

УДК 512.54

М. Е. Гришко^{1, 2}

Один из возможных способов разбиения файла на буферы при булевом сжатии

Ключевые слова: булев полином, поле принадлежности, порождающий базис

Предложен один из вариантов разбиения дискретного файла на буферы. Описанный алгоритм позволяет объединить кортежи таким образом, что становится возможным выразить набор булевых переменных x_k через полиномы f_l , которые в свою очередь зависят от x_k . Преимущество рассматриваемого здесь алгоритма состоит в его способности отделить построение порождающего базиса от построения кодирующего уравнения для файла. Это значительно облегчает задачу построения кодирующего полинома.

Keywords: boolean polynom, belonging field, generating bases.

We present the new method of splitting some discrete file to buffers. This method simplifies the problem of construction a coding polynom.

Стандартные методы сжатия информации основаны на том, что буферы, на которые разделен сжимаемый файл, независимы друг от друга. Таким образом, обеспечивается выполнение теоремы Шеннона. В работе [1] предложен подход к сжатию дискретной информации без потерь, предполагающий существование нелинейных зависимостей между отдельными буферами. Суть данного метода заключается в том, что кортежи, которые объединяются в буферы, рассматриваются как решения булева уравнения. Буферы могут быть, вообще говоря, различной длины. Код сжимаемого таким образом файла состоит из трех полей — поля принадлежности, дающего информацию о том, какие из 2^n возможных кортежей входят в данный буфер, поля кратности, содержащего числа повторов кортежей в буфере, и поля порядка, по которому можно восстановить, как были расставлены кортежи в буфере. Для двух последних полей алгоритмы кодирования и декодирования построены в работах [2, 4, 5, 3, 8, 9, 10, 11]. Соответствующие алгоритмы для поля принадлежности построены в [6]. Там предложен метод, основанный на использовании кодирующего полинома, общего для всего файла. При описанном подходе каждый буфер задается набором порождающих полиномов, при подстановке которых в кодирующее уравнение мы получаем булев полином f_l , кодирующий поле принадлежности данного буфера. Это позволяет существенно сократить длину поля принадлежности. Осталось решить вопрос, что же брать в качестве порождающего базиса. Частично данная проблема будет решена ниже.

1. Постановка задачи

Разобьем дискретный файл на кортежи по n битов. Далее объединим их последовательно по m_l штук, где индекс l пробегает значения от 1 до L (L — количество буферов, на которые разбит файл). Такой набор из m

¹Ивановский государственный университет; E-mail: mgrishko@mail.ru.

²Работа выполнена при финансовой поддержке РФФИ (проект 10-07-00350а).

кортежей назовем буфером. Каждый буфер кодируется полем принадлежности, которое будет иметь длину 2^n и задаваться коэффициентами C_j^l разложения булева полинома f_l по базису Лагранжа

$$f_l(x_k) = \sum_{j=0}^{2^n-1} C_j^l L_j(x_k). \quad (1)$$

Решения полинома f_l дают кортежи, входящие в буфер. Таким образом, это будет набор из 2^n нулей и единиц, где на месте кортежа, входящего в буфер, стоит ноль, а на остальных позициях — единицы. Поле принадлежности всего файла состоит из полей принадлежности для каждого отдельного буфера. В работе [6] для сокращения длины поля принадлежности предлагается ввести общий для всего файла кодирующий полином

$$F(e_i) = \sum_{j=0}^{2^I-1} C_j^e L_e(e_i) \quad (2)$$

и систему полиномов порождающего базиса

$$\varphi_p(x_k) = \sum_{j=0}^{2^I-1} C_{pj} L_j(x_k). \quad (3)$$

При подстановке элементов порождающего базиса в кодирующий полином мы будем получать поле принадлежности соответствующего буфера. Методы решения кодирующего уравнения предложены в работе [6]. Кодирующий полином строится для каждого конкретного разбиения. И в случае, если полином не будет иметь решения для определенного разбиения, то это совсем не значит, что он не имеет решения вовсе. Вполне возможно, что для другого разбиения кодирующее уравнение будет иметь решение. Поэтому имеет смысл отделить задачу построения системы полиномов порождающего базиса от построения самого кодирующего уравнения. В статье [12] показано, что удобно в качестве системы порождающих полиномов брать либо сами булевы переменные x_k , либо часть булевых полиномов f_l . В данной работе решается задача о том, как разбить дискретный файл так, чтобы система полиномов порождающего базиса была образована частью булевых полиномов f_l .

2. Основная идея

Для работы с булевыми полиномами f_l удобно пользоваться матричным представлением, где строками будут коэффициенты C_j^l соответствующего полинома:

$$\begin{pmatrix} C_1^1 & C_2^1 & \dots & C_k^1 \\ C_1^2 & C_2^2 & \dots & C_k^2 \\ \vdots & \vdots & \vdots & \vdots \\ C_1^L & C_2^L & \dots & C_k^L \end{pmatrix}.$$

Число строк такой матрицы будет L , а число столбцов — n . Буферы, на которые делится файл, имеют размеры много меньше в сравнении с самим файлом. Очевидно, что строк в матрице будет много больше, чем столбцов.

