М. Ф. БОНДАРЕНКО, канд. техн. наук, *В. Л. СИГАЛОВ*

ЗАДАЧА ДЕКОМПОЗИЦИИ УРАВНЕНИЙ АЛГЕБРЫ КОНЕЧНЫХ ПРЕДИКАТОВ

Алгебра конечных предикатов (а. к. п.), введенная в работах [1, 2], используется для моделирования алфавитного оператора в системах искусственного интеллекта. Декомпозицией уравнений а. к. п. называется процесс выделения часто встречающихся общих частей логического уравнения в самостоятельное уравнение и оформления в головном для этих частей уравнении ссылок на них.

Обычно при составлении уравнений, описывающих некоторый логический объект, повторяющиеся логические правила дают возможность производить эвристическую декомпозицию уравнений а. к. п. Однако иногда логика объекта может быть сильно запутана, и повторяющиеся правила трудно выявить эвристически. Данная статья посвящена формализации декомпозиции уравнений а. к. п. Декомпозиция уравнений, как оптимизационная задача, может производиться по различным критериям. Критерий минимальности числа операций уменьшает суммарную длину системы уравнений а. к. п., что используется при программировании решения уравнений. Она выделяет класс быстро решаемых задач для конкретных начальных значений. Время решения общими методами уравнений, к которым применена декомпозиция, увеличивается. Для аппаратурной реализации логических схем декомпозиция уравнений уменьшает аппаратурные затраты. Еще один важный аспект применения декомпозиции, специфический для теории искусственного интеллекта: декомпозиция позволяет в сложной логической системе выделить завершенные правила, используемые в разных частях системы. Иначе это называется *процессом формализации понятий*. Формализация декомпозиции уравнений а. к. п. вытекает из теоремы о разложении.

По теореме о разложении всякая формула а. к. п. может быть представлена в виде

$$f(x_1, \ldots, x_i, \ldots, x_n) = \bigvee_{j=1}^{p_i} x_i^{\sigma_j} f^{x_i^{\sigma_j}}(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n), \tag{1}$$

где p_i — область определения переменной x_i; $f^{x_i^{\sigma_j}}(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$ — функция, полученная в результате подстановки $x_i = \sigma_j$. Графическим представлением формулы (1) является дерево. Каждая вершина его обозначает переменную, по которой ведется разложение, а исходящие дуги соответствуют формулам

$$f^{x_i^{\sigma_1}}, f^{x_i^{\sigma_2}}, \ldots, f^{x_i^{\sigma_p}}. \tag{2}$$

Декомпозиция возможна, если некоторые l функций $(l < p)$ эквивалентны: $f^{x_i^{\sigma_{m_1}}} \sim f^{x_i^{\sigma_{m_2}}} \sim \ldots \sim f^{x_i^{\sigma_{m_l}}}$ (3);

$$\bigwedge_{k=1}^{l} f^{x_i^{\sigma_{m_k}}} \vee \bigwedge_{k=1}^{l} \overline{f^{x_i^{\sigma_{m_k}}}}. \tag{4}$$

Здесь используется понятие многоаргументной эквиваленции, введенное в работе [3]. Можно показать, что многоаргументная эквиваленция аналогична термину попарной эквиваленции. Пусть $\varphi_1, \varphi_2, \ldots, \varphi_n$ — некоторые функции а. к. п., тогда система уравнений

$$\begin{aligned} &\varphi_1\varphi_2 \vee \overline{\varphi}_1\overline{\varphi}_2 = 1; \\ &\varphi_2\varphi_3 \vee \overline{\varphi}_2\overline{\varphi}_3 = 1; \\ &\ldots\ldots\ldots\ldots \\ &\varphi_{n-1}\varphi_n \vee \overline{\varphi}_{n-1}\overline{\varphi}_n = 1 \end{aligned} \tag{5}$$

эквивалентна многоаргументной эквиваленции этих же функций по определению $\sim(\varphi_1, \varphi_2, \ldots, \varphi_n) = \bigwedge_{k=1}^{n} \varphi_k \vee \bigwedge_{k=1}^{n} \overline{\varphi_k}$ **(6)**.

Рассмотрим случай регулярного дерева. Дерево называется регулярным, если все его пути в одинаковом порядке обходят помеченные соответствующими переменными вершины графа. Условимся для удобства, что дерево есть направленный граф. Направление дуг зафиксируем от корня к листьям.

Назовем вершину сходной, если все исходящие из нее дуги определяют попарно эквивалентные функции. Далее для краткости будем говорить: «дуги попарно эквивалентны» или «подграфы попарно эквивалентны».

Назовем вершину частично сходной, если только часть дуг, исходящих из этой вершины, попарно эквивалентны между собой.

Для определения сходности и частичной сходности введем следующий формализм. Каждой вершине поставим в соответствие множество $M = (L_1, L_2, \ldots, L_t)$ **(7)**, где L_i — множество номеров попарно эквивалентных дуг: $L_i = (e_1, e_2, \ldots, e_{g_i})$ **(8)**. Здесь e_j — номер дуги, вступающей в i-ю эквиваленцию.

