Математика и её приложения: ЖИМО. 2011. Вып. 1 (8). С. 105 – 112.

УДК 512.54

А. А. Толстопятов[1]

Разбиение файла на буферы в случае, когда порождающие — часть множества булевых полиномов, кодирующих поля принадлежности

Ключевые слова: булевы полиномы, разбиение файла на буферы, булево сжатие.

Предложено описание разбиения файла на буферы с помощью матриц и построена алгебра таких матриц. Получено условие на такие матрицы, при которых система булевых уравнений может быть решена относительно каждой из булевых переменных. Описаны преобразования таких матриц при разбиении файла на буферы и показано, что множество таких матриц вместе с их преобразованиями образует категорию. Рассмотрена схема построения алгоритма разбиения файла на буферы для случая, когда порождающие – часть множества булевых полиномов, кодирующих поля принадлежности.

Keywords: boolean polynoms, splitting a file on buffers, boolean compress.

We present the description of the file splitting on buffers by the matrix algebra and construct the algebra of such matrixes. We receive the matrix condition for solvability of the system of Boolean equations for every variable. We describe transformations of these matrixes; the set of such matrixes with transformations form a category. We consider the scheme of creation the algorithm of file splitting on buffers for case, when the set of generators is a subset of the set of Boolean polynoms coding the accessory fields.

Главным адаптационным свойством алгоритма построения кода при булевом сжатии файлов, является возможность по-разному выбирать разбиение файла на буферы. Но это свойство делает задачу о выборе нужного разбиения экспоненциально сложной, т. к. существует 2^{N-1} различных разбиений, где N — число кортежей, на которые разбит файл до их объединения в буферы. Для выбора подходящего разбиения файла нужно записывать кодирующее уравнение и проверять, существуют ли у него решения. А для этого нужно выбирать систему порождающих булевых полиномов. Как показано в [2], одним из удобных для вычисления такой системы порождающих является множество булевых полиномов, выбранных из полиномов, кодирующих первые поля принадлежности. Алгоритм построения кодирующего уравнения в случае такого выбора рассмотрен в [3]. Именно для этого случая в [1] был построен алгоритм разбиения файла на буферы. Однако, этот алгоритм обладал тем недостатком, что число порождающих было 2^n, т. е. экспоненциально велико. Поэтому необходимо для указанного выше выбора порождающих получить такое описание всех возможных разбиений файла на буферы, из которых уже можно было бы

[1]Ивановский государственный университет; E-mail: khash2@mail.ru. Работа выполнена при финансовой поддержке РФФИ (проект 10-07-00350а).

выбрать систему порождающих, содержащих наименьшее число булевых полиномов. Эта проблема рассматривается в данной работе.

1. Постановка задачи

Пусть файл разбит на N кортежей длиной в n бит каждый, и пусть эти кортежи объединены в L буферов, причем m_l — число кортежей в l-м буфере ($l = 1, \ldots, L$). Это объединение и есть разбиение файла на буферы. Всего существует $2^N - 1$ разных разбиений с $L = 1, 2, \ldots, N$. Каждому буферу ставится в соответствие булев полином $f_l(x_i),\ i = 1, \ldots, n$, такой, что уравнение

$$f_l(x_i) = 0 \tag{1}$$

имеет s_l решений, где $s_l\ (\leq m_l)$ — число разных кортежей в l-м буфере. Булево сжатие основано на существовании решения кодирующего уравнения (см. [3])

$$F(e_k^l) = f_l\,, \tag{2}$$

где $F(e_k^l)$ — кодирующий полином, а e_k^l — элементы из множества порождающих булевых полиномов $\varphi_p,\ p = 1, \ldots, P,\ k = 1, \ldots, I$, где I — число булевых переменных, от которых зависит F. Параметрами, задающими разбиение файла на буферы, являются n, I, P, L. Коэффициент сжатия файла k зависит от этих параметров следующим образом (см. [3])

$$k = \frac{n\sum\limits_{l=1}^{L} m_l}{2^I + 2^n P + LI\log_2 P + \log_2 \prod\limits_{l=1}^{L} \dfrac{C_{m_l-1}^{s_l-1} m_l!}{\prod\limits_{k=1}^{s} n_k^l!}}\,, \tag{3}$$

где $n_k^l\ (k = 1, \ldots, s_l)$ — числа повторов k-го кортежа в l-м буфере.

