Вестник Ивановского государственного университета

Серия «Биология, Химия, Физика, Математика»

Вып. 3 / 2003 С. 138 – 140

С. И. Хашин, Ю. А. Хашина

ОПТИМИЗАЦИЯ АЛГОРИТМА СЖАТИЯ МЕТОДАМИ БУЛЕВОЙ АЛГЕБРЫ

Предложен способ оптимизации алгоритма сжатия методами булевой алгебры. Предварительное преобразование исходных данных позволяет значительно упростить вид преобразованной информации.

The way of optimization of algorithm of compression by meyhods of Boolean algebra is offered. Preliminary transormation of initial data allows to simplify considerably the kind of the transformed information.

УДК 512.54.

Введение

В статье [2] изложен подход к сжатию информации методами, основанными на булевых алгебрах. Его идея состоит в следующем. Исходный файл разбивается на *кортежи* длиной по n бит, которые затем объединяются в *буферы* по m кортежей в каждом. Рассмотрим буфер B, состоящий из кортежей $B = (c_1, \ldots, c_m)$. Среди кортежей c_i могут встретиться повторяющиеся. Обозначим через $M = (c'_1, \ldots, c'_s)$ неупорядоченное множество, состоящее из всех кортежей $(c_1, \ldots, c_m)$, $(s \leqslant m)$. Найдем булево уравнение $f(x_1, \ldots, x_n) = 0$ от n переменных, множество решений которого совпадает с множеством M. Чтобы восстановить буфер B при заданном многочлене $f(x_1, \ldots, x_n)$, требуется, во-первых, указать сколько раз в кортеже B встречается каждый из элементов $(c'_1, \ldots, c'_s)$, а во-вторых, в каком порядке. В работе [2] эти данные названы полем вхождения (многочлен f), полями кратности и порядка соответственно.

На следующих этапах работы алгоритма применяются обычные методы архивации [1].

Оптимизация. Объединение второго и третьего полей

Рассмотренный выше подход может быть подвергнут определенной оптимизации. В работе [2] длина второго поля оценена как $\log_2(D^s_{m-s})$ для некоторых коэффициентов D^s_{m-s}, третьего — как

$$\log_2 \frac{m!}{\prod_{k=1}^{s} n_k!}.$$

Работа выполнена при финансовой поддержке РФФИ (проект 02-01-00079).

В реальных ситуациях можно ожидать, что чаще всего все кортежи из набора $B = (c_1, \ldots, c_m)$ окажутся различными, то есть поле кратности будет содержать лишь тривиальную информацию. Таким образом, может оказаться, что "поле кратности"лишь усложняет алгоритм и не несет в себе содержательной информации. Поэтому вполне естественно было бы объединить данные из второго и третьего полей.

Пусть требуется задать последовательность кортежей

$$B = (c_1, \ldots, c_m),$$

каждый из которых является элементом множества $M = (c'_1, \ldots, c'_s)$. Будем считать, что кортеж c_i является k_i-м элементом множества M. Тогда последовательность B определяется набором чисел

$$k_1, \ldots, k_m, \qquad k_i \in [1, \ldots, s].$$

При таком кодировании суммарное количество битов, заменяющих второе и третье поля, равно $(\log_2(s))^m$. При вполне естественном предположении $m \ll 2^n$, которое будет выполняться при любой практической реализации, кортежи в наборе $B = (c_1, \ldots, c_m)$ почти всегда будут оказываться все различные, то есть $s = m$. Поэтому среднюю длину части кода, предназначенной для записи этой информации, при таком предположении, можно оценить как $(\log_2(n))^m$.

Данный подход позволяет сделать еще одно существенное дополнение к алгоритму. Если многочлен $f_1(x_1, \ldots, x_n)$ будет иметь "лишние"корни, то его никак нельзя использовать в оригинальной версии алгоритма. Предложенная модификация позволяет использовать многочлен f_1 точно так же, как и точный многочлен f. Роль булева многочлена теперь состоит не в том, чтобы точно описать множество буферов, а в том, чтобы ограничить их число.