Если $M = \varnothing$ — вершина несходная, $t = 1$ и $g_1 = p$, где p — область определения переменной, — вершина сходная, в противном случае вершина частично сходная. На рис. 1 показан пример частично сходной вершины ($M = (L_1, L_2)$; $L_1 = (1,3)$; $L_2 = (2,4)$; $p = \sigma$).

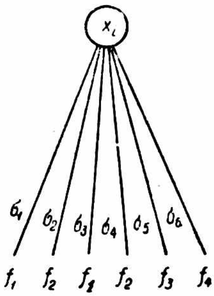

Рис. 1

Определим стоимость функции числом входящих в ее скобочную запись переменных. Тогда стоимость несходной вершины равна

$$C^{н} = \sum_{i=1}^{p} C_i + 2p, \tag{9}$$

где C_i — стоимость подграфа для i-й дуги; p — область определения переменной. Стоимость сходной вершины равна $C^{с} = C_i + 1$ **(10)**.

Для определения стоимости частично сходной вершины определим общее число входящих в эквиваленцию дуг:

$$N = \sum_{j=1}^{t} g_j. \tag{11}$$

Здесь t — число членов множества M; g_j — число членов множества L_j.

$$C^{ч.с} = \sum_{i=1}^{p-N} C_i + \sum_{j=1}^{t} C_j + 2(p - N) + \sum_{j=1}^{t} (g_j + 1), \tag{12}$$

где C_i — стоимость дуги, не входящей в эквиваленцию; C_j — стоимость дуги-представителя, входящей в L_j. Стоимость вершины, показанной на рис. 1, равна $N = 2 + 2 = 4$; $C^{ч.с} = C_{f_3} + C_{f_4} + C_{f_1} + C_{f_2} + 4 + 6$ **(13)**.

Рядом регулярного дерева будем называть совокупность вершин, соответствующих одной переменной. Каждому ряду итеративного разложения функции по существенной переменной поставим в соответствие множество $B^k = (H_1^k, H_2^k, \ldots, H_r^k)$ **(14)**; H_i^k — множество номеров попарно эквивалентных дуг; k — номер ряда; $H_i^k (v_1, v_2, \ldots, v_{s_r})$ **(15)**, где v_i — номер дуги, вступающей в эквиваленцию.

Ребра, функции которых тождественно равны булевой константе, назовем концевыми. Стоимость концевого ребра равна

$$C_k = \begin{cases} = 1 \text{ — если ребро } \sim 1; \\ = 0 \text{ — если ребро } \sim 0. \end{cases} \quad (16)$$

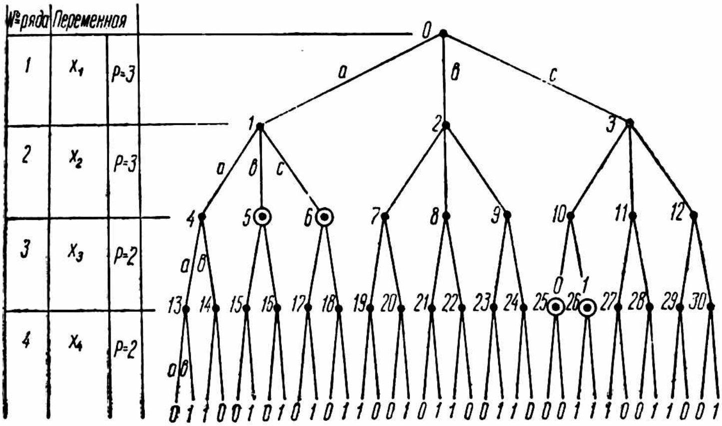

Рис. 2

Стоимость функции зависит от порядка разложения по существенной переменной. Для каждого конкретного случая разложения стоимость функции подсчитывается итеративно по рядам

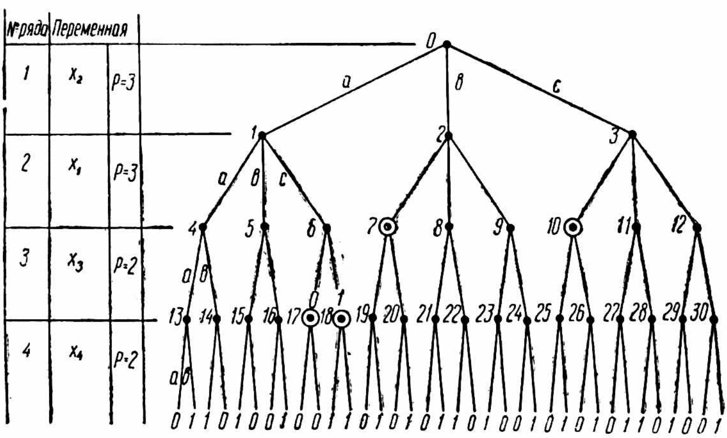

Рис. 3

дерева. При этом стоимость ребра, входящего в множество H_i^k, учитывается только один раз.

