ПРОГРАММНО-
ТЕХНИЧЕСКИЕ
КОМПЛЕКСЫ

В.Н. ОПАНАСЕНКО, С.Л. КРЫВЫЙ

УДК 51.681.3

СИНТЕЗ АДАПТИВНЫХ ЛОГИЧЕСКИХ СЕТЕЙ НА ОСНОВЕ ПОЛИНОМА ЖЕГАЛКИНА

Аннотация. Рассмотрена задача адаптации логической сети на основе универсальных логических элементов для реализации задачи классификации входного множества двоичных векторов. Адаптация состоит в определении типов логических функций для составных компонентов логической сети посредством описания ее полиномом, коэффициенты которого задаются матрицей Адамара или полиномом Жегалкина.

Ключевые слова: адаптация, булева функция, универсальный логический элемент, полином.

ВВЕДЕНИЕ

Широкий спектр задач классификации требует адаптации (реконфигурации) структуры под заданный алгоритм функционирования [1–4]. Появление кристаллов ПЛИС типа FPGA [5], которые представляют функциональное поле универсальных логических элементов (ЛЭ), позволило решить вопросы аппаратной реализации алгоритмов путем конфигурации структуры кристалла на выполнение требуемого алгоритма, описанного в работе [6].

Относительно топологии [7] адаптивная логическая сеть (АЛС) представляет собой матрицу логических элементов, которые группируются в функциональные узлы (ФУ) и функциональные блоки (ФБ), местоположение которых закреплено, при этом изменение их функционирования происходит в зависимости от класса задач и их назначения.

Универсальным ЛЭ будем называть комбинационный автомат $L = \langle n, F \rangle$, где n — количество двоичных входов или размерность входных переменных ЛЭ; $F = \{f_\rho\}$, $\rho = [1 \div 2^{2^n}]$, — множество булевых функций, реализуемых ЛЭ.

Универсальность ЛЭ заключается в возможности его настройки на реализацию произвольной булевой функции. Для случая $n = 2$ ЛЭ реализует одну из 16 логических функций, представляющих полный (базовый) набор функций двух переменных:

$$f_1 = a + b;\ f_2 = a + \bar{b};\ f_3 = \bar{a} + b;\ f_4 = \bar{a} + \bar{b};\ f_5 = a \,\&\, b;$$

$$f_6 = a \,\&\, \bar{b};\ f_7 = \bar{a} \,\&\, b;\ f_8 = \bar{a} \,\&\, \bar{b};\ f_9 = a \oplus b;\ f_{10} = a \oplus \bar{b};$$

$$f_{11} = a;\ f_{12} = b;\ f_{13} = \bar{a};\ f_{14} = \bar{b};\ f_{15} = 0;\ f_{16} = 1.$$

Структура АЛС может быть описана системой $A = \langle n, h, F, S, L, m, D, X, Y \rangle$, где n — разрядность входных двоичных векторов (размерность АЛС по входу); h — выходная разрядность, $h = \overline{1, n}$ (размерность АЛС по выходу); $F = \{F_{ij}\}$ — множество логических функций системы; S — структура связей между ЛЭ;

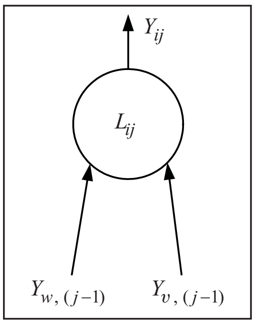

Рис. 1. Структура универсального ЛЭ

$L=\{L_{ij}\}$ — множество ЛЭ (i — порядковый номер ЛЭ; j — номер уровня обработки); m — количество уровней обработки; $D=\{d\}$ — множество n-мерных двоичных векторов — обучающая выборка; X — полное множество входных двоичных векторов; $Y=\{Y_{ij}\}$ — функция всей сети, $Y_{ij}=f_{ij}(Y_{v,\,(j-1)}, Y_{w,\,(j-1)})$ — значение функции f_{ij}, реализуемой элементом L_{ij}, структура которого приведена на рис. 1 (v, w — значение индекса i для входов ЛЭ).

В дальнейшем ограничимся рассмотрением АЛС типа «треугольная матрица» (ТМ) — это ФБ, состоящий из l функциональных узлов, для $h=1$, различаемых по топологическому признаку [8]. ФУ представляет собой комбинационный автомат без памяти, имеющий k-разрядный вход, u-разрядный выход и m строк матрицы. В рамках одного уровня тип функции задается для каждого ЛЭ в отдельности.

Базовый набор логических функций определяется размерностью универсального ЛЭ и представляет полный набор логических функций для заданного количества входных переменных (двоичных векторов).

ПОСТАНОВКА ЗАДАЧИ

Задача построения логической сети сводится к задаче определения необходимого набора функций, как суперпозиции элементарных базовых функций и формирования вложенных функциональных блоков.

