное определение каноничности не дает никаких преимуществ. Для всех остальных графов оно предпочтительнее. Оно позволяет сократить, во-первых, число просматриваемых вариантов матриц  $\mathbf{Y}$  ( $M_{\mathbf{Y}}$  вместо  $M_{\mathbf{Y}}$ ,  $M_{\mathbf{Y}} \subseteq M_{\mathbf{Y}}$ ) и, во-вторых, число проверочных подстановок, устанавливающих их каноничность (это подстановки из группы  $S_p$  с порядком  $\prod_{k=1}^K |I_k|!$  вместо симметрической группы

подстановок множества  $I = \bigcup_{k=1}^K I_k$  с порядком  $p! = |I|!$ ).

Пусть в матрице смежности определены элементы ее  $i$  первых строк (столбцов). Такая матрица называется [18] частично заполненной. Обозначим ее  $\mathbf{Y}_i$ . Если у матриц  $\mathbf{Y}$ ,  $\mathbf{Y}_i$  совпадают первые  $i$  строк и столбцов, матрица  $\mathbf{Y}$  называется [18] пополнением матрицы  $\mathbf{Y}_i$ . Очевидно, что одной матрице  $\mathbf{Y}_i$  соответствует в общем случае не единственное пополнение  $\mathbf{Y}$ .

Пусть множество  $S_i$  составляют те элементы множества  $S_p = \bigotimes_{k=1}^K S_k$ , которые стабилизируют подмножество из первых  $i$  элементов множества  $I$ . Доказано утверждение [18], из которого следует, что матрица  $\mathbf{Y}$ , которая является пополнением матрицы  $\mathbf{Y}_i$ , не может быть канонической, если для матрицы  $\mathbf{Y}_i$  принимает истинное значение предикат

$$P(\mathbf{Y}_i) \equiv (\exists g \in S_i : \mathbf{G}^{-1} \mathbf{Y}_i \mathbf{G} \mathbf{f} \mathbf{Y}_i). \quad (2.15)$$

Хотя выбранный нами способ канонического упорядочения множества графов отличается от предложенного в работе [18], а множество  $S_i$  в (2.15) составляет лишь часть множества проверочных подстановок, применяемых в этой работе, в нашем случае указанное утверждение также справедливо.

Доказательство его аналогично приведенному в [18], поэтому здесь оно опускается.

Частично заполненную матрицу  $\mathbf{Y}_i$ , для которой предикат (2.15) принимает ложное значение, логично также называть канонической. Каждый раз после выбора варианта заполнения очередной,  $i$ -й строки матрицы смежности необходимо согласно [18] выполнять проверку каноничности частично заполненной матрицы  $\mathbf{Y}_i$ ,  $i \in \{1, 2, \dots, p\}$ , по предикату (2.15). При этом отсекается наибольшее количество изоморфных графов в процессе их построения. Более высокая эффективность предиката каноничности (2.15) по сравнению с (2.14) объясняется тем, что он позволяет отсекал неканонические матрицы  $\mathbf{Y}_i$ , не достраивая их до конца, вместе с семейством матриц  $\mathbf{Y}$ , которые являются ее пополнениями. Обозначим  $\mathbf{y}_i$  вектор из элементов  $i$ -й строки матрицы  $\mathbf{Y}$ .

**У т в е р ж д е н и е 2.5.** *Если  $\mathbf{Y}$  – каноническая матрица, то для любых  $i', i'' \in I_k$*

$$i' \succ^{lex} i'' \Leftrightarrow \mathbf{y}_{i'} \geq \mathbf{y}_{i''} .$$

**Д о к а з а т е л ь с т в о.** Доказываем от противного. Допустим, что матрица  $\mathbf{Y}$  каноническая, но в ней  $\mathbf{y}_{i'} \prec^{lex} \mathbf{y}_{i''}$  при  $i' \succ^{lex} i''$ . Пусть  $n_i$  – порядковый номер единичной координаты вектора  $\mathbf{y}_i$ ,  $n_i = 1, 2, \dots, d_i$ . Нетрудно убедиться, что в векторе  $\mathbf{y}$   $n_i$ -я единичная координата вектора  $\mathbf{y}_{i''}$  встречается не раньше  $n_i$ -й единичной координаты вектора  $\mathbf{y}_{i'}$ . Поэтому перестановкой строк с номерами  $i', i''$  с одновременной перестановкой  $i'$ -го и  $i''$ -го столбцов можно только лексикографически увеличить вектор  $\mathbf{y}$ , что противоречит предположению о каноничности матрицы  $\mathbf{Y}$ .

Из утверждения 2.5 следует, что в канонической матрице строки (столбцы), принадлежащие подмножеству  $I_k(J_k)$ , должны быть лексикографически упорядочены как  $(p-1)$ -мерные векторы (диагональные элементы матрицы  $\mathbf{Y}$  пропускаются). Из этого необходимого условия каноничности матрицы  $\mathbf{Y}$  вытекает, что при выбранном определении каноничности в процессе порождения матриц  $\mathbf{Y}$  возможно такое сокращение перебора: необходимо просматривать только варианты матрицы  $\mathbf{Y}$  с лексикографически упорядоченными в пределах каждого множества  $I_k(J_k)$  строками (столбцами). Тогда при проверке каноничности  $\mathbf{Y}$  по предикату (2.15) можно опустить подстановки-транспозиции.

Так как мы условились рассматривать только графы с вершинами, упорядоченными по их степеням, на элементы искомой матрицы смежности дополнительно накладываются следующие ограничения:

$$\sum_{j=1}^p y_{ij} = d_i, \quad i = 1, 2, \dots, p. \quad (2.16)$$

Варианты матрицы смежности, для которых выполняется ограничение (2.16), будем считать допустимыми. Условия (2.16) в некоторых случаях позволяют построить пополнение матрицы  $\mathbf{Y}_i$  (или отказаться от пополнения) без перебора, который заменяется несложными вычислениями. Этот прием называется форсированием [18]. Рассмотрим ситуации, когда возможно форсирование при выбранном определении каноничности.

Пусть  $\mathbf{y}'_i(\mathbf{y}''_i)$  – поддиагональная (наддиагональная) часть вектора  $\mathbf{y}_i$ ,  $d'_i(d''_i)$  – число ненулевых элементов в  $\mathbf{y}'_i(\mathbf{y}''_i)$ ,  $m'_i(m''_i)$  – число координат вектора  $\mathbf{y}'_i(\mathbf{y}''_i)$ . Очевидно, что

$$d'_i = \sum_{j=1}^{i-1} y_{ij}; \quad (2.17)$$

$$d''_i = \sum_{j=i+1}^p y_{ij}. \quad (2.18)$$

Из (2.16) – (2.18) следует, что матрица  $\mathbf{Y}_i$  является недопустимой в ситуациях, когда

$$d''_{i'} = d_{i'} - d'_{i'} < 0, \quad i' \in \{i+1, i+2, \dots, p-1\}; \quad (2.19)$$

$$d''_{i'} = d_{i'} - d'_{i'} > m''_{i'}, \quad i' = i+1; \quad (2.20)$$

$$d'_{i'} > d_{i'}, \quad i' = p. \quad (2.21)$$

Из (2.16) – (2.18) также видно, что все координаты  $\mathbf{y}''_i$  могут быть только нулевыми, если

$$d'_{i'} = d_{i'}, \quad i' \in \{i+1, i+2, \dots, p-1\}; \quad (2.22)$$

или единичными, если

$$d_{i'} - d'_{i'} = m''_{i'}, \quad i' = i+1. \quad (2.23)$$



### 2.4.3. Применение информации о симметриях графа

Пусть  $\mathbf{Y}_i$  – частично заполненная каноническая матрица смежности. Требуется найти для нее канонические пополнения  $\mathbf{Y}_{(i+1)}$ , т.е. перебрать все такие варианты заполнения  $(i+1)$ -й строки в соответствии с заданным разбиением степеней вершин искомого графа, при котором матрица  $\mathbf{Y}_{(i+1)}$  также была бы канонической. Пусть  $d_{(i+1)}$  – требуемая степень вершины  $v_{(i+1)}$  искомого графа. Так как поддиагональная часть  $(i+1)$ -й строки заполняется по мере заполнения первых  $i$  строк матрицы  $\mathbf{Y}_i$  (по симметрии относительно главной диагонали), задача сводится к перебору вариантов размещения

$$d''_{(i+1)} = d_{(i+1)} - d'_{(i+1)} \quad (2.24)$$

единиц в наддиагональной части  $(i+1)$ -й строки, при которых матрица  $\mathbf{Y}_{(i+1)}$  была бы канонической.

Пусть  $\text{Aut}(\Gamma_i)$  – группа автоморфизмов графа  $\Gamma_i$ , описываемого матрицей  $\mathbf{Y}_i$ , а  $\mathbf{Y}_{(i+1)}^*$  – такая матрица смежности, что все элементы, кроме, может быть, элементов  $(i+1)$ -й строки и  $(i+1)$ -го столбца, равны нулю. Искомую матрицу  $\mathbf{Y}_{(i+1)}$  представим в виде суммы  $\mathbf{Y}_i$  и  $\mathbf{Y}_{(i+1)}^*$ :

$$\mathbf{Y}_{(i+1)} = \mathbf{Y}_i + \mathbf{Y}_{(i+1)}^*.$$

**У т в е р ж д е н и е 2. 6.** *Для того чтобы матрица  $\mathbf{Y}_{(i+1)}$  была канонической, необходимо, чтобы предикат*

$$P(\mathbf{Y}_{(i+1)}^*) \equiv (\exists g \in \text{Aut}(\Gamma_i) \quad \mathbf{G}^{-1} \mathbf{Y}_{(i+1)}^* \mathbf{G} \mathbf{f} \mathbf{Y}_{(i+1)}^* \vee \mathbf{G}^{-1} \mathbf{Y}_{(i+1)}^* \mathbf{G} = \mathbf{Y}_{(i+1)}^*). \quad (2.25)$$

*принимал ложное значение.*

Справедливость утверждения следует из того, что подстановки  $g \in \text{Aut}(\Gamma_i)$  не меняют первых  $i$  строк матрицы  $\mathbf{Y}_i$  и элементов заполненных столбцов ее  $(i+1)$ -й строки.

Рассмотрим  $(i+1)$ -ю строку как булев вектор  $\mathbf{y}_{(i+1)}$ , в котором  $(i+1)$  первых координат определены. Теперь задача построения канонических вариантов матрицы  $\mathbf{Y}_{(i+1)}$  сведена к задаче порождения  $(p-i-1)$ -мерных булевых векторов  $\mathbf{y}_{(i+1)}''$  с  $d_{(i+1)}''$  ненулевыми элементами в условиях, когда на множестве номеров его элементов задана группа автоморфизмов.

На множестве номеров  $j$  наддиагональных элементов  $(i+1)$ -й строки этой матрицы рассмотрим отношение эквивалентности, полагая

$$\forall j, j' \in \{i+2, i+3, \mathbf{K}, p\} \quad j \sim j' \Leftrightarrow \exists g \in \text{Aut}(\Gamma_i): jg = j'.$$

Этой эквивалентностью множество номеров  $j$  наддиагональных элементов  $y_{ij}$  матрицы  $\mathbf{Y}_i$  разбивается на классы эквивалентности ( $h$ -классы), частным случаем которых являются «куски постоянства» из [18] и множества  $k$ -эквивалентных номеров строк (столбцов) из [20]. Вершины  $v_j, v_{j'}$  графа  $\Gamma_i$ , соответствующие номерам  $j, j'$  столбцов матрицы  $\mathbf{Y}_i$ , назовем эквивалентными по  $\text{Aut}(\Gamma_i)$  или  $h$ -эквивалентными.

Из определения матрицы  $\mathbf{Y}_{(i+1)}^*$  следует, что каноничность вектора  $\mathbf{y}_{(i+1)}''$  является необходимым и достаточным условием ее каноничности, т.е. предикат (2.25) можно заменить предикатом

$$P(\mathbf{y}_{(i+1)}'') \equiv (\forall g \in \text{Aut}(\mathbf{Y}_i) \quad \mathbf{y}_{(i+1)}'' \stackrel{lex}{\geq} \mathbf{y}_{(i+1)}''), \quad (2.26)$$

принимающим истинное значение только для канонического вектора  $\mathbf{y}_{(i+1)}''$ .

Для сокращения числа просматриваемых вариантов векторов  $\mathbf{y}_{(i+1)}''$  можно использовать признаки каноничности из п. 2.2.3. Покажем на примерах, что число просматриваемых матриц  $\mathbf{Y}_{(i+1)}$  при таком использовании информации о симметриях сокращается.

**Пример 2.8.** Задана матрица

$$\mathbf{Y}_4 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Требуется выделить в векторе  $\mathbf{y}_5$   $h$ -классы и перечислить варианты его заполнения при  $d_5'' = 3$ , обеспечивающие каноничность матрицы  $\mathbf{Y}_5$ .

С учетом группы автоморфизмов  $\text{Aut}(\Gamma_i) = \text{Aut}(\Gamma_4)$  в векторе  $\mathbf{y}_5$  только два  $h$ -класса:  $J_1 = (6, 7, 8)$  и  $J_2 = (9, 10, 11, 12)$ . На элементах каждого действует симметрическая группа автоморфизмов. Поэтому в соответствии с признаками каноничности из пп. 2.2.2, 2.2.3 при  $d_5'' = 3$  достаточно рассмотреть вместо восьми вариан-

тов вектора  $\mathbf{y}_5''$  по [18] всего четыре варианта:  $(1, 1, 1, 0, 0, 0, 0)$ ;  $(1, 1, 0, 1, 0, 0, 0)$ ;  $(1, 0, 0, 1, 1, 0, 0)$ ;  $(0, 0, 0, 1, 1, 1, 0)$ .

**Пример 2.9.** Задана матрица

$$\mathbf{Y}_4 = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

Требуется, как и в предыдущем примере, выделить в векторе  $\mathbf{y}_5''$   $h$ -классы и перечислить варианты его заполнения при  $d_5'' = 2$ .

С учетом группы автоморфизмов  $\text{Aut}(\Gamma_4)$   $h$ -классы вектора  $\mathbf{y}_5''$  – это  $J_1 = (6)$ ,  $J_2 = (7, 8)$ ,  $J_3 = (9, 10)$ . Два последних  $h$ -класса  $H$ -эквивалентны. Поэтому в соответствии с утверждением 2.1 достаточно рассмотреть вместо пяти вариантов вектора  $\mathbf{y}_5''$  по [18] три варианта:  $(1, 1, 0, 0, 0)$ ;  $(0, 1, 1, 0, 0)$ ;  $(0, 1, 0, 1, 0)$ .

**Пример 2.10.** Пусть в предыдущем примере при заполнении 5-й строки выбран последний вариант. Тогда  $h$ -классы вектора  $\mathbf{y}_6''$  – это  $J_1 = (7, 9)$  и  $J_2 = (8, 10)$ . Поэтому при  $d_6'' = 2$ , как следует из утверждения 2.1, достаточно рассмотреть вместо шести вариантов вектора  $\mathbf{y}_6''$  по [18] три варианта:  $(1, 1, 0, 0)$ ;  $(1, 0, 1, 0)$ ;  $(1, 0, 0, 1)$ .

Пусть  $S$  – симметрическая группа подстановок множества номеров  $V$  вершин графа  $\Gamma$  ( $I$  строк,  $J$  столбцов матрицы  $\mathbf{Y}$ ). Назовем подстановки  $g_v, g_w \in S$  и соответствующие им подстановочные матрицы  $\mathbf{G}_v, \mathbf{G}_w$  эквивалентными, если они одинаково действуют на матрицу смежности:

$$g_v \sim g_w \Leftrightarrow \mathbf{G}_v^{-1} \mathbf{Y} \mathbf{G}_v = \mathbf{G}_w^{-1} \mathbf{Y} \mathbf{G}_w. \quad (2.27)$$

**У т в е р ж д е н и е 2.7.** Для того чтобы подстановки  $g_v, g_w \in S$  были эквивалентными, достаточно, чтобы группа автоморфизмов графа  $\Gamma$  содержала элемент  $h$  такой, что

$$g_v = g_w h.$$

**Доказательство.**

$$\begin{aligned} \mathbf{G}_v^{-1} \mathbf{Y} \mathbf{G}_v = \mathbf{G}_w^{-1} \mathbf{Y} \mathbf{G}_w &\Leftrightarrow \mathbf{G}_w \mathbf{G}_v^{-1} \mathbf{Y} \mathbf{G}_v \mathbf{G}_w^{-1} = \mathbf{Y} \Leftrightarrow \mathbf{G}_w \mathbf{G}_v^{-1} \mathbf{Y} (\mathbf{G}_w \mathbf{G}_v^{-1})^{-1} = \mathbf{Y} \Leftrightarrow \\ &\Leftrightarrow \mathbf{G}_w \mathbf{G}_v^{-1} \in \text{Aut}(\Gamma) \Leftrightarrow \exists h \in \text{Aut}(\Gamma): g_v = g_w h. \end{aligned}$$

Отметим, что рассмотренное отношение эквивалентности разбивает группу  $S$  на классы эквивалентных подстановок – левые [21] смежные классы группы  $S$  по ее подгруппе  $\text{Aut}(\Gamma)$ :

$$S = \bigcup_{l=1}^L S_l, \quad S_1 = \text{Aut}(\Gamma).$$

Из определения (2.27) эквивалентных элементов  $S$  следует, что для проверки каноничности матрицы  $\mathbf{Y}$  достаточно из каждого класса  $S_l$ , где  $l = 2, 3, \dots, L$ , использовать только одну подстановку. Это значит, что при наличии сведений о симметриях графа число проверочных подстановок можно было бы сократить до  $(L-1)$ , где  $L$  – число смежных классов по  $\text{Aut}(\Gamma)$ . Оно равно по теореме Лагранжа [21; 22] частному от деления числа элементов в группе  $S$  на число элементов в подгруппе  $\text{Aut}(\Gamma)$ .

Будем характеризовать подстановку

$$g = \begin{pmatrix} 1 & 2 & \dots & p \\ \alpha_1 & \alpha_2 & \dots & \alpha_p \end{pmatrix}$$

вектором  $\mathbf{a} = (\alpha_1, \alpha_2, \dots, \alpha_p)$ , подразумевая под координатой  $\alpha_i \in \{1, 2, \dots, p\}$  порядковый номер вершины графа, либо порядковый номер строки (столбца) матрицы  $\mathbf{Y}$ . Упорядочим подстановки следующим образом:

$$g_v \mathbf{f} g_w \Leftrightarrow \mathbf{a}_v \stackrel{\text{lex}}{\geq} \mathbf{a}_w. \quad (2.28)$$

Очевидно, что элементы любого класса  $S_l$  эквивалентных подстановок также оказываются упорядоченными отношением (2.28). В качестве единственного представителя каждого класса выберем его максимальный элемент. Для него и только для него истинным является предикат

$$P(g) \equiv (\forall h \in \text{Aut}(\Gamma) \quad g \mathbf{f} gh \vee g = gh). \quad (2.29)$$

Максимальный элемент множества  $S_l$  назовем канонической подстановкой. Поскольку между векторами  $\mathbf{a}$  и подстановками  $g \in S$  существует взаимно однозначное соответствие, вектор  $\mathbf{a}$  может быть применен для представления подстановки  $g \in S$  множества  $\{1, 2, \dots, p\}$ . Тогда предикат (2.29) можно заменить предикатом  $P(\mathbf{a})$  из п. 2.2.2, считая  $H = \text{Aut}(\Gamma)$ . Это позволяет свести отбор канонических подстановок к отбору канонических векторов  $\mathbf{a}$  и, следовательно, воспользоваться признаками каноничности, рассмотренными в п. 2.2.2.

**Пример 2.11.** На рис 2.1 изображены два графа с четырьмя вершинами. Симметрическая группа  $S$  содержит 24 элемента. Они могут быть представлены векторами из примера 2.2. Кроме того, группы автоморфизмов из примера 2.2 отражают симметрии графов  $\Gamma_1, \Gamma_2$ :  $\text{Aut}(\Gamma_1) = H_1$ ,  $\text{Aut}(\Gamma_2) = H_2$ . Поэтому перечисленные в

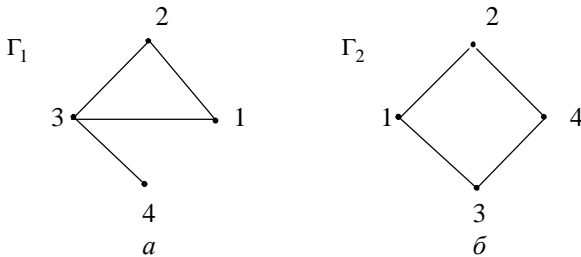


Рис. 2.1

примере 2.2 старшие векторы могут быть использованы для отбора неэквивалентных проверочных подстановок: для графа  $\Gamma_1$  число последних сокращается до 11, а для графа  $\Gamma_2$  – до двух, если не считать тождественную подстановку.

При проверке каноничности графа более характерна ситуация, когда априорная информация о симметриях отсутствует и приобретается в процессе испытания его подстановками. В этом случае число проверочных подстановок не всегда удастся сократить до  $(L-1)$ , так как в зависимости от выбранной последовательности их применения автоморфизм  $h^*$  может встретиться позже, чем подстановка, которая является старшей по  $h^*$ . Тем не менее, и в этом случае указанное использование симметрий графа для повышения эффективности проверки его каноничности полезно. Чтобы показать это, достаточно выбрать любую последовательность проверочных подстановок. Выберем последовательность, согласованную с порядком на множестве постановок, который задан отношением (2.28). Заметим, что этот порядок согласуется с частичным упорядочением проверочных подстановок по включению, рассмотренным в работе [20].

Для этого представим векторами  $\mathbf{a}_j$  множество  $G_j$  подстановок, переводящих последовательность  $(1, 2, \dots, j)$  в последовательность  $(\alpha_1, \alpha_2, \dots, \alpha_j)$ . Упорядоченность векторов  $\mathbf{a}_j$ ,  $j = 1, 2, \dots, p$ , в виде регулярного дерева с корнем  $\mathbf{a}_0$  и концевыми вершинами  $\mathbf{a}_p$  отображает упорядоченность множеств  $G_j$ : два вектора  $\mathbf{a}_{(j+1)} = (\alpha_1, \alpha_2, \mathbf{K}, \alpha_j, \alpha_{(j+1)})$  и

$\mathbf{a}'_{(j+1)} = (\alpha_1, \alpha_2, \mathbf{K}, \alpha_j, \alpha'_{(j+1)})$  с общей корневой вершиной  $\mathbf{a}_j = (\alpha_1, \alpha_2, \mathbf{K}, \alpha_j)$  соответствуют подмножествам  $G_{(j+1)}$ ,  $G'_{(j+1)}$  множества  $G_j$ . Пусть  $\alpha_{(j+1)} < \alpha'_{(j+1)}$ . Тогда при обходе дерева вариантов вектора  $\mathbf{a}$  вершина  $\mathbf{a}_{(j+1)}$  встречается раньше, чем  $\mathbf{a}'_{(j+1)}$ . Следовательно, в процессе проверки каноничности матрицы  $\mathbf{Y}$  множество подстановок  $G_{(j+1)}$  используется раньше, чем множество  $G'_{(j+1)}$ .

Пусть  $H_{\{j\}}$  – подгруппа группы  $S$ , которая состоит из подстановок, сохраняющих нумерацию первых  $j$  строк и столбцов матрицы  $\mathbf{Y}$ . Очевидно, что, действуя подстановкой  $h \in H_{\{j\}}$  на любой элемент  $g \in G_{(j+1)}$ , его можно перевести в некоторый элемент  $g \in G'_{(j+1)}$  и наоборот, любой элемент  $g \in G'_{(j+1)}$  – в некоторый элемент  $g \in G_{(j+1)}$ . Элементы  $g \in G_{(j+1)}$ ,  $g \in G'_{(j+1)}$ , связанные между собой автоморфизмом  $h$ , по утверждению 2.7 эквивалентны по воздействию на матрицу  $\mathbf{Y}$ . Следовательно, множества  $G_{(j+1)}$ ,  $G'_{(j+1)}$  являются равноценными, и между их элементами существует взаимно однозначное соответствие. Тогда при проверке каноничности  $\mathbf{Y}$  достаточно рассмотреть одно из них. Если представить множества  $G_{(j+1)}$ ,  $G'_{(j+1)}$  соответственно векторами  $\mathbf{a}_{(j+1)}$ ,  $\mathbf{a}'_{(j+1)}$ , отсечение множества подстановок  $G'_{(j+1)}$  интерпретируется отсечением  $\mathbf{a}'_{(j+1)}$ .

**Пример 2.12.** В вершинах регулярного дерева на рис. 2.2 подстановки подмножеств  $G_j$  для графа с четырьмя вершинами представлены векторами  $\mathbf{a}_j$ ,  $j = 1, 2, 3, 4$ . В усеченных деревьях на рис. 2.3, 2.4 отмечены подстановки, выявляющие автоморфизмы графов  $\Gamma_1$ ,  $\Gamma_2$  из примера 2.11. Как видно из рис. 2.3, 2.4, число подстановок, необходимых для проверки их каноничности (при последовательности их перебора, отраженной в дереве рис. 2.2), в первом случае сокращается до 11, во втором – до 5. Эти подстановки на рис. 2.3, 2.4 указаны стрелками.

Таким образом, задача отбора проверочных подстановок, представляющих классы эквивалентности  $S_l$ ,  $l = 2, 3, \dots, L$ , сводится к задаче отбора канонических векторов  $\mathbf{a}$ . Это позволяет применить при отборе канонических подстановок признаки каноничности, рассмотренные в п. 2.2.2.

Эквивалентность проверочных подстановок полезно использовать и

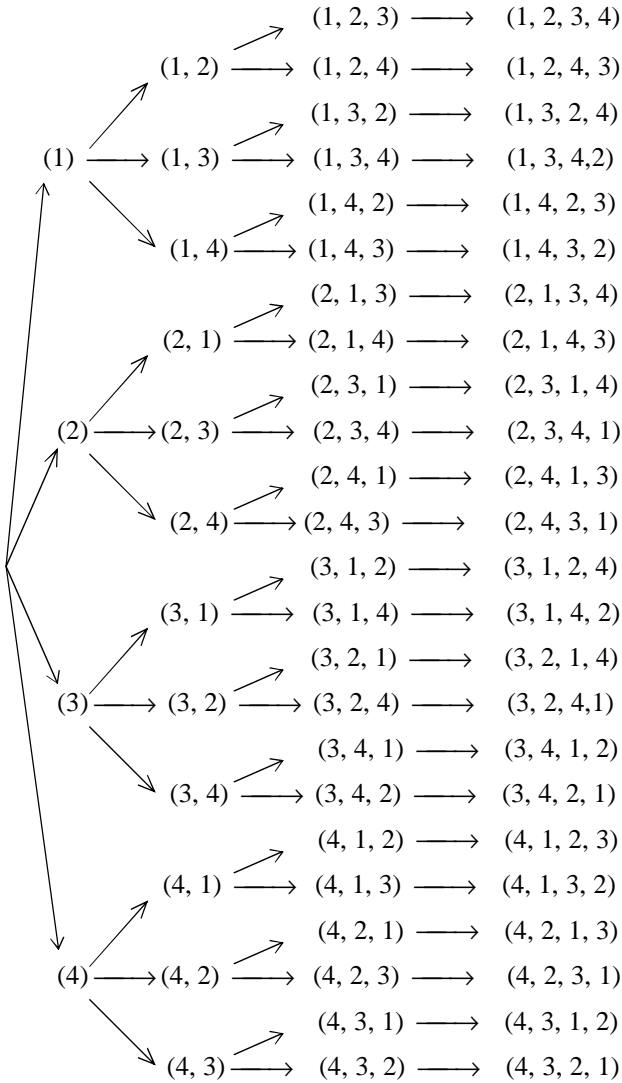


Рис. 2.2

при построении графов. Эффект от такого использования их симметрий усиливается при выбранном определении каноничности графа. Покажем это.

Назовем подстановку  $g \in S_i$ ,  $i = 1, 2, \dots, p$ , увеличивающей (уменьшающей) для матрицы  $\mathbf{Y}_i$ , если  $\mathbf{G}^{-1}\mathbf{Y}_i\mathbf{G} \mathbf{f} \mathbf{Y}_i$  ( $\mathbf{G}^{-1}\mathbf{Y}_i\mathbf{G} \mathbf{p} \mathbf{Y}_i$ ). Проверку ка-

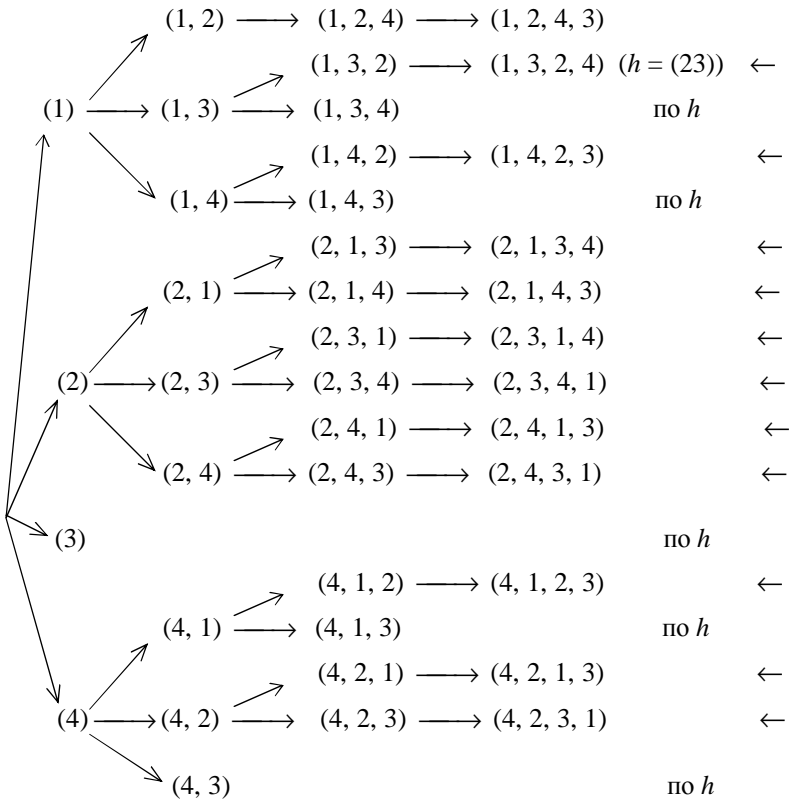


Рис. 2.3

ноничности матрицы  $\mathbf{Y}_i$  по предикату (2.15) можно рассматривать как поиск увеличивающей подстановки: если увеличивающая подстановка найдется, значит, матрица  $\mathbf{Y}_i$  не каноническая.

Пусть  $\mathbf{Y}_i$  – частично заполненная матрица,  $i$  – число строк, заполненных в соответствии с заданным разбиением, причем  $i \in I_{(k+1)}$ ,  $i = i' + i''$ ,  $i' = |V_1| + |V_2| + \mathbf{L} + |V_k|$ ,  $1 < k < K$ ,  $i' < i'' \leq i' + |V_{(k+1)}|$ , и пусть  $\mathbf{Y}_{i'}$  – частично заполненная каноническая матрица.

**У т в е р ж д е н и е 2. 8.** Подстановка  $g$ , увеличивающая матрицу  $\mathbf{Y}_i$ , принадлежит  $\text{Aut}(\Gamma_{i'})$ .

Д о к а з ы в а е м от противного. Если  $g \notin \text{Aut}(\Gamma_{i'})$ , то для матрицы



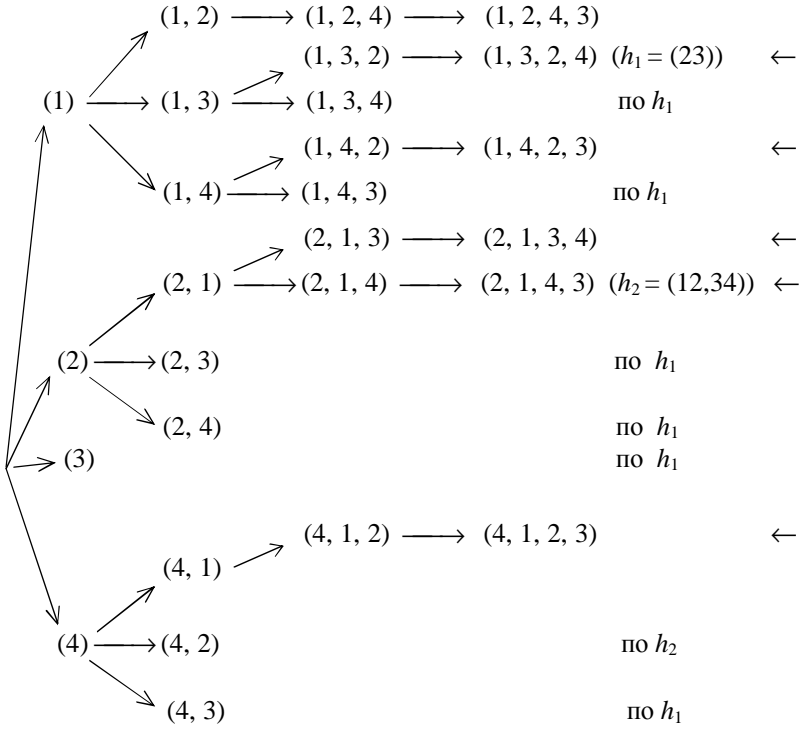


Рис. 2.4

$\mathbf{Y}_{i'}$  – это подстановка либо увеличивающая ( $\mathbf{G}^{-1}\mathbf{Y}_{i'}\mathbf{G} \mathbf{f} \mathbf{Y}_{i'}$ ), либо уменьшающая ( $\mathbf{G}^{-1}\mathbf{Y}_{i'}\mathbf{G} \mathbf{p} \mathbf{Y}_{i'}$ ). Поскольку матрица  $\mathbf{Y}_{i'}$  каноническая, увеличивающей она быть не может. Если подстановка  $g$  является уменьшающей для  $\mathbf{Y}_{i'}$ , то для  $\mathbf{Y}_i$  она тем более будет уменьшающей, так как ненулевые элементы любой строки  $\mathbf{Y}_{i'}$  в векторе  $\mathbf{y}$  занимают старшие места по сравнению с ненулевыми элементами строк с номерами  $i'+1, i'+2, \dots, |I|$  матрицы  $\mathbf{Y}_i$ , имеющими те же порядковые номера.

Обозначим буквой  $\Gamma_1$  граф, соответствующий частично заполненной матрице  $\mathbf{Y}_i$  с  $i = |V_1|$ ,  $\Gamma_2$  – с  $i = |V_1| + |V_2|$  и т.д. Дополним этот список  $(p, 0)$ -графом  $\Gamma_0$  с множеством вершин  $V$ , разбитых на классы эквивалентности  $V_1, V_2, \mathbf{K}, V_K$ . Ему соответствует матрица  $\mathbf{Y}_i = \mathbf{Y}_0$ . Группа автоморфизмов такого графа совпадает с группой  $S_p$ . Тогда из утверждения 2.8 вытекает

С л е д с т в и е 2.8.1.

$$\text{Aut}(\Gamma_0) \supseteq \text{Aut}(\Gamma_1) \supseteq \text{Aut}(\Gamma_2) \supseteq \mathbf{L} \supseteq \text{Aut}(\Gamma_k) \supseteq \mathbf{L} \supseteq \text{Aut}(\Gamma_K).$$

Из утверждения 2.8 и следствия 2.8.1 видно, что при выбранном определении каноничности графа число проверочных подстановок сокращается всякий раз при переходе от  $|V_k|$  к  $|V_{(k+1)}|$ ,  $k = 0, 1, \dots, K-1$ .

Теперь обратимся снова к утверждению 2.7. Понятно, что оно справедливо и для графа  $\Gamma_k$ ,  $k \in \{1, 2, \dots, K\}$ . Следовательно, группа  $\text{Aut}(\Gamma_k)$  также в общем случае распадается на классы эквивалентных подстановок. Тогда для проверки каноничности матрицы  $\mathbf{Y}_i$  с  $i \in I_{(k+1)}$  по предикату (2.15) достаточно ограничиться множеством канонических подстановок, представляющих каждый класс эквивалентности группы  $\text{Aut}(\Gamma_k)$ . Напомним, что из этого множества нужно исключить подстановки-транспозиции, если строки матрицы  $\mathbf{Y}_i$  как векторы упорядочены лексикографически, и тождественную подстановку. Полученное таким образом множество проверочных подстановок обозначим символом  $H_k$ , и предикат (2.15) заменим предикатом

$$P(\mathbf{Y}_i) \equiv (\forall g \in H_k \quad \mathbf{Y}_i \mathbf{f} \mathbf{G}^{-1} \mathbf{Y} \mathbf{G}_i \vee \mathbf{Y}_i = \mathbf{G}^{-1} \mathbf{Y} \mathbf{G}_i), \quad (2.30)$$

принимающим истинное значение только для канонической матрицы  $\mathbf{Y}_i$ .

#### 2.4.4. Алгоритм порождения неизоморфных $(p, q)$ -графов

В соответствии с изложенными соображениями можно представить алгоритм построения всех неизоморфных  $(p, q)$ -графов состоящим из двух частей: порождение монотонно убывающих графических [2] разбиений числа  $2q$  на  $p$  частей; построение всех графов с заданным разбиением. Методы и алгоритмы генерирования упорядоченной последовательности разбиений известны [2; 3]. Известны также методы и алгоритмы отсекающего неграфических разбиений [2]. Поэтому остановимся на алгоритме генерирования  $(p, q)$ -графов с заданным разбиением  $\Pi = (d_1, d_2, \dots, d_p)$ .

Пусть  $\mathbf{Y}_i$  – частично заполненная каноническая матрица, причем  $i \in I_{(k+1)}$ ,  $i = i' + i''$ ,  $i' = |V_1| + |V_2| + \mathbf{L} + |V_k|$ ,  $1 < k < K$ ,  $i' < i'' \leq i' + |V_{(k+1)}|$ . Приведенный далее алгоритм построения канонических матриц  $\mathbf{Y}$ , представляющих неизоморфные графы  $\Gamma$  с заданным разбиением  $\Pi$  множества его вершин, отличается от известных [18; 20] наиболее полным использованием информации о симметриях матрицы  $\mathbf{Y}$  и ее подматриц для отсекающего их изоморфных вариантов. Хотя решение задачи выявления симметрий в

нем не отражено. Кроме требуемого разбиения для работы с алгоритмом необходимо задать  $i' = 0$ ,  $i = 1$ ,  $d_i' = 0$ .

Алгоритм 2.2.

Шаг 1. С учетом группы автоморфизмов  $\text{Aut}(\mathbf{Y}_{(i-1)})$  выделить в наддиагональной части не заполненной пока  $i$ -й строки матрицы  $\mathbf{Y}_i$   $h$ -классы и  $H$ -эквивалентные  $h$ -классы. Вычислить  $d_i''$  по (2.17), (2.24). Положить  $i^* = i$ .

Шаг 2. Рассматривая наддиагональный участок  $i$ -й строки матрицы  $\mathbf{Y}_i$  как булев вектор  $\mathbf{y}_i''$ , определить очередной в лексикографическом порядке вариант размещения в нем  $d_i''$  единиц и  $m_i'' - d_i''$  нулей, удовлетворяющий всем известным признакам каноничности (см. п. 2.2.3). Если варианты исчерпаны, идти к шагу 12.

Шаг 3. Если  $i, (i-1) \in I_k$ , проверить  $i$ -ю строку на лексикографическую упорядоченность по отношению к  $(i-1)$ -й строке согласно утверждению 2.5. При нарушении упорядоченности идти к шагу 2.

Шаг 4. Если при формировании вектора  $\mathbf{y}_i''$  выполнены только необходимые условия каноничности, проверить его по предикату (2.26). Если вектор  $\mathbf{y}_i''$  не канонический, идти к шагу 2.

Шаг 5. Для заполненных строк определить (по симметрии относительно главной диагонали) элементы соответствующих столбцов матрицы  $\mathbf{Y}_i$ .

Шаг 6. Проверить полученную матрицу  $\mathbf{Y}_i$  на допустимость по соотношениям: (2.19) для  $i + 1, i + 2, \dots, p - 1$ ; (2.20) для  $(i+1) \leq (p - 1)$ ; (2.21) для  $i = p$ . Если матрица  $\mathbf{Y}_i$  относится к недопустимым, вернуться к шагу 2, положив  $i = i^*$ .

Шаг 7. Если для строки с номером  $(i+1) < (p-1)$  выполнено условие (2.23), заполнить ее форсированно единицами. Если для строк с номерами  $i + 1, i + 2$  и т.д., следующих подряд, выполнено условие (2.22), форсированно заполнить их нулями. Если форсирование состоялось, положить  $i^* = i$ , а  $i$  присвоить номер  $\mathfrak{S}$  последней заполненной строки, и перейти к шагу 5.

Шаг 8. Если  $i = i' + 1$ , идти к шагу 10.

Шаг 9. Проверить каноничность матрицы  $\mathbf{Y}_i$  по предикату (2.30) на подстановках  $g \in H_k$ . Если  $P(\mathbf{Y}_i)$  ложно, перейти к шагу 2. Если  $P(\mathbf{Y}_i)$  истинно, матрица  $\mathbf{Y}_i$  каноническая. При  $i = p - 1$  зафиксировать искомую матрицу  $\mathbf{Y}$  и идти к шагу 12.

Шаг 10. Если  $i = p - 1$ , идти к шагу 12.

Шаг 11. Если  $i \in I_k$ ,  $(i + 1) \in I_{(k+1)}$ , положить  $i' := i$ ,  $i := i + 1$  и идти к шагу 1. Если  $i, (i + 1) \in I_k$ , положить  $i := i + 1$  и идти к шагу 1.

Шаг 12. Уменьшая  $i$ , вернуться к максимальному его значению, при котором варианты  $y_i''$  не исчерпаны. Если  $i \in I_k$ , положить  $i' := |V_1| + |V_2| + L + |V_{k-1}|$  и идти к шагу 1 с найденными значениями  $i$  и  $i'$ . Если нет таких значений  $i$ , конец.

Работу алгоритма проиллюстрируем примером.

**Пример 2.13.** Пусть требуется построить неизоморфные  $(5, 5)$ -графы. Графические разбиения числа  $2q = 10$  на  $p = 5$  частей:  $\Pi_1 = (2, 2, 2, 2, 2)$ ;  $\Pi_2 = (3, 2, 2, 1)$ ;  $\Pi_3 = (3, 3, 2, 1, 1)$ ;  $\Pi_4 = (3, 3, 2, 2, 0)$ ;  $\Pi_5 = (4, 2, 2, 1, 1)$ . Построим все неизоморфные графы с  $\Pi_2 = (3, 2, 2, 2, 1)$ .

При выбранном разбиении множество проверочных подстановок  $H_k$  получается путем исключения из группы  $\text{Aut}(\mathbf{Y}_0)$  транспозиций и тождественной подстановки:  $g_1 = (3, 4, 2)$ ,  $g_2 = (4, 2, 3)$ . Зададим  $i' = 0$ ;  $i = 1$ ;  $d_i^i = 0$ .

Шаг 1.  $J_1 = (2, 3, 4)$ ,  $J_2 = (5)$ ,  $d_1'' = d_1 = 3$ ,  $i^* = 1$ .

Шаг 2.  $y_1'' = (1, 1, 1, 0)$  по условию каноничности из утверждения 2.1.

$$\text{Шаг 5.} \quad \mathbf{Y}_1 = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & & & \\ 1 & & 0 & & \\ 1 & & & 0 & \\ 0 & & & & 0 \end{pmatrix}.$$

Шаг 6.  $m_2'' > d_2'' = 1 > 0$ ;  $d_3'' = 1 > 0$ ;  $d_4'' = 1 > 0$ ;  $d_5' < d_5$ .

Шаг 11.  $i' = 1$ ,  $i = 2$ .

Шаг 1. Группы автоморфизмов  $\mathbf{Y}_1$  и  $\mathbf{Y}_0$  совпадают, следовательно,  $H_1 = H_0$ ,  $J_1 = (3, 4)$ ,  $J_2 = (5)$ ,  $d_2'' = 1$ ,  $i^* = 2$ .

Шаг 2.  $y_2'' = (1, 0, 0)$  по условию каноничности из утверждения 2.1.

$$\text{Шаг 5.} \quad \mathbf{Y}_2 = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & & \\ 1 & 0 & & 0 & \\ 0 & 0 & & & 0 \end{pmatrix}.$$

Шаг 6.  $m_3'' = d_3'' = 0$ ;  $d_4'' = 1 > 0$ ;  $d_5' < d_5$ .

Шаг 7.  $d_3'' = d_3$ ;  $d_4 - d_4' = m_4''$ ;  $y_3'' = (0, 0)$ ;  $y_4'' = (1)$  (заполняются форсированно);  $i^* = 2, i = 4$ .

$$\text{Шаг 5.} \quad \mathbf{Y}_4 = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{vmatrix}.$$

Шаг 6.  $d_5' = 1 = d_5$ .

Шаг 9.

$$\begin{vmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{vmatrix} \begin{vmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{vmatrix} < \mathbf{Y}_4.$$

$$\begin{vmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{vmatrix} \begin{vmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{vmatrix} < \mathbf{Y}_4;$$

Фиксируем каноническую матрицу  $\mathbf{Y}^{(1)} = \mathbf{Y}_4 = \mathbf{Y}_5$ .

Шаг 12.  $i' = 1, i = 2$ .

Шаг 1.  $J_1 = (3, 4), J_2 = (5), d_2'' = 1, i^* = 2$ .

Шаг 2.  $y_2'' = (0, 0, 1)$  по условию каноничности из утверждения 2.1.

$$\text{Шаг 5.} \quad \mathbf{Y}_2 = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & & \\ 1 & 0 & & 0 & \\ 0 & 1 & & & 0 \end{vmatrix}.$$

Шаг 6.  $m_3'' > d_3'' = 1 > 0$ ;  $d_4'' = 1 > 0$ ;  $d_5' = 1 = d_5$ .

Шаг 11.  $i = 3$ .

Шаг 1.  $J_1 = (4), J_2 = (5); d_3'' = 1, i^* = 3$ .

Шаг 2.  $y_3'' = (1, 0)$  по условию каноничности из утверждения 2.1.

Шаг 3.  $(1, 0, 0, 0, 1) \stackrel{lex}{\prec} (1, 0, 0, 1, 0)$ .

Шаг 2.  $\mathbf{y}_3'' = (0, 1)$  по условию каноничности из утверждения 2.1.

Шаг 3.  $(1, 0, 0, 0, 1) = (1, 0, 0, 0, 1)$ .

Шаг 5. 
$$\mathbf{Y}_3 = \left\| \begin{array}{ccccc} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & \\ 0 & 1 & 1 & & 0 \end{array} \right\|.$$

Шаг 6.  $m_4'' = d_4'' = 1 > 0$ ;  $d_5' > d_5$ ,  $\mathbf{Y}_2$  – недопустимая матрица.

Шаг 2. Варианты  $\mathbf{y}_3''$  исчерпаны.

Шаг 12.  $i = 1, i' = 0$ .

Шаг 1.  $J_1 = (2, 3, 4)$ ,  $J_2 = (5)$ ,  $d_1'' = 3$ ,  $i^* = 1$ .

Шаг 2.  $\mathbf{y}_1'' = (1, 1, 0, 1)$  по условию каноничности из утверждения 2.1.

Шаг 5. 
$$\mathbf{Y}_1 = \left\| \begin{array}{ccccc} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & & & \\ 1 & & 0 & & \\ 0 & & & 0 & \\ 1 & & & & 0 \end{array} \right\|.$$

Шаг 6.  $m_2'' > d_2'' = 1 > 0$ ;  $d_3'' = 1 > 0$ ;  $d_4'' = 2 > 0$ ;  $d_5' = d_5$ .

Шаг 11.  $i' = 1, i = 2$ .

Шаг 1. Группа  $\text{Aut}(\mathbf{Y}_{(i-1)}) = \text{Aut}(\Gamma_k) = \text{Aut}(\Gamma_1)$  состоит из подстановок  $e$  и  $g = (2, 3)$ . Следовательно,  $J_1 = (3)$ ,  $J_2 = (4)$ ,  $J_3 = (5)$ ,  $d_2'' = 1$ ,  $i^* = 2$ .

Шаг 2.  $\mathbf{y}_2'' = (1, 0, 0)$  по условию каноничности из утверждения 2.1.

Шаг 5. 
$$\mathbf{Y}_2 = \left\| \begin{array}{ccccc} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & & \\ 0 & 0 & & 0 & \\ 1 & 0 & & & 0 \end{array} \right\|.$$

Шаг 6.  $m_3'' > d_3'' = 0$ ;  $d_4'' > 0$ ;  $d_5' = d_5$ .

Шаг 7.  $d_3' = d_3$ ;  $\mathbf{y}_3'' = (0, 0)$  (заполняются форсированно);  $i^* = 2$ ;  $i = 3$ .

Шаг 5. 
$$\mathbf{Y}_2 = \begin{vmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \\ 1 & 0 & 0 & & 0 \end{vmatrix}.$$

Шаг 6.  $m_4'' < d_4'' = 2$ ;  $\mathbf{Y}_2$  – недопустимая матрица,  $i = i^* = 2$ .

Шаг 2.  $\mathbf{y}_2'' = (0, 1, 0)$  по условию каноничности из утверждения 2.1.

Шаг 5. 
$$\mathbf{Y}_2 = \begin{vmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & & \\ 0 & 1 & & 0 & \\ 1 & 0 & & & 0 \end{vmatrix}.$$

Шаг 6.  $m_3'' > d_3'' = 1 > 0$ ;  $d_4'' = 1 > 0$ ;  $d_5' = d_5$ .

Шаг 11.  $i = 3$ .

Шаг 1.  $J_1 = (4)$ ;  $J_5 = (5)$ ;  $d_3'' = 1$ ,  $i^* = 3$ .

Шаг 2.  $\mathbf{y}_3'' = (1, 0)$  по условию каноничности из утверждения 2.1.

Шаг 3.  $(1, 0, 0, 1, 0) = (1, 0, 0, 1, 0)$ .

Шаг 5. 
$$\mathbf{Y}_3 = \begin{vmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & \\ 1 & 0 & 0 & & 0 \end{vmatrix}.$$

Шаг 6.  $d_4'' = 0$ ;  $d_5' = d_5$ .

Шаг 7.  $d_4' = d_4$ ;  $\mathbf{y}_4'' = (0)$  (заполняется форсированно),  $i = 4$ ,  $i^* = 3$ .

Шаг 5. 
$$\mathbf{Y}_4 = \begin{vmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{vmatrix}.$$

Шаг 6.  $d_5' = d_5$ .

Шаг 9.  $H_k = \emptyset$ . Поэтому проверку по предикату (2.30) пропускаем.

Фиксируем каноническую матрицу  $\mathbf{Y}^{(2)} = \mathbf{Y}_4 = \mathbf{Y}_5$ .

Шаг 12.  $i = 3, i' = 1$ .

Шаг 1.  $J_1 = (4); J_5 = (5); d_3'' = 1, i^* = 3$ .

Шаг 2.  $\mathbf{y}_3'' = (0, 1)$  по условию каноничности из утверждения 2.1.

Шаг 4.  $(1, 0, 0, 1, 0) \succ^{lex} (1, 0, 0, 0, 1)$ .

Шаг 5. 
$$\mathbf{Y}_3 = \begin{vmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{vmatrix}.$$

Шаг 6.  $m_4'' = d_4'' = 1 > 0; d_5' > d_5$ ; следовательно, матрица  $\mathbf{Y}_2$  относится к недопустимым;  $i = i^* = 3$ .

Шаг 2. Варианты  $\mathbf{y}_3''$  исчерпаны.

Шаг 12.  $i = 2, i' = 1$ .

Шаг 1.  $J_1 = (3); J_2 = (4); J_3 = (5); d_2'' = 1, i^* = 2$ .

Шаг 2.  $\mathbf{y}_2'' = (0, 0, 1)$  по условию каноничности из утверждения 2.1.

Шаг 5. 
$$\mathbf{Y}_2 = \begin{vmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{vmatrix}.$$

Шаг 6.  $m_3'' > d_3'' = 1 > 0; d_4'' = 2 > 0; d_5' > d_5$ ;  $\mathbf{Y}_2$  — недопустимая матрица.  $i^* = 2$ .

Шаг 2. Варианты  $\mathbf{y}_2''$  исчерпаны.

Шаг 12. Нет таких значений  $i$ , конец.

Графы, представленные найденными каноническими матрицами, приведены на рис. 2.5, а, б.

Аналогично по алгоритму 2.2 строятся канонические матрицы  $\mathbf{Y}$  с разбиениями  $\Pi_1, \Pi_3, \Pi_4, \Pi_5$ :



$$\mathbf{Y}^{(3)} = \begin{vmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{vmatrix}; \mathbf{Y}^{(4)} = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{vmatrix};$$

$$\mathbf{Y}^{(5)} = \begin{vmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{vmatrix}; \mathbf{Y}^{(6)} = \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{vmatrix}.$$

Читателю предоставляется самостоятельно убедиться в этом. Графы, построенные по матрицам  $\mathbf{Y}^{(3)}$ ,  $\mathbf{Y}^{(4)}$ ,  $\mathbf{Y}^{(5)}$ ,  $\mathbf{Y}^{(6)}$ , приведены на рис 2.5, в, г, д, е соответственно.

