Используя правила перевода из одной системы счисления в другую, получим двоичный эквивалент десятичного числа, равный 100110100. Полученное двоичное число соответствует биномиальному 4110.

Таким образом, в настоящей статье предложен быстродействующий метод преобразования позиционных кодов в биномиальные с многозначным алфавитом.

SUMMARY

The algorithm transforming positional codes into multiple valued binomial ones is proposed. Its feature is high speed when coding the numbers of a large digit length. It may be applied to development of the combinatorial codes by scheme or program ways.

СПИСОК ЛИТЕРАТУРЫ

1.Толстяков В.С. и др. Обнаружение и исправление ошибок в дискретных устройствах. - М.: Сов. радио, 1972. - 288с.

2.А.с.1547071. Преобразователь кодов. Борисенко А.А., Соловей В.А., Мирошниченко В.М. - Опубл. 28.02.90, БИ N8.

3.Борисенко А.А. Об одной системе счисления с биномиальным основанием. - Рук. деп. в ВИНИТИ, 1982.-N874-82.

4.Самофалов К.Г., Романкевич А.М., Валуйский В.Н. и др. Прикладная теория цифровых автоматов. - К.: Выща школа, 1987.-375с.

5.Борисенко А.А., Губарев С.И.,Вулаенко С.И. Об одном способе преобразования чисел из позиционного кода в систему остаточных классов. - АСУ и приборы автоматики : Респ. межвед. науч.-техн. сб. - Харьков : Выща школа , ХГУ, 1976. - Вып. 40.

Поступила в редколлегию 29 сентября 1995 г.

УДК 681.32

К ВЫЧИСЛЕНИЮ ЧИСЛА СОЧЕТАНИЙ

Арбузов В. В., асп., Кулик И. А., асп., Бережная О. В., асп.

Целый ряд аппаратных средств, таких, как устройства нумерационного кодирования на основе биномиальной системы счисления, устройства контроля канала связи и устройства сжатия, требуют для своего функционирования подсчет числа сочетаний C_n^k [1,2]. При этом актуальными являются задачи нахождения точных целочисленных значений C_n^k и обеспечения максимального быстродействия вычислительных алгоритмов. Решение этих задач на уровне программно-аппаратной реализации для относительно больших значений n и k сталкивается со следующими трудностями. Во-первых, громоздкость вычислений, связанная с подсчетом факториалов чисел; во-вторых, сложность представления конечного результата, достигающего во многих случаях больших значений (например, $C_{500}^{250} \approx 1,17 \times 10^{149}$). Существующие способы подсчета числа сочетаний или имеют оценочный характер, или требуют больших аппаратурных и временных затрат [3, 4]. Значительное снижение этих затрат при построении эффективного специализированного вычислительного устройства связано с изучением свойств самой функции C_n^k и особенностей ее вычисления, учет которых позволит уменьшить априорную неопределенность вычислительной процедуры. Это приводит к уменьшению количества информации, получаемой в процессе вычисления, и предоставляет возможность экономии аппаратурных и временных ресурсов для ее обработки, хранения и передачи.

Поэтому для подсчета точного целочисленного значения C_n^k при больших значениях n и k необходимо решить следующие задачи:

1. Найти рациональную форму представления точного целочисленного значения C_n^k для больших n и k, позволяющую уменьшить разрядную сетку результата.

2. На основе изучения свойств функции C_n^k разработать алгоритмы, сводящие к минимуму общее количество арифметических операций при подсчете C_n^k и обеспечивающие эффективное кодирование параметров n, k функции и ее значения.

Проведенные исследования показали, что для решения этих задач можно использовать каноническую форму записи чисел n, k и C_n^k.

Известно, что любое целое число z может быть представлено как произведение простых чисел [5]:

$$z = q_1^{a_1} q_2^{a_2} \dots q_j^{a_j} \dots q_m^{a_m}, \tag{1}$$

где - j-й элемент ряда простых чисел, $j=1, 2, \dots, m$;

a_j - показатель степени простого числа q_j;

m - количество простых чисел, необходимых для представления числа z.

Такая форма записи числа z, в соответствии с основной теоремой арифметики [5], является единственной и позволяет однозначно представлять любое целое число с точностью до порядка множителей.

