

М. Е. Гришко¹

Медленный алгоритм кодирования и декодирования поля кратности и поля порядка одним числом при булевом сжатии файлов

Ключевые слова: булева алгебра, поле порядка, поле кратности.

Предложен алгоритм, позволяющий кодировать поле кратности и поле порядка вместе одним числом. Алгоритм основан на процедуре последовательной генерации упорядоченной таблицы всех возможных типов повторов и их перестановок. Выполнение построенного в этой работе алгоритма занимает длительный промежуток времени, но достаточно легко реализуется программно, не требует больших затрат вычислительных ресурсов, что является основным его преимуществом.

We suggest the algorithm, which allows to code a multiplicity field and an order field by one number. This algorithm is based on a procedure that generates the ordered table of all possible types of repetitions and their permutations. This algorithm is slow, but it is easily enough realized by program; it does not demand great computing resources.

В вопросе сжатия дискретной информации без потерь был предложен подход, основанный на использовании булевых уравнений. В работах [1–8]² изложены математические основы данного подхода. Предполагается, что код исходного файла будет содержать три поля:

- 1) поле принадлежности;
- 2) поле кратности;
- 3) поле порядка.

Ранее предполагалось кодировать поле кратности двумя числами. В работах [9, 10] реализована возможность кодировать поле кратности одним числом. Также в других работах построены алгоритмы кодирования поля порядка [3, 8]. В работе [7] рассмотрена возможность объединить поле кратности и поле порядка. Таким образом, кодировать их можно совместно одним числом.

1. Постановка задачи

Поставим задачу следующим образом. Разобьем исходный файл на кортежи по n битов. Далее объединим кортежи в буфер. Вообще говоря, буферы могут иметь различную длину. Для удобства в данной работе бу-

¹Ивановский государственный университет; E-mail: mgrishko_37@mail.ru. Работа выполнена при финансовой поддержке РФФИ (проект 07-07-00155).

²См. библиографический список к статье “Быстрый алгоритм кодирования и декодирования поля кратности и поля порядка одним числом при булевом сжатии файлов”.

дет рассмотрено объединение кортежей в буферы одинаковой длины по m штук.

Очевидно, что существует 2^n возможных отличных друг от друга кортежа. Для того, чтобы установить, какие из них входят в рассматриваемый нами буфер, необходимо задать поле принадлежности. Его структура и алгоритм кодирования описаны в работе [6].

Среди кортежей, входящих в один буфер, будут встречаться повторяющиеся. Обозначим число различных кортежей в буфере через s . Назовем *числом кратности* n_k количество повторов каждого кортежа. Тогда последовательность чисел n_1, n_2, \dots, n_s , где n_k – кратность k -го кортежа ($k = 1, 2, \dots, s$), назовем *типом повторов*. Будет выполняться соотношение

$$\sum_{k=1}^s n_k = m. \quad (1)$$

В работе [9] описан способ построения упорядоченной таблицы, содержащей все возможные типы повторов для заданного буфера (то есть для определенных чисел m и s). Но тип повторов содержит далеко не всю необходимую информацию о виде кодируемого буфера. Для каждого типа повторов строится таблица, состоящая из всех возможных перестановок чисел кратности, входящих в тип повторов. Назовем эту таблицу *полем порядка*. Тогда последовательность j_1, j_2, \dots, j_m , в которой s элементов различны, будем называть *перестановкой* для определенного типа повторов.

В этой работе представлен способ построения таблицы, включающей в себя и поле кратности, и поле порядка. Таблица будет иметь вид:

$$\begin{aligned} & \text{I. } n_k = 1, n_s = m - s + 1, k = 1, 2, \dots, s - 1; \\ & \quad 1) j_i = 1, i = 1, \dots, n_1, \\ & \quad \quad j_i = 2, i = n_1 + 1, \dots, n_1 + n_2, \\ & \quad \quad j_i = 3, i = n_1 + n_2 + 1, \dots, n_1 + n_2 + n_3, \\ & \quad \quad \dots \\ & \quad 2) \dots \\ & \text{II. } n_k = 1, n_{s-1} = 2, n_s = m - s, k = 1, 2, \dots, s - 2; \\ & \quad 1) \dots \\ & \quad \dots \\ & C_{m-1}^{s-1}. n_1 = m - s + 1, n_k = 1, k = 2, \dots, s, \\ & \quad 1) \dots \end{aligned} \quad (2)$$

Очевидно, что она содержит все возможные для заданного буфера типы повторов и соответствующие им перестановки. Длину всей таблицы можно вычислить, используя формулу

$$N = \sum_{i=1}^{C_{m-1}^{s-1}} \frac{m!}{\prod_{k=1}^s n_k^i}. \quad (3)$$

Исходя из вышесказанного, можно сформулировать задачи, решаемые в данной работе:

- задача кодирования;
- задача декодирования.

