

Юрий Григорьевич Егоров родился в 1956 г., окончил в 1978 г. Серпуховское высшее военное командное училище. Д-р техн. наук, профессор кафедры “Системы и приборы ориентации, стабилизации и навигации” МГТУ им. Н.Э. Баумана. Действительный член Академии навигации и управления движением. Автор 56 научных работ в области автономной выставки и калибровки инерциальных навигационных систем.

Yu.G. Yegorov (b. 1956) graduated from the Serpukhov Higher Military Command School in 1978. D. Sc. (Eng.), professor of “Systems and Devices of Orientation, Stabilization and Navigation” department of the Bauman Moscow State Technical University. Full member of the Academy of Navigation and Motion Control. Author of 56 publications in the field of autonomous adjustment and calibration of inertial navigation systems.

Гассан Махмуд Аль-Хуссейн родился в 1963 г., окончил в 1988 г. университет в Сирии. Аспирант кафедры “Системы и приборы ориентации, стабилизации и навигации” МГТУ им. Н.Э. Баумана. Автор 2 научных работ в области калибровки инерциальных навигационных систем.

Ghassan Mahmoud Al-hussein (b. 1963) graduated from the Syrian University of Aleppo in 1988. Post-graduate of “Systems and Devices of Orientation, Stabilization and Navigation” department of the Bauman Moscow State Technical University. Author of 2 publications in the field of calibration of inertial navigation systems.

ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА

УДК 658.512

И. В. Рудakov, М. Давудпур

АЛГОРИТМ ДЕКОМПОЗИЦИИ ФОРМАЛЬНОЙ МОДЕЛИ ФУНКЦИОНАЛЬНОГО БЛОКА ДИСКРЕТНОГО УСТРОЙСТВА

Моделирование больших систем представляет собой задачу большой размерности, поэтому одним из методов исследования таких систем является метод декомпозиции, позволяющий разбивать исследуемую схему на части, проверяя работу каждой части, и последовательно добавлять к проверенной части новые фрагменты.

Метод анализа вычислительных дискретных устройств в рамках иерархических уровней проектирования (вычислительные системы или функциональные блоки) недостаточно эффективен, так как не позволяет учитывать, например, такие характеристики дискретного устройства, как временные параметры элементов, входящих в модель и, следовательно, выполнять надежную верификацию проекта. В работах по многоуровневому анализу предлагается рассматривать схемы устройства с разной степенью детализации. Предлагаемый

в настоящей работе метод декомпозиции модели функционального блока позволяет выполнять анализ правильности функционирования сложного дискретного устройства путем автоматического перехода на более низкий иерархический уровень.

Постановка задачи декомпозиции функционального блока дискретного устройства. Известно [1], что формализация процесса функционирования сложного дискретного устройства в виде сети Петри может быть определена в виде структурного автомата.

Автомат A называется реализацией автомата A_1 , ($A=R(A_1)$), если у автомата A существует подавтомат, изоморфный A_1 . Таким образом, если автомат A реализует автомат A_1 , то поведение автомата A с точностью до обозначения состояний, входных и выходных последовательностей совпадает с поведением A_1 в области определения A_1 , так как у автомата A должен быть некий подавтомат A_2B , изоморфный A_1 .

Задача декомпозиции автомата формулируется следующим образом: построить такую сеть автоматов N , результирующий автомат A_N которой реализует заданный автомат A , т.е. $A_N=R(A)$ [2], где $R(A)$ — реализация заданного автомата A .