Рассмотрим два примера декомпозиции одной и той же функции, заданной таблично. Зафиксируем порядок разложения по рядам дерева от корня к листьям: (x_1, x_2, x_3, x_4) (рис. 2; $\odot$ — обозначение сходности). Определим множество $B^k (k = 1,4)$: $B^1 = \varnothing$; $B^2 = (H_1^2, H_2^2, H_3^2)$; $H_1^2 = (4, 8, 9)$; $H_2^2 = (5, 6)$; $H_3^2 = (7, 11, 12)$; $B^3 = (H_1^3, H_2^3)$; $H_1^3 = (13, 15, 16, 17, 18, 20, 21, 23, 28, 30)$; $H_2^3 = (14, 19, 22, 24, 27, 29)$. Расчет стоимости дерева и результирующая декомпозиция представлены ниже.

Стоимость	Результат декомпозиции
$C_0 = C_1 + C_2 + C_3 + 6 = 33;$	$f_0 = x_1^a f_1 \vee x_2^b f_2 \vee x_1^c f_3;$
$C_1 = C_4 + C_5 + 2 + 3 = 12;$	$f_1 = x_1^a f_4 \vee (x_2^b \vee x_2^c) f_5;$
$C_2 = C_7 + 2 + 3 = 6;$	$f_2 = x_2^a f_7 \vee (x_2^b \vee x_2^c) f_4;$
$C_3 = C_{10} + 2 + 3 = 6;$	$f_3 = x_2^a f_{10} \vee (x_2^b \vee x_2^c) f_7;$
$C_4 = C_{14} + C_{15} + 4 = 6;$	$f_4 = x_3^a f_{15} \vee x_3^b f_{14};$
$C_5 = 1;$	$f_5 = f_{15};$
$C_7 = 4;$	$f_7 = x_3^a f_{14} \vee x_3^b f_{15};$
$C_{10} = 1;$	$f_{10} = x_3^a;$
$C_{14} = 1;$	$f_{14} = x_4^a;$
$C_{15} = 1;$	$f_{15} = x_4^b.$

Установим новый порядок разложения по рядам дерева от корня к листьям: (x_2, x_1, x_3, x_4) (рис. 3). Определим множество B^k $(k = 1, 4)$: $B^1 = (H_1^1)$; $H_1^1 = (2, 3)$; $B^2 = (H_1^2, H_2^2, H_3^2)$; $H_1^2 = (4, 8, 11)$; $H_2^2 = (7, 10)$; $H_3^2 = (5, 9, 12)$; $B^3 = (H_1^3, H_2^3)$; $H_1^3 = (13, 16, 19, 20, 21, 24, 25, 26, 27, 30)$; $H_2^3 = (14, 15, 22, 23, 28, 29)$.
Стоимость подсчитывается ниже.

Как видим, порядок разложения, представленный на рис. 3, экономичнее на четыре операнда.

Общее количество регулярных деревьев для функции от n переменных равно $n!$. Таким образом, предъявленный алгоритм вряд ли может иметь практическое значение. Скорее всего он может стать той теоретической базой, на основании которой могут формироваться более практичные алгоритмы.

Стоимость	Результат декомпозиции
$C_0 = C_1 + C_2 + 2 + 3 = 29;$	$f_0 = x_2^a f_1 \vee (x_2^b \vee x_2^c) f_2;$
$C_1 = C_4 + C_5 + C_6 + 6 = 17;$	$f_1 = x_1^a f_4 \vee x_1^b f_5 \vee x_1^c f_6;$
$C_2 = C_7 + 6 = 7;$	$f_2 = x_1^a f_7 \vee x_1^b f_4 \vee x_1^c f_5;$
$C_4 = C_{13} + C_{14} + 4 = 6;$	$f_4 = x_3^a f_{13} \vee x_3^b f_{14};$
$C_5 = 4;$	$f_5 = x_3^a f_{14} \vee x_3^b f_{13};$
$C_6 = 1;$	$f_6 = x_3^b;$
$C_7 = 1;$	$f_7 = f_{13};$
$C_{13} = 1;$	$f_{13} = x_4^b;$
$C_{14} = 1;$	$f_{14} = x_4^a.$

Список литературы : 1. *Шабанов-Кушнаренко Ю. П.* О теории интеллекта. — Проблемы бионики, 1979, вып. 22, с. 109—112. 2. *Шабанов-Кушнаренко Ю. П.* Об алгебре предикатов с отрицанием. — АСУ и приборы автоматики, 1978, вып. 22, с. 42—49. 3. *Сигалов В. Л.* Задача факторизации формул алгебры конечных предикатов. — Вопросы применения ЭВМ при разработке и исследовании техн. систем. — К.: 1981, ИК АН УССР, препринт 80—42, с. 7—16.

Поступила в редколлегию 07.12.81.