Отметим, что вместо $\{\widehat{\alpha}_{ij}, \widehat{\beta}_{ij}\}$ могут быть использованы более точные оценки максимального правдоподобия.

Подстановка $\{\widehat{\alpha}_{ij}, \widehat{\beta}_{ij}\}$ в (16), (17) и далее в (13) приводит к оценкам $\{\hat{q}_i(\cdot)\}$. Используя теперь $\{\widehat{Q}_i(\cdot), \hat{q}_i(\cdot)\}$ в (10), получаем адаптивное решающее правило:

$$d = d_1(x,\ A) = \arg\max_{i \in D} (\ln \pi_i + \ln \widehat{q}_i(u) + \ln \widehat{Q}_i(v)). \qquad (20)$$

Это правило оказывается состоятельным и может быть исследовано методом асимптотических разложений [9].

Таблица эффективности адаптивного решающего правила

N_2 \ N_1	3	4	5	6	7	8	9
1	0,110	0,073	0,063	0,050	0,053	0,070	0,063
2	0,133	0,087	0,060	0,053	0,057	0,073	0,073
3	0,110	0,070	0,043	0,043	0,050	0,060	0,070
4	0,093	0,063	0,053	0,050	0,057	0,063	0,070
5	0,103	0,073	0,067	0,070	0,070	0,087	0,083

Численный пример. Исследование эффективности решающего правила (20) проводилось на ЭВМ в случае шести равновероятных классов ($L=6$, $\pi_i=1/6$) узкополосных псевдослучайных сигналов, описанных в [2]; размерность пространства наблюдения $N=60$. По обучающим выборкам объема $n_i=50$ построено решающее правило (20). По независящей от A экзаменационной выборке объемом 300 (по 50 реализаций из каждого класса) оценивалась вероятность p ошибочного решения. В таблице представлены точечные оценки $p=p(N_1, N_2)$ в зависимости от числа инвариантных признаков.

ЛИТЕРАТУРА

1. Андерсон Т. Статистический анализ временных рядов.— М., 1980.
2. Омельченко В. А., Матевицкий Е. О.— Изв. вузов СССР. Радиоэлектроника, 1979, т. 22, № 12, с. 16.
3. Сенин А. Г. Распознавание случайных сигналов.— Новосибирск, 1974.
4. Кокс Д. Р.— В сб.: Распознавание образов при помощи цифровых вычислительных машин. М., 1977.
5. Файн В. С. Опознавание изображений.— М., 1970.
6. Леман Э. Проверка статистических гипотез.— М., 1979.
7. Харин Ю. С.— Изв. АН СССР. Техн. кибернетика, 1977, № 5, с. 158.
8. Дэйвид Г. Порядковые статистики.— М., 1979.
9. Харин Ю. С.— В кн.: Теория вероятностей и ее применение, 1983, т. 28, вып. 3, с. 592.

Поступила в редакцию
24.02.84.

УДК 33.083.722

М. К. БУЗА

АНАЛИЗ ЭФФЕКТИВНОСТИ КОДИРОВАНИЯ ДАННЫХ С ПОМОЩЬЮ ВЫЧЕТОВ

Эффективность обработки информации на ЭВМ, базирующейся на модулярной арифметике, и сложность основных ее аппаратурных компонент во многом определяет выбор модулей системы в коде вычетов. Из общих соображений можно предположить, что оптимальной в плане

простоты аппаратуры и алгоритмов выполнения одно- и многосвязных операций будет система модулей, требующая наименьшего числа малоразрядных оснований при фиксированном диапазоне M. Оказывается, что это не так. Выясним некоторые условия, которые следует налагать на модули для повышения эффективности кодирования и обработки данных в системе в коде вычетов (СКВ).

Основные обозначения: $P_1, P_2, \ldots, P_n$ — набор модулей СКВ; $m_1, m_2, \ldots, m_n$ — веса ортогональных базисов; r_A — ранг числа A; W — область определения операций при заданном наборе модулей.

Разработка алгоритмов выполнения операций в СКВ показывает, что основным требованием, предъявляемым к модулям системы, является однозначность кодирования данных. Получим условия, налагаемые на модули для удовлетворения указанному требованию.

Пусть для некоторых $P_1, P_2, \ldots, P_n$ выполняются условия

$$P_j = k_j P_s, \tag{1}$$

где $j \in N$, N — множество номеров всех модулей, $k_j \geqslant 2$, $j \neq s$. Не нарушая общности рассуждений, предположим, что $j = 1, 2, \ldots, j_0$; $j_0 < n$, $s = j_0+1, j_0+2, \ldots, n$. Тогда в силу (1) $A \equiv A + \prod_{j=1}^{j_0} P_j \pmod{P_i}$, $i = 1, 2, \ldots, n$ для всех A и $A + \prod_{j=1}^{j_0} P_j$ из области определения, т. е. коды указанных чисел в выбранной системе модулей будут одинаковы. Отсюда следует, что однозначность кодирования данных требует, чтобы для всех $i, j \in N$ и $i \neq j$ было $(P_i, P_j) = 1$.