Присоединение полей кратности и порядка к полю вхождения

Конструкцию, предложенную в [2], можно видоизменить так, чтобы ценой увеличения основной части кода ("поле вхождения") "поля кратности и порядка"оказались излишними.

Кортежи c_i, из которых состоит буфер $B = (c_1, \ldots, c_m)$, будем рассматривать как целое неотрицательное число из отрезка $[0 \ldots 2^n - 1]$. Если нам известно неупорядоченное множество $M = (c'_1, \ldots, c'_s)$, состоящее из всех кортежей $(c_1, \ldots, c_m)$, то для восстановления исходного буфера B требуются "поля кратности и порядка". Преобразуем буфер (последовательность целых неотрицательных чисел) B в некоторый новый буфер $B'' = (c''_1, \ldots, c''_m)$ по следующему правилу.

$$\begin{aligned} c''_1 &= c_1, \\ c''_2 &= c''_1 + 1 + c_2, \\ c''_3 &= c''_2 + 1 + c_3, \\ &\ldots \\ c''_m &= c''_{m-1} + 1 + c_m. \end{aligned}$$

При этом, например, буфер $(0, 0, \ldots, 0)$ переходит в буфер $(0, 1, 2, \ldots, m-1)$, буфер $(2^n-1, 2^n-1, \ldots, 2^n-1)$ — в

$$(2^n-1, 2\cdot 2^n-1, 3\cdot 2^n-1, \ldots, m\cdot 2^n-1).$$

Предложение. *Последовательность чисел (буфер) B'' обладает следующими свойствами:*

1. *Буфер B однозначно восстанавливается из B''.*
2. $c''_1 < c''_2 < \cdots < c''_m$.
3. $c''_i < i \cdot 2^n$.

Доказательство. 1. $c_i = c''_i - c''_{i-1} - 1$. 2. c''_i получается из c''_{i-1} прибавлением 1 и неотрицательного числа c_i. 3. Индукция по i: пусть $c''_i \leqslant i \cdot 2^n - 1$. Тогда

$$c''_{i+1} = c''_i + 1 + c_i \leqslant i \cdot 2^n - 1 + 1 + 2^n - 1 = (i+1)\cdot 2^n - 1.$$

Следствие 1. *Последовательность чисел $(c''_1, \ldots, c''_m)$ однозначно восстанавливается по неупорядоченному множеству составленному из тех же чисел.*

Доказательство. По п. 2 предложения, числа c''_i не повторяются и монотонно возрастают. Поэтому, расположив числа из множества в порядке возрастания, мы как раз и получим исходную последовательность чисел.

Следствие 2. *Предположим, что число m является степенью 2: $m = 2^{m'}$. Тогда буфер B'' будет состоять из кортежей длины $n + m'$.*

Доказательство. Из доказанного предложения следует:

$$c''_i \leqslant c''_m < m \cdot 2^n = 2^{m'+n}.$$

При $m = 2^{m'}$ для данного буфера B и полученного из него буфера B'' рассмотрим $f''(x_1, \ldots, x_n, x_{n+1}, \ldots, x_{n+m'}) = 0$ — булево уравнение от $n+m'$ переменных, корнями которого являются все кортежи из B''. Зная многочлен f'', мы можем найти его корни и согласно следствию 1 восстановить буфер B''. Затем согласно п. 1 предложения сможем восстановить и исходный буфер B.

Другими словами, мы можем применить идею булевского сжатия в чистом виде: не требуется ни поля кратности, ни поля порядка. Правда, длина каждого буфера увеличилась с n до $n + \log_2(m)$ бит, но это адекватная плата за упрощение задачи.

Библиографический список

1. *Ватолин Д., Ратушняк А., Смирнов М., Юдин В.* Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео // М.: Диалог — МИФИ, 2002.
2. *Толстопятов А. А.* О структуре дискретной информации и общих условиях ее сжатия // Вестн. Иван. гос. ун-та. 2002. Вып. 3. С. 80—82.