ИНФОРМАТИКА

2009 **октябрь-декабрь** **№ 4**

ПАРАЛЛЕЛЬНЫЕ ВЫЧИСЛЕНИЯ

УДК 681.3.06(082)

А.А. Коляда, Н.А. Коляда, В.В. Ревинский, А.Ф. Чернявский, Е.В. Шабинская

УМНОЖЕНИЕ ПО БОЛЬШОМУ МОДУЛЮ В МИНИМАЛЬНО ИЗБЫТОЧНОЙ МОДУЛЯРНОЙ СИСТЕМЕ СЧИСЛЕНИЯ С ПРИМЕНЕНИЕМ ОПЕРАЦИЙ МАСШТАБИРОВАНИЯ

Предлагается новый метод умножения по большим простым модулям в минимально избыточной модулярной системе счисления (МИМСС). Его основу составляют быстросходящаяся рекурсивная схема приведения к остатку (схема спуска Ферма) и высокоскоростной алгоритм масштабирования табличного типа. Исследуются проблемы корректности метода и даются оценки его эффективности. Синтезируется мультипликативный алгоритм, который в сравнении с аналогами позволяет уменьшить количество таблиц для формирования базовых интегральных характеристик на 35–40 % и сократить временные затраты как минимум в 1,6 раза.

Введение

Главное отличительное свойство модулярных систем счисления (МСС), постоянно привлекающее к себе внимание специалистов по высокопроизводительным параллельным вычислениям, состоит в отсутствии межразрядных связей при выполнении модульных операций. Особое значение данное свойство модулярной арифметики (МА) имеет для приложений модулярной вычислительной технологии (МВТ) на диапазонах больших чисел и прежде всего для приложений в криптографии [1–12]. В свете сказанного естественной стратегией при разработке методологического и алгоритмического обеспечения МВТ для криптосистем является стремление реализовать отмеченное фундаментальное преимущество МА над позиционной арифметикой в максимальной мере.

Несмотря на сравнительную сложность выполнения в МСС операций масштабирования и расширения модулярного кода (МК), используемых в мультипликативных процедурах по большим простым модулям, известные лучшие разработки на базе МА для систем криптографической защиты информации (СКЗИ) по ряду важных показателей превосходят позиционные аналоги. К главным достоинствам криптосистем с применением МА относятся более высокая скорость шифрования данных, идеальная приспособленность модулярных вычислительных структур (МВС) к табличным реализациям, способность к гибкой реконфигурации и др.

В начале текущего десятилетия рядом исследователей высказывалось мнение, что все основные усовершенствования МВТ для криптосистем уже достигнуты. При этом считалось, что наиболее успешные разработки в этой области предложили К. Рош, Р. Рош, П. Корнерап и др. [1, 9, 10]. Однако с развитием новых подходов к выполнению немодульных операций, в частности операций расширения МК в рамках мультипликативных процедур по большим простым модулям [11], эффективность разработок на основе МСС существенно возросла. Это позволило говорить об указанном направлении исследований как о прорыве на пути повышения производительности СКЗИ с применением МСС.

Идея, положенная в основу предложенной технологии синтеза процедуры расширения МК, восходит к работам А. Шеноу и Р. Кумаресан [13, 14]. Она состоит в использовании вместо точных интегральных характеристик МК (ИХМК), в частности ранга, их приближенных аналогов. Это ведет к существенному упрощению алгоритмов умножения по большим простым модулям при одновременном повышении быстродействия. Вместе с тем в рамках предложенного подхода расширяется диапазон изменения результата операции умножения, что усложняет его коррекцию – приведение к остатку по рабочему модулю.

Представляемая в настоящей статье разработка по применению минимально избыточных МВС для выполнения операции умножения по большим простым модулям находится в русле

обозначенного направления исследований [11], являясь его развитием. Синтезируемые на базе минимально избыточного модулярного кодирования немодульные процедуры используют интервально-индексные характеристики – интервальный индекс (ИИ) и главный ИИ, которые вычисляются по расчетным соотношениям, столь же простым, как и выражения для расчета приближенных значений ИХМК – ранга [11]. При этом, однако, ИИ и главный ИИ чисел формируются без погрешностей. Отмеченным обстоятельством, в конечном счете, и обусловлены преимущества предлагаемого алгоритма умножения по большим простым модулям на основе минимально избыточной модулярной арифметики (МИМА).

1. Минимально избыточные модулярные системы счисления

В множестве **Z** целых чисел (ЦЧ) классическая МСС определяется набором попарно простых модулей (оснований) $m_1, m_2, \ldots, m_k$ ($k \geq 2$). В МСС ЦЧ X представляется в виде $X = (\chi_1, \chi_2, \ldots, \chi_k)$, где $\chi_i = |X|_{m_i}$ ($i = \overline{1, k}$); через $|a|_m$ обозначается элемент множества $\mathbf{Z}_m = \{0, 1, \ldots, m-1\}$, сравнимый с a (в общем случае рациональным числом) по модулю m. В МСС с основаниями $m_1, m_2, \ldots, m_k$ может быть представлено $M_k = \prod_{i=1}^{k} m_i$ ЦЧ. Обычно в качестве рабочего диапазона используются кольца $\mathbf{Z}_{M_k} = \{0, 1, \ldots, M_k - 1\}$ и $\mathbf{Z}^{-}_{M_k} = \{-\lfloor M_k/2 \rfloor, -\lfloor M_k/2 \rfloor + 1, \ldots, \lceil M_k/2 \rceil - 1\}$ соответственно наименьших неотрицательных и абсолютно наименьших вычетов по модулю M_k (через $\lfloor a \rfloor$ и $\lceil a \rceil$ обозначаются ближайшие к a соответственно слева и справа ЦЧ). При этом позиционное значение числа X может быть восстановлено по МК $(\chi_1, \chi_2, \ldots, \chi_k)$ с помощью одной из формул

$$X = \sum_{i=1}^{k} M_{i,k} \left| M_{i,k}^{-1} \chi_i \right|_{m_i} - M_k \rho_k(X); \tag{1}$$

$$X = \sum_{i=1}^{k-1} M_{i,k-1} \left| M_{i,k-1}^{-1} \chi_i \right|_{m_i} + M_{k-1} I(X), \tag{2}$$