Функциональный блок типа ТМ [8] предназначен для разбиения полного множества n-разрядных векторов X на два подмножества векторов: X_1 и X_2, заданных посредством обучающей выборки $D\subseteq X_1$ ($X_1\bigcup X_2=X$; $X_1\bigcap X_2=\varnothing$). ТМ реализует отображение $\Im: X\to Y$ такое, что $\Im: X_1\to 1$, $\Im: X_2\to 0$.

Структурно ТМ [9] представляет собой m-уровневую иерархическую матрицу. Под уровнем понимается ФУ, каждый ЛЭ которого настраивается на реализацию произвольной булевой функции и реализует отображение l-мерных двоичных векторов ($l=\overline{2,\,n}$) в u-мерные ($l\geq u$) векторы. Задача настройки (адаптации) ТМ формулируется следующим образом. Пусть имеется полное множество n-мерных двоичных векторов $X=\{x_p\}$, где $p=\overline{1,\,2^n}$, и задано множество n-мерных двоичных векторов $D\subset X$, которое является обучающей выборкой для алгоритма классификации. Для произвольного входного множества n-мерных двоичных векторов $G=\{g\}$ ($G\subset X$) необходимо реализовать следующую функцию:

$$Y(g)=\begin{cases}1 \;\; (\forall g\in D),\\ 0 \;\; (\forall g\notin D).\end{cases} \qquad (1)$$

В общем случае задача адаптации структуры ТМ на реализацию функции (1) сводится к задаче построения универсального логического элемента произвольной разрядности на основе ЛЭ фиксированной разрядности и состоит в определении структуры связей S и типов логических функций f_{ij} для этих ЛЭ, что в совокупности реализует отображение $\Psi: X\to Y$.

На основании (1) рассмотрим ТМ с сотовой структурой связи S (рис. 2), которая имеет следующие характеристики: $n=3$, $m=(n-1)$; $j=\overline{1,\,(n-1)}$, $i=\overline{1,\,(n-j)}$; $v=i$, $w=(i+1)$;

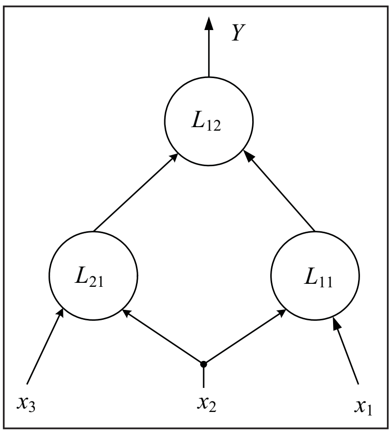

Рис. 2. ТМ с сотовой структурой связи

Card $\{L_{ij}\} = (n^2 - n)/2$.

ПЕРВЫЙ МЕТОД РЕШЕНИЯ ПРЯМОЙ ЗАДАЧИ СИНТЕЗА

Для определения множества логических функций $F = \{f_{ij}\}$ можно применять подход [10], использующий полиномы для структурного описания ТМ. Математическая модель (1) описывается булевой сетью, которая представляется многоуровневой комбинационной схемой, а вершинам сети соответствуют логические элементы.

В общем случае задача синтеза структуры ТМ сводится к определению типов логических функций f_{ij} для всех ЛЭ сети. Для определения множества логических функций $F = \{f_{ij}\}$ будем использовать полиномы для описания булевой сети [9, 10].

При кодировании значений булевой функции и ее аргументов перейдем к другому кодированию, используя значения 1 и –1. Таким образом, множество переменных $X = \{x_1, x_2, \ldots, x_n\}$ для булевой функции f от n переменных будет представляться множеством $E = \{e_1, e_2, \ldots, e_n\}$, где $e_i = (-1)^{x_i}$, а множество значений $Y = \{y_1, y_2, \ldots, y_{2^n-1}\}$, где $y_j = \{0, 1\}$, представится множеством $V = \{v_0, v_1, \ldots, v_{2^n-1}\}$, где $v_j = (-1)^{y_j}$.

Для любой булевой функции f от n переменных, принимающих значения из множества $\{1, -1\}$, существует эквивалентный полином $P_{f(n)}$ с коэффициентами из множества действительных чисел [10]: $f(X) = P_{f(n)}(X)$. Коэффициенты полинома для функции f запишем посредством матрицы Адамара

$$A_{2^n} = \frac{1}{2^n} H_n V_n,$$

где $A_{2^n} = \{a_0, a_1, \ldots, a_{2^n-1}\}$ — множество коэффициентов полинома; H_n — матрица Адамара размера $2^n \times 2^n$; V_n — множество значений булевой функции.

