УДК 519.7

И.Д. Вечирская, З.В. Дударь, А.А. Иванилов, В.А. Лещинский

ЛИНЕЙНЫЕ ЛОГИЧЕСКИЕ ОПЕРАТОРЫ В ВИДЕ СХЕМ И ГРАФОВ

В [1] был обозначен подход к построению высокопроизводительных ЭВМ нового типа, основанных на параллельной архитектуре и ориентированных на обработку логических данных и естественно-языковой информации. Для ЭВМ такого типа введен термин «мозгоподобная». Основой мозгоподобных ЭВМ призвана стать аппаратная реализация новой алгебрологической конструкции, называемой логической сетью. В [1] приведен пример логической сети, которая описана системой бинарных предикатов. Однако, чтобы с помощью сети можно было извлекать знания из описываемой ею модели, необходимо решать логические уравнения, выраженные бинарными предикатами. Сами по себе предикаты не позволяют этого делать, они лишь описывают модель.

Важным логико-математическим инструментом, позволяющим решать логические уравнения, являются линейные логические операторы [2, 3]. Когда в логическом уравнении с двумя неизвестными задается множество значений одной предметной переменной, то линейный логический оператор находит множество значений второй переменной [2, 3]. Поэтому функционирующая логическая сеть представляет собой систему взаимодействующих операторов. Таким образом, одним из базовых элементов мозгоподобной ЭВМ призвана стать аппаратная реализация линейного логического оператора.

В свете вышесказанного актуальным становится вопрос о преобразовании математической формулы линейного логического оператора в техническую схему. После этого логическую сеть можно реализовать в виде устройства, состоящего из взаимодействующих линейных логических операторов.

К настоящему времени уже имеется проект конкретной логической сети в виде описания такого устройства на языке VHDL. На кафедре автоматизации проектирования вычислительной техники Харьковского национального университета радиоэлектроники разработана специализированная интегральная схема логической сети, описывающей связь грамматических признаков с окончаниями полных непритяжательных имен прилагательных русского языка [4]. Данная логическая сеть представлена на рис. 1 в виде неориентированного графа.

Каждая дуга на рис. 1 разворачивается в двудольный граф, подобный изображенному на рис. 3. Далее на основе двудольного графа можно построить пару линейных логических операторов (рис. 4, 5) и реализовать их аппаратно. Вершины выражают следующие признаки имени прилагательного: y_2 – род; y_1 –число; y_3 – падеж; t – архаичность; y_4 – одушевленность; v – ячейка парадигмы; y – класс окончания; x_1 – первая буква окончания; x_2 – вторая буква окончания; x_3 – третья буква окончания; x – окончание; u_1 – ударность; u_2 – смягчение; u_4 – последняя буква основы; u – тип основы.

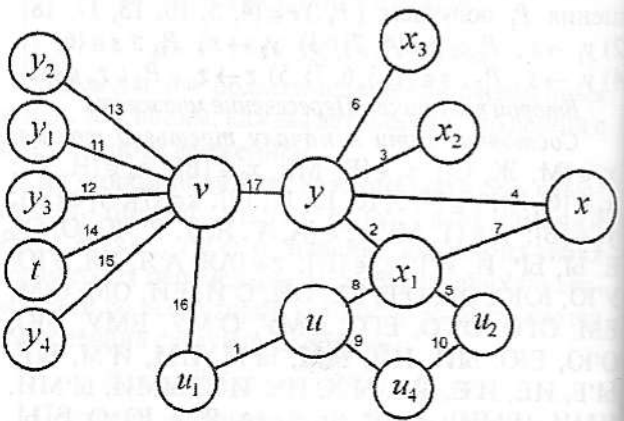


Рис. 1

Данная логическая сеть отвечает на различные запросы относительно прилагательных. Сеть может, например, решать задачу склонения: она установит окончание (или несколько окончаний) некоторого прилагательного, если указать все (или некоторые) его грамматические признаки. Сеть решает и обратную задачу, а также множество похожих задач, связанных с прилагательными.

Цель данной статьи – указать способ технической реализации линейного логического оператора. Средством математического описания при этом служит аппарат алгебры предикатов [5, 6, 7, 8], а средством технической реализации – комбинационные схемы, которые в [6] называются переключательными цепями.

Поскольку переключательная цепь линейного логического оператора служит удачным обобщением переключательной цепи многозначного отображения, сначала приведем математическое описание и реализацию для отображения, а затем – для оператора.

