

Dimitr B. Šiškov

Агрегатная декомпозиция автоматов и управление агрегатами

*Kybernetika*, Vol. 3 (1967), No. 5, (438)--450

Persistent URL: <http://dml.cz/dmlcz/124361>

## Terms of use:

© Institute of Information Theory and Automation AS CR, 1967

Institute of Mathematics of the Academy of Sciences of the Czech Republic provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these

*Terms of use.*



This paper has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library*  
<http://project.dml.cz>

## Агрегатная декомпозиция автоматов и управление агрегатами

Димитър Б. Шишков

Рассматривается способ агрегатного представления (декомпозиция) произвольного дискретного автомата в виде некоторой блочной структуры, состоящей из агрегата управления и подчиненных агрегатов. Определяются функции агрегата управления и приводятся алгоритмы синтеза управляющего и подчиненных агрегатов. Дается обобщение задачи синтеза управляющего агрегата относительно произвольной сложной системы управления, агрегаты которой описаны автоматным языком.

### 1. ПОСТАНОВКА ЗАДАЧИ

Агрегатирование сложных систем обычно может быть произведено неоднозначно. При этом, в зависимости от принятого уровня абстракции, любой агрегат системы (рассматриваемой в смысле, указанной в [1]) представим также в виде некоторой сложной системы [2].

Для решения множества задач синтеза сложных систем можно использовать довольно развитый аппарат теории автоматов [3]. Это основывается на общности фундаментальных черт теории управления и общей теории автоматов при рассмотрении детерминированных систем обработки информации и универсальность атоматной интерпретации поведения объектов и способов их управления [4].

В настоящей работе предлагается регулярный метод представления агрегатов, описываемых автоматами, в виде сложных систем управления с определенной структурой. Такая задача может возникать при оптимизации структуры сложной системы, поскольку организация системы имеет существенное значение для определения качественных сторон алгоритмов управления [5], а также при структурном синтезе агрегатов управления сложных систем. В некоторой степени она аналогична задаче декомпозиции автоматов и поэтому имеет самостоятельное значение для решения чисто автоматных проблем: агрегатирование сложных автоматов с целью применения методов синтеза, ограничен-

ных сложностью автоматов (размещение (кодирование) состояний, минимизация операторов), синтез надежных автоматов [6] и ряд других задач блочного синтеза.

Задача агрегатного представления автоматов может быть поставлена следующим образом.

Задан дискретный автомат  $D$ . Требуется представить его в виде некоторой сложной системы (рис. 1), эквивалентной исходного автомата, составленной

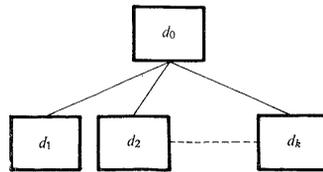


Рис. 1.

управляющим агрегатом  $d_0$  и подчиненными агрегатами  $d_1, d_2, \dots, d_k$ , каждый из которых может быть связан с другими подчиненными агрегатами.

Не снижая общность постановки задачи, введем ограничение, что автомат  $D$  является конечным начальным автоматом, определенным множествами внутренних состояний  $A = (a_1, a_2, \dots, a_n)$ , где  $a_1$  — начальное состояние автомата, входных сигналов  $X = (x_1, x_2, \dots, x_m)$  и выходных сигналов  $Y = (y_1, y_2, \dots, y_p)$  и функциями возбуждения и выходов  $\delta(a, x)$  и  $\lambda(a, x)$  соответственно.