Матрица Адамара n-го порядка H_n представляет собой квадратную матрицу размера n, содержащую два типа элементов: 1 и –1. Матрицу Адамара можно построить для любого значения n:

$$H_0 = \|1\|;\; H_1 = \begin{Vmatrix} +1 & +1 \\ +1 & -1 \end{Vmatrix};\; H_n = \begin{Vmatrix} +H_{n-1} & +H_{n-1} \\ +H_{n-1} & -H_{n-1} \end{Vmatrix}.$$

Функция от одной переменной представляется полиномом $P_{f(1)} = a_0 + a_1 e_1$. Функции от двух и трех переменных представляются полиномами

$$P_{f(2)} = a_0 + a_1 e_1 + a_2 e_2 + a_3 e_1 e_2;$$
$$P_{f(3)} = a_0 + a_1 e_1 + a_2 e_2 + a_3 e_1 e_2 + a_4 e_3 + a_5 e_1 e_3 + + a_6 e_2 e_3 + a_7 e_1 e_2 e_3. \quad (2)$$

Соответственно полином от n переменных имеет вид

$$P_{f(n)} = P_{f(n-1)} + a_{n+1} e_n P_{f(n-1)}.$$

Пример 1. Рассмотрим прямую задачу синтеза параметрического модуля ТМ со следующими параметрами: размерность двоичных векторов ($n = 3$) и обучающая выборка $D = \{(1, 1, 1); (1, -1, 1); (-1, 1, -1); (-1, -1, 1)\}$. Таким образом, имеем трехвходовую ТМ с элементами $L_{1,1}, L_{1,2}, L_{2,1}$. Необходимо определить множество логических функций $\{f_{1,1}, f_{2,1}, f_{1,2}\}$ путем определения множеств коэффициентов полинома (2). На основе исходных данных (обучающая выборка D) имеем следующие значения для v_j:

$$v_0 = -1;\; v_1 = 1;\; v_2 = -1;\; v_3 = 1;\; v_4 = 1;\; v_5 = -1;\; v_6 = -1;\; v_7 = 1.$$

Таблица 1

Аргументы		Значение функции f_i															
x_0	x_1	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}	f_{16}
0	0	+ 1	− 1	− 1	− 1	+ 1	+ 1	+ 1	− 1	+ 1	− 1	+ 1	+ 1	− 1	− 1	+ 1	− 1
0	1	− 1	+ 1	− 1	− 1	+ 1	+ 1	− 1	+ 1	− 1	+ 1	+ 1	− 1	− 1	+ 1	+ 1	− 1
1	0	− 1	− 1	+ 1	− 1	+ 1	− 1	+ 1	+ 1	− 1	+ 1	− 1	+ 1	+ 1	− 1	+ 1	− 1
1	1	− 1	− 1	− 1	+ 1	− 1	+ 1	+ 1	+ 1	+ 1	− 1	− 1	− 1	+ 1	+ 1	+ 1	− 1

Таблица 2

Номер варианта	Типы логических функций		
	$f_{1,1}$	$f_{2,1}$	$f_{1,2}$
1	$e_1 \oplus e_2$	$\overline{e_2} \& \overline{e_3}$	$y_{1,1} \oplus y_{1,2}$
2	$\overline{e_1} \oplus e_2$	$e_2 + e_3$	$y_{1,1} \oplus y_{1,2}$
3	$\overline{e_1}$	$\overline{e_2} \& e_3$	$y_{1,1} \oplus y_{1,2}$
4	e_1	$e_2 + \overline{e_3}$	$y_{1,1} \oplus y_{1,2}$

Для получения коэффициентов полинома необходимо решить систему из восьми уравнений с двенадцатью неизвестными. В данном случае получаем четыре решения. В соответствии с полученными результатами из таблицы истинности логических функций (табл. 1) определяются непосредственно функции. Результаты возможных вариантов (четырех) настройки структуры (типы логических функций $\{f_{1,1},\ f_{1,2},\ f_{2,1}\}$ для логических элементов $L_{1,1}, L_{1,2}, L_{2,1}$) треугольной матрицы, представленной на рис. 2, приведены в табл. 2. В общем случае (для n переменных) при использовании этого метода необходимо решать систему из 2^n линейных уравнений с $2(n^2 - n)$ неизвестными.

Метод решения систем уравнений для определения полинома (2) выполняется одним из модифицированных алгоритмов, описанных в работах [11, 12].

ВТОРОЙ МЕТОД РЕШЕНИЯ ПРЯМОЙ ЗАДАЧИ СИНТЕЗА

Определить полином $P_{f(3)}$ можно и другим способом — с помощью решения системы линейных неоднородных диофантовых уравнений (СЛНДУ) в поле вычетов F_2 по модулю 2. В общем случае для решения СЛДУ в поле F_p, где p — простое число, был разработан алгоритм, подробно описанный в работах [11, 12]. Поэтому здесь приведем необходимые свойства этого алгоритма, который назван TSS-алгоритмом.

Теорема 1. TSS-алгоритм решения СЛНДУ строит общее решение этой системы в виде

$$x = x^1 \oplus \sum_{i=1}^{k} x_i,$$

где x^1 — частное решением СЛНДУ, а x_i — базисные решения системы линейных однородных диофантовых уравнений (СЛОДУ), соответствующей данной СЛНДУ, за время, пропорциональное величине $m \times n$, где m — число уравнений, а n — число неизвестных.