Таким образом, существует взаимно-однозначное отображение f_k между множеством Z целых чисел и множеством соответствующих векторов показателей степени

$V = \{A_1, A_2, \dots, A_i, \dots\}$, где $A_i = (a_{1i}, a_{2i}, \dots, a_{mi})$:

$$f_k : Z \to V .$$

Представление любых целых чисел z_i и z_j из множества Z в виде векторов A_i и A_j позволяет трудоемкие операции умножения и деления над числами z_i и z_j заменить простыми операциями сложения и вычитания над их соответствующими векторами A_i и A_j.

Согласно [6], объем временных затрат, необходимый для умножения и для сложения двух r-разрядных чисел, пропорционален r^2 и r соответственно. Отсюда следует, что и время, необходимое для суммирования векторов степени A_i и A_j, будет меньше в r раз времени, необходимого для умножения соответствующих им чисел z_i и z_j, что значительно снижает временные затраты. Например, при умножении чисел z_i и z_j, каждое из которых равно C_{100}^{50} и требует для своего двоичного кодирования $r=97$ разрядов, временные затраты при использовании канонической формы записи на вычисление выражения

$$z_i \cdot z_j = (A_i + A_j) = ((a_{1i} + a_{1j}), (a_{2i} + a_{2j}), \dots, (a_{mi} + a_{mj}))$$

будут в $r=97$ раз меньше, чем при обычном умножении чисел z_i и z_j.

Значительная часть аппаратурных затрат при построении устройств подсчета C_n^k, при достаточно больших n и k, связана с его двоичным представлением. Как уже отмечалось, при $n=100$ и $k=50$ для хранения значения числа сочетаний C_{100}^{50} понадобилось бы $\log_2 C_{100}^{50} \approx 97$ разрядов. Использование особенностей подсчета и свойств самого числа сочетаний позволяют не только снизить время вычислений, но и перейти к его более рациональному виду. Это показывают сформулированные ниже утверждения 1 и 2.

Утверждение 1. Для представления числа сочетаний в канонической форме $C_n^k = q_1^{a_1} q_2^{a_2} \dots q_j^{a_j} \dots q_m^{a_m}$ достаточен набор простых чисел q_j, значения которых не больше n, т. е. $q_j \le n$ при $j=1, \dots, m$.

Доказательство. Очевидно, что данное утверждение может считаться доказанным, если удастся показать, что процедура получения числа C_n^k определяется функцией $C_n^k = f(n,k)$, характер операций с числами n и k которой не приводит к получению в выражении (1) простых чисел $q_i > n$. Допустим $q_1 < q_2 < \dots < q_j < \dots < q_m$. Исходя из определения, что простыми числами являются целые положительные числа $q_j > 1$, делители которых исчерпываются числами q_j и 1 [6], следует, что невозможно из множества $\{q_1, q_2, \dots, q_j, \dots, q_m\}$ с помощью только операций умножения и деления получить хоть одно простое число $q_{m+1} > q_m$.

Анализ выражения для подсчета числа сочетаний

$$C_n^k = f(k,n) = \frac{n!}{k!(n-k)!} = q_1^{a_1} q_2^{a_2} \dots q_m^{a_m} \quad (2)$$

показывает, что перечень операций с числами в функции (2) для получения каждого элемента $q_j^{a_j}$ ограничивается только операциями умножения и деления над числами менее n, что не может привести к появлению в канонической форме простых чисел $q_j > n$.

Утверждение 2. Число сочетаний C_n^k в канонической форме записи содержит, начиная с $q_j \ge n-k+1$ для $k \le n/2$ и с $q_j \ge k+1$ для $k < n/2$, весь ряд простых чисел, для каждого из которых $a_j = 1, j = 1, 2, \dots, m$.

