О некоторых направлениях повышения быстродействия алгоритма сравнения чисел в системе остаточных классов

Ю. Д. ПОЛИССКИЙ

Украина, НИИ автоматизации черной металлургии

Рассмотрены некоторые алгоритмические пути повышения быстродействия операции сравнения чисел в системе остаточных классов.

Розглянуті деякі алгоритмічні напрямки підвищення швидкодії операції порівняння чисел в системі залишкових класів.

Some algorithmic ways of increase of fast-acting of operation of comparison of numbers are considered in the system of residual classes.

Введение. В настоящее время значительное количество работ посвящено вопросам поиска путей повышения эффективности вычислений на основе представления чисел в системе остаточных классов (СОК) [1]. СОК называется система счисления, в которой произвольное число N представляется в виде набора наименьших неотрицательных остатков по модулям $m_1, m_2, \ldots, m_n$.

$$N = \left(N\left(\mathrm{mod}\, m_1\right), N\left(\mathrm{mod}\, m_2\right), \ldots, N\left(\mathrm{mod}\, m_n\right)\right)$$

или $N = \left(\alpha_1, \alpha_2, \ldots, \alpha_n\right)$.

Здесь $\alpha_i = N\left(\mathrm{mod}\, m_i\right)$. При этом, если все целые числа N принадлежат диапазону $[0, M)$, объем которого равен $M = \prod_{i=1}^{n} m_i$ а модули m_i взаимно простые, то каждому набору $\left(\alpha_1, \alpha_2, \ldots, \alpha_n\right)$ соответствует только одно число N из этого диапазона.

Преимущества СОК [2] по сравнению с позиционными системами счисления заключаются в высокой степени параллелизма при выполнении арифметических операций сложения, вычитания и умножения, которые производятся над остатками независимо друг от друга по простым правилам, а также малоразрядность остатков, высокая точность, способность системы к самокоррекции. Однако возникают серьезные трудности при реализации немодульных операций, для выполнения которых необходимо знание цифр операндов по всем разрядам. К таким операциям относятся, в частности, определение принадлежности числа данной половине диапазона, деление чисел, определение четности числа в системе нечетных модулей СОК, сравнение чисел, определения позиционных характеристик числа.

Состояние вопроса. Между немодульными операциями существуют определенные взаимосвязи [3]. Поэтому, получив решение одной из операций, можно найти решения остальных.

Результаты исследований, представленные в [4], показали, что в основу всех немодульных операций может быть положен алгоритм сравнения чисел.

Основная часть. Пусть $N_1 = \left(\alpha_1 \alpha_2, \ldots, \alpha_n\right)$ и $N_2 = \left(\beta_1 \beta_2, \ldots, \beta_n\right)$ – сравниваемые числа. Необходимо определить результат

$$R = \begin{cases} R_1, N_1 > N_2 \\ R_2, N_1 < N_2 \end{cases}$$

Пусть системой оснований полиадического кода также является система модулей $m_1, m_2, \ldots, m_n$. Тогда число N в полиадическом коде представляется следующим образом:

$$N = \pi_1 + \pi_2 m_1 + \ldots + \pi_i m_1 m_2 \ldots m_{i-1} + \cdots +$$

$$+ \pi_{n-1} m_1 m_2 \ldots m_{n-2} + \pi_n m_1 m_2 \ldots m_{n-1},$$

где π_i -позиционная характеристика i-го разряда, $\pi_i = 0, 1, 2, \ldots, m-1$, $i = 1, 2, \ldots, n$.

В настоящее время известны два подхода к сравнению чисел. По первому для каждого из сравниваемых чисел вычисляются позиционные характеристики, после чего осуществляется поразрядное сравнение этих характеристик традиционными методами сравнения. Второй подход, впервые предложенный в [5] и развитый в работе [6], основан на вычислении «внутренних» характеристик - приведенных остатков сравниваемых чисел. Блок-схема данного алгоритма представлена на *рис.1*.