Построение полинома $P_{f(3)}$ состоит в том, что по заданной обучающей выборке для него строится СЛНДУ, решения которой дают различные варианты синтеза узлов структуры. При этом данный метод, который назовем волновым, можно усовершенствовать и распространить на общий случай с n входами структуры для заданного способа соединений.

Перейдем к описанию второго метода решения прямой задачи синтиеза на примере обучающей выборки, приведенной в примере 1.

Пример 2. Представим выражение (2) для трех переменных в виде полинома Жегалкина для сотовой структуры связей:

$$P_{f(3)} = (\underbrace{a_0 + a_1 e_1 + a_2 e_2 + a_3 e_1 e_2}_{L_{11}}) + (\underbrace{a_4 + a_5 e_2 + a_6 e_3 + a_7 e_2 e_3}_{L_{21}}) = L_{12}. \quad (3)$$

Обучающая выборка для L_{12} имеет вид

$$D = \{(0, 0, 0), (0, 1, 0), (1, 0, 1), (1, 1, 0)\},$$

где $x = (e_3, e_2, e_1) \in D$. По этой обучающей выборке строим СЛНДУ, подставляя элементы из D в полином $P_{f(3)}$, на которых он должен принимать значение 1:

$$S_1 = \begin{cases} a_0 + 0a_1 + 0a_2 + 0a_3 + a_4 + 0a_5 + 0a_6 + 0a_7 = 1, \\ a_0 + 0a_1 + a_2 + 0a_3 + a_4 + a_5 + 0a_6 + 0a_7 = 1, \\ a_0 + a_1 + 0a_2 + 0a_3 + a_4 + 0a_5 + a_6 + 0a_7 = 1, \\ a_0 + 0a_1 + a_2 + 0a_3 + a_4 + a_5 + a_6 + a_7 = 1. \end{cases} \quad (4)$$

В матричном виде эта СЛНДУ записывается как $Aa^t = b^t$, где $a^t = (a_0, a_1, \ldots, a_7)^t$, $b^t = (1, 1, \ldots, 1)^t$ — векторы-столбцы размера 8, а матрица A имеет вид

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

Частным решением S_1 есть вектор $x^1 = (1, 0, 0, 0, 0, 0, 0, 0)$, а базисным решением системы линейных однородных диафантовых уравнений, которое соответствует S_1, есть векторы

$$x_1 = (1, 0, 0, 0, 1, 0, 0, 0), \; x_2 = (0, 0, 0, 1, 0, 0, 0, 0),$$
$$x_3 = (0, 0, 1, 0, 0, 1, 0, 0), \; x_4 = (0, 1, 0, 0, 0, 0, 1, 1).$$

Общее решение системы (4) принимает вид $x = x^1 \oplus d_1 x_1 \oplus d_2 x_2 \oplus d_3 x_3 \oplus$ $\oplus\, d_4 x_4$, где $d_i \in \{0, 1\}$, $i = 1, 2, 3, 4$.

Варьируя значениями коэффициентов d_i, найдем 16 возможных вариантов реализации вычислительной структуры, приведенной на рис. 2. Рассмотрим некоторые из них.

1. Для $d_1 = d_2 = d_3 = 0, d_4 = 1$ получаем решение $(1, 1, 0, 0, 0, 0, 1, 1)$, которому соответствует

$$L_{1,1} = (1 \oplus e_1) = \overline{e_1}, \; L_{2,1} = e_3 \oplus e_2 e_3 = \overline{e_2} e_3, \text{ т.е. } L_{1,2} = \overline{e_1} \oplus \overline{e_2} e_3.$$

2. Для $d_1 = d_2 = 0, d_3 = d_4 = 1$ получаем решение $(1, 1, 1, 0, 0, 1, 1, 1)$, которому соответствует

$$L_{1,1} = (1 + e_1 + e_2) = \overline{e_1} \oplus e_2, \; L_{2,1} = (e_2 + e_3 + e_2 e_3) = e_2 + e_3 \overline{e_2},$$

т.е. $L_{1,2} = (\overline{e_1} \oplus e_2) \oplus (e_2 \cup e_3)$.

3. Для $d_1 = d_2 = d_3 = d_4 = 1$ получаем решение $(0, 1, 1, 1, 1, 1, 1, 1)$, которому соответствует

$$L_{1,1} = e_1 \oplus e_2 \oplus e_1 e_2 = e_1 \cup e_2, \; L_{1,2} = 1 \oplus e_2 \oplus e_3 \oplus e_2 e_3 = \overline{e_2}\, \overline{e_3}.$$

4. Для $d_1 = 1, d_2 = d_3 = 0, d_4 = 1$ получаем решение $(0, 1, 0, 0, 1, 0, 1, 1)$, которому соответствует

$$L_{1,1} = e_1, \; L_{1,2} = 1 \oplus e_3 \oplus e_2 e_3 = 1 \oplus \overline{e_2} e_3 = e_2 \cup \overline{e_3}.$$

Не все полученные функции на элементах, не входящих в D, будут иметь значение 0. Например, для функции из п. 3 на элементе (0, 1, 1) получаем значение 1.