где $M_{i,l} = M_l / m_i$, $M_l = \prod_{i=1}^{l} m_i$ ($l = k-1, k$); $\rho_k(X)$ и $I(X)$ – ИХМК, называемые соответственно рангом и ИИ числа X.

Модульные операции (сложения, вычитания и умножения) над ЦЧ A и B, заданными своими МК:

$$A = (\alpha_1, \alpha_2, \ldots, \alpha_k), \mathrm{B} = (\beta_1, \beta_2, \ldots, \beta_k) \; (\alpha_i = |A|_{m_i}, \; \beta_i = |B|_{m_i} \; (i = \overline{1, k})),$$

в МСС выполняются независимо по каждому из модулей, т. е. по правилу

$$\begin{gathered} A \circ B = (\alpha_1, \alpha_2, \ldots, \alpha_k) \circ (\beta_1, \beta_2, \ldots, \beta_k) = \\ = (|\alpha_1 \circ \beta_1|_{m_1}, \; |\alpha_2 \circ \beta_2|_{m_2}, \ldots, \; |\alpha_k \circ \beta_k|_{m_k}) \quad (\circ \in \{+, -, \cdot\}). \end{gathered} \tag{3}$$

В свойстве (3) заключается главное фундаментальное преимущество МА над арифметикой позиционных систем счисления (ПСС).

Известно, что использование числовых систем с избыточной кодовой организацией, в том числе и модулярных, зачастую позволяет существенно улучшить их арифметические или иные свойства. Применяемое в настоящей статье минимально избыточное модулярное кодирование $\Phi_{МИМСС} : \mathbf{D} \to \mathbf{Z}_{m_1} \times \mathbf{Z}_{m_2} \times \ldots \times \mathbf{Z}_{m_k}$ ($m_1, m_2, \ldots, m_k$ – модули МИМСС) предусматривает

применение рабочего диапазона **D**, мощность которого меньше мощности диапазона $\mathbf{Z}^{-}_{M_k}$ классической (неизбыточной) МСС с основаниями $m_1, m_2, \ldots, m_k$. Сущность реализуемого принципа раскрывает следующая теорема.

Теорема 1. *Для того чтобы в МСС с попарно простыми основаниями $m_1, m_2, \ldots, m_k$ ИИ $I(X)$ каждого элемента X диапазона* $\mathbf{D} = \mathbf{Z}^{-}_{2M} = \{-M, -M+1, ..., M-1\}$ $(M = m_0 M_{k-1})$ *полностью определялся компьютерным ИИ – вычетом* $\hat{I}_k(X) = \left|I(X)\right|_{m_k}$, *необходимо и достаточно, чтобы k-й модуль МСС удовлетворял условию* $m_k \geq 2m_0 + \rho$, *где* m_0 *– вспомогательный модуль, ограниченный снизу порогом* ρ; ρ *– максимальное значение ранговой характеристики* $(k-1)$*-го порядка, т. е. ИХМК* $\rho_{k-1}(X)$, *удовлетворяющей равенству*

$$\left|X\right|_{M_{k-1}} = \sum_{i=1}^{k-1} M_{i,k-1} \left| M^{-1}_{i,k-1} \chi_i \right|_{m_i} - M_{k-1} \rho_{k-1}(X).$$

При этом для $I(X)$ верны следующие расчетные соотношения:

$$I(X) = \begin{cases} \hat{I}_k(X), & \text{åñëè } \hat{I}_k(X) < m_0, \\ \hat{I}_k(X) - m_k, & \text{åñëè } \hat{I}_k(X) \geq m_k - m_0 - \rho; \end{cases} \tag{4}$$

$$\hat{I}_k(X) = \left| \sum_{i=1}^{k} R_{i,k}(\chi_i) \right|_{m_k}; \tag{5}$$

$$R_{i,k}(\chi_i) = \left\lfloor \frac{m_k}{m_i} \left| M^{-1}_{i,k} \chi_i \right|_{m_i} \right\rfloor = \left| -m_i^{-1} \left| M^{-1}_{i,k-1} \chi_i \right|_{m_i} \right|_{m_k} (i \neq k); \quad R_{k,k}(\chi_k) = \left| \frac{\chi_k}{M_{k-1}} \right|_{m_k}. \tag{6}$$

Очевидно, что МСС с модулями $m_1, m_2, ..., m_k$ и рабочим диапазоном **D**, выбираемыми в соответствии с теоремой 1, имеет наименьшую избыточность, когда выполняется равенство $m_k - 2m_0 - \rho = \left| m_k - \rho \right|_2$. Именно в этом случае МСС называется минимально избыточной. Несмотря на то, что вводимая дополнительная избыточность в МК весьма незначительна, именно она позволяет существенно упростить алгоритмы выполнения в МИМСС немодульных операций, в частности операций масштабирования.