Задача о разбиении файла на буферы заключается в том, чтобы для заданного файла, разбитого на кортежи, найти такие параметры разбиения и числа $m_l\ (l = 1, \ldots, L)$, для которых выполнено два условия:

$$\begin{aligned} &1)\ k > 1,\\ &2)\ \text{уравнение (2) имеет решение.} \end{aligned} \tag{4}$$

Для того, чтобы проверить (4), нужно записать (2) и (3), а для этого нужно иметь f_l и φ_p. Как показано в [2], [3], удобным для проверки (4) выбором φ_p является выбор, когда

$$\varphi_P \in \{f_l\}. \tag{5}$$

Так как существует C_P^L разных способов осуществить такой выбор, а $L >> P$, то возникает задача найти такое описание всех возможных выборов (5), с которым удобно было бы проводить вычисления.

2. Описание разбиений файла с помощью матриц

Задание разбиения файла с помощью чисел m_l неконструктивно, т. к. оно не позволяет записывать условия (4). Для того, чтобы получить конструктивное описание разбиения файла, рассмотрим случай, когда выполнено (5). Запишем полином f_l в лагранжевом базисе:

$$f_k = \sum_{j=0}^{2^n-1} \alpha_{kj} L_j(x_i), \tag{6}$$

где

$$L_j(x_i) = \prod_{k=1}^{n} L_{j_k}(x_i), \tag{7}$$

j_k — коэффициент разложения числа j по степеням 2,

$$j = \sum_{k=1}^{n} j_k 2^{k-1}, \tag{8}$$

а

$$L_0(x_i) = x_i + 1, \quad L_1(x_i) = x_i. \tag{9}$$

Представление (7) означает, что при построении строки матрицы α_{kj} из (6), которая содержит 2^n нулей и единиц, справедливо следующее: если кортеж с номером j входит в k-й буфер, то $\alpha_{kj} = 0$, если не входит, то $\alpha_{kj} = 1$. Матрицы α_{kj} задают разбиение файла на буферы не однозначно. А именно, это разбиение, как и положено полю принадлежности, задает только однократные вхождения в буфер того кортежа, на месте которого стоит ноль, и невхождение кортежа, на месте которого стоит 1. Однако, этот, казалось бы, недостаток предложенного описания разбиения файла на буферы, легко превратить в достоинство. Для этого достаточно начать с разбиения файла, когда $L = N$, а значит каждый буфер содержит только один кортеж. Далее, превращаем этот буфер в строку длины 2^n, состоящую из одного нуля на месте числа, соответствующего этому кортежу и $2^n - 1$ единиц на всех остальных местах. Передвижение границы между буферами приводит к появлению новых нулей вместо единиц на соответствующих местах при неизменной длине буфера в 2^n нулей и единиц. Такое представление разбиения файла на буферы с помощью матриц α_{kj} дает не только все возможные разбиения, но и преобразования между ними.