Доказательство. Рассмотрим доказательство утверждения для $k \le n/2$. При этом выражение (2) представим в следующем виде:

$$C_n^k = \frac{(n-k+1)(n-k+2)\dots n}{1 \cdot 2 \dots k}. \quad (3)$$

Нетрудно заметить, что множество $R = \{r_\nu; n-k+1 \le r_\nu \le n, \nu = 1, 2, \dots, k\}$ сомножителей числителя состоит из двух подмножеств, элементы которых подчинены отношению строгого порядка. Первое подмножество $Q = \{q_j; n-k+1 \le q_j \le n\}$ содержит только простые числа q_j. Все остальные элементы множества R, не входящие в Q, образуют множество S, каждый элемент которого является целым положительным числом, который можно разложить на простые сомножители в виде (1). Очевидно, что S является разностью R/Q. Совокупность сомножителей знаменателя выражения (3) образует множество $Y = \{1, 2, \dots, k\}$.

Для того чтобы утверждать, что, начиная с определенного простого числа q_j в разложении (1) участвуют все простые числа от n-k+1 до n включительно, необходимо показать, что в процессе преобразования выражения (3) к виду (1) не происходит взаимного сокращения элементов множеств Q и Y, множества не содержат общих элементов и являются непересекающимися, т.е. $Q \cap Y = \varnothing$. Очевидно, что для выполнения этого требования достаточно, чтобы $\min Q > \max Y$, т.е. n-k+1>k. После преобразования получим $(n+1)/2 > k$. Данное неравенство соблюдается при изменении k от 1 до $n/2$. Учитывая, что доказательство утверждения производится для $k \le n/2$, можно считать доказанной первую часть утверждения 2 о полноте ряда простых чисел $q_j \in Q$.

При доказательстве второй части утверждения, что каждое простое число $q_j \in Q$ в разложении (1) имеет показатель степени $a_j = 1$, достаточно доказать, что показатель степени a_j не может быть больше 1, т.к. в процессе преобразования выражения (3) к виду (1) в множестве S не найдется такого элемента, в разложении которого участвует хоть один элемент $q_j \in Q$, т.е. в его разложении участвуют простые числа $q_j < \min Q$.

Если при разложении $\hat{s} = \max S$ это требование удастся доказать для двух сомножителей с $a_j = 1$, один из которых является наименьшим простым числом $q_1 = 2$, тем более это будет справедливо для всех остальных элементов множества S, для большего числа сомножителей, наименьший из которых может быть больше двух, и для $a_j > 1$.

Допустим $\max S = q_1 q_j = 2q_j$. Так как справедливо неравенство $\max S \le n$, то $q_j \le n/2$. Исходя из того, что $\min Q = n\text{-}k+1$, то для любого $k = 1, \ldots, n/2$ всегда $\min Q \le (n/2)+1$. Таким образом, всегда $q_j < \min Q$, что и требовалось доказать.

Аналогично производится доказательство утверждения 2 и для случая $k > n/2$.

Приведем пример двоичного представления числа C_{100}^{50} в канонической форме. В соответствии с утверждением 1 его разложение должно содержать простые числа в пределах от 2 до 100, количество которых в этом диапазоне равно 25. При этом по утверждению 2 простые числа $q_i \ge 50$ должны обязательно иметь единичную степень. Действительно,

$$C_{100}^{50} = 2^4 \times 3^4 \times 11 \times 13 \times 17 \times 19 \times 29 \times 31 \times 53 \times 59 \times 61 \times 67 \times$$
$$\times 71 \times 79 \times 83 \times 89 \times 97 .$$

Закодируем показатели степени двоичным алфавитом, представляя при этом отсутствующие простые числа логическими нулями. Согласно утверждению 2 конечную часть разложения числа C_{100}^{50}, содержащую заведомо известные простые числа $q_j \ge n-k+1$ при $k \le n/2$ или $q_j \ge k+1$ при $k > n/2$ с показателями степени равными единице, можно закодировать двоичным словом, обозначающим их количество. В результате двоичный эквивалент числа C_{100}^{50} имеет вид: 100.100.0.0.1.1.1.1.0.1001. Для его записи необходимы 17 двоичных разрядов в отличие от обычного двоичного кодирования, требующего 97 разрядов.

В заключение можно сделать вывод, что при построении специализированных устройств подсчета C_n^k при больших n и k рациональным с точки зрения уменьшения аппаратурных и временных затрат есть переход от обычного двоичного формата чисел к двоичной канонической форме их записи. Целесообразным при этом является изучение свойств самого числа сочетаний и особенностей процедуры его вычисления, что позволит найти эффективные схемотехнические решения.