Основную идею решения поставленной задачи можно разделить на два этапа: сначала выразить набор булевых переменных x_k через часть булевых полиномов f_l (обозначим ее как f_l^*), а потом подставить во все остальные полиномы f_l вместо x_k их выражения через f_l^* . Таким образом, система порождающих полиномов будет явно зависеть от части f_l . Для того, чтобы стало возможным выражение набора булевых переменных x_k через полиномы f_l , разделим файл особым способом. Разобьем файл так, чтобы часть буферов состояла всего из одного кортежа. Определим условие: все такие буферы-кортежи различны между собой. Очевидно, их максимальное количество будет равно 2^n . При выполнении данного условия в вышепредставленной матрице можно выделить подматрицу вида

$$\begin{pmatrix} 0 & 1 & 1 & 1 & \dots & 1 \\ 1 & 0 & 1 & 1 & \dots & 1 \\ 1 & 1 & 0 & 1 & \dots & 1 \\ \vdots & \vdots & \vdots & \vdots & & \\ 1 & 1 & \dots & 1 & 1 & 0 \end{pmatrix}.$$

Таким образом, мы получим совокупность полиномов f_l^* , через которые, пользуясь правилами сложения и умножения булевых полиномов, можно получить аналогичный полином с интересующим нас набором коэффициентов C_j^l в разложении Лагранжа. А так как булевы переменные x_k по сути своей также являются полиномами, то и их возможно выразить через часть f_l^* .

3. Алгоритм деления файла на буферы

- (1) Разобьем дискретный файл на кортежи по n битов.
- (2) Первый кортеж выделим в отдельный буфер.
- (3) Далее рассматриваем второй кортеж. Если он отличен от первого, его так же выделяем в отдельный буфер. Если совпадает с первым, то переходим к третьему кортежу.
- (4) Таким образом, необходимо последовательно перебирать кортежи до тех пор, пока не встретятся 2^n различных кортежа, т. е. не будет полон набор f_l^* булевых полиномов.
- (5) Все кортежи, что стоят между f_l^* и f_{l+1}^* , следует объединять в отдельный буфер.
- (6) После того, как набор f_l^* достиг своего максимального значения, файл делится на буферы, длина которых удовлетворяет неравенству (52) в работе [7]:

$$\frac{(2^{I-n} + P)n}{n^2 - (I + I \cdot 2^{-n} \log_2 P)n + \log_2 n} < L < \frac{(N_\Phi 2^{-n} - I \cdot 2^{-n} \log_2 P - P)n}{(1 + I \cdot 2^{-n} \log_2 P)n - \log_2 n},$$

где n — размер кортежа в битах, I — число булевых переменных, от которых зависит кодирующий полином, P — число порождающих полиномов, N_Φ — длина файла.

4. Заключение

Предложенный в данной работе способ разбиения дискретного файла на буферы позволяет взять в качестве системы порождающих полиномов либо сами булевы переменные x_k , либо часть полиномов f_l , кодирующих поле принадлежности каждого буфера. Это значительно упрощает задачу решения кодирующего уравнения путем отделения проблемы построения базиса порождающих полиномов от построения самого кодирующего уравнения.

Задача, поставленная автором, решена. Очевидно, что это далеко не единственное возможное разбиение. И предложенный выше метод, вероятно, не является оптимальным. Таким образом, вопрос оптимизации алгоритма разбиения файла на буферы остается открытым.

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

Список литературы

1. Толстопятов А. А. О структуре дискретной информации и общих условиях ее сжатия // Вестник ИвГУ. – 2002. – Вып. 3. – С. 80–82.
2. Толстопятов А. А. Вычисление длины поля кратности при булевом сжатии файлов // Вестник ИвГУ. – 2004. – Вып. 3. – С. 71–76.
3. Толстопятов А. А. Быстрый алгоритм кодирования и декодирования поля порядка при булевом сжатии файлов // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2007. – Вып. 1 (4). – С. 35–46.
4. Толстопятов А. А. Медленный алгоритм кодирования и декодирования поля кратности при булевом сжатии файлов // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2007. – Вып. 1 (4). – С. 47–52.
5. Толстопятов А. А. Быстрый алгоритм кодирования и декодирования поля кратности при булевом сжатии файлов // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2007. – Вып. 1 (4). – С. 53–78.
6. Толстопятов А. А. Алгоритм кодирования и декодирования поля принадлежности // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2008. – Вып. 1 (5). – С. 53–76.
7. Толстопятов А. А. Алгоритм разбиения файла на буферы при булевом сжатии // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2008. – Вып. 1 (5). – С. 59–68.
8. Толстопятов А. А. Возможность кодирования поля кратности и поля порядка одним числом // Математика и ее приложения: Журн. Иванов. матем. об-ва. – 2009. – Вып. 1 (6). – С. 121–128.
9. Толстопятов А. А., Хашинов С. И. Алгоритм построения поля порядка при булевом сжатии // Вестник ИвГУ. – 2004. – Вып. 3. – С. 139–143.
10. Толстопятов А. А., Гришко М. Е. Быстрый алгоритм кодирования и декодирования поля кратности одним числом при булевом сжатии файлов // Вестник ИвГУ. – 2009. – Вып. 2. – С. 45–52.
11. Толстопятов А. А., Гришко М. Е. Медленный алгоритм кодирования и декодирования поля кратности одним числом при булевом сжатии файлов // Вестник ИвГУ. – 2009. – Вып. 2. – С. 53–55.
12. 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.

Поступила в редакцию 8.12.2010.