Теорема 2. *Пусть в МИМСС с попарно простыми основаниями $m_0, m_1, \ldots, m_k$ и рабочим диапазоном* $\mathbf{D} = \mathbf{Z}^{-}_{2M}$ *задано ЦЧ* $X = (\chi_1, \chi_2, \ldots, \chi_k)$ $(X \in \mathbf{D})$ *и пусть m – вспомогательный модуль, выбираемый из условия $m \geq k$, S – некоторый натуральный масштаб. Тогда дробь X/S может быть аппроксимирована целым числом* $\hat{X} = G(X) + \Gamma(X)$,

где $G(X) = \sum_{i=1}^{k-1} \left\lfloor S^{-1} M_{i,k-1} \left| M^{-}_{i,k-1} \ \chi_i \right|_{m_i} \right\rfloor + \left\lfloor S^{-1} M_{k-1} \ I(X) \right\rfloor + \left\lfloor m^{-1} r(X) \right\rfloor;$

$\Gamma(X) = \lrcorner (2\,|r(X)|_m + k - 1) / (2m) \llcorner;$

$I(X)$ *– ИИ числа X;*

$r(X) = \sum_{i=1}^{k-1} \left\lfloor mS^{-1} \left| M_{i,k-1} \,|\, M^{-1}_{i,k-1} \ \chi_i \,|_{m_i} \right|_S \right\rfloor + \left\lfloor mS^{-1} \left| \ M_{k-1} I(X) \right|_S \right\rfloor;$

через $\lrcorner a \llcorner$ обозначается ближайшее к вещественной величине a ЦЧ:

$$\lrcorner a \llcorner = \begin{cases} \lfloor a \rfloor, & \text{åñëè } a < \lfloor a \rfloor + 0{,}5; \\ \lceil a \rceil, & \text{åñëè } a \geq \lfloor a \rfloor + 0{,}5. \end{cases}$$

При этом погрешность $\Delta(X) = X/S - \hat{X}$ *указанной аппроксимации удовлетворяет неравенству*

$$-\frac{m+k-1}{2m} \le \Delta(X) < \frac{m+k}{2m}.$$

Для реализации расчетных соотношений теоремы 2 необходимы наборы вычетов: $\mathbf{C}_i(\chi_i) = \langle R_{i,0}(\chi_i),\ R_{i,1}(\chi_i), \ldots,\ R_{i,k}(\chi_i) \rangle$ и $\mathbf{C}_k(\hat{I}_k(X)) = \langle R_{k,0}(\hat{I}_k(X)),\ R_{k,1}(\hat{I}_k(X)),\ \ldots,\ R_{k,k}(\hat{I}_k(X)) \rangle$,

где
$$R_{i,0}(\chi_i) = \left\lfloor mS^{-1} \left| M_{i,k-1} \mid M_{i,k-1}^{-1} \chi_i \mid_{m_i} \right|_S \right\rfloor; \tag{7}$$

$$R_{i,j}(\chi_i) = \left| \left\lfloor S^{-1} M_{i,k-1} \left| M_{i,k-1}^{-1} \chi_i \right|_{m_i} \right\rfloor \right|_{m_j} \quad (j = \overline{1,k}); \tag{8}$$

$$R_{k,0}(\hat{I}_k(X)) = \begin{cases} \left\lfloor mS^{-1} \left| M_{k-1} \hat{I}_k(X) \right|_{S_l} \right\rfloor, & \text{åñëè } \hat{I}_k(X) < m_0, \\ \left\lfloor mS^{-1} \left| M_{k-1}(\hat{I}_k(X) - m_k) \right|_S \right\rfloor, & \text{åñëè } \hat{I}_k(X) \ge m_k - m_0 - \rho; \end{cases} \tag{9}$$

$$R_{k,j}(\hat{I}_k(X)) = \begin{cases} \left| \left\lfloor S^{-1} M_{k-1} \hat{I}_k(X) \right\rfloor \right|_{m_j}, & \text{åñëè } \hat{I}_k(X) < m_0, \\ \left| \left\lfloor S^{-1} M_{k-1}(\hat{I}_k(X) - m_k) \right\rfloor \right|_{m_j}, & \text{åñëè } \hat{I}_k(X) \ge m_k - m_0 - \rho \end{cases} \tag{10}$$

($j = \overline{1,k}$; $i = \overline{1,k-1}$).

Фундаментальная роль в разработанном МИМА-алгоритме умножения по большим простым модулям принадлежит операциям масштабирования. Основой для выполнения этих операций служит процедура масштабирования, синтезированная на базе теорем 1 и 2 [15]. Для данной процедуры далее употребляется условное обозначение $\hat{X} = SC(X;S)$.

2. Синтез МИМА-процедуры умножения по большим простым модулям на базе рекурсивной схемы деления спуска Ферма

Пусть в МИМСС с основаниями $m_1, m_2, \ldots, m_k$ и динамическим диапазоном $\mathbf{D} = \mathbf{Z}_{2M}^-$, параметр M мощности которого удовлетворяет условию $M \ge (p-1)^2$ (p – большой простой модуль), заданы операнды операции $\gamma = |AB|_p : A = (\alpha_1, \alpha_2, ..., \alpha_k)$ и $B = (\beta_1, \beta_2, ..., \beta_k)$. Так как при $M \ge (p-1)^2$ произведение $C = AB$ в МИМСС вычисляется тривиальным образом: $C = (\gamma_1, \gamma_2, ..., \gamma_k) = (|\alpha_1\beta_1|_{m_1}, |\alpha_2\beta_2|_{m_2}, ..., |\alpha_k\beta_k|_{m_k}$, исследуемая проблема фактически сводится к разработке процедуры расширения минимально избыточного модулярного кода (МИМК) $(\gamma_1, \gamma_2, ..., \gamma_k)$ ЦЧ C на модуль p, т. е. к получению элемента $\gamma = |C|_p$ поля $\mathbf{Z}_p$. Другими словами, задача состоит в разработке для рассматриваемой МИМСС подходящей базовой процедуры деления с остатком произвольного элемента C динамического диапазона на модуль p.