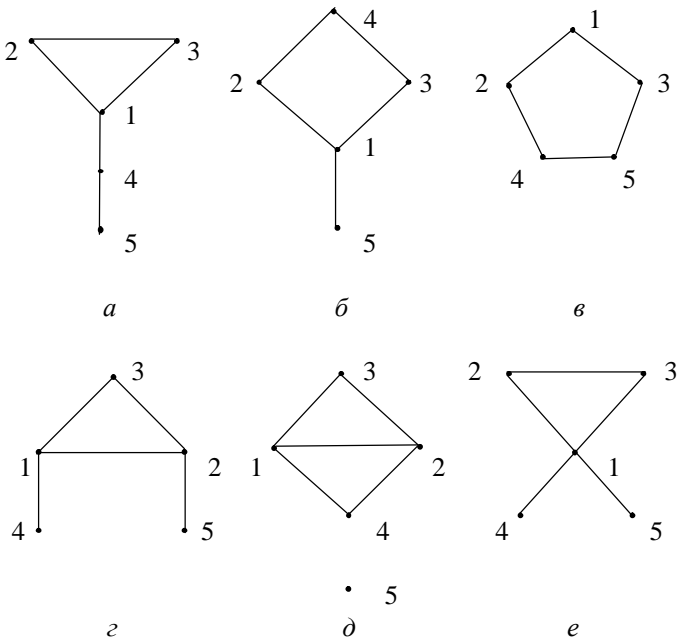


Рис 2.5

## 2.5. Конструктивное перечисление неизоморфных деревьев

### 2.5.1. Постановка задачи и выбор способа представления дерева

Дерево – связный ациклический граф. Применение деревьев для моделирования структурных свойств систем и отношений между объектами различной природы широко распространено: при разработке информационных систем, в том числе систем связи; при исследовании структур химических соединений; при перечислении разделительных схем, химико-технологических схем и т.д. Кроме того, деревья, являясь простейшими связными графами, служат хорошими моделями для решения различных графовых задач. По этим причинам внимание исследователей давно привлекает задача построения каталогов деревьев ([2; 23–25] и др.) с целью проверки рабочих гипотез и накопления фактов для построения новых.

Известно, что задача построения деревьев с заданным числом вершин решается однозначно только при  $p = 2$ . С увеличением  $p$  число  $t$  вариантов деревьев с  $p$  вершинами быстро растет по закону

$$t = p^{(p-2)}.$$

Вместе с ростом числа деревьев увеличивается число их изоморфных вариантов.

Большинство известных методов и алгоритмов конструктивного перечисления деревьев ([25] и др.) имеет общий недостаток, резко понижающий их эффективность: они по несколько раз порождают изоморфные деревья, процедура опознания и исключения которых достаточно трудоемка.

Путем использования идеи канонического упорядочения комбинаторных объектов удалось построить [24] линейную процедуру конструктивного перечисления неизоморфных деревьев, т.е. с оценкой  $O(N)$ , где  $N$  – число неизоморфных деревьев. Однако недостаток этой процедуры состоит в том, что результатом ее применения к построению деревьев с  $p$  вершинами являются также деревья с  $(p-1)$ ,  $(p-2)$ , ..., 2, 1 вершинами, т.е. в число  $N$  включается избыточное число деревьев, которые не являются решением поставленной задачи. Наша цель – показать, что с применением информации о симметриях дерева данная задача может быть решена более эффективно. Покажем это на примере корневых деревьев, число которых растет с ростом  $p$  по закону

$$t = p^{(p-1)},$$

т.е. еще быстрее, чем в случае некорневых деревьев.

Корневое (ориентированное, выходящее [27]) дерево можно рассматривать как  $Q$ -структуру с  $|I| = q$ ,  $|J| = p$ ,  $L_i = \{1, 2\}$ ,  $i = 1, 2, \dots, q = p - 1$ . В множестве  $L_i$  единица соответствует началу ориентированного  $i$ -го ребра дерева, а двойка – его концу. Как  $Q$ -структура дерево может быть представлено взвешенной матрицей инцидентностей  $\mathbf{X} = \|x_{ij}\|_{q \times p}$ ,  $x_{ij} \in \{0, 1, 2\}$ . В этом случае генерирование корневых неизоморфных деревьев представляется задачей второго порядка сложности, и к ее решению могут быть применены средства, рассмотренные в гл. 1. Покажем, что, учитывая особенности корневых деревьев, поставленную задачу можно свести к задачам первого порядка сложности, т.е. к векторным задачам.

Пусть  $1$  – порядковый номер корневой вершины. Выделим два подмножества во множестве  $J$  номеров вершин дерева:  $J_1 = \{1\}$  и  $J_2 = \{2, 3, \dots, p\}$ . Упорядочим множество корневых деревьев с  $p$  вершинами по старшинству их матриц  $\mathbf{X}$ . Будем называть дерево  $T$  каноническим, если канонической является матрица  $\mathbf{X}$ , т.е. ее невозможно сделать старше никакими подстановками элементов множеств  $I, J$ , сохраняющими разбиение множества  $J$ . Отметим, что выбранный способ канонического упорядочения корневых деревьев не согласуется с предложенным в работе [24].

В матрице  $\mathbf{X}$   $i$ -му ребру,  $i = 1, 2, \dots, q$ , дерева  $T$  соответствует  $i$ -й вектор-строка  $\mathbf{x}_i$ , а вершинам  $v_{j_i}'$ ,  $v_{j_i}''$ , инцидентным этому ребру, отвечают элементы  $1$  и  $2$  на пересечении  $i$ -й строки соответственно с  $j_i'$ -м и  $j_i''$ -м столбцами. Из построения при выбранной нумерации вершин дерева и его ребер, а также из утверждений 1.8, 1.9 следует, что в канонической матрице  $\mathbf{X}$  ненулевые элементы вектора-строки  $\mathbf{x}_i$  упорядочены по старшинству в алфавите  $(1, 2)$ , а векторы-строки  $\mathbf{x}_i$  – упорядочены лексикографически в алфавите  $(1, 2, 0)$ . Кроме этих признаков каноничности корневному дереву присущ еще один специфический признак, указанный в следующем утверждении.

**Утверждение 2.9.** *Если дерево  $T$  является каноническим, между порядком на множестве  $I$  номеров его ребер  $i_1, i_2, \dots, i_q$  и порядком на множестве  $J$  номеров его же вершин  $j_1', j_2', \dots, j_p'$  существует соотношение*

$$i_\alpha < i_\beta \Leftrightarrow j_\alpha' < j_\beta'.$$

**Доказательство.** Доказывается от противного. Допустим, что дерево  $T$  каноническое, т.е. описывается канонической матрицей  $\mathbf{X}$ , но

$j''_{\alpha} > j''_{\beta}$  при  $i_{\alpha} < i_{\beta}$  (равенство  $j''_{\alpha}$  и  $j''_{\beta}$  исключается, так как в корневом дереве не могут два ребра входить в одну вершину). Но тогда матрицу  $\mathbf{X}$  можно увеличить простой перестановкой столбцов с номерами  $j''_{\alpha}$  и  $j''_{\beta}$ . Значит, матрица  $\mathbf{X}$  не была канонической, что противоречит первоначальному допущению.

Так как требование утверждения 2.9 относится к любой паре ребер дерева  $T$ , из этого утверждения вытекает

**С л е д с т в и е 2.9.1.** В канонической матрице  $\mathbf{X}$  корневого дерева  $T$  для всех  $i \in I$   $j''_i = i + 1$ .

Процедура построения дерева отличается тем, что каждое добавленное ребро присоединяет к нему только одну вершину. Из следствия 2.9.1 заключаем, что при построении канонического дерева с  $p$  вершинами каждая новая вершина присоединяется к нему в порядке увеличения ее номера: первое ребро к тривиальному  $(1, 0)$ -дереву, содержащему только корневую вершину, присоединяет вершину с номером 2, второе – вершину с номером 3 и т.д.

Пусть  $V = (v_1, v_2, \dots, v_p)$  – множество вершин дерева  $T$ , а  $(v_{j'_i}, v_{j''_i})$  – его  $i$ -е ребро,  $j'_i, j''_i \in \{1, 2, \dots, p\}$ ,  $i \in \{1, 2, \dots, p-1\}$ , причем  $j'_i (j''_i)$  – номер вершины, из которой ребро выходит (в которую оно входит). Дерево представим множеством таких пар. Ограничимся рассмотрением только тех деревьев  $T$ , которые удовлетворяют перечисленным условиям каноничности. При выбранном представлении дерева эти условия интерпретируются следующими требованиями к порядку на множестве номеров его вершин  $v_{j'}, v_{j''}$ :

- 1)  $j'_i < j''_i$  в силу упорядоченности ненулевых координат вектора  $\mathbf{x}_i$ ;
- 2)  $j'_i \geq j'_{(i-1)}$  в силу лексикографической упорядоченности векторов-строк  $\mathbf{x}_i$  матрицы  $\mathbf{X}$ ;
- 3) множество пар  $(j'_i, j''_i)$  должно быть упорядочено лексикографически в алфавите  $(1, 2, \dots, p)$ ;
- 4) номера  $j''_i$  должны образовывать нарастающую последовательность целых чисел, пробегающих все значения от 2 до  $p$ .

Если при генерировании деревьев ограничиться только построением деревьев с перечисленными свойствами, число просматриваемых вариантов резко сократится без риска потерять неизоморфные варианты.

Заметим, что при выбранном представлении любое корневое дерево с  $p$  вершинами характеризуется одинаковой последовательностью координат  $j_i''$ . Различаются они между собой только последовательностью их координат  $j_i'$ . Таким образом, в рассматриваемом множестве деревьев каждое дерево может быть однозначно представлено вектором  $J' = (j_1', j_2', \dots, j_q')$ . Теперь задача генерирования множества неизоморфных корневых деревьев с  $p$  вершинами может быть сформулирована как задача порождения множества  $q$ -мерных векторов с повторяющимися координатами из множества  $\{1, 2, \dots, p\}$ , которые не приводятся к порожденным ранее с помощью симметрий дерева и его составных частей. Согласно предложенному выше подходу к решению этой задачи необходимо изучить симметрии корневых деревьев и использовать информацию о них для отсекаания изоморфных вариантов в процессе генерирования таких векторов.

### 2.5.2. Симметрии корневого дерева

Известно [ 27; 28], что группа автоморфизмов дерева  $T$  с  $p$  вершинами может быть построена из симметрических групп  $H_{r_1}, H_{r_2}, \mathbf{K}, H_{r_i}$  подстановок конечных множеств, мощности которых  $r_1, r_2, \mathbf{K}, r_i$  удовлетворяют соотношениям

$$r_1 + r_2 + \mathbf{K} + r_i \leq p,$$

путем повторного применения к этим группам операций прямого суммирования [2; 22; 28] и сплетения [2; 28]. Отсутствие циклических подстановок (циклов) в автоморфизмах дерева, очевидно, связано с отсутствием циклов в самом дереве.

Нетрудно догадаться, что в соответствии с введенной здесь терминологией указанные множества с мощностями  $r_1, r_2, \mathbf{K}, r_i$  – это классы эквивалентности по симметриям дерева или его поддеревьев ( $h$ -классы). Наша задача состоит в том, чтобы выявить эти классы, их соподчиненность в дереве и на основе этих сведений разработать методику и алгоритм построения группы автоморфизмов всего дерева из симметрических групп подстановок элементов этих классов.

Пусть  $H$  – группа автоморфизмов корневого дерева  $T$  с  $M$  ярусами. Под ярусом подразумевается подмножество вершин дерева, равноотстоящих от корневой вершины. Очевидно следующее

**Утверждение 2.10.** Под действием любого автоморфизма  $h \in H$  корневого дерева  $T$  множество  $V_m$  вершин его  $m$ -го яруса,  $m \in \{1, 2, \dots, M\}$ , отображается на себя.

Из утверждения 2.10 следует, что каждый из искомым классов эквивалентности – это подмножество множества вершин одного яруса дерева. Поэтому удобнее считать, что дерево строится путем постепенного наращивания числа его ярусов, чтобы проследить образование классов эквивалентности, их изменение, связь между ними и влияние этих факторов на группу автоморфизмов дерева  $T$ .

Пусть  $T_m$  – часть дерева  $T$ , содержащая  $m$  его ярусов,  $m \in \{0, 1, 2, \dots, M\}$  ( $m = 0$  соответствует дереву  $T_0$ , состоящему из одной корневой вершины). Тогда  $V_m$  – это множество его концевых вершин,  $J_m$  – множество порядковых номеров этих вершин. Множество  $V_m$  состоит из подмножеств  $V_{mj}$ , элементы каждого из которых смежны общему элементу  $v_j \in V_{(m-1)}$ . Назовем множество  $V_{mj}$   $j$ -классом.

Таким образом, в дереве  $T$  каждому элементу  $v_j$ , принадлежащему  $(m-1)$ -му ярусу, поставлено в соответствие множество вершин  $m$ -го яруса. Тем самым задано многозначное (точечно-множественное) отображение  $\alpha_j: v_j \rightarrow V_{mj}$ . Это значит, что множество  $V_{mj}$  вершин  $m$ -го яруса дерева  $T$  можно рассматривать как образ вершины  $v_j$  из  $(m-1)$ -го яруса,

$$V_{mj} = v_j \alpha_j, \quad j \in J_{(m-1)}, \quad (2.31)$$

а  $v_j$  – как прообраз множества  $V_{mj}$ ,

$$v_j = V_{mj} \alpha_j^{-1}. \quad (2.32)$$

Выражения (2.31), (2.32) отражают соподчиненность  $j$ -классов, относящихся с двум соседним ярусам. Множество элементов  $m$ -го яруса можно представить через элементы его  $(m-1)$ -го яруса как

$$V_m = \bigcup_{j \in J_{(m-1)}} V_{mj} = \bigcup_{j \in J_{(m-1)}} v_j \alpha_j. \quad (2.33)$$

Очевидно, что дерево  $T_m$  инвариантно к любым перестановкам  $v_j \in V_{mj}$ ,  $j \in J_{(m-1)}$ , и, следовательно, элементы симметрической группы подстановок множества  $V_{mj}$  являются элементами группы автоморфизмов

дерева  $T_m$ . Введем для нее обозначение  $H(V_{mj})$ , указывающее в скобках множество, на котором эти подстановки определены:

$$H(V_{mj}) = H(v_j \alpha_j), \quad j \in J_{(m-1)}.$$

Множества  $V_{mj}$ ,  $j \in J_{(m-1)}$ , не пересекаются, поэтому подстановки элементов этих множеств выполняются независимо друг от друга. Это значит, что образованная прямым суммированием симметрических групп  $H(V_{mj})$  группа

$$H(V_m) = \bigotimes_{j \in J_{(m-1)}} H(V_{mj}) = \bigotimes_{j \in J_{(m-1)}} H(v_j \alpha_j) \quad (2.34)$$

также является элементом группы автоморфизмов дерева  $T_m$ .

Сопоставив (2.33), и (2.34), видим, что объединению подмножеств  $V_{mj}$  множества вершин  $m$ -го яруса соответствует прямая сумма симметрических групп подстановок элементов этих множеств.

Перейдем к дереву  $T_{(m+1)}$ , добавив  $(m+1)$ -й ярус к дереву  $T_m$ .

Теперь вершины  $(m+1)$ -го яруса дерева  $T_{(m+1)}$  являются концевыми. Все, что сказано о концевых вершинах дерева  $T_m$ , можно отнести и к вершине  $v_j \in V_{(m+1)}$ , т.е. тогда множество вершин  $V_{(m+1)}$   $(m+1)$ -го яруса дерева  $T_{(m+1)}$  можно рассматривать как объединение подмножеств

$$V_{(m+1)j'} = v_{j'} \alpha_{j'}, \quad j' \in J_{mj'}, \quad (2.35)$$

каждое из которых является многозначным отображением  $\alpha_{j'}$ , ставящим в соответствие элементу  $v_{j'} \in V_m$  множество  $V_{(m+1)j'}$ ,  $\alpha_{j'} : v_{j'} \rightarrow V_{(m+1)j'}$ . Для них справедливы соотношения, аналогичные (2.32), (2.33), (2.34):

$$v_{j'} \in V_{(m+1)j'} \alpha_{j'}^{-1} \quad (2.36)$$

$$V_{(m+1)} = \bigcup_{j' \in J_m} v_{j'} \alpha_{j'} = \bigcup_{j' \in J_m} V_{(m+1)j'}; \quad (2.37)$$

$$H(V_{(m+1)}) = \bigotimes_{j' \in J_m} H(V_{(m+1)j'}) = \bigotimes_{j' \in J_m} H(v_{j'} \alpha_{j'}). \quad (2.38)$$

Из (2.31), (2.35) следует, что каждое множество  $V_{mj}$ ,  $j \in J_{(m-1)}$ , является образом для элементов  $v_j \in V_{(m-1)}$ , и одновременно его элементы  $v_{j'} = v_j \alpha_j$  являются прообразами множеств  $V_{(m+1)j'}$ ,  $j' \in J_{mj}$ :

$$V_{mj} = \bigcup_{j' \in J_{mj}} v_{j'} = \bigcup_{j' \in J_{mj}} V_{(m+1)j'} \alpha_{j'}^{-1}. \quad (2.39)$$

Множество всех вершин  $m$ -го яруса  $V_m$  представим через элементы  $(m+1)$ -го яруса

$$V_m = \bigcup_{j \in J_{(m-1)}} V_{mj} = \bigcup_{j \in J_{(m-1)}} \bigcup_{j' \in J_{mj}} V_{(m+1)j'} \alpha_{j'}^{-1} \quad (2.40)$$

и проследим, как переход от  $T_m$  к  $T_{(m+1)}$  отразится на взаимозаменяемости элементов  $v_{j'} \in V_{mj}$ ,  $j \in J_{(m-1)}$ .

Сначала рассмотрим случай, когда все вершины  $v_{j'} \in V_{mj}$  имеют одинаковые полустепени исхода  $d_j$ . Под действием  $h \in H(V_{mj})$ ,  $j \in J_{(m-1)}$ , вместе с вершинами  $m$ -го яруса, принадлежащими множеству  $V_{mj}$ , меняются местами множества  $V_{(m+1)j'}$  вершин  $(m+1)$ -го яруса, которые являются образами вершин  $v_{j'} \in V_{mj}$ , без изменения порядка следования элементов каждого из этих множеств внутри его. В рассматриваемом случае все множества  $V_{(m+1)j'}$ ,  $j' \in J_{mj}$ , равномощны и взаимозаменяемы. В свою очередь, на каждом из множеств  $V_{(m+1)j'}$ ,  $j' \in J_{mj}$ , определена симметрическая группа автоморфизмов  $H(V_{(m+1)j'})$ , как было уже отмечено выше. Вместе все группы  $H(V_{(m+1)j'})$ ,  $j' \in J_{mj}$ , образуют группу

$$H(V_{(m+1)j}) = \bigotimes_{j' \in J_{mj}} H(V_{(m+1)j'}),$$

так как множества  $V_{(m+1)j'}$ ,  $j' \in J_{mj}$ , не пересекаются и подстановки элементов этих множеств выполняются независимо. Очевидно, что дерево  $T_{(m+1)}$  инвариантно к любому сочетанию подстановок  $h \in H(V_{mj})$  и  $h \in H(V_{(m+1)j'})$ . Тогда  $H(V_{(m+1)j})$ -сплетение вокруг  $H(V_{mj})$  [28]

$$H(V_{mj})[H(V_{(m+1)j})] = H(V_{mj}) \left[ \bigotimes_{j' \in J_{mj}} H(V_{(m+1)j'}) \right] \quad (2.41)$$



также является элементом группы автоморфизмов дерева  $T_{(m+1)}$ . Точнее это группа автоморфизмов  $j$ -поддерева, т.е. части дерева  $T_{(m+1)}$ , висящей на вершине  $v_j \in V_{(m-1)}$ .

Сопоставив (2.40) и (2.41), убеждаемся, что операция сплетения соответствует совокупности операций обратных отображений множеств  $V_{(m+1)j}$ , т.е. переходу от  $j$ -классов  $(m+1)$ -го яруса к смежным вершинам  $m$ -го яруса. Таким образом, операцией объединения выражается связь между группами подстановок на  $j$ -классах каждого яруса, а операцией сплетения – на соседних ярусах. Опираясь на указанную закономерность, для  $m$ -го яруса по (2.40) запишем

$$H(V_m) = \bigotimes_{j \in J_{(m-1)}} H(V_{mj}) [H(V_{(m+1)j})] = \bigotimes_{j \in J_{(m-1)}} H(V_{mj}) \left[ \bigotimes_{j' \in J_{mj}} H(V_{(m+1)j'}) \right]. \quad (2.42)$$

Сравнивая (2.34) и (2.42), видим изменения в группе автоморфизмов дерева  $T_m$  после добавления к нему  $(m+1)$ -го яруса.

Пусть теперь вершины множества  $V_{mj}$  разбиты на подмножества  $V_{mjk}$ , состоящие из вершин  $v_j$  с одинаковыми полустепенями исхода  $d_j$ ,  $k = 1, 2, \dots, K_{mj}$ , где  $K_{mj}$  – число таких подмножеств:

$$V_{mj} = \bigcup_{k=1}^{K_{mj}} V_{mjk}, \quad j \in J_{(m-1)}.$$

Условимся множество  $V_{mjk}$ ,  $j \in J_{(m-1)}$ ,  $k \in \{1, 2, \dots, K_{mj}\}$ , называть  $d$ -классом. Учитывая соотношение

$$V_{mjk} = \bigcup_{j' \in J_{mjk}} V_{(m+1)j'} \alpha_{j'k}^{-1},$$

где  $\alpha_{j'k} : v_{j'} \rightarrow V_{(m+1)j'}$ ,  $v_{j'} \in V_{mjk}$ ,

получим

$$V_{mj} = \bigcup_{k=1}^{K_{mj}} \left( \bigcup_{j' \in J_{mjk}} V_{(m+1)j'} \alpha_{j'k}^{-1} \right). \quad (2.43)$$

Если в только что рассмотренном случае дерево  $T_{(m+1)}$  инвариантно к любым перестановкам вершин  $v_{j'} \in V_{mj}$ , то теперь – только к тем из них,

которые характеризуются одинаковыми полустепенями исхода  $d_{j'}$ , т.е. внутри  $d$ -класса. Значит, теперь симметрические группы подстановок множеств  $V_{mjk}$  являются элементами группы автоморфизмов дерева  $T_{(m+1)}$ .

По (2.43) запишем для  $v_j$ -поддерева дерева  $T_{(m+1)}$ , у которого корневой является вершина  $v_j$  из  $(m-1)$ -го яруса,

$$H(V_{mj}) = \bigotimes_{k=1}^{K_{mj}} H(V_{mjk}) \left[ \bigotimes_{j' \in J_{mjk}} H(V_{(m+1)j'} \right], \quad (2.44)$$

а с учетом (2.33) – для дерева  $T_{(m+1)}$

$$H(V_m) = \bigotimes_{j \in J_{(m-1)}} \left\{ \bigotimes_{k=1}^{K_{mj}} H(V_{mjk}) \left[ \bigotimes_{j' \in J_{mjk}} H(V_{(m+1)j'} \right] \right\}. \quad (2.45)$$

Из (2.45) видно, что дополнительному разбиению множества  $V_{mj}$  вершин  $m$ -го яруса соответствует появление дополнительного суммирования в выражении для  $H(V_m)$ .

Отметим, что множество вершин  $m$ -го яруса дерева  $T$ , на котором определена симметрическая группа автоморфизмов, составляет класс эквивалентности по симметриям, т.е.  $h$ -класс. Значит,  $h$ -классами являются  $d$ - и  $j$ -классы деревьев  $T_m$  и  $T_{(m+1)}$ , а операции сплетения указывают на наличие  $H$ -эквивалентных  $h$ -классов в них.

Из сказанного следует, что построение группы автоморфизмов дерева сводится к попеременному применению операций объединения симметрических групп подстановок на  $d$ - и  $j$ -классах каждого яруса дерева с операциями сплетения этих групп с соответствующими симметрическими группами подстановок на  $d$ - и  $j$ -классах, которые принадлежат соседним ярусам, расположенным ближе к корневой вершине. Последовательность этих операций определяется конфигурацией дерева  $T$ . Очевидно, что подстановки на  $d$ - и  $j$ -классах, содержащих по единственному элементу, следует исключать в процессе построения группы автоморфизмов дерева  $T$  при условии, что эти классы не могут поменяться местами при перестановке вершин, образами которых они являются.

Пример.2.14. Пусть требуется построить группу автоморфизмов  $H_T$  дерева  $T$ , приведенного на рис.2.6.

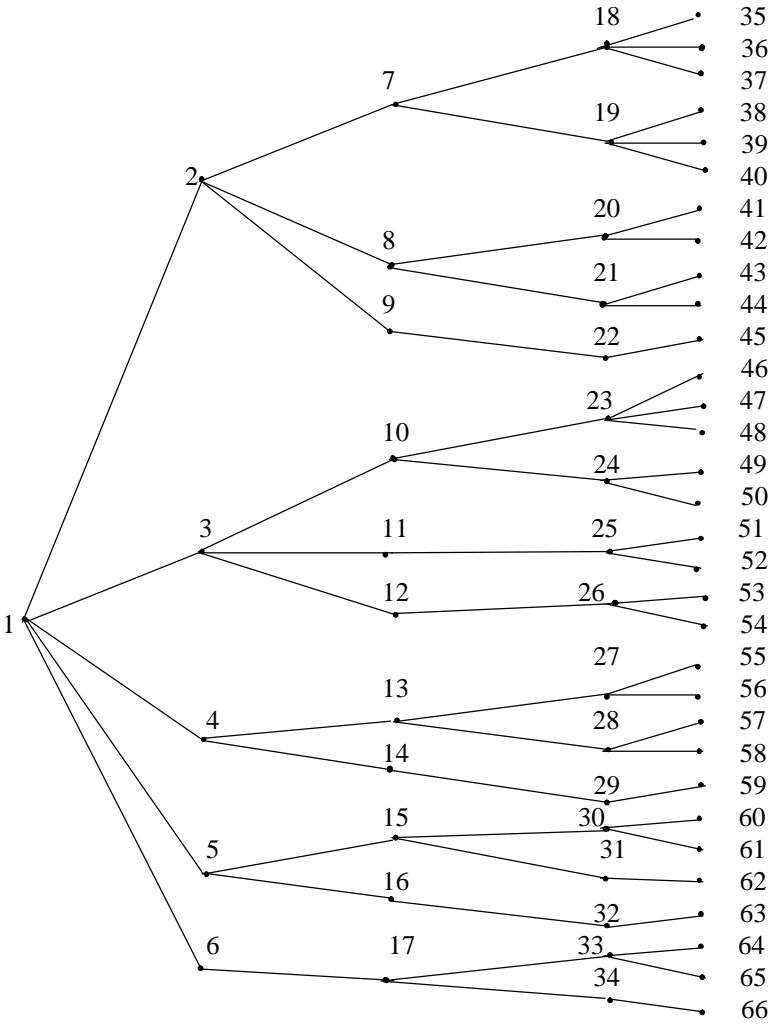


Рис. 2.6

Условимся вершины дерева  $v_j \in V$  представлять их порядковыми номерами  $j \in J$ . Руководствуясь взаимным расположением  $d$ - и  $j$ - классов дерева на рис.2.6, и чередуя соответствующие операции объединения и сплетения симметрических групп подстановок на этих классах, запишем сначала группу автоморфизмов  $H_T$  дерева  $T$  без всяких исключений:

$$\begin{aligned}
 H_T = & H(1) [ H(2) [ H(7) [ H(18, 19) [ H(35, 36, 37) \otimes H(38, 39, 40) ] ] \otimes \\
 & \otimes H(8) [ H(20, 21) [ H(41, 42) \otimes H(43, 44) ] ] \otimes H(9) [ H(22) [ H(45) ] ] ] \otimes \\
 & \otimes H(3) [ H(10) [ H(23) [ H(46, 47, 48) ] \otimes H(24) [ H(49, 50) ] ] \otimes \\
 & \otimes H(11, 12) [ H(25) [ H(51, 52) ] \otimes H(26) [ H(53, 54) ] ] ] \otimes \\
 & \otimes H(4) [ H(13) [ H(27, 28) [ H(55, 56) \otimes H(57, 58) ] ] \otimes \\
 & \otimes H(14) [ H(29) [ H(59) ] ] ] \\
 & \otimes H(5) [ H(15) [ H(30) [ H(60) \otimes H(61) ] \otimes [ H(31) [ H(62) ] ] \otimes \\
 & \otimes H(16) [ H(32) [ H(63) ] ] ] \otimes \\
 & \otimes H(6) [ H(17) [ H(33) [ H(64, 65) ] \otimes H(34) [ H(66) ] ] ] ] .
 \end{aligned}$$

Цифры над квадратными скобками – это номера ярусов дерева  $T$ . Исключив подстановки на множествах, содержащих по одному элементу, получаем искомую группу автоморфизмов  $H_T$ :

$$\begin{aligned}
 H_T = & H(18, 19) [ H(35, 36, 37) \otimes H(38, 39, 40) ] \otimes H(20, 21) [ H(41, 42) \otimes H(43, 44) ] \otimes \\
 & \otimes H(46, 47, 48) \otimes H(49, 50) \otimes H(11, 12) [ H(25) [ H(51, 52) ] \otimes H(26) [ H(53, 54) ] ] \otimes \\
 & \otimes H(27, 28) [ H(55, 56) \otimes H(57, 58) ] \otimes H(60, 61) \otimes H(64, 65) .
 \end{aligned}$$

Последний результат нетрудно получить непосредственно по рис. 2.6, если ясен механизм образования группы автоморфизмов деревьев.

Сопоставление (2.42) и (2.45) приводит к выводу, что переход от дерева  $T_m$  к дереву  $T_{(m+1)}$ , т.е. добавление  $(m+1)$ -го яруса, сопровождается дроблением  $h$ -классов  $m$ -го яруса. Частный случай, при котором сохраняются  $h$ -классы  $m$ -го яруса, – это когда степени всех вершин  $v_j \in V_m$  одинаковы.

Очевидно, что такая тенденция – к постепенному дроблению  $h$ -классов  $m$ -го яруса – будет наблюдаться и в дальнейшем при добавлении очередных ярусов:  $(m+2)$ -го,  $(m+3)$ -го, ...,  $M$ -го. Для сохранения эквивалентности элементов  $m$ -го яруса необходимо выполнить специальные условия, о которых пойдет речь далее, в утверждении 2.12.

Но сначала рассмотрим еще один признак каноничности деревьев, сокращающий поиск неизоморфных вариантов в процессе их синтеза.

**У т в е р ж д е н и е 2.11.** *Если дерево  $T_{(m+1)}$  каноническое, каждое подмножество  $V_{mj}$  вершин его  $m$ -го яруса,  $m = 1, 2, \dots, M-1$ , упорядочено по невозрастанию полустепеней исхода  $d_j$ .*

**Д о к а з а т е л ь с т в о .** Обратимся к матрице  $X_{(m+1)}$  дерева  $T_{(m+1)}$ . В ней ребрам, исходящим из вершины  $v_j \in V_{(m+1)}$ , соответствует  $d_j$  строк с единицами в  $j$ -м столбце. Чем старше вершина  $v_j$ , тем меньше порядковые номера этих строк.

Доказываем от противного. Допустим, что дерево канонично, значит и его матрица  $X_{(m+1)}$  канонична, но при этом для вершин  $v_a, v_b \in V_m$  нарушена указанная упорядоченность  $a, b \in J$ , т.е. в матрице  $X_{(m+1)}$  сначала встречается  $d_a$  строк с единицами в  $a$ -м столбце, затем  $d_b$  строк с единицами в  $b$ -м столбце при  $d_a < d_b$ .

Поменяем местами вершины  $v_a$  и  $v_b$  в дереве  $T_{(m+1)}$ , упорядочив их по невозрастанию полустепеней исхода. Тогда в матрице  $X_{(m+1)}$  поменяются местами соответствующие столбцы и группы строк. В результате в строке матрицы  $X_{(m+1)}$ , которая совпала с  $d_b$ -й строкой во множестве из  $d_b$  строк, единица переместится на старшее место. При этом во всех ее предшествующих строках никаких изменений не произойдет, так как поменялись местами только вершины  $v_a, v_b$ , относящиеся к одному классу эквивалентности  $V_{mj}$  дерева  $T_m$ . Следовательно, матрица  $X_{(m+1)}$  станет старше, что противоречит предположению о ее каноничности.

Учитывая утверждение 2.11, будем рассматривать только множество  $M_T$  деревьев, у которых вершины  $v_j \in V_{mj}$  каждого яруса упорядочены по убыванию их полустепеней исхода  $d_j$ . Тогда множества  $V_{mj}$  распадаются на компактные подмножества  $V_{mjk}$ , для которых выполняется соотношение

$$k' < k'' \Leftrightarrow d'_k \exists d''_k.$$

Множеству вершин  $v_j \in V_m$  дерева  $T \in M_T$  поставим в соответствие множество  $D_m = (d_1, d_2, \mathbf{K}, d_{|V_m|})$  их степеней  $d_j$ . Множество вершин  $m'$ -го яруса дерева  $T$ , в которое отображается вершина  $v_j \in V_m$ ,  $m < m'$ , назовем ее образом в  $m'$ -м ярусе. Тогда подмножество  $\Delta D_{m'}(v_j)$ ,  $v_j \in V_m$ , множества  $D_{m'}$ , соответствующее образу вершины  $v_j$  в  $m'$ -м ярусе, назовем разбиением по степеням ее образа в  $m'$ -м ярусе.

**У т в е р ж д е н и е 2.12.** *Для того чтобы вершины  $v_a, v_b$   $m$ -го яруса дерева  $T \in M_T$  относились к общему  $h$ -классу, необходимо и достаточно, чтобы совпали разбиения по степеням их образов в каждом из последующих ярусов, т.е. с номерами  $m+1, m+2, \dots, M$ .*

**Д о к а з а т е л ь с т в о .**

*Необходимость.* Принадлежность вершин  $v_a, v_b$  одному классу эквивалентности дерева  $T$  означает, что они взаимозаменяемы, и вместе с ними взаимозаменяемыми являются  $v_a$ - и  $v_b$ -поддеревья, т.е. части дерева  $T$ , для которых вершины  $v_a$  и  $v_b$  – корневые. Если взаимозаменяемые поддеревья поменять местами, множество вершин  $m'$ -го яруса,  $m' \in \{m+1, m+2, \dots, M\}$ ,  $v_a$ -поддерева перейдет во множество вершин  $m'$ -го яруса  $v_b$ -поддерева, и наоборот. Это означает взаимозаменяемость ярусов  $v_a$ - и  $v_b$ -поддеревьев с одинаковыми порядковыми номерами  $m'$ . Вершина  $m'$ -го яруса одного поддерева может перейти в вершину  $m'$ -го яруса второго поддерева при условии, что они характеризуются одинаковыми полустепенями исхода. Поэтому  $v_a$ - и  $v_b$ -поддеревья имеют одинаковое разбиение по полустепеням исхода вершин ярусов с одинаковыми порядковыми номерами  $m'$ . Множество вершин  $m'$ -го яруса  $v_a$  ( $v_b$ )-поддерева – это образ вершины  $v_a$  ( $v_b$ ) в  $m'$ -м ярусе дерева  $T$ . Следовательно, разбиения образов вершин  $v_a, v_b$  в каждом из ярусов с номерами  $m'$  идентичны. А в деревьях  $T \in M_T$  они совпадают.

*Достаточность.* Пусть образы вершин  $v_a, v_b \in V_m$  характеризуются одинаковыми разбиениями

$$\Delta D_{m'}(v_a) = \Delta D_{m'}(v_b), \quad m' = m+1, m+2, \dots, M.$$

Если  $T \in M_T$ , в нем вершины каждого множества  $V_{m'j}$ ,  $m' \in \{m+1, m+2, \dots, M\}$ , упорядочены по невозрастанию полустепеней исхода. Этого достаточно, чтобы разбиения  $\Delta D_m(v_a)$  и  $\Delta D_m(v_b)$  однозначно определяли строение  $(m+1)$ -го яруса каждого из поддеревьев, разбиения  $\Delta D_{(m+1)}(v_a)$  и  $\Delta D_{(m+1)}(v_b)$  –  $(m+2)$ -го яруса и т.д. Таким образом, по разбиениям образцов вершины  $v_a(v_b)$  в последующих ярусах однозначно определяется  $v_a$ -поддерево ( $v_b$ -поддерево). Идентичными разбиениями определяется идентичность построенных деревьев, означающая, что отображения, описывающие одно из них, переходят в отображения, описывающие другое поддерево, при условии, что все индексы, означающие номера вершин одного, заменены соответствующими индексами другого. Идентичность отображений, описывающих оба поддерева, означает сохранение отношения инцидентностей при переходе от одного поддерева к другому. Следовательно,  $v_a$ - и  $v_b$ -поддеревья, построенные по разбиениям  $\Delta D_{m'}(v_a)$  и  $\Delta D_{m'}(v_b)$ , взаимозаменяемы. Тогда и вершины  $v_a, v_b$  в дереве  $T$ , как и в дереве  $T_{m'}$ , будут взаимозаменяемыми. Это значит, что они относятся к общему  $h$ -классу.

Утверждение 2.12 дает необходимый и достаточный признак изоморфности  $v_a$ - и  $v_b$ -поддеревьев, который по эффективности не хуже известного признака Эдмондса [29], заключающегося в совпадении кортежей, специально построенных для сравниваемых деревьев из степеней их вершин. Длина такого кортежа определяется числом вершин дерева. В первом приближении можно считать, что сформулированный в утверждении 2.12 признак изоморфности означает сопоставление тех же кортежей, только по частям. Эффект от такой проверки по частям состоит в том, что при отсутствии изоморфности нет необходимости находить и сопоставлять все составные части, какими являются разбиения отображений  $\Delta D_{m'}(v_a)$  и  $\Delta D_{m'}(v_b)$ ,  $m' = m+1, m+2, \dots, M$ .

### 2.5.3. Алгоритм разбиения множества вершин дерева на классы эквивалентности

В процессе структурного синтеза неизоморфных деревьев процедура разбиения множеств их вершин на  $h$ -классы и  $H$ -эквивалентные  $h$ -классы должна выполняться многократно. Поэтому целесообразно ее автоматизировать.

зировать. В приведенном далее алгоритме решения этой задачи вершины дерева  $v_j$ ,  $j \in \{1, 2, \dots, p\}$ , представлены их порядковыми номерами  $j$ .

Исходными данными для работы предлагаемого алгоритма разбиения является дерево  $T$ , представленное множеством выборок  $\mathbf{j}_i = (j_i', j_i'')$ ,  $i = 1, 2, \dots, q$ , соответствующих множеству его реберных пар  $(v_{j_i'}, v_{j_i''})$ .

В этом случае автоматизация вычисления образа вершины в последующих ярусах дерева не вызывает затруднений: эта задача сводится к поиску в списке выборок  $\mathbf{j}_i$ ,  $i = 1, 2, \dots, q$ , номеров смежных вершин по цепочке от  $j_i'$  к  $j_i''$ . Описание ее решения опускается в предлагаемом алгоритме, чтобы не затемнять его. Обоснованием алгоритма являются сведения о симметриях дерева из п. 2.5.2.

Алгоритм 2.3.

Шаг 1. Положить  $m = 0$ ,  $J_m = J_0 = (1)$ .

Шаг 2. Для элементов  $j \in J_m$  в порядке возрастания  $j$  построить множество отображений  $J_{(m+1)j} = j\alpha_j$ , объединив их в упорядоченное множество  $J_{(m+1)} = \mathbf{U} J_{(m+1)j}$ . Считая число элементов в каждом  $j$ -классе  $J_{(m+1)j}$ , построить множество  $D_m(v_j)$  полустепеней исхода вершин с номерами  $j \in J_m$ . Пометить первый элемент каждого  $j$ -класса  $J_{(m+1)j}$ . Если  $m = 0$ , идти к шагу 7.

Шаг 3. Разбить каждый  $j$ -класс с  $|J_{mj}| > 1$  на  $d$ -классы по полустепеням исхода его элементов: признак начала нового  $d$ -класса – изменение значения элемента  $D_m(v_j)$ . Пометить первый элемент каждого нового  $d$ -класса, если он не был помечен ранее как начальный элемент  $j$ -класса. Если  $m < 2$ , идти к шагу 7.

Шаг 4.  $m' = 1$ .

Шаг 5. Разбить каждый содержащий более одного элемента  $d$ -класс  $J_{m'}$ , полученный к этому моменту, по разбиению отображений его элементов в  $J_m$ . Для этого вычислить разбиение отображения  $\Delta D_m(v_j)$  каждого элемента  $j$  этого класса. Сравнить разбиение отображения  $\Delta D_m(v_j)$  каждого элемента  $v_j$  этого  $d$ -класса, кроме первого, с разбиением отображения  $\Delta D_m(v_{(j-1)})$  предыдущего элемента  $v_{(j-1)}$ ,  $v_j, v_{(j-1)} \in V_{m'}$ . Если разбиения



отображений  $\Delta D_m(v_j)$  и  $\Delta D_m(v_{(j-1)})$  отличаются, пометить  $j$  как начальный элемент нового  $d$ -класса.

Шаг 6.  $m' := m' + 1$ . Если  $m' < m$ , идти к шагу 5.

Шаг 7.  $m := m + 1$ . Если  $m \neq M$ , идти к шагу 2.

Шаг 8. Конец.

Рассмотрим на примере разбиение вершин дерева на классы эквивалентности по алгоритму 2.3.

**Пример 2.14.** Определим классы эквивалентности для дерева, приведенного на рис. 2.6.

Шаг 1.  $m = 0$ ,  $J_m = J_0 = (1)$ .

Шаг 2.  $J_1 = 1 \cdot \alpha_1 = (\underline{2}, 3, 4, 5, 6)$ ,  $D_0(v_1) = (5)$ .

Шаг 7.  $m = 1$ .

Шаг 2.  $J_2 = (\underline{7}, 8, 9, \underline{10}, 11, 12, \underline{13}, 14, \underline{15}, 16, \underline{17})$ ;  $D_1(v_2) = (3, 3, 2, 2, 1)$ .

Шаг 3.  $J_1 = (\underline{2}, 3, \underline{4}, 5, \underline{6})$ .

Шаг 7.  $m = 2$ .

Шаг 2.  $J_3 = (\underline{18}, 19, \underline{20}, 21, \underline{22}, \underline{23}, 24, \underline{25}, \underline{26}, \underline{27}, 28, \underline{29}, \underline{30}, 31, \underline{32}, \underline{33}, 34)$ .  $D_2(v_3) = (2, 2, 1, 2, 1, 1, 2, 1, 2, 1, 2)$ .

Шаг 3.  $J_2 = (\underline{7}, 8, \underline{9}, \underline{10}, \underline{11}, 12, \underline{13}, \underline{14}, \underline{15}, \underline{16}, \underline{17})$ .

Шаг 4.  $m' = 1$ .

Шаг 5.  $\Delta D_2(v_2) = (2, 2, 1)$ ;  $\Delta D_2(v_3) = (2, 1, 1)$ ;  $\Delta D_2(v_2) \neq \Delta D_2(v_3)$ ;  
 $J_1 = (\underline{2}, \underline{3}, \underline{4}, 5, \underline{6})$ ;  $\Delta D_2(v_4) = (2, 1)$ ;  $\Delta D_2(v_5) = (2, 1)$ ;  $\Delta D_2(v_4) = \Delta D_2(v_5)$ ;  
 $J_1 = (\underline{2}, \underline{3}, \underline{4}, 5, \underline{6})$ .

Шаг 6.  $m' = 2 = m$ .

Шаг 7.  $m = 3$ .

Шаг 2.  $J_4 = (\underline{35}, 36, 37, \underline{38}, 39, 40, \underline{41}, 42, \underline{43}, 44, \underline{45}, \underline{46}, 47, 48, \underline{49}, 50, \underline{51}, 52, \underline{53}, 54, \underline{55}, 56, \underline{57}, 58, \underline{59}, \underline{60}, 61, \underline{62}, \underline{63}, \underline{64}, 65, \underline{66})$ ;  $D_3 = (3, 3, 2, 2, 1, 3, 2, 2, 2, 2, 1, 2, 1, 1, 2, 1)$ .

Шаг 3.  $J_3 = (\underline{18}, 19, \underline{20}, 21, \underline{22}, \underline{23}, \underline{24}, \underline{25}, \underline{26}, \underline{27}, 28, \underline{29}, \underline{30}, \underline{31}, \underline{32}, \underline{33}, \underline{34})$ .

Шаг 4.  $m' = 1$ .

Шаг 5.  $D_3(v_4) = (2, 2, 1)$ ,  $D_3(v_5) = (2, 1, 1)$ ,  $D_3(v_4) \neq D_3(v_5)$ ;  $J_1 = (\underline{2}, \underline{3}, \underline{4}, \underline{5}, \underline{6})$ .

Шаг 6.  $m' := m' + 1 = 2$ .

Шаг 5.  $D_3(v_7) = (3, 3)$ ,  $D_3(v_8) = (2, 2)$ ,  $D_3(v_7) \neq D_3(v_8)$ ;  $J_2 = (\underline{7}, \underline{8}, \underline{9}, \underline{10}, \underline{11}, 12, \underline{13}, \underline{14}, \underline{15}, \underline{16}, \underline{17})$ ;  $D_3(v_{11}) = (2)$ ,  $D_3(v_{12}) = (2)$ ,  $D_3(v_{11}) = D_3(v_{12})$ ;  $J_2$  не меняется.

*Разбиение множества вершин дерева на  $h$ -классы  
и  $H$ -эквивалентные  $h$ -классы (к примеру 2.16)*

$J_0$	$J_1$	$J_2$	$J_3$	$J_4$
1	2	7	18	35
			19	36
				37
		20	38	
			39	
		21	40	
			41	
		22	42	
			43	
	3	10	8	44
				45
			9	46
		23	47	
			24	48
		25	49	
			50	
		26	51	
			52	
	27	53		
		54		
	4	13	11	55
				56
			12	57
		58		
14		59		
		60		
5	15	30	61	
			62	
		31	63	
	16	32	64	
		33	65	
6	17	34	66	

Шаг 6.  $m' = 3 = m$ .

Шаг 7.  $m = 4 = M$ .

Шаг 8. Конец.

Полученное по алгоритму 2.3 разбиение множества вершин дерева (рис. 2.6) по ярусам и по классам эквивалентности с указанием прообраза для каждого  $h$ -класса и их взаимосвязи представлены в виде таблицы, в которой столбцы соответствуют ярусам дерева, а строки – вершинам дерева. По ней нетрудно записать группу автоморфизмов дерева.

При решении задачи генерирования неизоморфных деревьев нужна не группа автоморфизмов, а информация о вершинах дерева  $T_m$ , которые не могут стать старше под действием его симметричных преобразований. Этому условию удовлетворяют вершины дерева, помеченные в результате работы алгоритма 2.3, кроме старших вершин не старших среди  $H$ -эквивалентных  $h$ -классов. Для получения указанной информации можно использовать вариант алгоритма 2.3 при следующей редакции описания шага 2:

Шаг 2. Для элементов  $j \in J_m$  в порядке возрастания  $j$  построить множество отображений  $J_{(m+1)j} = j\alpha_j$ , объединив их в упорядоченное множество  $J_{(m+1)} = \mathbf{U} J_{(m+1)j}$ . Считая число элементов в каждом  $j$ -классе  $J_{(m+1)j}$ , построить множество  $D_m(v_j)$  полустепеней исхода вершин с номерами  $j \in J_m$ . Если  $m \neq 0$ , среди следующих подряд  $j$ -классов  $J_{(m+1)j}$ , содержащих одинаковое число элементов, выделить те, прообразы  $j \in J_m$  которых относятся к общему  $d$ -классу, и пометить первый элемент старшего из этих  $j$ -классов (признак общего  $d$ -класса – общий помеченный элемент). Пометить старшие элементы всех прочих  $j$ -классов  $J_{(m+1)j}$ . Если  $m = 0$ , идти к шагу 7.

**Пример 2.15.** В примере 2.14 после такого изменения в алгоритме 2.3 получим следующую разметку множества  $J_4$ :

$J_4 = ( \underline{35}, 36, 37, 38, 39, 40, \underline{41}, 42, 43, 44, \underline{45}, \underline{46}, 47, 48, \underline{49}, 50, \underline{51}, 52, 53, 54, \underline{55}, 56, 57, 58, \underline{59}, \underline{60}, 61, \underline{62}, \underline{63}, \underline{64}, 65, \underline{66} )$ .

## 2.5.4. Алгоритм построения неизоморфных корневых деревьев

Пусть требуется построить множество неизоморфных корневых деревьев с  $p$  вершинами.

Поддерево, содержащее ребра с номерами  $1, 2, \dots, i, i < q$ , обозначим символом  $T^{(i)}$ , чтобы не путать с деревом  $T_m$ , состоящим из  $m$  ярусов. Бу-

дем строить дерево  $T^{(i)}$ , добавляя к дереву  $T^{(i-1)}$   $i$ -е ребро и рассматривая при этом только такие варианты его подсоединения, которые не приводятся симметриями  $T^{(i-1)}$  к уже построенным ранее вариантам. Для этого используем разбиение множества вершин дерева  $T^{(i-1)}$  на  $h$ -классы и  $H$ -эквивалентные  $h$ -классы, которое выполняется по алгоритму 2.3. Отнесем это разбиение к множеству  $J_{(i-1)}$  номеров вершин дерева.

**У т в е р ж д е н и е 2.13.** Пусть  $T^{(i-1)}$  – каноническое дерево. Для того чтобы дерево  $T^{(i)}$  также было каноническим, необходимо и достаточно, чтобы номер  $j_i' \geq j_{(i-1)}'$  вершины, из которой исходит добавленное  $i$ -е ребро, находился в старшем из  $H$ -эквивалентных  $h$ -классов и был старшим в нем.

**Д о к а з а т е л ь с т в о .** Обратимся к матричному представлению рассматриваемых деревьев. Матрица  $X_i$  является канонической, если ее невозможно увеличить никакими подстановками, не влияющими на матрицу  $X_{(i-1)}$ , т.е. автоморфизмами последней. Добавление  $i$ -го ребра к дереву  $T^{(i)}$ , исходящего из его  $j_i'$ -й вершины, интерпретируется пополнением матрицы  $X_{(i-1)}$   $i$ -й строкой с единицей в  $j_i'$ -м столбце и двойкой в  $j_i''$ -м столбце.

**Необходимость.** Доказываем от противного. Пусть матрица  $X_i$  является канонической, но при этом единица находится не на старшем месте  $h$ -класса ее вектора-строки  $x_i$  (не в старшем из  $H$ -эквивалентных  $h$ -классов). Ее можно перевести на старшее место (в старший  $h$ -класс) автоморфизмом матрицы  $X_{(i-1)}$ . Так как матрица  $X_{(i-1)}$  при этом не меняется, для матрицы  $X_i$  это увеличивающая подстановка, что противоречит предположению о каноничности этой матрицы.

**Достаточность.** Из утверждения 1.8 следует, что нахождение единицы на старшем месте старшего из  $H$ -эквивалентных  $h$ -классов является необходимым условием каноничности вектора  $x_i$ . Из утверждения 2.1 (см. п. 2.2.3) следует, что это не только необходимое, но и достаточное условие каноничности вектора  $x_i$ , так как на элементах каждого  $h$ -класса и каждого множества  $H$ -эквивалентных  $h$ -классов дерева действует симметрическая группа автоморфизмов матрицы  $X_{(i-1)}$ .

Таким образом, канонический вектор  $x_i$  не может быть увеличен лексикографически никакими подстановками его координат. Следовательно, он

не может быть увеличен никакими перестановками одних только столбцов матрицы  $\mathbf{X}_i$ . Поэтому увеличивающую подстановку для матрицы  $\mathbf{X}_i$  следует искать среди автоморфизмов матрицы  $\mathbf{X}_{(i-1)}$ , одновременно меняющих местами ее строки и столбцы.

Вершинам  $v_a$  и  $v_b$ ,  $a'b$ , дерева  $T^{(i-1)}$ , эквивалентным по группе  $\text{Aut}(\mathbf{X}_{(i-1)})$ , в матрице  $\mathbf{X}_{(i-1)}$  соответствуют столбцы с одинаковым количеством единиц в  $j'_a$ -м и  $j'_b$ -м столбцах. Перестановка этих столбцов вместе с перестановкой строк, содержащих единицы на пересечениях с указанными столбцами, матрицу  $\mathbf{X}_{(i-1)}$  не меняют. А для матрицы  $\mathbf{X}_i$  такая перестановка может стать увеличивающей только в случае, если при переходе к ней от матрицы  $\mathbf{X}_{(i-1)}$  количество единиц в  $j'_b$ -м столбце возрастет. Такое возрастание возможно только вследствие увеличения степени исхода  $d_b$  вершины  $v_b$  при переходе от дерева  $T^{(i-1)}$  к дереву  $T^{(i)}$ , т.е. в результате добавления ребер, исходящих из нее. Но по условию утверждения 2.13 такая ситуация исключается: вершина  $v_b$  не может быть старшей в своем  $h$ -классе или в старшем из  $H$ -эквивалентных  $h$ -классов, так как  $a'b$ . Таким образом, при выполнении условий утверждения 2.13 матрица  $\mathbf{X}_i$  и соответствующее дерево  $T^{(i)}$  являются каноническими.

Утверждение 2.13 является обоснованием приведенного ниже алгоритма генерирования неизоморфных деревьев.

Отберем из множества  $J_{(i-1)}$  только старшие элементы и только из старших  $h$ -классов среди множества  $H$ -эквивалентных. Обозначим упорядоченное множество номеров отобранных вершин символом  $J'_i$ . Из утверждения 2.13 следует, что каноническими будут только те деревья  $T^{(i)}$ , для которых  $J'_i$  выбрано из множества  $J'_i$ .

