УДК 681.04

Ю.Д.Полисский

СРАВНЕНИЕ ЧИСЕЛ В СИСТЕМЕ ОСТАТОЧНЫХ КЛАССОВ

(*НИИ автоматизации черной металлургии, г.Днепропетровск*)

В статье выполнен обзор основных работ по сравнению чисел, представленных в системе статочных классов.

In the article are realized the survey of principial works from number comparing represented in residue class system.

В настоящее время в связи с развитием современной техники, информационных и управляющих систем все большее применение находят новые принципы на основе представления данных в системе остаточных классов [1].

Системой остаточных классов (СОК) называется система счисления [2,3], в которой произвольное число A представляется в виде набора наименьших неотрицательных остатков по модулям m_1, m_2, …, m_n, т.е. $A=[A(\mathrm{mod}\ m_1), A(\mathrm{mod}\ m_2), \ldots, A(\mathrm{mod}\ m_n)]$ или

$$A=(\alpha_1,\alpha_2,\ldots,\alpha_n). \qquad (1)$$

Здесь $\alpha_i=A(\mathrm{mod}\ m_i)$. При этом, если числа m_i взаимно простые, то представление числа A в виде (1) является единственным, а объем диапазона [0,M) представимых чисел в этом случае равен

$$M=m_1m_2\ldots m_n. \qquad (2)$$

Для чисел диапазона [0,M), представленных в виде (1), арифметические операции сложения, вычитания и умножения выполняются с остатками α_i независимо друг от друга, причем по простым правилам.

К достоинствам такого представления чисел относится также малоразрядность остатков, высокая точность и надежность, способность системы к самокоррекции. Однако возникают серьезные трудности при реализации так называемых немодульных операций, для выполнения которых необходимо знание цифр операндов по всем разрядам [4]. Одной из таких немодульных операций является операция сравнения чисел.

Решение любой задачи управления всегда включает сравнение в каждый момент времени состояния управляемых объектов с заданным состоянием, соответствующим алгоритму функционирования системы. Это сравнение может быть достаточно простым, но зачастую требует решения весьма сложных многомерных задач. Целью сравнения в наиболее простом случае является обнаружение факта совпадения или несовпадения значений величин. В более сложных случаях операции сравнения включают оценку степени несовпадения состояний сравниваемых объектов, выделение объекта с доминирующим в некотором смысле состоянием и оценку степени доминирования и т.п. В дальнейшем будем применять термин «сравнение чисел», подразумевая под ним целенаправленный процесс обработки информации для получения указанных результатов.

Операция сравнения чисел [5] в одних случаях представляет собой сопоставление значений двух чисел A и B (попарное сравнение) и проверку наличия того или иного признака у результата, в других случаях при сравнении анализируется группа чисел (групповое сравнение), а результат представляет собой наибольшее или наименьшее их значение, признак положения чисел по отношению к граничным значениям либо к некоторому фиксированному значению или длину диапазона значений анализируемых чисел. Конкретное выполнение операции сравнения при решении прикладных задач может осуществляться программно либо специальными устройствами сравнения (или компараторами), реализующими выбранные алгоритмы.

Устройство сравнения - устройство, предназначенное для попарного или группового сравнения данных с выработкой результата сравнения в виде сигналов и чисел, представляющих собой качественные и количественные оценки значений сравниваемых данных.

КЛАССИФИКАЦИЯ СРАВНЕНИЯ ЧИСЕЛ

Решение задач сравнения чисел, в результате которого достигается та или иная цель, осуществляется применением широкого класса различных операций по обработке сравниваемых чисел. При описании процесса сравнения указание цели и применяемых операций дает необходимое и достаточное представление о сравнении, поэтому целесообразно оба эти момента отразить в его определении.

Будем под сравнением чисел понимать совокупность операций над сравниваемыми числами, имеющих целью установить количественные и качественные оценки соотношения сравниваемых чисел. Такими оценками являются признаки равенства или неравенства сравниваемых чисел, значения большего, меньшего из них, наибольшей, наименьшей разности, признак выхода какого-либо из сравниваемых чисел за установленные границы или значение отклонения исследуемых чисел от одного или нескольких фиксированных чисел. Наконец, может предусматриваться совместное получение указанных результатов.

В основу классификации сравнения чисел положены признаки, позволяющие выделить существенные для анализа и синтеза алгоритмов и устройств сравнения характеристики процесса сравнения. Эти признаки включают указание вида сравнения и достигаемой цели, используемых алгоритмов и этапности процесса сравнения.