В качестве формальной модели дискретного устройства в алгебраической теории автоматов [2, 3] используется понятие сети, определяемое следующим образом:

$$N = (Z, \{KA_i\}, W, \{f_i\}, \{PS_i\}, g),$$

где Z — входной алфавит сети; $\{KA_i = (C_i, Z_i, SG_i)\}$, $i = 1, \dots, n$ — множество компонентных автоматов сети. Входной алфавит компонентного автомата KA_i определяется выражением

$$Z_i = \begin{cases} Z_{1i} * Z_{2i}, & \text{если } Z_{1i} < > 0, \\ Z_{2i}, & \text{если } Z_{1i} = 0; \end{cases}$$

здесь Z_{1i} , Z_{2i} — внутренний и внешний входные алфавиты KA_i ; C_i — множество состояний KA_i ; $SG_i: C_i * Z_i$ — функция переходов KA_i , где $*$ — декартово произведение; W — выходной алфавит сети; $f_i: A_j \Rightarrow Z_{1i}$, $i \geq 1$, $j \leq n$ — множество функций соединения компонентных автоматов сети ($A_j = A_1 * A_2 * \dots * A_n$; A_j — автомат сети [2]); $PS_i: Z \Rightarrow Z_2$, $i = 1, \dots, n$ — множество входных функций; $g: A_i * Z \Rightarrow W$ — выходная функция сети ($A_i = A_1 * A_2 * \dots * A_n$ [2]).

Определяемую таким образом сеть рассматривают как общую модель совместной работы совокупности автоматов, так как в ней нет ограничений ни на выбор элемента базиса $\{KA_i\}$, ни на выбор структуры $\{f_i\}$.

Введем понятие функционирования сети автоматов в определенные тактовые моменты времени $t_n = \{t_1, t_2, t_3, \dots\}$.

Значение каждой переменной множества Z (входного алфавита сети) состоит из элементов множества $E = \{0, 1\}$. Пронумеровав множество индексов переменных из Z и используя понятие информационной группы [2], можно получить множество $E_z = E_{z1}, E_{z2}, E_{z3}, \dots, E_{zR}$, т.е. множество конечных последовательностей входных букв сети N длиной R .

Аналогичным образом запишем множество конечных последовательностей выходных букв сети N длиной R : $E_w = E_{w1}, E_{w2}, E_{w3}, \dots, E_{wR}$.

Для сети автоматов введем детерминированную функцию H , которая устанавливает соответствие между последовательностями входных и выходных букв, т.е. $H : E_z \Rightarrow E_w$, и для функции H введем множество K — множество чисел, которыми пронумерованы состояния функции: $K = \{k_1, k_2, k_3, \dots\}$.

Детерминированная последовательностная функция задается следующими соотношениями [2]:

$$\begin{aligned} k(1) &= k_1; \\ k(v+1) &= PH(k(v), e_z); \\ e_w &= PS(k(v), e_z), \end{aligned} \quad (1)$$

где $PH : E_z * K \Rightarrow K$; $PS : E_z * K \Rightarrow E_w$; $k(1)$ — начальное значение множества K ; $k(v)$ — состояние функции H на шаге с номером v ; e_z и e_w — входные буквы конечных последовательностей E_Z и E_W , подаваемые на сеть N на шаге с номером v .

Представим входы сети N переменными из множества Z , а саму сеть автоматов — последовательностной детерминированной функцией H . Тогда, если на вход N подавать буквы последовательности E_Z , то на выходе N будет последовательность выходных букв E_W , причем $E_W = H(E_Z)$, т.е. на каждом шаге на вход сети N поступают входные буквы e_z , сеть находится в одном из внутренних состояний, кодируемых с помощью элементов из множества K , а с выхода N снимаются выходные буквы e_w .

Входная буква и состояние на данном шаге однозначно определяют выходную букву на этом же шаге и состояние сети N на следующем, соответственно рекуррентным соотношением (1). Таким образом, функционирование сети автоматов N определяем как процедуру, устанавливающую соответствие между входными последовательностями, состояниями и выходными последовательностями.