Пусть $M=\{a_1,a_2,...,a_k\}$ – фиксированное множество, состоящее из k элементов; A – какое-нибудь из его подмножеств $A \subseteq M$. Для множества A составляем набор $(\alpha_1,\alpha_2,...,\alpha_k)$ логических элементов $\alpha_1,\alpha_2,...,\alpha_k$ по следующему правилу: если

$\alpha_i \in A$, то $\alpha_i = 1$; если же $\alpha_i \notin A$, то $\alpha_i = 0$, $(i = \overline{1,k})$. Набор $(\alpha_1, \alpha_2, ..., \alpha_k)$ называется *характеристикой или координатным представлением* множества A, а логические элементы $\alpha_1, \alpha_2, ..., \alpha_k$ – *координатами* множества A. Пусть, например, $M = \{a,b,c,d\}$, $A = \{c,d\}$. Тогда характеристика множества A равна $(0,0,1,1)$. Можно по характеристике найти множество: если $M = \{a,b,c,d\}$ и характеристика A равна $(1,0,0,0)$, то $A = \{a\}$.

Произвольное отображение (в общем случае многозначное)

$$\underset{M_1 \times M_2 \times ... \times M_n \to N}{f(x_1, x_2, ..., x_n) = y}, \ \left(y \in N, N = \{b_1, b_2, ..., b_k\}\right)$$

ставит в соответствие набору значений аргументов $(x_1, x_2, ..., x_n)$ множество $B \subseteq N$, равное

$$B = \left\{ y \middle| F(x_1, x_2, ..., x_n, y) = 1 \right\},$$

где F – предикат, *соответствующий* отображению f или *характеристическая функция* отображения f (если n-местное отображение f рассматривать как $n+1$-местное отношение F). Пусть $\beta_1, \beta_2, ..., \beta_l$ – координаты множества B. Тогда

$$\beta_1 = F(x_1, x_2, ..., x_n, b_1),$$
$$\beta_2 = F(x_1, x_2, ..., x_n, b_2),$$
$$\cdots\cdots\cdots\cdots\cdots\cdots$$
$$\beta_l = F(x_1, x_2, ..., x_n, b_l).$$

Переключательную цепь удобно строить, когда отображение приведено к данному виду. Приведем пример построения схемы, реализующей какое-нибудь отображение. Берем предикат, связывающий род с первой буквой трехбуквенных окончаний полных непритяжательных имен прилагательных:

$$F(x,y) = x^а y^{ж} \vee x^{я} y^{ж} \vee x^{у} y^{ж} \vee x^{ю} y^{ж} \vee x^{о} y^{ж} \vee x^{о} y^{м} \vee x^{о} y^{с} \vee x^{е} y^{ж} \vee x^{е} y^{м} \vee x^{е} y^{с} \vee x^{ы} y^{ж} \vee x^{ы} y^{м} \vee x^{ы} y^{с} \vee x^{и} y^{ж} \vee x^{и} y^{м} \vee x^{и} y^{с}. \quad (1)$$

Строим соответствующее ему отображение $f(x) = y$. Принимаем $M = \{а,я,у,ю,о,е,ы,и\}$, $N = \{ж,м,с\}$. Имеем:

$$\beta_{ж} = x^{у} \vee x^{ю} \vee x^{а} \vee x^{я} \vee x^{о} \vee x^{е} \vee x^{ы} \vee x^{и},$$
$$\beta_{м} = x^{о} \vee x^{е} \vee x^{ы} \vee x^{и}, \quad (2)$$
$$\beta_{с} = x^{о} \vee x^{е} \vee x^{ы} \vee x^{и}.$$

Схема, реализующая отображение f, имеет вид, показанный на рис. 2.

Схема находит полный образ предмета. Пусть, например $x = e$. Тогда $\beta_{ж} = 1$, $\beta_{м} = 1$, $\beta_{с} = 1$; $y \in \{ж,м,с\}$. То есть полным образом предмета e является все множество N. Для нахождения прообраза нужно строить схему, действующую в обратную сторону. Исходную и обратную схемы легко строить по двудольному графу. Узлы на входе графа заменяются разветвлением проводов. При переходе от исходной схемы к схеме, действующей в обратную сторону, элементы разделения заменяются разветвлениями проводов, а разветвления проводов – элементами разделения. Узлы на выходе графа превращаются в узловые точки схемы.