## 2. СИНТЕЗ АГРЕГАТОВ

Разделим произвольным способом множество внутренних состояний автомата на  $k$  непересекающихся подмножеств

$$(1) \quad A_\alpha = \{\bar{d}_1; \bar{d}_2; \dots; \bar{d}_k\},$$

где каждый блок подмножества  $d_i$  ( $i = 1, 2, \dots, k$ ) содержит не менее, чем два состояния (т.е.  $2k \leq n$ ), а  $\alpha$  — индекс принятого варианта разделения. Тогда все переходы автомата  $D$ , совершаемые при поступлении некоторой буквы входного алфавита  $X$ , могут быть разделены на две группы:

1. Переходы, осуществляемые между состояниями подмножества  $\bar{d}_i$  (т.е.  $\delta(a_i, x_j) = a_k$ , где  $a_i \in \bar{d}_i$  и  $a_k \in \bar{d}_i$ ), которые будем называть *внутренними переходами* и

2. Переходы между состояниями, включенных в разные подмножества ( $\delta(a_i, x_j) = a_k$ , где  $a_i \in \bar{d}_i$ , а  $a_k \in \bar{d}_j$ ;  $j = 1, 2, \dots, i - 1, i + 1, \dots, k$ ), которые будем называть *внешними переходами*.

Если при выбранном способе разделения (1) автомата  $D$  положим, что все внешние переходы являются запрещенными, получим неполностью определенный автомат  $D'$ , состоящий из  $k$  несвязанных между собой автоматов  $d'_1, d'_2, \dots, d'_k$ . Любой автомат  $d'_i$  ( $i = 1, 2, \dots, k$ ) будет определен на множествах состояний  $\bar{d}_i$ , входных и выходных сигналов  $X$  и  $Y$  (отдельные элементы которых могут оказаться пустыми относительно  $d'_i$ ) и некоторые функции  $\delta'(a, x)$ ,  $\lambda'(a, x)$ , являющиеся интервалами функций переходов и выходов заданного автомата  $D$ :  $\delta(a, x)$  и  $\lambda(a, x)$  соответственно.

Таким образом, если входные слова, преобразуемые автоматом  $D$ , не содержат последовательности букв, вызывающих внешние переходы, то функциони-

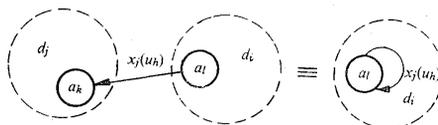


Рис. 2.

рование автомата  $D$  выразится в разнесенной во времени работе агрегатов  $d'_1, d'_2, \dots, d'_k$ .

Учитывая внешние переходы автоматов  $d_1, d_2, \dots, d_k$  и временные соотношения их работы, нетрудно убедиться, что задача эквивалентного преобразования заданного автомата  $D$  в сложной системе агрегатов будет решена полностью (без вышеуказанных ограничений на входных словах) введением управляющего автомата  $d_0$  (см. рис. 1), выполняющего функцию временного переключения сигналов алфавита  $X$  между агрегатами  $d_1, d_2, \dots, d_k$ .

Переключение сигналов входного алфавита  $X$  к подчиненному агрегату  $d_i$  ( $i = 1, 2, \dots, k$ ) должно происходить при любых переходах  $\delta(a_l, x_j) = a_k \in \bar{d}_i$ . Поэтому, кроме букв алфавита  $X$ , входной алфавит агрегата  $d_0$  должен включать и буквы некоторого вспомогательного алфавита  $U = (u_1, u_2, \dots, u_v)$ , который будем называть *алфавитом переключения*;  $v$  — количество внешних переходов агрегатированного автомата  $D$ . Каждую букву этого алфавита  $u_h$  ( $h = 1, 2, \dots, v$ ) будем отождествлять с некоторым выходным сигналом  $u_h$  которым отмечен  $h$ -тый внешний переход автомата. Так как внешние переходы размечены также некоторыми символами выходного алфавита  $Y$ , то каждая буква алфавита  $U$  может совпадать с некоторой (в частности пустой) буквой алфавита  $Y$ .