Пусть сеть автоматов N задает некоторую операцию O_m , определяемую функцией H . Множество всех операций, реализуемых сетью

автоматов N , обозначим как $O = \{O_1, O_2, O_3, \dots, O_m, \dots\}$, где O_m — некоторая m -я операция, задаваемая N . Множество O определяет последовательность операций над входным алфавитом переменных Z с целью вычислить выходной алфавит переменных из множества W посредством функций перехода, соединения и выходной функции сети (или через последовательностную детерминированную функцию H). Функционирование сети N определяется выполнением операции $W = O_m(Z)$, задаваемым ее кодом $m = 1, 2, 3, \dots, M$. Иначе, сеть N осуществляет преобразование информации из слов входного алфавита в слова выходного алфавита в соответствии с алгоритмом преобразования.

Алгоритм декомпозиции сложного дискретного устройства, формализованного в виде сети Петри (сети компонентных автоматов), основывается на общей теореме декомпозиции автоматов [4], утверждающей, что множеству разбиений состояний автомата можно поставить в соответствие абстрактную сеть автоматов, если и только если множество разбиений является ортогональным. При этом устанавливается взаимно-однозначное соответствие между разбиениями и компонентными автоматами.

В доказательстве достаточности этой теоремы приводится конструктивный способ построения сети автоматов, результирующий автомат которой реализует заданный автомат. На этом способе базируется алгоритм декомпозиции формальной модели дискретного устройства, формализованного в виде сети Петри. Исходными данными для декомпозиции устройства является собственно модель устройства и множество разбиений состояний устройства, которые должны быть ортогональными. Согласно утверждению [3], формальная модель дискретного устройства может быть представлена в виде сети автоматов N , поэтому в дальнейшем будем рассматривать декомпозицию сети.

Алгоритм декомпозиции формальной модели дискретного устройства состоит из следующих действий.

1. Каждому разбиению множества состояний P_i ставим в соответствие некоторую функцию $F_i: C_n * Z_n \Rightarrow P_i$, такую, что

$$F_i(C_v, Z_f, T_m) = P_i(SG_n(C_v, Z_f, T_m)),$$

иначе: значение функции F_i , определяемое из (C_v, Z_f, T_m) , равно блоку P_i , который содержит состояние $C_s = SG_n(C_v, Z_f, T_m)$.

2. Образует разбиения TU_i и RU_i на множестве C_n и Z_n следующим образом:

— состояния C_v и C_s находятся в одном блоке разбиения TU_i , если и только если для любого Z_f справедливо:

$$F_i(C_v, Z_f, T_m) = F_i(C_s, Z_f, T_m), \text{ т.е. } C_v < - > C_s(TU_i);$$

– буквы входного алфавита Z_f и Z_t находятся в одном блоке разбиения RU_i , если и только если для любого C_v справедливо:

$$F_i(C_v, Z_f, T_m) = F_i(C_v, Z_t, T_m), \text{ т.е. } Z_f < - > Z_t(RU_i). \quad (2)$$

3. Строим сеть автоматов N , которая определяется как

$$N = (Z, \{KA_i\}, W, \{f_i\}, \{PS_i\}, g).$$

Для этого необходимо определить все компоненты, входящие в описание сети N .

Предположим, что входной алфавит сети N равен входному алфавиту схемы S , т.е. $Z = Z_n$; выходной алфавит сети N равен выходному алфавиту схемы S , т.е. $W = W_n$.

Строим компонентные автоматы $\{KA_i\}$: $KA_i = (C_i, Z_i, SG_i)$, $i = \overline{1, n}$ (n – количество компонентных автоматов).

Под декомпозицией схемы S , представленной сетью автоматов, понимается элемент сложного дискретного устройства, входящий в модель функционального блока.

Определим каждую составляющую, которая входит в описание KA_i :

а) предположим, что количество состояний элемента структуры (KA_i) равно количеству блоков в разбиении P_i , т.е. $C_i = P_i$;

б) чтобы определить входной алфавит элемента структуры (KA_i) , необходимо использовать введенные ранее разбиения TU_i и RU_i .