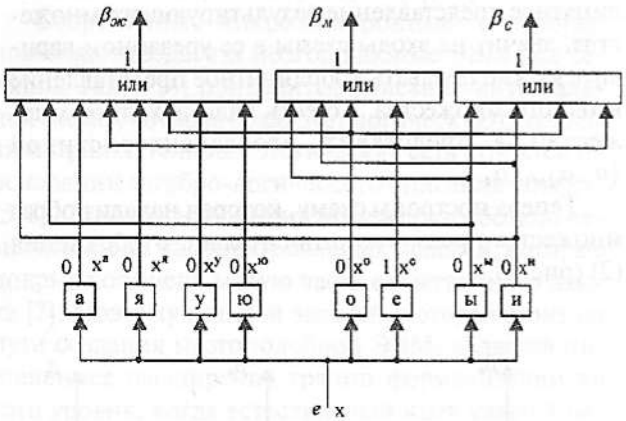
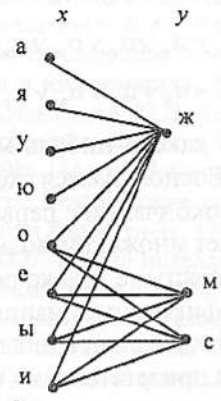


Рис. 2

Двудольный граф строится по бинарному отношению или предикату. Двудольный граф для бинарного предиката $F(x,y)$ изображен на рис. 3. По двудольному графу легко находить полный образ и прообраз предмета вручную. Для этого не нужно строить исходную и обратную переключательные цепи – одного графа достаточно. При этом можно заметить, что двудольный граф позволяет легко находить полный образ не только одного, но и нескольких предметов, то есть можно находить образ множества $A \subseteq M$. То же самое и в другую сторону: можно находить прообраз некоторого множества $B \subseteq N$.



F(x, y)

Рис. 3

Для того, чтобы находить образ множества, переключательная цепь, показанная на рис. 2, не годится. После прохождения входного сигнала через элементы узнавания предмета только по одному из проводов пойдет сигнал 1, по остальным – 0. Между тем, если «отрезать» всю нижнюю часть схемы с элементами узнавания предмета, то единичные сигналы можно будет пустить хоть по всем проводам. Полученная таким образом схема устроена так, что никакой коллизии не произойдет. В этом случае на входные двоичные коды схема будет реагировать выходными. Но на выходе мы регистрируем координатное представление результирующего множества, значит на входы схемы в ее урезанном варианте можно подавать координатное представление входного множества. То есть надо исходное множество A представить его характеристикой $(\alpha_1,\alpha_2,...,\alpha_k)$.

Теперь построим схему, которая находит образ множества предметов относительно отображения (2) (рис. 4).

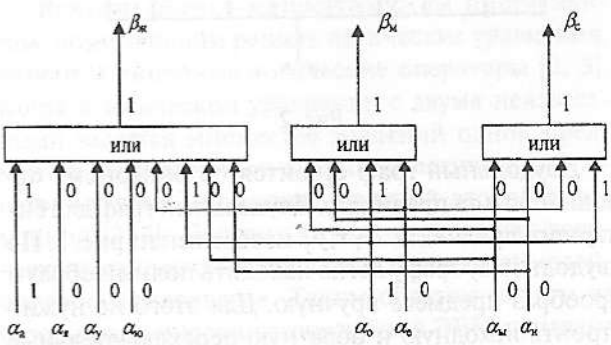


Рис. 4

Для того, чтобы описать эту переключательную цепь математически, достаточно в выражении (2) заменить $x^а$, $x^я$, $x^у$, $x^ю$, $x^о$, $x^е$, $x^ы$, $x^и$ на $\alpha_а$, $\alpha_я$, $\alpha_у$, $\alpha_ю$, $\alpha_о$, $\alpha_е$, $\alpha_ы$, $\alpha_и$ соответственно:

$$\begin{aligned}\beta_ж &= \alpha_а \vee \alpha_я \vee \alpha_у \vee \alpha_ю \vee \alpha_о \vee \alpha_е \vee \alpha_ы \vee \alpha_и,\\ \beta_м &= \alpha_о \vee \alpha_е \vee \alpha_ы \vee \alpha_и,\\ \beta_с &= \alpha_о \vee \alpha_е \vee \alpha_ы \vee \alpha_и.\end{aligned} \qquad (3)$$

Найдем образ какого-нибудь множества для отображения (2). Воспользуемся схемой на рис. 4. Берем множество окончаний с первой буквой *а* и *о*. Схема порождает множество возможных родов: $\{а,о\}\to\{ж,м,с\}$, то есть не только реальные окончания рода, но и фиктивные, например, буква *а* оказалась первой буквой окончания полных непритяжательных имен прилагательных среднего рода, хотя таковой на самом деле не является.