Следовательно, любой внешний переход  $\delta(a_l, x_j) = a_k$ , ( $a_l \in \bar{d}_i$ ,  $a_k \in \bar{d}_j$ ), формирующий выходной сигнал переключения  $u_h$  должен или не изменить исходное для него состояние ( $a_l$ ) автомата  $d_i$  или установить его в некотором начальном (нерабочем) состоянии. Так как при агрегатирование автоматов по (1), обычно

отдельные подчиненные агрегаты являются неинициальными автоматами, первая модель (замещение всех внешних заходящих\* переходов автомата  $d_i$  с петлевыми переходами, при которых вырабатывается пара выходных сигналов  $y, u$  (рис. 2)) оказывается более общей и будет использована в дальнейшем. Таким образом, выходной алфавит любого управляемого агрегата  $d_i$  будет расширен буквами  $U$ , которыми размечены трансформированные петлевые переходы.

Остался невыясненным вопрос о входном алфавите подчиненных агрегатов. Для этого следует определить смысл внешних переходов относительно агрегатов, для которых они являются входными переходами. Поясним это на примере рис. 2. Сигнал  $u_i$  должен не только переключить сигналы входного алфавита  $X$  с агрегата  $d_i$  на агрегат  $d_j$ , но и поставить  $d_j$  в состояние  $a_k$ , независимо от того в каком состоянии он находился до этого. Следовательно, под сигналом  $u_k$  автомат должен переходить в состояние  $a_k$  из любого состояния  $a \in \bar{d}_j$ . Это указывает на возможность минимизировать подчиненные агрегаты за счет означения всех внешних переходов, входящих в одно и то же состояние, одинаковыми сигналами алфавита  $U$ . Итак входной алфавит подчиненного агрегата  $d_i$  образуется буквами алфавита  $X$  и буквами алфавита  $U$ , которыми отмечены все входящие в  $d_i$  внешние переходы.

Отметим, что в частном случае (когда внешние переходы относятся только к одному состоянию агрегата  $d_i$  и агрегат  $d_i$  может быть определен как инициальный автомат), входной алфавит для некоторых подчиненных агрегатов может совпадать с алфавитом  $X$ .

Сформулируем алгоритм синтеза подчиненных агрегатов.

**Алгоритм 1.** 1. Выбирается способ разделения заданного автомата (1).

2. Определяется алфавит переключения  $U$  — все внешние переходы размечаются произвольным образом выходными сигналами последовательности  $u_1, u_2, \dots, u_n$  (при этом все внешние переходы, входящие в одинаковые состояния подчиненных агрегатов, размечаются одинаковыми сигналами  $u_k$ ).

3. Дополняется выходной алфавит  $Y$  подчиненных агрегатов буквами алфавита переключения  $U$  — все выходящие внешние переходы замещаются петлевыми внутренними переходами, формирующими выходные сигналы, которыми размечены соответствующие переходы.

4. Дополняется входной алфавит  $X$  подчиненных агрегатов буквами алфавита переключения  $U$  (только для неинициальных подчиненных агрегатов) — для выходных сигналов алфавита  $U$ , соответствующих внешним входящим переходам образуются переходы в том же самом состоянии подчиненного агрегата, в которое входит внешний переход, из всех его состояний.

**Пример.** Синтезируем подчиненные агрегаты автомата, показанного на рис. 3,

\* Здесь и далее для означения переходов автомата принята терминология С. Berge [7].

442 образованные согласно геометрическим разделениям, означенным пунктирными линиями.

Условие задачи фиксирует следующее разделение автомата

$$(2) \quad \{\bar{d}_1; \bar{d}_2; \bar{d}_3; \bar{d}_4\} = \{0, 1, 6, 7; 2, 3, 4, 5; 10, 11, 12, 13; 8, 9, 14, 15\}.$$

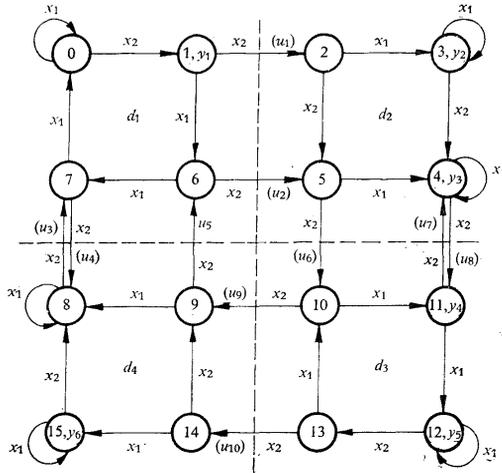


Рис. 3.