Входной алфавит определяется выражением (2). Полученные разбиения TU_i и P_i являются парой разбиений, т.е. каждый блок TU_i отображается любым входным сигналом в блок P_i , при этом TU_i является максимальным разбиением, образующим пару (TU_i, P_i) . В работе [2] показано, что если на вход функции $\{f_i\}$ поступают $P_{i1}, P_{i2}, P_{i3}, \dots, P_{iR}$ (R – количество входов функции $\{f_i\}$) – выходы компонентных автоматов $KA_{i1}, KA_{i2}, KA_{i3}, \dots, KA_{iR}$, то $P_{i1}, P_{i2}, \dots, P_{iR}$ – пары разбиений.

Следовательно, для определения элементов структуры, выходы которых присоединяются ко входу $\{f_i\}$, необходимо найти такое произведение P_i , которое не превосходит TU_i (такое произведение всегда найдется, так как множество P_i является ортогональным), и тогда выходы $KA_{i1}, KA_{i2}, \dots, KA_{iR}$ должны быть соединены со входом $\{f_i\}$.

в) находим функцию переходов компонентного автомата KA_i : $SG_i: P_i * Z_i \Rightarrow P_i$, т.е. значение функции переходов компонентного автомата равно блоку разбиения, содержащему $SG_n(P_i, RU_i)$.

д) кроме рассмотренных параметров, входящих в описание компонентного автомата, определяем времена переходов KA_i из одного состояния в другое при поступлении букв входного алфавита. В

описание функционирования сети автомата входит множество времен переходов из одного состояния в другое — T_n .

Пусть $P_i = \{B_1, B_2, B_3, \dots, B_l, \dots\}$ — некоторое множество разбиений состояний модели функционального блока, где B_l — первый блок в разбиении P_i .

При декомпозиции формальной модели, представленной сетью компонентных автоматов, число состояний компонентного автомата равно числу блоков в разбиении P_i . Следовательно, множество состояний можно выразить следующим образом: $C_i = \{B_1, B_2, B_3, \dots, B_l, \dots\}$. Поэтому множество времен перехода состояний компонентного автомата KA_i определяется следующим выражением:

$$TKA_i = \{T_{12}, T_{13}, \dots, T_{1n}, T_{23}, T_{24}, \dots, T_{2n}, \dots\},$$

где n — количество компонентных автоматов.

4. Строим функции соединения компонентных автоматов $f_i: A_j \Rightarrow Z_i$, где $A_j = A_1 * A_2 * \dots * A_n$.

Если использовать разбиения на множестве состояний модели, то это можно переписать в виде

$$f_i : P_{i1} * P_{i2} * \dots * P_{iR} \Rightarrow RU_i.$$

Обозначим множество R -компонентных векторов (компоненты которых являются блоками, взятыми по одному из каждого разбиения P_{ij} , где $j = i_1, i_2, i_3, \dots, i_R$) множеством U_i , т.е.

$$U_i = P_{i1} * P_{i2} * P_{i3} * \dots * P_{iR}.$$

Образуем подмножество U_{is} множества U_i таким образом, что в U_{is} попадают только те векторы из множества U_i , у которых пересечение всех компонентов не пусто.

Учитывая это, получим $f_i : U_{is} \Rightarrow RU_i$.

Значение функции соединения компонентного автомата f_i равно тому блоку разбиения RU_i , в который входит пересечение компонентов U_s .

5. Определяем множество входных функций следующим образом: $PS_i : Z \Rightarrow Z_i$, где $i = \overline{1, n}$.

Значение входной функции PS_i при подаче на вход компонентного автомата KA_i буквы входного алфавита z_f равно блоку разбиения TU_i , содержащему букву z_f . Следовательно, автомат KA_i не отличает те буквы входного алфавита Z , которые входят в один блок разбиения TU_i .