Список выборок  $\mathbf{j}_i = (j'_i, j''_i)$ , соответствующих реберным парам  $(v_{j'_i}, v_{j''_i})$ , которыми описывается каждое из искомым неизоморфных деревьев  $T$ , содержит  $q$  элементов. В соответствии со следствием 2.9.1 ограничимся рассмотрением только тех вариантов деревьев  $T$ , для которых выполняется соотношение  $j''_i = i + 1$ ,  $i = 1, 2, \dots, q$ . Учтем, что существует единственный вариант дерева  $T^{(1)}$ . Поэтому перебор начнем с построения канонических вариантов дерева  $T^{(2)}$ , задавшись значениями  $j'_1 = 1$ ,  $j''_1 = 2$

для дерева  $T^{(1)}$ . Тогда поставленная задача сводится к построению вариантов  $(q-1)$ -мерных последовательностей  $J' = (j_2', j_3', \dots, j_q')$ , которые не могут быть приведены одна к другой действием симметрий поддеревьев дерева  $T$ . Каждой такой последовательности соответствует каноническое дерево  $T$  с  $p = q + 1$  вершинами.

Итак, исходные данные для работы алгоритма генерирования неизоморфных корневых деревьев – это число вершин  $p$  и выборка  $\mathbf{j}' = (j_1', j_1'')$ , определяющая положение первого ребра в дереве  $T^{(1)}$ .

Алгоритм 2.4.

Шаг 1. Положить  $i = 2$ .

Шаг 2. По алгоритму 2.3 разбить множество  $J_{(i-1)}$  номеров вершин дерева  $T^{(i-1)}$  по его симметриям на  $h$ -классы и  $H$ -эквивалентные  $h$ -классы.

Шаг 3. Построить упорядоченное множество  $J_i'$  допустимых в соответствии с утверждением 2.13 значений  $j_i'$ .

Шаг 4. Если множество  $J_i'$  не пусто, выбрать его старший элемент в качестве  $j_i'$ , исключив его из списка  $J_i'$ , и положить  $j_i'' = i + 1$ . Иначе идти к шагу 7.

Шаг 5. Если  $i = q$ , зафиксировать неизоморфный вариант дерева  $T$ . Идти к шагу 4.

Шаг 6. Положить  $i := i + 1$ . Идти к шагу 2.

Шаг 7. Найти наибольшее значение  $i$ , для которого не исчерпан список  $J_i'$  допустимых вариантов  $j_i'$ , и с этим значением  $i$  идти к шагу 4. Если для каждого значения  $i$  исчерпан такой список, конец.

Работу алгоритма 2.4 проиллюстрируем следующим примером построения первых пяти канонических деревьев с  $p = 8$ .

**Пример 2.16.**

Шаг 1.  $i = 2$ .

Шаг 2.  $J_1 = (\underline{1}, \underline{2})$ .

Шаг 3.  $J_2' = (1, 2)$ .

Шаг 4.  $j_2' = 1$ ,  $J_2' = (2)$ ,  $j_1'' = i + 1 = 3$ .

Шаг 6.  $i = 3$ .

Шаг 2.  $J_2 = (\underline{1}, \underline{2}, 3)$ .

Шаг 3.  $J_3' = (1, 2)$ .

Шаг 4.  $j_3' = 1, J_3' = (2), j_1'' = i + 1 = 4$ .

Шаг 6.  $i = 4$ .

Шаг 2.  $J_3 = (\underline{1}, \underline{2}, 3, 4)$ .

Шаг 3.  $J_4' = (1, 2)$ .

Шаг 4.  $j_4' = 1, J_4' = (2), j_1'' = i + 1 = 5$ .

Шаг 6.  $i = 5$ .

Шаг 2.  $J_4 = (\underline{1}, \underline{2}, 3, 4, 5)$ .

Шаг 3.  $J_5' = (1, 2)$ .

Шаг 4.  $j_5' = 1, J_5' = (2), j_1'' = i + 1 = 6$ .

Шаг 6.  $i = 6$ .

Шаг 2.  $J_5 = (\underline{1}, \underline{2}, 3, 4, 5, 6)$ .

Шаг 3.  $J_6' = (1, 2)$ .

Шаг 4.  $j_6' = 1, J_6' = (2), j_1'' = i + 1 = 7$ .

Шаг 6.  $i = 7$ .

Шаг 2.  $J_6 = (\underline{1}, \underline{2}, 3, 4, 5, 6, 7)$ .

Шаг 3.  $J_7' = (1, 2)$ .

Шаг 4.  $j_7' = 1, J_7' = (2), j_1'' = i + 1 = 8$ .

Шаг 5. Дерево, соответствующее вектору  $J' = (1, 1, 1, 1, 1, 1)$ , показано на рис. 2.7, а.

Шаг 4.  $j_7' = 2, J_7' = \emptyset, j_1'' = i + 1 = 8$ .

Шаг 5. Дерево, соответствующее вектору  $J' = (1, 1, 1, 1, 1, 2)$ , показано на рис. 2.7, б.

Шаг 7.  $i = 6$ .

Шаг 4.  $j_6' = 2, J_6' = \emptyset, j_1'' = i + 1 = 7$ .

Шаг 6.  $i = 7$ .

Шаг 2.  $J_6 = (\underline{2}, \underline{3}, 4, 5, 6, \underline{7})$ .

Шаг 3.  $J_7' = (2, 3, 7)$ .

Шаг 4.  $j_7' = 2, J_7' = (3, 7), j_1'' = i + 1 = 8$ .

Шаг 5. Дерево, соответствующее вектору  $J' = (1, 1, 1, 1, 2, 2)$ , приведено на рис. 2.7, в.

Шаг 4.  $j_7' = 3, J_7' = (7), j_1'' = i + 1 = 8$ .

Шаг 5. Дерево, соответствующее вектору  $J' = (1, 1, 1, 1, 2, 3)$ , приведено на рис. 2.7, г.

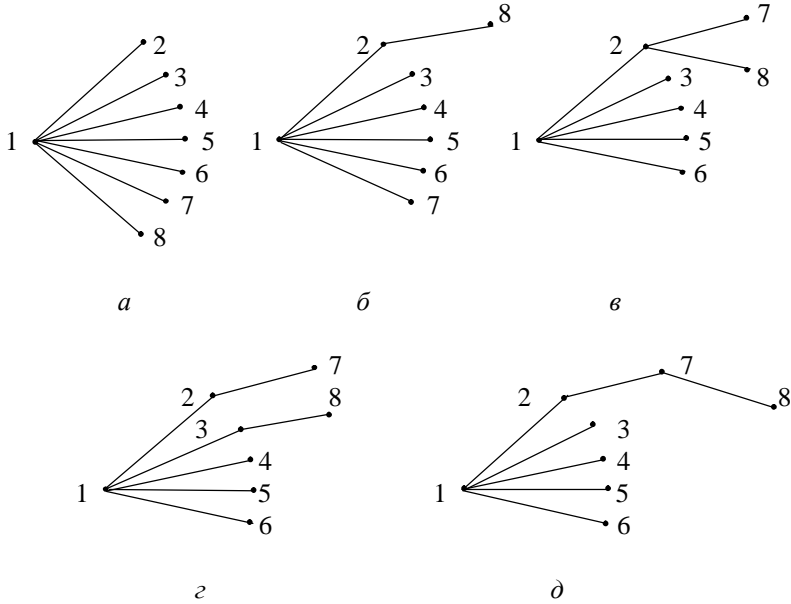


Рис. 2.7

Шаг 4.  $j_7' = 7$ ,  $J_7' = \emptyset$ ,  $j_1'' = i + 1 = 8$ .

Шаг 5. Дерево, соответствующее вектору  $J' = (1, 1, 1, 1, 2, 7)$ , приведено на рис. 2.7, д.

При выбранном способе упорядочения старшее дерево всегда имеет форму звезды с  $p$  лучами (см. рис. 2.7, а). Учитывая это, эффективность алгоритма 2.4 нетрудно повысить, опустив в нем построение старшего дерева.



СИММЕТРИИ И СИНТЕЗ ПРОГРАММ

**3.1. Симметрии  $n$ -мерного логического пространства**

Область определения булевых функций  $n$  переменных  $f(x_1, x_2, \mathbf{K}, x_n)$  можно рассматривать как совокупность  $2^n$   $n$ -мерных векторов  $(x_1, x_2, \dots, x_n)$ , координатами которых являются буквы 0 и 1 двоичного алфавита. Геометрическая интерпретация этой области – это множество вершин единичного  $n$ -мерного куба, образующее  $n$ -мерное логическое пространство. Каждой вершине соответствует конституента единицы, т.е. произведение

$$\tilde{x}_1 \tilde{x}_2 \mathbf{L} \tilde{x}_n,$$

где  $\tilde{x}_i$  – либо  $x_i$ , либо  $\bar{x}_i$ . Будем использовать обозначение конституент символом  $p_{d_1 d_2 \mathbf{K} d_n}$ , где  $d_i = 1$ , если  $\tilde{x}_i = x_i$ , и  $d_i = 0$  при  $\tilde{x}_i = \bar{x}_i$ ; одновременно это будет обозначением точки  $n$ -мерного логического пространства. Применяется и более компактное обозначение конституент единицы – при помощи десятичного эквивалента  $\alpha$  двоичного числа  $d_1 d_2 \mathbf{L} d_n$ :  $p_{d_1 d_2 \mathbf{K} d_n} = p_\alpha$ . Например,  $p_{101} = p_5$ ,  $p_{111} = p_7$ ,  $p_{001} = p_1$ .

Структура и свойства  $n$ -мерного логического пространства, определяющие структуру и свойства булевых функций, достаточно подробно описаны в работе [30]. Приведем ряд сведений из этой работы.

$n$ -Мерное логическое пространство является метрическим. В нем под расстоянием  $\Delta(p_\alpha, p_\beta)$  между двумя точками подразумевается число инверсирований аргументов, необходимое для перевода точки  $p_\alpha$  в точку  $p_\beta$ . Величина  $\Delta(p_\alpha, p_\beta)$ , действительно, удовлетворяет всем требованиям, оправдывающим название "расстояние":

- 1)  $\Delta(p_\alpha, p_\beta) = 0$ , если  $p_\alpha = p_\beta$  ( $\alpha = \beta$ );
- 2)  $\Delta(p_\alpha, p_\beta) = \Delta(p_\beta, p_\alpha)$  – свойство симметрии;
- 3)  $\Delta(p_\alpha, p_\beta) + \Delta(p_\beta, p_\gamma) \geq \Delta(p_\alpha, p_\gamma)$  – неравенство треугольника.

В работах [30; 31] и других изучены инварианты  $n$ -мерного логического пространства. Его группа автоморфизмов  $L_n$  порождена двумя подгруп-

пами – подгруппой  $G_n$  перестановок аргументов и подгруппой  $R_n$  инверсирований аргументов:

$$L_n = \{ G_n, R_n \}.$$

Перестановку аргументов  $x_1, x_2, \mathbf{K}, x_n$ , выражаемую подстановкой

$$\left( \begin{array}{c} x_1, x_2, \mathbf{K}, x_n \\ x_{i_1}, x_{i_2}, \mathbf{K}, x_{i_n} \end{array} \right),$$

будем обозначать символом  $S_{i_1, i_2, \mathbf{K}, i_n}$ . Вместо индексов  $i_1, i_2, \mathbf{K}, i_n$  (или  $i_1 i_2 \mathbf{K} i_n$ ) применяют также циклическое разложение подстановки

$$\left( \begin{array}{cccc} 1 & 2 & \mathbf{L} & n \\ i_1 & i_2 & \mathbf{L} & i_n \end{array} \right).$$

Инверсирование аргументов обозначим символом  $N_{c_1, c_2, \mathbf{K}, c_n}$ , где  $c_i = 1$ , если  $x_i$  инверсируется (заменяется на  $\bar{x}_i$ ), и  $c_i = 0$ , если  $x_i$  не инверсируется. В индексах  $i_1, i_2, \mathbf{K}, i_n$  и  $c_1, c_2, \mathbf{K}, c_n$  запятые опускаются, если нет опасности принять получаемые выражения  $i_1 i_2 \mathbf{K} i_n$  и  $c_1 c_2 \mathbf{K} c_n$  за произведения.

Так как символом  $N_{c_1 c_2 \mathbf{L} c_n}$  устанавливается взаимно однозначное соответствие между элементами группы  $R_n$  и  $n$ -разрядными двоичными числами  $c_1 c_2 \mathbf{K} c_n = \sum_{i=1}^n c_i \cdot 2^{(n-i)}$ , можно вместо  $N_{c_1 c_2 \mathbf{L} c_n}$  писать  $N_\alpha$ , где  $\alpha$  – десятичный эквивалент двоичного числа  $c_1 c_2 \mathbf{K} c_n$ .

Пусть  $c_1 c_2 \mathbf{K} c_n$  и  $d_1 d_2 \mathbf{L} d_n$  – произвольные  $n$ -разрядные двоичные числа. Тогда  $N_{c_1 c_2 \mathbf{L} c_n} N_{d_1 d_2 \mathbf{L} d_n} = N_{(c_1 \oplus d_1)(c_2 \oplus d_2) \mathbf{L} (c_n \oplus d_n)}$ , где  $c_i \oplus d_i = c_i \bar{d}_i \vee \bar{c}_i d_i$ . Из коммутативности булевой операции  $\oplus$  следует, что группа  $R_n$  – абелева.

Положим по определению

$$c_1 c_2 \mathbf{K} c_n \oplus d_1 d_2 \mathbf{L} d_n = (c_1 \oplus d_1)(c_2 \oplus d_2) \mathbf{L} (c_n \oplus d_n),$$

и условимся называть выражение в левой части равенства поразрядной суммой двоичных чисел  $c_1 c_2 \mathbf{K} c_n$  и  $d_1 d_2 \mathbf{L} d_n$ . Тогда

$$N_{c_1 c_2 \mathbf{L} c_n} N_{d_1 d_2 \mathbf{L} d_n} = N_{c_1 c_2 \mathbf{L} c_n \oplus d_1 d_2 \mathbf{L} d_n},$$

откуда вытекает, что группа  $R_n$  изоморфна группе, образованной  $n$ -разрядными двоичными числами относительно операции поразрядного сложения. Поэтому результаты, полученные для последней группы, например в [32], непосредственно применимы и к  $R_n$ .

Пусть  $e$  – тождественное преобразование. Тогда  $N_{00\mathbf{L}0} = e$  и

$$N_{c_1 c_2 \mathbf{L} c_n} N_{c_1 c_2 \mathbf{L} c_n} = N_{(c_1 \oplus c_1)(c_2 \oplus c_2) \mathbf{L} (c_n \oplus c_n)} = N_{00\mathbf{L}0} = e,$$

т. е. каждое преобразование  $N_{c_1 c_2 \mathbf{L} c_n}$  имеет порядок 2. Иными словами, каждое преобразование  $N_{c_1 c_2 \mathbf{L} c_n}$  обратно самому себе:

$$(N_{c_1 c_2 \mathbf{L} c_n})^{-1} = N_{c_1 c_2 \mathbf{L} c_n}.$$

Группа  $R_n$  имеет неприводимую систему образующих из элементов

$$N_1, N_2, \mathbf{L}, N_{2^{(i-1)}}, \mathbf{L}, N_{2^{(n-1)}},$$

где  $1, 2, \dots, 2^{(i-1)}, \dots, 2^{(n-1)}$  – десятичные эквиваленты двоичных чисел, а именно числу  $2^{(n-i)}$  отвечает двоичное число  $c_1 c_2 \mathbf{K} c_n$ , в котором  $c_i = 1$ ,  $c_1 = \mathbf{K} = c_{(i-1)} = c_{(i+1)} = \mathbf{K} = c_n = 0$ .

Таким образом, подгруппа  $R_n$  – периодическая абелева группа порядка  $2^n$ , где каждый элемент имеет порядок 2.

Подгруппа  $G_n$  – не абелева. Для нее справедливы соотношения

$$S_{k_1 k_2 \mathbf{K} k_n} S_{i_1 i_2 \mathbf{K} i_n} = S_{t_1 t_2 \mathbf{K} t_n};$$

$$(S_{i_1 i_2 \mathbf{K} i_n})^{-1} = S_{j_1 j_2 \mathbf{K} j_n},$$

где

$$\begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ t_1 & t_2 & \mathbf{L} & t_n \end{pmatrix} = \begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ k_1 & k_2 & \mathbf{L} & k_n \end{pmatrix} \begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ i_1 & i_2 & \mathbf{L} & i_n \end{pmatrix} = \begin{pmatrix} i_1 & i_2 & \mathbf{L} & i_n \\ t_1 & t_2 & \mathbf{L} & t_n \end{pmatrix} \begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ i_1 & i_2 & \mathbf{L} & i_n \end{pmatrix};$$

$$\begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ j_1 & j_2 & \mathbf{L} & j_n \end{pmatrix} = \begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ i_1 & i_2 & \mathbf{L} & i_n \end{pmatrix}^{-1} = \begin{pmatrix} i_1 & i_2 & \mathbf{L} & i_n \\ 1 & 2 & \mathbf{L} & n \end{pmatrix}.$$

Как видим, в цитируемой работе [30] принято считать, что произведение

$$\begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ k_1 & k_2 & \mathbf{L} & k_n \end{pmatrix} \begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ i_1 & i_2 & \mathbf{L} & i_n \end{pmatrix}$$

означает сначала  $\begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ i_1 & i_2 & \mathbf{L} & i_n \end{pmatrix}$ , потом  $\begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ k_1 & k_2 & \mathbf{L} & k_n \end{pmatrix}$ .

Группа  $R_n$  не является циклической для  $n > 2$ , группа  $G_n$  – для  $n > 3$ . Система образующих группы  $G_n$  состоит из двух элементов [21; 30]:

$$S_{213} \mathbf{K}_n = S_{(1,2)}; S_{23} \mathbf{K}_{n1} = S_{(1,2,\mathbf{L},n)}.$$

Группа  $G_n$  изоморфна симметрической группе степени  $n$ . Ее порядок равен  $n!$ . Единичный элемент группы  $G_n$  – тождественная подстановка

$$e = S_{12\mathbf{L}n}.$$

Тождественное преобразование  $e = N_{00\mathbf{L}0} = S_{12\mathbf{L}n}$  является единственным общим элементом групп  $G_n, R_n$ . Группы  $G_n, R_n$  перестановочны. Более того, каждый элемент группы  $G_n$  перестановочен с группой  $R_n$ :

$$N_{c_1 c_2 \mathbf{L} c_n} S_{i_1 i_2 \mathbf{K} i_n} = S_{i_1 i_2 \mathbf{K} i_n} N_{c_1 c_2 \mathbf{L} c_n}; \quad (3.1)$$

$$S_{i_1 i_2 \mathbf{K} i_n} N_{c_1 c_2 \mathbf{L} c_n} = N_{c_{j_1} c_{j_2} \mathbf{L} c_{j_n}} S_{i_1 i_2 \mathbf{K} i_n},$$

где  $c_{j_1} c_{j_2} \mathbf{K} c_{j_n}$  получаем из преобразования  $c_{i_1} c_{i_2} \mathbf{K} c_{i_n}$  с помощью подстановки  $S_{i_1 i_2 \mathbf{K} i_n}$ :

$$\begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ j_1 & j_2 & \mathbf{L} & j_n \end{pmatrix} = \begin{pmatrix} 1 & 2 & \mathbf{L} & n \\ i_1 & i_2 & \mathbf{L} & i_n \end{pmatrix}^{-1} = \begin{pmatrix} i_1 & i_2 & \mathbf{L} & i_n \\ 1 & 2 & \mathbf{L} & n \end{pmatrix}.$$

Согласно равенству (3.1) всякий элемент из  $L_n$ , например какой-нибудь элемент вида  $S_\delta N_\varepsilon S_\theta N_\eta S_\omega$ , можно представить как  $N_\alpha S_\beta = S_\beta N_\gamma$ . Поскольку группы  $G_n, R_n$  имеют только один общий элемент – тождественное преобразование, такое представление единственно. Иными словами, группа  $L_n$  является произведением своих подгрупп  $G_n, R_n$ , и ее порядок  $n! 2^n$ . Символически

$$L_n = G_n R_n = R_n G_n. \quad (3.2)$$

Из равенства (3.1) следует, что  $R_n$  – нормальный делитель группы  $L_n$ .

Справедливо равенство

$$N_{11\mathbf{L}1} S_{i_1 i_2 \mathbf{K} i_n} = S_{i_1 i_2 \mathbf{K} i_n} N_{11\mathbf{L}1},$$

означающее, что преобразование  $N_{1|L_1} = N_{(2^n-1)}$  перестановочно с группой  $G_n$ . Других нетождественных преобразований  $N_{c_1 c_2 L c_n}$ , перестановочных с группой  $G_n$ , не существует.

Как известно [21; 22], подгруппа  $H$  группы  $G$ , состоящая из всех элементов группы  $G$ , перестановочных с подгруппой  $H$  группы  $G$ , называется нормализатором подгруппы  $H$  в группе  $G$ . Из сказанного следует, что нормализатором подгруппы  $R_n$  в группе  $L_n$  является вся группа  $L_n$ , а нормализатором подгруппы  $G_n$  в группе  $L_n$  – подгруппа  $\{N_{1|L_1}\}G_n = \{N_{(2^n-1)}\}G_n$ , т.е. произведение подгруппы  $G_n$  на подгруппу  $\{N_{1|L_1}\}$ , порожденную преобразованием  $N_{1|L_1}$  и потому состоящую из двух элементов:  $N_{1|L_1}$ ,  $e = N_{1|L_1}^2$ .

Произвольные перестановки точек  $p_{c_1, c_2, \mathbf{K}, c_n}$   $n$ -мерного булевого пространства, т.е. точечные преобразования, образуют группу, изоморфную симметрической группе  $G$  степени  $2^n$ . Преобразование  $S_{i_1 i_2 \mathbf{K} i_n} N_{c_1 c_2 L c_n}$  меняет местами точки  $p_{c_1, c_2, \mathbf{K}, c_n}$  и  $p_{d_1, d_2, \mathbf{K}, d_n}$  конstituенты единицы:

$$S_{i_1 i_2 \mathbf{K} i_n} N_{c_1 c_2 L c_n} p_{d_1, d_2, \mathbf{K}, d_n} = \\ = S_{i_1 i_2 \mathbf{K} i_n} p_{c_1 \oplus d_1, c_2 \oplus d_2, \mathbf{K}, c_n \oplus d_n} = p_{c_1 \oplus d_1, c_2 \oplus d_2, \mathbf{K}, c_n \oplus d_n}.$$

Таким образом, группа  $L_n$  может быть вложена в группу точечных преобразований алгебры булевых функций  $n$  переменных. Группа  $L_n$  является подгруппой группы  $G_{2^n}$ , сохраняющей метрику  $n$ -мерного булевого пространства. Следовательно, это подгруппа изометрических преобразований  $n$ -мерного булевого пространства.

$n$ -Мерное булево пространство изоморфно  $n$ -мерному единичному гиперкубу в евклидовом пространстве. Его можно также отождествить с  $n$ -мерным гипероктаэдром – гипертелом, двойственным гиперкубу и вписываемым в него. Переход от гиперкуба к гипероктаэдру соответствует дуальному автоморфизму алгебры булевых функций и может быть уподоблен переходу от конstituент единицы к конstituентам нуля. "Булево" расстояние  $\Delta(p_{c_1, c_2, \mathbf{K}, c_n}, p_{d_1, d_2, \mathbf{K}, d_n})$  можно истолковать как число ребер гиперкуба, по которым нужно пройти, чтобы попасть из вершины  $p_{c_1, c_2, \mathbf{K}, c_n}$

в вершину  $p_{d_1, d_2, \mathbf{K}, d_n}$ . Так как группа  $L_n$  есть группа изометрических преобразований  $n$ -мерного логического пространства и последнее изоморфно  $n$ -мерному единичному гиперкубу в евклидовом пространстве, группа  $L_n$  изоморфна группе всех изометрических преобразований  $n$ -мерного евклидова пространства, переводящих этот гиперкуб в себя. В табл. 3.1 сопоставлены эти группы для случая  $n = 2$ , когда они невелики и поэтому легко обозримы.

Группу, переводящую гиперкуб в себя, принято называть группой симметрий гиперкуба. Группу  $L_n$ , изоморфную группе симметрий гиперкуба, естественно также называть группой симметрий  $n$ -мерного логического пространства. А изометрическое отображение  $n$ -мерного логического пространства на себя с помощью оператора из группы его автоморфизмов  $L_n$  условимся называть симметричным преобразованием  $n$ -мерного логического пространства.

Преобразования  $S_{i_1 i_2 \mathbf{K} i_n} \in G_n$  оставляют неподвижными точки  $p_0 = \bar{x}_1 \bar{x}_2 \mathbf{K} \bar{x}_n$  и  $p_{(2^n - 1)} = x_1 x_2 \mathbf{K} x_n$ . Этим преобразованиям отвечают вращения гиперкуба вокруг оси, проходящей через вершины  $(-1/2, -1/2, \mathbf{K} \mathbf{K}, -1/2)$ ,  $(1/2, 1/2, \mathbf{K}, 1/2)$ , и зеркальные отображения относительно гиперплоскостей, проходящих через эту ось. Преобразования  $N_{c_1 c_2 \mathbf{L} c_n} \in R_n$ , где  $c_1 = c_2 = \mathbf{K} = c_k = 1$ , а все прочие  $c_i$  равны 0, представляют собой зеркальные отображения гиперкуба относительно  $(n - k)$ -мерной координатной гиперплоскости; в частности,  $N_{(2^n - 1)} \in R_n$  есть зеркальное отражение относительно центра гиперкуба.

Можно рассматривать  $n$ -мерное логическое пространство как геометрическую интерпретацию одного важного случая булевой функции  $n$  переменных – тождественной единицы. Переобозначение переменных как с инверсированием, так и без него сопровождается перестановкой слагаемых этой функции и сомножителей в ее слагаемых. В силу коммутативности операций логического сложения и логического умножения такие перестановки не меняют функцию. Следовательно, можно считать, что природа симметрий указанной функции ( $n$ -мерного логического пространства) заключается в коммутативности операций  $\wedge$  и  $\vee$  над булевыми переменными, обуславливающей взаимозаменяемость как сомножителей внутри слагаемого, так и слагаемых функции.

Таблица 3.1

Изоморфизм группы симметрий 2-мерного булевого пространства и единичного гиперкуба

Изоморфные системы	Геометрический гиперфигуры	Вращение				Зеркальное отражение относительно осей			
		на $0^\circ$	на $180^\circ$	на $90^\circ$	на $270^\circ$	$\delta$	$\epsilon$	$\zeta$	$\eta$
Единичный гиперкуб	<p>Diagram of a unit hypercube (square) with vertices labeled A, B, C, D and axes labeled f, g, h, h<sub>0</sub>.</p>	$\begin{pmatrix} \text{дос} \\ \text{дос} \end{pmatrix}$	$\begin{pmatrix} \text{дос} \\ \text{сбас} \end{pmatrix}$	$\begin{pmatrix} \text{дос} \\ \text{дос} \end{pmatrix}$	$\begin{pmatrix} \text{дос} \\ \text{сбас} \end{pmatrix}$	$\begin{pmatrix} \text{дос} \\ \text{дос} \end{pmatrix}$	$\begin{pmatrix} \text{дос} \\ \text{сбас} \end{pmatrix}$	$\begin{pmatrix} \text{дос} \\ \text{дос} \end{pmatrix}$	$\begin{pmatrix} \text{дос} \\ \text{сбас} \end{pmatrix}$
2-Мерное булево пространство	<p>Diagram of a unit hypercube (square) with vertices labeled 00, 01, 10, 11.</p>	$\xi$	$N_{11}$	$N_{01}S_{21}$	$N_{10}S_{21}$	$S_{21}$	$N_{11}S_{21}$	$N_{01}$	$N_{10}$

### 3.2. Структура множества булевых функций

Известно, что множество булевых функций  $n$ -переменных насчитывает  $2^{2^n}$  элементов.

Геометрически булева функция  $n$  переменных в  $n$ -мерном логическом пространстве отображается множеством вершин  $n$ -мерного единичного куба, соответствующих ее конституентам единицы (помеченных вершин [33]).

Рассмотрим изменение функции  $f(x_1, x_2, \mathbf{K}, x_n)$  под действием  $T \in L_n$ , т. е. в результате переименования ее аргументов.

**У т в е р ж д е н и е 3. 1.** *Две функции  $f(x_1, x_2, \mathbf{K}, x_n)$  и  $g(x'_1, x'_2, \mathbf{K}, x'_n)$  являются изоморфными, если одна переходит в другую под действием переименования  $T \in L_n$ .*

**Д о к а з а т е л ь с т в о .** Будем рассматривать булеву функцию  $n$  переменных как систему из  $n$  объектов  $x_1, x_2, \mathbf{K}, x_n$  с заданным в них отношением  $f(x_1, x_2, \mathbf{K}, x_n)$ . Две системы объектов  $X = \{x_1, x_2, \mathbf{K}, x_n\}$  и  $X' = \{x'_1, x'_2, \mathbf{K}, x'_n\}$  с указанными в них отношениями называются изоморфными, если их можно поставить в такое взаимно однозначное соответствие

$$X' = \varphi(X), X = \psi(X'),$$

что из наличия отношения  $f(x_1, x_2, \mathbf{K}, x_n)$  следует отношение  $g(x'_1, x'_2, \mathbf{K}, x'_n)$  и наоборот. Для булевых функций  $n$  переменных такое соответствие задается переименованием  $T \in L_n$  переменных  $x_i, x'_i \in X$  [30]:

$$g(x'_1, x'_2, \mathbf{K}, x'_n) = f(T(x_1), T(x_2), \mathbf{K}, T(x_n)) = Tf(x_1, x_2, \mathbf{K}, x_n);$$

$$f(x_1, x_2, \mathbf{K}, x_n) = g(T^{-1}(x'_1), T^{-1}(x'_2), \mathbf{K}, T^{-1}(x'_n)) = T^{-1}g(x'_1, x'_2, \mathbf{K}, x'_n).$$

Значит, функции  $f(x_1, x_2, \mathbf{K}, x_n)$  и  $g(x'_1, x'_2, \mathbf{K}, x'_n)$  изоморфны.

Изоморфизм (однотипность [30]) булевых функций означает тождественность их строения, инвариантность структуры функций к переименованию ее аргументов. Как видим, изоморфизм булевых функций порожден симметриями  $n$ -мерного булевого пространства. Из групповых свойств симметричных преобразований этого пространства следует, что изоморфные преобразования булевых функций – это взаимно однозначное отображение множества булевых функций на себя. Области транзитивности группы автоморфизмов  $n$ -мерного булевого пространства – это классы изоморфных функций. Следовательно, множество булевых функций  $n$  переменных рас-



падает на попарно непересекающиеся классы – множества изоморфных функций. Каждую функцию данного класса можно выбрать в качестве представителя этого класса.

Под изоморфностью функций подразумевается не только подобие в строении, но и общие свойства членов класса. Например, все функции одного класса характеризуются одинаковым числом конститuent единицы, называемым рангом. Поэтому нужно говорить о ранге класса изоморфных функций: равенство рангов – необходимое условие изоморфности функций.

Операция  $\varphi(f, g, \mathbf{K}, h)$  над функциями  $f, g, \mathbf{K}, h$  называется инвариантной относительно группы  $L_n$ , если при всех  $T \in L_n$  и всех  $f, g, \mathbf{K}, h$  равенство  $F = \varphi(f, g, \mathbf{K}, h)$  влечет  $TF = \varphi(Tf, Tg, \mathbf{K}, Th)$ . Нетрудно убедиться, что операции  $f \vee g, f \wedge g, f \oplus g, \bar{f}$  инвариантны относительно группы  $L_n$ . Таким образом, для любых  $T \in L_n$  и любых булевых функций  $n$  переменных справедливы тождества

$$T(f \vee g) = Tf \vee Tg; T(f \wedge g) = Tf \wedge Tg; T\bar{f} = \overline{Tf}; T(f \oplus g) = Tf \oplus Tg;$$

Булева функция, изоморфная своему отрицанию, называется антивариантной. К антивариантным относятся все самодвойственные функции. Ясно, что для антивариантной функции все изоморфные функции являются антивариантными, т.е. функции  $f(x_1, x_2, \mathbf{K}, x_n)$ , для которых  $\bar{f}(x_1, x_2, \mathbf{K}, x_n) = f(\bar{x}_1, \bar{x}_2, \mathbf{K}, \bar{x}_n)$ .

Отношением эквивалентности (изоморфизма) между функциями характеризуется степень однородности их пространства, подобно тому, как группа движений характеризует однородность метрического евклидова пространства, а группа преобразований Лоренца – однородность четырехмерного мира теории относительности. Систему наименований  $n$  переменных, включая их положительные и отрицательные представления, можно называть "системой координат" ("системой отсчета") в пространстве булевых функций  $n$  переменных. Если в этом смысле выбрать для всех функций  $n$  переменных одну и ту же "систему координат", изоморфные функции будут аналогами геометрических фигур одинаковой формы, занимающих различные положения относительно одной и той же системы геометрических координат. Возможность представления булевых функций  $n$  переменных множеством точек  $p_{c_1, c_2, \mathbf{K}, c_n}$  вполне гармонирует со сравнением между изоморфными функциями и геометрическими фигурами одинаковой формы. При заданной "системе координат", т. е. определенном наименовании аргументов, можно считать, что группа  $L_n$  описывает преобразование "простран-

ства" булевых функций, не нарушающих их строения, и поэтому может рассматриваться как группа автоморфизмов этого пространства.

Вместе с тем изоморфные функции можно рассматривать как запись одной и той же логической формы в разных "системах координат". Изоморфному преобразованию булевой функции  $n$  переменных соответствует перестановка координатных осей ( $S_{i_1 i_2} \mathbf{K}_{i_n} \in G_n$ ) или изменение их ориентации ( $N_{c_1 c_2} \mathbf{L}_{c_n} \in R_n$ ). В силу относительности движения можно считать, что движется гиперкуб, а "система координат" ("система отсчета") неподвижна. Следовательно, группу  $L_n$  можно рассматривать как группу преобразований координат в пространстве булевых функций  $n$  переменных, которые характеризуют относительность производимых в нем наблюдений.

Изоморфные булевы функции отличаются только наименованием аргументов, а потому описывают в исчислении высказываний по существу одну и ту же логическую форму сложных высказываний и могут рассматриваться как разные формы записи одной логической функции. Действительно, что такое перестановка аргументов, как не их переименование? Кроме того, в классической логике благодаря закону двойного отрицания ( $x = \bar{\bar{x}}$  \*) имеет место симметрия утверждения и отрицания как для высказываний (суждений), так и для понятий. Не изменяя критерия истинности или ложности, любое высказывание можно представить в виде либо утверждения, либо отрицания ("идет снег" = "неправда, что не идет снег"). Не изменяя и содержания и объема понятий, можно определить его и через положительный либо отрицательный термин (металлоид = неметалл). Поэтому замена  $x$  на  $\bar{x}$  – не что иное, как переименование взаимозаменяемых  $x$  на  $\bar{x}$ . Таким образом, множество изоморфных функций можно рассматривать как множество различных записей (формул) одной и той же функции.

### 3.3. Симметрии булевых функций

Из рефлексивности отношения изоморфности следует, что каждая булева функция переходит в себя хотя бы при одном  $T \in L_n$ . В крайнем случае это тождественное преобразование  $e$ . Однако возможны и нетривиальные случаи. Изоморфизм на себя образует подгруппу  $H_n$  группы  $L_n$  – группу автоморфизмов (инерции [30]) булевой функции. Следовательно, для  $h \in H_n$

$$hf(x_1, x_2, \mathbf{K}, x_n) = f(x_1, x_2, \mathbf{K}, x_n). \quad (3.3)$$

Назовем функции  $f(x_1, x_2, \mathbf{K}, x_n)$ , для которых группа  $H_n$  состоит не только из одного элемента  $\{e\}$ , симметричными. Остальные булевы функции называются [30] обыкновенными.

**Пример 3.1.** Для функции двух переменных  $f = x \vee y$  группа автоморфизмов  $H_2 = \{e, S_{21}\}$ , она совпадает с  $G_2$ ; для функции  $f = xy \vee \overline{xy}$  группа автоморфизмов  $H_2 = \{e, N_{11}, S_{21}\}$ . Следовательно, обе функции симметричны.

Рассмотрим геометрическую интерпретацию группы автоморфизмов булевой функции.

Заданием функции  $f(x_1, x_2, \mathbf{K}, x_n)$  множество точек  $P_{c_1, c_2, \mathbf{K}, c_n}$   $n$ -мерного булевого пространства делится на два подмножества: к одному относятся точки, покрытые функцией, к другому – не покрытые. При этом нарушается взаимозаменяемость всех точек  $n$ -мерного булевого пространства. Группу автоморфизмов  $H_n$  функции  $f(x_1, x_2, \mathbf{K}, x_n)$  составляют те автоморфизмы  $n$ -мерного логического пространства, которые осуществляют изометрическое отображение множества помеченных точек на себя. Если группа автоморфизмов  $L_n$   $n$ -мерного булевого пространства отображает симметрии этого пространства, то группа  $H_n$  отображает симметрии булевой функции.

Если изоморфное преобразование функции с помощью  $T \in L_n$  рассматривать как преобразование "логических координат", то симметричные булевы функции можно сравнить с фигурами, которые выглядят одинаково в различных "системах отсчета" (или с фигурами, которые не меняются в результате их преобразования с помощью  $T \in L_n$ ).

Группы автоморфизмов изоморфных функций изоморфны, и их порядки совпадают. Отсюда, если булева функция симметрична, то симметричны и изоморфные ей функции, а если функция обыкновенна, обыкновенны и изоморфные функции. Очевидно, что каждый класс обыкновенных изоморфных функций содержит ровно  $n! 2^n$  элементов, а всякий класс симметричных изоморфных функций содержит заведомо меньше, чем  $n! 2^n$ , элементов.

Структура множества булевых функций, в частности связь между изоморфными функциями, определяется взаимосвязью между симметриями булевых функций и симметриями  $n$ -мерного логического пространства.

Если разложить группу  $L_n$  на левосторонние смежные классы по  $H_n$ , каждому классу  $UH_n$ , где  $U \in L_n$ , будет соответствовать одна и только одна Функция  $g(x_1, x_2, \mathbf{K}, x_n)$ , изоморфная функции  $f(x_1, x_2, \mathbf{K}, x_n)$ , а именно

$$g(x_1, x_2, \mathbf{K}, x_n) = Uh f(x_1, x_2, \mathbf{K}, x_n) = Uf(x_1, x_2, \mathbf{K}, x_n),$$

где  $h$  – любой элемент группы  $H_n$ . Таким образом, каждый левосторонний смежный класс  $UH_n$  – это множество элементов  $T \in L_n$ , переводящих функцию  $f(x_1, x_2, \mathbf{K}, x_n)$  в изоморфную функцию  $g(x_1, x_2, \mathbf{K}, x_n)$ .

**Пример 3.2.** В табл. 3.2 приведено левостороннее разложение группы  $L_2$  по  $H_2 = G_2$ . В ней для каждого смежного класса указана функция, полученная из функции  $f = x \vee y$  посредством преобразования ее с помощью элементов из этого класса.

Т а б л и ц а 3.2

Разложение группы  $L_2$  по подгруппе  $G_2$

Смежные классы	Булевы функции
$e, S_{21}$	$x \vee y$
$N_{01}, N_{01} S_{21}$	$x \vee \bar{y}$
$N_{10}, N_{10} S_{21}$	$\bar{x} \vee y$
$N_{11}, N_{11} S_{21}$	$\bar{x} \vee \bar{y}$

Число изоморфных функций, равное индексу  $j$  группы  $H_n$  в группе  $L_n$ , связано с порядком  $k$  группы  $H_n$  и порядком группы  $L_n$ , равным  $n! 2^n$ , в соответствии с теоремой Лагранжа [21; 22] следующим соотношением:

$$jk = n! 2^n.$$

**Пример 3.3.** Для функции  $f = x \vee y$  (см. табл. 3.2) порядок группы  $H_2 = G_2$  равен 2, индекс – 4, так что произведение порядка на индекс равно 8, т. е. равно порядку всей группы  $L_n = L_2$ .

Если трансформировать элементы  $h \in H_n$  элементами  $T \in L_n$  из левостороннего смежного класса, соответствующего функции  $g(x_1, x_2, \mathbf{K}, x_n)$ , получим множество элементов

$$b = ThT^{-1},$$

составляющих подгруппу, сопряженную с  $H_n$ . Эта подгруппа есть группа автоморфизмов функции  $g(x_1, x_2, \mathbf{K}, x_n)$  и, следовательно, отображает ее симметрии. Аналогично строятся все сопряженные с  $H_n$  подгруппы. Мно-

жеству всех изоморфных между собой функций отвечает класс сопряженных подгрупп группы  $L_n$ . Каждая из них является группой автоморфизмов одной из функций  $g(x_1, x_2, \mathbf{K}, x_n)$ , изоморфных  $f(x_1, x_2, \mathbf{K}, x_n)$ , и отображает ее симметрии.

Согласно [21] число подгрупп, сопряженных с подгруппой  $H_n$  группы  $L_n$ , равно индексу нормализатора подгруппы  $H_n$  в  $L_n$ . Может оказаться, что индекс этот меньше мощности класса изоморфных функций. Тогда сопряженные группы не только изоморфны, но и совпадают. В частности, все изоморфные функции имеют одну группу автоморфизмов тогда и только тогда, когда эта группа является нормальным делителем  $L_n$ . Таков, например, случай обыкновенных функций: все они имеют общую группу автоморфизмов  $\{e\}$ . Один и тот же класс сопряженных подгрупп может отображать симметрии нескольких изоморфных функций. В этом случае функции называются изомерными.

**Пример 3.4.** Сопряженные подгруппы (табл. 3.3) получены из группы  $G_2$  (см. пример 3.2) посредством трансформирования элементами левосторонних смежных классов группы  $L_2$  по  $G_2$  из табл. 3.2. Эти сопряженные подгруппы являются

Т а б л и ц а 3.3

*Подгруппы, сопряженные с  $G_2$*

Сопряженные подгруппы	Булевы функции
$e, S_{21}$	$x \vee y$
$e, N_{11} S_{21}$	$x \vee \bar{y}$
$e, N_{11} S_{21}$	$\bar{x} \vee y$
$e, S_{21}$	$\bar{x} \vee \bar{y}$

группами автоморфизмов изоморфных функций, представленных в правой части табл. 3.3. Число изоморфных функций равно 4. Если перебрать все 8 элементов группы  $L_2 = \{e, S_{21}, N_{10}, N_{11}, N_{01}, N_{11}S_{21}, N_{01}S_{21}, N_{10}S_{21}\}$  функции  $f(x_1, x_2) = x \vee y$ , легко убедиться, что с группой  $G_2$  перестановочны лишь преобразования  $e, S_{21}, N_{11}, N_{11}S_{21}$ . Эти элементы образуют нормализатор группы  $G_2$ . Его порядок равен 4. Отсюда по теореме Лагранжа его индекс равен  $8/4 = 2$ , что дает число сопряженных подгрупп в классе, соответствующем множеству функций, изоморфных функции  $f = x \vee y$ . Из табл. 3.3 видно, что два элемента, из которых состоит

класс сопряженных с  $G_2$  подгрупп группы  $L_2$ , – это подгруппы  $\{e, S_{21}\}$  и  $\{e, N_{11}S_{21}\}$ .

Ввиду инвариантности операции отрицания относительно  $L_n$  изоморфность  $f(x_1, x_2, \mathbf{K}, x_n)$  и  $g(x_1, x_2, \mathbf{K}, x_n)$  влечет изоморфность  $\bar{f}$  и  $\bar{g}$ . В самом деле, если  $g = Tf$ , где  $T \in L_n$ , то и  $\bar{g} = T\bar{f} = \overline{Tf}$ , и обратно. Следовательно, функции  $f, \bar{f}$  обладают одинаковыми группами автоморфизмов, т.е. они изомерны.

В работе Шеннона [31] установлено, что число симметричных функций  $n$  переменных не превышает  $n!2^n 2^{3 \cdot 2^{(n-2)}}$ . С ростом  $n$  эта величина растет медленнее, чем общее число булевых функций  $2^{2^n}$ . Поэтому, как показано в [30],

$$\lim_{n \rightarrow \infty} \frac{n! 2^n 2^{3 \cdot 2^{(n-2)}}}{2^{2^n}} = 0,$$

т. е. с ростом  $n$  доля симметричных функций в общем числе булевых функций  $n$  переменных уменьшается: при больших  $n$  почти все булевы функции обыкновенны, а в области малых  $n$  сосредоточена основная часть симметричных функций. Согласно результатам анализа [34] на симметричность представителей всех 380 классов эквивалентности-изоморфизма функций от четырех переменных ( $n = 4$ ), 75 % из них симметричны, остальные обыкновенны.

Приведенные данные о соотношении между числом симметричных и обыкновенных булевых функций при различных  $n$  позволяют сделать выводы об эффективности использования информации о симметриях булевых функций при различных  $n$ .

### 3.4. Выявление симметрий булевых функций

Очевидный алгоритм выявления симметрий булевых функций состоит из применения к нему всех  $n!2^n$  преобразований группы  $L_n$  и выделения тех из них, которые переводят функцию  $f(x_1, x_2, \mathbf{K}, x_n)$  в себя. Выделенные преобразования образуют группу  $H_n$  автоморфизмов булевой функции, характеризующую ее симметричность (или обыкновенность, если  $H_n = e$ ).

Комбинаторный характер этого алгоритма означает его высокую трудоемкость, поэтому совершенствование методов поиска симметрий булевых функций не прекращается. Наиболее существенных результатов удалось достичь для частных случаев постановки задачи выявления симметрий. Здесь повышение эффективности достигается путем использования специфики каждой конкретной задачи. Такие задачи и методы их решения рассматриваются в следующем параграфе. Здесь остановимся на наиболее общих методах.

Развитие общих методов отражено в работах [15; 30; 34; 35].

Задача выявления симметрий, как видно из тривиального алгоритма, следующего из определения симметрий, распадается на две комбинаторные подзадачи: преобразования функций  $f(x_1, x_2, \mathbf{K}, x_n)$  и сопоставления функций до и после преобразования. Методы их решения развиваются независимо. Основные направления развития – сокращение числа преобразований (сопоставлений) и повышение их эффективности.

Один из подходов к сокращению числа сравнений функций заключается в уменьшении множества испытательных преобразований  $T \in L_n$ . Это возможно при условии, что теоретически доказана невозможность потерь искомого  $h \in H_n$  вследствие такого сокращения.

Известно [35], что вместо  $n!2^n$  достаточно обойтись  $(n! + 2^n)$  сравнениями функций  $S_{i_1 i_2 \mathbf{K} i_n} f(x_1, x_2, \mathbf{K}, x_n)$ ,  $N_{c_1 c_2 \mathbf{L} c_n} f(x_1, x_2, \mathbf{K}, x_n)$ , так как из  $f = S_{\beta} N_{\alpha} f$  следует

$$N_{\alpha} f = S_{\beta} f. \quad (3.4)$$

Это непосредственно вытекает из свойства преобразования  $N_{\alpha}$  – быть обратным себе. Однако и это много, так как число  $(n! + 2^n)$  быстро увеличивается с ростом  $n$ .

Как показано в п. 3.2, подгруппа симметрий  $H_n \in L_n$  разбивает группу  $L_n$  на классы эквивалентности – левосторонние смежные классы. Все элементы одного класса одинаково действуют на функцию  $f(x_1, x_2, \mathbf{K}, x_n)$ . Это позволяет сократить число пробных преобразований  $T \in L_n$  при поиске симметрий булевой функции: достаточно использовать по одному представителю от каждого класса эквивалентности. Тогда число пробных преобразований сокращается до количества, равного индексу  $j$  подгруппы  $H_n$  в группе  $L_n$ .

Практически отсечение пробных преобразований  $T \in L_n$ , эквивалентных по  $h \in H_n$ , осуществляется по схеме, рассмотренной в п. 2.4.3. Для этого необходимо в процессе построения функций  $f'(x_1, x_2, \mathbf{K}, x_n) = Tf(x_1, x_2, \mathbf{K}, x_n)$ ,  $T \in L_n$ , пополнять множество  $H_n$  теми из них, для которых выполняется равенство (3.3), и не использовать эквивалентные по  $h \in H_n$  преобразования  $T$  при дальнейшем построении функций  $f'$ . При решении этой задачи полезно использовать каноническое упорядочение множества преобразований  $T \in L_n$ .

Некоторые общие методы сокращения перебора  $T \in L_n$  и повышения эффективности сравнения  $f$  и  $f' = Tf$  при выявлении симметрий булевой функции разработаны для конкретных способов представления исследуемой функции. Рассмотрим такие методы для функций, представленных в совершенной дизъюнктивной нормальной форме (СДНФ) и в дизъюнктивной нормальной форме (ДНФ).

#### 3.4.1. Выявление симметрий функции, заданной в СДНФ

Представим булеву функцию, заданную в СДНФ, булевой матрицей инцидентностей  $\mathbf{F}$ , в которой переменным соответствуют столбцы, а слагаемым – строки. В матрице инцидентностей переименование переменных отображается перестановкой столбцов, а изменение знака – заменой в соответствующем столбце единиц нулями, а нулей единицами. Если  $T = N_\alpha S_\beta$  – инвариант, новая матрица  $\mathbf{F}' = \mathbf{F}\mathbf{T}$  совпадает с исходной точностью до последовательности строк ( $\mathbf{T}$  – подстановочная матрица, соответствующая подстановке  $T \in L_n$ ). Другими словами, измененная матрица  $\mathbf{F}'$  должна приводиться к исходной матрице  $\mathbf{F}$  перестановкой строк.

Если функция представлена матрицей инцидентностей, к ней применимы средства сокращения перебора  $T \in G_n$ , рассмотренные в п. 1.6. Это, в первую очередь, разбиение множества строк и столбцов матрицы  $\mathbf{F}$  на классы эквивалентности. Но есть и особенности указанного разбиения, связанные с инверсированием переменных, т.е. с необходимостью проверять кроме пробных преобразований  $T \in G_n$  еще и преобразования  $T \in R_n$ . Рассмотрим эти особенности.

Условимся характеризовать столбцы матрицы инцидентностей отно-



шением  $r$  числа нулей к числу единиц в каждом. В зависимости от соотношения величин  $r_i, r_j$  двух столбцов матрицы  $\mathbf{F}$ , подлежащих преобразованию с помощью оператора  $T \in L_n$ , возможны следующие ситуации:

1.  $r_i = r_j \neq 1$ . Очевидно, что симметрии следует искать только среди операторов  $T \in G_n$ , т. е. среди тех, которые преобразуют  $x_i$  в  $x_j$  без инверсии.

2.  $r_i = 1/r_j \neq 1$ . В этом случае достаточно проверить только те операторы  $T \in L_n$ , которые преобразуют  $x_i$  в  $x_j$  с инверсией.

3.  $r_i = r_j = 1$ . Только в этом случае следует рассмотреть все операторы  $T \in L_n$  – как с инверсией, так и без нее.

4.  $r_i \neq r_j$ . В этом случае вообще нет смысла рассматривать соответствующие операторы  $T \in L_n$ , так как среди них не может быть инвариантов.

Для второй ситуации предлагается [15; 34] сразу перейти к подстановкам без инверсий, предварительно изменив в матрице  $\mathbf{F}$  знак аргумента с  $r_j < 1$  на обратный, в результате чего все элементы столбца меняются на противоположные. Это изменение в матрице, называемое ее стандартизацией [15; 34], компенсируется инверсией соответствующей переменной в каждой из выявленных симметрий. Таким образом, вторая ситуация приводится к первой, когда рассматриваются только подстановки без инверсий.

Для сопоставления функций  $f, f' = Tf$  используется сравнение каждой строки матрицы  $\mathbf{F}$  с каждой строкой матрицы  $\mathbf{F}'$ . Эта задача сводится к сличению двух массивов двоичных чисел или их десятичных эквивалентов. Сокращение числа сравнений возможно, как показано в п. 1.6.1, за счет разбиения множества строк матрицы на классы эквивалентности. Тогда каждая строка одной матрицы сравнивается только с эквивалентными строками из другой. Пусть  $m$  – число строк в матрице  $\mathbf{F}$ , равное числу конъюнкций в СДНФ, а  $m_k$  – число строк в  $k$ -м классе эквивалентности. Тогда число сравнений сокращается в  $m^2 / (\sum_k m_k^2)$  раз.

Допустим, что после стандартизации матрицы  $\mathbf{F}$  в ней выполнено разбиение строк и столбцов на классы эквивалентности и выделено подмножество столбцов с  $r = 1$ . Если  $n_1$  – число столбцов в этом подмножестве, то число перебираемых вариантов  $T \in R_n$  (с инверсией) сокращается до  $2^{n_1}$ , т.е.

в  $2^{(n-n_1)}$  раз. Часто и это число удается дополнительно сократить, если использовать уже выполненное разбиение матрицы  $\mathbf{F}$ . Действительно, из совпадения матриц для  $T \in H_n$  следует, что их разбиения совпадают. Поэтому нет смысла рассматривать те преобразования  $T \in R_n$  (инверсии), которые не могут быть компенсированы перестановками строк и столбцов, допустимыми при этом разбиении.

С учетом отмеченных особенностей предлагается следующий порядок выявления симметрий булевой функции, представленной в СДНФ:

1. Привести матрицу инцидентностей  $\mathbf{F}$  функции к стандартной форме.
2. Разбить множества строк и столбцов на классы эквивалентности, как рекомендуется в п. 1.6.1.
3. В множестве классов эквивалентности по столбцам выделить класс с  $r = 1$ . Из множества преобразований  $T \in L_n$ , относящихся к элементам этого класса, исключить недопустимые при выбранном разбиении.
4. Организовать перебор оставшихся преобразований  $T \in L_n$ , применяя общие рекомендации, данные в начале п. 3.4, т. е. дополнительно сокращая пробные преобразования  $T \in L_n$  за счет соотношения (3.4) и отсекаем тех преобразований, которые симметричны уже выявленным.
5. Для каждого отобранного преобразования  $T \in L_n$  построить соответствующую матрицу  $\mathbf{F}' = \mathbf{F}\mathbf{T}$  и сравнить ее с матрицей  $\mathbf{F}$ , учитывая разбиение строк последней.