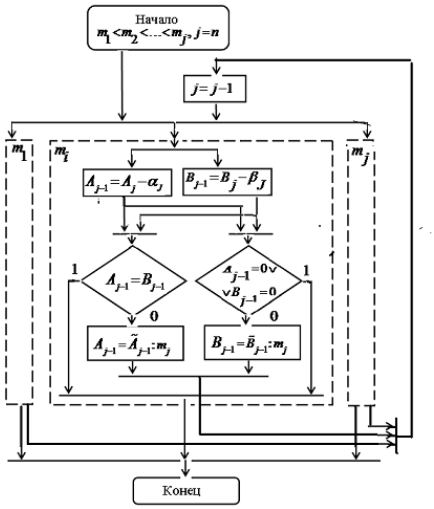

Рис.1. Блок-схема алгоритма сравнения при втором подходе

Если для каждого $N\left(\alpha_1,\alpha_2,\ldots,\alpha_n\right)\in M-1, M=\prod_{i=1}^{n} m_i$ составить разности $\tilde{\alpha}_i=\left(\alpha_i-\alpha_n\right)\left(\mathrm{mod}\, m_i\right)$, $i=1,2,\ldots,n-1$, то весь диапазон M окажется разбитым на $K=\dfrac{M}{m_n}$ поддиапазонов длины m_n, внутри каждого из которых значения $\tilde{\alpha}_i$ одинаковы. Большим (меньшим) из чисел поддиапазона при этом является число с большим (меньшим) значением α_n.

Составим для j-й итерации разности

$$\tilde{N}_1^{\,j}=N_1^{\,j-1}-\tilde{\alpha}_{n\;j-1}^{\,j-1}=\left(\tilde{\alpha}_{1\;j-1}^{\,j},\ldots,\tilde{\alpha}_{i\;j-1}^{\,j},\ldots,\tilde{\alpha}_{(n-1)\;j-1}^{\,j}\right)$$

$$\tilde{N}_1^{\,j}=N_1^{\,j-1}-\tilde{\beta}_{n\;j-1}^{\,j-1}=\left(\tilde{\beta}_{1\;j-1}^{\,j},\ldots,\tilde{\beta}_{i\;j-1}^{\,j},\ldots,\tilde{\beta}_{(n-1)\;j-1}^{\,j}\right),$$

где $\tilde{\alpha}_{i\;j-1}^{\,j}=\left(\tilde{\alpha}_{i\;j-1}^{\,j-1}-\tilde{\alpha}_{n\;j-1}^{\,j-1}\right)\left(\mathrm{mod}\, m_i\right)$,

$\tilde{\beta}_{i\;j-1}^{\,j}=\left(\tilde{\beta}_{i\;j-1}^{\,j-1}-\tilde{\beta}_{n\;j-1}^{\,j-1}\right)\left(\mathrm{mod}\, m_i\right)$, $i=1,2,\ldots,n-1$. Тогда

$$R=\begin{cases}R_1,\left(\tilde{N}_1^{J}=\tilde{N}_2^{J}\right)\cap\left(\alpha^{j-1}{}_{n\;j-1}>\beta^{j-1}{}_{n\;j-1}\right)\\ R_2,\left(\tilde{N}_1^{J}=\tilde{N}_2^{J}\right)\cap\left(\alpha^{j-1}{}_{n\;j-1}<\beta^{j-1}{}_{n\;j-1}\right)\end{cases}.$$

Если $\tilde{N}_1\neq\tilde{N}_2$, то выполняется $(j+1)$-я итерация.

Если процесс сравнения продолжается до $(n-1)$-й итерации, то

$$R=\begin{cases}R_1,\left(\tilde{N}_1^{\,n-1}\neq\tilde{N}_2^{\,n-1}\right)\cap\left(\alpha_{n\;n-1}^{\;n-1}>\beta_{n\;n-1}^{\;n-1}\right)\cup\\ \quad\cup\left(\left(\tilde{N}_1^{\,n-1}=\tilde{N}_2^{\,n-1}\right)\cap\left(\tilde{\alpha}_{n\;n-2}^{\;n-2}>\tilde{\beta}_{n\;n-2}^{\;n-2}\right)\right)\\ R_2,\left(\left(\tilde{N}_1^{\,n-1}\neq\tilde{N}_2^{\,n-1}\right)\cap\left(\alpha_{n\;n-1}^{\;n-1}<\beta_{n\;n-1}^{\;n-1}\right)\right)\cup\\ \quad\cup\left(\left(\tilde{N}_1^{\,n-1}=\tilde{N}_2^{\,n-1}\right)\cap\left(\tilde{\alpha}_{n\;n-2}^{\;n-2}<\tilde{\beta}_{n\;n-2}^{\;n-2}\right)\right)\end{cases}$$