6. Строим выходную функцию сети компонентных автоматов:

$$g : A_i * Z \Rightarrow W, \text{ где } A_i = A_1 * A_2 * \dots * A_n, \quad i = \overline{1, n}.$$

Используя разбиения на множестве состояний модели функционального блока, получим

$$g : (P_{i1} * P_{i2} * P_{i3} * \dots * P_{in}) * Z \Rightarrow W.$$

Пусть $P_{i1} * P_{i2} * \dots * P_{in} = M$.

Образуем множество M_1 , в которое попадают только те векторы из множества M , у которых пересечение всех компонентов не пусто. Учитывая это, получим $g : M_1 * Z \Rightarrow W$.

Значение выходной функции сети N совпадает со значением функции выхода декомпозируемой модели на паре (C_q, z_f) , где C_q — состояние, попавшее в пересечение вектора $M_q < -M_1$.

Анализируя алгоритм декомпозиции модели функционального блока, основанный на конструктивном способе построения сети N , можно заключить, что свойства компонентных автоматов и их связи между собой зависят от выбранного множества разбиений $\{P_i\}$, $i = \overline{1, n}$. Число разбиений в множестве $\{P_i\}$ равно числу компонентных автоматов в сети N . Число блоков разбиений B_i в множестве $\{P_i\}$ равно числу внутренних состояний компонентного автомата KA_i . Элементы структуры имеют меньше состояний, чем функциональный блок, так как число блоков в разбиении P_i , кроме нулевого разбиения $P_i(0)$, меньше числа элементов в множестве C_n состояний исходной модели.

Разбиение P_i при помощи функции F_i однозначно определяют разбиения TU_i и RU_i . Разбиение TU_i показывает, какие элементы воздействуют на i -й элемент структуры, а разбиение RU_i определяет, какие буквы входного алфавита Z не различаются i -м элементом. Множества TU_i и RU_i являются наибольшими разбиениями, причем, чем больше TU_i , тем меньше выходов других элементов связано с KA_i , и чем больше RU_i , тем проще зависимость SG_i от внешнего алфавита Z .

Из конструктивного способа построения сети N следует, что структура сети определяется неоднозначно, так как условие

$$(P_{i1} * P_{i2} * \dots * P_{iR}) \leq TU_i, \quad (3)$$

определяющее компонентные автоматы $KA_{i1}, KA_{i2}, \dots, KA_{iR}$, может выполняться при различных сочетаниях разбиений из множества $\{P_i\}$, где $i = \overline{1, n}$.

В работе [3] введено понятие структурной матрицы сети N , под которой понимается квадратная матрица M_n размерности n , столбцам

матрицы M_n соответствуют разбиения $TU_{i1}, TU_{i2}, \dots, TU_{iN}$, а строкам — разбиения $P_{i1}, P_{i2}, \dots, P_{iN}$. Понятие матрицы M_n имеет важное значение при определении структуры сети N с минимальным числом связей. Формирование матрицы M_n происходит следующим образом: в i -й столбец записываются единицы в строки с индексами $i1, i2, \dots, iR$, если и только если справедливо условие (3). Если обозначить элемент матрицы $m[i, j]$, то компонентный автомат KA_i влияет на поведение KA_j (например, если $i < j$, то выход KA_i соединен со входом KA_j через функцию f_j).

Кроме того, структурная матрица M_n позволяет получить различные виды декомпозиции модели, т.е. параллельную или последовательную декомпозицию устройства.

Программа, реализующая решение поставленной задачи, написана с использованием структурного и объектно-ориентированного подходов. Язык программирования Object Pascal позволяет в полной мере использовать всю мощь этих подходов.

Конструктивный метод декомпозиции проиллюстрируем примером декомпозиции автомата $N = (Z, \{KA_i\}, W, \{f_i\}, \{PS_i\}, g)$, где $\{PS_i\} : Z * \{KA_i\} \rightarrow Z$; $g : Z * \{KA_i\} \rightarrow W$.

Пусть выбор начальных разбиений осуществлен следующим образом: $\Pi_1. \Pi_2. \Pi_3 = \Pi(0)$.

