01.00.00 Physics-mathematics sciences

01.00.00 Физико-математические науки

UDC 519.7

Encoding and Decoding Procedures for Arrangements

Alexander A. Babaev

Saint Petersburg State University, St. Petersburg, Russia
195027, St. Petersburg, Sredneokhtinsky Av., 8, Apt. 68
PhD, Associate Professor
E-mail: a.babaev@hotmail.com

Abstract. This article discusses an algorithm based on the encoding procedure for representing a set of arrangement elements as a single number. Also the author provides the procedure for the inverse transformation of the code into arrangement elements. In addition the Article includes recommendations on the use of the above procedures in combinatorial algorithms of optimization.

Keywords: algorithm; decoding; encoding; combinatorics; permutation; procedure; arrangement; a combination; the elements of arrangement.

Введение. Перестановки, сочетания и размещения как комбинации некоторого числа элементов являются фундаментальными понятиями комбинаторики [1 – 4] и используются при формализации математических моделей задач оптимизации для различных практических приложений. В частности, решение прикладной комбинаторной задачи сводится к нахождению такой совокупности элементов некоторой комбинации, которая оптимизирует значение целевой функции.

При конечном числе значений каждого элемента комбинации того или иного типа конечно и количество возможных комбинаций этих элементов. Следовательно, каждой комбинации можно поставить в соответствие код или номер, выражающийся целым неотрицательным числом z, $0 \le z < N$. Представление комбинации в виде одного числа $\boldsymbol{z}$ позволяет более экономно расходовать оперативную и внешнюю память при организации вычислительного процесса в различных алгоритмах оптимизации.

Однако в последующем появляется необходимость по коду z декодировать некоторую комбинацию или восстановить последовательность составляющих ее элементов. Способы кодирования-декодирования перестановок и сочетаний приведены в работах [5, 6]. Настоящая работа посвящена рассмотрению этих вопросов применительно к размещениям.

Кодирование размещения. Под размещением $R = (r_1, r_2, \cdots, r_n)$, будем понимать произвольную комбинацию n чисел, выбранных из m чисел натурального ряда 1, 2, ⋯ , m. Возможное число всех размещений из m элементов по n определяется формулой

$$N = \frac{m!}{n(m-n)!}.$$

Преобразование размещения R в его свертку z, $0 \le z < N$, можно осуществить с помощью следующей процедуры.

Процедура *SR* (*m*, *n*, *R*, *z*).

Вход. Вычислить $w = n!$; $l = 0$; $k = 1$.

C1. Положить $q = m + 1$; $j = 1$.

C2. Если $(r_j < q) \wedge (r_j > l)$, то положить $q = r_j$; $i = j$.

Если $j = n$, то вычислить $j = j + 1$ и перейти к метке C2.

Положить $c_k = q$; $P_i = k$; $l = q$.

Если $k < n$, то вычислить $k = k + 1$ и перейти к метке C1.

Выполнить процедуру свертки перестановки SP (n, P, z).

Положить $q = z$.

Выполнить процедуру свертки сочетания SC (m, n, C, z).

Вычислить $z = z \times w + q$.

Выход.

Рассмотрим пример. Пусть $m = 5$, $n = 3$, $R = (2, 5, 1)$. Тогда процесс кодирования будет вестись в следующей последовательности.

Вход. $w = 6$; $l = 0$; $k = 1$.

С1. $q = 6$; $j = 1$.

С2 .$(r_1 < q) \wedge (r_1 > l)$, $q = 2$; $i = 1, j < n, j = 2$.

$(r_2 > q), j < n, j = 2$.

$(r_3 < q) \wedge (r_3 > l)$, $q = 1$; $i = 3, j = n$.

$c_1 = 1$; $p_3 = 1$; $l = 1$; $k < n, k = 2$.

С1. $q = 6$; $j = 1$.

С2. $(r_1 < q) \wedge (r_1 > l)$, $q = 2$; $i = 1, j < n, j = 2$.

$(r_2 > q), j < n, j = 3$.

$(r_3 < q) \wedge (r_3 = l), j = n$;

$c_2 = 2$; $p_1 = 2$; $l = 2$; $k < n, k = 2$.

С1. $q = 6$; $j = 1$.

С2. $(r_1 < q) \wedge (r_1 = l), j < n, j = 2$.

$(r_2 < q) \wedge (r_2 > l)$, $q = 5$; $i = 2, j < n, j = 3$.

$(r_3 < q) \wedge (r_3 < l), j = n$;

$c_3 = 5$; $p_2 = 3$; $l = 5$; $k = n$.

Свертка $P = (2, 3, 1)$ есть $z = 3$; $q = 3$.

Свертка $C = (1, 2, 5)$ есть $z = 2$.

$z = 15$.

Выход.

Таким образом, в результате проведенных вычислений получим, что размещению $R = (2, 5, 1)$ соответствует код $z = 15$.

Декодирование размещения. Восстановление элементов размещения по его коду предлагается осуществлять с использованием следующей процедуры развертки размещения.

Процедура *RR* (*m*, *n*, *z*, *R*).

Вход. Вычислить $w = n!$; $q = z$.

Вычислить $z = \text{ent}\,(z / w)$; $l = z$.

Выполнить процедуру развертки сочетания $RC\,(m, n, z, C)$.