**Пример 3.5.** Пусть требуется определить симметрии функции, заданной матрицей инцидентностей

$$\mathbf{F} = \begin{matrix} & \begin{matrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \end{matrix} \\ \begin{matrix} \left| \right. \\ \left| \right. \\ \left| \right. \\ \left| \right. \\ \left| \right. \\ \left| \right. \\ \left| \right. \\ \left| \right. \\ \left| \right. \\ \left| \right. \end{matrix} & \begin{matrix} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \end{matrix} & \begin{matrix} \left. \right| \\ \left. \right| \\ \left. \right| \\ \left. \right| \\ \left. \right| \\ \left. \right| \\ \left. \right| \\ \left. \right| \\ \left. \right| \\ \left. \right| \end{matrix} \end{matrix}.$$

Выполним разбиение матрицы. Начинаем с анализа соотношения  $r$  числа нулей и единиц в столбцах матрицы. Здесь они выписаны в той же последовательности, что и столбцы матрицы:

номер столбца	1	2	3	4	5	6
отношение	7/1	7/1	4/4	4/4	5/3	3/5

После выполнения стандартизации матрицы множество ее строк разбиваем на классы эквивалентности по числу единиц, а множество столбцов – по соотношению нулей и единиц:

$$\begin{array}{cccccc}
 x_1 & x_2 & x_3 & x_4 & x_5 & \bar{x}_6
 \end{array}$$

$$\mathbf{F} = \left\| \begin{array}{c|c|c}
 \begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} & \begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array} & \begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} \\
 \begin{array}{cc} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{array} & \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{array} & \begin{array}{cc} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{array} \\
 \begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} & \begin{array}{cc} 0 & 0 \\ 1 & 1 \end{array} & \begin{array}{cc} 1 & 1 \\ 0 & 0 \end{array}
 \end{array} \right\| .$$

Теперь каждый блок матрицы разбиваем на  $d$ -классы:

$$\begin{array}{cccccc}
 x_1 & x_2 & x_3 & x_4 & x_5 & \bar{x}_6
 \end{array}$$

$$\mathbf{F} = \left\| \begin{array}{c|c|c|c}
 \begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} & \begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array} & \begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} & \begin{array}{cc} 0 & 0 \\ 0 & 0 \end{array} \\
 \begin{array}{cc} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{array} & \begin{array}{cc} 0 & 1 \\ 1 & 0 \\ 1 & 0 \end{array} & \begin{array}{cc} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{array} & \begin{array}{cc} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{array} \\
 \begin{array}{cc} 1 & 0 \\ 0 & 1 \end{array} & \begin{array}{cc} 0 & 0 \\ 1 & 1 \end{array} & \begin{array}{cc} 1 & 1 \\ 0 & 0 \end{array} & \begin{array}{cc} 1 & 1 \\ 0 & 0 \end{array}
 \end{array} \right\| .$$

Разбиение матрицы  $\mathbf{F}$  показывает, что для выявления ее симметрий достаточно проверить перестановку переменных  $x_5$  и  $\bar{x}_6$ , а также перестановки переменных  $x_3$  с  $x_4$  с различными комбинациями их инверсирования. Обратим внимание на блок из нулей на пересечении 7-й строки и с 3-м и 4-м столбцами: инверсирование его элементов никакими перестановками строк и столбцов матрицы, допускаемыми ее разбиением, невозможно компенсировать. Поэтому остается проверить следующие подстановки:  $N_{000001} S_{123465}$ ,  $N_{000001} S_{124365}$ ,  $N_{000001} S_{124365}$ . В результате проверки убеждаемся, что это искомые автоморфизмы.

**Пример 3.6.** Найдем группу  $H_n$  булевой функции, заданной матрицей

$$\mathbf{F} = \begin{array}{c} x_1 \ x_2 \ x_3 \ x_4 \\ \left\| \begin{array}{cccc} 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{array} \right\| \end{array}.$$

Анализируем соотношения  $r$  нулей и единиц в столбцах матрицы:

номер столбца	1	2	3	4
отношение $r$	2/2	1/3	1/3	2/2

Матрица записана в стандартной форме. Разбиваем множеств ее строк и столбцов на  $d$ -классы, предварительно для наглядности упорядочив их по числу единиц:

$$\mathbf{F} = \begin{array}{c} x_1 \ x_4 \ x_2 \ x_3 \\ \left\| \begin{array}{cc|cc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{array} \right\| \end{array}.$$

Класс эквивалентности с  $r_1 = 1$  – это  $\{1,4\}$ . Из разбиения не следует дополнительных ограничений на преобразования  $N_\alpha$ , действующие на этом множестве. Таким образом, с учетом выполненного разбиения остается проверить подстановки  $S_{1234}, S_{1324}, S_{4231}, S_{4321} \in G_4$  и все возможные варианты инверсий на множестве  $\{1,4\}$  при каждой из этих подстановок. В данном случае это преобразования  $N_{0000}, N_{0001}, N_{1000}, N_{1001}$ . Всего вместе с единичной подстановкой необходимо выполнить  $2^n 2! = 2! = 16$  преобразований. Покажем, как с применением (3.4) сокращается этот перебор. Выписываем набор матриц  $\mathbf{F}' = \mathbf{F}\mathbf{T}$ , соответствующих  $S_\beta \in G_4$ :

$$\begin{array}{c} x_1 \ x_4 \ x_2 \ x_3 \\ \left\| \begin{array}{cccc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{array} \right\|, \end{array} \begin{array}{c} x_4 \ x_1 \ x_2 \ x_3 \\ \left\| \begin{array}{cccc} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{array} \right\|, \end{array} \begin{array}{c} x_1 \ x_4 \ x_3 \ x_2 \\ \left\| \begin{array}{cccc} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{array} \right\|, \end{array} \begin{array}{c} x_4 \ x_1 \ x_3 \ x_2 \\ \left\| \begin{array}{cccc} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{array} \right\| \end{array}.$$

Выписываем матрицы  $\mathbf{F}''$ , полученные инверсированием элементов столбцов матрицы  $\mathbf{F}$  в соответствии с перечисленными преобразованиями  $N_\alpha \in R_4$ :

$$\begin{array}{cccc} x_1 & x_4 & x_2 & x_3 & x_1 & \bar{x}_4 & x_2 & x_3 & \bar{x}_1 & x_4 & x_3 & x_2 & \bar{x}_1 & \bar{x}_4 & x_3 & x_2 \end{array}$$

$$\left\| \begin{array}{cccc} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \end{array} \right\|, \left\| \begin{array}{cccc} 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \end{array} \right\|, \left\| \begin{array}{cccc} 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{array} \right\|, \left\| \begin{array}{cccc} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{array} \right\|.$$

Попарно сопоставляем между собой матрицы первого и второго множеств, у которых совпадают разбиения числа единиц и нулей по строкам, и выписываем сочетания подстановок и инверсий для совпадающих матриц:

$$e, N_{0000} S_{4321}, N_{1001} S_{4231}, N_{1001} S_{1324}.$$

Это и есть искомые симметрии заданной функции. Таким образом, число проверочных преобразований  $T \in L_4$  сокращено с 16 до 8.

### 3.4.2. Выявление симметрий функции, заданной в ДНФ

Один из способов уменьшения числа сравнений заключается в сокращении записи самой функции, т.е. в переходе от СДНФ к сокращенной или минимальной ДНФ. Очевидно, что в этом случае анализ функций  $f(x_1, x_2, \mathbf{K}, x_n)$ ,  $f' = Tf(x_1, x_2, \mathbf{K}, x_n)$  на эквивалентность осуществляется сложнее, чем при представлении их в СДНФ. Но количество сравнений функций до и после их преобразований уменьшается, и зачастую существенно. Как и в подходе, рассмотренном в п. 3.4.1, поиск симметрий сводится к сопоставлению покрытий этих двух функций. Только в данном случае эта задача не решается простым сравнением двух массивов двоичных чисел или их десятичных эквивалентов. Проанализируем возможные соотношения между покрытиями двух функций  $f, f'$ , представив их для наглядности соответствующими множествами  $M, M'$  точек  $n$ -мерного логического пространства (рис. 3.1):

- а) множества  $M, M'$  не пересекаются (рис. 3.1, а);
- б) множества  $M, M'$  пересекаются на подмножестве  $M_\cap = M \cap M'$  (рис. 3.1, б);
- в) множества  $M, M'$  пересекаются, и при этом одно из них поглощается другим:  $M_\cap = M \subset M'$  или  $M_\cap = M' \subset M$  (рис. 3.1, в);
- г) множества  $M, M'$  совпадают:  $M_\cap = M = M'$  (рис. 3.1, г).

Представим заданные функции интервальными [36] матрицами инцидентностей. В интервальной матрице, как и в матрице инцидентностей,

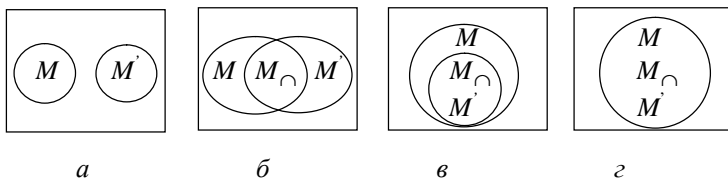


Рис 3.1

столбцы соответствуют переменным булевой функции, а строки – ее слагаемым. Каждая строка может рассматриваться как интервальный вектор  $x_i$ ,  $i = 1, 2, \dots, t$ , с координатами  $(0, 1, -)$ , представляющий множество точек  $n$ -мерного логического пространства (интервал), покрытых соответствующим слагаемым. Тогда множество  $M$  точек, покрываемых булевой функцией, можно представить через интервалы  $M_i$ :

$$M = \bigcup_{i=1}^m M_i; \quad M' = \bigcup_{j=1}^{m'} M'_j.$$

Очевидно следующее

**Утверждение 3.2.** Для того чтобы функции  $f(x_1, x_2, \mathbf{K}, x_n)$ ,  $f' = Tf(x_1, x_2, \mathbf{K}, x_n)$  были эквивалентными, необходимо и достаточно, чтобы каждый интервал  $M_i$ ,  $i = 1, 2, \dots, t$ , функции  $f$  покрывался интервалами  $M'_j$ ,  $j = 1, 2, \dots, t'$ , функции  $f'$  и наоборот – каждый интервал  $M'_j$ ,  $j = 1, 2, \dots, t'$ , функции  $f'$  покрывался интервалами  $M_i$ ,  $i = 1, 2, \dots, t$ , функции  $f$ .

Действительно, условиями утверждения исключаются ситуации, отраженные в рис. 3.1, а, б, в, и допускается только ситуация, представленная на рис. 3.1, г.

Приведенный далее алгоритм сравнения функций  $f$ ,  $f'$  основан на утверждении 3.2. В нем проверка покрытия интервалов осуществляется с помощью операций над интервалами: вычисление пересечения интервалов  $M_{ij} = M_i \mathbf{I} M'_j$  и вычисление остатка интервала  $M_i$ , не покрытого интервалом  $M'_j$ .

Рассмотрим сначала эти операции, представив интервалы  $M_i$  и  $M'_j$  соответствующими интервальными векторами  $\mathbf{x}_i, \mathbf{x}'_j$ .

Напомним [36], что пересечение двух интервалов  $M_i, M'_j$  – также интервал  $M_{ij}$ , представленный интервальным вектором  $\mathbf{x}_{ij}$ , который строится по следующим правилам: совпадающие координаты векторов  $\mathbf{x}_i, \mathbf{x}'_j$  переносятся в вектор  $\mathbf{x}_{ij}$  без изменений; если в одном из векторов координата имеет неопределенное значение ("–"), а в другом соответствующая координата определена, то в вектор  $\mathbf{x}_{ij}$  переносится эта определенная координата; если хотя бы одной координате вектора  $\mathbf{x}_i$ , равной 1 (0), соответствует в векторе  $\mathbf{x}'_j$  координата 0 (1), пересечение пусто (векторы  $\mathbf{x}_i, \mathbf{x}'_j$  ортогональны). Например, при  $\mathbf{x}_i = (1, 0, 0, -, -, -, 1)$  и  $\mathbf{x}'_j = (1, -, 0, -, -, 1, -)$   $\mathbf{x}_{ij} = (1, 0, 0, -, -, 1, 1)$ ; а при  $\mathbf{x}_i = (1, 1, -, 0, 0, -, 1)$  и  $\mathbf{x}'_j = (-, -, 1, 0, -, 1, 0)$  пересечение пусто.

Под непокрытым остатком интервала  $M_i$  будем подразумевать множество точек этого интервала, которые не поглощаются интервалами  $M'_1, M'_2, \dots, M'_{m'}$ . Обозначим этот остаток  $\delta^{(i)}$ .

Остаток интервала  $M_i$ , не покрытый интервалом  $M'_j, j \in \{1, 2, \dots, m'\}$ , вычисляется как

$$\delta_j^{(i)} = M_i \setminus (M_i \mathbf{I} M'_j), \quad (3.5)$$

а остаток интервала  $M_i$ , не покрытый интервалами  $M'_1, M'_2, \dots, M'_\alpha, \alpha \in \{1, 2, \dots, m'\}$ , вычисляется по рекуррентной формуле

$$\delta_{1,2,\dots,\alpha}^{(i)} = \delta_{1,2,\dots,(\alpha-1)}^{(i)} \setminus (\delta_{1,2,\dots,(\alpha-1)}^{(i)} \mathbf{I} M'_\alpha).$$

Непокрытый остаток интервала на множестве точек  $n$ -мерного булевого пространства может быть представлен в общем случае не одним интервальным вектором, а их совокупностью. И вычисление этого остатка можно заменить операциями над векторами. Для этого построим по векторам  $\mathbf{x}_i, \mathbf{x}'_j$  множество производных векторов  $\mathbf{x}_{ij}$ , применив следующий алгоритм.

Алгоритм 3.1.

Шаг 1. Построить множество  $I = \{i_1, i_2, \mathbf{K}, i_\beta, \mathbf{K}, i_{|I|}\}$  номеров неопределенных координат вектора  $\mathbf{x}_i$ , которым в векторе  $\mathbf{x}'_j$  соответствуют определенные координаты.

Шаг 2. Для построения производного вектора  $\mathbf{x}_{i_\beta, j}$  с порядковым номером  $i_\beta \in I$  в векторе  $\mathbf{x}_i$  заменить неопределенные координаты с порядковыми номерами  $i_1, i_2, \mathbf{K}, i_{(\beta-1)}$  соответствующими координатами из вектора  $\mathbf{x}'_j$ , а координату с номером  $i_\beta$  – соответствующей инверсированной координатой из  $\mathbf{x}'_j$ .

Шаг 2 выполнить для каждого  $i_\beta \in I$ .

Например, для  $\mathbf{x}_i = (1, -, 0, -, -, 1, -)$  и  $\mathbf{x}'_j = (-, 0, -, 1, 0, 1, -)$  в соответствии с алгоритмом 3.1 множество  $I = (2, 4, 5)$  и множество производных векторов – это  $(1, 1, 0, -, -1, -)$ ,  $(1, 0, 0, 0, -, 1, -)$ ,  $(1, 0, 0, 1, 1, 1, -)$ .

Нетрудно убедиться в том, что множество производных векторов, построенных по алгоритму 3.1, отображает остаток множества  $M_i$ , не покрытый множеством  $M'_j$ . Это можно проверить, например, аналитически с помощью разложения слагаемого  $p_i$  функции  $f_i$ , соответствующего вектору  $\mathbf{x}_i$ , по  $x_{i_\beta}$ ,  $i_\beta \in I$ , которое в данном случае является операцией, обратной склеиванию:

$$p_i = x_{i_\beta} p_i \vee \bar{x}_{i_\beta} p_i. \quad (3.6)$$

Одно из слагаемых правой части (3.6), как следует из определения  $I$ , совместимо [36] со слагаемым  $p'_j$  функции  $f'$ , соответствующим вектору  $\mathbf{x}'_j$ , а другое – ортогонально ему. Разлагая последовательно по всем  $x_{i_\beta}$ ,  $i_\beta \in I$ , каждую вновь полученную совместимую составляющую, пока не исчерпается множество  $I$ , выделяем ту из них, которая совпадает с пересечением  $p_i \mathbf{I} p'_j$ . Следовательно, конъюнкция ортогональных слагаемых, полученных в процессе разложения, представляет собой аналитическое описание остатка  $\delta_j^{(i)}$  в (3.5). Учитывая правила построения совместимых и ортогональных слагаемых, убеждаемся, что переход от каждого слагаемого



этого остатка к соответствующему интервальному вектору приведет к тому же набору векторов  $\mathbf{x}_{i_\beta}$ , что и алгоритм 3.1.

Теперь рассмотрим алгоритм сравнения функций  $f, f'$ .

Алгоритм 3.2.

Шаг 1. Для каждого  $i \in \{i_1, i_2, \mathbf{K}, i_m\}$  вычислить по алгоритму 3.1 непокрытый остаток  $\delta_{1,2,\mathbf{K},j,\mathbf{K},m'}^{(i)}$ . В случае обнаружения  $i_\alpha \in \{i_1, i_2, \mathbf{K}, i_m\}$ , при котором  $\delta_{1,2,\mathbf{K},j,\mathbf{K},m'}^{(i)} \neq \emptyset$ , идти к 4.

Шаг 2. Для каждого  $j \in \{1, 2, \dots, m'\}$  вычислить по алгоритму 3.1 непокрытый остаток  $\delta_{1,2,\mathbf{K},i,\mathbf{K},m}^{(j)}$ . При обнаружении  $j_\beta \in \{1, 2, \dots, m'\}$ , для которого  $\delta_{1,2,\mathbf{K},i,\mathbf{K},m}^{(j)} \neq \emptyset$ , идти к шагу 4.

Шаг 3. Функции  $f, f'$  эквивалентны.

Шаг 4. Функции  $f, f'$  не эквивалентны.

**Пример 3.7.** Пусть необходимо сравнить функции

$$f = \bar{x}_1 \bar{x}_2 \bar{x}_3 x_4 \vee \bar{x}_1 \bar{x}_2 \bar{x}_4 \vee x_2 x_3 x_4,$$

$$f' = \bar{x}_1 \bar{x}_2 x_3 \bar{x}_4 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee x_2 x_3 x_4.$$

Интервальные матрицы для этих функций

$$\mathbf{F} = \left\| \begin{array}{cccc} 0 & 0 & 0 & 1 \\ 0 & 0 & - & 0 \\ - & 1 & 1 & 1 \end{array} \right\|; \mathbf{F}' = \left\| \begin{array}{cccc} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & - \\ - & 1 & 1 & 1 \end{array} \right\|.$$

Перечислим результаты решения этой задачи на различных этапах в соответствии с алгоритмом 3.2:

$$\text{Шаг 1. } i = 1, \delta_1^{(1)} = \{(0, 0, 0, 1)\}, \delta_{1,2}^{(1)} = \emptyset;$$

$$i = 2, \delta_1^{(2)} = \{(0, 0, 0, 0)\}, \delta_{1,2}^{(2)} = \emptyset;$$

$$i = 3, \delta_1^{(3)} = \{(-, 1, 1, 1)\}, \delta_{1,2}^{(3)} = \{(-, 1, 1, 1)\}, \delta_{1,2,3}^{(3)} = \emptyset.$$

$$\text{Шаг 2. } i = 1, \delta_1^{(1)} = \{(0, 0, 1, 0)\}, \delta_{1,2}^{(1)} = \emptyset;$$

$$i = 2, \delta_1^{(2)} = \{(0, 0, 0, 0)\}, \delta_{1,2}^{(2)} = \emptyset;$$

$$i = 3, \delta_1^{(3)} = \{(-, 1, 1, 1)\}, \delta_{1,2}^{(3)} = \{(-, 1, 1, 1)\}, \delta_{1,2,3}^{(3)} = \emptyset.$$

Шаг 3. Функции эквивалентны.

**Пример 3.8.** Пусть необходимо проверить эквивалентность функций

$$f = \bar{x}_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_2 x_3 \vee x_1 \bar{x}_3,$$

$$f' = \bar{x}_1 \bar{x}_2 \vee \bar{x}_2 x_3 \vee x_1 x_2 \bar{x}_3.$$

Их интервальные матрицы

$$\mathbf{F} = \left\| \begin{array}{ccc} 0 & 0 & 0 \\ - & 0 & 1 \\ 1 & - & 0 \end{array} \right\|; \quad \mathbf{F}' = \left\| \begin{array}{ccc} 0 & 0 & - \\ - & 0 & 1 \\ 1 & 1 & 0 \end{array} \right\|.$$

Решение по алгоритму 3.2:

Шаг 1.

$$i = 1, \delta_1^{(1)} = \emptyset;$$

$$i = 2, \delta_1^{(2)} = \{(1, 0, 1)\}, \delta_{1,2}^{(2)} = \emptyset;$$

$$i = 3, \delta_1^{(3)} = \{(1, -, 0)\}, \delta_{1,2}^{(3)} = \{(1, -, 0)\}, \delta_{1,2,3}^{(3)} = \{(1, 0, 0)\} \neq \emptyset.$$

Шаг 4. Функции не эквивалентны.

Аналитические методы сравнения двух функций [37; 38 и др.] здесь не приводятся. Они достаточно трудоемки, но хуже поддаются автоматизации по сравнению с описанным методом.

При задании булевой функции в ДНФ не существует простых способов разбиения множеств строк и столбцов матрицы  $\mathbf{F}$  на классы эквивалентности. Поэтому для сокращения числа проверочных преобразований  $T \in L_n$  можно рекомендовать общие методы, рассмотренные в начале п. 3.4, например, с помощью выявленных симметрий  $T \in H_n$ .

### 3.5. Выявление симметрий булевых функций специальных видов

Для иллюстрации упрощения процедуры распознавания симметрий частных случаев булевых функций остановимся на функциях двух видов: симметрических и ортогональных. Первые широко используются в синтезе

логических схем, так как они допускают наиболее простую схемную реализацию. Значение вторых читатель уяснит из последующего изложения.

### 3.5.1. Симметрические булевы функции

В большинстве литературных источников [30; 33; 35] под симметрической подразумевается функция, инвариантная к любому переименованию без инверсирования ее переменных\*.

Симметрическая функция может быть задана списком чисел  $\alpha_k$  единиц в наборе переменных  $x_1, x_2, \mathbf{K}, x_n$ , обращающих функцию  $f(x_1, x_2, \mathbf{K}, x_n)$  в единицу. Такое число  $\alpha_i$  называется рабочим [33]. В геометрической интерпретации  $n$ -мерного логического пространства рабочее число  $\alpha_i$  – это номер уровня гиперкуба  $E^n$ , на котором симметрическая функция равна единице. Поэтому для симметрических функций введено специальное обозначение – через рабочие числа:  $S_{\alpha_1 \alpha_2 \mathbf{K} \alpha_i \mathbf{K}}(x_1, x_2, \mathbf{K}, x_n)$ .

**Пример 3.9.** На рис. 3.2 для симметрической функции

$$S_{23} = x_1 x_2 x_3 \vee \bar{x}_1 x_2 x_3 \vee x_1 \bar{x}_2 x_3 \vee x_1 x_2 \bar{x}_3.$$

с рабочими числами 2, 3 выделены уровни гиперкуба с номерами 2, 3.

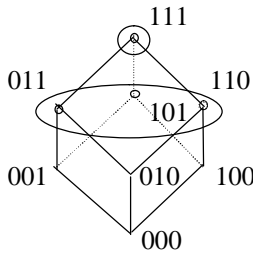


Рис 3.2

---

\* Такие функции предлагается [15; 42] называть абсолютно симметричными (totally symmetric), что, пожалуй, лучше отражает суть этого понятия. Однако в отечественной литературе за ними закрепился термин "симметрические". Этот термин использован в нашем изложении как традиционный для нашей литературы.

Симметрическая функция с одним рабочим числом называется элементарной [33]. Ей соответствует один из уровней гиперкуба  $E^n$ .

Из определения симметрической функции следует, что для нее задача выявления симметрий может быть заменена задачей опознания этой функции по каким-либо, по возможности нетрудоемким, признакам, а значит, предпочтительно не комбинаторного характера.

Для опознания симметрической функции к ней достаточно применить образующие группы  $G_n$  [30]  $S_{(1,2)} = S_{2134} \mathbf{K}_n$  и  $S_{(1,2,\mathbf{L},n)} = S_{231} \mathbf{K}_{n1}$ : функция симметрическая, если для нее выполняется равенство

$$f(x_1, x_2, \mathbf{K}, x_n) = f(x_2, x_1, x_3, x_4, \mathbf{K}, x_n) = f(x_2, x_3, \mathbf{K}, x_n, x_1).$$

Любая симметрическая функция может быть однозначно разложена на элементарные функции:

$$S_{\alpha_1 \alpha_2 \mathbf{K} \alpha_k} (x_1, x_2, \mathbf{K}, x_n) = \sum_{i=1}^k S_{\alpha_i} (x_1, x_2, \mathbf{K}, x_n).$$

При этом число элементарных функций  $C(n, \alpha_i)$  с рабочим числом  $\alpha_i$

$$C(n, \alpha_i) = n! / (n - \alpha_i)! \alpha_i!.$$

Отсюда другой способ опознания симметрических функций: записать функцию в виде суммы элементарных функций для каждого рабочего числа  $\alpha_i$  и сравнить количество  $N_i$  элементарных функций  $S_{\alpha_i}(x_1, x_2, \mathbf{K}, x_n)$

в разложении с  $C(n, \alpha_i)$ . Функция симметрическая, если для любой ее элементарной функции выполняется равенство между этими величинами. При матричном представлении исследуемой функции этот метод интерпретируется подсчетом числа строк в подматрицах матрицы  $\mathbf{F}$  инцидентностей с одинаковым числом  $\alpha_i$  единиц.

**Пример 3.10.** Функция, представленная матрицей инцидентностей

$$\mathbf{F}_1 = \begin{vmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 1 \end{vmatrix}$$

является симметрической, поскольку для нее  $N_0 = C(3, 0) = 1$ ;  $N_1 = 0$ ;  $N_2 = C(3, 2) = 3$ ;  $N_3 = C(3, 3) = 1$ . Следовательно, эту функцию можно записать как

$$f(x_1, x_2, x_3) = S_{0,2,3}(x_1, x_2, x_3).$$

А функция с матрицей инцидентностей

$$\mathbf{F}_2 = \begin{array}{c} \begin{array}{cccc} & x_1 & x_2 & x_3 & x_4 \end{array} \\ \left\| \begin{array}{cccc} 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{array} \right\| \end{array}$$

не является симметрической, так как для нее  $N_1 = 2 \neq C(4, 1) = 4$ ;  $N_2 = 1 \neq C(4, 2) = 6$ ;  $N_3 = 2 \neq C(4, 3) = 4$ .

Существуют наглядные способы опознания симметрической функции, основанные на ее графическом представлении, при котором функциональной симметрии переменных  $x_1, x_2, \mathbf{K}, x_n$  соответствует геометрическая симметрия. Примером может служить метод, предложенный в работе [39].

Частным случаем симметрической функции является частично симметрическая функция [40], т.е. симметрическая относительно некоторого подмножества всех переменных функции  $f(x_1, x_2, \mathbf{K}, x_n)$ . Методы распознавания такой симметричности рассмотрены, например, в работах [40-42].

Для обобщения приведенного здесь метода определения симметрических функций предлагается [15] допустить инверсирование переменных при их перестановках. Например, функция  $S = x_1 \bar{x}_2 x_3$  по первому (узкому) определению может быть представлена через элементарные как  $S_2(x_1, x_2, x_3)$ ,

а по второму (широкому) – как  $S_3(x_1, \bar{x}_2, x_3)$ , поскольку равна единице на наборе  $x_1 = \bar{x}_2 = x_3$  или  $x_1 = x_3 = 1$  и  $x_2 = 0$ .

Опознание симметрических функций, соответствующих второму определению, сводится к опознаванию симметрических функций, заданных первым определением, простой подстановкой  $\bar{x}_i$  вместо  $x_i$ . В матричном варианте это означает предварительную стандартизацию матрицы с последующим компенсирующим инверсированием переменных в описаниях симметрических функций.

**Пример 3.11.** Пусть заданной является функция  $\mathbf{F}_2$  из примера 3.10. Соответствующая стандартная матрица инцидентностей после упорядочения строк по весам имеет вид

$$\mathbf{F}_2 = \begin{array}{c} x_1 \quad \bar{x}_2 \quad x_3 \quad \bar{x}_4 \\ \left\| \begin{array}{cccc} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{array} \right\| . \end{array}$$

Сверху указаны переменные, инверсированные в процессе стандартизации матрицы. Теперь в матрице  $\mathbf{F}_2 \quad N_1 = 4 = C(4, 1); \quad N_2 = 1 = C(4, 4)$ . Следовательно, заданная функция является симметрической относительно переменных  $x_1, \bar{x}_2, x_3, \bar{x}_4$ , т.е. ее можно записать как  $S_{1,4}(x_1, \bar{x}_2, x_3, \bar{x}_4)$ .

### 3.5.2. Ортогональные булевы функции

Булева функция  $f(x_1, x_2, \mathbf{K}, x_n)$  называется ортогональной [43], если для каждой пары ее конъюнкций  $p_i, p_j, \quad i, j \in \{1, 2, \dots, n\}$ , справедливо соотношение

$$p_i p_j = 0.$$

**У т в е р ж д е н и е 3.3.** Если булева функция  $f(x_1, x_2, \mathbf{K}, x_n)$  ортогональна, то функция  $f' = Tf(x_1, x_2, \mathbf{K}, x_n)$  также ортогональна.

**Д о к а з а т е л ь с т в о.** Для того чтобы булева функция была ортогональной, необходимо и достаточно, чтобы каждая пара ее слагаемых  $p_i$  и  $p_j, \quad i, j \in \{1, 2, \dots, n\}$ , содержала хотя бы одну общую переменную  $x_k, \quad k \in \{1, 2, \dots, n\}$ , в качестве сомножителя и при этом в одно слагаемое она входила с инверсией, в другое – без инверсии [43]. Под действием  $T \in L_n$  переменная  $x_k$  может оказаться инверсированной только одновременно в каждом из слагаемых  $p_i, p_j$ . Следовательно, условие ортогональности при преобразовании не нарушится. Оно тем более не нарушится, если  $T \in G_n$ , т.е. преобразование  $T$  не инверсирует переменную  $x_k$ . Очевидно (по индукции), что это справедливо и для случая, когда число переименованных переменных больше единицы.

Пусть  $M_i$  – интервал слагаемого  $p_i$ ,  $i \in \{1, 2, \dots, m\}$ , функции  $f$ , а  $M_j$  – интервал слагаемого  $p_j$ ,  $j \in \{1, 2, \dots, m\}$ , функции  $f' = T f$ . Обозначим пересечение этих интервалов  $M_{ij}$ .

**У т в е р ж д е н и е 3.4.** *Для того чтобы ортогональная функция  $f(x_1, x_2, \mathbf{K}, x_n)$  совпала с функцией  $f'(x_1, x_2, \mathbf{K}, x_n) = T f(x_1, x_2, \mathbf{K}, x_n)$ , необходимо и достаточно, чтобы для каждого слагаемого  $p_i$  функции  $f(x_1, x_2, \mathbf{K}, x_n)$  выполнялось равенство*

$$|M_i| = \sum_{i=1}^m |M_{ij}|.$$

*Доказательство.*

*Необходимость.* Если две функции  $f$ ,  $f'$  тождественно равны между собой, то совпадают их покрытия соответственно  $M$ ,  $M'$  в  $n$ -мерном пространстве. И значит, существует взаимно однозначное соответствие между точками одного и другого покрытий. Таким образом,  $M = M'$ .

Покрытие функции  $f$  выражается через интервалы ее слагаемых  $M_i$ ,  $i \in \{1, 2, \dots, m\}$ :

$$M = \bigcup_{i=1}^m M_i. \quad (3.4)$$

В силу ортогональности функции  $f$  из (3.4) получаем

$$|M| = \sum_{i=1}^m |M_i|. \quad (3.5)$$

В свою очередь, интервал  $M_i$  слагаемого  $p_i$  выражается через подмножества  $M_{ij}$ ,  $j \in \{1, 2, \dots, m\}$ :

$$M_i = \bigcup_{j=1}^m M_{ij}, \quad i=1, 2, \dots, m. \quad (3.6)$$

Из ортогональности функции  $f'$  вытекает, что интервалы ее слагаемых не пересекаются между собой, а значит, и не пересекаются их части  $M_{ij}$ , покрывающие интервал  $M_i$ .

Таким образом, из ортогональности  $f'$  и из (3.6) следует

$$|M_i| = \sum_{j=1}^m |M_{ij}|, \quad i=1, 2, \dots, m. \quad (3.7)$$

*Достаточность.* Мощность интервала  $p_i$  вычисляется как

$$|M_i| = 2^{v_i}, \quad (3.8)$$

где  $v_i$  – число переменных, отсутствующих среди сомножителей  $p_i$ . Значение  $|M_i|$  не меняется под действием  $T \in L_n$ , поскольку преобразование  $T \in L_n$  не влияет на величину  $v_i$ . Следовательно, множество интервалов  $p_j$  функции  $f'$  характеризуется тем же набором мощностей, что и множество интервалов  $p_i$  функции  $f$ . В силу ортогональности функции  $f'$  по аналогии с равенством (3.5) получаем равенство

$$|M'| = \sum_{j=1}^m |M_j|. \quad (3.9)$$

Для каждого слагаемого  $p_j$ ,  $j \in \{1, 2, \dots, m\}$ , можно записать

$$M_j = \bigcup_{i=1}^m M_{ji}, \quad (3.10)$$

где  $M_{ji}$  – пересечение интервала  $p_j$  с интервалом  $p_i$ ,  $i \in \{1, 2, \dots, m\}$ . Из ортогональности  $f$  следует, что множества  $M_{ji}$ , покрывающие интервал  $p_j$ , также ортогональны между собой. Поэтому из (3.10) получаем

$$|M_j| = \sum_{i=1}^m |M_{ji}|. \quad (3.11)$$

Множества  $M_{ij}$  ( $M_{ji}$ ), покрывающие интервал  $p_i$  ( $p_j$ ), ортогональны в силу ортогональности  $p_j$ ,  $j \in \{1, 2, \dots, m\}$  ( $p_i$ ,  $i \in \{1, 2, \dots, m\}$ ). Таким образом, множества  $M_{ij}$  ( $M_{ji}$ ) между собой не пересекаются. Поэтому из (3.5), (3.7) следует

$$|M| = \sum_{i=1}^m \sum_{j=1}^m |M_{ij}|,$$

а из (3.9), (3.11) –

$$|M'| = \sum_{j=1}^m \sum_{i=1}^m |M_{ji}|.$$



Итак, обе функции  $f, f'$  покрывают равное число точек  $n$ -мерного булевого пространства. Это число совпадает с числом точек, покрываемых совокупностью множеств  $M_{ij} (M_{ji})$ . При этом точки, покрываемые множествами  $M_{ij} (M_{ji})$  не повторяются. В этом случае функция  $f'$  покрывает по частям точки  $f$ , а функция  $f$  – точки  $f'$ . Следовательно, покрываемые ими множества  $M, M'$  совпадают, и функции  $f, f'$  эквивалентны между собой (см. утверждение 3.2).

Из утверждения 3.4 можно сделать вывод, что сравнение ортогональной функции  $f$  с функцией  $f' = Tf, T \in L_n$ , сводится к вычислению для каждого  $p_i, i \in \{1, 2, \dots, m\}$ , суммы мощностей множеств  $M_{ij}, j = 1, 2, \dots, m$ , и сравнению ее с мощностью интервала  $p_i$ . Очевидно, что для этой цели может быть использовано равенство (3.11) вместо равенства (3.7), так как они идентичны.

Практически мощность множества  $M_{ij}$  вычисляется как

$$|M_{ij}| = 2^{v_{ij}}, \quad (3.12)$$

где  $v_{ij}$  – число переменных функции  $f(x_1, x_2, \dots, x_n)$ , отсутствующих одновременно среди сомножителей  $p_i$  и среди сомножителей  $p_j$ . В соответствующих интервальных векторах  $\mathbf{x}_i, \mathbf{x}'_j$  – это число совпадающих неопределенных координат (" – ").

Процедура сравнения функций  $f, f'$ , основанная на утверждении 3.4, эффективнее процедуры сравнения для любых функций, рассмотренных в п.3.4. И эта эффективность достигнута благодаря использованию свойства ортогональности функции  $f$ .

Остановимся теперь на сокращении числа сравнений за счет разбиения на классы эквивалентности как множества переменных функции  $f$ , так и множества ее слагаемых или (в случае представления функции интервальной матрицей) за счет разбиения строк и столбцов матрицы.

Задачу разбиения множества столбцов интервальной матрицы  $\mathbf{F}$  сведем к аналогичной задаче для матрицы инцидентностей, рассмотренной ранее в п. 3.5.1.

Число единиц  $n_k^1$  (нулей  $n_k^0$ ) в  $k$ -м столбце матрицы инцидентностей – это число точек  $n$ -мерного булевого пространства, покрытых функцией  $f$ , в которых  $k$ -я переменная принимает значение "1" ("0"). Это число мож-

но записать в виде суммы составляющих  $n_{k_i}^1$  ( $n_{k_i}^0$ ), каждая из которых представляет собой число таких точек в интервале  $i$ -го слагаемого функции:

$$n_k^1 = \sum_{i=1}^m n_{k_i}^1, \quad (3.13)$$

$$n_k^0 = \sum_{i=1}^m n_{k_i}^0. \quad (3.14)$$

Соотношения (3.13), (3.14) справедливы только для ортогональных функций, у которых интервалы отдельных слагаемых не пересекаются. Значения  $n_{k_i}^1$  ( $n_{k_i}^0$ ) определяются, как и мощности интервалов, числом неопределенных коэффициентов (" - ") в соответствующих векторах  $\mathbf{x}_i$ , которые являются строками интервальной матрицы ортогональной функции:

$$n_{k_i}^\sigma = \begin{cases} 2^{v_i}, & \text{если } k\text{-я координата вектора равна } \sigma; \\ 2^{(v_i-1)}, & \text{если } k\text{-я координата вектора равна " - "}; \\ 0, & \text{если } k\text{-я координата вектора равна } \bar{0}. \end{cases} \quad (3.15)$$

В (3.15)  $\sigma=1$ , когда  $\mathbf{x}_{k_i}=1$ , и  $\sigma=0$ , когда  $\mathbf{x}_{k_i}=0$ . Таким образом, разбиение столбцов интервальной матрицы ортогональной функции не отличается от разбиения столбцов матрицы инцидентностей. Только информация о числе нулей и единиц в столбцах матрицы получается путем анализа первой.

Очевидно, что слагаемое  $p_i$  функции  $f$  может совпадать со слагаемым  $p_j$  функции  $f'$ , если мощности их интервалов совпадают. Иначе нет смысла сравнивать между собой эти слагаемые (или соответствующие строки интервальных матриц этих функций). Это дает основание для предварительного разбиения строк матриц  $\mathbf{F}$ ,  $\mathbf{F}'$  на классы эквивалентности по мощностям соответствующих интервалов, определенных по (3.8), т.е. по числу неопределенных координат в строках.

**Пример 3.12.** Пусть требуется определить симметрии ортогональной функции, заданной матрицей инцидентностей

$$\mathbf{F} = \begin{matrix} & x_1 & x_2 & x_3 & x_4 & x_5 \\ \begin{matrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{matrix} & \begin{vmatrix} 1 & 1 & 1 & 1 & - \\ 1 & 1 & - & 0 & 0 \\ 1 & 0 & 0 & - & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 \end{vmatrix} \end{matrix}.$$

С помощью соотношений (3.13) – (3.15) вычисляем отношение  $r$  числа нулей к числу единиц соответствующей матрицы инцидентностей:

номер столбца	1	2	3	4	5
отношение	7/1	1/7	3/5	5/3	3/5

Приводим матрицу к стандартной форме

$$\mathbf{F}' = \begin{matrix} & \bar{x}_1 & x_2 & x_3 & \bar{x}_4 & x_5 \\ \begin{matrix} 1 \\ 1 \\ 1 \\ 0 \\ 1 \end{matrix} & \begin{vmatrix} 1 & 1 & 1 & 0 & - \\ 1 & 1 & - & 1 & 0 \\ 1 & 1 & 0 & - & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{vmatrix} \end{matrix}.$$

Как видно из стандартной матрицы  $\mathbf{F}'$ , классы эквивалентности номеров столбцов – это  $\{1, 2\}$ ,  $\{3, 4, 5\}$ .

Теперь вычисляем мощности интервалов строк матрицы инцидентностей, соответствующей интервальной матрице после ее стандартизации, как было показано в п. 3.5.2:

номер строки	1	2	3	4	5
мощность интервала	2	2	2	1	1

Отсюда классы эквивалентности строк – это  $\{1, 2, 3\}$ ,  $\{4, 5\}$ .

По результатам разбиения матрицы  $\mathbf{F}'$  заданной функции видно, что осталось проанализировать, как действуют на функцию подстановки без инверсии на множествах столбцов  $\{1, 2\}$ ,  $\{3, 4, 5\}$ . Число подстановок множества строк, необходимых для приведения одной матрицы к другой, сокращается за счет найденного разбиения этого множества на классы эквивалентности  $\{1, 2, 3\}$ ,  $\{4, 5\}$ .

С целью отсеечения проверочных подстановок, которые симметричны выявленным ранее, воспользуемся идеей канонического упорядочения: будем рассматривать

подстановки, не нарушающие разбиения матрицы  $\mathbf{F}'$ , в порядке лексикографического убывания последовательностей индексов в их обозначениях:

$$S_{12345}, \underline{S_{12354}}, \underline{S_{12435}}, S_{12453}, S_{12534}, S_{12543}, S_{21345}, S_{21354}, \\ S_{21435}, S_{21453}, S_{21534}, S_{21543}.$$

В этом случае, как следует из п.1.5.3, п. 2.4.3, достаточно проверить только две подстановки, чтобы выявить симметрии заданной функции. Они в приведенном списке подстановок подчеркнуты. Проверка показывает, что матрица  $\mathbf{F}'$  к ним инвариантна. Инвариантность  $\mathbf{F}'$  к перестановке одинаковых столбцов очевидна. Переход от перечисленных перестановок  $S_{i_1 i_2 \dots i_5} \in G_5$  к симметриям матрицы  $\mathbf{F}$  осуществляется добавлением к каждой множителя  $N_{10010}$ .

### 3.6. Симметрии системы булевых функций

Будем говорить, что система булевых функций

$$\begin{aligned} y_1 &= f_1(x_1, x_2, \mathbf{K}, x_n); \\ y_2 &= f_2(x_1, x_2, \mathbf{K}, x_n); \\ &\dots\dots\dots \\ y_m &= f_m(x_1, x_2, \mathbf{K}, x_n) \end{aligned} \tag{3.16}$$

является симметричной, если она переходит в себя под действием переименования ее переменных. Обозначим группу симметрий системы (3.16) символом  $H_n^{(m)}$ . Как следует из определения, ее элементы составляют те преобразования  $T \in L_n$ , к которым система инвариантна.

Система (3.16) может быть представлена парой матриц:  $\mathbf{F}, \mathbf{Y}$ . Матрица  $\mathbf{F}$  задает своими строками систему интервалов или элементарных конъюнкций. Каждая строка представляет одну из множества неповторяющихся конъюнкций в функциях системы. Поэтому число  $l$  строк матрицы  $\mathbf{F}$  равно числу этих конъюнкций. Столбцы матрицы  $\mathbf{F}$  соответствуют переменным функции. Таким образом, матрица  $\mathbf{F}$  может быть булевой или интервальной, что соответствует заданию булевых функций системы (3.16) в СДНФ или ДНФ соответственно.

Матрица  $\mathbf{Y}$  отражает отношение конъюнкций, представленных строками матрицы  $\mathbf{F}$ , к различным функциям системы (3.16). Это булева матрица. Она содержит  $l$  строк и  $m$  столбцов. На пересечении строки с номером  $i \in \{1, 2, \dots, l\}$  и столбца с номером  $j \in \{1, 2, \dots, m\}$  находится единица, если  $j$ -я функция содержит  $i$ -е слагаемое, и нуль – в противном случае.

**Пример 3.13.** Система функций

$$y_1 = x_1 x_2 \bar{x}_4 \vee \bar{x}_1 \vee x_2 \bar{x}_3;$$

$$y_2 = x_2 \bar{x}_3 \vee \bar{x}_1;$$

$$y_3 = x_1 \bar{x}_2 x_4 \vee x_2 \bar{x}_3 \vee x_1 x_3 \bar{x}_4$$

может быть представлена матрицами

$$\mathbf{F} = \begin{array}{c} x_1 \quad x_2 \quad x_3 \quad x_4 \\ \left\| \begin{array}{cccc} 1 & 1 & - & 0 \\ 0 & - & - & - \\ - & 1 & 0 & - \\ 1 & 0 & - & 1 \\ 1 & - & 1 & 0 \end{array} \right\| \end{array}; \quad \mathbf{Y} = \begin{array}{c} y_1 \quad y_2 \quad y_3 \\ \left\| \begin{array}{ccc} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{array} \right\| \end{array}.$$

Под действием преобразований  $T \in L_n$  меняются местами столбцы матрицы  $\mathbf{F}$  с одновременным инверсированием их элементов или без него. Если  $T=h \in H_n^{(m)}$ , это изменение компенсируется перестановкой строк матрицы  $\mathbf{F}$  с одновременной перестановкой соответствующих строк матрицы  $\mathbf{Y}$ . Таким образом, симметрии системы (3.16) могут быть выявлены по матрице  $\mathbf{F}$  путем воздействия на нее преобразований  $T \in L_n$  и выделения тех из них, для которых эта матрица восстанавливается перестановкой строк. Очевидно, что для выявления симметрий системы (3.16) по матрице  $\mathbf{F}$  применимы все рассмотренные методы. Понятно, что выбирать метод следует в соответствии с формой задания функций системы (СДНФ или ДНФ) и с их спецификой.

**Пример 3.14.** Пусть требуется определить симметрии системы функций

$$y_1 = \bar{x}_2 \bar{x}_4 \vee x_1 x_2 \bar{x}_3 \bar{x}_4 \vee x_1 \bar{x}_2 x_3 x_4 \vee x_2 x_4;$$

$$y_2 = \bar{x}_1 x_2 x_3 \bar{x}_4 \vee x_2 x_4 \vee x_1 \bar{x}_2 x_3 x_4 \vee \bar{x}_2 \bar{x}_4.$$

Для этой системы матрицы  $\mathbf{F}$ ,  $\mathbf{Y}$  имеют вид

$$\mathbf{F} = \begin{array}{c} x_1 \quad x_2 \quad x_3 \quad x_4 \\ \left\| \begin{array}{cccc} - & 0 & - & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ - & 1 & - & 1 \end{array} \right\| \end{array}; \quad \mathbf{Y} = \begin{array}{c} y_1 \quad y_2 \\ \left\| \begin{array}{cc} 1 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 1 \\ 1 & 1 \end{array} \right\| \end{array}.$$

Функции системы ортогональны и заданы в ДНФ.

Определим отношения  $r$  столбцов матрицы  $\mathbf{F}$  :

номер столбца	1	2	3	4
отношение $r$	5/6	5/6	5/6	6/5

Приводим  $\mathbf{F}$  к стандартной форме:

$$\mathbf{F} = \begin{array}{c} x_1 \quad x_2 \quad x_3 \quad \bar{x}_4 \\ \left\| \begin{array}{cccc} - & 0 & - & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ - & 1 & - & 0 \end{array} \right\| \end{array} .$$

Выполняем разбиение множеств строк и столбцов матрицы в соответствии с рекомендациями из п. 1.6.1 и п. 3.5:

$$\mathbf{F} = \begin{array}{c} x_1 \quad x_3 \quad x_2 \quad \bar{x}_4 \\ \left\| \begin{array}{cc|cc} - & - & 0 & 1 \\ - & - & 1 & 0 \\ \hline 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ \hline 0 & 1 & 1 & 1 \end{array} \right\| \end{array} .$$

Как видно из приведенного разбиения, достаточно рассмотреть подстановки  $N_{0001} S_{1432}$ ,  $N_{0001} S_{3214}$ ,  $N_{0001} S_{3412}$ .

Сравнение матриц  $\mathbf{F}, \mathbf{F}' = \mathbf{F}\mathbf{T}$  осуществляется по методике, описанной в п. 3.4. Такой анализ показывает, что все три перечисленные преобразования являются симметриями заданной системы булевых функций.

Каждая функция системы (3.16) может иметь свою группу симметрий  $H_n^{(i)}$ . Следовательно, группа симметрий системы (3.16) может рассматриваться как пересечение групп симметрий отдельных функций

$$H_n^{(m)} = \bigcap_{i=1}^m H_n^{(i)} .$$

Очевидно, что группа симметрий отдельной функции, входящей в состав системы, в общем случае богаче группы симметрий всей системы.

### 3.7. Метод синтеза оптимальных бинарных программ

Интерес к программной реализации булевых функций (наряду со схемной реализацией, которая начала применяться раньше) возник в связи с применением вычислительных средств в разнообразных процессах управления. Постоянно усложняющиеся функции систем управления при ограниченной вычислительной мощности этих средств стимулируют поиски способов оптимизации управляющих программ. Основные критерии оптимальности – объем памяти, занимаемой программой, и время ее выполнения [44].

Здесь рассмотрено одно из направлений синтеза бинарных программ, основанное на разложении булевых функций, подлежащих реализации, по переменным [44; 45] и др. Это направление относится к наиболее перспективным в силу того, что, во-первых, разложение функции по переменным хорошо формализовано и, значит, легко автоматизируется. Это важно, так как решение задачи синтеза программ, а тем более поиска оптимального варианта решения, как правило, содержит комбинаторные элементы и, следовательно, относится к весьма трудоемким. Во-вторых, бинарные программы, которые получаются в результате разложения булевых функций по переменным, характеризуются, как показано в работе [44], оптимальным или близким к оптимальному временем их выполнения: обработка любого входного набора переменных выполняется такой программой не более чем за  $n$  шагов. Объем бинарной программы, определяемый числом команд условного перехода (УП) в ней, зависит от выбора последовательности разложения. Известные методы выбора являются либо приближенными, например в [46], либо эвристическими, например в [45], т.е. недостаточно обоснованными, и поэтому не гарантируют получения оптимального решения. Предлагаемый здесь метод выбора и разработанный на его основе метод синтеза бинарных программ дает лучшие результаты, о чем свидетельствуют примеры их применения, приведенные в п. 3.9. Он предназначен для синтеза древообразных бинарных программ. О бинарных программах общего вида будет сказано в п. 3.8.

Рассмотрим связь между булевой функцией  $f(x_1, x_2, \mathbf{K}, x_n)$  и реализующей ее бинарной программой с целью выявления факторов, влияющих на объем программы.

Представим программу бинарным деревом [44], в котором команды УП отображены вершинами, команды присвоения функции значений 1 ("истинно") или 0 ("ложно") – концевыми вершинами, а переходы между командами – ребрами, которым приписаны условия переходов:  $x_i$ , если условие перехода  $x_i = 1$ , или  $\bar{x}_i$ , если  $\bar{x}_i = 1$ . Ветвь дерева, которая начинается в кор-

невой вершине и заканчивается в вершине, соответствующей 1 (0), назовем 1-ветвью (0-ветвью). Припишем каждой 1-ветви слово (терм), образуемое конкатенацией букв, которыми помечены ребра на пути движения от корня дерева к соответствующей конечной вершине. Запись функции в ДНФ, каждое слагаемое которой является словом, представленным одной из 1-ветвей бинарной программы, условимся называть программно представимой (ПП). Между деревом программы и ПП записью функции существует взаимно однозначное соответствие. Поэтому ПП запись можно рассматривать как аналитическое представление бинарной программы.

Возможность аналитического представления программы указывает на то, что ее оптимизацию можно начинать с оптимизации записи реализуемой функции.

Представим объем  $V$  программы как

$$V = \sum_{i=1}^n l_i, \quad (3.17)$$

где  $l_i$  – число вершин  $i$ -го яруса дерева. Из (3.17) видно, что объем  $V$  можно сокращать, уменьшая значения  $l_i$ ,  $i = 1, 2, \dots, n$ . Так как в  $l_i$  учитываются вершины дерева, через которые проходят ветви программы длиной  $i$  и более, то, очевидно, один из способов уменьшения  $l_i$  состоит в сокращении числа таких ветвей.

В силу бинарности программы можно записать

$$l_i = 2^{(i-1)} - \sum_{k=1}^K 2^{(i-i_k-1)}, \quad (3.18)$$

где  $K$  – суммарное число 1- и 0-ветвей программы с длиной  $i_k < i$ . Из (3.18) следует, что величина  $l_i$  тем меньше, чем больше число  $K$  ветвей длиной  $i_k < i$  и чем меньше значения  $i_k$ ,  $k = 1, 2, \dots, K$ .

Таким образом, сокращение объема  $V$  может быть достигнуто путем сокращения числа и длин 1- и 0-ветвей программы. Количество 1-ветвей равно количеству конъюнкций в ПП записи булевой функции, а длина 1-ветви – рангу соответствующей конъюнкции. Следовательно, оптимизация части параметров программы может быть достигнута целиком за счет выбора ПП записи булевой функции: из требования минимальности числа и длин 1-ветвей программы вытекает требование минимальности числа и рангов конъюнкций в соответствующей ПП записи булевой функции.

Другие требования к ПП записи булевой функции сформулированы в двух очевидных утверждениях:



**Утверждение 3.5.** *ПП запись булевой функции ортогональна.*

**Утверждение 3.6.** *Хотя бы одна из переменных  $x_i$  булевой функции  $f(x_1, x_2, \mathbf{K}, x_n)$  входит в каждое слагаемое ее ПП записи.*

Понятно, что эти утверждения в равной степени относятся и к ПП записям каждой из компонент разложения [33] булевой функции. Утверждения 3.5 и 3.6, отображающие свойства ПП записи булевой функции, следует рассматривать как необходимые условия, которые должна выполнять запись функции, чтобы она была ПП.