При условии, чтобы на всех выборках, не входящих в D, значение $P_{f(3)}$ равнялось нулю, получим СЛНДУ вида

$$S_2 = \begin{cases} a_0 + 0a_1 + 0a_2 + 0a_3 + a_4 + 0a_5 + 0a_6 + 0a_7 = 1, \\ a_0 + 0a_1 + a_2 + 0a_3 + a_4 + a_5 + 0a_6 + 0a_7 = 1, \\ a_0 + a_1 + 0a_2 + 0a_3 + a_4 + 0a_5 + a_6 + 0a_7 = 1, \\ a_0 + 0a_1 + a_2 + 0a_3 + a_4 + a_5 + a_6 + a_7 = 1, \\ a_0 + a_1 + a_2 + a_3 + a_4 + a_5 + a_6 + a_7 = 0, \\ a_0 + a_1 + 0a_2 + 0a_3 + a_4 + 0a_5 + 0a_6 + 0a_7 = 0, \\ a_0 + a_1 + a_2 + a_3 + a_4 + a_5 + 0a_6 + 0a_7 = 0, \\ a_0 + 0a_1 + 0a_2 + 0a_3 + a_4 + 0a_5 + a_6 + 0a_7 = 0. \end{cases} \qquad (5)$$

С помощью TSS-алгоритма находим такие решения:

$x^1 = (1, 1, 0, 0, 0, 0, 1, 1)$ — частное решение S_2;

$$\left.\begin{array}{l} x_1 = (1, 0, 0, 0, 1, 0, 0, 0) \\ x_2 = (0, 0, 1, 0, 0, 1, 0, 0) \end{array}\right\} \text{ — базисные решения СЛОДУ для } S_2.$$

Общее решение имеет вид $x = x^1 \oplus d_1 x_1 \oplus d_2 x_2$.

В результате получим четыре комбинации:

$$d_1 = 1,\ d_2 = 0:\ y_1 = x^1 \oplus x_1 = (1, 1, 0, 0, 1, 0, 1, 1) \Rightarrow e_1 \oplus (1 \oplus e_3 \oplus e_2 e_3) = \\ = e_1 \oplus (e_2 \cup \overline{e_3});$$

$$d_1 = 0,\ d_2 = 1:\ y_2 = x^1 \oplus x_2 = (1, 1, 1, 0, 0, 1, 1, 1) \Rightarrow (1 \oplus e_1 \oplus e_2) \oplus \\ \oplus (e_2 \oplus e_3 \oplus e_2 e_3) = (\overline{e_1} \oplus e_2) \oplus (e_2 \cup e_3);$$

$$d_1 = d_2 = 1:\ y_3 = x_1 \oplus x_2 \oplus x_3 = (0, 1, 1, 0, 1, 1, 1, 1) = (e_1 \oplus e_2) \oplus \\ \oplus (1 \oplus (e_2 \cup e_3)) = (e_1 \oplus e_2) \oplus \overline{e_2}\,\overline{e_3};$$

$$d_1 = d_2 = 0:\ y_4 = x^1 = (1, 1, 0, 0, 0, 0, 1, 1) = (1 \oplus e_1) \oplus (e_3 \oplus e_2 e_3) = \overline{e_1} \oplus \overline{e_2} e_3.$$

Эти комбинации дают те же функции в узлах структуры, что и в примере 1 (см. табл. 2).

Корректность описанного способа вытекает из корректности TSS-алгоритма и корректности применяемых тождеств алгебры Жегалкина.

Заметим, что если обучающая выборка изменяется, то матрица системы S_2 не изменяется, а изменяются только свободные члены этой системы. Процесс решения остается тем же, но синтезируемые функции изменяются. Например, если $D = \{(0, 0, 0), (0, 1, 0), (1, 0, 1), (0, 1, 1)\}$, то S_2 принимает вид

$$Aa = (1, 1, 1, 0, 0, 0, 1, 0)^t,$$

где A — матрица S_2.

Решения системы уравнений имеют вид $x^1 = (1, 1, 0, 1, 0, 0, 1, 0)$, $x_1 = (1, 0, 0, 0, 1, 0, 0, 0)$, $x_2 = (0, 0, 1, 0, 0, 1, 0, 0)$, т.е. общим решением будет $x = x^1 \oplus d_1 x_1 \oplus d_2 x_2$.