Вычислить $z = q - l \times w$.

Выполнить процедуру развертки перестановки $RP\,(n, z, P)$.

Определить: $k = P_i$; $r_i = c_k$, $i = 1, 2, \cdots, n$.

Выход.

Для нашего примера по коду $z = 15$ развертка размещения на составляющие ее элементы производится следующим образом.

Вход. $w = 6$; $q = 15$;

$z = 2$; $l = 2$.

$C = (1, 2, 5)$;

$z = 3$.

$P = (2, 3, 1)$.

$i = 1$; $k = 2$; $r_1 = 2$;

$i = 2$; $k = 3$; $r_2 = 5$;

$i = 3$; $k = 1$; $r_3 = 1$.

Выход.

Таким образом, в соответствии с логической схемой процедуры РЗР получим, что коду $z = 15$ соответствует размещение $R = (2, 5, 1)$.

Заключение. Отметим следующие возможности практического приложения процедур кодирования и декодирования размещений [4, с. 73–75].

Если организовать последовательное обращение к процедуре декодирования с числами $z = 0, 1, \cdots, N - 1$, то получим полное множество лексикографически отличающихся друг от друга комбинаций размещений. Этот способ получения размещений может быть использован в методе лексикографического перебора [3, с. 33–38].

Если организовать формирование случайного числа z в диапазоне от 0 до $N - 1$, то после обращения к процедуре декодирования получим случайную комбинацию размещения. Этот прием получения комбинаций может быть использован в методе случайного поиска [2, с. 194–195].

Большой эффект использование процедур кодирования и декодирования размещений может дать в динамическом программировании [2, с. 150–153]. В общепринятом виде алгоритм динамического программирования состоит из двух этапов – прямого и обратного хода.

Наличие двух этапов в алгоритме динамического программирования предполагает организацию табулирования в памяти компьютера членов рекуррентных соотношений для всех n шагов прямого хода, что обусловливает емкостную сложность алгоритма [7]. Для снижения емкостной сложности алгоритма предлагается отказаться от табулирования и хранения членов функционального уравнения для всех n шагов прямого хода, оставляя в памяти значения членов $f_i(R_i)$ и соответствующих им кодов (сверток) формируемых размещений только для текущего i-го шага, $i = 1, \cdots, n$.

Тогда на n-м шаге оптимальному значению целевой функции $f_n(R_n^*)$ будет соответствовать полученный с помощью процедуры кодирования код размещения z^*. Обратный ход алгоритма для восстановления элементов размещения $R^* = (r_1^*, r_2^*, \cdots, r_n^*)$ заменяется обращением к процедуре декодирования. Это исключает необходимость просмотра членов рекуррентных соотношений n-го, $(n-1)$-го, ⋯, 1-го шагов, а, следовательно, отпадает надобность их табулирования и хранения в памяти компьютера при прямом ходе.

В методе ветвей и границ [8] процедуру кодирования размещений целесообразно применять для нумерации вершин дерева вариантов. Процедуру декодирования следует использовать для определения ветви, соединяющей корень дерева с концевой вершиной, где получено оптимальное значение целевой функции. При этом отпадает необходимость хранения в памяти компьютера ранее разветвленных вершин и указателей связи между ними в дереве вариантов, что снижает емкостную сложность реализуемого алгоритма.

Примечания:

1. Кнут Д. Искусство программирования, том 4, вып. 2. Генерация всех кортежей и перестановок. М.: ООО «И.Д. Вильямс», 2008. 160 с.
2. Рейнгольд Э., Нивергельт Ю., Део Н. Комбинаторные алгоритмы. Теория и практика. М.: Мир, 1980. 478 с.
3. Романовский И.В. Дискретный анализ. СПб.: Невский диалект, 2000. 240 с.
4. Бабаев А. Модели административного управления. Методы и алгоритмы оптимизации. – Saarbrücken: LAP LAMBERT Academic Publishing, 2012. 330 с.
5. Бабаев А.А. Процедуры кодирования и декодирования перестановок // Кибернетика. 1984. № 6. С. 75–76.
6. Бабаев А.А. Процедуры кодирования и декодирования сочетаний // Кибернетика. 1989. № 5. С. 120–122.
7. Бабаев А.А. Оценка сложности комбинаторных алгоритмов // Материалы международной научной конференции «Экономическое развитие: теория и практика». Секция 3–7. СПб.: ОЦЭиМ, 2007. С. 25–26.
8. Бабаев А.А. Организация поиска решений на деревьях детерминированной структуры // Электронное моделирование. 1985. № 1. С. 19–25.

УДК 519.7

Процедуры кодирования и декодирования размещений

Александр Александрович Бабаев

Санкт-Петербургский государственный университет, Россия
195027, Санкт-Петербург, Среднеохтинский проспект, д. 8, кв. 68
Кандидат технических наук, доцент
E-mail: a.babaev@hotmail.com

Аннотация. Рассматривается алгоритм представления набора элементов размещения в виде одного числа на основе процедуры кодирования. Предлагается процедура обратного преобразования заданного кода в элементы исходного размещения. Даются рекомендации по использованию приведенных процедур в комбинаторных алгоритмах оптимизации на размещениях.

Ключевые слова: алгоритм; декодирование; кодирование; комбинаторика; перестановка; процедура; размещение; сочетание; элементы размещения.