3. Алгебра матриц α_{kj}

Все возможные булевы полиномы (6) образуют алгебру, т. к. их можно складывать и умножать. Но и все матрицы α_{kj} из (6), если $k = 1, 2, \ldots, L$, а число буферов L не меняется, тоже образуют алгебру, с которой удобнее

иметь дело, т. к. элементы матриц α_{kj} — числа из поля $GF(2)$. Сложение в этой алгебре, если помимо полинома (6) есть полином

$$f_l = \sum_{i=0}^{2^n-1} \alpha_{lj} L_j(x_i), \tag{10}$$

определяется как

$$f_k + f_l = \sum_{i=0}^{2^n-1} (\alpha_{kj} + \alpha_{lj}) L_j(x_i), \tag{11}$$

а умножение полиномов (6) и (10) в силу того, что

$$L_k L_j = \delta_{kj} L_j \quad (\text{не суммировать по } j), \tag{12}$$

определяется как

$$f_k f_l = \sum_{i=0}^{2^n-1} (\alpha_{kj}\alpha_{lj}) L_j(x_i). \tag{13}$$

Формулу (13) легко обобщить на случай s сомножителей. Действительно, если

$$f_{k_m} = \sum_{i=0}^{2^n-1} \alpha_{k_m j} L_j(x_i), \tag{14}$$

то

$$\prod_{m=1}^{s} f_{k_m} = \sum_{i=0}^{2^n-1} \left(\prod_{m=1}^{s} \alpha_{k_m j} \right) L_j(x_i). \tag{15}$$

Эта алгебра матриц α_{kj} оказывается полезной при построении разбиения файла на буферы, такого, что система булевых уравнений (1), рассматриваемых как система относительно булевых переменных x_i может быть разрешена относительно всех этих булевых переменных. Это как раз и гарантирует возможность выбора φ_p согласно (5).

4. Преобразования матриц α_{kj} при изменении разбиения файла

Так как каждая матрица α_{kj} задает разбиение файла, — точнее, каждое разбиение файла может быть описано с помощью α_{kj}, — то изменение разбиения будет изменять α_{kj}. Чтобы описать это изменение, нужно рассмотреть два случая:

1) при изменении разбиения файла не меняется число буферов,

2) при изменении разбиения файла число буферов меняется.

В первом случае изменение матриц α_{kj} может быть описано с помощью двух квадратных невырожденных матриц O_{km} и R_{js}, причем первая из них размерности $L \times L$, а вторая — $2^n \times 2^n$. Тогда

$$\widetilde{\alpha}_{ks} = O_{km} \alpha_{mj} R_{js}, \tag{16}$$

или, опуская индексы,

$$\tilde{\alpha} = O\alpha R. \tag{17}$$

Поскольку $\det O \neq 0$ и $\det R \neq 0$, то существуют обратные матрицы O^{-1} и R^{-1},

$$OO^{-1} = I, \quad RR^{-1} = I, \tag{18}$$

где I — единичная матрица. Тогда из (17) и (18) будем иметь

$$\alpha = O^{-1}\tilde{\alpha}R^{-1}. \tag{19}$$

Для трех разных разбиений файла, задаваемых матрицами $\alpha_1, \alpha_2, \alpha_3$, существуют такие матрицы O_{21}, R_{12}, O_{32}, R_{23}, O_{31} и R_{13}, что справедливы соотношения

$$\begin{aligned} \alpha_2 &= O_{21}\alpha_1 R_{12}, \\ \alpha_3 &= O_{32}\alpha_2 R_{23}, \\ \alpha_3 &= O_{31}\alpha_1 R_{13}, \end{aligned} \tag{20}$$

поэтому

$$\alpha_3 = O_{32}O_{21}\alpha_1 R_{12}R_{23}. \tag{21}$$

Сравнивая (21) с каждой формулой из (20) получим, что

$$O_{31} = O_{32}O_{21}, \ R_{13} = R_{12}R_{23}. \tag{22}$$

Формулы (22) задают умножение на парах матриц (O, R). Так как это умножение, как умножение любых матриц, ассоциативно, то с учетом (18) оно задает групповую структуру на множестве пар (O, R).