SUMMARY

It is proposed to use the canonical number record form when computing the number of combinations. In the case of scheme or program realization it is able to reduce hardware and time expenses greatly when n and k are large. The proved out assertions on the properties of the canonical form for the combinations and the example are given.

СПИСОК ЛИТЕРАТУРЫ

1. Борисенко А.А. Алгоритмы построения кодов с постоянным весом на основе биномиальных чисел. АСУ и приборы автоматики: Респ. межвед. науч.-техн. сб., Харьков, Изд-во "Высшая школа", ХГУ, 1985, вып. 73, с. 78-80.
2. Березюк Н. Т и др. Кодирование информации (двоичные коды). - Харьков: Изд-во "Высшая школа", 1978. - 252 с.

3. Кнут Д. Искусство программирования для ЭВМ. Москва,Изд-во Мир, 1976. - 732с.
4. Райзер Г. Дж. Комбинаторная математика. Москва, Изд-во Мир, 1966. - 153с.
5. Кострикин А.И. Введение в алгебру. - Москва, Изд-во Наука, 1977. - 496с.
6. Самофалов К. Г. и др. Прикладная теория цифровых автоматов. - Киев, Изд-во Высшая школа, 1987. - 375с.

Поступила в редколлегию 9 октября 1995г.

УДК 621.394

СИНТЕЗ СИСТЕМЫ ОЦЕНКИ ПЕРЕХОДНОЙ МАТРИЦЫ СОСТОЯНИЙ НЕСТАЦИОНАРНЫХ ОБЪЕКТОВ УПРАВЛЕНИЯ

Володченко Г.С., проф., Новгородцев А.И., ст. преп., Полонский А.Д., ст. преп.

При разработке адаптивных систем управления нестационарными объектами возникает необходимость автоматического учета текущей информации о переходной матрице состояния объекта управления, т.е. учет его динамических характеристик нестационарных объектов управления в процессе их нормальной эксплуатации возможен с применением в контуре управления специализированных вычислительных машин.

Постановка задачи. Пусть нестационарный объект управления в общем случае описывается дифференциальным уравнением вида

$$a_m(t)\frac{d^m x(t)}{dt^m} + a_{m-1}(t)\frac{d^{m-1}x(t)}{dt^{m-1}} + \ldots + a_1(t)\frac{dx(t)}{dt} + a_o(t)x(t) = b(t)U(t), \qquad (1)$$

а $U(t)$ имеет вид ступенчатой функции.

Тогда, пользуясь методом переменных состояний при $t>0$, имеем:

$$\begin{aligned}
&\dot U = 0,\\
&\dot x_1 = x_2,\\
&\dot x_2 = x_3,\\
&\ldots\ldots\ldots\ldots,\\
&\dot x_{m-1} = x_m,\\
&\dot x_m = \frac{b(t)}{a_m(t)}U - \frac{a_{o}(t)}{a_m(t)}x_1 - \ldots - \frac{a_{m-1}(t)}{a_{m}(t)}x_m,
\end{aligned} \qquad (2)$$

откуда имеем

$$V = \begin{bmatrix} U \\ x_1 \\ x_2 \\ \ldots \\ x_m \end{bmatrix} \qquad (3)$$

- расширенный вектор состояния;

$$A(t) = \begin{bmatrix} 0 & 0 & 0 & \ldots & 0 \\ 0 & 1 & 0 & \ldots & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & 0 & \ldots & 1 \\ \frac{b(t)}{a_m(t)} & -\frac{a_o(t)}{a_m(t)} & -\frac{a_1(t)}{a_m(t)} & -\ldots & -\frac{a_{m-1}(t)}{a_m(t)} \end{bmatrix}$$

- расширенная переменная матрица коэффициентов.

Рассматривая решение задачи синтеза алгоритма системы оценки переходной матрицы состояния как итерационный процесс, нахождение последнего будем искать в классе стационарных систем. Предполагая квазистационарность объекта управления на интервале $nT_1 \le t \le (n+1)T_1$,