Пример сравнения пары чисел при данном подходе (базовый алгоритм) представлен в табл.1

Таблица 1. Сравнение по базовому алгоритму

Модули		2	3	5	7	11	13
Числа	**4319**	1	2	4	0	7	3
	9987	1	0	2	5	10	3
Итерация 1	-3	0	2	1	4	4	=
	-3	0	0	4	2	7	=
	:13	0	2	2	3	2	=
	:13	0	0	3	5	9	=
Итерация 2	-2	0	0	0	1	=	=
	-9	1	2	2	4	=	=
	:11	0	0	0	2	=	=
	:11	1	0	4	6	=	=
Итерация 3	-2	0	1	3	=	=	=
	-6	1	0	3	=	=	=
	:7	0	1	4	=	=	=
	:7	1	0	4	=	=	=
Итерация 4	-4	0	0	=	=	=	=
	-4	1	0	=	=	=	=
	:5	0	0	=	=	=	=
	:5	1	0	=	=	=	=
Итерация 5	0	0		=	=	=	=
	0	1		=	=	=	=

Определим временны*е* оценки каждого из подходов.

По первому подходу для определения позиционных характеристик n - разрядных чисел требуется $T_{1,1}=n-1$ итераций, поскольку $\pi_1=\alpha_1$.

Определение количества итераций при сравнении позиционных характеристик n - разрядных чисел выполним с помощью дерева возможных исходов, представленного на *рис.2.*

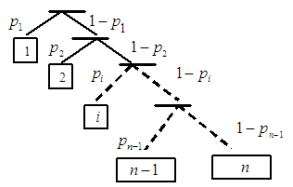

Рис.2. Дерево возможных исходов

p_i - вероятность события, которое заключается в том, что результат сравнения будет получен на i-ой итерации.

При сравнении по первому подходу $p_1 = \frac{1-m_i}{m_i}$. Для системы модулей, например,

$$m_1 = 2, m_2 = 3, m_3 = 5, m_4 = 7, m_5 = 11,$$
$$m_6 = 13, \quad m_7 = 19, m_8 = 29$$

количество итераций $T_{1,2} = 1,085$. Округляем до ближайшего большего, т.е. $T_{1,2} = 2$. Таким образом, при сравнении по первому подходу пары чисел в данной системе модулей быстродействие алгоритма $\Omega_0 = \frac{1}{n+1} = 0,111$.

Определение быстродействия при втором подходе также выполним с помощью дерева возможных исходов, представленного на *рис.2.* При этом $p_i = \frac{3}{M_i} \times \sum_1^i m_j$, $i = 1,2,\ldots,n$, а теоретическое быстродействие $\Omega_{1\grave{O}} = 0,158.$ Моделирование процесса сравнения по данному алгоритму при равновероятном распределении чисел дает $\Omega_{1\acute{Y}} = 0,155$ и $\Delta_1 = \frac{\Omega_{1T} - \Omega_{1\acute{Y}}}{\Omega_{1\grave{O}}} * 100 = 1,9\%$, что свидетельствует о хорошем согласовании с теоретическим результатом. Выигрыш в быстродействии по отношению к традиционному алгоритму $\theta_1 = \frac{\Omega_1}{\Omega_0} = 1,42$.

Эффективная реализация данного подхода зависит от выбора значения $m_i, i = 1,2,\ldots,n-(j-1)$ на каждой j-й итерации, $j = 1,2,\ldots,n-1$ получения приведенных остатков.

Наибольшая вероятность $p_i = \frac{1}{\prod_{t=1}^{i-1} m_i}$ попадания сравниваемых чисел в диапазон длины m_i на j-й итерации достигается при наименьшем значении $\prod_{i=1}^{n} m_i$. Следовательно, на каждой итерации в качестве $m_i, i = 1,2,\ldots,n-(j-1)$ следует принимать наибольший из модулей.