Во втором случае, когда при изменении разбиения файла меняется число буферов, преобразования матриц α тоже будут иметь вид (17). Однако, если α задает разбиение с числом буферов L_1, а $\tilde{\alpha}$ — с L_2, то матрица O в (17) перестает быть квадратной и будет иметь размерность $L_2 \times L_1$. Матрица R в (17) остается квадратной, причем по-прежнему $\det R \neq 0$. Но тогда, если есть последовательность разбиений файла на буферы

$$\alpha_1 \to \alpha_2 \to \cdots \to \alpha_s, \tag{23}$$

причем разбиение, задаваемое матрицей α_1 содержит L_1 буферов, матрицей α_2 — L_2 и т. д., то на парах (O, R) можно задать операцию умножения. Однако, эта операция будет частичной, поэтому ее можно понимать как морфизмы в категории матриц α_{kl}.

5. Условия решения системы булевых уравнений относительно одной переменной

Система порождающих $\varphi_p(x_i),\ p = 1, 2, \ldots, P$, — это P булевых полиномов, через которые могут быть выражены полиномы $f_k(x_i)$:

$$f_k(x_i) = f_k(\varphi_p(x_i)). \tag{24}$$

Построение кодирующего уравнения, из которого получаются f_k упрощается, если за порождающее удается взять часть полиномов f_k. Это значит, что остальные полиномы f_k, не вошедшие в порождающие, могут быть выражены через те f_k, которые вошли в эту систему порождающих. Чтобы это было возможно, достаточно найти такое разбиение файла, т. е. систему булевых уравнений (1), которая может быть решена относительно булевых переменных x_i:

$$x_i = x_i(f_{k_p}), \tag{25}$$

где f_{k_p}, $p = 1, 2, \ldots, P$, — система порождающих. Тогда, подставляя (25) в (6), найдем, что P из L уравнений (6), когда $k = k_p$, обратятся в тождества, а для остальных будем иметь, что

$$f_k = \sum_{j=0}^{2^n-1} \alpha_{kj}(L_j(x_i(f_{k_p}))), \tag{26}$$

т. е.

$$f_k = f_k(f_{k_p}). \tag{27}$$

Чтобы найти условия, когда явно получается (25), нужно путем сложения и умножения уравнений системы (6), с учетом (12) получить следующие n соотношений:

$$\sum_{j=0}^{2^n-1} \beta_{ij} L_j(x_i) = F_i(f_k), \tag{28}$$

где $F_i(f_k)$ — булев полином от f_k. Учитывая, что булев полином — это полилинейная функция от каждой из булевых переменных, а также соотношения (7) и (9), уравнение (28) можно переписать так:

$$x_i \sum_{j=0}^{2^n-1} \gamma_{ij} L_j(x_s) + (x_i + 1) \sum_{j=0}^{2^n-1} \xi_{ij} L_j(x_s) = F_i(f_k), \tag{29}$$

где множество булевых переменных $\{x_s\}$ есть

$$\{x_s\} = \{x_1, x_2, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n\}. \tag{30}$$

Поэтому полиномы Лагранжа $L_j(x_s)$ зависят от $n-1$ булевой переменной (30), а не от n. Каждое из n уравнений (29) может быть решено относительно каждой из x_i только тогда, когда

$$\sum_{j=0}^{2^{n-1}-1} \gamma_{ij} L_j(x_s) = 1 \tag{31}$$

и

$$\sum_{j=0}^{2^{n-1}-1} \xi_{ij} L_j(x_s) = 0. \tag{32}$$

Но поскольку справедливо равенство

$$\sum_{j=0}^{2^{n-1}-1} L_j(x_s) = 1, \tag{33}$$

то условие (31) превращается в следующее

$$\gamma_{ij} = 1, \tag{34}$$

причем $i = 1, \ldots, n;\ j = 0, 1, \ldots, 2^{n-1} - 1$. А в силу независимости полиномов Лагранжа, (32) будет выполнено только при условии

$$\xi_{ij} = 0. \tag{35}$$

Таким образом, условия (34) и (35) решают первую задачу о разрешимости системы булевых уравнений относительно отдельной переменной.