Теперь обратимся к линейным логическим операторам [2, 3]. В рассматриваемой ситуации *линейным логическим оператором* $L(A)=B$ является преобразование вида

$$B(y)=\exists x\in M\big(F(x,y)\wedge A(x)\big), \qquad (4)$$

которое преобразует одноместные предикаты $A(x)$, заданные на множестве M, в одноместные предикаты $B(y)$, заданные на множестве N. В выражении (4) бинарный предикат $F(x,y)$ фиксирован и задан на $M\times N$, он называется ядром оператора L; квантор существования означает логическую сумму.

Между одноместными предикатами и множествами существует взаимно однозначное соответствие. Предикат $A(x)$ на $M=\{a_1,a_2,...,a_k\}$ соответствующий множеству A с характеристикой $(\alpha_1,\alpha_2,...,\alpha_k)$, запишется формулой:

$$A(x)=\alpha_1x^{a_1}\vee\alpha_2x^{a_2}\vee...\vee\alpha_kx^{a_k}=\bigvee_{i=1}^{k}\alpha_ix^{a_i}.$$

Для предиката $A(x)$ набор $(\alpha_1,\alpha_2,...,\alpha_k)$ называется его *координатным представлением*. Таким образом, когда это удобно, под $A(x)$ можно понимать множество A.

Пользуясь координатным представлением предикатов, можно строить схемы линейных логических операторов. Пусть $M=\{a_1,a_2,...,a_k\}$. Тогда перепишем (4) в развернутом виде:

$$\begin{aligned}B(y)=F(a_1,y)A(a_1)\vee F(a_2,y)A(a_2)\vee...\vee\\ \vee F(a_k,y)A(a_k).\end{aligned}$$

Пусть $N=\{b_1,b_2,...,b_l\}$. Тогда (4) примет еще более развернутый вид:

$$\begin{aligned}B(b_1)&=F(a_1,b_1)A(a_1)\vee F(a_2,b_1)A(a_2)\vee...\\ &\vee F(a_k,b_1)A(a_k),\\ &...\\ B(b_l)&=F(a_1,b_l)A(a_1)\vee F(a_2,b_l)A(a_2)\vee...\\ &\vee F(a_k,b_l)A(a_k).\end{aligned} \qquad (5)$$

Пусть $(\alpha_1,\alpha_2,...,\alpha_k)$ – характеристика множества A, $A\subseteq M$; $(\beta_1,\beta_2,...,\beta_l)$ – характеристика множества B, $B\subseteq N$. Тогда

$$\begin{aligned}&\alpha_1=A(a_1),\alpha_2=A(a_2),...,\alpha_k=A(a_k),\\ &\beta_1=B(b_1),\beta_2=B(b_2),...,\beta_l=B(b_l).\end{aligned} \qquad (6)$$

С учетом (6) выражение (5) примет вид

$$\begin{aligned}\beta_1&=F(a_1,b_1)\alpha_1\vee F(a_2,b_1)\alpha_2\vee...\vee F(a_k,b_1)\alpha_k,\\ &...\\ \beta_l&=F(a_1,b_l)\alpha_1\vee F(a_2,b_l)\alpha_2\vee...\vee F(a_k,b_l)\alpha_k.\end{aligned} \qquad (7)$$

По формуле (7) и строится схема линейного логического оператора. При этом множества A и B

ляются своими характеристиками. Кроме того, линейный логический оператор может быть задан с помощью двудольного графа, такого, как показан на рис. 3.

Выразим по формуле (7) линейный логический оператор, ядром которого является предикат $F(x,y)$ из (1). Имеем

$$\beta_ж = 1\alpha_а \vee 1\alpha_я \vee 1\alpha_у \vee 1\alpha_ю \vee 1\alpha_о \vee 1\alpha_е \vee 1\alpha_ы \vee 1\alpha_и,$$

$$\beta_м = 0\alpha_а \vee 0\alpha_я \vee 0\alpha_у \vee 0\alpha_ю \vee 1\alpha_о \vee 1\alpha_е \vee 1\alpha_ы \vee 1\alpha_и,$$

$$\beta_с = 0\alpha_а \vee 0\alpha_я \vee 0\alpha_у \vee 0\alpha_ю \vee 1\alpha_о \vee 1\alpha_е \vee 1\alpha_ы \vee 1\alpha_и.$$

Результат совпадает с выражением (3). То есть переключательная цепь (рис. 4), отыскивающая образ множества относительно отображения $f(x)=y$, реализует линейный логический оператор с ядром $F(x,y)$. Таким образом, упростив схему для реализации отображения так, чтобы можно было находить образ множества, мы естественно пришли на схему для линейного логического оператора. Затем данный результат подтвердился выкладками. Пользуясь взаимно однозначным соответствием между одноместными предикатами и множествами, можно считать, что линейный логический оператор (4) преобразует множества.