Поскольку данный вывод основан на вероятностной оценке, в реальной практике могут встретиться пары сравниваемых чисел, для которых выбор наибольшего из модулей не дает лучшего результата (*табл.2* и *табл.3*).

Таблица 2

Итерации	Числа	Модули					
		2	**3**	**5**	**7**	**11**	**13**
		Остатки					
	13012	0	1	2	6	10	12
	13013	1	2	3	0	0	0
1		0	1	0	1	9	
		1	2	3	0	0	
	1000	0	1	0	6	10	
	1001	1	2	1	0	0	
…	…	…	…	…	…	…	…
4		0	1				
		0	1				

Таблица 3

Итерации	Числа	Модули					
		13	**3**	**5**	**7**	**11**	**2**
		Остатки					
	13012	12	1	2	6	10	0
	13013	0	2	3	0	0	1
1		0	1	2	6	10	
		0	1	2	6	10	

В связи с этим следующее направление состоит в том, что сравнение осуществляется одновременно по всем n системам модулей $m_1, m_2, \ldots, m_n$, причем для $s = i$ в качестве старшего модуля выбирается m_i. Вероятность получения результата сравнения только на i-ой итерации

$$p_{si} = 1 - \prod_{s=1}^{n} (1 - \frac{3}{M_{si}}), \quad i = n-1, n-2, \ldots, i, \ldots, 2, 1$$
$$M_{si} = \frac{Ms(i+1)}{m_{s(i+1)}}; \quad M_{sn} = m_{s1} m_{s2} \ldots m_{sn-1} m_{sn}$$

Теоретическое быстродействие $\Omega_{2T} = 0,171$. Моделирование дает $\Omega_{2\acute{Y}} = 0,181$, $\Delta_2 = \frac{\Omega_{2\acute{Y}} - \Omega_{2\grave{O}}}{\Omega_{2\grave{O}}} * 100 = 5,4\%$, что согласуется с теоретическим результатом. Выигрыш в быстродействии $\theta_1 = \frac{\Omega_1}{\Omega_0} = 1,543$.

Одним из направлений существенного повышения быстродействия является вычисление приведенных остатков по парам модулей (парное агрегирование), и выбор вместо α_s сочетания

$$\alpha_s \alpha_t, \quad s \neq t, \quad s = 1,2,\ldots,n, \quad t = 1,2,\ldots,n.$$

На каждой итерации по значениям остатков чисел для пары модулей выбираем константы для вычитания из остатков по остальным модулям. Выигрыш при этом в худшем случае, когда сравнение продолжается о

$n-1$ -й итерации, $\theta = \dfrac{n-1}{\dfrac{n}{2}-1}$. Для нашего примера при n =8 θ =2,33.

Как показано выше, на каждой итерации в качестве $m_i, i = 1,2,\ldots,n-(j-1)$ следует принимать наибольший из модулей. Поэтому для уменьшения количества итераций представляется целесообразным выполнять процесс сравнения чисел, записанных в системе основных модулей $m_1, m_2, \ldots, m_i, \ldots, m_t, \ldots, m_n$, на системе модулей $m_1, m_2, \ldots, \tilde{m}_i, \ldots, \tilde{m}_t, \ldots, m_n$, где $\tilde{m}_i$ и $\tilde{m}_t$ - дополнительные большие модули. Выполнение других вычислительных операций осуществляется по основным модулям системы.

Таблице 4 и *таблице 5* иллюстрируют сравнение пары чисел $N_1 = 9876$ и $N_2 = 2123$ по базовому алгоритму и методом введения больших модулей соответственно.