Тогда, если

1) $d_1 = d_2 = 1$, то $x = (0, 1, 1, 1, 1, 1, 1, 0) \Rightarrow$ $(e_1 \oplus e_2 \oplus e_1 e_2) \oplus (1 + e_2 \oplus e_3) = \overline{e_1} e_2 \oplus \oplus \overline{e_2} = (e_1 \cup e_2) \oplus (\overline{e_2} \oplus e_3)$;

2) $d_1 = 0$, $d_2 = 1$, то $x = (1, 1, 1, 1, 0, 1, 1, 0) \Rightarrow$ $(1 \oplus e_1 \oplus e_2 \oplus e_1 e_2) \oplus (e_2 \oplus e_3) =$ $= (1 \oplus (e_1 \cup e_2)) \oplus (e_2 \oplus e_3) = \overline{e_1}\,\overline{e_2} \oplus (e_2 \oplus e_3)$;

3) $d_1 = 1,\ d_2 = 0$, то $x = (0, 1, 0, 1, 1, 0, 1, 0) \Rightarrow (e_1 \oplus e_1 e_2) \oplus (1 + e_3) = e_1 \overline{e_2} \oplus \overline{e_3}$;

4) $x^1 (1, 1, 0, 1, 0, 0, 1, 0) \Rightarrow (1 + e_1 + e_1 e_2) \oplus e_3 = (1 + e_1 \overline{e_2}) \oplus e_3 = (\overline{e_1} \cup e_2) \oplus e_3$.

Если проанализировать базисные решения СЛОДУ, которая соответствует СЛНДУ, то первое решение добавляет две единицы к частному решению СЛДУ, а второе решение добавляет две переменные e_2. Это значит, что любые комбинации общего решения приводят в итоге к функции $e_1 \overline{e_2} \oplus \overline{e_3}$. Данный факт свидетельствует о том, что все найденные выражения для функции f_{12} должны быть эквивалентными, а это приводит к полиному Жегалкина $e_1 \overline{e_2} \oplus \overline{e_3}$, который является каноническим [11, 12].

ОБОБЩЕНИЕ ВОЛНОВОГО АЛГОРИТМА

Рассмотрим четырехвходовую структуру ТМ ($n = 4$), т.е. синтезируемая функция имеет вид

$$P_{f(4)} = (a_0 + a_1 e_1 + a_2 e_2 + a_3 e_1 e_2) + (b_0 + b_1 e_2 + b_2 e_3 + b_3 e_2 e_3) + \\ + (c_0 + c_1 e_3 + c_2 e_4 + c_3 e_3 e_4). \qquad (6)$$

Задача синтеза в этом случае усложняется, поскольку произвольно задавать выборку D невозможно ввиду наличия общего элемента $L_{2,1}$ в структуре. Возникает вопрос: как найти обучающую выборку, по которой синтезируется полином вида (6), если задана обучающая выборка D_3 для полинома (3)?

При этих условиях решение задачи синтеза существенно упрощается. Действительно, если известны выборка D_3 и синтезированный по ней полином, то значение полинома на наборах,входящих в D_3, будет всегда равно единице, а на наборах, не входящих в D_3, значение полинома равно нулю. Это обстоятельство дает возможность по выборке D_3 и ее дополнению — выборке $\overline{D}_3$ построить выборки D_4 и $\overline{D}_4$, а также построить непосредственно полином $P_{f(4)}$, не прибегая к решению СЛНДУ, построенной по выборке D_4. Действительно, выборка D_4 строится по выборкам D_3 и $\overline{D}_3$ добавлением четвертой компоненты, равной нулю, к каждому элементу выборки D_3 и компоненты, равной единице, к каждому элементу выборки $\overline{D}_3$. А полином $P_{f(4)}$ принимает вид $P_{f(4)} = P_{f(3)} \oplus e_4$.

Рассмотрим пример.

Пример 3. Представим обучающую выборку $D_3 = \{(0, 0, 0), (0, 1, 0), (1, 0, 1), (1, 1, 0)\}$, рассмотренную в примере 1. Выборка $\overline{D}_3 = \{(1, 0, 0), (0, 0, 1), (1, 1, 1), (0, 1, 1)\}$ включает наборы, на которых полином $P_{f(3)}$ принимает значение 0. По выборке D_3 был синтезирован полином $P_{f(3)} = \overline{e_1} \oplus \overline{e_2} e_3$. По выборкам D_3 и $\overline{D}_3$ строим выборки D_4 и $\overline{D}_4$:

$$D_4 = \{(0, 0, 0, 0), (0, 0, 1, 0), (0, 1, 0, 1), (0, 1, 1, 0), (1, 1, 0, 0), (1, 0, 0, 1), (1, 1, 1, 1), (1, 0, 1, 1)\},$$

$$\overline{D}_4 = \{(1, 0, 0, 0), (1, 0, 1, 0), (1, 1, 0, 1), (1, 1, 1, 0), (0, 1, 0, 0), (0, 0, 0, 1), (0, 1, 1, 1), (0, 0, 1, 1)\}.$$