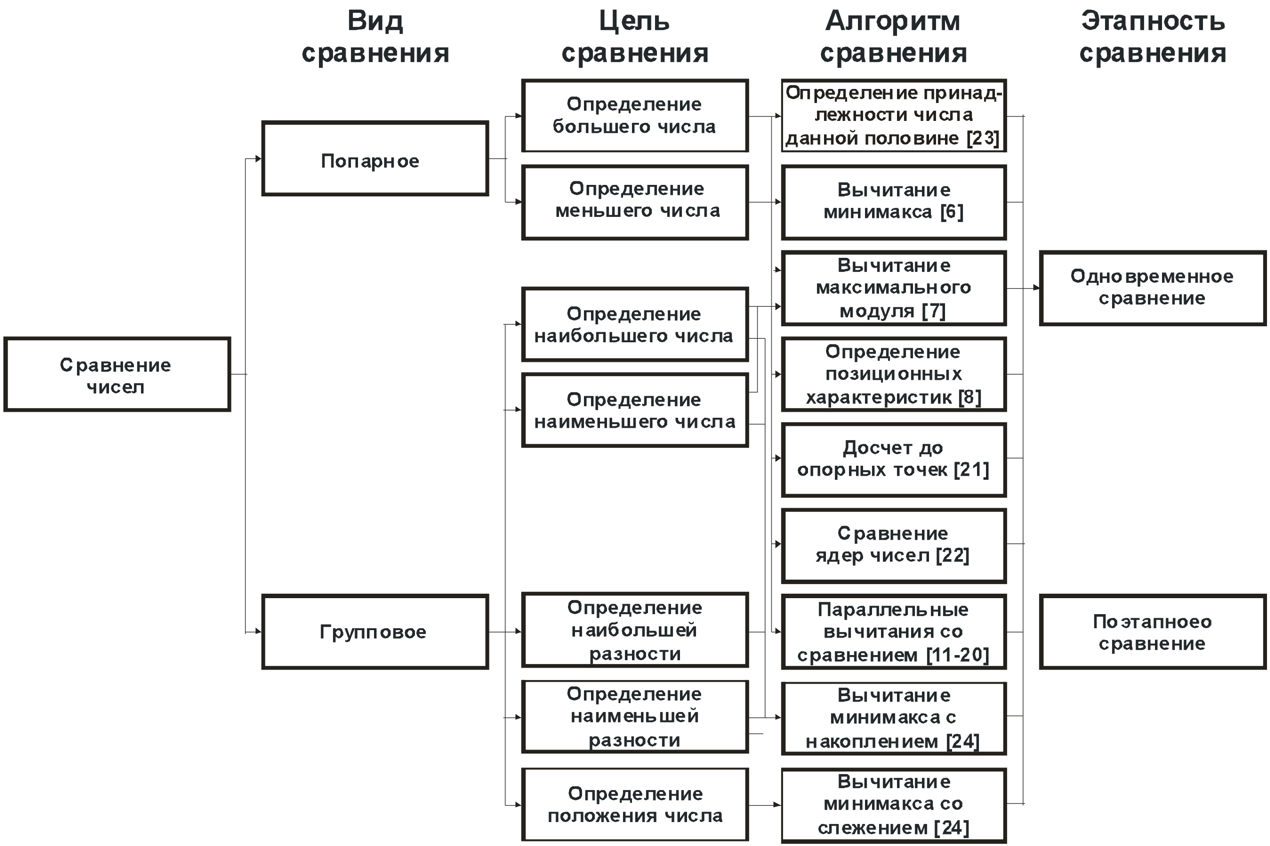
В зависимости от вида сравнения алгоритмы могут быть разделены на две группы: алгоритмы попарного и группового сравнения данных. Это соответствует двум отмеченным ранее типам решаемых задач: сравнения состояния объекта с заданным состоянием и выделения объекта с доминирующим в некотором смысле состоянием. Целесообразность выделения алгоритма попарного сравнения в качестве самостоятельного подтверждается не только количеством работ, посвященных этому виду сравнения, но и тем, что во многих случаях групповое сравнение осуществляется поэтапно с использованием попарного сравнения. Кроме того, для попарного сравнения разработаны специальные методы и схемы, позволяющие осуществлять ускоренное решение задач сравнения. Эта принадлежность к попарному или групповому сравнению данных и выбрана в качестве наиболее общего признака классификации.

Следующий признак классификации включает указание цели, которая должна быть достигнута в результате сравнения. Как отмечалось ранее, при попарном сравнении - это определение равенства или неравенства чисел, большего или меньшего из них и их разности. При групповом сравнении количество решаемых задач увеличивается. Помимо выбора наибольшего или наименьшего из группы чисел дополнительно возникают задачи определения наибольшей и наименьшей разности, а также положения исследуемых чисел по отношению к одному или нескольким фиксированным числам.

Решение каждой из указанных задач может быть осуществлено применением тех или иных алгоритмов. Поэтому указание используемого алгоритма является следующим классификационным признаком сравнения.

Наконец, при групповом сравнении в одних случаях осуществляется одновременное сравнение всех данных, в других случаях - это поэтапный процесс, каждый этап которого основан на попарном сравнении. От этапности процедуры сравнения зависит быстродействие, которое является критерием эффективности алгоритма сравнения. Кроме того, этапность влияет на схемное решение устройства сравнения. Поэтому указание этапности процесса сравнения также принято в качестве классификационного признака.

Ниже приведена иллюстрация классификации сравнения чисел, выполненной в соответствии с описанными признаками, библиография основных работ, в которых приведены теоретические и практические результаты – в списке литературы [1-25].


Вид сравнения
Цель сравнения
Алгоритм сравнения
Этапность сравнения
Сравнение чисел
Попарное
Групповое
Определение большего числа
Определение меньшего числа
Определение наибольшего числа
Определение наименьшего числа
Определение наибольшей разности
Определение наименьшей разности
Определение положения числа
Определение принадлежности числа данной половине [23]
Вычитание минимакса [6]
Вычитание максимального модуля [7]
Определение позиционных характеристик [8]
Досчет до опорных точек [21]
Сравнение ядер чисел [22]
Параллельные вычитания со сравнением [11-20]
Вычитание минимакса с накоплением [24]
Вычитание минимакса со слежением [24]
Одновременное сравнение
Поэтапноео сравнение

ОСНОВНЫЕ РЕШЕНИЯ ПО СРАВНЕНИЮ ЧИСЕЛ

Рассмотрение будем производить согласно принятой классификации, при этом вначале описывается попарное, затем групповое сравнение чисел.