Размерность алфавита переключения  $v$  определится числом внешних переходов, пересеченных пунктирными линиями (рис. 3)

$$(3) \quad U = (u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9, u_{10}).$$

В соответствии п. 3 алгоритма 1 уточняем выходные алфавиты подчиненных агрегатов

$$(4) \quad \begin{aligned} d_1 &= (y_1, u_1, u_2, u_4), \\ d_2 &= (y_2, y_3, u_6, u_8), \\ d_3 &= (y_4, y_5, u_7, u_9, u_{10}), \\ d_4 &= (y_6, u_3, u_5) \end{aligned}$$

и строим их неполные графы — рис. 4а–г.

По п. 4 алгоритма 1 находим входные алфавиты подчиненных агрегатов

(5)

$$\begin{aligned}
 d_1 &- (x_1^1, x_2^1, u_3, u_5), \\
 d_2 &- (x_1^2, x_2^2, u_1, u_2, u_7), \\
 d_3 &- (x_1^3, x_2^3, u_6, u_8), \\
 d_4 &- (x_1^4, x_2^4, u_4, u_9, u_{10}).
 \end{aligned}$$

На основании (2), (3), (4), (5) и рис. 3 и 4 нетрудно получить окончательный

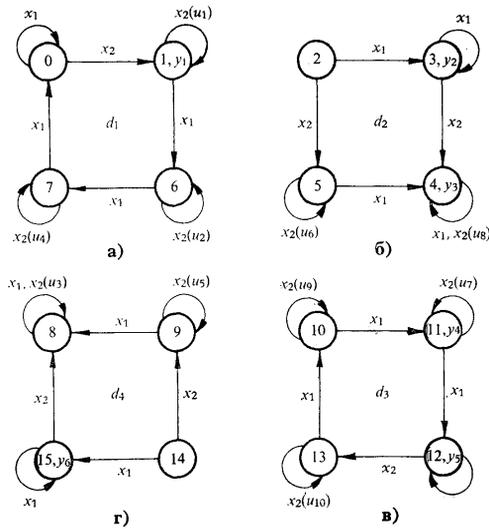


Рис. 4.

Табл. 1а.

Выходные сигналы	—	$y_1$	—	—
Состояния	0	1	6	7
Входные сигналы				
$x_1^1$	0	6	7	0
$x_2^1$	1	$1, u_1$	$6, u_2$	$7, u_4$
$u_3$	7	7	7	7
$u_5$	6	6	6	6

Табл. 1б.

Выходные сигналы	—	$y_2$	$y_3$	—
Состояния				
Входные сигналы	2	3	4	5
$x_1^2$	3	3	4	4
$x_2^2$	5	4	4, $u_8$	5, $u_6$
$u_1$	2	2	2	2
$u_2$	5	5	5	5
$u_7$	4	4	4	4

Табл. 1в.

Выходные сигналы	—	$y_4$	$y_5$	—
Состояния				
Входные сигналы	10	11	12	13
$x_1^3$	11	12	12	10
$x_2^3$	10, $u_9$	11, $u_7$	13	13, $u_{10}$
$u_6$	10	10	10	10
$u_8$	11	11	11	11

Табл. 1г.