Для ортогональной функции справедливо соотношение

$$\sum_{i=1}^{N_p} 2^{(n-r_i)} = m, \quad (3.19)$$

где  $N_p$  – число конъюнкций;  $r_i$  – ранг  $i$ -й конъюнкции;  $m$  – число точек  $n$ -мерного булевого пространства, покрытых функцией. Из (3.19) видно, что при фиксированных значениях  $m, n$  сокращения числа  $N_p$  конъюнкций можно достичь уменьшением значений  $r_i$ . С уменьшением  $N_p$  и  $r_i$ ,

$i = 1, 2, \dots, N_p$ , сокращается общее число  $N = \sum_{i=1}^{N_p} r_i$  букв в записи функции.

Отсюда следует, что для ортогональных функций задача минимизации числа  $N_p$  конъюнкций совпадает с задачей минимизации числа  $N$  букв в записи функции (известно [36], что в общем случае эти задачи, а значит, и их решения не совпадают). Решением этих задач является минимальная ортогональная ДНФ (МОДНФ). В ней значения  $N_p, N$  минимальны. При этом соотношение рангов  $r_i, i = 1, 2, \dots, n$ , в пределах их суммы, ограниченной величиной  $N$ , определяется решением уравнения (3.19). Таким образом, ПП запись функции оптимальна при совпадении ее с МОДНФ.

Согласно двум утверждениям, приведенным далее, само по себе представление булевой функции в ПП МОДНФ еще не гарантирует реализации ее в виде бинарной программы с минимальным объемом.

**Утверждение 3.7.** *Для того чтобы в бинарной программе, реализующей булеву функцию в ПП МОДНФ, были минимальными число 1-ветвей и их суммарная длина, необходимо на каждом шаге разложения выбирать для исключения переменную, которая содержится в каждой конъюнкции функции (ее компоненте разложения).*

Доказательство. В общем случае булева функция может быть представлена как

$$f(x_1, x_2, \mathbf{K}, x_n) = x_i A_i \vee \bar{x}_i B_i \vee C_i, \quad (3.20)$$

где  $A_i$  ( $B_i$ ) – часть функции, которую можно рассматривать как коэффициент при  $x_i$  ( $\bar{x}_i$ );  $C_i$  – часть функции, не зависящая от  $x_i$ . В результате разложения функции по переменной  $x_i$  получим

$$\begin{aligned} f(x_1, x_2, \mathbf{K}, x_i, \mathbf{K}, x_n) &= \\ &= x_i f(x_1, x_2, \mathbf{K}, 1, \mathbf{K}, x_n) \vee \bar{x}_i f(x_1, x_2, \mathbf{K}, 0, \mathbf{K}, x_n). \end{aligned} \quad (3.21)$$

Обозначим компоненты разложения [33] символами  $f_{i_+}, f_{i_-}$ :

$$f_{i_+} = f(x_1, x_2, \mathbf{K}, 1, \mathbf{K}, x_n); \quad (3.22)$$

$$f_{i_-} = f(x_1, x_2, \mathbf{K}, 0, \mathbf{K}, x_n). \quad (3.23)$$

Программная интерпретация разложения (3.21) – это разбиение синтезируемой программы на две подпрограммы, реализующие функции  $f_{i_+}, f_{i_-}$ , с выделением одной элементарной команды условного перехода по  $x_i$  к одной или другой подпрограмме. Так же интерпретируется и каждый последующий шаг разложения. Разложение каждой из конъюнкций по ее последней переменной интерпретируется в программе командой условного перехода к присвоению функции значения 1 ("истинно") или 0 ("ложно"). Таким образом, количество команд в программе определяется числом шагов разложения функции. Понятно, что это количество тем меньше, чем больше число конъюнкций, у которых удастся уменьшить число переменных (понижить ранг) за один шаг разложения, т.е. за счет добавления в программе одной команды условного перехода. Проследим, как это число зависит от выбора последовательности переменных, по которым выполняется разложение.

Ранг каждой конъюнкции  $A_i, B_i$  за один шаг разложения уменьшается на единицу, а ранги конъюнкций составляющей  $C_i$  остаются без изменения. Число конъюнкций составляющих  $A_i, B_i$  либо сохраняется после шага разложения, либо сокращается, если на этом шаге хотя бы для одной конъюнкции было выполнено разложение по его последней переменной. Число конъюнкций  $C_i$ , как видно из (3.20) – (3.23), после шага разложения удваивается. Следовательно, наличие составляющих  $C_i$  в любой компоненте разложения не способствует уменьшению числа таких шагов. Выбор для ис-

ключения переменной, которая содержится в каждой конъюнкции, означает отсутствие в  $f_{i_+}, f_{i_-}$  составляющей  $C_i$ . В соответствии с утверждением 3.6 в ПП МОДНФ, как и в каждой компоненте ее разложения, всегда найдется хотя бы одна такая переменная.

Так как на каждом шаге разложения ранг каждой конъюнкции уменьшается на единицу, длина каждой 1-ветви равна рангу соответствующей конъюнкции. количество 1-ветвей в синтезированной программе при отсутствии составляющей  $C_i$  равно числу  $N_p$  конъюнкций в исходной функции. Тогда из минимальности числа  $N_p$  конъюнкций ПП МОДНФ и минимальности суммы рангов ее конъюнкций  $N$  следует, что построенная по ней древовидная программа будет иметь минимальное число 1-ветвей, равное  $N_p$ , и минимальную суммарную длину 1-ветвей, равную  $N$ . Утверждение доказано.

Булева функция, представленная в ПП МОДНФ, как любая ее компонента разложения, может иметь не единственную переменную  $x_i$ , содержащуюся в каждой ее конъюнкции. Пусть  $M$  – множество таких переменных функции (ее компоненты разложения), а  $M_0$  – подмножество  $M$  переменных, вошедших во все конъюнкции только с отрицанием или только без отрицания.

**У т в е р ж д е н и е 3. 8.** *Для того чтобы число ветвей бинарной программы и их длины были минимальными, необходимо на каждом шаге разложения булевой функции, представленной в ПП МОДНФ, выбирать для исключения переменную  $x_i \in M_0 \subseteq M$ , если  $M_0$  не пусто.*

**Д о к а з а т е л ь с т в о.** Если на некотором шаге разложения выбрана переменная  $x_i \in M_0$ , одна из двух ветвей, которые исходят из вершины дерева программы, представляющей УП по  $x_i$ , на этом заканчивается присвоением функции значения 0 (1). Допустим теперь, что разложение по  $x_i \in M_0$  отложено хотя бы на один шаг, т.е. на данном шаге выбрана для исключения переменная  $x_i \notin M_0$ . Тогда переменная  $x_i$  с инверсией или без инверсии войдет в каждое слагаемое обеих компонент разложения функции по  $x_i$ . Если на новом шаге выбрать для исключения  $x_i \in M_0$ , то теперь две ветви закончатся присвоением 0, вместо одной, как в предыдущем случае, и каждая из них будет длиннее ветви, полученной в предыдущем случае, на одну вершину. Таким образом, отложив разложение, мы обусловили увеличение числа и длин 0-ветвей. Утверждение доказано.

Полезно рассмотреть геометрическую интерпретацию ПП МОДНФ, наглядно иллюстрирующую роль последней в определении пути к минимальной по объему бинарной программе.

Разложению функции  $f(x_1, x_2, \mathbf{K}, x_n)$  по  $x_i$  соответствует разбиение  $n$ -мерного булевого пространства, где она представлена помеченными точками, на два  $(n-1)$ -мерных с исключением переменной  $x_i$ . При этом часть помеченных точек попадает в одно  $(n-1)$ -мерное подпространство, часть – в другое. Последовательность разложения функции, соответствующая 1-ветви, интерпретируется выделением с помощью таких разбиений подпространства, совпадающего с одной из покрывающих граней. Минимизация объема программы интерпретируется минимизацией числа шагов разбиения, необходимого для выделения каждой из покрывающих граней.

Геометрическая интерпретация ПП МОДНФ – это совокупность минимального числа непересекающихся граней, покрывающих помеченные точки  $n$ -мерного пространства. Грани в данном случае являются максимальными [33]. Такое покрытие наиболее благоприятно для выделения искомого граней за минимальное число шагов. В этом случае критерий из утверждения 3.7 указывает путь для выделения одних граней без разрушения других, а критерий из утверждения 3.8 обеспечивает своевременное (при первой же возможности) отсечение "пустого" полупространства, т.е. не содержащего искомого покрывающих граней. Тем самым критерии из утверждений 3.7, 3.8 обеспечивают сокращение числа шагов разложения, а значит, и объема программы.

Рассмотрим случай, когда подлежащая программной реализации функция (или ее компонента разложения), представленная в МОДНФ, не содержит ни одной переменной, вошедшей в каждую конъюнкцию, т.е. в соответствии с утверждением 3.6 не является ПП. Тогда недостающую переменную можно ввести в любую конъюнкцию, например, с помощью подстановки

$$p_i = x_i p_i \vee \bar{x}_i p_i, \quad (3.24)$$

не меняющей функцию и не нарушающей ее ортогональности. Из (3.24) видно, что это преобразование в общем случае ведет к увеличению как числа  $N_p$  конъюнкций в записи функции, так и числа букв в ней, а значит, и объема реализующей программы. Понятно, что при разложении функции следует стремиться по возможности уменьшить количество таких преобразований. В этом случае может помочь критерий из утверждения 3.7, но при условии, что под  $M$  подразумевается множество переменных, которые вошли не во все конъюнкции функции, заданной МОДНФ (или ее компоненты

разложения), а в максимальное их число. В геометрической интерпретации рассмотренная ситуация означает, что выделение покрывающих граней невозможно без разрушения других граней. Критерий из утверждения 3.7 (при указанной поправке к нему) обеспечивает выбор, при котором число таких разрушений минимально. Таким образом, представление реализуемой функции в МОДНФ при синтезе минимальных по объему бинарных программ полезно даже тогда, когда она не является ПП.

Заметим, что МОДНФ, как и МДНФ, в общем случае не единственна для заданной функции. Каждое из множеств  $M$ ,  $M_0$  также может содержать более одного элемента. Поэтому ограниченный перебор при поиске программных реализаций функции с минимальным объемом остается неизбежным. Тем более, что решение этой задачи в общем случае также неоднозначно. Но предлагаемый подход позволяет резко сократить перебор без риска потерять искомые оптимальные программы.

Изложенные соображения являются обоснованием метода поиска минимальных по объему бинарных программ, состоящего из таких этапов:

1. Приведение заданной функции к МОДНФ (одно из возможных решений этой задачи рассмотрено в п. 3.10).

2. Разложение каждой из МОДНФ по переменным, выбранным на каждом шаге разложения по критериям из утверждений 3.7, 3.8.

На каждом из этих этапов построения искомой бинарной программы возможна неоднозначность. Если рассмотреть все возможные варианты, получим множество оптимальных и близких к оптимальным по объему программ, так как для ограничения перебора предложено использовать необходимые, но не достаточные признаки оптимальности.

При реализации системы булевых функций  $f_i(x_1, x_2, \mathbf{K}, x_n)$ ,  $i = 1, 2, \dots, m$ , сокращение объема программы достигается совмещением в ней проверки истинности совпадающих подфункций нескольких функций системы. Это выполняется наиболее эффективно, если перед реализацией преобразовать систему функций приведением к МОДНФ каждого пересечения по  $m, m-1, \dots, 1$  покрытий функций системы. После такого преобразования система функций может быть представлена [36] совокупностью системы МОДНФ, в которой отображены полученные минимальные ортогональные покрытия без повторов, и системы инцидентностей, отображающей отношение каждого покрытия системы МОДНФ к функциям  $f_i(x_1, x_2, \mathbf{K}, x_n)$ ,  $i = 1, 2, \dots, m$ .

Нетрудно убедиться, что для системы МОДНФ справедливы утверждения (3.5) – (3.8). Доказательства аналогичны приведенным. Поэтому при

выборе последовательности разложения функций системы по переменным можно использовать предложенный метод.

### 3.8. Применение информации о симметриях булевой функции при синтезе бинарных программ

Рассмотрим структуру множества программных реализаций булевой функции. Пусть  $H_n = \{h\}$  – группа автоморфизмов этой функции. Как уже говорилось, действие оператора  $h = N_\alpha S_\beta \in H_n$  на функцию  $f(x_1, x_2, \mathbf{K}, x_i, \dots, x_j, \dots, x_n)$  заключается в переименовании ее переменных с инверсированием ( $x_i$  в  $\bar{x}_j$  и  $x_j$  в  $\bar{x}_i$ ) или без него ( $x_i$  в  $x_j$  и  $x_j$  в  $x_i$ ).

**Утверждение 3.9.** Действием симметричных преобразований  $h \in H_n$  функции  $f(x_1, x_2, \mathbf{K}, x_n)$  множество бинарных программ, ее реализующих, разбивается на классы эквивалентности.

**Доказательство.** Из утверждения 3.1 следует, что множество всех возможных записей булевой функции разбивается на классы эквивалентности (изоморфизма) действием  $h \in H_n$ : для любой записи найдется в этом множестве такая, что приводится к первой с помощью  $h \in H_n$ . Это же относится к ПП записям функции. Но между ПП записью функции и ее программной реализацией в виде бинарной программы существует взаимно однозначное соответствие. Тогда разбиением множества ПП записей функции на классы эквивалентности по  $h \in H_n$  индуцируется разбиение ее программных реализаций, что и требовалось доказать.

Нетрудно построить правила перехода от бинарной программы к эквивалентной ей по  $h \in H_n$ . Для этого рассмотрим преобразование УП "если  $x_i = 1$ , перейти к выполнению  $k_a$ -й команды; если  $x_i = 0$ , –  $k_b$ -й команды", в дальнейшем для краткости обозначаемое "УП ( $x_i$ )  $k_a, k_b$ ". Ему соответствует шаг разложения функции  $f(x_1, x_2, \mathbf{K}, x_i, \dots, x_j, \dots, x_n)$  по  $x_i$ :

$$f(\mathbf{K}, x_i, \dots, x_j, \dots) = x_i f_{i_+}(\mathbf{K}, 1, \dots, x_j, \dots) \vee \bar{x}_i f_{i_-}(\mathbf{K}, 0, \dots, x_j, \dots). \quad (3.25)$$

Пусть оператор  $h \in H_n$  реализует перестановку  $x_i$  и  $x_j$ . Из инвариантности функции к этой перестановке следует

$$h[f(\mathbf{K}, x_i, \dots, x_j, \dots)] = x_j f_{i_+}(\mathbf{K}, x_i, \dots, 1, \dots) \vee \bar{x}_j f_{i_-}(\mathbf{K}, x_i, \dots, 0, \dots). \quad (3.26)$$

Сравнив правые и левые части (3.25), (3.26), заключаем, что изменение в функции под действием  $h \in H_n$  интерпретируется в программе заменой фрагмента "УП  $(x_i)k_a, k_b$ " фрагментом "УП  $(x_j)k_a, k_b$ ". Пусть теперь оператор  $h \in H_n$  меняет местами  $x_i$  и  $\bar{x}_j$ . Тогда

$$h[f(\mathbf{K}, x_i, \dots, x_j, \dots)] = \bar{x}_j f_{i_+}(\mathbf{K}, x_i, \dots, 0, \dots) \vee x_j f_{i_-}(\mathbf{K}, x_i, \dots, 1, \dots),$$

что соответствует замене фрагмента "УП  $(x_i)k_a, k_b$ " фрагментом "УП  $(\bar{x}_j)k_a, k_b$ ", или, что то же самое, фрагментом "УП  $(x_j)k_b, k_a$ ". Последнее означает замену переменной  $x_i$  на  $x_j$  с одновременной заменой адресов команд  $k_b$  и  $k_a$  на противоположные.

Из рассмотренных правил преобразования программы следует, что под действием  $h \in H_n$  не меняется число УП в программе, а значит, и ее объем. Следовательно, все программные реализации булевой функции, относящиеся к одному классу эквивалентности по ее симметриям, характеризуются одинаковым объемом. Очевидно, что из множества минимальных по объему программных реализаций заданной функции достаточно найти одну, остальные могут быть построены преобразованием ее по рассмотренным правилам. Подобное построение равноценных по объему программ, принадлежащих к общему классу эквивалентности по  $h \in H_n$ , может быть полезным при оптимизации программ не только по объему, но и по другим критериям.

Пусть  $M_s$  —  $h$ -класс, т.е. множество переменных функции  $f(x_1, x_2, \dots, \mathbf{K}, x_n)$ , которые могут меняться местами с инверсированием или без него под действием подстановки  $h \in H_n$ . Разумеется, что в множестве  $M(M_0)$  может быть не единственный  $h$ -класс. Построение одного представителя от класса изоморфных программ обеспечивается выбором для исключения единственной переменной от каждого класса эквивалентности.

Заметим, что каждая компонента разложения  $f_{i_+}$  ( $f_{i_-}$ ) характеризуется своим множеством  $H_{(n-k)i_+}$  ( $H_{(n-k)i_-}$ ),  $k = 1, 2, \dots, n$ , симметричных преобразований. Следовательно, утверждение 3.9 относится в равной мере к множеству подпрограмм, реализующих каждую компоненту разложения. Поэтому все, что сказано об эквивалентности программ по  $h \in H_n$ , относится к эквивалентности подпрограмм по симметриям соответствующих компонент разложения. Из п. 3.3 следует, что при больших значениях  $n$  чаще встречаются обыкновенные функции. Но по мере уменьшения числа переменных при разложении функции число симметричных компонент разложе-

ния увеличивается. Значит рассмотренное применение информации о симметриях полезно при программной реализации любых функций.

В предложенном методе синтеза бинарных программ выбор последовательности исключения переменных на каждом шаге разложения функции сведен к задаче синтеза  $Q$ -структуры единичного порядка сложности в следующей постановке: выбрать для отождествления с одной переменной ( $|I| = |J| = 1$ ), по которой предстоит выполнить разложение, один из элементов множества  $M$  или  $M_0$  ( $L_i = M$  или  $L_i = M_0$ ,  $i = |I| = 1$ ) при условии, что на нем действует группа автоморфизмов.

Рассмотренный метод построения бинарных программ обеспечивает порождение всех минимальных по объему программных реализаций функции, имеющих древовидную структуру. Один из известных [47] способов дальнейшего сокращения объема программы заключается в объединении между собой повторяющихся подпрограмм найденной древовидной программы. На дереве программы это интерпретируется объединением одинаковых поддеревьев, начиная с одинаковых вершин, которые являются корневыми для этих поддеревьев. Эффект от такого приема тем сильнее, чем большее число вершин удастся объединить. На этом этапе оптимизации программ полезно еще одно применение информации о симметриях булевой функции: число объединенных вершин может быть увеличено и с помощью симметричных преобразований  $h \in H_{(n-k)_+}$ , или  $h \in H_{(n-k)_-}$ ,  $i \in \{1, 2, \dots, n\}$ ,  $k = 1, 2, \dots, n$ , если их применить к соответствующим подпрограммам найденных древовидных программ.

Рассмотрим примеры применения информации о симметриях булевой функции.

### 3. 9. Примеры синтеза оптимальных программных реализаций булевой функции

Для иллюстрации предлагаемого метода синтеза бинарных программ с минимальным объемом и его преимуществ перед известными методами взяты примеры, выбранные авторами этих методов [46; 45] .

**Пример 3.15.** Пусть задана функция [46]

$$f(x_1, x_2, \mathbf{K}, x_5) = x_1 x_2 \bar{x}_3 \bar{x}_4 \vee \bar{x}_1 x_2 \bar{x}_4 \vee x_3 x_4 x_5 \vee \bar{x}_3 \bar{x}_4 \bar{x}_5 ,$$

подлежащая реализации минимальной по объему бинарной программой.

Приведем ее к МОДНФ:

$$f(x_1, x_2, \mathbf{K}, x_5) = x_1 x_2 \bar{x}_3 \bar{x}_4 \vee \bar{x}_1 x_2 \bar{x}_4 \vee x_3 x_4 x_5 \vee \bar{x}_2 \bar{x}_3 \bar{x}_4 \bar{x}_5 .$$



В данном случае эта задача решается однозначно. Для этой функции  $M = \{x_4\}$ , поэтому выбираем для исключения  $x_4$ . Дальнейшие промежуточные преобразования приведем без пояснений:

$$f_{4_+} = x_3 x_5, \quad M = M_s = \{x_3, x_5\}, \quad \text{выбираем } x_3; \quad f_{3_+} = x_5, \quad M = \{x_5\}, \quad f_{5_+} = 1.$$

$$f_{4_-} = x_1 x_2 \bar{x}_3 \vee \bar{x}_1 x_2 \vee \bar{x}_2 \bar{x}_3 \bar{x}_5, \quad M = \{x_2\};$$

$$f_{2_+} = x_1 \bar{x}_3 \vee \bar{x}_1, \quad M = \{x_1\}; \quad f_{1_+} = \bar{x}_3, \quad f_{3_-} = 1. \quad f_{1_-} = 1.$$

$$f_{2_-} = \bar{x}_3 \bar{x}_5, \quad M = M_s = \{x_3, x_5\}, \quad \text{выбираем } x_3; \quad f_{3_-} = \bar{x}_5, \quad M = \{x_5\}, \quad f_{5_-} = 1.$$

Результат разложения, представлен в виде ПП записи функции, каждой конъюнкции которой соответствует 1-ветвь программы:

$$f(x_1, x_2, \mathbf{K}, x_5) = x_4 \underline{x_3 x_5} \vee \bar{x}_4 x_2 \bar{x}_1 \vee \bar{x}_4 x_2 x_1 \bar{x}_3 \vee \bar{x}_4 \bar{x}_2 \bar{x}_3 \bar{x}_5.$$

Подчеркнутые подфункции симметричны, значит, в каждой из них может быть выбрана противоположная последовательность сомножителей. Всего получается четыре варианта, соответствующие четырем вариантам минимальных программ с древесной структурой, имеющих одинаковые объемы. Из них только один вариант подфункции  $\bar{x}_5 \bar{x}_3$  допускает объединение УП по  $x_3$  в соответствующей программе. После замены подчеркнутой подфункции  $\bar{x}_3 \bar{x}_5$  симметричной подфункцией  $\bar{x}_5 \bar{x}_3$  остается две изоморфные записи функции, которым соответствуют две программы, эквивалентные по симметриям заданной функции. Приведем одну из них после объединения команд УП по  $x_3$ :

- |                       |  |
|-----------------------|--|
| 1. УП ( $x_4$ ) 2, 3. | 6. УП ( $x_5$ ) 9, 7.                          |
| 2. УП ( $x_3$ ) 4, 9. | 7. УП ( $x_3$ ) 9, 8.                          |
| 3. УП ( $x_2$ ) 5, 6. | 8. $f(x_1, x_2, \mathbf{K}, x_5) = 1$ , конец. |
| 4. УП ( $x_5$ ) 8, 9. | 9. $f(x_1, x_2, \mathbf{K}, x_5) = 0$ , конец. |
| 5. УП ( $x_1$ ) 7, 8. |  |

Из этой программы получается эквивалентная, если в командах 2 и 4 поменять местами переменные  $x_3$  и  $x_5$ .

Заметим, что методика, предложенная в [46], при решении этого примера приводит к программе с восемью командами УП.

**Пример 3.16.** Пусть требуется построить минимальную по числу команд УП программу, реализующую функцию [45]:

$$f(x_1, x_2, \mathbf{K}, x_5) = x_1 \bar{x}_2 x_3 \vee \bar{x}_1 \bar{x}_3 x_4 \vee x_1 x_3 \bar{x}_5 \vee x_1 x_2 x_4 \vee \bar{x}_2 x_3 x_5 \vee \bar{x}_3 \bar{x}_4 x_5.$$

Соответствующие МОДНФ:

$$\begin{aligned}
 f' (x_1, x_2, \mathbf{K}, x_5) &= \\
 &= x_1 x_2 x_4 \vee x_1 x_2 x_3 \bar{x}_4 \bar{x}_5 \vee x_1 \bar{x}_2 x_3 \vee \bar{x}_1 \bar{x}_2 x_3 x_5 \vee \bar{x}_3 \bar{x}_4 x_5 \vee \bar{x}_1 \bar{x}_3 x_4 ; \\
 f'' (x_1, x_2, \mathbf{K}, x_5) &= \\
 &= x_1 x_2 x_4 x_5 \vee x_1 x_2 \bar{x}_3 x_4 \bar{x}_5 \vee x_1 x_3 \bar{x}_5 \vee \bar{x}_2 x_3 x_5 \vee \bar{x}_3 \bar{x}_4 x_5 \vee \bar{x}_1 \bar{x}_3 x_4 ; \\
 f''' (x_1, x_2, \mathbf{K}, x_5) &= \\
 &= x_1 x_2 \bar{x}_3 x_4 \vee x_1 x_2 x_3 x_4 x_5 \vee x_1 x_3 \bar{x}_5 \vee \bar{x}_2 x_3 x_5 \vee \bar{x}_3 \bar{x}_4 x_5 \vee \bar{x}_1 \bar{x}_3 x_4 .
 \end{aligned}$$

Из них только одна функция  $f''' (x_1, x_2, \mathbf{K}, x_5)$  удовлетворяет условию утверждения 3.6, т.е. необходимому условию программной представимости. Поэтому начнем с разложения этой функции:

$$M = \{ x_3 \};$$

$$f_{3_+} = x_1 x_2 x_4 x_5 \vee x_1 \bar{x}_5 \vee \bar{x}_2 x_5, \quad M = \{ x_5 \};$$

$$f_{5_+} = x_1 x_2 x_4 \vee \bar{x}_2, \quad M = \{ x_2 \}; \quad f_{5_-} = x_1, \quad M = \{ x_1 \}, \quad f_{1_+} = 1.$$

$$f_{2_+} = x_1 x_4, \quad M = M_s = \{ x_1, x_4 \}, \text{ выбираем } x_1; \quad M = \{ x_4 \}; \quad f_{4_+} = 1. \quad f_{2_-} = 1.$$

$$f_{3_-} = x_1 x_2 \bar{x}_3 x_4 \vee \bar{x}_3 \bar{x}_4 x_5 \vee \bar{x}_1 \bar{x}_3 x_4, \quad M = \{ x_4 \};$$

$$f_{4_+} = x_1 x_2 \vee \bar{x}_1, \quad M = \{ x_1 \}; \quad f_{1_+} = x_2, \quad M = \{ x_2 \}; \quad f_{2_+} = 1. \quad f_{1_-} = 1.$$

$$f_{4_-} = x_5, \quad M = \{ x_5 \}; \quad f_{5_+} = 1.$$

В процессе разложения по предложенному методу убеждаемся, что компоненты разложения, полученные на всех этапах преобразования функции, удовлетворяют условию утверждения 3.6. Значит, запись  $f''' (x_1, x_2, \mathbf{K}, x_5)$  является ПП, и, следовательно, нет смысла рассматривать другие МОДНФ, так как результат будет заведомо хуже.

Результат разложения в ПП форме:

$$f''' (x_1, x_2, \mathbf{K}, x_5) = x_3 x_5 \bar{x}_2 \vee x_3 x_5 x_2 \underline{x_1 x_4} \vee x_3 \bar{x}_5 x_1 \vee \bar{x}_3 x_4 x_1 x_2 \vee \bar{x}_3 x_4 \bar{x}_1 \vee \bar{x}_3 \bar{x}_4 x_5.$$

Подчеркнутая подфункция симметрична, и, следовательно, возможны два варианта последовательностей переменных в ней. Заменяем ее симметричной подфункцией  $x_4 x_1$ , при которой возможно объединение УП по  $x_1$  в соответствующей программе. В результате получается программа с девятью УП:

- |                         |  |
|-------------------------|--|
| 1. УП ( $x_3$ ) 2, 3.   | 7. УП ( $x_1$ ) 9, 10.                   |
| 2. УП ( $x_5$ ) 4, 5.   | 8. УП ( $x_5$ ) 10, 11.                  |
| 3. УП ( $x_4$ ) 7, 8.   | 9. УП ( $x_2$ ) 10, 11.                  |
| 4. УП ( $x_2$ ) 6, 10.  | 10. $f(x_1, x_2, \mathbf{K}, x_5) = 1$ . |
| 5. УП ( $x_1$ ) 10, 11. | 11. $f(x_1, x_2, \mathbf{K}, x_5) = 0$ . |
| 6. УП ( $x_4$ ) 5, 11.  |  |

Эта же функция в [45] реализована по предложенному там методу программой с десятью командами УП.

**Пример 3.17.** Пусть подлежащая программной реализации функция

$$f(x_1, x_2, \mathbf{K}, x_5) = x_1 x_2 \vee x_1 \bar{x}_3 x_4 \vee x_2 x_5 \vee \bar{x}_3 x_4 x_5,$$

симметрична относительно переменных  $x_1$  и  $x_5$ ,  $\bar{x}_3$  и  $x_4$ . Она имеет две МОДНФ:

$$f'(x_1, x_2, \mathbf{K}, x_5) = x_1 x_2 \vee x_1 \bar{x}_2 \bar{x}_3 x_4 \vee \bar{x}_1 x_2 x_5 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 x_4 x_5,$$

$$f''(x_1, x_2, \mathbf{K}, x_5) = x_1 x_2 \bar{x}_5 \vee x_1 \bar{x}_2 \bar{x}_3 x_4 \bar{x}_5 \vee x_2 x_5 \vee \bar{x}_2 \bar{x}_3 x_4 x_5.$$

Одна МОДНФ приводится к другой перестановкой  $x_1$  и  $x_5$ .

Для разложения выберем первую МОДНФ.  $M = \{x_1, x_2\}$ .

Выбираем  $x_1$ :  $f_{1+} = x_2 \vee \bar{x}_2 \bar{x}_3 x_4$ ,  $M = \{x_2\}$ ,  $f_{2+} = 1$ ;  $f_{2-} = \bar{x}_3 x_4$ ;

$$f_{1-} = x_2 x_5 \vee \bar{x}_2 \bar{x}_3 x_4 x_5, \quad M = \{x_2, x_5\}, \quad M_0 = \{x_5\}, \quad f_{5+} = x_2 \vee \bar{x}_2 \bar{x}_3 x_4, \quad f_{5-} = 0;$$

$$M = \{x_2\}, \quad f_{2+} = 1, \quad f_{2-} = \bar{x}_3 x_4.$$

Выбираем  $x_2$ :  $f_{2+} = x_1 \vee \bar{x}_1 x_5$ ;  $M = \{x_1\}$ ,  $f_{1+} = 1$ ,  $f_{1-} = x_5$ ;

$$f_{2-} = x_1 \bar{x}_3 x_4 \vee \bar{x}_1 \bar{x}_3 x_4 x_5; \quad M = \{x_1, x_3, x_4\}, \quad M_0 = \{x_3, x_4\}, \quad \text{выбираем } x_4:$$

$$f_{4+} = x_1 \bar{x}_3 \vee \bar{x}_1 \bar{x}_3 x_5, \quad f_{4-} = 0; \quad M = \{x_1, x_3\}, \quad M_0 = \{x_3\}, \quad f_{3+} = 0, \quad f_{3-} = x_1 \vee \bar{x}_1 x_5.$$

По результатам разложения строим две неэквивалентные по  $h \in H_n$  программы с пятью УП.

Первая программа:

- |                       |   |
|-----------------------|---|
| 1. УП ( $x_1$ ) 2, 3. | 5. УП ( $x_3$ ) 7, 6.                           |
| 2. УП ( $x_2$ ) 6, 4. | 6. $f(x_1, x_2, \mathbf{K}, x_5) = 1$ , конец . |
| 3. УП ( $x_5$ ) 2, 7. | 7. $f(x_1, x_2, \mathbf{K}, x_5) = 0$ , конец . |
| 4. УП ( $x_4$ ) 5, 7. |   |

Вторая программа:

- |                       |   |
|-----------------------|---|
| 1. УП ( $x_2$ ) 2, 3. | 5. УП ( $x_3$ ) 7, 2.                           |
| 2. УП ( $x_1$ ) 6, 4. | 6. $f(x_1, x_2, \mathbf{K}, x_5) = 1$ , конец . |
| 3. УП ( $x_4$ ) 5, 7. | 7. $f(x_1, x_2, \mathbf{K}, x_5) = 0$ , конец . |
| 4. УП ( $x_5$ ) 6, 7. |   |

От каждой из этих программ можно перейти к трем эквивалентным по  $h \in H_n$  программам, переставив  $x_1$  с  $x_5$  и (или)  $\bar{x}_3$  с  $x_4$ . Всего получается восемь равноценных по числу УП программных реализаций функции  $f(x_1, x_2, \mathbf{K}, x_5)$ .

**Пример 3.18.** Задана система функций, подлежащая программной реализации:

$$y_1 = \bar{x}_2 \vee \bar{x}_1 x_2 x_3;$$

$$y_2 = \bar{x}_2 \bar{x}_4 \vee x_1 x_2 \bar{x}_4 \vee \bar{x}_1 x_2 x_3.$$

После приведения к МОДНФ она принимает вид

$$y_1 = \bar{x}_2 \bar{x}_4 \vee \bar{x}_1 x_2 x_3 \vee \bar{x}_2 x_4;$$

$$y_2 = \bar{x}_2 \bar{x}_4 \vee x_1 x_2 \bar{x}_4 \vee \bar{x}_1 x_2 x_3.$$

Систему МОДНФ составляют  $\bar{x}_2 \bar{x}_4$ ,  $\bar{x}_1 x_2 x_3$ ,  $x_1 x_2 \bar{x}_4$ ,  $\bar{x}_2 x_4$ .

По этой системе с помощью критерия из утверждений 3.7, 3.8 определяем последовательность разложения заданных функций:

$$y_1 = x_2 \bar{x}_1 x_3 \vee \bar{x}_2 \bar{x}_4 \vee \bar{x}_2 x_4;$$

$$y_2 = x_2 x_1 \bar{x}_4 \vee x_2 \bar{x}_1 x_3 \vee \bar{x}_2 \bar{x}_4.$$

После объединения совпадающих команд получим следующую программу:

- |                        |                   |
|------------------------|-------------------|
| 1. УП ( $x_2$ ) 2, 5.  | 6а. Перейти к 4.  |
| 2. УП ( $x_1$ ) 6, 3.  | 7. $y_1 = 1$ .    |
| 3. УП ( $x_3$ ) 7, 8.  | 7а. Перейти к 10. |
| 4. УП ( $x_4$ ) 9, 10. | 9. $y_2 = 0$ .    |
| 5. $y_1 = 1$ .         | 8. $y_1 = 0$ .    |
| 5а. Перейти к 4.       | 10. $y_2 = 1$ .   |
| 6. $y_1 = 0$ .         |                   |

### 3.10. Приведение булевой функции к минимальной ортогональной форме

Методы ортогонализации булевой функции различны для разных форм ее задания. Меньше всего от МОДНФ отличается МДНФ. В основе преобразования МДНФ в МОДНФ лежит подстановка (3.24), которую можно рассматривать как элемент преобразования функции в процессе ее разложения по переменной  $x_i$ , обуславливающей увеличение числа букв или одновременно букв и слов в преобразуемой записи функции. Поскольку критерии утверждений 3.7, 3.8 указывают способ разложения, при котором это увеличение минимально, применение их при разложении непосредственно МДНФ дает неплохие результаты.

Наиболее сложен случай представления функций в совершенной дизъюнктивной нормальной форме или помеченными точками в  $n$ -мерном булевом пространстве. Этот случай рассмотрен здесь.

Пусть  $m$  – число помеченных точек  $n$ -мерного булевого пространства, которыми задана функция,  $m \leq 2^n$ . Рассмотрим некоторое ортогональное покрытие [36] заданной функции множеством  $i$ -кубов,  $i = 1, 2, \dots, W$  ( $W$  – максимальная размерность кубов, которые можно построить на  $m$  точках). Пусть в этом покрытии  $k_i$  – мощность множества попарно ортогональных кубов  $i$ -го порядка,  $k_i \geq 0$ ,  $i = 1, 2, \dots, W$ . Тогда для этого покрытия имеет место равенство

$$k_0 + 2k_1 + \dots + 2^i k_i + \dots + 2^W k_W = m. \quad (3.27)$$

Условимся решение уравнения (3.27) характеризовать вектором

$$\mathbf{k} = (k_0, k_1, \dots, k_W).$$

Координатами вектора  $\mathbf{k}$  определяется число  $C(\mathbf{k})$  букв в функции, представленной в соответствующей ему ортогональной форме (цена покрытия):

$$C(\mathbf{k}) = k_0 n + k_1 (n-1) + \dots + k_W (n-W). \quad (3.28)$$

Поиск минимального ортогонального покрытия заданной функции  $f(x_1, x_2, \mathbf{K}, x_n)$  предлагается [48] выполнить по следующей схеме:

1. Найти решение уравнения (3.27) с минимальной ценой  $C(\mathbf{k})$ .
2. Проверить, реализуется ли это решение на комплексе кубов, заданных функцией  $f(x_1, x_2, \mathbf{K}, x_n)$ . Если реализуется, искомое решение найдено.

3. Построить очередное решение уравнений (3.27) с ценой  $C(\mathbf{k})$ , равной цене предыдущего варианта или минимально отличающейся от нее. Перейти к п. 2.

Пункты 2 и 3 необходимо повторять до тех пор, пока не будет найдено реализуемое решение  $\mathbf{k}$ .

При построении МОДНФ по указанной схеме предполагается, что известны решения следующих задач:

Первая – построение (генерирование) вариантов решения уравнения (3.27).

Вторая – упорядочение решений  $\mathbf{k}$  по цене покрытия  $C(\mathbf{k})$ .

Третья – проверка реализуемости вектора  $\mathbf{k}$  на комплексе кубов, заданном функцией  $f(x_1, x_2, \mathbf{K}, x_n)$ .

Решение первой задачи. Введем вектор

$$\mathbf{A} = (2\alpha_1, 2\alpha_2 - \alpha_1, \dots, 2\alpha_{(i+1)} - \alpha_i, \dots, 2\alpha_W - \alpha_{(W-1)}, -\alpha_W), \quad (3.29)$$

где  $\alpha_i \in Z$ .

Нетрудно убедиться, что вектор  $\mathbf{k} + \mathbf{A}$  является решением уравнения (3.27), если его решением является вектор  $\mathbf{k}$ . Это проверяется непосредственной подстановкой  $\mathbf{k}' = \mathbf{k} + \mathbf{A}$  вместо  $\mathbf{k}$  в левую часть равенства (3.27).

В [48] показано, что множество векторов  $\mathbf{A}$  образует аддитивную группу  $G$  с образующими

$$\mathbf{a}_i = (0, 0, \dots, 2, -1, 0, \dots, 0), \quad (3.30)$$

где число 2 является  $(i-1)$ -й, а  $(-1)$  –  $i$ -й координатой вектора  $\mathbf{a}_i$ ,  $i = 1, 2, \dots, W$ :

$$\mathbf{A} = \sum_{i=1}^W \alpha_i \mathbf{a}_i. \quad (3.31)$$

Единичным элементом группы служит нулевой вектор, а элементом, обратным вектору  $\mathbf{A}$ , является вектор  $\mathbf{A}^{-1} = \sum_{i=1}^W (-\alpha_i) \mathbf{a}_i$ .

Группа  $G$ , очевидно, позволяет построить множество всех возможных решений (3.27), имея одно произвольное решение. Однако эти решения могут иметь как положительные, так и отрицательные координаты. Поскольку отрицательная координата не может быть мощностью множества  $i$ -кубов, потребуем от вектора  $\mathbf{k}'$ , чтобы его координаты не были отрицательными:

$$\begin{aligned}
&k_0 + 2\alpha_1 \geq 0; \\
&\dots \\
&k_i + 2\alpha_{(i+1)} - \alpha_i \geq 0; \\
&\dots \\
&k_W - \alpha_W \geq 0.
\end{aligned}
\tag{3.32}$$

Перебирая все  $\alpha_i$ ,  $i = 1, 2, \dots, W$ , удовлетворяющие (3.32), получаем набор векторов  $\mathbf{A}$ , под действием которых исходный вектор  $\mathbf{k}$  преобразуется в вектор  $\mathbf{k}'$  с неотрицательными координатами, что обеспечивает полноту множества решений первой задачи.

Теперь остановимся на выборе исходного вектора  $\mathbf{k}$ . Как видно из (3.28), уменьшению цены покрытия  $C(\mathbf{k})$  способствует выбор для покрытия  $m$  точек преимущественно  $i$ -кубов больших размерностей, а увеличению  $C(\mathbf{k})$  –  $i$ -кубов малых размерностей. Поэтому решение  $\mathbf{k}^*$  уравнения (3.27) с минимальной ценой  $C(\mathbf{k}^*)$  строим следующим образом. Максимальное значение координаты  $k_W$  определяем как максимальное число  $W$ -мерных ортогональных кубов, которое можно построить на  $m$  точках  $n$ -мерного логического пространства:

$$k_W = [m/2^W]. \tag{3.33}$$

Максимальное значение координаты  $k_{(W-1)}$  вектора  $\mathbf{k}$  определяется числом ортогональных  $(W-1)$ -мерных кубов, которое можно построить на  $m - 2^W k_W$  точках, оставшихся не покрытыми  $W$ -кубами:

$$k_{(W-1)} = [(m - 2^W k_W) / 2^{(W-1)}]. \tag{3.34}$$

Аналогично находятся максимальные значения всех последующих координат вектора  $\mathbf{k}^*$ . Очевидно,  $i$ -я координата вектора  $\mathbf{k}^*$  с минимальной ценой  $C(\mathbf{k}^*)$  имеет вид

$$k_i = [(m - \sum_{j=i+1}^W 2^j k_j) / 2^i]. \tag{3.35}$$

Из (3.28) следует, что решение  $\mathbf{k}^{**}$  уравнения (3.27) с максимальной ценой  $C(\mathbf{k}^{**})$  должно иметь максимальную координату  $k_0$ . Она соответствует покрытию заданных  $m$  точек  $i$ -мерного логического пространства 0-мерными кубами:

$$\mathbf{k}^{**} = (m, 0, 0, \dots, 0). \tag{3.36}$$

Очевидно, что к вектору  $\mathbf{k}^{**}$  нельзя применять преобразования (3.31), в которых  $\alpha_i > 0$ , учитывая (3.32). А к вектору  $\mathbf{k}^*$  нельзя применять преобразования (3.31) с  $\alpha_i < 0$ . Назовем преобразования  $\mathbf{A}$  с  $\alpha_i \geq 0$  прямыми преобразованиями над  $\mathbf{k}$ , а с  $\alpha_i \leq 0$  – обратными. Преобразования  $\mathbf{a}_i$ ,  $i = 1, 2, \dots, W$ , соответствующие образующим группы  $G$ , назовем прямыми элементарными преобразованиями. Разрешенное неравенствами (3.32) прямое элементарное преобразование  $\mathbf{a}_i$  назовем для краткости допустимым (по отношению к вектору  $\mathbf{k}$ ), если в векторе  $\mathbf{k}$   $i$ -я координата ненулевая.

Геометрическая интерпретация прямого элементарного преобразования – это разбиение одного из покрывающих  $n$ -мерных кубов на два  $(n-1)$ -мерных. А обратное элементарное преобразование – это замена двух  $(i-1)$ -мерных покрытий одним  $i$ -мерным.

Таким образом, применяя последовательно к вектору  $\mathbf{k}^*$ , построенному по (3.33)–(3.35), прямые допустимые элементарные преобразования, получим все решения первой задачи. Другой способ получения этих решений: выбрать в качестве исходного вектора  $\mathbf{k}^{**}$  (3.36) и применять к нему последовательно допустимые элементарные преобразования –  $\mathbf{a}_i = (0, 0, \dots, -2, 1, 0, \dots, 0)$ .

Решение второй задачи. Вектор  $\mathbf{k}^*$  с минимальной ценой покрытия  $C(\mathbf{k}^*)$  не обязательно реализуется на комплексе кубов, заданном функцией  $f(x_1, x_2, \mathbf{K}, x_n)$ . Поэтому возникает задача генерирования векторов  $\mathbf{k}$  в порядке неубывания  $C(\mathbf{k})$  при исходном векторе  $\mathbf{k}^*$ , построенном по (3.33) – (3.35).

Проследим, как меняется цена покрытия  $C(\mathbf{k})$  под действием на вектор  $\mathbf{k}$  прямого допустимого преобразования.

Применим к вектору  $\mathbf{k}$  прямое элементарное допустимое преобразование:

$$\mathbf{k}' = \mathbf{k} + \mathbf{a}_i = (k_0, \dots, k_{(i-1)} + 2, k_i - 1, k_{(i+1)}, \dots, k_W).$$

Тогда согласно (3.28)

$$C(\mathbf{k}') = C(\mathbf{k}) + n - 1 + 2 = C(\mathbf{k}) + C(\mathbf{a}_i). \quad (3.37)$$

В (3.37)  $C(\mathbf{a}_i)$  – это приращение цены покрытия под действием элементарного преобразования. Как видно из (3.37), при  $i_1 < i_2$   $C(\mathbf{a}_{i_1}) > C(\mathbf{a}_{i_2})$ .

Отсюда ясно, что для построения последовательности векторов



$$\mathbf{k} + \mathbf{a}_{i_0}, \mathbf{k} + \mathbf{a}_{i_1}, \dots, \mathbf{k} + \mathbf{a}_{i_r} \quad (3.38)$$

с нарастающей ценой  $C(\mathbf{k} + \mathbf{a}_i)$  из одного исходного вектора  $\mathbf{k}$  необходимо выбрать допустимые  $\mathbf{a}_i$  с  $i_0 > i_1 > \dots > i_r$ .

Применим теперь к вектору  $\mathbf{k}$  любое преобразование  $\mathbf{A}$  из  $G$  и посмотрим, как в результате изменится цена покрытия:

$$C(\mathbf{k} + \mathbf{A}) = C(\mathbf{k}) + \sum_{i=1}^W \alpha_i C(\mathbf{a}_i). \quad (3.39)$$

Обозначим символом  $C(\mathbf{A})$  приращение цены покрытия:

$$C(\mathbf{A}) = \sum_{i=1}^W \alpha_i (n - i + 2) = (n + 2) \sum_{i=1}^W \alpha_i - \sum_{i=1}^W i \alpha_i.$$

Введем норму вектора  $\sigma$  :

$$\sigma = \|\mathbf{k}\| = \sum_{i=0}^W k_i > 0, \quad k_i \geq 0. \quad (3.40)$$

Геометрическая интерпретация нормы вектора  $\sigma$  – число покрывающих граней. Аналогично получим норму

$$\|\mathbf{A}\| = 2\mathbf{a}_1 + \dots + 2\mathbf{a}_W - \mathbf{a}_{(W-1)} - \mathbf{a}_W = \mathbf{a}_1 + \mathbf{a}_2 + \dots + \mathbf{a}_W = \sum_{i=1}^W \alpha_i > 0,$$

означающую приращение числа покрывающих граней в результате преобразования вектора  $\mathbf{k}$ , и норму

$$\|\mathbf{a}_i\| = 1,$$

означающую, что элементарное преобразование дает увеличение числа покрывающих граней на единицу.

Очевидно, что  $\|\mathbf{k}'\| = \|\mathbf{k}\| + \|\mathbf{A}\|$  и

$$C(\mathbf{A}) = (n + 2) \|\mathbf{A}\| - \sum_{i=1}^W i \alpha_i. \quad (3.41)$$

Следовательно, чем больше норма  $\|\mathbf{A}\|$  вектора преобразования, тем больше цена  $C(\mathbf{k}')$  покрытия производного варианта решения  $\mathbf{k}'$  по сравнению с  $C(\mathbf{k})$  исходного решения  $\mathbf{k}$ . Поэтому представляется естественным последовательно применять допустимые преобразования  $\mathbf{A}$  в порядке возрастания их норм с тем, чтобы упорядочить решения уравнения (3.27) в сторону увеличения значения  $C(\mathbf{k})$ . Для этого от вектора  $\mathbf{k}$  с нормой  $\sigma$

перейдем к последовательности (3.38) векторов  $\mathbf{k} + \mathbf{a}_i$  с нормой  $\sigma + 1$ , как сказано ранее. Затем, применив по такому же принципу набор допустимых прямых элементарных преобразований к каждому из векторов последовательности (3.38) в порядке их построения, получим несколько наборов векторов типа (3.38) с нормой  $\|\mathbf{k} + \mathbf{a}_i + \mathbf{a}_j\| = \sigma + 2$ . Действуя аналогично, до тех пор пока не получится вектор  $\mathbf{k}^* = (m, 0, 0, \dots, 0)$  с максимальной нормой  $\sigma = m$ , построим таким образом все допустимые векторы  $\mathbf{k}$  в порядке нарастания их нормы.

Среди векторов, сгенерированных указанным способом, могут быть одинаковые, так как

$$\mathbf{k} + \mathbf{a}_i + \mathbf{a}_j = \mathbf{k} + \mathbf{a}_j + \mathbf{a}_i, \quad i, j \in \{1, 2, \dots, W\}.$$

Для исключения повторов к вектору  $\mathbf{k} + \mathbf{a}_{i_j}$  следует применить все допустимые элементарные преобразования, кроме  $\mathbf{a}_W, \mathbf{a}_{i_1}, \dots, \mathbf{a}_{i_{(j-1)}}$ . Это условие можно выразить, используя лексикографическую упорядоченность векторов  $\mathbf{k}$ . К вектору  $\mathbf{k} + \mathbf{a}_{i_j}$  можно применить преобразование  $\mathbf{a}_i$  с  $i < i_{(j-1)}$ , где  $i_{(j-1)}$  определяется по предыдущему вектору, который лексикографически старше  $\mathbf{k} + \mathbf{a}_{i_j}$ .

Полученное множество решений первой задачи строго упорядочено по построению. В нем можно выделить классы эквивалентности по нормам векторов  $\|\mathbf{k}\|$ . Элементы каждого класса следуют друг за другом компактно, поэтому в дальнейшем классы называются массивами. Элементы каждого массива упорядочены в силу упорядоченности всего множества, а массивы упорядочены по норме.

Заметим, что порядок возрастания цен  $C(\mathbf{k})$  может нарушиться даже в пределах одного массива, так как в выражении

$$C(\mathbf{k} + \mathbf{a}_i + \mathbf{a}_j) = C(\mathbf{k}) + C(\mathbf{a}_i) + C(\mathbf{a}_j) = C(\mathbf{k}) + 2(n+2) - (i+j)$$

сумма  $(i + j)$  может принимать одинаковые значения для различных пар  $(i, j)$ . Поэтому предложенный порядок применения элементарных преобразований позволяет получить только частичное упорядочение решений первой задачи по  $C(\mathbf{k})$ .

Пусть  $\tilde{\mathbf{k}}$  – любой вектор некоторого массива, а  $\mathbf{k}$  – первый вектор

этого массива. Нетрудно убедиться, что цена  $C(\mathbf{k}')$  является нижней гранью множества цен  $C(\tilde{\mathbf{k}})$ . Покажем это.

Так как каждый первый вектор массива с нормой  $\sigma + \|\mathbf{A}\|$  строится только из первого вектора предыдущего массива с нормой  $\sigma + \|\mathbf{A}\| - 1$  посредством применения допустимого элементарного преобразования с наибольшим номером  $i_{\max}$ , он имеет вид

$$\mathbf{k}' = \mathbf{k} + \sum_{j=i_{\max}}^W \alpha_j \mathbf{a}_j.$$

Другие векторы этого массива имеют вид

$$\tilde{\mathbf{k}}' = \mathbf{k} + \sum_{j=1}^W \alpha_j \mathbf{a}_j.$$

Цены соответствующих покрытий:

$$C(\mathbf{k}') = C(\mathbf{k}) + C\left(\sum_{j=i_{\max}}^W \alpha_j \mathbf{a}_j\right) = C(\mathbf{k}) + (n+2) \|\mathbf{A}\| - \sum_{j=i_{\max}}^W j \alpha_j, \quad (3.42)$$

$$C(\tilde{\mathbf{k}}') = C(\mathbf{k}) + (n+2) \|\mathbf{A}\| - \sum_{j=1}^W j \alpha_j. \quad (3.43)$$

Внутри массива  $\sum_{j=i_{\max}}^W \alpha_j = \sum_{j=1}^W \alpha_j = \|\mathbf{A}\|$ . Тогда с учетом (3.42), (3.43) получим

$$C(\tilde{\mathbf{k}}') - C(\mathbf{k}') = \sum_{j=i_{\max}}^W j \alpha_j - \sum_{j=1}^W j \alpha_j > 0. \quad (3.44)$$

Действительно, вторую сумму в (3.44) находим из первой заменой части слагаемых таким же количеством слагаемых, но с меньшими значениями. Таким образом,

$$C(\tilde{\mathbf{k}}') > C(\mathbf{k}'). \quad (3.45)$$

Из неравенства (3.45) следует, что цена  $C(\mathbf{k}')$  – это нижняя грань множества цен  $C(\tilde{\mathbf{k}})$ .

Очевидно, что первые векторы массивов строго упорядочены по цене покрытия, поскольку первый вектор каждого массива получается из первого вектора предыдущего массива с применением элементарного преобразования, а это увеличивает цену покрытия.

Следовательно, цена покрытия, соответствующая первому вектору любого массива, является нижней гранью для цен покрытий, соответствующих всем последующим векторам. И далее, если  $C_0$  – цена некоторого покрытия и  $C_0 \leq C(\mathbf{k})$ , то среди векторов, следующих за  $\mathbf{k}$ , нет векторов, которым соответствуют покрытия с ценой, равной или меньшей  $C_0$ .

Используя результаты решений первой и второй задач, построим алгоритм преобразований исходного вектора  $\mathbf{k}$  (3.33)-(3.35) в множество векторов  $\mathbf{k}$ , частично упорядоченное по цене покрытия  $C(\mathbf{k})$ .

Алгоритм 3.4.

Шаг 1. Исходный вектор  $\mathbf{k}$  подвергается допустимым элементарным преобразованиям  $\mathbf{a}_i$  в порядке убывания  $i$ , начиная с  $i = W$ .