Попарное сравнение чисел

Пусть A и B – сравниваемые числа:

$$A=(\alpha_1,\alpha_2,\ldots,\alpha_n),$$
$$B=(\beta_1, \beta_2,\ldots,\beta_n). \qquad (3)$$

Необходимо определить результат A > B, A < B или A=B. Сравнение будем проводить для ненулевых чисел, в противном случае результат очевиден.

Если системой управления не предъявляются высокие требования к быстродействию, в основу сравнения может быть положен следующий алгоритм [6]. На каждом такте сравнения для первого и второго из сравниваемых чисел одновременно определяется максимальный остаток по всем модулям

$$r^1_{\mathbf{x}}= \max\{\alpha_1, \alpha_2, \ldots, \alpha_n\},$$
$$r^2_{\mathbf{x}}=\max\{\boldsymbol{\beta}_1,\boldsymbol{\beta}_2,\ldots,\boldsymbol{\beta}_n\}, \qquad (4)$$

выбирается

$$r^{к1}_{\mathbf{m}}=\min\{r^1_x,r^2_x,\ldots,r^к_x\}=(r^{к1}_{\mathbf{1}}, r^{к1}_{\mathbf{2}},\ldots, r^{к1}_{\mathbf{i}},\ldots, r^{к1}_{\mathbf{n}})$$

и одновременно для обоих чисел находятся

$$A^1=A - r^{к1}_{\mathbf{m}} =(\alpha^1_1,\alpha^1_2,\ldots, \alpha^1_n)$$

и (6)

$$B^1=B - r^{к1}_{\mathbf{m}} =(\beta^1_1, \beta^1_2,\ldots,\beta^1_n),$$

Где: $\alpha^1_1= (\alpha_1- r^{к1}_{\mathbf{1}}) \pmod{m_1}$,

$\alpha^1_2= (\alpha_2- r^{к1}_{\mathbf{2}}) \pmod{m_2}$,

……………………….,

……………………….,

$\alpha^1_i= (\alpha_i- r^{к1}_{\mathbf{i}}) \pmod{m_i}$,

……………………….,

……………………….,

$\alpha^1_n= (\alpha_n- r^{к1}_{\mathbf{n}}) \pmod{m_n}$

и

$$\beta^1{}_1 = (\beta_1 - r^{к1}{}_{\mathbf{1}}) \pmod{m_1},$$
$$\beta^1{}_2 = (\beta_2 - r^{к1}{}_{\mathbf{2}}) \pmod{m_2},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\beta^1{}_i = (\beta_i - r^{к1}{}_{\mathbf{i}}) \pmod{m_i},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\beta^1{}_n = (\beta_n - r^{к1}{}_{\mathbf{n}}) \pmod{m_n}.$$

Если одно из чисел (6) обратилось в нуль, процесс сравнения завершается. В противном случае процесс определения (4), (5), (6) продолжается для чисел A^1 и B^1, A^2 и B^2, …, A^s и B^s до тех пор, пока одно из чисел не обратится в нуль. Если $A^s = A^{s-1} - \mathbf{r}^{кs}{}_{\mathbf{m}} = 0$, а $B^s = B^{s-1} - \mathbf{r}^{кs}{}_{\mathbf{m}} \neq 0$, то $A<B$.

Если распределение чисел равновероятно, то среднее значение числа из диапазона (2) $A_{ср} = M/2$, и количество К тактов сравнения можно приближенно оценить следующим образом. Пусть $m_1<m_2<\ldots<m_n$. Тогда $\alpha_{max} = \beta_{max} = (m_n/2)$ и $r = \min\{\alpha_{max}, \beta_{max}\} = (m_n/2)$. Тогда $К = (A_{ср}/r) = = m_1 m_2 \ldots m_{n-1}$.

Почти такое же быстродействие дает сравнение по алгоритму [7]. Пусть $m_1<m_2<\ldots<m_n$. На первом такте одновременно выполняются операции

$$A_1 = A - \alpha_n = (\alpha^1{}_1, \alpha^1{}_2, \ldots, \alpha^1{}_i, \ldots, \alpha^1{}_{n-1}, 0)$$

и

$$B_1 = B - \beta_n = (\boldsymbol{\beta}^{\mathbf{1}}{}_{\mathbf{1}}, \boldsymbol{\beta}^{\mathbf{1}}{}_{\mathbf{2}}, \ldots, \boldsymbol{\beta}^{\mathbf{1}}{}_{\mathbf{i}}, \ldots, \boldsymbol{\beta}^{\mathbf{1}}{}_{\mathbf{n-1}}, 0), \qquad (7)$$

где

$$\alpha^1{}_1 = (\alpha_1 - \alpha_n) \pmod{m_1},$$
$$\alpha^1{}_2 = (\alpha_2 - \alpha_n) \pmod{m_2},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\alpha^1{}_i = (\alpha_i - \alpha_n) \pmod{m_i},$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,$$
$$\alpha^1{}_n = 0 \pmod{m_n}.$$

и

$\beta^1{}_1 = (\beta_1 - \beta_n) \pmod{m_1}$,
$\beta^1{}_2 = (\beta_2 - \beta_n) \pmod{m_2}$,
………………………,
………………………,
$\beta^1{}_i = (\beta_i - \beta_n) \pmod{m_i}$,
………………………,
………………………,
$\beta^1{}_n = 0 \pmod{m_n}$.