Выходные сигналы	—	—	—	$y_6$
Состояния				
Входные сигналы	8	9	14	15
$x_1^4$	8	8	15	15
$x_2^4$	8, $u_3$	9, $u_5$	9	8
$u_4$	8	8	8	8
$u_9$	9	9	9	9
$u_{10}$	14	14	14	14

вид таблиц переходов и выходов подчиненных агрегатов  $d_1$ ,  $d_2$ ,  $d_3$  и  $d_4$  — табл. 1а—г соответственно.

### б) Синтез управляющего агрегата

Для синтеза управляющего агрегата исходными данными являются:

1. Алфавит переключения  $U = (u_1, u_2, \dots, u_v)$ ,
2. Число ( $k$ ) подчиненных агрегатов и
3. Временные соотношения работы подчиненных агрегатов в зависимости от поступления сигналов переключения  $u_1, u_2, \dots, u_v$ .

Алфавит переключения ( $U$ ) и количество ( $k$ ) подчиненных агрегатов определяются на этапе синтеза подчиненных агрегатов. Временные соотношения работы

Табл. 2.

Подчиненные агрегаты	1	2	3	4
Сигналы переключения	(0, 1, 6, 7)	(2, 3, 4, 5)	(10, 11, 12, 13)	(8, 9, 14, 15)
$u_1$	2	—	—	—
$u_2$	2	—	—	—
$u_3$	—	—	—	1
$u_4$	4	—	—	—
$u_5$	—	—	—	1
$u_6$	—	3	—	—
$u_7$	—	—	2	—
$u_8$	—	3	—	—
$u_9$	—	—	4	—
$u_{10}$	—	—	4	—

агрегатов в зависимости от поступления сигналов переключения задаются таблицей переключения. Таблица переключения содержит  $k$  колонок и  $v$  строк. В клетке таблицы, соответствующей колонке  $d_j$  и строке  $u_i$ , записывается номер того подчиненного агрегата  $d_j$  ( $j = 1, 2, \dots, k$ ), который должен включиться после формирования сигнала переключения  $u_i$ , если до этого функционировал агрегат  $d_i$ .

Поясним составление таблицы переключения для рассмотренного в предыдущем примере автомата (рис. 3) — табл. 2.

Из рис. 3 видно, что если в некоторый момент времени в рабочем состоянии находился автомат  $d_1$  и поступили такие последовательности входных сигналов алфавита  $X$ , которые вызывают формирование сигналов  $u_1, u_4$  или  $u_2$ , входной алфавит автомата должен переключиться к автоматам  $d_2, d_4$  или  $d_2$  соответственно (см. колонку 1 табл. 2). Аналогично заполнены и остальные клетки таблицы переключения.

Приведем алгоритм синтеза управляющего агрегата  $d_0$ .

**Алгоритм 2. 1.** Строится таблица переключений для выбранного варианта разделения.

2. Составляется таблица переходов и выходов агрегата: число состояний равно числу подчиненных агрегатов ( $k$ ); входными сигналами являются сигналы множества  $X$  и  $U$  (их число равно  $m + v$ ); под воздействием входного сигнала  $x_i$  агрегат  $d_0$  не изменяет свое состояние и формирует выходной сигнал  $x_i^k$  ( $k$  — номер исходного состояния); под сигналом  $u_h$  ( $h = 1, 2, \dots, v$ ) агрегат должен перейти в состояние, определяемое таблицей переключения.

Табл. 3.