Таблица 4. Сравнение по базовому алгоритму

Модули							
Числа	Итерация	**2**	**3**	**5**	**7**	**11**	**13**
9876		0	0	1	6	9	9
2123		1	2	3	2	0	4
9867		1	0	2	4	0	0
2119	**1**	1	1	4	5	7	0
759		1	0	4	3	0	
163		1	1	3	2	9	
759		1	0	4	3	0	
154	**2**	0	1	4	0	0	
69		1	0	4	6		
14		0	2	4	0		
63		1	0	3	0		
14	**3**	0	2	4	0		
9		1	0	4			
2		0	2	2			
5		1	2	0			
0	**4**	0	0	0			
1		1	1				
0		0	0				

Таблица 5. Сравнение введением больших модулей

Модули									
Числа		**2**	**3**	**5**	**7**	***59***	***61***	**11**	**13**
9876		0	0	1	6	***23***	***55***	9	9
2123		1	2	3	2	**58**	**49**	0	4
9821	**1**	1	2	1	0	**27**	**0**	9	6
2074		0	1	4	2	***9***	**0**	6	7
161		1	2	1	0	43		7	5
34		0	1	4	6	34		1	8
118	**2**	0	1	3	6	8		8	1
0		**0**	**0**	**0**	**0**	**0**		**0**	**0**

При этом

$$p_{si} = \frac{3}{\sum\limits_{s=1}^{n}\left(\left|\frac{M_{si}+1}{m_{si}+1}\right|+1\right)} - \frac{2}{\sum\limits_{s=1}^{n}\left(\left|\frac{M_{si}+1}{m_{si}+1}\right|+1\right)^2},$$

$i = 1,\ldots,n-1,$

$$M_{si} = \frac{Ms(i+1)}{m_{s(i+1)}}; M_{sn} = m_{s1}\ldots m_{sn-1}m_{sn}$$

Быстродействие $\Omega_{1\acute{Y}} = 0,246$, выигрыш в быстродействии $\theta_3 = \dfrac{\Omega_3}{\Omega_0} = 2,21$.

Введение дополнительных модулей позволяет одновременно со сравнением чисел решить и другие немодульные операции, в частности, определение выхода числа за диапазон. В работе [4] показано, что для однозначного определения выхода

$S_{\Pi} = N_1 \times N_2 = T \times \dfrac{M}{2}$ за диапазон M необходимо расширить его до $M \times \dfrac{\sqrt{M}\downarrow}{2}$, где $T = \sqrt{M}\downarrow$ - ближайшее меньшее к $\sqrt{M}$ целое число. Для данной системы основных модулей $\dfrac{\sqrt{M}\downarrow}{2} = 89$, что обеспечивается двумя введенными дополнительными модулями $\tilde{m}_5 = 59$ и $\tilde{m}_6 = 61$.

Выводы

Рассмотрены некоторые алгоритмические пути повышения быстродействия операции сравнения чисел в СОК. Полученные при этом оценки позволяют сделать заключение об эффективности предложенных решений. Представляется целесообразным рассматривать представленные алгоритмические решения в качестве возможных направлений повышения быстродействия операции сравнения чисел в системе остаточных классов.

ЛИТЕРАТУРА

1. Акушский И. Я., Юдицкий Д. И. Машинная арифметика в остаточных классах. — М. : оветское радио, 1968. — 440 с.
2. Червяков Н. И. Методы и принципы построения модулярных нейрокомпьютеров. Труды Юбилейной Международной научно-технической конференции «50 лет модулярной арифметики», Россия, Москва, Зеленоград, 23—25 ноября 2005, издательство МИЭТ, — С. 239—249.
3. Полисский Ю. Д. Формирование позиционных характеристик при табличной реализации алгоритмов системы остаточных классов // Сборник трудов конференции «Моделирование-2008, SIMULATION-2008». — Т. 2. — 14—16 мая 2008. — Киев. — С. 489—95.

4. Полисский Ю. Д. Новые способы выполнения сложных операций в системе остаточных классов // Электронное моделирование. — 2011. — Т. 33. — №5. — С. 73—81.
5. Факторович М. Г., Полисский Ю. Д. Устройство для сравнения чисел, выраженных в системе остаточных классов. Авт. свид. СССР №608155 М.Кл^2 G 06 F 7/04, 1978.
6. Полисский Ю. Д. Некоторые вопросы выполнения сложных операций в системе остаточных классов. // Электронное моделирование. — 2008. — Т. 30. — №2. — С. 115—120.

пост.08.07.13