Для нахождения прообраза множества относительно заданного отображения $f(x)=y$ строится схема, действующая в обратную сторону. Это линейный логический оператор вида:

$$A'(x)=\exists y \in N\left(F(x,y)\wedge B(y)\right), \qquad (8)$$

где в общем случае $A'(x)\neq A(x)$.

Построим схему, реализующую отображение, обратное к отображению (2) (рис. 5):

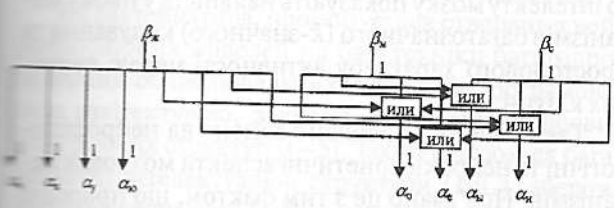


Рис. 5

При движении в обратную сторону элементы разделения превращаются в узлы и наоборот. Найдем образ какого-нибудь множества по данной схеме. По схеме на рис. 4 мы нашли: $\{а,о\}\to\{ж,м,с\}$. Теперь по данной схеме получаем:

$$\{ж,м,с\}\to\{а,я,у,ю,о,е,ы,и\}.$$

Убеждаемся в том, что $A'(x)\neq A(x)$.

Таким образом, при переходе на линейные логические операторы мы избавляемся от элементов описывания предметов и более рационально используем схемы, получая возможность находить не только полные образы предметов, но и образы множеств. Для линейного логического оператора был найден общий способ аппаратной реализации в виде переключательной цепи, которая строится по формуле (7). Полученные схемы позволят строить логические сети мозгоподобных ЭВМ, используя линейный логический оператор в качестве основного строительного элемента. Так, каждая дуга логической сети на рис. 1 представляет собой пару комбинационных схем изображенных на рис. 4, 5 и описывается парой операторов (4), (8).

Современные микроэлектронные технологии позволяют создавать мозгоподобные ЭВМ уже сегодня. Так, в [4] приводится описание интегральной схемы, которая реализует логическую сеть анализа прилагательных. Логическая сеть строится на основании алгебро-логического описания объекта [5, 6]. Однако в настоящий момент разработано незначительное число формальных моделей, которые покрывают очень малую часть естественного языка [7]. Поэтому главной задачей, которая стоит на пути создания мозгоподобной ЭВМ, является постепенное расширение границ формализации до того уровня, когда естественный язык удастся охватить целиком.

Главной отличительной чертой человека, как существа мыслящего, является владение естественным языком. Поэтому можно сказать, что «полноценная» мозгоподобная ЭВМ будет создана лишь тогда, когда будет построена логическая сеть, полностью реализующая механизм естественного языка.

Список литературы: 1. *Бондаренко М.Ф.* / М.Ф. Бондаренко, З.В. Дударь, И.А. Ефимова, В.А. Лещинский, С.Ю. Шабанов-Кушнаренко. О мозгоподобных ЭВМ // Радиоэлектроника и информатика. 2004. № 4. С. 83-99. 2. *Ротин И.М.* Линейные и билинейные логические операторы и их применение в автоматизированных информационных системах. Дисс...канд. техн наук. Харьков: ХТУРЭ. 1994. 163 с. 3. *Гвоздинская Н.А.* / Н.А. Гвоздинская, З.В. Дударь, С.А. Пославский, Ю.П. Шабанов-Кушнаренко. О действиях с линейными логическими операторами // Радиоэлектроника и информатика. 1998. № 3. С. 128–130. 4. *Bondarenko M.F., Hahanova I.V.* Logic networks application for computing process organization // Радиоэлектроника и информатика. 2003. № 3. С. 150-156. 5. *Шабанов-Кушнаренко Ю.П.* Теория интеллекта. Математические средства. Харьков: Вища шк. Изд-во при Харьк. ун-те, 1984. 144 с. 6. *Шабанов-Кушнаренко Ю.П.* Теория интеллекта. Технические средства. Харьков: Вища шк. Изд-во при Харьк. ун-те, 1986. 136 с. 7. *Шабанов-Кушнаренко Ю.П.* Теория интеллекта. Проблемы и перспективы. Харьков: Вища шк. Изд-во при Харьк. ун-те, 1987. 160 с. 8. *Баталин А.В., Тевяшев А.Д., Шабанов-Кушнаренко Ю.П.* О системном анализе информационных процессов // Радиоэлектроника и информатика. 1998. № 3. С. 102–110.

Поступила в редколлегию 14.10.2004