Если $A_1 = B_1$, то $A>B$ при $\alpha_n > \beta_n$, и процесс сравнения на этом завершается. Если же $A_1 \neq B_1$, то A_1 и B_1 кратны m_n и могут оказаться кратными одному или нескольким из остальных модулей системы. Поэтому на последующих тактах из обоих чисел A_1 и B_1, A_2 и B_2, …,A_s и B_s одновременно вычитается некоторое число $r=(m_n+\pi)$ до тех пор, пока одно из чисел не обратится в нуль. Если
$A_s = A_{s-1} - r = 0$, а $B_s=B_{s-1}-r\neq 0$, то $A<B$. При этом для равновероятного распределения чисел $\pi \approx (1/ m_1) +$ $+ (1/ m_2) + \ldots + (1/ m_{n-1})$.

Гораздо большее быстродействие достигается при сравнении по алгоритму [8]. Пусть $m_1<m_2<\ldots<m_n$. На первом такте одновременно выполняются операции $A_1=A- \alpha_1$ и $B_1=B- \beta_1$. Если $A_1= B_1$, то $A>B$ при $\alpha_1 > \beta_1$, и процесс сравнения на этом завершается. Если же $A_1\neq B_1$, то соотношение, например, $\alpha_1 > \beta_1$, запоминается. Числа A_1 и B_1 кратны m_1. На втором такте одновременно выполняются операции $A_2=(A_1/m_1)$, где $A_2=(\alpha^1{}_2, \ldots,\alpha^1{}_n)$, и $B_2 = (B_1/m_1)$, где
$B_2=(\beta^1{}_2, \ldots,\beta^1{}_n)$, с сокращением объема диапазона $(0, M_1]$ чисел до $M_1= m_2\ldots m_n$. На следующем такте одновременно выполняются операции $A_3=A_2- \alpha^1{}_2$ и $B_3=B_2- \beta^1{}_2$. Если при этом $A_3=B_3$ и $\alpha^1{}_2<\beta^1{}_2$, то предпочтение отдается последнему соотношению, и $A<B$. Если же $A_3=B_3$ и $\alpha^1{}_2=\beta^1{}_2$, то о результате сравнения судят по запомненному соотношению $\alpha_1 > \beta_1$, и процесс сравнения на этом завершается. Если же $A_3\neq B_3$, то вместо $\alpha_1>\beta_1$ запоминается соотношение $\alpha^1{}_2<\beta^1{}_2$. Числа A_3 и B_3 кратны m_2. В том случае, если ни на одном из тактов не обнаруживается $A_i=B_i$, описанные действия повторяются до получения $A_{2(n-1)}=(\alpha^{n-1}{}_n)$ и $B_{2(n-1)}=(\beta^{n-1}{}_n)$, а о результате сравнения судят по соотношению между $\alpha^{n-1}{}_n$ и $\beta^{n-1}{}_n$. По существу здесь выполняется потактовый переход к полиадическому коду. В [9] показано, что не существует наилучшего метода определения значений позиционных характеристик, при котором бы не нарушалась их однозначность, чем переход числа от системы остаточных классов к полиадическому коду, поскольку величина числа в СОК

зависит от всех остатков числа. В [10] доказана невозможность сравнения чисел на основе анализа лишь соответствующих компонент этих чисел ни, пользуясь терминологией авторов, в логической, ни в алгебраической, ни в смешанной постановке задачи.

Верхняя оценка количества тактов сравнения по данному алгоритму $K=2(n-1)$.

Алгоритм, описанный в работах [11-20], основан на идее [7]. Здесь также на первом такте одновременно выполняются операции

$$A_1=A-\alpha_n=(\alpha^1{}_1,\alpha^1{}_2,\ldots,\alpha^1{}_i,\ldots,\alpha^1{}_{n-1},0)$$

и

$$B_1=B-\beta_n=(\boldsymbol{\beta}^1{}_1,\ \boldsymbol{\beta}^1{}_2,\ldots,\ \boldsymbol{\beta}^1{}_i,\ldots,\boldsymbol{\beta}^1{}_{n-1},0).$$

Если $A_1= B_1$, то $A>B$ при $\alpha_n > \beta_n$, и процесс сравнения на этом завершается. Если же $A_1\neq B_1$, то A_1 и B_1 кратны m_n. Однако здесь с целью повышения быстродействия потактное вычитание m_n производится не в процессе сравнения, а выполнено предварительно с сохранением в виде констант полученных значений. Количество таких констант $K= m_1m_2\ldots m_{n-1}$. Например, для системы модулей $m_1 =2$, $m_2 =3$, $m_3 =5$ таблица констант имеет вид:

$(0 \div (N-1))^* m_n$ $(\Delta ^*5)$	$m_1 =2$	$m_2 =3$	Позиция нуля
	$\alpha^1{}_1$	$\boldsymbol{\beta}^1{}_1$	
$0^* m_3 = 0$	0	00	1
$1^* m_3 = 5$	1	10	2
$2^* m_3 = 10$	0	01	3
$3^* m_3 = 15$	1	00	4
$4^* m_3 = 20$	0	10	5
$5^* m_3 = 25$	1	01	6

Далее из этой таблицы по значениям $\alpha^1{}_1$ и $\boldsymbol{\beta}^1{}_1$ выбирается соответствующая запись и для каждого числа строится однорядный унитарный код длины $m_1m_2\ldots m_{n-1}$ двоичных разрядов, в каждом из которых только на r_1 – ом и r_2 – ом ($A_1- r_1{}^* m_n =0$ и $B_1- r_2{}^* m_n =0$) местах будут нули, а на остальных единицы. При этом номер позиции нуля и характеризует величину числа. Затем выполняется сравнение номеров позиций нуля. Если $r_1< r_2$, то $A<B$, если же $r_1> r_2$, то $A>B$.