Известные методы общего деления чисел, представленных в МК, можно разбить на две основные группы [8, 16]. К первой из них относятся субтрактивные методы [17, 18], базирующиеся на операциях сложения, вычитания, определения знака числа и контроля аддитивного переполнения. Чаще всего соответствующие процедуры деления осуществляют вычитание из

делимого кратных делителя до тех пор, пока не будет получен остаток от деления делимого на делитель, а значит, и неполное частное.

Вторую группу составляют методы деления, которые можно квалифицировать как мультипликативные [7, 16, 19–21]. Методы данного класса реализуют различные варианты итеративной схемы, в рамках которой деление чисел сводится к операциям умножения. На практике наибольшее распространение получили рекурсивные схемы деления Ньютона, спуска Ферма и др. Благодаря поразрядному характеру операции модульного умножения методы второй группы позволяют значительно уменьшить время выполнения общего деления целых чисел в МСС.

Как показывает анализ, согласно критерию минимума временных затрат при приемлемом суммарном количестве необходимых таблиц в качестве базового метода для вычисления $\gamma = |C|_p$ целесообразно принять метод деления, реализующий МИМА-версию рекурсивной схемы спуска Ферма. Суть представляемого здесь метода состоит в получении некоторого целочисленного приближения $\hat{\mathbf{Q}}$ к дроби C/p, погрешность которого $\Delta = C/p - \hat{\mathbf{Q}}$ удовлетворяет заданному условию, например $0 \le \Delta < 1$ или $-\varepsilon \le \Delta < \varepsilon$, где ε – фиксированная константа из интервала [1/2; 1). Обычно в качестве таких приближений используются ЦЧ $Q = \lfloor C/p \rfloor$ или $Q^{-} = \lfloor C/p \rceil$, которым по лемме Евклида отвечают соответственно остаток $\gamma = |C|_p$ и симметрический остаток $\gamma^{-} = |C|_p^{-}$.

Алгоритм деления чисел C и p методом спуска Ферма имеет рекурсивную структуру. В ходе первой итерации для дроби C/p по некоторому правилу находится начальное целочисленное приближение Q_1. Если его погрешность $\Delta_1 = C/p - Q_1 = C_1/p$, где $C_1 = C_0 - pQ_1$, $C_0 = C$, согласуется с требованием к точности операции, то процесс деления заканчивается. В противном случае осуществляется переход к следующей итерации. На i-й итерации ($i > 1$) в соответствии с применяемым правилом определяется целочисленное приближение q_i к погрешности $\Delta_{i-1} = C_{i-1}/p = (C_{i-2} - pQ_{i-1})/p$ $(i-1)$-й итерации. Если при этом текущая погрешность $\Delta_I = C_{i-1}/p - Q_i = (C_{i-1} - pQ_i)/p = C_i/p$ выполняемой операции не выходит за установленные границы, то деление чисел C и p завершается. В противном случае описанный рекурсивный процесс продолжается.

Правило аппроксимации дробей C_{i-1}/p целыми числами Q_i $(i \ge 1)$ конструируется так, чтобы последовательность $|C_0|, |C_1|, |C_2|, \ldots$, а значит, и последовательность $|\Delta_1|, |\Delta_2|, \ldots$ строго убывали. При соответствующем ограничении на точность конечного результата это обеспечивает сходимость алгоритма деления.

Предположим, что процесс деления завершился на r-й итерации. Тогда согласно изложенному

$$C/p = \sum_{i=1}^{r} Q_i + C_r/p\,. \tag{11}$$

Так как погрешность $\Delta_r = C_r/p$ заключительной итерации находится в требуемых границах, из (11) следует, что искомым приближением к дроби C/p является ЦЧ $\hat{Q} = \sum_{i=1}^{r} Q_i$.

Применяемый способ целочисленной аппроксимации дробей C_{i-1}/p $(i \ge 1)$ предусматривает:
− формирование для p приближенного значения вида

$$\tilde{p} = S = \hat{p}\tilde{S}\ , \tag{12}$$

где $\hat{p} \in \{1, 2, ..., \Lambda - 1\}$; Λ и $\tilde{S}$ – натуральные числа, определяющие точность приближения;

– использование для целочисленной оценки дробей C_{i-1}/p операций масштабирования C_{i-1} на масштаб $S = \widetilde{p}$ (см. (12)), выполняемых с помощью процедуры $SC(C_{i-1}; S)$.

Существенной особенностью СКЗИ является наличие в наборе базовых параметров компонент долговременного использования [9–11, 22, 23]. Это позволяет все вычисления, связанные с параметрами данного типа, выделить в специальный вычислительный процесс и выполнять его предварительно до основного вычислительного процесса, реализуемого в реальном времени. Поскольку к предварительным вычислениям не выдвигаются столь жесткие требования по временным затратам, как к основному вычислительному процессу, то учет отмеченного фактора при синтезе базовых процедур МВТ для криптосистем является важным и эффективным оптимизирующим средством. Сказанное в полной мере относится к параметру p. Поэтому приближение $\widetilde{p}$ для модуля p, которое служит рабочим масштабом S в предлагаемом алгоритме деления, может быть определено в рамках предварительных вычислений. Кроме того, на этапе предварительных вычислений осуществляется и формирование таблиц, необходимых для применяемой процедуры масштабирования SC (набора вычетов для расчета ИИ (см. теорему 2)), а также таблиц, описываемых формулами (7)–(10).