Шаг 2. К каждому из вновь полученных векторов  $\mathbf{k}$  в порядке их построения применяем допустимые элементарные преобразования, предварительно сравнив каждый вектор, подлежащий преобразованию, с предыдущим по координатно, начиная с конца: если первый лексикографически старше, его преобразование следует начинать с первой отличающейся координаты, иначе – с первой ненулевой, следующей за ней в порядке убывания  $i$ .

Преобразования второго шага повторяются до тех пор, пока не получим вектор  $\mathbf{k} = (m, 0, 0, \dots, 0)$ .

**Пример 3.19.** Упорядоченное множество векторов  $\mathbf{k}$  (табл. 3.4) построено по алгоритму 3.4 из вектора  $\mathbf{k}^* = (0, 2, 1)$ . Заметим, что при малых значениях  $n$  получается строгий порядок по  $C(\mathbf{k})$ .

Т а б л и ц а 3.4

*Векторы  $\mathbf{k}$ , упорядоченные по цене покрытия  $C(\mathbf{k})$*

$\sigma$	$\mathbf{k}$	$C(\mathbf{k})$
3	(0, 2, 1)	8
4	(0, 4, 0)	12
-	(2, 1, 1)	13
5	(2, 3, 0)	17
-	(4, 0, 1)	18
6	(4, 2, 0)	22
7	(6, 1, 0)	27
8	(8, 0, 0)	32

Решение третьей задачи. Здесь приведен тривиальный алгоритм решения этой задачи. В связи с его большой трудоемкостью следующий параграф посвящен повышению эффективности ее решения с применением информации о симметриях функции  $f(x_1, x_2, \mathbf{K}, x_n)$ , подлежащей приведению к МОДНФ. Пусть  $K = \bigcup_i K_i$  – комплекс кубов, заданный этой функцией.

Алгоритм 3.5.

Шаг 1. В множестве  $K_W \in K$  выбираем подмножества  $K_{Wj_W}$ ,  $j_W = 1, 2, \dots, J$ , попарно ортогональных между собой  $W$ -кубов, обладающих мощностью  $k_W$  ( $J$  – число таких подмножеств). Если таких подмножеств нет, вектор  $\mathbf{k}$  нереализуем. Если есть, положим  $K_U = K_{Wj_W}$ , выбрав одно из них.

Шаг 2. Для  $K_U$  находим в множестве  $K_{(W-1)} \in K$  подмножества мощности  $k_{(W-1)}$  ( $W-1$ )-кубов, которые попарно ортогональны между собой и с кубами из  $K_U$ . Если таких подмножеств нет, вектор  $\mathbf{k}$  относится к нереализуемым. Если есть, выбираем один из них и строим множество  $K_U = K_{Wj_W} \bigcup K_{(W-1)j_{(W-1)}}$ .

Шаг 2 повторяем для  $K_i$ ,  $i = W - 2, W - 3, \dots, 1, 0$ .

Неоднозначность построения подмножеств  $K_{ij_i}$  означает, что процесс построения МОДНФ в общем случае ветвится на шагах 1 и 2 алгоритма 3.5.

Предложенные решения выделенных подзадач исходной задачи поиска МОДНФ заданной функции  $f(x_1, x_2, \mathbf{K}, x_n)$  позволяют указать алгоритм решения последней.

Алгоритм 3.6.

Шаг 1. По заданной СДНФ булевой функции строим комплекс кубов  $K$  [49]. В множестве  $K_W$  максимальной размерности  $W$  выделяем подмножества  $K_{Wj_W}$ ,  $j_W = 1, 2, \dots, J$ , попарно ортогональных кубов таких, что позволяет уточнить реализуемое  $k_W$ .

Шаг 2. Вычисляем координаты  $k_i$ ,  $i = 0, 1, 2, \dots, W$ , по (3.33)-(3.35) вектора  $\mathbf{k}$ , выбрав значение координаты  $k_W$ , полученное на шаге 1 алгоритма 3.5.

Шаг 3. Проверяем реализуемость вектора  $\mathbf{k}$  на комплексе кубов  $K$  по

алгоритму 3.5. Если реализуется, искомая МОДНФ найдена. Иначе – вычисляем  $\sigma$  по (3.40) и идем к шагу 4.

Шаг 4. Последовательно строим векторы  $\mathbf{k}$  по алгоритму 3.4, проверяя каждый на реализуемость по алгоритму 3.5. Для первого же реализуемого вектора  $\mathbf{k}$  вычисляем норму. Если она равна  $\sigma + 1$ , искомая МОДНФ найдена. Иначе – вычисляем  $C_0 = C(\mathbf{k})$  и фиксируем (запоминаем) найденное решение  $\mathbf{k}$  и  $C_0$ .

Шаг 5. Продолжаем строить векторы  $\mathbf{k}$  по алгоритму 3.4, вычисляя для каждого  $C(\mathbf{k})$  значение  $\sigma$ , и фиксируя при каждом увеличении  $\sigma$  последнее значение  $C(\mathbf{k})$ . При этом реализуемость проверяется только для векторов  $\mathbf{k}$  с  $C(\mathbf{k}) < C_0$ . Если найден реализуемый вектор  $\mathbf{k}$  с  $C(\mathbf{k}) < C_0$ , фиксируем новое значение  $\mathbf{k}$  и  $C_0 = C(\mathbf{k})$ . Процесс построения новых векторов  $\mathbf{k}$  заканчивается при  $C(\mathbf{k}) > C_0$ . Последний зафиксированный вектор  $\mathbf{k}$  является решением задачи.

Проиллюстрируем предлагаемый метод поиска МОДНФ на примере.

**Пример 3.20.** Пусть требуется построить МОДНФ для булевой функции, заданной следующими точками: (1,1,1,1), (1,1,1,0), (1,0,1,1), (0,1,1,1), (1,1,0,0), (0,1,0,1), (0,1,1,0), (0,0,1,0). Для нее  $n = 4$ ,  $m = 8$ . Действуем в соответствии с алгоритмом 3.6.

Шаг 1. Строим комплекс кубов по заданным точкам:

0-кубы	1-кубы	2-кубы
1 1 1 0 1 0 0 0	1 1 - 1 - 0 0 0	-
1 1 0 1 1 1 1 0	1 - 1 1 1 1 1 -	1
1 1 1 1 0 1 0 1	1 1 1 - 1 1 - 1	1
1 0 1 1 0 0 1 0	- 1 1 0 0 - 1 0	-

По комплексу кубов определяем  $W = 2$ ,  $k_W = 1$ .

Шаг 2. Вычисляем по (3.35)

$$\mathbf{k} = (0, 2, 1).$$

Шаг 3. Действуя в соответствии с алгоритмом 3.5, убеждаемся, что  $\mathbf{k}$  не реализуется на построенном комплексе кубов. Вычисляем по (3.40)  $\sigma^* = 3$ .

Шаг 4. Строим очередной вектор  $\mathbf{k}$  по алгоритму 3.4 (см. табл. 3.4):

$$\mathbf{k} = (0, 4, 0).$$

Проверка реализуемости этого вектора по алгоритму 3.5 показывает, что он реализуется следующим набором  $i$ -кубов:  $(0, 1, -, 1)$ ,  $(0, -, 1, 0)$ ,  $(1, 1, -, 0)$ ,  $(1, -, 1, 1)$ . Вычисляем по (3.40)  $\sigma = 4 = \sigma^* + 1$ . Следовательно, найденный набор  $i$ -кубов соответствует искомой МОДНФ. Строим ее по этому набору:

$$f(x_1, x_2, x_3, x_4) = \bar{x}_1 x_2 x_4 \vee \bar{x}_1 x_3 \bar{x}_4 \vee x_1 x_2 \bar{x}_4 \vee x_1 x_3 x_4.$$

### 3.11. Применение информации о симметриях булевой функции в процессе приведении ее к МОДНФ

Рассмотрим структуру множества  $i$ -кубов комплекса  $K = \bigcup_{i=1}^W K_i$ , заданную на нем симметриями булевой функции  $f(x_1, x_2, \mathbf{K}, x_n)$ .

В множестве  $K$   $i$ -кубов можно выделить классы эквивалентности по  $h \in H_n$ , т.е. подмножества, которые отображаются на себя под действием симметричных преобразований. Заметим, что любой класс эквивалентности по симметриям является подклассом класса  $K_i$ , так как размерность  $i$ -куба не меняется под действием симметричных преобразований  $h \in H_n$ .

Из утверждения 3.3 следует, что ортогональность функции не нарушается под действием симметричного преобразования. Поэтому множество попарно ортогональных кубов  $K_{ij_i}$  остается таким и после его симметричного преобразования.

Эквивалентность по  $h \in H_n$  отдельных  $i$ -кубов, принадлежащих различным классам  $K_{ij_i}$ , индуцирует эквивалентность и этих классов. Иными словами, множество  $K_{ij_i}$  может быть переведено симметричными преобразованиями в другое множество или в себя. И наконец, эквивалентность по симметриям целых классов  $K_{ij_i}$  индуцируется эквивалентность по  $h \in H_n$  искомым ортогональным функциям.

Симметричные преобразования не меняют числа букв в функции, а значит, и ее цены. Таким образом, действием этих преобразований множество МОДНФ разбивается на классы эквивалентности по  $h \in H_n$ . Если построена МОДНФ, которая является представителем класса эквивалентности по симметриям, весь класс восстанавливается с помощью нетрудоемких симметричных преобразований этого представителя. Тем самым обуславливается возможность сокращения перебора в процессе проверки реализуемости вектора  $\mathbf{k}$  на комплексе кубов по алгоритму 3.5, рассмотренному в

предыдущем параграфе. Далее приведен метод сокращения, основанный на идее канонического упорядочения комбинаторных объектов.

Представим  $i$ -куб трюичным [36] вектором-столбцом

$$\mathbf{x}_q = \begin{pmatrix} \delta_{1q} \\ x_{1q} \\ \delta_{2q} \\ x_{2q} \\ \mathbf{L} \\ \delta_{lq} \\ x_{lq} \\ \mathbf{L} \\ \delta_{nq} \\ x_{nq} \end{pmatrix},$$

где  $x_{lq}^{\delta_{lq}} = 1$ , если  $\delta_{lq} = 1$ ;  $x_{lq}^{\delta_{lq}} = 0$ , если  $\delta_{lq} = 0$ ;  $x_{lq}$  обозначается знаком "–", если  $\delta_{lq}$  безразлично. Список таких столбцов образует прямоугольную таблицу, представляющую комплекс кубов  $K = \{ \mathbf{x}_q \}$ . Каждый ее столбец-вектор  $\mathbf{x}_q$ , поставлен в соответствие одному из  $i$ -кубов, а каждая строка – одной из переменных заданной СДНФ. Поэтому для обозначения номеров строк и столбцов таблицы будем использовать те же обозначения, что и для номеров переменных ( $l$ ) и  $i$ -кубов ( $q$ ).

Зададим на множестве векторов  $\mathbf{x}_q$  лексикографический порядок в алфавите  $(0, 1, -)$ . Задание лексикографического порядка на множестве  $K = \{ \mathbf{x}_q \}$  индуцирует порядок на каждом из подмножеств  $K_{ij_i}$ , а также на множестве этих подмножеств. В этом случае среди подмножеств  $K_{ij_i}$  можно выделить старшее (младшее) подмножество. Так как одно из таких подмножеств может быть приведено к другому с помощью симметричного преобразования, очевидно следующее

**У т в е р ж д е н и е 3. 10.** *Для того чтобы подмножество  $K_{ij_i} \in K_i$ ,  $i \in \{0, 1, 2, \dots, W\}$ ,  $i$ -кубов было старшим (младшим), необходимо и достаточно, чтобы оно не могло стать старше (младше) под воздействием хотя бы одного из симметричных преобразований.*

Утверждение указывает признак, по которому можно распознать старший (младший) класс  $K_{ij_i} \in K_i$  без сравнения его с другими эквивалентными ему по симметриям классами  $K_{ij_i'} \in K_i$ ,  $j_i' \neq j_i$ , т.е. рассма-



тривая его изолированно. Эта задача сводится к поиску симметричного преобразования  $h \in H_n$ , под действием которого множество  $K_{ij_i}$  могло стать старше (младше).

Практически задача сводится к проверке каноничности векторов-столбцов  $x_q$  с повторяющимися координатами при условии, что на множестве его координат действует группа автоморфизмов  $H_n$ . Возможность распознать и использовать единственный из множества эквивалентных по симметриям  $h \in H_n$  классов  $K_{ij_i}$  в точках ветвления процесса, описанного алгоритмом 3.5, позволяет сократить перебор без риска потерять искомые МОДНФ.

В силу изложенного в алгоритм 3.5 можно внести следующее изменение: на шаге 2 вместо всех подмножеств  $K_{ij_i}$  ограничимся рассмотрением только представителей своих классов эквивалентности по  $h \in H_n$ , другими словами – только не симметричных между собой подмножеств  $K_{ij_i}$ .

Рассмотрим предложенные изменения алгоритма 3.5 на примере.

**Пример 3.20.** Пусть требуется проверить реализуемость вектора  $k^* = (1, 1, 2)$  на комплексе кубов

$$K_0 = \begin{Bmatrix} 00001001011 \\ 00010010101 \\ 00100110111 \\ 01000101111 \end{Bmatrix}, K_1 = \begin{Bmatrix} 000-0-00010-01-1 \\ 00-0000-10-0101- \\ 0-00-011-0111-11 \\ -00011-00-11-111 \end{Bmatrix}, K_2 = \begin{Bmatrix} 0-0-0- \\ 00-0-- \\ -0--11 \\ --01-1 \end{Bmatrix}$$

заданной функции

$$f = (x_1, x_2, x_3, x_4) = \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2\bar{x}_3x_4 \vee \bar{x}_1\bar{x}_2x_3\bar{x}_4 \vee \bar{x}_1\bar{x}_2x_3x_4 \vee x_1\bar{x}_2\bar{x}_3\bar{x}_4 \vee x_1\bar{x}_2\bar{x}_3x_4 \vee x_1\bar{x}_2x_3\bar{x}_4 \vee x_1\bar{x}_2x_3x_4 \vee x_1x_2\bar{x}_3\bar{x}_4 \vee x_1x_2\bar{x}_3x_4 \vee x_1x_2x_3\bar{x}_4 \vee x_1x_2x_3x_4.$$

Группа симметрий этой функции  $H_n = \{e, (1, 2)(3, 4)\}$ .

Подмножества  $K_{2j_2}$  в множестве  $K_2$  комплекса кубов:

$$K_{21} = \begin{Bmatrix} -0 \\ 0- \\ 01 \\ -- \end{Bmatrix}; K_{22} = \begin{Bmatrix} -- \\ 0- \\ 01 \\ -1 \end{Bmatrix}; K_{23} = \begin{Bmatrix} 0- \\ -0 \\ -- \\ 01 \end{Bmatrix}; K_{24} = \begin{Bmatrix} 0- \\ -- \\ -1 \\ 01 \end{Bmatrix}.$$

Среди них подмножества  $K_{23}, K_{24}$  являются старшими (каноническими). Поэтому в дальнейших построениях МОДНФ будем рассматривать только эти варианты. Положим  $K'_U = K_{23}$ ,  $K''_U = K_{24}$  и переходим к поиску подмножеств  $K'_{1j_1}, K''_{1j_1}$ , ортогональных подмножествам  $K'_U, K''_U$ . Это соответственно

$$K'_{11} = \begin{Bmatrix} - \\ 1 \\ 1 \\ 1 \end{Bmatrix}, \quad K''_{11} = \begin{Bmatrix} 1 \\ 0 \\ 0 \\ - \end{Bmatrix}.$$

Следовательно, на этом этапе

$$K'_U = \begin{Bmatrix} 0 & - & - \\ - & 0 & 1 \\ - & - & 1 \\ 0 & 1 & 1 \end{Bmatrix}, \quad K''_U = \begin{Bmatrix} 0 & - & 1 \\ - & - & 0 \\ - & 1 & 0 \\ 0 & 1 & - \end{Bmatrix}.$$

Подмножества  $K'_{0j_0}, K''_{0j_0}$ , ортогональные  $K'_U, K''_U$ , имеют вид

$$K'_{0j_0} = \begin{Bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{Bmatrix}; \quad K''_{0j_0} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{Bmatrix}.$$

Таким образом, окончательно запишем

$$K'_U = \begin{Bmatrix} 0 & - & - & 1 \\ - & 0 & 1 & 0 \\ - & - & 1 & 0 \\ 0 & 1 & 1 & 0 \end{Bmatrix}, \quad K''_U = \begin{Bmatrix} 0 & - & 1 & 0 \\ - & - & 0 & 0 \\ - & 1 & 0 & 0 \\ 0 & 1 & - & 1 \end{Bmatrix}.$$

По найденным  $K'_U, K''_U$  строим две МОДНФ, не эквивалентные между собой по симметриям заданной функции  $f = (x_1, x_2, x_3, x_4)$ :

$$f_1 = \bar{x}_1 \bar{x}_4 \vee \bar{x}_2 x_4 \vee x_2 x_3 x_4 \vee x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4;$$

$$f_2 = \bar{x}_1 \bar{x}_4 \vee x_3 x_4 \vee x_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 x_4.$$

Используя их, восстанавливаем эквивалентные по  $h = (1, 2)(3, 4) \in H_n$  МОДНФ:

$$f_3 = \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 x_3 \vee x_1 x_3 x_4 \vee \bar{x}_1 x_2 \bar{x}_3 \bar{x}_4;$$

$$f_4 = \bar{x}_2 \bar{x}_3 \vee x_3 x_4 \vee x_1 \bar{x}_2 \bar{x}_3 \vee \bar{x}_1 \bar{x}_2 x_3 \bar{x}_4.$$

## СИММЕТРИИ И СИНТЕЗ СХЕМ

**4.1. Синтез линейных избирательных устройств и проблема изоморфизма**

К линейным избирательным устройствам относятся формирующие и задерживающие цепи, фильтры и фазовращатели, амплитудные и фазовые корректоры, согласователи со специальными амплитудно-частотными и фазо-частотными характеристиками (АЧХ и ФЧХ), моделирующие, имитирующие и другие четырехполюсники. Как составные части разнообразных систем связи, промышленной электроники и автоматического управления, радиотехнических и других систем линейные избирательные устройства в значительной степени определяют параметры этих систем.

Любое избирательное устройство характеризуется ограниченным набором типов элементов, из которых оно состоит. Состав элементов является одним из критериев классификации линейных избирательных устройств: *RC*-цепи, *RLC*-цепи, *LC*-цепи, *RLCM*-цепи или цепи с индуктивными связями, активные *RC*-цепи. Последние содержат, кроме конденсаторов с емкостью *C* и резисторов с сопротивлением *R*, активные элементы. Исторически первым активным элементом была электронная лампа, затем лампу сменил транзистор. В настоящее время в качестве активного элемента наиболее широко применяется операционный усилитель, выполненный в виде интегральной микросхемы или ее части.

Функции линейных избирательных цепей постепенно усложняются, и вместе с тем растет их сложность при том же ограниченном наборе типов элементов, входящих в их состав. Следовательно, одновременно возрастает степень однородности этих устройств. Как следует из сказанного в гл. 1, однородность служит предпосылкой для разбиения множества устройств одного назначения, характеризующихся одним и тем же составом элементов, на классы эквивалентности-изоморфизма. Изоморфными являются устройства, схемы которых, отображая их структуры, совпадают с точностью до нумерации их узлов и элементов [50; 51], т.е. могут быть приведены друг к другу перенумерацией узлов и элементов. Классы эквивалентности в общем случае тем обширнее, чем однороднее состав элементов устройств и богаче группы их симметрий. Практический интерес представляют лишь оригинальные, значит неизоморфные, схемы. Поэтому при решении задачи оптимизации устройства путем выбора его структуры проектировщика должны интересовать лишь представители классов эквивалентности-изоморфизма.

Проблема отсеечения изоморфных вариантов не стояла перед проектировщиками простейших избирательных цепей, тем более что их структурой обычно задавались, и оптимизация устройства велась в основном за счет выбора их параметров (оптимизация в пространстве параметров).

Попытки оптимизации устройства с помощью выбора его структуры, т.е. в пространстве структур, перспективность которой была понята уже давно, стимулировали развитие методов структурного синтеза электрических цепей. Первые методы структурного синтеза ([52; 53] и др.) приводили лишь к структурам определенных видов. При таких сильных ограничениях на структуры синтезируемых устройств проблема изоморфизма не привлекала к себе особого внимания. Но это нельзя отнести к достоинствам методов, так как структурные ограничения одновременно существенно ограничивают возможность оптимизации.

Позже появились методы структурного синтеза, объединяющие оптимизацию в пространстве структур и в пространстве параметров. К ним относится, например, метод переменных состояний ([54] и др.). Указанные методы успешно применяются лишь при несложных схемах устройств, поскольку они приводятся к задачам нелинейного программирования, для решения которых в общем случае нет эффективных методов. При таком подходе решение проблемы отсеечения изоморфных вариантов также не относится к первоочередному.

Чтобы наиболее эффективно использовать структуру устройства для его оптимизации по любым качественным показателям, в том числе и по числу задействованных в нем элементов, необходимо производить выбор из полной совокупности конкурентоспособных схем заданного функционирования. На современном этапе развития теории электрических цепей эта задача сводится к трем последовательно решаемым подзадачам.

Первая – формальное представление заданного функционирования в виде схемных функций, реализуемых в выбранном классе линейных избирательных цепей (задача аппроксимации ([55; 56] и др.)).

Вторая – построение множества схемных реализаций ([51] и др.) таких схемных функций (задача реализации или непосредственно структурного синтеза).

Третья – параметрическая оптимизация каждой из полученных схем (или по крайней мере наиболее перспективных из них) по заданной совокупности качественных показателей и отбор схем с наилучшими показателями (задача параметрической оптимизации ([57] и др.)).

Такая декомпозиция исходной задачи поиска оптимальной схемы заданного функционирования обусловлена тем, что каждая из подзадач в об-

щем случае чрезвычайно трудоемка, и оптимальное решение каждой из них требует своих математических моделей и средств (методов, алгоритмов). Решение второй задачи без ограничения на искомую структуру осуществляется комбинаторными методами, тем более что в общем случае она решается неоднозначно. При таком подходе проблема отсечения изоморфных вариантов схем стоит очень остро, так как во многих случаях количество изоморфных вариантов существенно превышает количество неизоморфных схем.

Рассмотрим проблему изоморфизма и ее решение, выбрав наиболее распространенный класс линейных избирательных схем – активные  $RC$ -схемы с операционными усилителями (ОУ). В качестве схемной функции, подлежащей реализации в выбранном классе схем, возьмем коэффициент передачи напряжения, имеющий важное практическое значение.

#### 4.2. Связь между структурой схемы и ее схемной функцией

Результатом решения задачи аппроксимации является дробно-рациональная функция комплексной частоты  $p$

$$T(p) = \frac{U_{вых}(p)}{U_{вх}(p)} = \frac{\sum_{i=1}^m a_i p^i}{\sum_{j=1}^n b_j p^j}, \quad (4.1)$$

физически реализуемая в выбранном классе схем. В (4.1)  $U_{вх}(p)$  – операторное отображение напряжения на входе искомой схемы, а  $U_{вых}(p)$  – на ее выходе. Из условий физической реализуемости  $T(p)$  [50; 55; 58] в классе активных  $RC$ -схем следует, что  $a_i, b_j$  – вещественные коэффициенты.

Функция (4.1) является исходной на этапе решения задачи структурного синтеза. Она отображает требования к АХЧ и ФХЧ проектируемого устройства с заданной степенью точности. Если задача аппроксимации решена оптимально, степень  $n$  знаменателя (4.1) имеет минимальное значение, при котором возможно достижение указанной точности аппроксимации.

Из известных математических моделей схемы при решении задач структурного синтеза наиболее удачно подходит расширенная [59] или плавающая [60] матрица проводимостей  $Y$ , состоящая из коэффициентов системы уравнений узловых напряжений схемы. В этой матрице наглядно и просто отражаются сведения о топологии схемы и об изменениях топологии, которые связаны с перебором схемных конфигураций. Переда-

точная функция  $T(p)$  схемы может быть представлена через многократные суммарные алгебраические дополнения (МСАД) [59; 61] матрицы  $\mathbf{Y}$ .

Пусть  $n_v, n_w, n_z$  – номера соответственно входного, выходного и общего (заземленного) узлов проектируемого устройства с  $n_z$  узлами;  $q_1(k), q_2(k)$  – номера узлов, к которым подключены соответственно инвертирующий и неинвертирующий входные полюсы  $k$ -го ОУ, а  $r_1(k), r_2(k)$  – номера узлов, к которым подключены соответственно инвертирующий и неинвертирующий выходные полюсы ОУ,  $k = 1, 2, \dots, n_A$ . Будем считать ОУ идеальным, т.е. обладающим бесконечно большим коэффициентом усиления напряжения, бесконечно большим входным и нулевым выходным сопротивлениями. Тогда коэффициент передачи напряжения, выраженный через МСАД расширенной матрицы проводимостей схемы, имеет вид

$$T(p) = \frac{\Delta_{n_v, [r_1(1) + r_2(1)], [r_1(2) + r_2(2)], \mathbf{K}, [r_1(n_A) + r_2(n_A)], n_z}}{\Delta_{n_w, [q_1(1) + q_2(1)], [q_1(2) + q_2(2)], \mathbf{K}, [q_1(n_A) + q_2(n_A)], n_z}} = \frac{\Delta_A}{\Delta_B}. \quad (4.2)$$

Рассмотрим матрицу  $\tilde{\mathbf{Y}}$  из элементов матрицы  $\mathbf{Y}$ , вошедших в алгебраические дополнения числителя и знаменателя (4.2). Как следует из (4.2), матрица  $\tilde{\mathbf{Y}}$  получается из матрицы  $\mathbf{Y}$  вычеркиванием из последней  $n_A + 2$  строк и  $n_A + 1$  столбцов и добавлением их элементов к оставшимся после вычеркивания элементам матрицы  $\mathbf{Y}$ . Таким образом, матрица  $\tilde{\mathbf{Y}}$  содержит  $\rho = n_z - n_A - 2$  строк и  $(\rho + 1)$  столбцов. В ней можно выделить блоки  $\tilde{\mathbf{Y}}_A$  и  $\tilde{\mathbf{Y}}_B$ , формирующие соответственно числитель и знаменатель (4.2).

Матрицу  $\mathbf{Y}$  представим в виде суммы двух однотипных матриц:

$$\mathbf{Y} = \mathbf{G} + p\mathbf{C},$$

где  $\mathbf{C}$  – матрица емкостей схемы,  $\mathbf{G}$  – матрица ее проводимостей. Тогда определители числителя и знаменателя (4.2) можно разложить по степеням  $p$  с помощью соотношения [59; 61]

$$s_i = (-1)^{(\sigma + \varepsilon)} \sum_i M_i^C M_{n_z - n_A - 2 - i}^G, \quad (4.3)$$

где  $s^i$  – полиномиальный коэффициент при  $p^i$ ;  $M_i^C$  – минор  $i$ -го порядка из матрицы емкостей  $\mathbf{C}_A$  (или  $\mathbf{C}_B$ );  $M_{n_z - n_A - 2 - i}^G$  – дополняющий его минор из

матрицы резистивных проводимостей  $\mathbf{G}_A$  ( $\mathbf{G}_B$ );  $\sigma$  – сумма номеров строк и столбцов, вычеркиваемых при образовании миноров;  $\varepsilon$  – число инверсий в их последовательностях.

В результате разложения полиномиальные коэффициенты (4.2) станут как функции параметров резистивных и емкостных элементов схемы. Приравняв каждую соответствующему численному полиномиальному коэффициенту из (4.1), получим систему уравнений для определения численных значений параметров двухполюсных элементов схемы на этапе параметрического синтеза. Чтобы система уравнений не оказалась переопределенной, число двухполюсных элементов должно быть не меньше величины  $m+n-n_0+2$ , где  $n_0$  – количество нулевых коэффициентов в последовательностях  $a_i, b_j$ . В [62] показано, что это число может быть уменьшено еще на единицу. Таким образом, по заданной схемной функции (4.1) определяется минимальное число двухполюсных элементов искомой схемы.

Сопоставив полиномиальные коэффициенты при одинаковых степенях  $p$  из (4.1) и (4.3), нетрудно определить состав этих двухполюсников. Для реализации ненулевого коэффициента  $b_n$  необходимо, чтобы схема имела не менее  $n_z = n + n_A + 2$  узлов и не менее  $n_C = n$  емкостей, для реализации  $b_0$  – не менее  $n_G = n$  резисторов.

Однако не всегда удается ограничиться минимальным числом узлов и элементов схемы. Как правило, к искомой схеме предъявляется ряд дополнительных требований (удобство настройки, минимальный разброс номинальных значений параметров, ограничения по чувствительности схемной функции к изменению параметров элементов, заданный динамический диапазон и др.). В связи с этим целесообразно рассматривать указанные значения  $n_z$ ,  $n_G$ ,  $n_C$  как предельно малые, так как при уменьшении хотя бы одного из них не удастся реализовать заданную схемную функцию. Увеличение значений  $n_z$ ,  $n_G$ ,  $n_C$  оправдано, когда оно применяется для достижения требуемых качественных показателей.

Каждое слагаемое полиномиального коэффициента  $s^i$  при  $p^i$  в схемной функции (4.2) после разложения ее алгебраических дополнений по степеням  $p$  – это произведение параметров ее пассивных двухполюсников. Условимся говорить, что подсхема из этих двухполюсников "реализует" слагаемое полиномиального коэффициента, и назовем ее элементарной. Из (4.3) видно, что она должна содержать не менее  $i$  элементов емкостного характера и  $(n_z - n_A - 2 - i)$  элементов – резистивного. Она не должна содержать контуры, ее граф является  $k$ -деревом [63].

Схема, полученная объединением одноименных узлов двух подсхем с различными матрицами  $\tilde{Y}'$  и  $\tilde{Y}''$ , описывается матрицей  $\tilde{Y}' + \tilde{Y}''$ . Если каждая из схем является элементарной, т.е. реализует одно из слагаемых полиномиального коэффициента схемной функции, то объединенная схема реализует одновременно оба слагаемых. Разумеется, если после объединения элементарных подсхем некоторые однотипные двухполюсники оказались включенными параллельно, такую схему можно упростить. Из сказанного ясно, что задачу структурного синтеза схем с ОУ можно свести к реализации слагаемых полиномиальных коэффициентов схемной функции. В частности, для выполнения требования  $a_i = 0$  или  $b_j = 0$  необходимо, чтобы в схеме отсутствовала соответствующая элементарная подсхема (структурный нуль) или в ней были две соответствующие подсхемы, обеспечивающие слагаемые  $a_i \neq 0$  или  $b_j \neq 0$ , но с противоположными знаками (параметрический нуль).

Заметим, что числом, местом и способом включения ОУ в схему определяются лишь номера строк и столбцов матрицы  $Y$ , участвующих в формировании полиномиальных коэффициентов.

### 4.3. Структура множества схемных реализаций заданной функции

Задача структурного синтеза схем, реализующих некоторую схемную функцию, решается неоднозначно. Пусть  $\lambda_i$  – схемная реализация (4.1), характеризующаяся  $n_z$  узлами,  $n_A$  ОУ,  $n_G$  проводимостями резистивными и  $n_C$  – емкостными, а  $\Lambda = \{\lambda_i\}$  – множество таких схем. Это множество при конечных значениях  $n_z, n_A, n_G, n_C$  конечно, хотя и может быть достаточно обширным.

Рассмотрим структуру этого множества.

Известно [64], что значение определителя не меняется в случае прибавления к элементам (вычитания из элементов) любой его строки (столбца) соответствующих элементов другой строки (столбца). Эта величина инвариантна также (с точностью до знака) к перестановкам строк и (или) столбцов определителя. Если указанные изменения выполнены над одноименными строками и столбцами МСАД числителя и знаменателя (4.2), им соответствуют определенные изменения в схеме. Схемотехнический смысл перестановки двух строк матрицы  $\tilde{Y}$ , из которой строятся МСАД (4.2), – это переключение входных, а столбцов – выходных полюсов ОУ без изме-



нения пассивной подсхемы. Перестановка слагаемых в суммарных строках (столбцах) матрицы  $\tilde{\mathbf{Y}}$  соответствует переключению между собой входных (выходных) полюсов одного ОУ без изменения пассивной части схемы. Различным способом перекодирования [66] суммируемых индексов в (4.2) соответствуют перемещения входных (выходных) полюсов ОУ из одних узлов схемы в другие.

Заметим, что знак функции (4.1) не меняется, если перечисленные изменения выполнены над одноименными строками и столбцами МСАД числителя и знаменателя (4.2). Остановимся именно на таких преобразованиях схемы. Это значит, что исключается из рассмотрения перемещение выходного полюса ОУ, соединенного с выходом схемы. Указанное исключение означает, что выбранные преобразования относятся только к внутренней части схемы, а ее внешние узлы остаются неизменными. Ограничимся рассмотрением структуры множества схем с ОУ, не имеющими непосредственных связей между собой. И выберем варианты преобразований, которые не приводят к таким связям.

Множество схем, полученных из схемы  $\lambda_{i0} \in \Lambda$  с использованием указанных преобразований, описывается общей функцией (4.2) и, следовательно, они являются эквивалентными по этой функции. Отмеченное отношение эквивалентности разбивает множество  $\Lambda$  схемных реализаций (4.2) на классы эквивалентности. Обозначим символом  $\Lambda_i$  множество схем, эквивалентных схеме  $\lambda_{i0}$ .

Теперь рассмотрим одновременную перестановку строк с номерами  $q_1(l)$  и  $q_1(k)$ ,  $q_2(l)$  и  $q_2(k)$ , а также столбцов с номерами  $r_1(l)$  и  $r_1(k)$ ,  $r_2(l)$  и  $r_2(k)$ . Такой перестановке строк и столбцов соответствует переключение двух ОУ –  $l$ -го и  $k$ -го. Схема инвариантна к указанному переключению ОУ: новая приводится к исходной только перенумерацией элементов схемы. Следовательно, такие схемы являются изоморфными. Значит, внутри каждого класса эквивалентности изоморфные схемы образуют подкласс.

Приведем оценку мощности классов эквивалентности, основанную на подходе, изложенном в работе [65]. При выполнении оценки использовано отображение рассмотренных преобразований схемы с помощью действий над индексами МСАД [66; 67] – перестановок индексов и переиндексации. В табл. 4.1 отражены эти действия над индексами вместе с соответствующими действиями над строками (столбцами) матрицы  $\mathbf{Y}$  и соответствующие преобразования схемы.

Пусть  $\mathbf{M}_k = (q_1(k), q_2(k), r_1(k), r_2(k))$  – вектор, отображающий расположение  $k$ -го ОУ в схеме. Множеством  $M_k = \{\mathbf{M}_k\}$ ,  $k \in \{1, 2, \dots, n_A\}$ ,

Соответствие между преобразованиями схемы, ее матрицы проводимостей и МСАД

Преобразование схемы	Преобразования матрицы $\mathbf{Y}$ при переходе к матрице $\tilde{\mathbf{Y}}$	Действия над индексами МСАД функции (4.2)
Переключение одноименных входных полюсов $k$ -го и $l$ -го ОУ	Перестановка строк $q_1(k)$ с $q_1(l)$ и $q_2(k)$ с $q_2(l)$	$q_1(k) \rightarrow q_1(l)$ $q_2(k) \rightarrow q_2(l)$ $q_1(l) \rightarrow q_1(k)$ $q_2(l) \rightarrow q_2(k)$
Одновременное переключение одноименных выходных полюсов $k$ -го и $l$ -го ОУ	Перестановка столбцов $r_1(k)$ с $r_1(l)$ и $r_2(k)$ с $r_2(l)$	$r_1(k) \rightarrow r_1(l)$ $r_2(k) \rightarrow r_2(l)$ $r_1(l) \rightarrow r_1(k)$ $r_2(l) \rightarrow r_2(k)$
Переключение входных инвертирующего и неинвертирующего полюсов $k$ -го ОУ	Замена номера вычеркиваемого столбца [номер $q_1(k)$ на номер $q_2(k)$ ] и столбца, элементы которого получают суммированием соответствующих элементов обоих столбцов [номер $q_2(k)$ на номер $q_1(k)$ ]	$q_1(k) + q_2(k) \rightarrow$ $\rightarrow q_2(k) + q_1(k)$
Переключение выходных инвертирующего и неинвертирующего полюсов $k$ -го ОУ	Замена номеров вычеркиваемой строки [номер $r_1(k)$ на номер $r_2(k)$ ] и строки, элементы которой получают суммированием соответствующих элементов обеих строк [номер $r_2(k)$ на номер $r_1(k)$ ]	$r_1(k) + r_2(k) \rightarrow$ $\rightarrow r_2(k) + r_1(k)$

описывается расположение всех  $n_A$  ОУ в схеме. Эквивалентное преобразование схемы  $\lambda_k$  в схему  $\lambda_l$  представим с помощью отображения  $f: M_k \rightarrow M_l = \{\mathbf{M}_l\}$ ,  $l \in \{1, 2, \dots, n_A\}$ . Отображение  $f$  представим как композицию нескольких видов отображений:

$$f = f_1 \circ f_2 \circ f_3 \circ f_4,$$

где  $f_1(f_2)$  – отображение, соответствующее переключению выходных (выходных) полюсов ОУ,  $f_3(f_4)$  – входных (выходных) полюсов одного ОУ.

Отображение  $f_1$  условимся применять только к ОУ, не имеющим непосредственных связей с выходом схемы. В реальных устройствах выход одного из ОУ совпадает с выходом устройства, что способствует развязке его с последующими цепями. Таким образом, будем применять его к  $(n_A - 1)$  ОУ. Образ  $\mathbf{M}_l$  при отображении  $f_1$  может быть представлен с помощью подстановки

$$g_r = \begin{pmatrix} 1 & 2 & \mathbf{L} & (n_A - 1) \\ k_{1r} & k_{2r} & \mathbf{L} & k_{(n_A-1)r} \end{pmatrix} \quad (4.4)$$

как

$$\mathbf{M}_l = (q_1(k), q_2(k), r_1(k_r), r_2(k_r)), \quad (4.5)$$

$$k, k_r, l \in \{1, 2, \dots, n_A - 1\},$$

где  $k_r$  – результат действия ограничения [8] подстановки  $g_r$  на элемент  $k$ ,  $k_r = k g_r$ .

Подстановки  $g_r$  образуют симметрическую группу  $S_r$ . Число эквивалентных преобразований, описанных с помощью  $f_1$ , равно порядку  $S_r$ :

$$N_1 = |S_r| = (n_A - 1)!$$

Образ  $\mathbf{M}_l$  при отображении  $f_2$  представим с помощью подстановки

$$g_q = \begin{pmatrix} 1 & 2 & \mathbf{L} & n_A \\ k_{1q} & k_{2q} & \mathbf{L} & k_{n_Aq} \end{pmatrix}, \quad (4.6)$$

как и в предыдущем случае:

$$\mathbf{M}_l = (q_1(k_q), q_2(k_q), r_1(k), r_2(k)), \quad (4.7)$$

$$k, k_q, l \in \{1, 2, \dots, n_A\},$$

где  $k_q = k g_q$ . Число эквивалентных преобразований этого вида определяется порядком симметрической группы  $S_q$  подстановок  $g_q$ :

$$N_2 = |S_q| = n_A!$$

Отображение  $f_3$  можно формализовать с помощью подстановки  $g_\gamma$  с системой образующих

$$\gamma_k = \begin{pmatrix} q_1(k) & q_2(k) \\ q_2(k) & q_1(k) \end{pmatrix}.$$

Образ  $\mathbf{M}_l$  при отображении  $f_3$  представим с помощью образующей  $\gamma_k$ :

$$\mathbf{M}_l = \begin{cases} ((q_1(k), q_2(k), r_1(k), r_2(k))) & \text{при четных степенях } \gamma_k; \\ ((q_2(k), q_1(k), r_1(k), r_2(k))) & \text{при нечетных степенях } \gamma_k; \end{cases}$$

$$k, l \in \{1, 2, \dots, n_A\}.$$

Каждая образующая  $\gamma_k$  является циклической подстановкой с длиной цикла равной двум. Поэтому число  $N_3$  эквивалентных преобразований, полученных с помощью отображений  $f_3$ , которое определяется порядком

группы  $S_\gamma$  подстановок  $g_\gamma = \prod_{k=1}^{n_A} \gamma_k^{\sigma_k}$ ,  $\sigma_k \in \{0, 1\}$ , равно

$$N_3 = |S_\gamma| = 2^{n_A}.$$

Аналогично формализуется отображение  $f_4$  с использованием подстановки  $g_\nu$  с образующими

$$\nu_k = \begin{pmatrix} r_1(k) & r_2(k) \\ r_2(k) & r_1(k) \end{pmatrix}.$$

В этом случае

$$\mathbf{M}_l = \begin{cases} ((q_1(k), q_2(k), r_1(k), r_2(k))) & \text{при четных степенях } \nu_k; \\ ((q_1(k), q_2(k), r_2(k), r_1(k))) & \text{при нечетных степенях } \nu_k; \end{cases}$$

$$k, l \in \{1, 2, \dots, n_A\}.$$

Количество эквивалентных схем, полученных с применением отображения

$f_4$ , равно порядку группы  $S_\nu$  подстановок  $g_\nu = \prod_{k=1}^{n_A} \nu_k^{\sigma_k}$ ,  $\sigma_k \in \{0, 1\}$ :

$$N_4 = |S_\nu| = 2^{n_A}.$$

Теперь определим мощность  $N_{\lambda_i}$  всего множества  $\Lambda_i$  эквивалентных схем, порожденных одной схемой  $\lambda_{i0}$ :

$$N_{\lambda_i} = N_1 N_2 N_3 N_4 = 2^{2n_A} n_A! (n_A - 1)!$$

Оценки мощностей  $N_{\lambda_i}$  классов эквивалентности для широко распространенных в практике схем с двумя-четырьмя ОУ приведены в табл. 4.2.

Мощности классов эквивалентности схем с ОУ

Количество ОУ	Включение полюсов ОУ	Преобразование	Мощность $N_{\lambda_i}$	Мощность $N_i$
2	Трехполюсное	$f = f_1 \bullet f_2$	2	2
2	Дифференциальное входных полюсов и недифференциальное выходных	$f = f_1 \bullet f_2 \bullet f_3$	8	8
3	Трехполюсное	$f = f_1 \bullet f_2$	12	6
3	Дифференциальное входных полюсов и недифференциальное выходных	$f = f_1 \bullet f_2 \bullet f_3$	96	48
4	Трехполюсное	$f = f_1 \bullet f_2$	144	24
4	Дифференциальное входных полюсов и недифференциальное выходных	$f = f_1 \bullet f_2 \bullet f_3$	2304	384

Из соотношений (4.4) – (4.7) видно, что подстановкам  $g_q \in S_q$ ,  $g_r \in S_r$  соответствуют подстановки  $g'_q$ ,  $g'_r$  множества номеров узлов схемы, объединенных с полюсами ОУ. Они образуют изоморфные группы  $S'_q$ ,  $S'_r$  соответственно.

Теперь перейдем к оценке мощности  $N_i$  множества неизоморфных схем. Как уже отмечалось, подсхема, состоящая из одних ОУ, инвариантна к перестановке последних. Следовательно, при  $k_{ir} = k_{iq} = k_i$ ,  $i = 1, 2, \mathbf{L}$ ,  $\mathbf{L}, (n_A - 1)$ , подстановки

$$h_A = \begin{pmatrix} 1 & 2 & \mathbf{L} & (n_A - 1) \\ k_1 & k_2 & \mathbf{L} & k_{(n_A - 1)} \end{pmatrix}$$

множества номеров ОУ отражают симметрии такой подсхемы. Они составляют подгруппу автоморфизмов группы  $S_{qr} = S_q \otimes S_r$ . Обозначим ее  $H_A$ . Ее порядок равен  $(n_A - 1)!$ .

Симметрии подсхемы из ОУ можно выразить также с помощью подстановок  $g = (g_q, g_r) \in S_{qr}$ , преобразующих множество  $M_k$  в множество  $M_l$ , удовлетворяющее условию

$$M_l = \{(q_1(k_q), q_2(k_q), r_1(k_r), r_2(k_r))\} = M_k.$$

Количество  $N_i$  неизоморфных схем в множестве  $\Lambda$  определим как число подстановок  $g = (g_q, g_r) \in S_{qr}$ , не эквивалентных между собой по  $h \in H_A$ . Оно равно числу смежных классов разложения группы  $S_{qr}$  по подгруппе  $H_A$ . В соответствии с теоремой Лагранжа [18]

$$N_i = 2^{2n_A} n_A! / (n_A - 1)! = 2^{2n_A} n_A!$$

Значения  $N_i$  даны в табл. 4.2 для приведенных в ней примеров.

Аналогично учитывается влияние симметрий пассивной подсхемы на число вариантов, неизоморфных схеме  $\lambda_{i0}$ . Группу автоморфизмов пассивной подсхемы составляют подстановки множества номеров строк и столбцов матрицы  $\mathbf{Y}$ , к перестановке которых матрица  $\tilde{\mathbf{Y}}$  инвариантна. Обозначим эти подстановки  $h_\gamma$  ( $h_\nu$ ), если они меняют местами узлы с номерами  $q_1(k)$  и  $q_2(k)$  ( $r_1(k)$  и  $r_2(k)$ ). Они образуют группу  $H_\gamma$  ( $H_\nu$ ). Прочие автоморфизмы пассивной подсхемы обозначим  $H_{qr}$ .

При наличии указанных симметрий пассивной подсхемы число  $N'_i$  неизоморфных схем составляет в соответствии с теоремой Лагранжа

$$N'_i = [2^{2n_A} / (|H_\gamma| |H_\nu|)] \cdot [n_A! / |H_{qr}|].$$

#### 4.4. Эквивалентные преобразования и структурный синтез схем

Анализ структуры множества схемных реализаций заданной функции подсказывает наиболее рациональный подход к решению задачи структурного синтеза. Целесообразно разбить процесс синтеза на два этапа. На первом этапе должно быть построено множество неэквивалентных схем, т.е. схем-представителей своих классов эквивалентности. На втором этапе с помощью эквивалентных преобразований по выбранной проектировщиком схеме-представителю  $\lambda_{i0}$  восстанавливается весь класс эквивалентности. Покажем, что такой подход, обеспечивая существенное сокращение множе-

ства просматриваемых схемных вариантов, тем не менее, не приводит к потере искомой оптимальной схемы.

Показатели качества схемы, которые при проектировании выступают в роли критериев оптимальности, разделим на две категории. К первой отнесем те из них, которые определяются из аналитического выражения  $T(p)$  схемной функции, например, чувствительность к изменению параметров элементов, разброс параметров, удобство настройки и регулировки частотной характеристики, диапазон реализуемых значений нулей и полюсов. Ко второй категории отнесем показатели качества, для оценки которых недостаточно аналитического выражения схемной функции  $T(p)$ , вычисленной по (4.2). Это показатели, отражающие влияние паразитных связей в схеме и влияние "неидеальностей" ее элементов, в первую очередь ОУ, для учета которых требуется более точная математическая модель каждого элемента схемы, чем выбранные нами. К таким показателям относятся, например, устойчивость схемы к возбуждению в ней колебаний различных видов (высокочастотных, низкочастотных, релаксационного характера), уровень внутренних шумов.

При использовании более точных моделей элементов схемы передаточную функцию  $T(p)$  устройства в линейном режиме можно представить как

$$T(p) = \frac{\Delta_A + \delta_A}{\Delta_B + \delta_B},$$

где  $\Delta_A, \Delta_B$  – доминирующие составляющие, определяющие качественные показатели первой категории;  $\delta_A, \delta_B$  – остаточные составляющие, которые необходимо учитывать при анализе показателей второй категории. Для схем, относящихся к общему классу эквивалентности, доминирующие составляющие совпадают. Следовательно, для всего класса одинаковы качественные показатели первой категории. Возможность оценить их по одной схеме, представляющей весь класс, позволяет проектировщику сделать вывод о пригодности или непригодности для его целей всего класса на первом этапе проектирования. В этом случае эквивалентные преобразования применяются не ко всем схемам, а только к отобранным, т.е. имеющим наилучшие или удовлетворительные показатели первой категории. После этого на множестве эквивалентных схем отбираются схемы по качественным показателям второй категории.

Такая организация поиска оптимальной схемы обеспечивает адапта-

цию процесса синтеза к решаемой задаче. За счет адаптации появляется возможность нахождения оптимального решения в пространстве схем даже в случае, когда это пространство практически необозримо.

Как уже отмечалось, передаточная функция  $T(p)$ , вычисленная по (4.2), т.е. при использовании недостаточно точных математических моделей элементов схемы, не дает полного представления об устойчивости схемы. Однако известны [65 – 67] структурные признаки неустойчивости некоторых видов: если в схеме есть хотя бы один ОУ, не охваченный отрицательной обратной связью по постоянному току, схема неустойчива и, следовательно, неработоспособна. Необходимая информация об обратных связях ОУ на частоте  $\omega = 0$  содержится в матрице  $\tilde{\mathbf{G}}$ , которая является составляющей матрицы  $\tilde{\mathbf{Y}}$ , представляющей схему на этапах структурного синтеза и эквивалентных преобразований. Выявление неустойчивых схем сводится к выявлению цепочек обратной связи для каждого ОУ, которые включают в себя нечетное число ОУ. Оно легко автоматизируется [65; 68]. Может оказаться неустойчивой синтезированная схема-представитель, а эквивалентные ей схемы будут устойчивыми. Поэтому анализ обратных связей целесообразно выполнять на этапе восстановления по одной схеме всего класса эквивалентности, подвергая такому анализу исходную схему и каждую эквивалентную ей. Таким образом, на этапе эквивалентных преобразований отсекается существенное количество функционально непригодных схем без решения задач параметрического синтеза и оценки каждой схемы по различным качественным показателям.

Учитывая важность эквивалентных преобразований для задач структурного синтеза, остановимся на эффективных алгоритмах преобразований  $f_1, f_2, f_3, f_4$ . Как следует из п. 4.3, задача конструктивного перечисления эквивалентных схем сводится к комбинаторной задаче перебора подстановок  $g_q \in S_q, g_r \in S_r, g_\gamma \in S_\gamma, g_v \in S_v$ . Чтобы исключить изоморфные варианты, достаточно не допустить использования для преобразования подстановок, эквивалентных по симметриям  $h \in H_A$  подсхем из ОУ и по симметриям  $h_{qr} \in H_{qr}, h_\gamma \in H_\gamma, h_v \in H_v$  пассивной подсхемы. В этом состоит общий подход, рассмотренный в гл.1. В данном случае, если учесть специфику решаемой задачи, удастся построить более эффективную методику отсекаания изоморфных вариантов.

**У т в е р ж д е н и е 4.1.** Любое преобразование  $f_1$  приводит к изоморфным вариантам, если его применять к схеме, полученной из схемы  $\lambda_{i0} \in \Lambda$  с помощью преобразования  $f_2$ .



**Доказательство.** Действие подстановки  $g_q \in S_q$  преобразования  $f_2$  на схему  $\lambda_{i0}$  заключается в замене узлов  $q_1(k)$ ,  $q_2(k)$ , к которым подключены входные полюсы  $k$ -го ОУ, узлами  $q_1(l)$ ,  $q_2(l)$ , к которым подключены входные полюсы  $l$ -го ОУ, и наоборот,  $k, l \in \{1, 2, \dots, n_A\}$ . Если при этом выполнить дополнительную замену  $r_1(k)$ ,  $r_2(k)$  на  $r_1(l)$ ,  $r_2(l)$  и наоборот,  $k$ -й и  $l$ -й ОУ поменяются местами при неизменной пассивной подсхеме схемы  $\lambda_{i0}$ . Таким образом, получится схема, изоморфная схеме  $\lambda_{i0}$ . Утверждение доказано.

Если принять во внимание, что преобразование  $f_2$  не может привести к изоморфным схемам, из утверждения 4.1 следует сделать вывод, что преобразования  $f_1$ , реализуемые подстановками  $g_r \in S_r$ , необходимо исключить во избежание построения изоморфных схем. Тем самым исключается необходимость выявления порождающих подстановок из группы  $S_{qr}$ , эквивалентных по  $h \in H_A$ .

**Утверждение 4.2.** Подстановка  $g_\gamma \in S_\gamma$  приводит к изоморфным схемам, если хотя бы одна из одноименных образующих в ней и в  $h_\gamma \in H_\gamma$  одновременно имеет нечетную степень.

**Доказательство.** Пусть  $k$ -е образующие в  $g'_\gamma \in S_\gamma$  и в  $h_\gamma \in H_\gamma$  одновременно имеют нечетную степень. Тогда  $k$ -я образующая подстановки  $g''_\gamma = g'_\gamma h_\gamma$  имеет четную степень. Схемы, соответствующие  $g'_\gamma$  и  $g''_\gamma$ , отличаются только способом включения входных полюсов  $k$ -го ОУ. Нечетная степень  $k$ -й образующей  $h_\gamma \in H_\gamma$  свидетельствует о том, что передаточная функция  $T(p)$  (4.2) инвариантна к такому переключению входных полюсов ОУ. Значит, схемы соответствующие  $g'_\gamma \in S_\gamma$  и  $g''_\gamma \in S_\gamma$ , отличаются только номерами узлов  $q_1(k)$ ,  $q_2(k)$  и, следовательно, являются изоморфными.