Приведенный алгоритм повышает быстродействие, однако требует существенных аппаратурных затрат для хранения количества констант, кратного $m_1m_2 \ldots m_{n-1}$, и хранения унитарных кодов длины $m_1m_2 \ldots m_{n-1}$.

Алгоритм [21] также основан на идее [7]. Среди чисел диапазона (0,M] есть числа $A_t=(\alpha^t_1,\alpha^t_2,\ldots,\alpha^t_i,\ldots,\alpha^t_j,..,\alpha^t_n)$, кратные одновременно нескольким модулям, например, m_i и m_j. Тогда $\alpha^t_i=0$, $\alpha^t_j=0$. Исходя из этого, в [21] в зависимости от требований к быстродействию системы и аппаратурным затратам выбираются определенные модули и составляется таблица констант – опорных точек. Для системы модулей $m_1=2$, $m_2=3$, $m_3=5$, $m_4=7$ ($M=m_1*m_2*m_3*m_4=210$) таблица опорных чисел, кратных, например, $m_2*m_4=21$ состоит из $(M/m_2*m_4)=10$ строк и имеет вид:

Опорная точка А	$(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$
0	(0, 0, 0, 0)
21	(1, 0, 1, 0)
42	(0, 0, 2, 0)
63	(1, 0, 3, 0)
84	(0, 0, 4, 0)
105	(1, 0, 0, 0)
126	0, 0, 1, 0)
147	(1, 0, 2, 0)
158	(0, 0, 3, 0)

Далее к обоим числам потактно добавляются единицы с одновременным сравнением образующихся значений $A_\tau=(\alpha^\tau_1, \alpha^\tau_2,\ldots, \alpha^\tau_i,.., \alpha^\tau_n)$ и $B_\tau=(\boldsymbol{\beta}^\tau_1, \boldsymbol{\beta}^\tau_2,\ldots, \boldsymbol{\beta}^\tau_i,.., \boldsymbol{\beta}^\tau_n)$ с константами таблицы. По достижении опорных точек их номера N_A и N_B запоминаются и затем сравниваются. Если $N_A > N_B$, то A>B, если же $N_A<N_B$, то A<B. При $N_A=N_B$ сравниваются количества подсуммированных единиц, которое не превышает в данном случае $K=m_2*m_4=21$. Количество тактов, необходимое для достижения опорных точек, и определяет быстродействие алгоритма.

Если нет ограничений на аппаратурные затраты, то для достижения максимального быстродействия целесообразно, по данному алгоритму, выбрать в качестве опорных точек числа диапазона, кратные наименьшему модулю. В этом случае приходим к таблице констант предыдущего алгоритма, а количество тактов, необходимое для достижения опорных точек, не превышает величины наименьшего модуля.

Представим

$A=R_{Ai}*m_i + \alpha_i$,

$B=R_{Bi}*m_i + \boldsymbol{\beta}_i$,

где $R_{Ai}=[A/m_i]$, $R_{Bi}=[B/m_i]$. Пусть $m_1<m_2<\ldots<m_n$. Работа алгоритма [22] основана на вычислении R_{Ai} и R_{Bi} и их последующем сравнении. Если $R_{Ai} > R_{Bi}$ $A>B$. Если $A=B$, то сравниваются α_i и $\boldsymbol{\beta}_i$. При $\alpha_i > \boldsymbol{\beta}_i$ $A>B$. Для принятой системы модулей по ортогональным базисам $H_1, H_2, \ldots, H_i\ldots, H_n$ находятся коэффициенты $r_1=(H_1/m_n)$, $r_2=(H_2/m_n)$, …, $r_i=(H_i/m_n)$, …, $r_n=((H_n-1)/m_n)$ и $r_0=(M/m_n)$, а R_A и R_B находятся, как

$R_A=((r_{1*}\alpha_1)(\mathrm{mod}\ r_0)+ (r_{2*}\ \alpha_2)\ (\mathrm{mod}\ r_0) +\ldots+ +(r_{i*}\ \alpha_i)\ (\mathrm{mod}\ r_0)+\ldots+(r_{n*}\ \alpha_n)\ (\mathrm{mod}\ r_0))\ (\mathrm{mod}\ r_0)$,

$R_B=((r_{1*}\boldsymbol{\beta}_1)(\mathrm{mod}\ r_0)+ (r_{2*}\ \boldsymbol{\beta}_2)\ (\mathrm{mod}\ r_0) +\ldots+ +(r_{i*}\ \boldsymbol{\beta}_i)\ (\mathrm{mod}\ r_0)+\ldots+(r_{n*}\ \boldsymbol{\beta}_n)\ (\mathrm{mod}\ r_0))\ (\mathrm{mod}\ r_0)$.

Оценка количества тактов сравнения по данному алгоритму $K=n$.

Будем отличать числа первой и второй половины диапазона. Если $2=m_1<m_2<\ldots<m_n$, то при $0\le A<(M/2)$ A- число первой половины, а при $(M/2)\le A<M$ A - число второй половины. Если же все модули нечетные, то при $0\le A\le((M-1)/2)$ A - число первой половины, а при $((M-1)/2)<A<M$ A - число второй половины.