Полином $P_{f(4)}$принимает следующий вид:

$$P_{f(4)} = \overline{e_1} \oplus \overline{e_2} e_3 \oplus e_4. \qquad (7)$$

По выборкам D_4 и $\overline{D}_4$ аналогичным способом строятся выборки D_5 и $\overline{D}_5$:

$$D_5 = \{(0, 0, 0, 0, 0), (0, 0, 0, 1, 0), (0, 0, 1, 0, 1), (0, 0, 1, 1, 0), (0, 1, 1, 0, 0), (0, 1, 0, 0, 1), \\ (0, 1, 1, 1, 1), (0, 1, 0, 1, 1), (1, 1, 0, 0, 0), (1, 1, 0, 1, 0), (1, 1, 1, 0, 1), (1, 1, 1, 1, 0), \\ (1, 0, 1, 0, 0), (1, 0, 0, 0, 1), (1, 0, 1, 1, 1), (1, 0, 0, 1, 1)\};$$

$$\overline{D}_5=\{(1,0,0,0,0),(1,0,0,1,0),(1,0,1,0,1),(1,0,1,1,0),(1,1,1,0,0),(1,1,1,1,1),$$
$$(1,1,0,0,1),(1,1,0,1,1),(0,1,0,0,0),(0,1,0,1,0),(0,1,1,0,1),(0,1,1,1,0),$$
$$(0,0,1,0,0),(0,0,0,0,1),(0,0,1,1,1),(0,0,0,1,1)\}.$$

Тогда полином $P_{f(5)}$ примет вид

$$P_{f(5)}=\overline{e_1}\oplus\overline{e_2}e_3\oplus e_4\oplus e_5.$$

ОБОСНОВАНИЕ АЛГОРИТМА СИНТЕЗА

Для обоснования алгоритма синтеза необходимо показать, что синтезированная структура ТМ корректно реализует классификацию. Другими словами, на всех наборах, входящих в обучающую выборку, на выходе ТМ формируется значение 1, а на наборах, не входящих в обучающую выборку, на выходе ТМ генерируется значение 0.

Докажем корректность алгоритма для четырехвходовой структуры ТМ с синтезированной функцией $P_{f(4)}=\overline{e_1}\oplus\overline{e_2}e_3\oplus e_4$ для обучающей выборки D_4. Убедимся в том, что выборка D_4 дает функцию (7) с помощью решения СЛНДУ, построенной по этой выборке. В этом случае СЛНДУ для полинома

$$P_{f(4)}=(a_0+a_1e_1+a_2e_2+a_3e_1e_2)+(b_0+b_1e_2+b_2e_3+b_3e_2e_3)+$$
$$+(c_0+c_1e_3+c_2e_4+c_3e_3e_4)$$

имеет следующий вид:

$$S=\begin{cases}
a_0+0a_1+0a_2+0a_3+b_0+0b_1+0b_2+0b_3+c_0+0c_1+0c_2+0c_3=1,\\
a_0+0a_1+a_2+0a_3+b_0+b_1+0b_2+0b_3+c_0+0c_1+0c_2+0c_3=1,\\
a_0+a_1+0a_2+0a_3+b_0+0b_1+b_2+0b_3+c_0+c_1+0c_2+0c_3=1,\\
a_0+0a_1+a_2+0a_3+b_0+b_1+b_2+b_3+c_0+c_1+0c_2+0c_3=1,\\
a_0+a_1+a_2+a_3+b_0+b_1+b_2+b_3+c_0+c_1+c_2+c_3=1,\\
a_0+a_1+0a_2+0a_3+b_0+0b_1+0b_2+0b_3+c_0+c_1+0c_2+0c_3=1,\\
a_0+0a_1+0a_2+0a_3+b_0+0b_1+b_2+0b_3+c_0+c_1+c_2+c_3=1,\\
a_0+a_1+a_2+a_3+b_0+b_1+0b_2+0b_3+c_0+c_1+0c_2+0c_3=1,\\
a_0+0a_1+0a_2+0a_3+b_0+0b_1+0b_2+0b_3+c_0+c_1+0c_2+0c_3=0,\\
a_0+0a_1+a_2+0a_3+b_0+b_1+0b_2+0b_3+c_0+0c_1+c_2+0c_3=0,\\
a_0+a_1+0a_2+0a_3+b_0+0b_1+b_2+0b_3+c_0+c_1+c_2+c_3=0,\\
a_0+0a_1+a_2+0a_3+b_0+b_1+b_2+b_3+c_0+c_1+c_2+c_3=0,\\
a_0+0a_1+0a_2+0a_3+b_0+0b_1+b_2+0b_3+c_0+c_1+0c_2+0c_3=0,\\
a_0+a_1+a_2+a_3+b_0+b_1+b_2+b_3+c_0+c_1+0c_2+0c_3=0,\\
a_0+a_1+a_2+a_3+b_0+b_1+0b_2+0b_3+c_0+0c_1+0c_2+0c_3=0,\\
a_0+a_1+0a_2+0a_3+b_0+0b_1+0b_2+0b_3+c_0+0c_1+0c_2+0c_3=0.
\end{cases}$$

Решениями этой системы являются такие векторы:

$x^1=(1100,0001,0110)$ — частное решение СЛНДУ;

$x_1=(1000,0000,1000)$, $x_2=(1000,1000,0000)$, $x_3=(0010,0100,0000)$, $x_4=$ $=(0000,0100,0100)$ представляют базисные решения СЛОДУ.

Общее решение имеет вид $x=x^1\oplus d_1x_1\oplus d_2x_2\oplus d_3x_3\oplus d_4x_4$.