Входные сигналы	Состояния			
	1	2	3	4
$x_1$	1, $x_1^1$	2, $x_1^2$	3, $x_1^3$	4, $x_1^4$
$x_2$	1, $x_2^1$	2, $x_2^2$	3, $x_2^3$	4, $x_2^4$
$u_1$	2	—	—	—
$u_2$	2	—	—	—
$u_3$	—	—	—	1
$u_4$	4	—	—	—
$u_5$	—	—	—	1
$u_6$	—	3	—	—
$u_7$	—	—	2	—
$u_8$	—	3	—	—
$u_9$	—	—	4	—
$u_{10}$	—	—	4	—

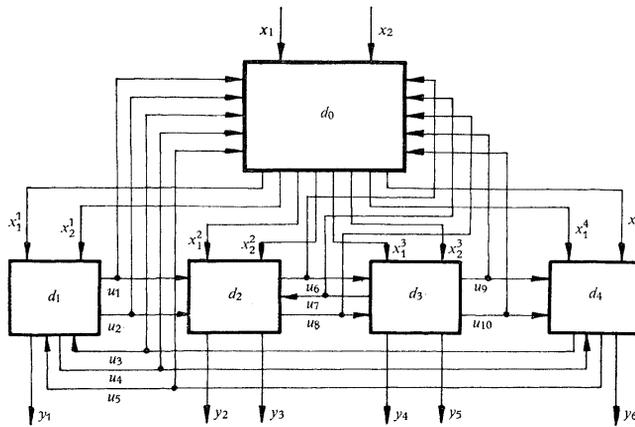


Рис. 5.

**Пример.** Следуя алгоритму 2, нетрудно построить таблицу (табл. 3) переходов и выходов агрегата управления для рассмотренного выше автомата (рис. 3).

Блочная схема агрегатированного автомата (рис. 3) приведена на рис. 5.

Применим предлагаемый метод агрегатирования автоматов для решения задачи экономичного кодирования сложных автоматов.

**Пример.** Найти экономичный вариант кодирования, следующий из алгоритмов [8, 9], автомата с линейной тактикой (автомат М. Л. Цетлина [10]), имеющего 16 состояний – табл. 4.

Табл. 4.

Выходные сигналы	Состояния															
	$y_0$	$y_0$	$y_0$	$y_0$	$y_0$	$y_0$	$y_0$	$y_0$	$y_1$							
Входные сигналы	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	1	2	3	4	5	6	8	8	9	10	11	12	13	14
1	1	2	3	4	5	6	7	15	9	10	11	12	13	14	15	7

В соответствии с [8] для выбора экономичного варианта кодирования требуется анализ 6435 неэквивалентных вариантов кодирования состояний автомата состояниями элементарных автоматов и соответствующий за ними направленный перебор из около  $5,5 \cdot 10^{10}$  [11] возможных неэквивалентных систем функций. Если допустить, что экономичный вариант может быть реализован на неминимальном количестве элементов памяти, то количество вариантов кодирования возрастет на 32767. Эта процедура должна быть исполнена для каждого автомата из выбранной системы элементарных автоматов. Очевидно, что при таких условиях использование алгоритмов [8, 9] весьма затруднено.

Применяя описанный метод, произведем агрегатирование заданного автомата по соотношению

$$\{\bar{d}_1; \bar{d}_2\} = \{0, 1, 2, 3, 4, 5, 6, 7; 8, 9, 10, 11, 12, 13, 14, 15\}.$$

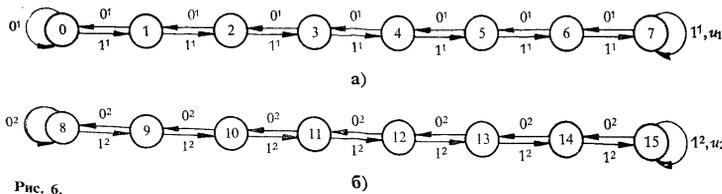


Рис. 6.

Следуя алгоритму 1, нетрудно увидеть, что автомат разбивается на два одинаковых (с точности до изоморфизма) инициальных автоматов (начальные состояния 7 и 15 соответственно) — рис. 6а и 6б.

Агрегат управления  $d_0$  получим по алгоритму 2 — рис. 7.

Экономичное кодирование агрегатированного автомата сведется до кодирования агрегата с 8-ми состояниями (35 (127) неэквивалентных вариантов кодирования) и агрегата с 2-мя состояниями (один вариант кодирования). Тогда-

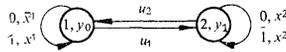


Рис. 7.