На основании утверждения 4.2 построим алгоритм генерирования подстановок  $g_\gamma \in S_\gamma$ , не эквивалентных по  $h_\gamma \in H_\gamma$ . Для этой цели предлагается [65] подстановке  $g_\gamma \in S_\gamma$  поставить в соответствие  $n_A$ -разрядное двоичное число  $d_g$ , у которого  $k$ -й разряд равен нулю при четной степени  $k$ -й

образующей и единице – при нечетной. Очевидно, что это соответствие является взаимно однозначным. Аналогично представим  $h_\gamma \in H_\gamma$  двоичным числом  $d_h$ . Тогда единицы в  $k$ -х разрядах чисел  $d'_g, d''_g$  свидетельствуют о том, что  $k$ -е образующие соответствующих  $g_\gamma \in S_\gamma$  имеют нечетную степень. Нетрудно убедиться, что действиями над двоичными числами можно интерпретировать перебор подстановок, не эквивалентных между собой по  $h_\gamma \in H_\gamma$ .

Упорядочим  $n_A$ -разрядные двоичные числа по возрастанию. Если  $k$ -е разряды чисел  $d'_g, d_h$  являются единичными, то числа  $d''_g$  – нулевыми. Замена единичного разряда нулевым означает уменьшение двоичного числа. Таким образом, число  $d''_g$  является повторением двоичного числа  $d'_g$ , которое уже встречалось раньше в списке упорядоченных двоичных чисел, и, значит, его не следует использовать для построения подстановок, осуществляющих эквивалентные преобразования. Генерирование двоичных чисел в порядке их возрастания легко реализуется, если взять в качестве исходного двоичное число с нулями во всех  $n_A$  разрядах и прибавлять на каждом шаге по единице в младший разряд.

Алгоритм 4.1.

Шаг 1. Построить очередное двоичное число  $d_g$ .

Шаг 2. Сравнить  $d_g$  поразрядно с каждым из двоичных чисел  $d_h$ , соответствующих  $h_\gamma \in H_\gamma$ . Если хотя бы одно сравнение покажет совпадение одноименных единичных разрядов  $d_g, d_h$ , идти к шагу 1.

Шаг 3. Перейти от числа  $d_g$  к соответствующей подстановке.

Шаги 1 – 3 повторять, пока не исчерпаны  $n_A$ -разрядные двоичные числа.

Элементы группы  $H_\gamma$ , которые используются в алгоритме 4.1, находятся как подстановки  $g_\gamma \in S_\gamma$ , к которым инвариантна пассивная под-схема схемы  $\lambda_{i0}$ .

Очевидно, что все сказанное о подстановках  $g_\gamma \in S_\gamma$  (утверждение 4.2, алгоритм 4.1, способ выявления симметрий  $h_\gamma \in H_\gamma$ ), можно отнести и к подстановкам  $g_\nu \in S_\nu$ , так как они идентичны.

Теперь приведем алгоритм восстановления класса эквивалентности  $\Lambda_i$  схемы  $\lambda_{i0}$  за исключением изоморфных ей схем.

Алгоритм 4.2.

Шаг 1. Построить очередную подстановку  $g_q \in S_q$ .

Шаг 2. С помощью соотношений (4.6), (4.7) перейти от схемы  $\lambda_{i0}$  к эквивалентной схеме  $(\lambda_{i0})f_2$ .

Шаг 3. Построить очередную подстановку  $g_\gamma \in S_\gamma$  по алгоритму 4.1.

Шаг 4. От схемы  $(\lambda_{i0})f_2$  с помощью подстановки  $g_\gamma \in S_\gamma$  перейти к эквивалентной схеме  $((\lambda_{i0})f_2)f_3$ .

Шаг 5. Построить очередную подстановку  $g_v \in S_v$  по алгоритму 4.1.

Шаг 6. С помощью подстановки  $g_v \in S_v$  от схемы  $((\lambda_{i0})f_2)f_3$  перейти к эквивалентной схеме  $((\lambda_{i0})f_2)f_3f_4$ .

Шаги 5 – 6 повторять, пока не исчерпаны подстановки  $g_v \in S_v$ , шаги 3–6 – пока не исчерпаны подстановки  $g_\gamma \in S_\gamma$ , шаги 1–6 – пока не исчерпаны подстановки  $g_q \in S_q$ . Шаги 3 – 6 имеют смысл при дифференциальном включении ОУ. Для ОУ в трехполюсном включении они опускаются.

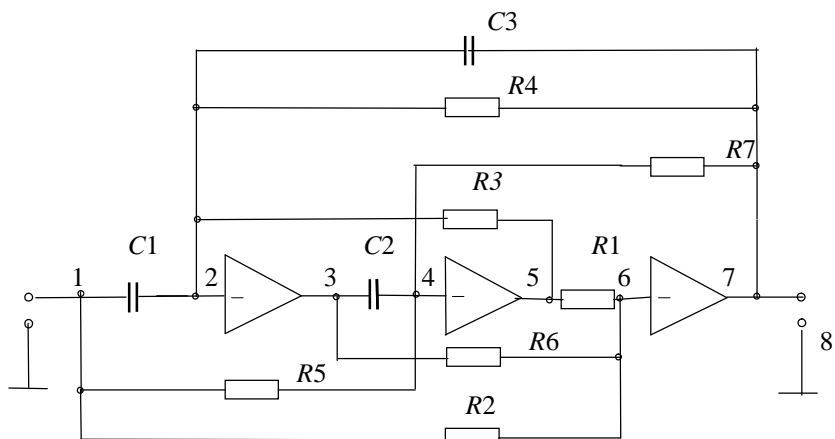
**Пример 4.1.** Если алгоритм 4.2 применить к схеме на ОУ в трехполюсном включении, показанной на рис. 4.1,  $a$ , с матрицей  $\tilde{Y}$

$$\begin{array}{c} \begin{array}{cccc} & 1 & 3 & 5 & 7 \\ \hline 2 & -pC_1 & & -G_3 & -pC_3 - G_4 \\ \hline 4 & -G_5 & -pC_2 & & -G_7 \\ \hline 6 & -G_2 & -G_6 & -G_1 & \end{array} \end{array}$$

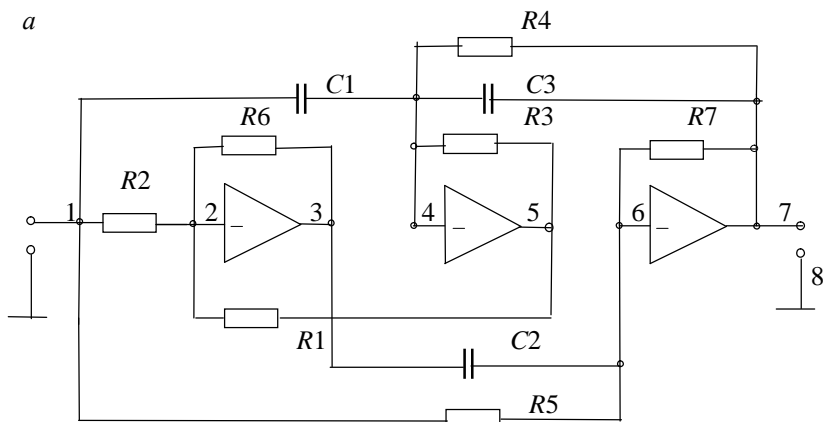
и передаточной функцией, вычисленной по (4.2),

$$T(p) = - \frac{p^2 C_1 C_2 G_1 - p C_2 G_2 G_3 + G_3 G_5 G_6}{p^2 C_2 C_3 G_1 + p C_2 G_1 G_4 + G_3 G_6 G_7},$$

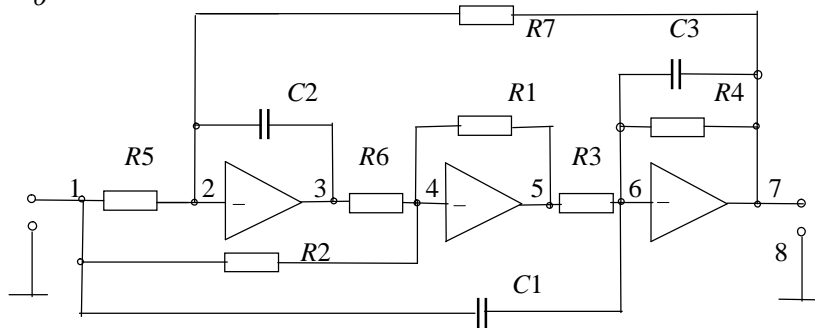
получим множество устойчивых эквивалентных схем, содержащее 2 элемента. Данные обо всем классе эквивалентности заданной схемы приведены в табл. 4.3. В ней для каждого двухполюсника указана пара номеров узлов, к которым он подключен в схеме. Устойчивые схемы, эквивалентные схеме рис. 4.1,  $a$  по  $T(p)$ , показаны на рис. 4.1,  $b$ ,  $в$ . В табл. 4.3 они имеют номера 5, 4 соответственно. Как видно из их пе-



*a*



*б*



*в*

Рис 4.1

редаточной функции  $T(p)$ , полученные схемы реализуют фазовый контур второго порядка, если  $C1 = C3, R1 = R2, R3 = R4, R5 = R7$ .

Таблица 4.3

Включение элементов к примеру 4.1

Элемент схемы	Варианты включения элементов					
	1	2	3	4	5	6
C1	1,2	1,2	1,4	1,6	1,4	1,6
C2	3,4	3,6	2,3	2,3	3,6	3,4
C3	2,7	2,7	4,7	6,7	4,7	6,7
R1	5,6	4,5	5,6	4,5	2,5	2,5
R2	1,6	1,4	1,6	1,4	1,2	1,2
R3	2,5	2,5	4,5	5,6	4,5	5,6
R4	2,7	2,7	4,7	6,7	4,7	6,7
R5	1,4	1,6	1,2	1,2	1,6	1,4
R6	3,6	3,4	3,6	3,4	2,3	2,3
R7	4,7	6,7	2,7	2,7	6,7	4,7

Забегая вперед, отметим, что алгоритм эквивалентных преобразований существенно эффективнее алгоритмов синтеза схем-представителей классов эквивалентности, которым посвящено последующее изложение. Это еще один аргумент в пользу предложенной организации структурного синтеза схем с использованием эквивалентных преобразований для порождения схемных вариантов.

#### 4.5. Формальная постановка задачи структурного синтеза на этапе реализации

На этапе реализации решения задачи структурного синтеза электронных устройств исходной является схемная функция, которая удовлетворяет условиям физической реализуемости в выбранном элементном базисе [55; 58]. Здесь выбран в качестве такой функции коэффициент передачи напряжения  $T(p)$  (4.1).

Условимся структуру функции (4.1) характеризовать двумя векторами  $\mathbf{A} = (a_0, a_1, \dots, a_n)$ ,  $\mathbf{B} = (b_0, b_2, \dots, b_n)$ , где

$$a_i' = \begin{cases} 1 & \text{при } a_i > 0, \\ 0 & \text{при } a_i = 0, \\ -1 & \text{при } a_i < 0; \end{cases} \quad (4.8)$$

$$b_j' = \begin{cases} 1 & \text{при } b_j > 0, \\ 0 & \text{при } b_j = 0. \end{cases} \quad (4.9)$$

В векторах **A**, **B** отражены спектры степеней [51] числителя и знаменателя функции (4.1), а также основные характеристики ее полиномиальных коэффициентов.

Представим функцию (4.2) схемы с ОУ в виде отношения полиномов

$$T(p) = \frac{\sum_{i=1}^{n^*} f_i p^i}{\sum_{j=1}^{n^*} \varphi_j p^j}, \quad (4.10)$$

где  $f_i$ ,  $\varphi_j$  – аналитические выражения полиномиальных коэффициентов  $a_i$ ,  $b_j$  через параметры элементов схемы, т. е. функции этих параметров. Для того чтобы функция  $f_i$  ( $\varphi_j$ ) могла реализовать полиномиальный коэффициент  $a_i \neq 0$  ( $b_j \neq 0$ ) из (4.1), необходимо, чтобы она содержала хотя бы одно слагаемое. Для реализации  $a_i < 0$  необходимо, чтобы соответствующая функция  $f_i$  содержала хотя бы одно отрицательное слагаемое. Для реализации  $a_i = 0$  ( $b_j = 0$ ) необходимо, чтобы либо  $f_i = 0$  ( $\varphi_j = 0$ ), т.е. в функции  $f_i$  ( $\varphi_j$ ) не было ни одного слагаемого (структурный нуль), либо в ней было не менее двух слагаемых с противоположными знаками (параметрический нуль).

Будем считать схему с функцией (4.10) решением задачи структурного синтеза на этапе реализации, если структуры функций (4.1), (4.10) одинаковы. Известно [51], что схема, обеспечивающая требуемую структуру заданной схемной функции, может рассматриваться как претендент на решение задачи структурного синтеза в целом. Окончательное заключение, является ли схема таким решением, выносится только после параметрического синтеза, т. е. после решения системы компонентных [57] уравнений

$$\begin{aligned} f_i &= a_i, \quad i = 1, 2, \dots, n^*; \\ \varphi_j &= b_j, \quad j = 1, 2, \dots, n^*. \end{aligned}$$

Таким образом, на этапе реализации осуществляется предварительный отбор с целью исключить из рассмотрения все прочие схемы.

Для дальнейших рассуждений построим вектор  $\mathbf{D} = (d_1, d_2, \dots, d_L)$  из коэффициентов  $a_i', b_j', i, j \in \{1, 2, \dots, n\}$ , которые отличны от нуля, и вектор  $\mathbf{D}_0 = (d_{01}, d_{02}, \dots, d_{0L_0})$  из нулевых коэффициентов (4.1). Здесь  $L (L_0)$  – общее число ненулевых (нулевых) полиномиальных коэффициентов функции  $T(p)$  (4.1). В частности, при реализации структурного нуля  $a_i = 0$  ( $b_j = 0$ ) этот коэффициент следует отнести к  $\mathbf{D}_0$ , а параметрического – к  $\mathbf{D}$ .

Кроме требования к спектру степеней функции (4.2) введем ограничение на число слагаемых в каждом полиномиальном коэффициенте этой функции. Известно, что большое число слагаемых в полиномиальных коэффициентах, как правило, означает невозможность развязанной настройки схемы. Следовательно, с помощью таких ограничений можно косвенно влиять на некоторые качественные показатели искомой схемы. Ограничения условимся задавать в виде вектора  $\mathbf{D}_s = (d_{s1}, d_{s2}, \dots, d_{sl}, \dots, d_{sL})$ , аналогичного вектору  $\mathbf{D}$ . В нем  $d_{sl}$  – допустимое число слагаемых в полиномиальном коэффициенте  $a_i$  или  $b_j$ , соответствующем элементу  $d_l \in \mathbf{D}$ .

Совокупностью векторов  $\mathbf{D}, \mathbf{D}_0, \mathbf{D}_s$ , таким образом, формально представлены требования к схемной функции искомой схемы и структуре ее полиномиальных коэффициентов, вытекающие из конкретной постановки задачи структурного синтеза. В широко распространенном в практике случае требуется реализация  $T(p)$  с точностью до знака. Поэтому условимся допускать одновременную замену всех коэффициентов  $a_i$  на  $(-a_i)$  и (или) всех коэффициентов  $b_j$  на  $(-b_j)$ .

Допустимый набор структурных элементов искомой схемы зададим числом  $n_z$  ее узлов, числом  $n_A$  ОУ, числом  $n_G$  резистивных двухполюсников и числом  $n_C$  емкостных двухполюсников. Остановимся на подходе к поиску оптимальной схемы, при котором сначала просматривается множество схем с минимальным числом элементов, необходимых для реализации заданной функции (4.1), т.е. простейшие схемы. Как уже говорилось, известны методы ([51; 58] и др.), позволяющие определить это число в каждой конкретной задаче структурного синтеза. Простейшие схемы, как правило, представляют наибольший интерес. Если ни одна из найденных простейших схем не обладает совокупностью качественных показателей, предусмотрен-

ных техническим заданием, число структурных элементов расширяется путем введения дополнительных ОУ, сопротивлений и емкостей, а также за счет увеличения числа узлов схемы. Для этой же цели используют переход к другим способам включения ОУ, например, от трехполюсного включения к дифференциальному.

Если  $n_c > n$ , есть опасность появления в схемной функции (4.2) ненулевых слагаемых  $a_i, b_j$  с  $i > m, j > n$ . В этом случае вектор  $\mathbf{D}_0$  следует дополнить  $n^* + n^* - n - m$  элементами  $a'_{(m+1)} = a'_{(m+2)} = \dots = a'_{n^*} = 0$ ,  $b'_{(n+1)} = b'_{(n+2)} = \dots = b'_{n^*} = 0$ .

Условимся рассматривать следующие практически оправданные способы включения ОУ в схему:

1. Общий полюс каждого ОУ объединен с общим (заземленным) узлом схемы.
2. Выход одного из ОУ совпадает с выходом схемы, что исключает влияние нагрузки на свойства схемы и обеспечивает согласование ее с последующими схемами.
3. ОУ не имеют непосредственных связей между собой. Это ограничение практически снимается на этапе параметрического синтеза, так как такие связи появляются, если параметры некоторых двухполюсников принимают предельные значения (0 или  $\infty$ ).
4. Способ включения всех ОУ одинаков: либо дифференциальное, либо трехполюсное. Такое ограничение также снимается в результате решения задачи параметрического синтеза.

После определения допустимого набора структурных элементов искомой схемы задача структурного синтеза схем с заданной функцией  $T(p)$  сводится к задаче воссоздания матрицы  $\tilde{\mathbf{Y}}$ , миноры которой удовлетворяли бы (4.1). Из п. 4.3 следует, что эта задача решается неоднозначно, и в процессе поиска ее решения неизбежно появление изоморфных вариантов.

При выбранном способе включения ОУ множество узлов схемы делится на непересекающиеся подмножества тех, к которым подключены только входные или только выходные полюсы ОУ, а также тех, к которым вообще не подключены ОУ. Такому разбиению узлов схемы соответствует разбиение матрицы  $\tilde{\mathbf{Y}}$  на блоки, каждый из которых представляет либо взвешенную матрицу инцидентностей, либо взвешенную матрицу смежности, либо вариант последней, полученный суммированием элементов пар ее строк (столбцов). Следовательно, для устранения изоморфных вариантов в процессе структурного синтеза таких схем применимы средства из глав 1, 2.



Покажем, что для рассматриваемого приложения эту задачу можно решить рациональнее.

Выделим следующие классы схем с ОУ:

- 1) с трехполюсным включением ОУ и внутренними узлами, каждый из которых объединен с полюсом одного из ОУ;
- 2) всех прочих схем с трехполюсным включением ОУ;
- 3) с дифференциальным включением ОУ.

#### 4.6. Реализация слагаемого полиномиального коэффициента передаточной функции

Рассмотрим строку матрицы  $\tilde{\mathbf{Y}}$  как подмножество ее элементов  $y_{ij}$ , а матрицу  $\tilde{\mathbf{Y}}$  – как семейство таких подмножеств. Построим для этого семейства трансверсаль [1]  $\mathbf{y}_t = (y_{1\beta_1}, y_{2\beta_2}, \dots, y_{\rho\beta_\rho})$  такую, что  $\beta_i \neq \beta_j$  при  $i \neq j$ ,  $i, j \in \{1, 2, \dots, \rho\}$ . Построим матрицу  $\mathbf{Y}_l = \left\| y_{\alpha\beta} \right\|_{\rho \times \rho}$ , у которой ненулевыми являются только элементы, относящиеся к трансверсали  $\mathbf{y}_t$ :  $y_{\alpha\beta} \neq 0$  при  $y_{\alpha\beta} \in \mathbf{y}_t$ ; кроме того,  $\eta$  элементов трансверсали  $\mathbf{y}_t$  являются элементами матрицы  $\mathbf{C}$ , а остальные  $(\rho - \eta)$  элементов – матрицы  $\mathbf{G}$ . Наличие в матрице  $\tilde{\mathbf{Y}}$  блока  $\tilde{\mathbf{Y}}_A$  ( $\tilde{\mathbf{Y}}_B$ ), содержащего матрицу  $\mathbf{Y}_l$  в качестве слагаемого, достаточно, чтобы коэффициент при  $p^\eta$  в числителе (знаменателе)  $T(p)$  был ненулевым. Это позволяет применить матрицу  $\mathbf{Y}_l$  для представления элементарной подсхемы, реализующей  $a'_i \in \mathbf{D}$  ( $b'_j \in \mathbf{D}$ ), и свести построение таких подсхем к конструктивному перечислению матриц  $\mathbf{Y}_l$ .

Из сказанного следует, что многообразие схемных реализаций заданной схемной функции естественно рассматривать как следствие неоднозначности реализации каждого ее полиномиального коэффициента. Таким образом, выделена комбинаторная подзадача задачи структурного синтеза схем, от эффективности решения которой зависит эффективность решения всей задачи в целом. Наилучшим является линейный алгоритм, временные затраты на выполнение которого пропорциональны числу возможных решений поставленной задачи. Поэтому сначала оценим это число.

**У т в е р ж д е н и е 4.3.** *Элементарная матрица  $\mathbf{Y}_l$ , реализующая коэффициент при  $p^\eta$ , может быть построена  $\eta!C(\rho, \eta)$  способами.*

Доказательство. Возьмем произвольную матрицу  $\mathbf{Y}_l$ . Переход к другим вариантам  $\mathbf{Y}_l$  будем осуществлять применением подстановок

$$g_\alpha = \begin{pmatrix} 1 & 2 & \mathbf{L} & \rho \\ \alpha_1 & \alpha_2 & \mathbf{L} & \alpha_\rho \end{pmatrix}$$

к множеству  $\{1, 2, \dots, \rho\}$  номеров ее строк или (и) подстановок

$$g_\beta = \begin{pmatrix} 1 & 2 & \mathbf{L} & \rho \\ \beta_1 & \beta_2 & \mathbf{L} & \beta_\rho \end{pmatrix}$$

к множеству  $\{1, 2, \dots, \rho\}$  номеров ее столбцов.

Множество подстановок  $g_\alpha$  образует симметрическую группу  $S_\alpha$ , множество подстановок  $g_\beta$  – симметрическую группу  $S_\beta$ . Построим группу  $S_{\alpha\beta} = S_\alpha \otimes S_\beta$  подстановок  $g = (g_\alpha, g_\beta)$  множества номеров строк и столбцов матрицы  $\mathbf{Y}_l$ . Очевидно, что число таких подстановок

$$|S_{\alpha\beta}| = \rho! \rho! . \quad (4.11)$$

Однако некоторые из них не меняют матрицы, т.е. для них выполняется равенство

$$\mathbf{G}_\alpha^{-1} \mathbf{Y}_l \mathbf{G}_\beta = \mathbf{Y}_l , \quad (4.12)$$

где  $\mathbf{G}_\alpha, \mathbf{G}_\beta$  – матричные подстановки, соответствующие подстановкам  $g_\alpha, g_\beta$ . Подстановки  $g = (g_\alpha, g_\beta)$ , которые удовлетворяют (4.12), составляют группу автоморфизмов матрицы  $\mathbf{Y}_l$ , определяющих ее симметрии. Пусть  $H$  – группа автоморфизмов матрицы  $\mathbf{Y}_l$ , а  $h = (h_\alpha, h_\beta)$  – ее элемент. Группа  $H$  разбивает группу  $S_{\alpha\beta}$  на смежные классы. Любой элемент одного класса отличается от другого множителем  $h \in H$ , т.е. они являются изоморфными. Следовательно, множество неизоморфных вариантов составляет система представителей от каждого класса эквивалентности, и их число совпадает с числом  $N_\rho$  смежных классов. Последнее определяется по теореме Лагранжа

$$N_\rho = |S_{\alpha\beta}| / |H|. \quad (4.13)$$

Для определения порядка группы  $H$  учтем природу симметрий матрицы  $\mathbf{Y}_l$ : инвариантность  $\mathbf{Y}_l$  к некоторым подстановкам определяется

тем, что в нее входят только элементы двух типов – емкостные и резистивные проводимости. Перестановка элементов одного типа между собой не меняет матрицы  $\mathbf{Y}_l$  и может быть выражена через одновременные подстановки соответствующих строк и столбцов. Поэтому порядок группы автоморфизмов определяется числом элементов  $\eta$  одного типа и  $(\rho - \eta)$  – другого типа:

$$|H| = \eta!(\rho - \eta)! \quad (4.14)$$

Из (4.11), (4.13), (4.14) следует, что число различных вариантов матрицы  $\mathbf{Y}_l$  равно

$$N_\rho = (\rho! \rho!) / (\eta!(\rho - \eta)!) = \rho! C(\rho, \eta).$$

Утверждение доказано.

Из утверждения 4.3 следует, что алгоритм построения всех возможных вариантов матрицы  $\mathbf{Y}_l$  оптимален, если временные затраты на его выполнение пропорциональны величине  $N_\rho = \rho! C(\rho, \eta)$ .

Построим такой алгоритм, выделив в нем два этапа:

1. Построение вариантов диагональной матрицы  $\mathbf{Y}_{\beta l}$  с трансверсалью  $\mathbf{y}_t = (y_{11}, y_{22}, \dots, y_{\rho\rho})$  из исходной матрицы  $\mathbf{Y}_{ll}$ ,

$$\mathbf{Y}_{\beta l} = \mathbf{G}_\beta^{-1} \mathbf{Y}_{ll} \mathbf{G}_\beta, \quad (4.15)$$

где  $\mathbf{G}_\beta$  – перестановочная матрица, соответствующая подстановке  $g_\beta \in S_\beta$ , которую условимся представлять вектором  $\mathbf{g}_\beta = (\beta_1, \beta_2, \mathbf{L}, \beta_\rho)$ .

2. Переход от каждой матрицы  $\mathbf{Y}_{\beta l}$  к вариантам недиагональной матрицы

$$\mathbf{Y}_l = \mathbf{G}_\alpha^{-1} \mathbf{Y}_{\beta l}, \quad (4.16)$$

где  $\mathbf{G}_\alpha$  – перестановочная матрица, соответствующая подстановке  $g_\alpha \in S_\alpha$ .

Учтем, что  $\mathbf{y}_t$  –  $\rho$ -мерный вектор с повторяющимися координатами. Подстановки на множествах его однотипных элементов, к которым вектор  $\mathbf{y}_t$  инвариантен, образуют его группу автоморфизмов  $H$  порядка  $\eta!(\rho - \eta)!$ . Каждая из подгрупп группы  $H$  порядка  $\eta!$  и порядка  $(\rho - \eta)!$  – это симметрическая группа подстановок элементов вектора  $\mathbf{y}_t$  одного из типов. Условимся рассматривать координаты вектора  $\mathbf{g}_\beta = (\beta_1, \beta_2, \mathbf{L}, \beta_\rho)$  в качестве порядковых номеров элементов вектора  $\mathbf{y}_t$ , а те из них, что относятся к од-

нотипным элементам, – в качестве эквивалентных по симметриям  $h \in H$ , т.е.  $h$ -классов. Из утверждения 2.1 следует, что для отбора векторов  $\mathbf{g}_\beta$ , не эквивалентных по симметриям  $h \in H$ , достаточно рассматривать только те из них, у которых координаты каждого  $h$ -класса упорядочены лексикографически.

Алгоритм 4.3

Шаг 1. Построить матрицу  $\mathbf{Y}_{1l}$ , в которой ненулевые элементы первых  $\eta$  строк являются проводимостями емкостного характера, а остальные – резистивного. Определить классы эквивалентности по ее симметриям.

Шаг 2. С учетом утверждения 2.1 построить очередной канонический вектор  $\mathbf{g}_\beta = (\beta_1, \beta_2, \mathbf{L}, \beta_\rho)$ .

Шаг 3. Построить очередной вариант матрицы  $\mathbf{Y}_{\beta l}$  по (4.15).

Шаг 4. Построить очередную подстановку  $g_\alpha \in S_\alpha$ .

Шаг 5. Построить очередной вариант матрицы  $\mathbf{Y}_l$  по (4.16).

Шаги 4 – 5 необходимо повторять, пока не будут исчерпаны подстановки  $g_\alpha$ , шаги 2 – 5 – пока не будут исчерпаны подстановки  $g_\beta$ .

Очевидно, что матрицы  $\mathbf{Y}_l$ , построенные по алгоритму 4.3, не повторяются, и общее число таких матриц равно  $N_\rho = \rho! C(\rho, \eta)$ , так как число повторяющихся шагов построения равно числу искомых решений. Следовательно, рассмотренный алгоритм является оптимальным.

Условимся подстановку  $g = (g_\alpha, g_\beta) \in S_{\alpha\beta}$  называть порождающей, так как согласно алгоритму 4.3 каждому варианту  $g_\beta$  поставлена в соответствие матрица  $\mathbf{Y}_{\beta l}$ , а каждому варианту  $g_\alpha$  – матрица  $\mathbf{Y}_l$ . Пусть множества подстановок  $g_\alpha, g_\beta$  упорядочены, как предложено в п. 2.4.3. Таким порядком индуцируется порядок на множестве последовательностей  $g = (g_\alpha, g_\beta)$ : это множество упорядочено по включению в виде дерева  $T_2$ , показанного на рис. 4.2. Тем самым индуцируется порядок на множе-

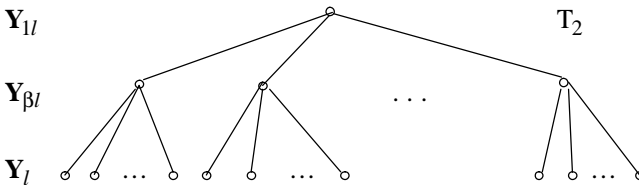


Рис. 4.2

стве генерируемых матриц  $\mathbf{Y}_B, \mathbf{Y}_I$ . Конструктивное перечисление всех возможных матриц  $\mathbf{Y}_I$  по алгоритму 4.3 интерпретируется, таким образом, обходом дерева  $T_2$ , показанного на рис. 4.2, с концевыми вершинами, соответствующими искомым матрицам  $\mathbf{Y}_I$ .

Матрицу  $\tilde{\mathbf{Y}}$ , блок  $\tilde{\mathbf{Y}}_A(\tilde{\mathbf{Y}}_B)$  которой совпадает с матрицей  $\mathbf{Y}_I$ , назовем элементарной и обозначим  $\tilde{\mathbf{Y}}_I'$ . Не всякая матрица  $\tilde{\mathbf{Y}}_I'$ , как и матрица  $\tilde{\mathbf{Y}}$ , является физически реализуемой, но ее можно привести к таковой. Условия их физической реализуемости определяются известными [51; 58] условиями физической реализуемости матрицы  $\mathbf{Y}$ : если элемент  $y_{ij}$  матрицы  $\mathbf{Y}$  ненулевой, то одновременно ненулевыми являются элементы  $y_{ji}, y_{ii}, y_{jj}$ ; при этом

$$y_{ji} = y_{ij}; \quad (4.17)$$

$$y_{ii} = y_{jj} = -y_{ij}. \quad (4.18)$$

Такая четверка ненулевых элементов матрицы  $\mathbf{Y}$  реализуется двухполюсником с проводимостью  $y = -y_{ij}$ , включенным между  $i$ - и  $j$ -м узлами схемы.

Элементы  $y_{ii}, y_{ij}, y_{ji}, y_{jj}$ , которые не вычеркиваются при переходе от матрицы  $\mathbf{Y}$  к матрице  $\tilde{\mathbf{Y}}$ , также должны быть ненулевыми в физически реализуемой матрице  $\tilde{\mathbf{Y}}$ . Они вычисляются с помощью соотношений (4.17), (4.18) для каждого элемента  $y_{ij} \in \mathbf{Y}$ . Тогда и переход от матрицы  $\tilde{\mathbf{Y}}_I'$  к физически реализуемой матрице  $\tilde{\mathbf{Y}}_I$  осуществляется путем пополнения первой элементами, соответствующими  $y_{ii}, y_{ji}, y_{jj}$ . Такое пополнение выглядит для классов схем, выделенных в п. 4.5, по-разному.

При переходе в соответствии с (4.2) от матрицы  $\mathbf{Y}$  схемы 1-го класса к матрице  $\tilde{\mathbf{Y}}$  элементы  $y_{ii}, y_{ji}, y_{jj}$  вычеркиваются. Значит, матрицы  $\tilde{\mathbf{Y}}_I'$  и  $\mathbf{Y}_I$  физически реализуемы, т.е. их пополнение при переходе к матрице  $\tilde{\mathbf{Y}}_I$  не требуется. Между матрицей  $\mathbf{Y}_I$  и элементарной подсхемой существует взаимно однозначное соответствие.  $\tilde{\mathbf{Y}}_I$  – это взвешенная матрица инцидентностей.

Для схем 2-го и 3-его классов характерно, что элемент матрицы  $\mathbf{Y}$ , отвечающий элементу трансверсали  $\mathbf{y}_t$ , может оказаться диагональным (при условии равенства  $i = j$ ). Схемотехническая интерпретация такого эле-

мента – подключение к  $i$ -му узлу схемы некоторого двухполюсника. Но в матрице  $\mathbf{Y}_l$ , соответствующей  $\mathbf{y}_t$ , отсутствует информация об узле схемы, к которому подключен второй полюс двухполюсника, что свидетельствует о возможности построения различных вариантов физически реализуемой матрицы  $\tilde{\mathbf{Y}}_l$  по одной матрице  $\mathbf{Y}_l$ , если по-разному выбирать пополняющие элементы  $y_{ij}$ ,  $y_{ji}$ ,  $y_{jj}$ . В частности, отсутствие пополнения интерпретируется заземлением второго полюса двухполюсника. Таким образом, в данном случае матрица  $\mathbf{Y}_l$  представляет целое семейство элементарных подсхем, соответствующих различным вариантам ее пополнения. Пусть трансверсаль  $\mathbf{y}_t$  содержит  $\mu$  диагональных элементов матрицы  $\mathbf{Y}$ . Тогда процедура пополнения  $\mathbf{Y}_l$  должна содержать  $\mu$  вложенных циклов, каждым из которых интерпретируется переключение второго полюса соответствующего двухполюсника. При  $\mu = 0$  матрица  $\mathbf{Y}_l$  представляет единственную элементарную подсхему. Матрица  $\tilde{\mathbf{Y}}_l$  этой подсхемы строится путем пополнения матрицы  $\tilde{\mathbf{Y}}_l'$  с учетом (4.17), (4.18).

В дереве вариантов  $T_3$  для схем 2-го и 3-го классов, показанном на рис. 4.3, переход от  $\mathbf{Y}_l$  к  $\tilde{\mathbf{Y}}_l$  интерпретируется добавлением третьего яруса.

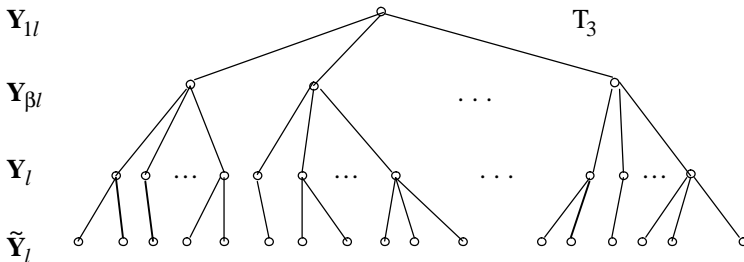


Рис. 4.3

#### 4.7. Реализация заданного спектра степеней

Одновременная реализация всех ненулевых слагаемых обеспечивается путем объединения одноименных узлов отдельных подсхем, каждая из которых реализует одно из таких слагаемых, с одновременной заменой каждого параллельного соединения однотипных ветвей одной ветвью. Для схем

1-го класса требуемое соотношение знаков полиномиальных коэффициентов числителя и знаменателя (4.1) достигается отсечением части порождающих подстановок  $g_\alpha \in S_\alpha$ : для того чтобы очередной из реализуемых коэффициентов  $a_i (b_j)$  по знаку совпадал с первым, необходимо при их реализации использовать подстановки с одинаковой четностью, а чтобы не совпадал – с различной. Для схем 2-го и 3-го классов можно применить этот же прием, если учесть, что каждый диагональный элемент матрицы  $\mathbf{Y}$ , вошедший в  $\mathbf{y}_l$ , означает изменение знака отвечающего ему полиномиального коэффициента на обратный. При определении четности порождающей подстановки нетрудно ввести соответствующую поправку [65].

Формализуем операцию объединения отдельных подсхем.

Пусть  $U_l$  – множество элементарных матриц  $\mathbf{Y}_l$ , реализующих  $l$ -й элемент множества  $\mathbf{D}$ . Выберем из каждого  $U_l$ ,  $l = 1, 2, \dots, L$ , по одному представителю и построим для каждого элементарную матрицу  $\tilde{\mathbf{Y}}_l'$  по следующему принципу: если матрица  $\mathbf{Y}_l$  реализует  $a_i \neq 0$ , добавим к ее столбцам  $i$ -й столбец из нулевых элементов; если  $b_j \neq 0$  –  $v$ -й столбец из таких же элементов. Перейдем от матрицы  $\tilde{\mathbf{Y}}_l'$  к матрице  $\tilde{\mathbf{Y}}_l$ . Общий элемент матрицы  $\tilde{\mathbf{Y}}_l$  обозначим  $(y_{\alpha\beta})_l$ . Пусть  $\tilde{\mathbf{Y}}^{(l)}$  – матрица с общим элементом  $y_{\alpha\beta}^{(l)}$ , построенная из  $l$  различных элементарных матриц  $\tilde{\mathbf{Y}}_l$  по следующей рекуррентной формуле для  $l = 1, 2, \dots, L$ :

$$y_{\alpha\beta}^{(l)} = \begin{cases} y_{\alpha\beta}^{(l-1)} & \text{при } (y_{\alpha\beta})_l = y_{\alpha\beta}^{(l-1)}; \\ y_{\alpha\beta}^{(l-1)} ((y_{\alpha\beta})_l) & \text{при } (y_{\alpha\beta})_l = 0 \text{ (} y_{\alpha\beta}^{(l-1)} = 0 \text{)}; \\ G_{\alpha\beta} + pC_{\alpha\beta} & \text{при других условиях.} \end{cases} \quad (4.19)$$

Операция (4.19) построения матрицы  $\tilde{\mathbf{Y}}^{(l)}$  отвечает операции склеивания  $k$ -деревьев [63; 68], поэтому назовем ее склеиванием подсхем. Реализация всех  $L$  ненулевых полиномиальных коэффициентов  $a_i, b_i \in \mathbf{D}$  может быть организована в виде циклической процедуры с параметром  $l$ : на  $l$ -м шаге матрица  $\tilde{\mathbf{Y}}^{(l-1)}$  склеивается с матрицей  $\mathbf{Y}_l$  в соответствии с (4.19). При этом предполагается, что  $\tilde{\mathbf{Y}}^{(l-1)} = \tilde{\mathbf{Y}}^{(0)}$  – матрица, все элементы которой являются нулевыми. Конструктивный процесс, на  $l$ -м шаге которого

матрица  $\tilde{\mathbf{Y}}^{(l-1)}$ ,  $l \in \{1, 2, \dots, L\}$ , поочередно склеивается с каждой матрицей  $\mathbf{Y}_l$ , гарантирует получение всех возможных схем, удовлетворяющих условиям (4.8), (4.9) для  $a_i, b_i \in \mathbf{D}$  и обеспечивающих хотя бы по одному слагаемому в ненулевых полиномиальных коэффициентах (4.1). Этот конструктивный процесс интерпретируется обходом дерева  $T_4$  вариантов  $\tilde{\mathbf{Y}}^{(l)}$  (рис. 4.4), содержащего  $|L|$  ярусов, из каждой вершины которого выходит  $|U_l|$  ветвей.

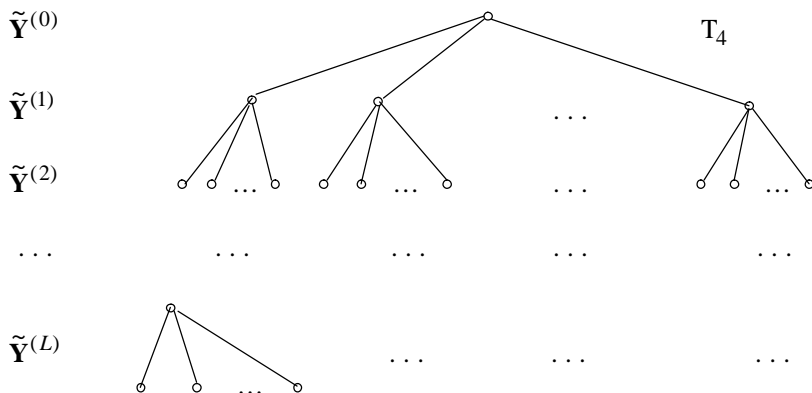


Рис. 4.4

Каждая матрица  $\tilde{\mathbf{Y}} = \tilde{\mathbf{Y}}^{(L)}$ , соответствующая концевой вершине дерева  $T_4$ , однозначно определяет схему. Однако среди схем, полученных таким образом, есть и такие, что ни при каких значениях параметров элементов не могут реализовать заданную функцию  $T(p)$ . Будем называть их функционально непригодными. Окончательное решение о функциональной пригодности или непригодности схемы можно принять только на этапе параметрического синтеза. Решение задачи параметрического синтеза выходит за рамки этой работы. Но есть и "структурные" признаки функциональной непригодности, по которым схема может быть отсечена на одном из этапов структурного синтеза. Рассмотрим некоторые из них.

К таким структурным признакам относится наличие хотя бы одной строки (столбца за исключением  $l$ -го и  $w$ -го) в матрице  $\tilde{\mathbf{Y}}$  с единствен-



ным ненулевым элементом  $y_{\alpha\beta}$ . В физически реализуемой матрице  $\mathbf{Y}$  ему отвечает элемент  $y_{ij}$ , в схеме – двухполюсник между  $i$ - и  $j$ -м узлами. При  $i = j$  через этот двухполюсник  $i$ -й узел соединен только с общим узлом схемы. Следовательно, в реальном устройстве он обесточен. При  $i = q(k)$ ,  $j = r(l)$ ,  $k = l$ , через такой двухполюсник осуществляется отрицательная обратная связь, которой охвачен ОУ с номером  $k$ . Но при этом в схеме отсутствуют цепи, по которым сигнал поступает на вход этого ОУ или снимается с его выхода. Указанный ОУ фактически не используется в схеме. Рассмотренным случаям соответствуют сократимые множители в числителе и знаменателе схемной функции  $T(p)$ , в том числе понижающие ее порядок. Такое сокращение означает, что полученная схема содержит набор структурных элементов, отличный от заданного, так как часть элементов на самом деле не принимает участия в работе соответствующего устройства. Допустим теперь, что  $k \neq l$ , т.е. один полюс указанного двухполюсника подключен к входу (выходу) одного ОУ, а второй – к выходу (входу) другого. С помощью эквивалентных преобразований этот случай приводится к только что рассмотренному. Но при эквивалентных преобразованиях передаточная функция не меняется. Следовательно, она имеет тот же недостаток – сократимые множители. Из сказанного следует, что необходимо отбирать только те матрицы  $\tilde{\mathbf{Y}}$ , у которых любая строка и любой столбец, кроме  $n$ -го и  $m$ -го, содержит не менее двух ненулевых элементов. Очевидно, что отсеечение вариантов матриц  $\tilde{\mathbf{Y}}$ , не удовлетворяющих этому требованию, может быть выполнено только на последнем ярусе дерева  $T_4$ .

Представленный способ построения схемных структур может приводить к схемам, число элементов которых превосходит заданное. Для отсеечения таких схем достаточно по матрице  $\tilde{\mathbf{Y}}^{(l)}$  вести подсчет числа элементов каждого типа. Если на  $l$ -м шаге конструктивного процесса число элементов некоторого типа превышает допустимое, эта матрица дальше не рассматривается, и в дереве вариантов  $T_4$  отсекается соответствующая вершина вместе с висящим на ней поддеревом.

При склеивании матриц  $\mathbf{Y}_l$ , реализующих ненулевые полиномиальные коэффициенты  $a_i \in \mathbf{D}$  ( $b_i \in \mathbf{D}$ ), может образоваться трансверсаль  $\mathbf{y}_l$ , обуславливающая неравенство нулю коэффициентов  $a_i \in \mathbf{D}_0$  или (и)  $b_i \in \mathbf{D}_0$ . Если по условиям задачи соответствующий ноль должен быть структурным, можно не продолжать дальнейшее формирование матрицы  $\tilde{\mathbf{Y}}$ . Это соответствует отсечению соответствующей вершины дерева  $T_4$  вместе с висящим на ней поддеревом. Кроме того, на каждом шаге конструктивного процесса

необходимо следить, чтобы число трансверселей, соответствующих  $a_i \in \mathbf{D}$  ( $b_i \in \mathbf{D}$ ), не превышало  $a_{si}$  ( $b_{si}$ ), если заданы ограничения на число слагаемых в полиномиальных коэффициентах (4.2). Если это произойдет при некотором значении  $l$ , продолжать построение не следует. В дереве вариантов  $T_4$ , отсечению такой матрицы на  $l$ -м шаге также отвечает отсечение вершины на  $l$ -м ярусе вместе с соответствующим поддеревом.

По структурным признакам, но более трудоемким, чем перечисленные здесь, удастся распознать и другие "дефекты" сгенерированных схем [65].

#### 4.8. Отсечение изоморфных матриц $\tilde{\mathbf{Y}}$

Напомним, что множество неизоморфных матриц  $\tilde{\mathbf{Y}}$  представляет множество неизоморфных и неэквивалентных схем.

Для отсечения изоморфных матриц  $\tilde{\mathbf{Y}}$  воспользуемся на каждом ярусе деревьев  $T_2, T_3, T_4$  информацией о тех симметриях составляющих  $\mathbf{Y}_{I_l}, \mathbf{Y}_{\beta l}, \tilde{\mathbf{Y}}^{(l-1)}$  матрицы  $\tilde{\mathbf{Y}}^{(l)}$ , которые совпадают с симметриями соответствующих подсхем (см. п. 4.3).

Пусть  $h = (h_\alpha, h_\beta)$  – элемент группы автоморфизмов  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)})$  матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$ , отражающей ее симметрии,  $h_1 = (h_{1\alpha}, h_{1\beta})$  – элемент группы  $\text{Aut}(\mathbf{Y}_{I_l})$  матрицы  $\mathbf{Y}_{I_l}$ ,  $h_2 = (h_{2\alpha}, h_{2\beta})$  – элемент группы  $\text{Aut}(\mathbf{Y}_{\beta l})$  матрицы  $\mathbf{Y}_{\beta l}$ .

Исходя из определения симметрий (см. п. 1.6) для матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$  они находятся как подстановки  $g \in S_{\alpha\beta}$ , удовлетворяющие равенству

$$h = g \in S_{\alpha\beta} \stackrel{df}{\Leftrightarrow} \mathbf{G}_\alpha^{-1} \tilde{\mathbf{Y}}^{(l-1)} \mathbf{G}_\beta = \tilde{\mathbf{Y}}^{(l-1)}. \quad (4.20)$$

С целью повышения эффективности выявления симметрий, которое обычно сводится к сокращению числа проверочных подстановок, целесообразно к матрице  $\tilde{\mathbf{Y}}^{(l-1)}$  применять рекомендации из п. 1.6. Кроме того, количество таких подстановок при выявлении симметрий матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$  удастся дополнительно сократить, если учесть ряд специфических особенностей рассматриваемого приложения.

Например, при определении групп автоморфизмов матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$  не допускается перестановка столбцов этих матриц, отвечающих  $\nu$ -му (вход-

ному) и  $w$ -му (выходному) узлам схемы, ни между собой, ни с другими столбцами. Эту особенность постановки задачи выявления симметрий в данном случае достаточно отразить в разбиении множества  $J$  номеров столбцов каждой из этих матриц на  $F_J$ -классы, считая указанные столбцы отдельными  $F_J$ -классами.

В работе [69] даны дополнительные рекомендации по повышению эффективности методов выявления симметрий для схем с ОУ, применимые к матрице  $\tilde{\mathbf{Y}}^{(l-1)}$ . В ней эта задача сведена к менее трудоемкому выявлению тех автоморфизмов подсхем схемы с ОУ, которые согласованы между собой на множестве номеров общих для этих подсхем узлов схемы. Выделению таких подсхем соответствует разбиение матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$  на блоки (см. пп. 4.5, 4.9).

Симметрии матриц  $\mathbf{Y}_{I_l}$ ,  $\mathbf{Y}_{\beta_l}$  легко определить как подстановки на множествах номеров строк и столбцов, меняющих местами их однотипные диагональные элементы.

Информация о симметриях матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$  позволяет на каждом шаге построения искомых схем находить порождающие подстановки  $g_{\mathfrak{g}} = (g_{\alpha_{\mathfrak{g}}}, g_{\beta_{\mathfrak{g}}})$ , которые эквивалентны уже примененным ранее, с целью их отсеечения. Для краткости будем называть их недопустимыми.

Рассмотрим вычисление недопустимых подстановок  $g_{\mathfrak{g}} = (g_{\alpha_{\mathfrak{g}}}, g_{\beta_{\mathfrak{g}}})$  на этапах перехода от матрицы  $\mathbf{Y}_{I_l}$  к матрице  $\mathbf{Y}_{\beta_l}$  и от матриц  $\mathbf{Y}_{\beta_l}$  к матрицам  $\mathbf{Y}_l$ .

Общий подход к определению подстановок  $g_{\mathfrak{g}} = (g_{\alpha_{\mathfrak{g}}}, g_{\beta_{\mathfrak{g}}})$  следует из механизма образования изоморфных систем, рассмотренного в п. 1.4. Действие автоморфизмов на стадии построения вариантов матрицы  $\mathbf{Y}_l$  из матрицы  $\mathbf{Y}_{I_l}$  сводится к тому, что появляется возможность одну и ту же матрицу  $\mathbf{Y}_l$  получить двумя путями:

действием порождающей подстановки  $g = (g_{\alpha}, g_{\beta})$ ,

$$\mathbf{Y}_l = \mathbf{G}_{\alpha}^{-1} \mathbf{G}_{\beta}^{-1} \mathbf{Y}_{I_l} \mathbf{G}_{\beta}, \quad (4.21)$$

и действием подстановки  $g_{\mathfrak{g}} = (g_{\alpha_{\mathfrak{g}}}, g_{\beta_{\mathfrak{g}}})$  вместе с действием автоморфизмов матриц  $\mathbf{Y}_{I_l}$ ,  $\mathbf{Y}_{\beta_l}$ ,  $\tilde{\mathbf{Y}}^{(l-1)}$ .

На этапе перехода от матрицы от матрицы  $\mathbf{Y}_{I_l}$  к матрице  $\mathbf{Y}_l$  оба указанных пути представим как

$$\mathbf{Y}_l = \mathbf{G}_\beta^{-1} \mathbf{Y}_{ll} \mathbf{G}_\beta, \quad (4.22)$$

и

$$\mathbf{Y}_l = \mathbf{H}_\alpha^{-1} \mathbf{G}_{\beta\alpha}^{-1} \mathbf{H}_{l\beta}^{-1} \mathbf{Y}_{ll} \mathbf{H}_{l\beta} \mathbf{G}_{\beta\alpha} \mathbf{H}_\beta \quad (4.23)$$

соответственно.

Приравняв правые части (4.22), (4.23), найдем эквивалентную матричную подстановку

$$\mathbf{G}_{\beta\alpha} = \mathbf{H}_{l\beta}^{-1} \mathbf{G}_\beta \mathbf{H}_\beta^{-1} = \mathbf{H}_{l\beta}^{-1} \mathbf{G}_\beta \mathbf{H}_\alpha^{-1}. \quad (4.24)$$

Из (4.24) следует, что такая эквивалентная подстановка возможна при наличии в группе  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)})$  автоморфизмов  $h = (h_\alpha, h_\beta)$  с  $h_\alpha = h_\beta$ , обеспечивающих диагональность матрицы  $\mathbf{Y}_{ll}$  при переходе к ней от диагональной же матрицы  $\mathbf{Y}_{ll}$ .

От эквивалентной матричной подстановки  $\mathbf{G}_{\beta\alpha}$  перейдем к эквивалентной подстановке  $g_{\beta\alpha}$ :

$$g_{\beta\alpha} = h_{l\beta}^{-1} g_\beta h_\beta^{-1}. \quad (4.25)$$

В алгоритме 4.3 уже предусмотрено отсечение изоморфных матриц  $\mathbf{Y}_{ll}$ , обусловленных собственными симметриями матрицы  $\mathbf{Y}_{ll}$ . Поэтому формулу (4.25) следует упростить при условии, что она применяется к множеству порождающих подстановок  $g$ , из которого предварительно удалены с помощью этого алгоритма подстановки, эквивалентные использованным ранее

$$g_{\beta\alpha} = g_\beta h_\beta^{-1}. \quad (4.26)$$

Алгоритм 4.3 основан на сведении задачи генерирования диагональных матриц  $\mathbf{Y}_{ll}$  к генерированию неизоморфных векторов с повторяющимися координатами (см. п. 4.6). Полезно такой же подход применить к отсечению недопустимых подстановок  $g_{\beta\alpha}$ , обусловленных, как следует из (4.26), симметриями матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$ . Эта задача наиболее просто решается в соответствии с утверждением 2.1, если множество подстановок  $h_{l\beta} = h_\beta$  образует симметрическую группу.

К допустимым порождающим подстановкам на этапе перехода от каждого варианта матрицы  $\mathbf{Y}_{\beta l}$  к вариантам матрицы  $\mathbf{Y}_l$  относятся только четные или только нечетные в зависимости от требуемого знака соответствующего полиномиального коэффициента (см. п. 4.7). Каждая из этих допустимых подстановок также может быть получена двумя путями:

действием подстановки  $g_\alpha$ ,

$$\mathbf{Y}_l = \mathbf{G}_\alpha^{-1} \mathbf{Y}_{\beta l}, \quad (4.27)$$

и действием эквивалентной подстановки  $g_{\alpha\beta}$  вместе с действием автоморфизмов  $h_2 = (h_{2a}, h_{2b}) \in \text{Aut}(\mathbf{Y}_{\beta l})$ ,  $h = (h_\alpha, h_\beta) \in \text{Aut}(\tilde{\mathbf{Y}}^{(l-1)})$ ,

$$\mathbf{Y}_l = \mathbf{H}_\alpha^{-1} \mathbf{G}_{\alpha\beta}^{-1} \mathbf{H}_{2\beta}^{-1} \mathbf{Y}_{\beta l} \mathbf{H}_{2\beta} \mathbf{H}_\beta. \quad (4.28)$$

Приравняв правые части (4.27), (4.28), находим эквивалентные матричные подстановки:

$$\mathbf{G}_{\alpha\beta} = \mathbf{H}_{2\beta}^{-1} \mathbf{G}_\alpha \mathbf{H}_\alpha^{-1}, \quad (4.29)$$

$$\mathbf{H}_{2\beta} \mathbf{H}_\beta = e. \quad (4.30)$$

Из (4.30) следует, что в рассматриваемом случае появление эквивалентных порождающих подстановок возможно только при условии  $\mathbf{H}_{2\beta} = \mathbf{H}_\beta^{-1}$ , и вычисляются они по формуле (4.29).