Задача кодирования поля кратности и поля порядка одним числом заключается в том, чтобы по заданному буферу вычислить номер типа повторов и его перестановки для этого буфера в описанной выше таблице.

Задача декодирования состоит в обратном: по заданному номеру в упорядоченной таблице (2) восстановить вид буфера.

2. Алгоритм кодирования

1. Пусть задан буфер исходного файла. По входным данным определяются значения m и s , используя которые мы можем однозначно построить таблицу (2) для заданного буфера.

2. Суть алгоритма состоит в том, что мы из i -той строки таблицы (2) можем однозначно сгенерировать $(i + 1)$ -ю строку. Для удобства введем обозначения:

$$J_i = (j_1^i, j_2^i, \dots, j_m^i), \quad J_{i+1} = (j_1^{i+1}, j_2^{i+1}, \dots, j_m^{i+1}).$$

3. Опишем процедуру последовательной генерации строк таблицы (2). Переход от i -й строки к $(i + 1)$ -й производится следующим образом.

3.1. Для i -й строки проверяется условие

$$j_p < j_{p+1} \tag{4}$$

последовательным перебором $p = 1, 2, \dots, s - 1$.

3.2. Находится максимальное значение индекса p , который удовлетворяет условию (4).

3.3. Таким образом, при генерации $(i + 1)$ -й строки из i -й первые $p - 1$ членов перестановки будут иметь вид:

$$j_n^{i+1} = j_n^i, \quad n = 1, 2, \dots, p - 1.$$

3.4. Далее на место $j_p^{(i+1)}$ -го члена перестановки из последовательности $(j_p^i, j_{p+1}^i, \dots, j_m^i)$ выбирается такой член, который удовлетворяет условию

$$j_p^{i+1} = j_p^{i+1} + q, \tag{5}$$

где $q = 1, 2, \dots$ с минимальным q .

3.5. Все остальные члены последовательности, включая j_p^i , упорядочиваются по возрастанию. Им присваиваются обозначения j_n^{i+1} , где n пробегает значения от $p + 1$ до m .

4. В случае, когда при выполнении пункта 3.1. невозможно найти такой индекс p , который бы удовлетворял условию (2), появляется необходимость в генерации нового типа повторов. Данная процедура подробно описана в работе [10].

5. На каждой итерации необходимо проводить процедуру сравнения полученной в ходе генерации перестановки с исходной последовательностью, код которой нужно найти.

6. Если последовательность J_i совпадает с искомой перестановкой, тогда номер i -й строки в таблице (2) и будет кодом поля кратности и поля порядка заданного буфера.

3. Алгоритм декодирования

В этом разделе описан алгоритм, позволяющий по заданному номеру N в таблице (2) восстановить структуру кодированного буфера. Его несложно описать, пользуясь процедурой построчной генерации таблицы (2), предложенной в предыдущем разделе.

Для декодирования поля порядка и поля кратности необходимо провести генерацию таблицы до строки с номером N . Данная строка содержит необходимую информацию о структуре декодируемого буфера.

4. Заключение

Таким образом, в настоящей работе предложен медленный алгоритм кодирования и декодирования поля порядка и поля кратности одним числом при булевом сжатии файлов. Алгоритм обеспечивает длину кода, меньшую, чем при кодировании вышеназванных полей по отдельности. Недостатком алгоритма можно считать то, что его выполнение предполагает построчную генерацию таблицы, что занимает достаточно много времени.

Автор выражает благодарность А. А. Толстопятову, доценту кафедры ТФМКМ физического факультета ИвГУ, за постановку задачи и полезные обсуждения.

Поступила в редакцию 19.06.2009.