Наиболее простые алгоритмы немодульных операций позволяет построить безранговая система в коде вычетов (БСКВ). Однако БСКВ накладывает ограничения на модули [1]. Требуется, чтобы они дополнительно удовлетворяли условию $r_A = \Sigma m_i, \forall A \in W, i \in N$, где N — множество, которое образуют номера всех весов ортогональных базисов. Оптимальные алгоритмы по быстродействию в БСКВ можно построить для наборов модулей, для которых веса ортогональных базисов $m_i = 1$ для $i = 1, 2, \ldots, n$.

Как показано в [2], наибольшую экономию в аппаратуре при проектировании ЭВМ в БСКВ получим, если набор модулей удовлетворяет условиям

$$\prod_{i=1}^{n/2} P_i + 1 = \prod_{i=n/2}^{n} P_i, \tag{2}$$

где $n = 2^k$, k — положительное целое число. Более того, исследования показали, что такие основания позволяют значительно упростить алгоритмы округления в БСКВ. Однако условие (2) кажется сильно ограничивающим набор модулей, но машинный эксперимент показал, что даже для $P_i \in [2, 2^7]$ при $n = 8$ таких наборов модулей около 10^4.

Пусть l_i — количество двоичных разрядов, необходимое для кодирования числа $P_i - 1$, $i = 1, 2, \ldots, n$, а $\delta = \sum_{i=1}^{n} (2^{l_i} - P_i)$. Тогда δ характеризует сложность вычислительного устройства, работающего по всему набору модулей. Значит, модули надо выбирать таким образом, чтобы δ было минимальным. Однако δ будет всегда больше нуля, ибо в силу однозначности кодирования данных $\log_2 M \leqslant \sum_{i=1}^{n}]\log_2 P_i[$.

При построении модульных сумматоров, если модули удовлетворяют условиям (2), возникает необходимость во введении большого числа обратных связей, что ведет к усложнению алгоритмов операций в СКВ и, следовательно, к удорожанию реализующих их вычислительных

устройств. Для сведения к минимуму количества обратных связей, а следовательно, к уменьшению δ, целесообразно модули выбирать равными степени двойки и степени двойки без единицы. В этом случае модульный сумматор будет иметь не более одной обратной связи. Более того, вычислитель в СКВ может быть получен из соответствующего двоичного вычислителя с незначительными перестройками, а именно: все регистры необходимо разделить в смысле управления на число сегментов, соответствующих количеству модулей в СКВ. Длина сегмента определяется двоичной кодировкой максимального числа по соответствующему модулю. В этом случае каждый из сегментов работает как отдельный регистр, с той лишь разницей, что при работе по модулю 2^k перенос из старшего двоичного разряда не используется, так как $2^k \equiv 0 \pmod{2^k}$, а при выполнении операции по модулю 2^k-1 такой перенос зацикливается в младший разряд, ибо $2^k \equiv 1 \pmod{2^k-1}$. Здесь, естественно, используется непозиционно-позиционный подход, при котором ЭВМ работает в СКВ, но внутриразрядные операции выполняются по правилам обычной двоичной арифметики.

Пусть число A в двоичной форме имеет вид

$$A = a_{nk}a_{nk-1}\ldots a_{2k}a_{2k-1}\ldots a_k a_{k-1}\ldots a_0. \tag{3}$$

Пусть $P_i = 2^k-1$. Предположим, что k младших двоичных цифр в (3) образуют число D_1, k следующих цифр — число D_2 и т. д. Тогда (3) перепишем в виде $A = 2^0 D_1 + 2^k D_2 + 2^{2k} D_3 + \ldots$ Но $|2^{lk}|_{2^k-1} \equiv 1$ при любом целом неотрицательном k. Следовательно,

$$|A|_{2^k-1} = |D_1 + D_2 + D_3 + \ldots|_{2^k-1}. \tag{4}$$

Таким образом, определение вычета двоичного числа по модулю 2^k-1 сводится к сложению нескольких k-разрядных двоичных чисел. Следует, однако, отметить, что при вычислении вычета по модулю 2^k-1 необходимо на последнем шаге быть осторожным, ибо, если $\Sigma D_i = 2^k-1$ из-за того, что перенос в k-ый разряд не происходит, получим некорректный результат. В связи с этим необходимо произвести зацикливание единицы из старшего разряда в младший либо воспользоваться соответствующим модульным вычитателем для $P_i = 2^k-1$, как это оговорено выше.

Отметим, что аналогично можно получить вычет по модулю $P_i = 2^k+1$. Выражение (4) в этом случае примет вид

$$|A|_{2^k+1} = |D_1 - D_2 + D_3 - \ldots|_{2^k+1}.$$

При работе с рациональными числами код числа A/B есть наименьшее неотрицательное решение системы линейных сравнений

$$Bx_i \equiv A \pmod{P_i}, \tag{5}$$

где $i = 1, 2, \ldots, n$. В общем случае выбранная система модулей $P_1, P_2, \ldots, P_n$ может оказаться избыточной, т. е. не все возможные комбинации вычетов используются для кодирования рациональных чисел. Избыточность системы зависит не только от модулей, но и от интервала изменения величин A и B.