следуя [9], нетрудно получить экономичные варианты функции возбуждения и выходов агрегатов.

а) агрегат  $d_1$ :

$$\begin{aligned} q_{01} &= \bar{x}^1 \bar{Q}_2 \bar{Q}_3, \\ q_{11} &= x^1 Q_2 \bar{Q}_3, \\ q_2 &= x^1 (\bar{Q}_1 \vee \bar{Q}_2) \vee \bar{x}^1 (Q_1 \vee Q_2) \vee Q_3, \\ q_3 &= x^1 \bar{Q}_2 \vee \bar{x}^1 Q_2, \\ u_1 &= x^1 Q_1 Q_2 \bar{Q}_3; \end{aligned}$$

б) агрегат  $d_2$ :

$$\begin{aligned} q_{04} &= \bar{x}^2 \bar{Q}_5 \bar{Q}_6, \\ q_{14} &= x^2 Q_5 \bar{Q}_6, \\ q_5 &= x^2 (\bar{Q}_4 \vee \bar{Q}_5) \vee \bar{x}^2 (Q_4 \vee Q_5) \vee Q_6, \\ q_6 &= x^2 \bar{Q}_5 \vee \bar{x}^2 Q_5, \\ u_2 &= x^2 Q_4 Q_5 \bar{Q}_6; \end{aligned}$$

в) агрегат  $d_0$ :

$$\begin{aligned} q_{07} &= u_2, \\ q_{17} &= u_1, \\ \bar{x}^1 &= \bar{x} \bar{Q}_7, \\ \bar{x}^2 &= \bar{x} Q_7, \\ x^1 &= x \bar{Q}_7, \\ x^2 &= x Q_7, \\ y_1 &= Q_7, \\ y_0 &= \bar{Q}_7, \end{aligned}$$

где  $q_1, q_4, q_7$  — функции возбуждения триггеров с отдельными входами

$Q_1, Q_4, Q_7; q_2, q_3, q_5, q_6$  — функции возбуждения триггеров со счетными входами  $Q_2, Q_3, Q_5, Q_6$ .

449

### 3. УПРАВЛЕНИЕ АГРЕГАТАМИ

Рассмотренную задачу синтеза агрегата управления можно считать частным случаем синтеза управляющего агрегата сложной системы по заданной совокупности подчиненных агрегатов, выраженных автоматным языком и по алгоритму их управления.

В обобщенной задаче синтеза управляющего агрегата  $D_0$  следует допустить, что возможна произвольная комбинация одновременного функционирования подчиненных агрегатов  $D_1, D_2, \dots, D_k$  (система  $d_1, d_2, \dots, d_k$  работает только последовательно), а функциональные связи (внешние переходы) между агрегатами в общем случае описываются некоторым функциональным алфавитом  $\mathbf{U} = (u_1, u_2, \dots, u_v)$ , где  $u_h$  ( $h = 1, 2, \dots, v$ ) — некоторая временная булева функция, равная сигналу  $u_h$ .

Тогда, синтез управляющего агрегатами  $D_1, D_2, \dots, D_k$  сложной системы агрегата  $D_0$  может быть произведен на основании таблицы переключения (аналогичной табл. 2), составленная с учетом временных соотношений работы подчиненных агрегатов и множество функционалов  $\mathbf{U} = (u_1, u_2, \dots, u_v)$ .

Число колонок таблицы переключения ( $N$ ) (число состояний управляющего агрегата) определится неравенством

$$(6) \quad 1 \leq N \leq 2^k - 1.$$

Зависимость (6) следует из того, что каждый подчиненный агрегат может находиться в один из двух возможных режимов: функционирование или простой (неисключающий внутренних процессов самого агрегата).