Алгоритм сравнения, приведенный в работе [23], которая является первой из известных работ по сравнению чисел в системе остаточных классов, основан на определении принадлежности чисел и их разности данной половине.

В соответствии с [23] модули системы упорядочиваются, как $2= m_1<m_2<\ldots<m_n$.

Испытуемое число $A=(\alpha_1, \alpha_2, \ldots, \alpha_n)$.

Тогда число, обладающее остатками $\alpha_2, \alpha_3, \ldots, \alpha_n$, является числом первой половины. Метод [23] основан на определении четности числа первой половины с остатками $\alpha_2, \alpha_3, \ldots, \alpha_n$ и сравнении ее с четностью испытуемого числа, которая определяется остатком α_1. В случае совпадения четностей данное число относится к первой половине, в противном случае – ко второй. При этом четность числа с остатками $\alpha_2, \alpha_3, \ldots, \alpha_n$ находится итеративным путем с переходом к полиадическому коду, выполняемым аналогично [8], и фиксацией смены четностей на каждом шаге итерации.

Если сравниваемые числа разных половин, например, A – число первой половины, а B - число второй половины, то результат $A<B$ очевиден. Если же A и B числа одной половины, то составляется разность $\Delta=A-B$ и определяется принадлежность Δ данной половине. В том случае, если Δ принадлежит первой половине, $A\geq B$. Верхняя оценка количества тактов сравнения, приведенная в [23], $K=1+6(n-2)$.

Групповое сравнение чисел

Пусть

$$\begin{aligned} &A_1=(\alpha^1{}_1, \alpha^1{}_2, \ldots, \alpha^1{}_n), \\ &A_2=(\alpha^2{}_1, \alpha^2{}_2, \ldots, \alpha^2{}_n), \\ &\ldots\ldots\ldots\ldots\ldots\ldots, \\ &\ldots\ldots\ldots\ldots\ldots\ldots, \\ &A_к=(\alpha^к{}_1, \alpha^к{}_2, \ldots, \alpha^к{}_n). \end{aligned} \tag{8}$$

сравниваемые числа.

Определение максимального и минимального из группы чисел при невысоких требованиях к быстродействию достигается на основе алгоритма, приведенного в работе [7], которая является, по-видимому, первой из известных работ по групповому сравнению чисел в системе остаточных классов. Пусть $m_1<m_2<\ldots<m_n$. Сравниваемые числа (8) представим в виде

$$\begin{aligned} &A_1=\theta_1*m_n + \alpha^1{}_n, \\ &A_2=\theta_2*m_n + \alpha^2{}_n, \\ &\ldots\ldots\ldots\ldots\ldots, \\ &\ldots\ldots\ldots\ldots\ldots, \\ &A_k=\theta_k*m_n + \alpha^k{}_n \end{aligned} \tag{9}$$

На первом такте одновременно для всех чисел (5) выполняется операция

$$\begin{aligned} &A^1{}_1=A_1- \alpha^1{}_n =\theta_1*m_n, \\ &A^1{}_2=A_2- \alpha^2{}_n =\theta_2*m_n, \\ &\ldots\ldots\ldots\ldots\ldots\ldots, \\ &\ldots\ldots\ldots\ldots\ldots\ldots, \\ &A^1{}_к=A_к- \alpha^к{}_n =\theta_к*m_n. \end{aligned} \tag{10}$$

На втором и последующих тактах сравнения одновременно из всех чисел (9) вычитается m_n. При этом в нуль раньше других обратится число с $\theta_{min}= \min\{\theta_1,\theta_2,\ldots, \theta_k\}$, которое сохраняется как минимальное. Соотношение величин $\alpha^1{}_n,\alpha^2{}_n,\ldots, \alpha^к{}_n$ в случае неодновременного обращения всех чисел в нуль не имеет значения, так как $\alpha^i{}_n < m_n$. При одновременном обращении в нуль нескольких чисел в качестве минимального сохраняется число из этой совокупности с $\alpha_{min}=\min\{\alpha^1{}_n,\alpha^2{}_n,\ldots, \alpha^к{}_n\}$. Процесс одновременного вычитания из всех чисел (9) величины m_n продолжается до обращения в нуль всех чисел. Последнее обратившееся в нуль число сохраняется как максимальное. При одновременном обращении в нуль нескольких чисел в качестве максимального сохраняется число из этой совокупности с наибольшим $\alpha^i{}_n$. Количество К тактов сравнения $К= 1+m_1m_2\ldots m_{n-1}$.

Расширение совокупности результатов – получение в результате сравнения наибольшего, наименьшего чисел, их максимальной и минимальной разности достигается с помощью алгоритма [24], использующего идею [6]. Здесь также на каждом такте сравнения одновременно для всех сравниваемых чисел определяются

$$\begin{aligned}
&r^1{}_{\mathbf{x}}= \max\{\alpha^1{}_1, \alpha^1{}_2, \ldots, \alpha^1{}_n\},\\
&r^2{}_{\mathbf{x}}= \max\{\alpha^2{}_1, \alpha^2{}_2, \ldots, \alpha^2{}_n\},\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&r^к{}_{\mathbf{x}}= \max\{\alpha^к{}_1, \alpha^к{}_2, \ldots, \alpha^к{}_n\},
\end{aligned}\qquad (11)$$