В разработанной МИМА-версии метода деления C на p по схеме спуска Ферма явно используется МИМК $(\pi_1, \pi_2, \ldots, \pi_k)$ модуля p $(\pi_i = |p|_{m_i} (i = \overline{1,k}))$. Предполагается, что требуемый МК формируется в процессе выбора p. Одновременно с этим при необходимости может быть получен и позиционный код числа p, например двоичный код:

$$p = \sum_{j=0}^{b-1} p_j 2^j = (p_{b-1}p_{b-2}\ldots p_0)_2 (b = \lceil \log_2 p \rceil; p_j \in \{0,1\}, p_{b-1} = 1) \tag{13}$$

или полиадический код:

$$p = \sum_{i=1}^{l} p_i M_{i-1} = \langle p_l p_{l-1} \ldots p_1 \rangle (1 < l < k; p_i \in \mathbf{Z}_{m_i},\ p_l \neq 0)\,. \tag{14}$$

Формирование для p аппроксимирующего значения (11) по позиционным представлениям (13) и (14) осуществляется соответственно по правилам

$$\hat{p} = (p_{b-1}p_{b-2}\ldots p_{b-\lambda})_2 = \lfloor p / 2^{b-\lambda} \rfloor, \tilde{S} = 2^{b-\lambda} \tag{15}$$

и

$$\hat{p} = \lfloor p_l / 2^{(u-\lambda+1)S(\lambda-u)} \rfloor, \tilde{S} = M_{l-1} 2^{(u-\lambda+1)S(\lambda-u)}, \tag{16}$$

где $\lambda = \lceil \log_2 \Lambda \rceil$ – разрядность величины $\hat{p}$; u – номер старшего ненулевого бита в двоичном представлении цифры p_l полиадического кода числа p (нумерация разрядов в двоичных кодах начинается с 0); $S(\lambda - u) = \begin{cases} 0, \text{åñëè } \lambda \geq u; \\ 1, \text{åñëè } \lambda < u. \end{cases}$

Отметим, что конкретный вид множителя $\widetilde{S}$ масштаба S оказывает существенное влияние на сложность расчета комплекта рабочих таблиц для процедуры масштабирования SC. С этой точки зрения более предпочтительной является аппроксимация модуля p, выполняемая на основе полиадического представления (14), т. е. согласно (16).

В случае, когда процесс выбора модуля p предусматривает получение лишь его МИМК $(\pi_1, \pi_2, \ldots, \pi_k)$, компоненты (15) или (16) требуемого масштаба (12) можно найти с помощью алгоритма кодового преобразования или алгоритма формирования ИХМК соответствен-

но [24]. При этом достаточно воспользоваться усеченными версиями указанных алгоритмов, обеспечивающими вычисления лишь некоторых приближений старших коэффициентов рассматриваемых позиционных представлений числа p. В частности, модуль p можно аппроксимировать при помощи приближенного значения $\hat{p}_{n-1}$ старшей цифры его n-разрядного кода ПСС с основанием 2^{32}:

$$p = (p_{n-1}p_{n-2}\dots p_0)_{2^{32}}\,(n = \lceil b/32 \rceil;\, p_j \in Z_{2^{32}}\,(j = \overline{0,n-1}),\, p_{n-1} \neq 0),$$

который определяется по (2^{32})-ичным кодам $(p_{n-1}^{(i)}p_{n-2}^{(i)}\dots p_0^{(i)})_{2^{32}}\,(i = \overline{1,k})$ слагаемых

$$P_i = \begin{cases} M_{i,k-1}\left|M_{i,k-1}^{-1}\pi_i\right|_{m_i} \text{ при } i \neq k, \\ M_{k-1}I(p) \text{ при } i = k \end{cases}$$

интервально-модулярной формы числа p:

$$p = \sum_{i=1}^{k} P_i = \sum_{i=1}^{k-1} M_{i,k-1}\left|M_{i,k-1}^{-1}\pi_i\right|_{m_i} + M_{k-1}I(p).$$

Требуемые расчетные соотношения для $\hat{p}_{n-1}$ имеют вид

$$\hat{p}_{n-1} = \left|\sum_{i=1}^{k} p_{n-1}^{(i)}\right|_{2^{32}} + \Pi_{n-2}; \tag{17}$$

$$\Pi_{n-2} = \left\lfloor 2^{-32}\sum_{i=1}^{k} p_{n-2}^{(i)} \right\rfloor. \tag{18}$$

Пусть u – номер старшего ненулевого бита в двоичном коде величины (17). Тогда в качестве приближения для p можно принять

$$\tilde{p} = \hat{p}\tilde{S} = \left\lfloor \hat{p}_{n-1} / 2^{(u-\lambda+1)S(\lambda-u)} \right\rfloor 2^{32(n-1)+(u-\lambda+1)S(\lambda-u)}. \tag{19}$$

Аналогично изложенному модуль p можно аппроксимировать при помощи приближенного значения $\hat{p}_l$ старшей l-й цифры p_l полиадического кода (14). Требуемое расчетное соотношение для $\hat{p}_l$ имеет вид

$$\hat{p}_l = \left|\hat{\rho}_{l-1}(p) + \hat{I}_l(p)\right|_{m_l}, \tag{20}$$

где

$$\hat{\rho}_{l-1}(p) = \left\lfloor m_{l-1}^{-1}\sum_{i=1}^{l-1} R_{i,l-1}(\pi_i) \right\rfloor; \tag{21}$$

$$\hat{I}_l(p) = \left|\sum_{i=1}^{l} R_{i,l}(\pi_i)\right|_{m_l}, \tag{22}$$

вычеты $R_{i,j}(\pi_i)$ $(i = \overline{1,j};\, j = l-1,l)$ вычисляются по формулам (6).