$N = 1$  соответствует параллельной работе всех подчиненных агрегатов, а  $N = 2^k - 1$  — функционирование, при котором встречаются все возможные комбинации параллельно-последовательной работы агрегатов. Для синтеза управляющего автомата при агрегатной декомпозиции автоматов (последовательная работа подчиненных агрегатов) соотношение (6) становится  $N = k$ .

Задание функционалов множества  $\mathbf{U} = (u_1, u_2, \dots, u_v)$  может быть представлено любым автоматным языком, а синтез описываемого ими автомата произведен известными методами детерминизации систем графов [12]. В более частных, но довольно широко распространенных случаях (например при синтезе устройства управления электронных цифровых машин), синтез функционалов множества  $\mathbf{U}$  сводится к синтезу микропрограммных автоматов [3].

(Поступило 27. февраля 1967г.)

- [1] Л. М. Дронфорт, С. Г. Орадовская, В. Б. Соколов: Некоторые вопросы теории систем. В сб. „Техническая кибернетика“, Наука, Москва 1965, 304—307.
- [2] Н. П. Бусленко: К теории сложных систем. Известия АН СССР „Техническая кибернетика“ (1963), 5, 7—18.
- [3] В. М. Глушков: Синтез цифровых автоматов. Физматгиз, Москва 1962.
- [4] M. A. Arbib: A Common Framework for Automata Theory and Control Theory. J. Soc. Industr. and Appl. Math. (1965), 2, 206—222.
- [5] A. Straszak: Algorytmy sterowania w wielkich systemach „Problemi sterowania wielkimi systemami“, Polska akademia nauk, Wrocław, Warszawa, Kraków 1964, 99—123.
- [6] Б. С. Флейшман: Статистическая теория надежного функционирования конечных автоматов при наличии помех. В сб. „Теория конечных и вероятностных автоматов“, Наука, Москва 1965, 381—323.
- [7] C. Berge: Théorie des graphes et ses applications. Dunod, Paris 1958.
- [8] R. E. Stearns, J. Hartmanis: On the State Assignment Problem for Sequential Machines II. IRE Trans. Electr. Comp. EC-10 (1961) 4, 593—604.
- [9] Л. К. Ларин, Л. М. Осинский: Экономичное кодирование состояний конечного автомата. Кибернетика (1966), 3, 46—51.
- [10] М. Л. Цетлин: О поведении конечных автоматов в случайных средах. Автоматика и телемеханика 22 (1961), 10, 1345—1354.
- [11] E. J. McCluskey, S. H. Unger: A Note on Number of Internal Variable Assignments for Sequential Switching Circuits. IRE Trans. Electr. Comp. EC-8 (1959), 4, 439—441.
- [12] В. С. Гринберг: Детерминизация систем графов и синтез конечных автоматов. Сибирский математический журнал 7 (1966), 6, 1259—1267.

## VÝTAH

## Rozklad automatů na agregáty a řízení agregátů

DIMITR B. ŠIŠKOV

Vyšetřuje se rozklad libovolného konečného automatu na jednodušší automaty tzv. agregáty, z nich jeden je řídicím automatem a ostatní ovládací. Určuje se funkce řídicího automatu a uvádí se algoritmy syntézy řídicího a ovládacích automatů.

Zobecňuje se úloha syntézy řídicího automatu vzhledem k řídicímu agregátu libovolné složité soustavy, jejíž agregáty jsou popsány jako abstraktní automaty. Řídicí agregát je tvořen řídicím blokem, jehož syntéza se opírá o obecný algoritmus řízení složité soustavy a funkčním blokem, určeným na základě jednotlivých funkčních vazeb mezi agregáty. Ukazuje se, že v mnohých případech (například syntéza řadiče samočinného počítače) vede syntéza funkčního bloku k syntéze automatů řízených mikroprogramy.

*Инж. Димитър Борисов Шишков, София — Ц, ул. Патриарх Евтими 54. България.*