и выбирается

$$r^к{}_{\mathbf{m}}=\min\{r^1{}_x,r^2{}_x,\ldots,r^к{}_x\}=(r^к{}_{\mathbf{1}},r^к{}_{\mathbf{2}},\ldots,r^к{}_{\mathbf{i}},\ldots,r^к{}_{\mathbf{n}}), \qquad (12)$$

и одновременно для всех чисел находятся

$$\begin{aligned}
&A^1{}_{\mathbf{1}}=A_{\mathbf{1}} - r^к{}_{\mathbf{m}} =(\alpha^1{}_{11},\alpha^1{}_{21},\ldots, \alpha^1{}_{n1}),\\
&A^1{}_{\mathbf{2}}=A_{\mathbf{2}} - r^к{}_{\mathbf{m}} =(\alpha^2{}_{12},\alpha^2{}_{22},\ldots, \alpha^2{}_{n2}),\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&A^1{}_{\mathbf{j}}=A_{\mathbf{i}} - r^к{}_{\mathbf{m}} =(\alpha^j{}_{1j},\alpha^j{}_{2j},\ldots, \alpha^j{}_{nj}),\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&A^1{}_{\mathbf{k}}=A_{\mathbf{k}} - r^к{}_{\mathbf{m}} =(\alpha^k{}_{1k},\alpha^k{}_{2k},\ldots, \alpha^k{}_{nk}),
\end{aligned}\qquad (13)$$

где

$$\alpha^j{}_{ij}= (\alpha^j{}_i- r^k{}_{\mathbf{i}}) \pmod{m_i},$$

и одновременно значения $r^к_{\mathbf{m}}$ накапливаются на регистрах наибольшего и наименьшего чисел. Процесс продолжается до тех пор, пока одно из чисел (13) не обратится в нуль. Это число и является наименьшим, оно записано на своем регистре.

Далее процесс определения (11), (12), (13) продолжается для всех ненулевых чисел с накоплением дополнительно и на регистре максимальной разности до обращения в нуль последнего из чисел, которое является наибольшим. Это число записано на своем регистре. При каждом очередном обращении в нуль на регистрах оставшихся ненулевыми чисел записаны значения разности между исходным и очередным обратившимся в нуль числом, из которых в процессе вычитания выбирается минимальное значение. При равновероятном распределении чисел приближенная временная оценка такая же, как и при попарном сравнении, использующем идею [6].

Определение ближайшего к заданному числу большего и меньшего из группы чисел с помощью алгоритма [25] также использует идею алгоритма [6]. Здесь также на каждом такте сравнения одновременно для всех сравниваемых чисел определяются

$$\begin{aligned}
&r^1_{\mathbf{x}}= \max\{\alpha^1_1,\ \alpha^1_2,\ \ldots,\ \alpha^1_n\},\\
&r^2_{\mathbf{x}}= \max\{\alpha^2_1,\ \alpha^2_2,\ \ldots,\ \alpha^2_n\},\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&r^к_{\mathbf{x}}= \max\{\alpha^к_1,\ \alpha^к_2,\ \ldots,\ \alpha^к_n\},
\end{aligned} \qquad (14)$$

и выбирается

$$r^к_{\mathbf{m}}=\min\{r^1_x,r^2_x,\ldots,r^к_x\}=(r^к_{\mathbf{1}},r^к_{\mathbf{2}},\ldots,r^к_{\mathbf{i}},\ldots,r^к_{\mathbf{n}}), \qquad (15)$$

и одновременно для всех чисел находятся

$$\begin{aligned}
&A^1_{\mathbf{1}}=A_{\mathbf{1}} -\ r^к_{\mathbf{m}} =(\alpha^1_{11},\alpha^1_{21},\ldots,\ \alpha^1_{n1}),\\
&A^1_{\mathbf{2}}=A_{\mathbf{2}} -\ r^к_{\mathbf{m}} =(\alpha^2_{12},\alpha^2_{22},\ldots,\ \alpha^2_{n2}),\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&A^1_{\mathbf{j}}=A_{\mathbf{i}} -\ r^к_{\mathbf{m}} =(\alpha^j_{1j},\alpha^j_{2j},\ldots,\ \alpha^j_{nj}),\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots,\\
&A^1_{\mathbf{k}}=A_{\mathbf{k}} -\ r^к_{\mathbf{m}} =(\alpha^k_{1k},\alpha^k_{2k},\ldots,\ \alpha^k_{nk}),
\end{aligned} \qquad (16)$$

где

$$\alpha^j_{ij} = (\alpha^j_i - r^k_{\mathbf{i}}) \pmod{m_i}.$$

Одновременно значения $r^к_{\mathbf{m}}$ накапливаются на регистрах. Процесс продолжается до тех пор, пока одно из чисел (16) не обратится в нуль.

Далее процесс определения (14), (15), (16) продолжается для всех ненулевых чисел до обращения в нуль последнего из чисел с накоплением на регистрах очередного и предыдущего из обращенных в нуль чисел. Такое определение основано на фиксировании последовательных моментов обращения в нуль заданного и последующего числа из группы чисел. При равновероятном распределении чисел приближенная временная оценка такая же, как и для предыдущего алгоритма.

ЗАКЛЮЧЕНИЕ

Проведенный обзор основных решений по сравнению чисел в системе остаточных классов позволяет высказать следующие соображения.