Искомое приближение для p осуществляется по правилу

$$\tilde{p} = \hat{p}\tilde{S} = \left\lfloor \hat{p}_l / 2^{(u-\lambda+1)S(\lambda-u)} \right\rfloor M_{l-1} 2^{(u-\lambda+1)S(\lambda-u)}. \quad (23)$$

Отметим, что ввиду малости величин (18) и (21) ($\Pi_{n-2} < k$, $\hat{\rho}_{l-1}(p) \le \rho_{l-1,\max} \le l-2$) для аппроксимации p вместо (17) и (20) можно использовать более простые оценки старших коэффициентов позиционных представлений p:

$$\hat{p}_{n-1} = \left| \sum_{i=1}^{k} p_{n-1}^{(i)} \right|_{2^{32}} \quad (24)$$

и

$$\hat{p}_l = \hat{I}_l(p). \quad (25)$$

Замечание 1. Все приведенные способы аппроксимации модуля p дают приближения $\tilde{p}$ с недостатком. Они основаны на усечении тех или иных позиционных представлений p. Для получения приближений $\tilde{p}$ с избытком достаточно используемые старшие части позиционных кодов p увеличить на некоторые минимальные поправки. В частности, для аппроксимации p с избытком по правилам типа (15) и (16) величину $\hat{p}$ следует увеличить на 1, а в случае применения правил (19) и (23) базовые ИХМК (17) и (20) можно инкрементировать на поправку $\exp_2((u - \lambda + 1)S(\lambda - u)) + 1$ (см. теорему 2 [24]). Для аппроксимации p с избытком на основе (24) и (25) базовые ИХМК надлежит увеличить соответственно на $\exp_2((u - \lambda + 1)S(\lambda - u)) + k$ и $\exp_2((u - \lambda + 1)S(\lambda - u)) + \rho_{l-1,\max} + 1$.

Изложенное позволяет сформулировать *алгоритм деления* на простой модуль p в МИМСС с основаниями ($m_1, m_2, \ldots, m_k$), диапазоном $\hat{\mathbf{D}} = \mathbf{Z}_p = \{0, 1, \ldots, p-1\}$ исходных данных и динамическим диапазоном $\mathbf{D} = \hat{\mathbf{D}}_{2M}^{-}$.

Входные данные алгоритма: произведение $C = AB$ ($A, B \in \hat{\mathbf{D}}$; $C \in \mathbf{D}$) и рабочий простой модуль p, представленные в МИМСС, $C = (\gamma_1, \gamma_2, \ldots, \gamma_k)$, $p = (\pi_1, \pi_2, \ldots, \pi_k)$.

Предварительно вычисляемые данные:

− приближенное значение $\tilde{p}$ модуля p, применяемое в качестве масштаба $S = \tilde{p}$ в базовой процедуре масштабирования (см. (12), (15), (16), (19), (23));

− таблицы ИИ *TIIi*, в память которых помещаются вычеты (6) согласно правилу $TIIi[\chi] = R_{i,k}(\chi)\ (\chi \in \mathbf{Z}_{m_i}; i = \overline{1,k})$;

− таблицы масштабирования *TSCij*, формируемые в соответствии с (7)–(10): $TSCij[\chi] = R_{i,j}(\chi)\ (\chi \in \mathbf{Z}_{m_i}; i = \overline{1,k}; j = \overline{0,k})$;

− таблица поправок для операции масштабирования: $TCOSC[x] = \lfloor x/m \rfloor + \lfloor (2|x|_m + k - 2)/(2m) \rfloor$ ($x \in \mathbf{Z}_{k(m-1)+1}$; m – вспомогательный модуль, используемый в алгоритме масштабирования (см. теорему 2)).

Д.1. Положить $s = 1$, $Q_0 = 0$, $C_1 = (\gamma_1^{(1)}, \gamma_2^{(1)}, \ldots, \gamma_k^{(1)}) = (\gamma_1, \gamma_2, \ldots, \gamma_k) = C$.

Д.2. По МИМК $(\gamma_1^{(s)}, \gamma_2^{(s)}, \ldots, \gamma_k^{(s)})$ числа C_s и масштабу $S = \tilde{p}$ (см. (12)) с помощью базовой процедуры масштабирования *SC* найти целочисленную оценку $\hat{C}_s = (\hat{\gamma}_1^{(s)}, \hat{\gamma}_2^{(s)}, \ldots, \hat{\gamma}_k^{(s)})$ дроби C_s/S. Для этого необходимо:

а) рассчитать компьютерный ИИ $\hat{I}_k(C_s)$ числа C_s (см. (5)):

$$\hat{I}_k(C_s) = \left| \sum_{i=1}^{k} TIIi[\gamma_i^{(s)}] \right|_{m_k};$$

б) вычислить величины

$$r(C_s) = \sum_{i=1}^{k-1} TSCi0[\gamma_i^{(s)}] + TSCk0[\hat{I}_k(C_s)];$$

$$R_j(C_s) = \left| \sum_{i=1}^{k-1} TSCij[\gamma_i^{(s)}] + TSCkj[\hat{I}_k(C_s)] \right|_{m_j} \quad (j = \overline{1,k});$$

в) определить цифры МИМК $(\hat{\gamma}_1^{(s)}, \hat{\gamma}_2^{(s)}, ..., \hat{\gamma}_k^{(s)})$ оценки $\hat{C}_s$ дроби C_s/S по формуле

$$\hat{\gamma}_j^{(s)} = \left| R_j(C_s) + TCOSC[r(C_s)] \right|_{m_j} \quad (j = \overline{1,k}).$$

Оценка $\hat{C}_s$ принимается в качестве приближения к дроби $\Delta_{s-1} = C_s/p$.