6. Схема разбиения файла на буферы, когда порождающие — часть множества булевых полиномов, кодирующих поля принадлежности

Чтобы ответить на вопрос о том, существует ли разбиение файла, удовлетворяющее (5), достаточно критерии (34) и (35) записать так, чтобы они превратились в условия на матрицы α. Решение этой задачи удобно получить не в базисе Лагранжа относительно f_k, а в базисе Жегалкина, т. е. в базисе мономов, построенных из f_k, являющихся порождающими. Мономы Жегалкина имеют следующий вид:

$$\begin{array}{l} 0;\ f_{k_1},\ f_{k_2},\ \ldots,\ f_{k_n};\ f_{k_1}f_{k_2},\ \ldots f_{k_{n-1}}f_{k_n}; \\ f_{k_1}f_{k_2}f_{k_3},\ \ldots,\ f_{k_{n-2}}f_{k_{n-1}}f_{k_n};\ \ldots;\ f_{k_1}f_{k_2}\ldots f_{k_n};\ 1. \end{array} \tag{36}$$

Поскольку при умножении полиномов f_k матрицы α_{kj} тоже умножаются, то для того, чтобы произведение из s полиномов ($0 \le s \le n$) было отлично от нуля, необходимо, чтобы у s разных полиномов f_k на одном и том же j-м месте в матрице α_{kj} стояли единицы. Поэтому большинство мономов из (36) обратятся в нули. Функции f_k из (27) — это линейные функции от мономов (36), т. е.

$$F_k(f_s) = \eta_0 + \sum_{\alpha=1}^{n} \eta_\alpha f_{k_\alpha} + \sum_{\alpha=1}^{n}\sum_{\beta=1}^{n} \eta_{\alpha\beta} f_{k_\alpha} f_{k_\beta} + \cdots + \\ + \sum_{\alpha=1}^{n}\sum_{\beta=1}^{n}\cdots\sum_{\omega=1}^{n} \eta_{\alpha\beta\ldots\omega} f_{k_\alpha} f_{k_\beta}\ldots f_{k_\omega}. \tag{37}$$

Подставляя (28) в (37) и приводя полученное уравнение к виду (29), получаем из (34) и (35) уравнения, содержащие в качестве неизвестных

элементы матриц α_{kj}, $k = k_1, k_2, \ldots, k_n$, и η_0, η_α, $\eta_{\alpha\beta}$, $\ldots$, $\eta_{\alpha\beta\ldots\omega}$. Если полученные булевы уравнения имеют решения, то выбранные f_k могут образовывать системы порождающих, если не имеют — то не могут. Предложенный алгоритм полностью перебор не исключает, т. к. из L полиномов f_k нужно выбрать n являющихся порождающими. Так как для достаточно длинных файлов $L >> n$, то существует C_L^n способов выбрать разные множества полиномов f_k — кандидатов в порождающие. Этого перебора можно избежать, если в (36) и (37) взять не какие-то n, а все L полиномов f_k. Тогда алгоритм получения нужных f_k остается прежним, но мы будем иметь все решения для f_k, которые содержат разное число подходящих f_k. Выбирая решение с наименьшим числом полиномов f_k, мы полностью исключаем перебор. Однако, цена этого исключения — очень сильное увеличение как числа рассматриваемых булевых уравнений, так и числа переменных, для которых эти уравнения записываются.

Список литературы

1. *Гришко М. Е.* Один из возможных способов разбиения файла на буферы // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2010. – Вып. 1 (7). – С. 25–28.
2. *Толстопятов А. А.* Построение системы порождающих полиномов при булевом сжатии файлов // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2010. – Вып. 1 (7). – С. 59–68.
3. *Толстопятов А. А.* Построение кодирующего уравнения при булевом сжатии файлов // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2010. – Вып. 1 (7). – С. 69–82.

Поступила в редакцию 15.12.2011