При всем разнообразии алгоритмов полученных решений все они основаны на двух принципах. Первый принцип заключается в предварительном преобразовании непозиционного представления в некоторое позиционное и использовании полученного позиционного представления либо непосредственно для решения основной задачи, либо для решения подзадач основной задачи. Второй принцип заключается в вычитании некоторых констант из сравниваемых чисел, выполняемом либо итерационно в процессе сравнения, либо выполненным предварительно. При этом быстродействие алгоритмов, использующих первый принцип, значительно выше, чем алгоритмов, использующих второй принцип на итерационной основе. Однако достигается это за счет увеличения аппаратурных затрат. Использование алгоритмов, использующих второй принцип с предварительно выполненным вычитанием, дает некоторое увеличение быстродействия по сравнению с алгоритмами, использующими первый принцип, но требует существенных аппаратурных затрат.

На основе выполненного обзора можно сделать вывод об отсутствии к настоящему времени эффективных решений по сравнению чисел в системе остаточных классов. Поскольку различными алгоритмами решение достигается с разными техническими и экономическими показателями, дальнейшие работы по сравнению чисел представляется целесообразным проводить в направлении достижения оптимального соотношения указанных показателей для заданных условий применения.

Литература

1.Н.И.Червяков. Методы и принципы построения модулярных нейрокомпьютеров. Сайт http://www.computer-museum.ru/, 2005

2.И.Я.Акушский, Д.И.Юдицкий. Машинная арифметика в остаточных классах. М.: Сов. Радио, 1968. 440 с.

3.А.Ахо, Дж.Хопкрофт, Дж.Ульман. Построение и анализ вычислительных алгоритмов. М.: Мир, 1979.

4.Л.Б.Копыткова, Н.И.Червяков. Реализация деления чисел в системе остаточных классов на модули системы. Вестник Ставропольского государственного университета, 34/2003, с.7-11

5.Ю.Д.Полисский. Цифровое сравнение данных а АСУ ТП и схемах автоматики. М.: Энергия, 1979. 136 с.

6.Ю.Д.Полисский, М.Г.Факторович. Устройство для сравнения чисел в системе остаточных классов. Авт. свид. СССР №618739 М. Кл^2 G 06 F 7/04, 1978.

7.М.Г.Факторович, Ю.Д.Полисский. Устройство для определения максимального и минимального из “n” чисел, представленных в системе остаточных классов. Авт. свид. СССР №603987 М. Кл^2 G 06 F 7/04, 1978.

8.М.Г.Факторович, Ю.Д.Полисский. Устройство для сравнения чисел, выраженных в системе остаточных классов. Авт. свид. СССР №608155 М. Кл^2 G 06 F 7/04, 1978.

9.Сабо Н. Определение знака в неизбыточных системах счисления остаточных классов. Кибернетический сборник, №8. М., «Мир», 1964.

10.Амербаев В.М., Касимов Ю.Ф. О сравнении чисел в непозиционных системах счисления. // Теория кодирования и оптимизации сложных систем. Алма-Ата: Наука, 1977. С. 47 — 54.

11.В.А.Краснобаев, Е.И.Бороденко, А.И.Бецков и др. Устройство для сравнения чисел в системе остаточных классов. Авт. свид. СССР №1037244 G 06 F 7/04, 1983.

12.В.А.Краснобаев, Л.Г.Трусей. Устройство для сравнения чисел в системе остаточных классов. Авт. свид. СССР №1121670 G 06 F 7/04, 1984.

13. В.А.Краснобаев. Устройство для сравнения чисел в системе остаточных классов. Авт. свид. СССР №1145338 G 06 F 7/04, 1985.
14. В.А.Краснобаев. Устройство для сравнения чисел в системе остаточных классов. Авт. свид. СССР №1160394 G 06 F 7/04, 1985.
15. В.И.Долгов, В.А.Краснобаев, А.В.Брезгунов. Устройство для сравнения чисел в системе остаточных классов. Авт. свид. СССР №1224803 G 06 F 7/04, 1986.
16. В.А.Краснобаев, И.Д.Горбенко, М.А.Гальцев и др. Устройство для сравнения чисел в системе остаточных классов. Авт. свид. СССР №1427358 G 06 F 7/04, 1987.
17. В.А.Краснобаев, О.Н.Фоменко, В.П.Ирхин и др. Устройство для сравнения в системе остаточных классов. Авт. свид. СССР №1552171 G 06 F 7/04, 1990.
18. В.А.Краснобаев. Методы арифметического сравнения чисел, представленных кодом системы остаточных классов // АСУ и приборы автоматики. 1987. Вып. 84. С. 74 -76.
19. В.А.Краснобаев. Методы сравнения чисел, представленных кодом системы остаточных классов // Электрон.моделирование. 1988. № 2. С. 84 – 87.
20. В.А.Краснобаев. Методы сравнения операндов в системе остаточных классов // Радиотехника. 2003. Вып. 134. С. 223 – 228 .
21. О.В.Ревинский. Устройство для сравнения чисел. Авт. свид. СССР №1439574 G O6 F 7/04, 1988
22. Хлевной С.Н., Сагдеев К.М. Устройство для сравнения чисел в модулярном коде. Авт. свид. СССР, G06 F 7/04, 1986
23. В.Н.Тейтельбаум. Сравнение чисел в чешской системе счисления. ДАН СССР, 1958, т.121, №5.
24. Ю.Д.Полисский. Устройство для сравнения чисел в системе остаточных классов. Патент Украины, 42649 A, G 06 F 7/04, 2001.
25. Ю.Д.Полисский. Устройство для определения в системе остаточных классов числа, ближайшего к заданному. Патент Украины, 47630 A, G 06 F 7/04,2002.