От эквивалентных матричных подстановок (4.29), (4.30) перейдем к эквивалентным подстановкам  $g_\beta = (g_{\alpha\beta}, g_{\beta\beta})$ :

$$g_{\alpha\beta} = h_{2\beta}^{-1} g_\alpha h_\alpha^{-1}, \quad (4.31)$$

$$h_{2\beta} h_\alpha = e. \quad (4.32)$$

Число эквивалентных подстановок  $g_{\alpha\beta}, g_{\beta\beta}$ , подлежащих исключению на  $l$ -м шаге, тем больше, чем богаче группа автоморфизмов матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$ . Например, в группе автоморфизмов  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = H_\alpha \otimes H_\beta$  матрицы  $\tilde{\mathbf{Y}}^{(l-1)} = \tilde{\mathbf{Y}}^{(0)}$  подгруппа  $H_\alpha$  совпадает с симметрической группой  $S_\alpha$ . Следовательно, при  $l = 1$  все подстановки  $g_\alpha \in S_\alpha$  эквивалентны между собой, поэтому в процедуре синтеза достаточно использовать единственную порождающую подстановку  $g_\alpha$ , например тождественную. Отно-

сительно подстановок  $g_\beta$  при  $l=1$  необходимо отметить следующее. Если матрица  $Y_{ll}$  содержит в качестве диагональных только однотипные элементы, рассматривается только  $C(\rho, \rho) = 1$  вариант порождающих подстановок  $g_\beta$ . Если же диагональные элементы матрицы  $Y_{ll}$  разнотипны, необходимо использовать  $C(\rho, \eta) \neq 1$  порождающих подстановок  $g_\beta$ . Однако равенству (4.28) при  $l = 1$  в этом случае удовлетворяют лишь две допустимые порождающие подстановки  $g_\beta$ . Они отличаются между собой только тем, что приводят к различным по типу проводимостям в  $v$ -м ( $w$ -м) столбце матрицы  $Y_{ll}$ . Таким образом, при  $l = 1$  в дереве вариантов  $T_2$  на первом ярусе, отражающем количество вариантов матрицы  $Y_{ll}$ , в первом случае единственная вершина, а во втором – две вершины. На втором ярусе  $T_2$  вследствие эквивалентности всех подстановок  $g_\alpha \in S_\alpha$  между собой при  $l = 1$  разветвления отсутствуют.

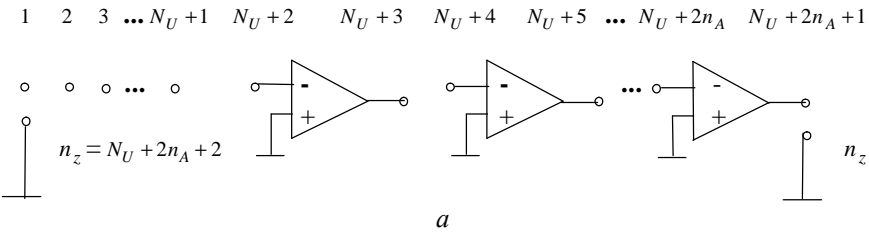
Таким образом, рассмотренное применение информации о симметриях матриц  $Y_{ll}$ ,  $Y_{\beta l}$ , и  $\tilde{Y}^{(l-1)}$  позволяет резко сократить просматриваемое множество матриц  $\tilde{Y}$ . Но оно не исключает необходимости сравнения каждого нового элемента этого множества с неизоморфными матрицами  $\tilde{Y}$ , полученными ранее. Дело в том, что к изоморфным и даже повторяющимся матрицам  $\tilde{Y}$  могут привести операции склеивания (4.19) при замене параллельных однотипных ветвей одной ветвью и пополнения при переходе от  $Y_l$  к  $\tilde{Y}_l'$ . Для уменьшения трудоемкости такого сравнения применимы методы разбиения множеств строк и столбцов матрицы  $\tilde{Y}$  из п.1.6. В работе [65] предложен эффективный способ сравнения матриц  $\tilde{Y}$ , основанный на сопоставлении численных характеристик их строк и столбцов.

#### 4.9. Алгоритм синтеза схем с операционными усилителями

Зададимся последовательностью реализации полиномиальных коэффициентов  $a_i, b_j \in \mathbf{D}$ . Тем самым каждому ненулевому коэффициенту заданной функции  $T(p)$  поставим в соответствие порядковый номер  $l \in \{1, 2, \dots, L\}$ , где  $L = |\mathbf{D}|$ . Целесообразно начинать с реализации коэффициентов числителя  $T(p)$ , если в нем число нулевых коэффициентов больше, чем в

знаменателе. В этом случае большая часть функционально непригодных структур отсекается на верхних ярусах дерева вариантов  $T_4$ .

Предлагаемый алгоритм синтеза схем с операционными усилителями основан на рассмотренном в п. 4.7 склеивании матриц  $Y_l$ ,  $l \in \{1, 2, \dots, L\}$ , реализующих каждый ненулевой полиномиальный коэффициент заданной функции  $T(p)$ . Кроме спектра степеней заданной функции, отраженного в векторе  $\mathbf{D}$ , исходными данными для работы этого алгоритма являются число  $n_z$  узлов искомой схемы, число  $n_A$  ОУ, число  $n_C$  емкостных двухполюсников и число  $n_G$  резистивных двухполюсников. Необходимо также задаться способом включения и расположением в схеме операционных усилителей, как было рекомендовано в п. 4.5. Тем самым определится связь между координатами ненулевых элементов матриц  $Y_l$ ,  $\tilde{Y}^{(l-1)}$ . Например, при трехполюсном включении ОУ выберем их расположение в схеме, показанное на рис. 4.5, а, а при дифференциальном включении полюсов ОУ

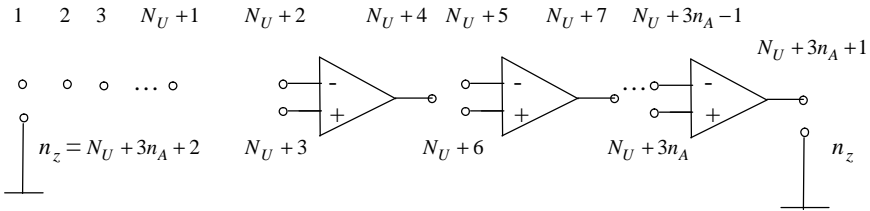


	1	2	3	...	$N_U+1$	$N_U+3$	$N_U+5$	...	$N_U+2n_A+1$
2		+		...				...	
3			+	...				...	
...	...	...	...	...	...	...	...	...	...
$N_U+1$				...	+			...	
$N_U+2$				...				...	
$N_U+4$				...				...	
...	...	...	...	...	...	...	...	...	...
$N_U+2n_A$				...				...	

б

Рис 4.5

– на рис. 4.6, *a*. В этих случаях связь между матрицами  $\tilde{\mathbf{Y}}^{(l)}$ ,  $\mathbf{Y}_l$  наглядно представлена на рис. 4.5, *б* и 4.6, *б* соответственно. Пусть  $N_U$  – количество внутренних узлов схемы, к которым подключены только пассивные двухполосники. Выберем, как показано на рис. 4.5, *a*, 4.6, *a*, нумерацию узлов схемы ОУ слева направо. Входному узлу схемы присвоим номер 1, выходному, совпадающему с выходом  $n_A$ -го ОУ, – номер  $(n_z - 1)$ , общему – номер  $n_z$ . Узлу схемы рис. 4.5, *a*, инцидентному входному полюсу  $i$ -го ОУ, присвоим номер  $2i + N_U$ , инцидентному выходному полюсу этого ОУ – номер  $2i + N_U + 1$ . Узлу схемы на рис. 4.6, *a*, инцидентному



*a*

	1	2	3	...	$N_U + 1$	$N_U + 2 +$ $+ N_U + 3$	$N_U + 4$	...	$N_U + 3n_A - 1 +$ $+ N_U + 3n_A$	$N_U +$ $+ 3n_A + 1$
2		+		...				...		
3			+	...				...		
...	...	...	...	...	...	...	...	...	...	...
$N_U + 1$				...	+			...		
$N_U + 2$				...		+		...		
$N_U + 3$				...		+		...		
...	...	...	...	...	...	...	...	...	...	...
$N_U +$ $+ 3n_A - 1$				...				...	+	
$N_U +$ $+ 3n_A$				...				...	+	

*б*

Рис 4.6



инвертирующему полюсу  $i$ -го ОУ, присвоим номер  $3i + N_U - 1$ , входному неинвертирующему – номер  $3i + N_U$ , выходному – номер  $3i + N_U + 1$ .

Алгоритм, приведенный далее, инвариантен к способу нумерации узлов и элементов схемы. От выбора нумерации зависит только переход от матрицы  $\tilde{\mathbf{Y}}^{(l)}$  к матрице  $\mathbf{Y}$  и обратно на шагах 1, 5, 10 алгоритма.

Алгоритм 4.4.

Шаг 1. Положить  $l = 0$ . Построить матрицу  $\tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(0)}$ . Выделить в ней 1-й и  $(n_z - 1)$ -й столбцы как отдельные классы эквивалентности.

Шаг 2. Положить  $l := l + 1$ . Если  $l > L$ , перейти к шагу 8.

Шаг 3. Определить группу автоморфизмов матрицы  $\tilde{\mathbf{Y}}^{(l-1)}$ .

Шаг 4. Построить по алгоритму 4.3 очередной вариант матрицы  $\mathbf{Y}_l$ , исключая подстановки  $g_{\beta_3}$  (4.24)-(4.26),  $g_{\alpha_3}$  (4.29)-(4.32), а также подстановки  $g_{\alpha}$  с недопустимой четностью. Если варианты  $\tilde{\mathbf{Y}}_l$  исчерпаны, идти к шагу 12. Перейти от матрицы  $\mathbf{Y}_l$  к первому из возможных вариантов матрицы  $\tilde{\mathbf{Y}}_l$  с учетом (4.17), (4.18) при синтезе схем 2-го или 3-го класса.

Шаг 5. Построить матрицу  $\tilde{\mathbf{Y}}^{(l)}$  по (4.19).

Шаг 6. Проверить, не превышает ли заданной величины количество элементов какого-либо типа в  $\tilde{\mathbf{Y}}^{(l)}$ . Если превышает, перейти к шагу 11.

Шаг 7. Проверить, удовлетворяет ли набор трансверселей матрицы  $\tilde{\mathbf{Y}}^{(l)}$  требованиям, отраженным в векторах  $\mathbf{D}$ ,  $\mathbf{D}_0$ ,  $\mathbf{D}_s$ . Если удовлетворяют, перейти к шагу 2, иначе – к шагу 11.

Шаг 8. Проверить матрицу  $\tilde{\mathbf{Y}}^{(l)}$  на наличие в ней строк или столбцов, кроме 1-го и  $(n_z - 1)$ -го, с единственным ненулевым элементом. Если такие есть, перейти к шагу 11.

Шаг 9. Проверить матрицу  $\tilde{\mathbf{Y}}^{(l)}$  на идентичность или эквивалентность построенным ранее. Если идентична или эквивалентна, перейти к шагу 11.

Шаг 10. От матрицы  $\tilde{\mathbf{Y}}^{(l)}$  перейти к матрице  $\mathbf{Y}$ , к синтезированной схеме и к аналитическим выражениям полиномиальных коэффициентов ее передаточной функции  $T(p)$ .

Шаг 11. Если не исчерпаны варианты пополнения  $\mathbf{Y}_l$ , выбрать очередной вариант и перейти к шагу 5, при прочих условиях – к шагу 4.

Шаг 12. Положить  $l := l - 1$ . Если  $l \geq 1$ , перейти к шагу 4.

Шаг 13. Конец.

На шаге 10 алгоритма для построения аналитических выражений поли-

номиальных коэффициентов можно воспользоваться одним из известных средств анализа в буквенно-цифровом виде, например, предложенным в работе [70].

В связи с большой трудоемкостью задачи структурного синтеза схем проиллюстрируем применение алгоритма 4.4 только фрагментами его работы для конкретного примера.

**Пример 4.2.** Пусть требуется построить каталог фазовых контуров второго порядка, содержащих восемь узлов, три ОУ в трехполюсном включении, три емкости и семь резисторов. Допускается только одно слагаемое в каждом полиномиальном коэффициенте его передаточной функции:

$$T(p) = \frac{a_2 p^2 + a_1 p + a_0}{b_2 p^2 + b_1 p + b_0},$$

где  $a_2 = b_2$ ,  $a_1 = -b_1$ ,  $a_0 = b_0$ .

Выберем следующий порядок реализации полиномиальных коэффициентов:

$l$	1	2	3	4	5	6
$a_i, b_i$	$a_2$	$a_1$	$a_0$	$b_2$	$b_1$	$b_0$

Тогда  $\mathbf{D} = (a_2', a_1', a_0', b_2', b_1', b_0') = (1, -1, 1, 1, 1, 1)$ ,  $\mathbf{D}_0 = (a_3', b_3') = (0, 0)$ ,  $\mathbf{D}_s = (a_{s2}, a_{s1}, a_{s0}, b_{s2}, b_{s1}, b_{s0}) = (1, 1, 1, 1, 1, 1)$ . Остановимся на нумерации узлов и элементов искомой схемы, показанной на рис. 4.5, а.

В приведенном далее фрагменте решения поставленной задачи с помощью алгоритма 4.4 индексы емкостей и сопротивлений – это координаты соответствующих элементов в матрице  $\mathbf{Y}$  и одновременно номера узлов схемы, к которым подключены соответствующие двухполюсники.

Шаг 1.  $l = 0$  (в приведенной ниже матрице номера узлов искомой схемы указаны в скобках),

$$\tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(0)} = \begin{array}{c} \begin{array}{cccc} (1) & (3) & (5) & (7) \\ 1 & 2 & 3 & 4 \end{array} \\ \left\| \begin{array}{ccc|c} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right\| \begin{array}{l} 1 \text{ (2)} \\ 2 \text{ (4)} \\ 3 \text{ (6)} \end{array} \end{array}$$

Шаг 2.  $l = 1 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(0)}) = H_\alpha \otimes H_\beta$ , где  $H_\alpha$  – симметрическая группа подстановок на множестве  $\{1, 2, 3\}$ ,  $H_\beta$  – на множестве  $\{2, 3\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta : e, (3, 1, 2)$  ( $g_\beta^{-1} : e, (2, 3, 1)$ ). Допустимые подстановки  $g_\alpha : e$ .

$$\mathbf{Y}_l = \mathbf{Y}_1 = \left\| \begin{array}{|c|c|c|} \hline -pC_{21} & 0 & 0 \\ \hline 0 & -pC_{43} & 0 \\ \hline 0 & 0 & -G_{65} \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(1)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} & 0 & 0 & 0 \\ \hline 0 & -pC_{43} & 0 & 0 \\ \hline 0 & 0 & -G_{65} & 0 \\ \hline \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 2 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(1)}) = \{e\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta : e, (2, 1, 3), (2, 3, 1)$ . Допустимые подстановки  $g_\alpha : (1, 3, 2), (2, 1, 3), (3, 2, 1)$ .

$$\mathbf{Y}_2 = \left\| \begin{array}{|c|c|c|} \hline -pC_{21} & 0 & 0 \\ \hline 0 & 0 & -G_{45} \\ \hline 0 & -G_{63} & 0 \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(2)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} & 0 & 0 & 0 \\ \hline 0 & -pC_{43} & -G_{45} & 0 \\ \hline 0 & -G_{63} & -G_{65} & 0 \\ \hline \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 3 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(2)}) = \{e\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta : e$ . Допустимые подстановки  $g_\alpha : e, (2, 3, 1), (3, 1, 2)$ .

$$\mathbf{Y}_3 = \left\| \begin{array}{|c|c|c|} \hline -G_{21} & 0 & 0 \\ \hline 0 & -G_{43} & 0 \\ \hline 0 & 0 & -G_{65} \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(3)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} - G_{21} & 0 & 0 & 0 \\ \hline 0 & -pC_{43} - G_{43} & -G_{45} & 0 \\ \hline & -G_{63} & -G_{65} & 0 \\ \hline \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Не удовлетворяет, так как  $a_{s1} \neq 1$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta : e$ . Допустимые подстановки  $g_\alpha : (2, 3, 1), (3, 1, 2)$ .

$$\mathbf{Y}_l = \mathbf{Y}_3 = \left\| \begin{array}{|c|c|c|} \hline 0 & -G_{23} & 0 \\ \hline 0 & 0 & -G_{45} \\ \hline -G_{61} & 0 & 0 \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(3)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} & -G_{23} & 0 & 0 \\ \hline 0 & -pC_{43} & -G_{45} & 0 \\ \hline -G_{61} & -G_{63} & -G_{65} & 0 \\ \hline \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 4 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(3)}) = \{e\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta: e, (1, 3, 2), (3, 1, 2)$ . Допускаются любые подстановки  $g_\alpha$ .

$$\mathbf{Y}_4 = \left\| \begin{array}{|c|c|c|} \hline -pC_{23} & 0 & 0 \\ \hline 0 & -pC_{45} & 0 \\ \hline 0 & 0 & -G_{67} \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(4)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} & -pC_{23} & 0 & 0 \\ \hline 0 & -pC_{43} & -pC_{45} - G_{45} & 0 \\ \hline -G_{61} & -G_{63} & -G_{65} & -G_{67} \\ \hline \end{array} \right\| .$$

Шаг 6. Число емкостей превышает  $n_C$ .

Шаги 4, 5.

$$\mathbf{Y}_4 = \left\| \begin{array}{|c|c|c|} \hline -pC_{23} & 0 & 0 \\ \hline 0 & 0 & -G_{47} \\ \hline 0 & -pC_{65} & 0 \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(4)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} & -pC_{23} - G_{23} & 0 & 0 \\ \hline 0 & -pC_{43} & -G_{45} & -G_{47} \\ \hline -G_{61} & -G_{63} & -pC_{65} - G_{65} & 0 \\ \hline \end{array} \right\| .$$

Шаг 6. Число емкостей превышает  $n_C$ .

Шаги 4, 5.

$$\mathbf{Y}_l = \mathbf{Y}_4 = \left\| \begin{array}{|c|c|c|} \hline 0 & -pC_{25} & 0 \\ \hline -pC_{43} & 0 & 0 \\ \hline 0 & 0 & -G_{67} \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(4)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} & -G_{23} & -pC_{25} & 0 \\ \hline 0 & -pC_{43} & -G_{45} & 0 \\ \hline -G_{61} & -G_{63} & -G_{65} & -G_{67} \\ \hline \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Не удовлетворяет, так как  $a_{s_2} \neq 1$ .

Дальше приведем фрагмент работы алгоритма 4.4, содержащий появление первой из искоемых схем.

Шаги 4, 5. Допустимые подстановки  $g_\beta: (1, 3, 2), (3, 1, 2)$ . Допускаются любые подстановки  $g_\alpha$ .

$$\mathbf{Y}_l = \mathbf{Y}_4 = \left\| \begin{array}{|c|c|c|} \hline 0 & 0 & -pC_{27} \\ \hline -pC_{43} & 0 & 0 \\ \hline 0 & -G_{65} & 0 \\ \hline \end{array} \right\| ; \tilde{\mathbf{Y}}^{(4)} = \left\| \begin{array}{|c|c|c|c|} \hline -pC_{21} & 0 & -G_{25} & -pC_{27} \\ \hline -G_{41} & -pC_{43} & 0 & 0 \\ \hline -G_{61} & -G_{63} & -G_{65} & 0 \\ \hline \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 5 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(4)}) = \{e\}$ .

Шаги 4,5. Допустимые подстановки  $g_\beta: e, (2, 1, 3), (2, 3, 1)$ . Допустимые подстановки  $g_\alpha: e, (2, 3, 1), (3, 1, 2)$ .

$$Y_l = Y_5 = \left\| \begin{array}{ccc} -pC_{23} & 0 & 0 \\ 0 & -G_{45} & 0 \\ 0 & 0 & -G_{67} \end{array} \right\| ; \tilde{Y}^{(5)} = \left\| \begin{array}{ccc} -pC_{21} & -pC_{23} & -G_{25} & -pC_{27} \\ -G_{41} & -pC_{43} & -G_{45} & 0 \\ -G_{61} & -G_{63} & -G_{65} & -G_{67} \end{array} \right\| .$$

Шаг 6. Число емкостей превышает  $n_c$ .

Шаги 4, 5. Допустимые подстановки  $g_\alpha: (2, 3, 1), (3, 1, 2)$ .

$$Y_5 = \left\| \begin{array}{ccc} 0 & -G_{25} & 0 \\ 0 & 0 & -G_{47} \\ -pC_{63} & 0 & 0 \end{array} \right\| ; \tilde{Y}^{(5)} = \left\| \begin{array}{ccc} -pC_{21} & 0 & -G_{25} & -pC_{27} \\ -G_{41} & -pC_{43} & 0 & -G_{47} \\ -G_{61} & -pC_{63} - G_{63} & -G_{65} & 0 \end{array} \right\| .$$

Шаг 6. Число емкостей превышает  $n_c$ .

Шаги 4, 5. Допустимые подстановки  $g_\alpha: (3, 1, 2)$ .

$$Y_5 = \left\| \begin{array}{ccc} 0 & 0 & -G_{27} \\ -pC_{43} & 0 & 0 \\ 0 & -G_{65} & 0 \end{array} \right\| ; \tilde{Y}^{(5)} = \left\| \begin{array}{ccc} -pC_{21} & 0 & -G_{25} & -pC_{27} - G_{27} \\ -G_{41} & -pC_{43} & 0 & 0 \\ -G_{61} & -G_{63} & -G_{65} & 0 \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 6 = L$ .

Шаг 3.  $\text{Aut}(\tilde{Y}^{(l-1)}) = \text{Aut}(\tilde{Y}^{(5)}) = \{e\}$ .

Шаги 4,5. Допустимые подстановки  $g_\beta: e$ . Допустимые подстановки  $g_\alpha: e, (2, 3, 1), (3, 1, 2)$ .

$$Y_6 = \left\| \begin{array}{ccc} -G_{23} & 0 & 0 \\ 0 & -G_{45} & 0 \\ 0 & 0 & -G_{67} \end{array} \right\| ; \tilde{Y}^{(6)} = \left\| \begin{array}{ccc} -pC_{21} & -G_{23} & -G_{25} & -pC_{27} - G_{27} \\ -G_{41} & -pC_{43} & -G_{45} & 0 \\ -G_{61} & -G_{63} & -G_{65} & -G_{67} \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Не удовлетворяет,  $a_{s1} \neq 1$ .

Шаги 4, 5. Допустимые подстановки  $g_\alpha: (2, 3, 1), (3, 1, 2)$ .

$$Y_6 = \left\| \begin{array}{ccc} 0 & -G_{25} & 0 \\ 0 & 0 & -G_{47} \\ -G_{63} & 0 & 0 \end{array} \right\| ; \tilde{Y}^{(6)} = \left\| \begin{array}{ccc} -pC_{21} & 0 & -G_{25} & -pC_{27} - G_{27} \\ -G_{41} & -pC_{43} & 0 & -G_{47} \\ -G_{61} & -G_{63} & -G_{65} & 0 \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 7 > L = 6$ .

Шаг 8. Нет строк или столбцов с единственным ненулевым элементом.

Шаг 9. Нет идентичных или эквивалентных схем..

Шаг 10. Схема, соответствующая матрице  $\tilde{\mathbf{Y}}^{(6)}$ , представлена на рис. 4.1, а. Для нее функция  $T(p)$  приведена в примере 4.1.

Рассмотрим еще один фрагмент работы программы.

Шаг 2.  $l = 1 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(0)}) = H_\alpha \otimes H_\beta$ , где  $H_\alpha$  – симметрическая группа подстановок на множестве  $\{1, 2, 3\}$ ,  $H_\beta$  – на множестве  $\{2, 3\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta : (3, 1, 2)$ . Допустимые подстановки

$g_\alpha : e$ .

$$\mathbf{Y}_l = \mathbf{Y}_1 = \left\| \begin{array}{ccc} -G_{21} & 0 & 0 \\ 0 & -pC_{43} & 0 \\ 0 & 0 & -pC_{65} \end{array} \right\| ; \quad \tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(1)} = \left\| \begin{array}{ccc} -G_{21} & 0 & 0 \\ 0 & -pC_{43} & 0 \\ 0 & 0 & -pC_{65} \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 2 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(1)}) = \{e, ((2, 3), (2, 3))\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta : e, (2, 1, 3)$ . Допустимые подстановки

$g_\alpha : (1, 3, 2), (2, 1, 3)$ .

$$\mathbf{Y}_l = \mathbf{Y}_2 = \left\| \begin{array}{ccc} -pC_{21} & 0 & 0 \\ 0 & 0 & -G_{45} \\ 0 & -G_{63} & 0 \end{array} \right\| ; \quad \tilde{\mathbf{Y}}^{(2)} = \left\| \begin{array}{ccc} -G_{21} - pC_{21} & 0 & 0 \\ 0 & -pC_{43} & -G_{45} \\ 0 & -G_{63} & -pG_{65} \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7.  $a_3 \neq 0$ .

Шаги 4, 5. Допустимые подстановки  $g_\alpha : (2, 1, 3)$ .

$$\mathbf{Y}_l = \mathbf{Y}_2 = \left\| \begin{array}{ccc} 0 & -G_{23} & 0 \\ -pC_{41} & 0 & 0 \\ 0 & 0 & -G_{65} \end{array} \right\| ; \quad \tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(2)} = \left\| \begin{array}{ccc} -G_{21} & -G_{23} & 0 \\ -pC_{41} & -pC_{43} & 0 \\ 0 & 0 & -pG_{65} - G_{65} \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7.  $a_{s2} \neq 1$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta : (2, 1, 3)$ . Допустимые подстановки

$g_\alpha : (1, 3, 2), (2, 1, 3)$ .

$$\mathbf{Y}_l = \mathbf{Y}_2 = \left\| \begin{array}{ccc|c} -G_{21} & 0 & 0 & 0 \\ 0 & 0 & -G_{45} & 0 \\ 0 & -pC_{63} & 0 & 0 \end{array} \right\| ; \tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(2)} = \left\| \begin{array}{ccc|c} -G_{21} & 0 & 0 & 0 \\ 0 & -pC_{43} & -G_{45} & 0 \\ 0 & -pG_{63} & -pG_{65} & 0 \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 3 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(2)}) = \{e\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta: e$ . Допустимые подстановки  $g_\alpha: e, (2, 3, 1), (3, 1, 2)$ .

$$\mathbf{Y}_3 = \left\| \begin{array}{ccc|c} -G_{21} & 0 & 0 & 0 \\ 0 & -G_{43} & 0 & 0 \\ 0 & 0 & -G_{65} & 0 \end{array} \right\| ; \tilde{\mathbf{Y}}^{(3)} = \left\| \begin{array}{ccc|c} -G_{21} & 0 & 0 & 0 \\ 0 & -pC_{43} - G_{43} & -G_{45} & 0 \\ 0 & -pG_{63} & -pG_{65} - G_{45} & 0 \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7.  $a_{s1} \neq 1$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta: e$ . Допустимые подстановки  $g_\alpha: (2, 3, 1), (3, 1, 2)$ .

$$\mathbf{Y}_l = \mathbf{Y}_3 = \left\| \begin{array}{ccc|c} 0 & -G_{23} & 0 & 0 \\ 0 & 0 & -G_{45} & 0 \\ -G_{61} & 0 & 0 & 0 \end{array} \right\| ; \tilde{\mathbf{Y}}^{(l)} = \tilde{\mathbf{Y}}^{(3)} = \left\| \begin{array}{ccc|c} -G_{21} & -G_{23} & 0 & 0 \\ 0 & -pC_{43} & -G_{45} & 0 \\ -G_{61} & -pG_{63} & -pG_{65} & 0 \end{array} \right\| .$$

Шаг 6. Не превышает.

Шаг 7. Удовлетворяет.

Шаг 2.  $l = 4 < L = 6$ .

Шаг 3.  $\text{Aut}(\tilde{\mathbf{Y}}^{(l-1)}) = \text{Aut}(\tilde{\mathbf{Y}}^{(3)}) = \{e\}$ .

Шаги 4, 5. Допустимые подстановки  $g_\beta: e, (1, 3, 2), (3, 1, 2)$ . Допускаются любые подстановки  $g_\alpha$ .

$$\mathbf{Y}_4 = \left\| \begin{array}{ccc|c} -pC_{23} & 0 & 0 & 0 \\ 0 & -pC_{45} & 0 & 0 \\ 0 & 0 & -G_{67} & 0 \end{array} \right\| ; \tilde{\mathbf{Y}}^{(4)} = \left\| \begin{array}{ccc|c} -G_{21} & -pC_{23} - G_{23} & 0 & 0 \\ 0 & -pC_{43} & -pC_{45} - G_{45} & 0 \\ -G_{61} & -pG_{63} & -pG_{65} & -G_{67} \end{array} \right\| .$$

Шаг 6. Число емкостей превышает  $n_C$ .

Читателю предоставляется убедиться самостоятельно в том, что, продолжая действовать в соответствии с алгоритмом 4.4, получаем еще одну схему (рис. 4.7, а). Передаточная функция этой схемы имеет вид

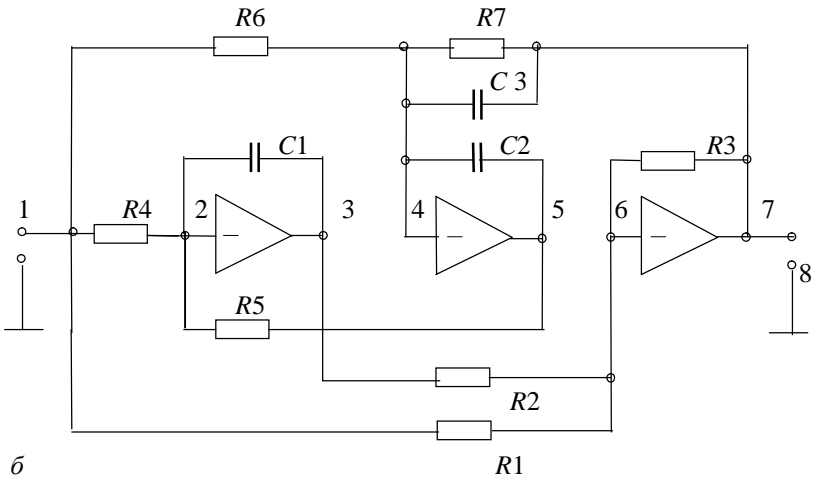
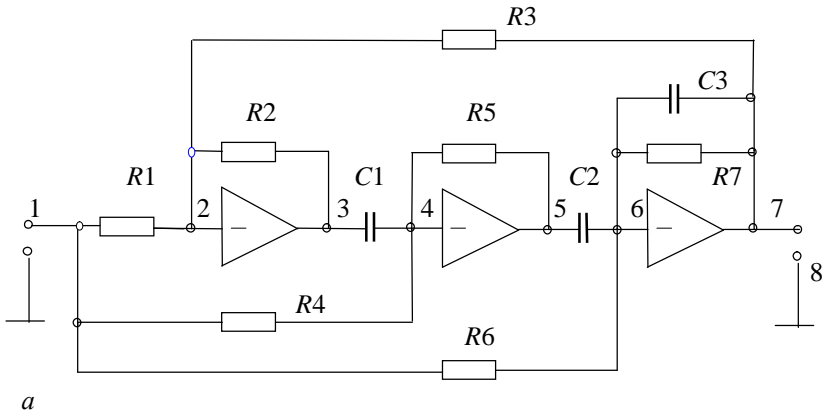


Рис 4.7



$$T(p) = - \frac{p^2 C_1 C_2 G_1 - p C_2 G_2 G_4 + G_2 G_5 G_6}{p^2 C_1 C_2 G_3 + p C_3 G_2 G_5 + G_2 G_5 G_7} = - \frac{G_1}{G_3} \cdot \frac{p^2 - \frac{G_2 G_4}{C_1 G_1} p + \frac{G_2 G_5 G_6}{C_1 C_2 G_1}}{p^2 + \frac{C_3 G_2 G_5}{C_1 C_2 G_3} p + \frac{G_2 G_5 G_7}{C_1 C_2 G_3}}.$$

В табл. 4.4 приведены данные о схеме на рис. 4.7, *a* (первая колонка табл. 4.4) и о схемах, полученных из нее посредством эквивалентных преобразований по алгоритму 4.2. Для каждого двухполосника указана пара номеров узлов  $i$  и  $j$ , к которым он подключен в схеме. Предполагается, что  $a = 1$ ,  $b = 7$ ,  $c = 8$ ,  $q(1) = 2$ ,  $r(1) = 3$ ,  $q(2) = 4$ ,  $r(2) = 5$ ,  $q(3) = 6$ ,  $r(3) = 7$ .

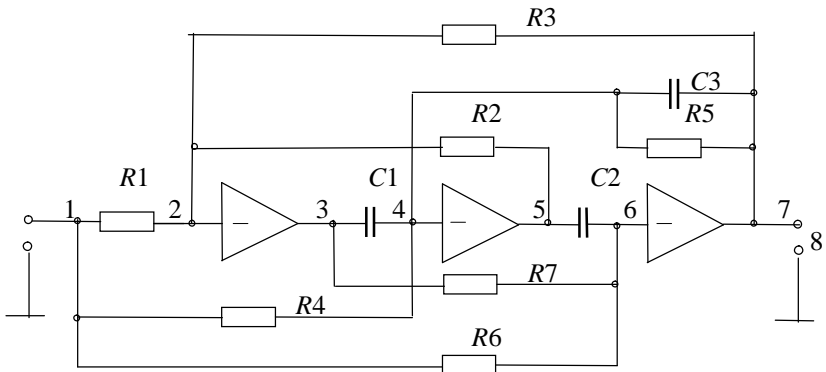
Т а б л и ц а 4.4

*Данные о схеме на рис. 4.7, a и эквивалентных ей схемах*

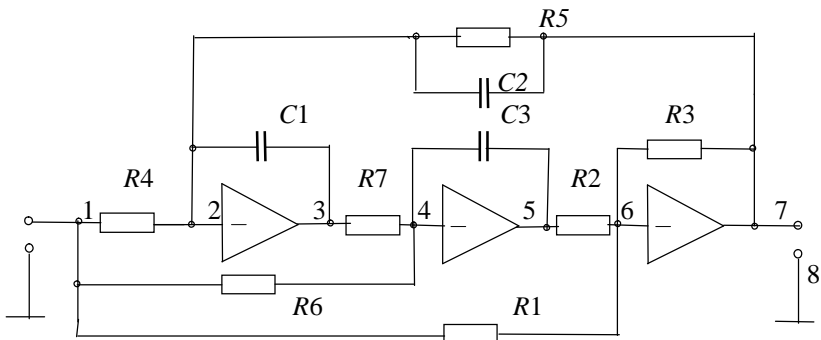
Элемент схемы	Варианты включения элементов					
	1	2	3	4	5	6
C1	3,4	3,6	2,3	2,3	3,6	3,4
C2	5,6	4,5	5,6	4,5	2,5	2,5
C3	6,7	4,7	6,7	4,7	2,7	2,7
R1	1,2	1,2	1,4	1,6	1,4	1,6
R2	2,3	2,3	3,4	3,6	3,4	3,6
R3	2,7	2,7	4,7	6,7	4,7	6,7
R4	1,4	1,6	1,2	1,2	1,6	1,4
R5	4,5	5,6	2,5	2,5	5,6	4,5
R6	1,6	1,4	1,6	1,4	1,2	1,2
R7	6,7	4,7	6,7	4,7	2,7	2,7

Среди схем табл. 4.4 устойчивыми являются схемы с данными, приведенными в первой и четвертой колонках. Они показаны на рис. 4.7, *a* и 4.7, *b* соответственно. Схему с данными из пятой колонки можно считать условно устойчивой, так как ее устойчивость достигается подбором параметров ее элементов. Остальные схемы таблицы 4.4 неустойчивы.

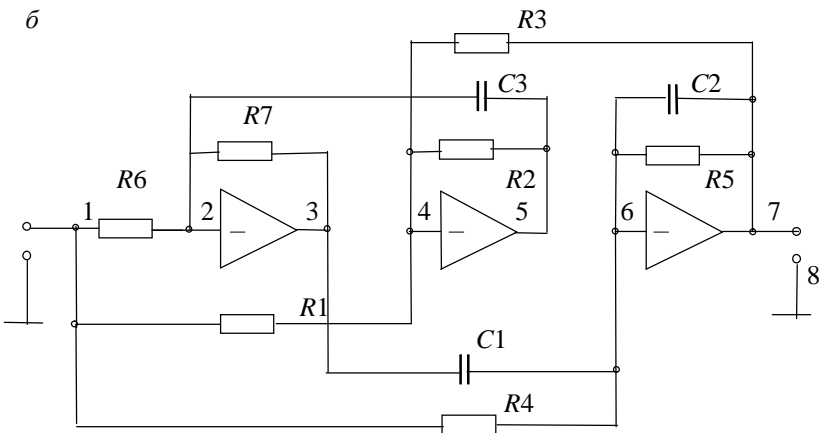
Данные о схемах в табл. 4.5 получены из схем табл. 4.4 применением к их матрицам  $\tilde{Y}$  перестановки второго и третьего столбцов. Сравнивая данные в табл. 4.4, 4.5, читатель может убедиться, что такое преобразование приводит к изоморфным схемам. Схемы на рис. 4.8 изоморфны устойчивым схемам из табл. 4.4: схема рис. 4.8, *a* изоморфна условно устойчивой схеме из табл. 4.4; схема рис. 4.8, *b* – схеме рис. 4.7, *b*, схема рис. 4.8, *в* – схеме рис. 4.7, *a*.



*a*



*б*



*в*

Рис 4.8

На рис. 4.8, б известная [71] схема фазового звена показана в более привычном начертании по сравнению с изображенной на рис. 4.7, б. Сопоставив аналитические выражения этой схемы и синтезированных схем рис. 4.1 (см. пример 4.1), нетрудно убедиться, что последние обладают следующим достоинством по сравнению с известной: в каждой из них возможна коррекция ФЧХ (регулировка частоты, на которой ФЧХ принимает значение  $180^\circ$ ) без изменения АЧХ одним резистором ( $R6$ ). Эти схемы обеспечивают неитерационную настройку частоты полюса ( $R7$ ), добротности полюса ( $R4$ ), частоты нуля ( $R5$ ), добротности нуля ( $R2$ ).

Т а б л и ц а 4.5

*Данные о схемах, изоморфных схемам из табл.4.4*

Элемент схемы	Варианты включения элементов					
	1	2	3	4	5	6
$C1$	3,6	3,4	3,6	3,4	2,3	2,3
$C2$	4,5	4,7	6,7	4,7	2,7	2,7
$C3$	6,7	5,6	2,5	2,5	5,6	4,5
$R1$	1,2	1,2	1,4	1,6	1,4	1,6
$R2$	2,5	2,5	4,5	5,6	4,5	5,6
$R3$	2,7	2,7	4,7	6,7	4,7	6,7
$R4$	1,6	1,4	1,6	1,4	1,2	1,2
$R5$	6,7	4,7	6,7	4,7	2,7	2,7
$R6$	1,4	1,6	1,2	1,2	1,6	1,4
$R7$	3,4	3,6	2,3	2,3	3,6	3,4

По алгоритму 4.4 получен ряд оригинальных схем, защищенных авторскими свидетельствами, например [72–75]. Некоторые из них внедрены в производство. Программная реализация алгоритма 4.4 вошла в состав программных средств [76], включенных в Государственный фонд алгоритмов и программ.

## Список использованной литературы

1. *Алгоритмические* исследования в комбинаторике /Под ред. И. А. Фараджева. М.: Наука, 1978. 187 с.
2. *Харари Ф.* Теория графов. М.: Мир, 1973. 301 с.
3. *Рейнгольд Э., Ниввергельт Ю., Део Н.* Комбинаторные алгоритмы. М.: Мир, 1980. 476 с..
4. *Гэри М., Джонсон Д.* Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982. 416 с.
5. *Большая советская энциклопедия* /Гл. ред. А. М. Прохоров. Изд. 3-е. М.: Сов. энциклопедия, 1976. Т. 23. 640 с.
6. *Норенков И.П.* Введение в автоматизированное проектирование технических устройств и систем. М.: Высш. шк. 1980. 311 с.
7. *Большая советская энциклопедия* /Гл. ред. А. М. Прохоров. Изд. 3-е. М.: Сов. энциклопедия, 1976. Т. 24. 608 с.
8. *Калужнин Л. А., Суцанский В. Н.* Преобразования и перестановки. М.: Наука, 1979. 112 с.
9. *Урманцев Ю. А.* Симметрия природы и природа симметрии. М.: Мысль, 1974. 229 с.
10. *Математическая энциклопедия* /Гл. ред. И. М. Виноградов. М.: Сов. энциклопедия, 1984. Т. 4. 1216 с.
11. *Математическая энциклопедия* /Гл. ред. И. М. Виноградов. М.: Сов. энциклопедия, 1979. Т 2. 1104 с.
12. *Расстригин Л. А.* Адаптация сложных систем. Рига: Зинанте, 1981. 375 с.
13. *Букатова М. Л.* Эволюционное моделирование и его приложение. М.: Наука, 1979. 230 с.
14. *Агасова С. М.* Алгоритмы синтеза автоматов на программируемых матрицах. М.: Радио и связь, 1987. 136 с.
15. *Mc-Cluskey E. J.* Detection of Group Invariance on Total Symmetry of a Boolean Function //BSTJ. 1956. Vol. 35, № 6. P. 1445 – 1453.
16. *Комлев О. В., Свирицева Э. А.* Выявление симметрий системы инцидентностей // Алгоритмы. Ташкент: Изд. Академии наук Республики Узбекистан, 1998. Вып 86. С. 100 – 105.
17. *Математическая энциклопедия* /Гл. ред. И. М. Виноградов. М.: Сов. Энциклопедия. 1982. Т.3. 1184 с.
18. *Фараджев И. А.* Генерирование неизоморфных графов с заданным распределением степеней вершин //Алгоритмические исследования в комбинаторике. М.: Наука, 1978. С. 11–19.
19. *Свирицева Э. А. Тартаковский И. И.* Вопросы эффективного генерирования графов //Радиотехника. 1980. Вып. 52. С. 132–137.
20. Алгоритм проверки каноничности частично заполненной матрицы / *В. А. Зайченко, А. В. Иванов, М. З. Розенфельд, И. А. Фараджев* //Алгоритмические исследования в комбинаторике. М.: Наука, 1978. С. 19–25.
21. *Курош А. Г.* Теория групп. М.: Наука, 1967. 648 с..

22. *Каргаполов М. И., Мерзляков Ю. И.* Основы теории групп. М.: Наука, 1977. 240 с.
23. *Диниц Е. А., Зайцев М. А.* Алгоритм генерации неизоморфных деревьев // Автоматика и телемеханика. 1977. № 4. С. 121 – 126.
24. *Диниц В. А., Зайцев М. А.* Линейное упорядочение и порождение деревьев // Алгоритм. исслед. в комбинаторике. М.: Наука, 1978. С. 85–99.
25. *James K. R., Riha W.* The produktion of trees and rooted trees of order  $\leq 15$ , classified according partition. Rept 47, Centre for Computer Studies, Univ. Leeds, 1974.
26. *Кристофидес Н.* Теория графов. Алгоритмический подход. М.: Мир, 1988. 432 с.
27. *Jordan C.* Sur les assemblages de lignes // *J.f. die reine und angewandte. Math.*, 70 (1869). P. 185–190.
28. *Поля Д.* Комбинаторные вычисления для групп, графов и химических соединений // Перечисл. задачи комбинатор. анализа. М.: Мир, 1979. С. 36–136.
29. *Зыков А. А.* Основы теории графов. М.: Наука, 1987. 384 с.
30. *Поваров Г. Н.* О групповой инвариантности булевых функций // Применение логики в науке и технике. М.: Изд. АН СССР, 1960. С.266–340.
31. *Shannon C. E.* The Synthesis of Two-Terminal Switching Circuits // *BSTJ*. 1949. Vol. 28, № 1. P. 59–98.
32. *Биркгоф Г.* Теория структур. М.: ИЛ, 1952.
33. *Яблонский С. В.* Введение в дискретную математику. М.: Наука, 1986. 384 с.
34. *Горбатов В. А., Останков Б. Л., Фролов С. А.* Регулярные структуры автоматного управления. М.: Машиностроение, 1980. 280 с.
35. *Shannon C. E.* Symbolic Analysis of Relax and Switching Circuits // *Trans. AIEE*. 1938. Vol 57. P. 713–723.
36. *Закревский А. Д.* Логический синтез каскадных схем. М.: Наука, 1981. 416 с.
37. *Поваров Г. Н.* О сравнении БФ // Синтез автоматов и управление на сетях связи. М.: Наука, 1973. С. 88–92.
38. *Caldwell S. H.* The Recognition and Identification of Symmetric Switching Circuits // *Trans. AIEE*. 1954. Vol. 73, part 1. P. 142–146.
39. *Поваров Г. Н.* Графический метод синтеза симметрических контактных схем // Приборостроение. 1956. № 12. С.7–9.
40. *Das S. R., Sheng C. L.* On detection total or partial Symmetry of Switching functions // *IEE, Trans on Computers*. 1971. March. Vol. C. 20, № 3. P. 352–355.
41. *Arnold R. F., Harrison M. A.* Algebraic properties of Symmetric and partially Symmetric Boolean functions // *IEEE Trans Electronic Computers*. 1963. June. Vol. EC-12. P. 244–251.
42. *Mukhopadhyay A.* Detection of total or partial Symmetry of a switching function with the use of decomposition charts // *IEEE Trans. Elektronik. Computers (Correspondence)*. 1963. October. Vol. EC-12. P. 553–557.
43. *Кобринский Н. Е., Трахтенброт Б. А.* Введение в теорию конечных автоматов М.: Госиздат физ.-мат. лит. 1962. 404 с.

44. Кузнецов О.П., Адельсон-Вельский Г. М. Дискретная математика для инженера. М.: Энергия, 1980. 341 с.
45. Горбатов В. А. Основы дискретной математики. М.: Высш. шк. 1986. 310 с.
46. Шишкин В. Н. Метод минимизации бинарных программ, реализующих логические функции // Автоматика и телемеханика. 1983. № 1. С. 121–129.
47. Золотаревская М. Н. О реализации систем булевых функций в ПЛУ // Проблемы управления в технике, экономике, биологии. М.: Наука, 1981. С. 64–70.
48. Князева В. Ф., Князев В. В., Свирицева Э. А. Приведение булевой функции к минимальной ортогональной форме // Кибернетика и системный анализ. 1993. № 5. С. 17–25
49. Сигорский В. П. Математический аппарат инженера. К.: Техніка, 1975. 768 с.
50. Мэзон С., Циммерман Г. Электронные цепи, сигналы и системы. М.: Изд-во иностр. лит., 1963. 619 с.
51. Оптимальная реализация линейных электронных RLC-схем / А. А. Ланнэ, В. Д. Михайлова, Б. С. Саркисян, Я. Н. Матвийчук. К.: Наук. думка, 1981. 208 с.
52. Балабанян Н. Синтез электрических цепей. М.: Госэнергоиздат, 1961. 416 с.
53. Гиллемин Е. И. Синтез пассивных цепей. М.: Связь, 1970. 720 с.
54. Синтез переменных состояний / П. А. Ионкин, Н. Г. Максимович, В. Г. Миронов и др. Львов: Вища шк., 1982. 311 с.
55. Ланнэ А. А. Оптимальный синтез линейных электрических цепей. М.: Связь, 1969. 293 с.
56. Ланнэ А. А. Оптимальный синтез линейных электронных схем. М.: Связь, 1978. 336 с.
57. Ланнэ А. А., Саркисян Б.С. Решение компонентных уравнений в задачах реализации электронных схем // Радиотехника, 1977. Вып. 32, № 7. С. 27–33.
58. Бандман О. М. Синтез электронных RC-схем. М.: Наука, 1966. 248 с.
59. Сигорский В. П., Петренко А. И. Алгоритмы анализа электронных схем. М.: Сов. радио, 1976. 608 с.
60. Величко Ю. Т. Мощности в линейной цепи // Радиотехника. 1969. Вып. 8. С 3–11.
61. Сигорский В. П. Теорема об определителе суммы матриц и ее применение для выражения коэффициентов полиномов функции электронной схемы // Радиотехника. 1968. № 10. С. 81–87.
62. Свирицова Е. О., Гринченко Л. В. Прямой синтез RLC-схем // Матеріали Харк. обл. наук.-техн. конф. Сер. Радіоелектроніка і керування. Х., 1972. С. 10–19.
63. Анисимов В. И., Козьмин Н. Г. Топологический анализ электронных схем методом  $k$ -деревьев // Теорет. электротехника. 1971. Вып. 12. С. 9–16.
64. Корн Г., Корн Т. Справочник по математике для научных работников и инженеров. М.: Наука, 1968. 720 с.
65. Хорошевский А. Н. Автоматизация синтеза линейных активных RC-схем с операционными усилителями: Дис. ... канд. техн. наук. Х., 1984. 197 с.
66. Мигулин Н. Н., Чаповский М. Э. Интегральные схемы в радиоэлектронных устройствах. К.: Техника, 1978. 232 с.

67. Подольский Л. Э. Об одном методе коррекции высокочастотных фильтров, основанном на эквивалентных преобразованиях //Радиотехника и электроника, 1977. Т. 22, № 8. С. 1742–1744.

68. Свирицева Э. А., Хорошевский А. Н. К автоматизации структурного синтеза РС-схем с операционными усилителями /Харьк. ин-т радиоэлектрон. К., 1984. 33 с. Деп. в УкрНИИТИ 06.08.84, № 183 УК-84.

69. Гринченко Л. В., Колесников О. В., Свирицева Э. А. Структурный синтез неизоморфных схем с операционными усилителями // Электронное моделирование. 1983. № 6. С. 54–60.

70. Трохименко Я. К. Метод обобщенных чисел и анализ линейных цепей. М.: Сов. радио, 1972. 311 с.

71. Знаменский А. Е., Теллюк И. Н. Активные РС-фильтры. М.: Связь, 1970. 279 с.

72. А. с. 470911, МКП<sup>3</sup>. Режекторный фильтр / Э. А. Свирицева, С. В. Березникер, Л. В. Гринченко и др. Оpubл. 15.05.75. Бюл. № 18. 2с.

73. А. с. 807484, МКП<sup>3</sup>. В 25 В 11/14. Активный РС-фильтр / Э. А. Свирицева, Ю. А. Хоменко. Оpubл. 23.02.81. Бюл. № 7. 2 с.

74. А. с. 926760, МКП<sup>3</sup>. Звено активного РС-фильтра с расширенной полосой пропускания / Э. А. Свирицева, А. Н. Хорошевский. Оpubл. 07.05.82. Бюл. № 17. 4 с.

75. А. с. 936388, МКП<sup>3</sup>. Звено активного РС-фильтра / Э. А. Свирицева, Ю. А. Хоменко. Оpubл. 15.06.82, Бюл. № 22. 4 с.

76. Программа структурного и параметрического синтеза схем с операционными усилителями СХЕМА / Э. А. Свирицева, А. Н. Хорошевский. Заяв. 29.12.89 в РФАП УССР, № П6384. К., 1989.

Наукове видання

**Свіршова Ельвіра Олександрівна**

**Структурний синтез  
неізоморфних систем  
з однорідними  
компонентами**

Російською мовою

Научное издание

**Свирищева Эльвира Александровна**

**Структурный синтез  
неизоморфных систем  
с однородными  
компонентами**

Редактор *О.Л.Алієва*

Виконавець комп'ютерного набору та оригіналу-макета *О.О.Помазановський*

Коректор *О.Л.Левкович*

Підписано до друку 21.10.98. Формат 60x84  $\frac{1}{16}$ . Папір друк. №1. Гарнітура Таймс.

Друк офсетний. Умов. друк.арк. 14,88. Умов. фарбо-відб. 15. Обл.-вид. арк. 16.

Тираж 300 пр. Зам. 32.

Харківський державний технічний університет радіоелектроніки.

Україна, 310726, Харків, просп. Леніна, 14.

Віддруковано з оригіналу-макета у видавництві

Харківського державного технічного університету радіоелектроніки.

Україна, 310726, Харків, просп. Леніна, 14.



**Свірцова Е.О.**

С24 Структурний синтез неізоморфних систем з однорідними компонентами. –Х. : ХТУРЕ, 1998. – 256 с.  
ISBN 5-7763-8886-4.

У монографії викладено теоретичні основи структурного синтезу неізоморфних систем з однорідними компонентами. Застосування запропонованих засобів (математичні моделі, методи та алгоритми) такого синтезу проілюстровано на задачах з області програмування, схемотехніки, конструктивного перерахування абстрактних комбінаторних об'єктів, котрі найчастіше використовуються для подання технічних структур та інших матеріальних систем.

Для наукових працівників, інженерів, аспірантів та студентів, які займаються питаннями оптимізації систем, що розробляються, у просторі структур, і математиків, які цікавляться застосуванням комбінаторних методів, теорії груп, теорії відношень.

Переопрацьовано у 2005 році.

С 1402050000 – 07  
98

ББК 22.174 я9