Пусть L — множество рациональных чисел, имеющих однозначное кодирование в заданной системе модулей. Модули $P_1, P_2, \ldots, P_n$ при фиксированных A и B (или A и B при фиксированных $P_1, P_2, \ldots, P_n$) следует определить так, чтобы $W \cap L = \varnothing$.

Наибольший практический интерес представляет случай, когда $A \in [1, B-1]$. Рассмотрим его. Так как модули $(P_i, P_j) = 1$, $i \neq j$; $i, j = 1, 2, \ldots, n$, то в силу свойства сравнений по модулю, равному НОК $(P_1, P_2, \ldots, P_n)$, из (5) имеем, что справедливо сравнение

$$BX \equiv A \pmod{M}. \tag{6}$$

Пусть $(B, M) = 1$, тогда решение (6) суть

$$X \equiv 1/B\,(A + kM) \pmod{M}, \tag{7}$$

где k из полной системы вычетов по модулю B. Выражение (7) будет и решением для (5). При $A=1$ получим, что $X\equiv 1/B\,(1+kM)\,(\mathrm{mod}\,M)$.

С другой стороны, пусть $A=t$, где $1\leqslant t\leqslant B-1$, тогда общее решение системы (6) примет вид

$$X\equiv 1/B\,(1+kM)\,t\,(\mathrm{mod}\ M). \qquad (8)$$

Формула (8) дает решение сравнения (6). Действительно, подставив (8) в (6), получим $B\ \ 1/B(1+kM)t\equiv A\,(\mathrm{mod}\ M)$, т. е. $t\equiv A\,(\mathrm{mod}\ M)$. Значит, X — решение сравнения (6).

Из (8) следует, что два рациональных числа со знаменателями B_1 и B_2 будут иметь различные коды при заданной системе модулей, если только $X_1\not\equiv X_2\,(\mathrm{mod}\,M)$, где

$$X_1\equiv 1/B_1(1+k_1M)\,t_1\,(\mathrm{mod}\,M),\quad X_2\equiv 1/B_2(1+k_2M)\,t_2\,(\mathrm{mod}\,M),$$
$$t_1\in[1,\ B_1-1],\qquad t_2\in[1,\ B_2-1]. \qquad (9)$$

Пусть τ — наименьшее неотрицательное решение сравнения $BX\equiv 1\times$ $\times\,(\mathrm{mod}\,M)$, тогда очевидны следующие утверждения.

Утверждение 1. $(M,\tau)=1$.

Утверждение 2. Если $(B,\ M)=1$ и $B<M$, то $(M-B,\ M)=1$ и $(B,\ M-B)=1$.

Из приведенных рассуждений следует

Утверждение 3. Если для выбранной системы модулей задать рациональные числа с такими знаменателями B_1 и B_2, что $B_1<M$, $(B_1,\ M)=1$, $B_2=M-B_1$, то $W\cap L=\varnothing$ и $X_1\not\equiv X_2\,(\mathrm{mod}\,M)$, где X_1 и X_2 из (9).

Доказательство. 1. $W\cap L=\varnothing$ следует из того, что, кроме $A\in[1,\ B-1]$, рассматриваем и числа, равные 1 и 0.

2. Покажем, что $X_1\not\equiv X_2\,(\mathrm{mod}\,M)$. Допустим обратное, т. е. $X_1\equiv$ $\equiv X_2\,(\mathrm{mod}\,M)$ при некоторых t_1 и t_2 из (9). Тогда получим

$$1/B_1(1+k_1M)\,t_1\equiv 1/B_2(1+k_2M)\,t_2\,(\mathrm{mod}\,M). \qquad (10)$$

Умножив (10) на B_2, получим $(M-B_1+k_1M(M-B_1))\,\frac{t_1}{B_1}\equiv t_2\,(\mathrm{mod}\,M)$, т. е. $M(1+k_1M-k_1B_1)\,\frac{t_1}{B_1}-t_1\equiv t_2\,(\mathrm{mod}\,M)$. Умножив полученное сравнение на B_1, имеем $B_1t_1+B_1t_2\equiv 0\,(\mathrm{mod}\,M)$. Но так как $(B_1,\ M)=1$, то $t_1+t_2\equiv 0\,(\mathrm{mod}\,M)$. Но $0<t_1+t_2<M-1$, следовательно, t_1+t_2 не может делиться на M. Получили противоречие.

Из утверждения 3 следует, что модули всегда можно выбрать таким образом, чтобы система не была избыточной.

Синхронизация начала обработки данных по всем модулям требует сокращения времени ожидания обработки по максимальному модулю. Следовательно, для ускорения обработки необходимо модули выбирать таким образом, чтобы при фиксированном W разность между величинами минимального и максимального модулей была наименьшей.

ЛИТЕРАТУРА

1. Буза М. К.— Вычислительная техника в машиностроении, 1974, АН БССР. Минск, вып. III 34, с. 96.

2. Driese E. C.— USAF Aeronautical Sys. Div. Tech. Rept., 1961, Dayton, p. 61.

Поступила в редакцию
25.02.84.