Поставим в соответствие каждому разбиению Π_i функцию $F_i, Z * \{KA_i\} \rightarrow \Pi_i$, такую, что $F_i(Z_m, \{KA_i\}_f) = \Pi_i(\{PS_i\}(Z_m, \{KA_i\}_f))$.

Значение функции F_i на паре $(Z_m, \{KA_i\}_f)$ равно блоку Π_i , в котором содержится состояние $Z_s = \{PS_i\}(Z_m, \{KA_i\}_f)$, $Z_m, Z_s \in Z, \{KA_i\}_f \in \{KA_i\}$.

Для блоков Π_1, Π_2 и Π_3 введем следующие обозначения: $\Pi_1 = \{b_1, b_2\} = B, \Pi_2 = \{c_1, c_2\} = C, \Pi_3 = \{d_1, d_2\} = D; F_1 : Z * \{KA_i\} \rightarrow B, F_2 : Z * \{KA_i\} \rightarrow C, F_3 : Z * \{KA_i\} \rightarrow D$.

Пусть $\Pi_1. \Pi_2. \Pi_3 = H, H = B.C.D, H = \{(b_1, c_1, d_1), (b_1, c_1, d_2), (b_1, c_2, d_1), (b_1, c_2, d_2), (b_2, c_1, d_1), (b_2, c_1, d_2), (b_2, c_2, d_1), (b_2, c_2, d_2)\}$. В соответствии с реализованным конструктивным методом декомпозиции выбранного устройства получим следующие компонентные автоматы: KA_1, KA_2, KA_3 .

Результаты исследований показали, что в целом алгоритм декомпозиции сложных дискретных устройств, основанный на определении сильносвязанных подсетей, имеет достаточную эффективность. Анализ результатов подтвердил, что при достаточной сложности и определенной топологии системы алгоритм уменьшает время анализа процесса функционирования такого класса устройств, причем также уменьшается общая вероятность ошибки в системе в силу подробного просмотра и анализа каждой подструктуры.

СПИСОК ЛИТЕРАТУРЫ

1. П и т е р с о н Д ж. Теория сетей Петри и моделирование систем: Пер. с англ. – М.: Мир, 1984. – 264 с.
2. Б а р а н о в С. И. Синтез микропрограммных автоматов (граф-схемы и автоматы). – Л.: Энергия, 1979. – 232 с.
3. М а г р у п о в Т. М. Графы, сети, алгоритмы и их приложения. – Ташкент: Фан, 1990, – 120 с.
4. К о т о в В. Е. Сети Петри. – М.: Наука, 1984, – 160 с.

Статья поступила в редакцию 26.09.2005

Игорь Владимирович Рудаков родился в 1958 г., окончил в 1981 г. МВТУ им. Н.Э.Баумана. Канд. техн. наук, доцент кафедры “Программное обеспечение ЭВМ и информационные технологии” МГТУ им. Н.Э.Баумана. Автор 50 научных работ в области исследования и разработки вычислительных устройств и программного обеспечения.

I.V. Rudakov (b. 1958) graduated from the Bauman Moscow Higher Technical School in 1981. Ph. D. (Eng.), assoc. professor of “Computer Software and Information Technology” department of the Bauman Moscow State Technical University. Author of 50 publications in the field of study and development of computing devices and software.



Марьям Давудпур родилась в 1974 г., окончила в 1999 г. Тегранский технический университет. Аспирантка кафедры “Программное обеспечение ЭВМ и информационные технологии” МГТУ им. Н.Э. Баумана. Автор 7 научных работ в области исследования и разработки вычислительной техники и программного обеспечения.

Maryam Davoudpour (b. 1974) graduated from the Tehran Central Technical University in 1999. Ph. D Candidate of “Computer Software and Information Technology” department of the Bauman Moscow State Technical University. Author of 7 publications in the field of study and development of computing devices and software.