Все 16 решений данной СЛНДУ в итоге дают функцию (7). Для примера рассмотрим две комбинации:

$$x^1=(1100,0001,0110)\Rightarrow(1\oplus e_1)\oplus e_2e_3\oplus e_3\oplus e_4=\overline{e_1}\oplus\overline{e_2}e_3\oplus e_4;$$

$$x^1\oplus x_4=(1100,0011,0010)\Rightarrow(1\oplus e_1)+(e_3\oplus e_2e_3)+e_4=\overline{e_1}\oplus\overline{e_2}e_3\oplus e_4.$$

При необходимости можно проверить соответствие и на остальных комбинациях. Аналогичным образом доказывается корректность алгоритма для произвольного количества входов ТМ.

СРАВНИТЕЛЬНЫЕ ОЦЕНКИ СЛОЖНОСТИ

Относительно сложностных оценок вначале заметим, что для TSS-алгоритма соответствующая СЛНДУ решается единственный раз для заданной обучающей выборки размерности $n=3$. Поэтому можно считать, что этот шаг в алгоритме синтеза выполняется за временную константу. Остальные шаги алгоритма сводятся к построению по заданному логическому выражению его отрицания с соответствующими подстановками переменных, продиктованными симметрией. Таким образом, второй способ синтеза адаптивной логической сети типа ТМ с сотовой структурой связи более эффективен, чем первый. В первом способе необходимо многократно решать СЛНДУ с 2^n уравнениями и $2(n^2-n)$ неизвестными, в результате получаем экспоненциальную сложность алгоритма.

ЗАКЛЮЧЕНИЕ

Рассмотрен подход к синтезу адаптивных структур, представленных многоуровневыми логическими схемами и описанных булевой сетью в виде ациклического графа, вершинами которого являются универсальные логические элементы. Синтез таких структур состоит в определении типов логических функций вершин графа при заданной обучающей выборке двоичных векторов, что позволяет использовать эту структуру для задачи классификации входных векторов. В отличие от известных методов синтеза многоуровневых логических схем в настоящей статье рассмотрены два подхода к синтезу таких схем. Первый основан на описании булевой сети полиномами Жегалкина, коэффициенты которого задаются посредством матрицы Адамара. Второй подход основан на применении алгоритма решения СЛНДУ в поле вычетов по модулю 2 с тем же описанием булевой сети. Второй подход можно обобщить на случай структур общего вида с n входами.

СПИСОК ЛИТЕРАТУРЫ

1. Palagin A.V., Opanasenko V.N. Reconfigurable computing technology // Cybernetics and Systems Analysis. — 2007. — **43**, N 5. — P. 675–686.
2. Куссуль Н.М., Шелестов А.Ю., Лавренюк А.М. Інтелектуальні обчислення. — Київ: Наук. думка, 2006. — 186 с.
3. Kondratenko Y.P., Gordienko E. Implementation of the neural networks for adaptive control system on FPGA // Proc. of 23rd DAAAM Intern. Symp. on Intelligent Manufacturing and Automation. — 2012. — **23**, N 1. — P. 389–392.
4. Palagin A., Opanasenko V., Krivoi S. The structure of FPGA-based cyclic-code converters // Optical Memory & Neural Networks (Information Optics). — 2013. — **22**, N 4. — P. 207–216.
5. Palagin A.V., Opanasenko V.N. Design and application of the PLD-based reconfigurable devices // Design of Digital Systems and Devices. — Berlin; Heidelberg: Springer-Verlag, 2011. — **79.** — P. 59–91.
6. Opanasenko V.N., Kryvyi S.L. Partitioning the full range of boolean functions based on the threshold and threshold relation // Cybernetics and Systems Analysis. — 2012. — **48**, N 3. — P. 459–468.
7. Палагин А.В., Опанасенко В.Н. Реконфигурируемые вычислительные системы. — Київ: Просвіта, 2006. — 295 с.
8. Опанасенко В.Н. Реконфигурируемые структуры типа «треугольная матрица» // Технології створення перспективних комп'ютерних засобів та систем з використанням новітньої елементної бази: Зб. наукових праць НАН України Ін-ту кібернетики ім. В.М. Глушкова. — Київ, 2000. — С. 31–35.
9. Опанасенко В.Н. Синтез параметрического модуля многоуровневой комбинационной логической схемы // Математические машины и системы. — 2001. — № 1, 2. — С. 34–39.
10. Bruck J., Blaum M. Neural networks, error-correcting codes, and polynomials over the binary n-cube // IEEE Transactions on Information Theory. — 1989. — **35**, N 5. — P. 976–987.

11. K r y v y i S . L . Algorithms for solving systems of linear Diophantine equations in integer domains // Cybernetics and Systems Analysis. — 2006. — **42**, N 2. — P. 163–175.
12. K r y v y i S . L . Algorithms for solving systems of linear Diophantine equations in residue fields // Cybernetics and Systems Analysis. — 2007. — **43**, N 2. — P. 171–178.

Поступила 29.12.2014