Д.3. Получить МИМК s-го приближения к частному: $Q_s = Q_{s-1} + \hat{C}_s$, а также невязки

$$C_{s+1} = C_s - p\hat{C}_s. \tag{26}$$

Д.4. Проверить условие $\hat{C}_s \notin \{-1, 0, 1\}$. При его выполнении s увеличить на 1 ($s = s + 1$) и затем перейти к шагу Д.2. В случае, когда $\hat{C}_s \notin \{-1, 0, 1\}$ (пусть это имеет место при $s = r$, $r \geq 1$), итеративный процесс деления C на p завершить. При этом в качестве искомых значений «неполного частного» и «остатка» зафиксировать соответственно

$$\hat{Q} = Q_r = \sum_{s=1}^{r} \hat{C}_s \tag{27}$$

и

$$\hat{\gamma} = C_{r+1} = C - p\hat{Q}. \tag{28}$$

3. Исследование корректности алгоритма

Остановимся кратко на вопросах точности и сходимости сформулированного алгоритма. Из (26) после деления на p получаем

$$\Delta_{s-1} - \Delta_s = \hat{C}_s. \tag{29}$$

Суммирование равенств (29) по s дает $\sum_{s=1}^{r} \Delta_{s-1} - \sum_{s=1}^{r} \Delta_s = \sum_{s=1}^{r} \hat{C}_s$ или $\Delta_r = \Delta_0 - \sum_{s=1}^{r} \hat{C}_s$. Отсюда с учетом (26), (27) ввиду $\Delta_0 = C/p$ заключаем, что погрешность операции деления ЦЧ C на модуль p, выполняемой рассмотренным способом, определяется соотношением

$$\Delta_{\text{ä},r} = C / p - \hat{Q} = \Delta_r = C_r / p - \hat{C}_r . \tag{30}$$

Так как $\hat{C}_r$ представляет собой результат масштабирования числа C_r на масштаб S (см. шаг Д.2), то

$$C_r / S = \hat{C}_r + \Delta_{\text{ì},r}, \tag{31}$$

где $\Delta_{\text{м},r}$ – погрешность указанной операции масштабирования, которая согласно теореме 2 удовлетворяет неравенству

$$\left|\Delta_{\text{м},r}\right| < \Delta_{\text{м}} < 1 \quad (\Delta_{\text{м}} = (m+k)/(2m); m \geq k). \tag{32}$$

Используя (31), выражение (30) можно записать в виде

$$\Delta_{\text{ä},r} = \frac{S}{p}(\hat{C}_r + \Delta_{\text{ì},r}) - \hat{C}_r . \tag{33}$$

Найдем условие, при котором погрешность алгоритма деления удовлетворяет неравенству $\left|\Delta_{\text{д},r}\right| \leq \Delta_{\text{д}} < 1$ ($\Delta_{\text{д}}$ – некоторый верхний порог). Согласно шагу Д.4 величина $\hat{C}_r$ может принимать лишь три значения: –1, 0, 1. Подставляя их поочередно в (33) и применяя (32), для $\Delta_{\text{д},r}$ получим соответственно оценки

$$-\frac{S}{p}(1+\Delta_{\text{м}}) + 1 < \Delta_{\text{д},r} < -\frac{S}{p}(1-\Delta_{\text{м}}) + 1; \tag{34}$$

$$-\frac{S}{p}\Delta_{\text{м}} < \Delta_{\text{д},r} < \frac{S}{p}\Delta_{\text{м}}; \tag{35}$$

$$\frac{S}{p}(1-\Delta_{\text{м}}) - 1 < \Delta_{\text{д},r} < \frac{S}{p}(1+\Delta_{\text{м}}) - 1. \tag{36}$$

Потребуем, чтобы интервалы изменения погрешности $\Delta_{\text{д},r}$, определяемые неравенствами (34)–(36), включались в интервал $(-\Delta_{\text{д}}; \Delta_{\text{д}})$. Простейшие выкладки показывают, что данное условие выполняется при следующем ограничении на масштаб S:

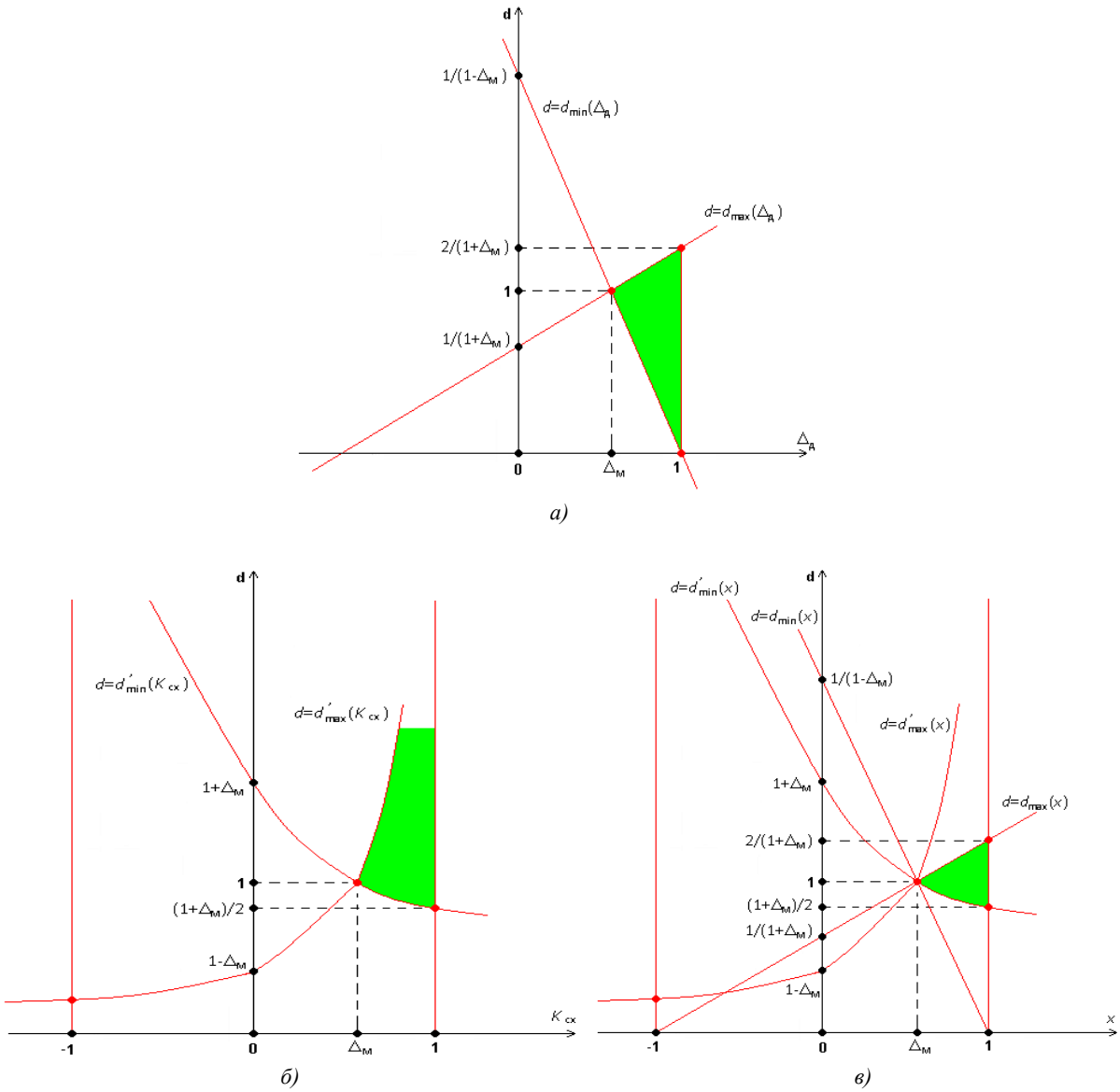
$$\frac{1-\Delta_{\text{д}}}{1-\Delta_{\text{м}}} < \frac{S}{p} < \frac{1+\Delta_{\text{д}}}{1+\Delta_{\text{м}}} \quad (\Delta_{\text{м}} \leq \Delta_{\text{д}} < 1). \tag{37}$$

Область изменения отношения $d = S/p = \tilde{p}/p$ при возможных значениях порогов $\Delta_{\text{д}}$ и $\Delta_{\text{м}}$ ограничена прямыми $d = d_{\min}(\Delta_{\text{д}}) = (1 - \Delta_{\text{д}})/(1 - \Delta_{\text{м}})$, $d = d_{\max}(\Delta_{\text{д}}) = (1 + \Delta_{\text{д}})/(1 + \Delta_{\text{м}})$ и $\Delta_{\text{д}} = 1$ (рис., *а*). Значению $\Delta_{\text{д}} = 1$ отвечает наибольший интервал изменения $d = S/p$ ($S/p \in \in (0; 2/(1 + \Delta_{\text{м}}))$. С уменьшением $\Delta_{\text{д}}$ интервал изменения для S/p сужается.

Перейдем к обоснованию корректности алгоритма деления.

Найдем условие сходимости реализуемого в рамках алгоритма итеративного процесса. С этой целью получим выражение для отношения C_{s+1}/C_s ($s = \overline{1,r}$). В соответствии с (26) имеем

$$\frac{C_{s+1}}{C_s} = 1 - \frac{p\hat{C}_s}{C_s} . \tag{38}$$



Область изменения $d = S/p$, определяемая параметрами:
а) $\Delta_м$ и $\Delta_д$; *б)* $\Delta_м$ и $K_{сх}$; *в)* точности и сходимости алгоритма деления

Поскольку $\hat{C}_s$ служит приближением к дроби C_s/S, которое определяется процедурой масштабирования SC с погрешностью $\Delta_{м,s}$, удовлетворяющей неравенству $|\Delta_{м,s}| < \Delta_м$, (38) можно записать в виде

$$\frac{C_{s+1}}{C_s} = 1 - \frac{p}{C_s}\left(\frac{C_s}{S} - \Delta_{м,s}\right) = 1 - \frac{p}{C} + \frac{p}{C_s}\Delta_{м,s}. \qquad (39)$$

Если $|C_s| < S$, то $\hat{C}_s \in \{-1, 0, 1\}$, что является признаком конца итеративного процесса (см. шаг Д.4). Пусть $|C_s| \ge S$. Тогда из (39) для отношения C_{s+1}/C_s следует

$$1 - \frac{p}{S}(1 + \Delta_м) < \frac{C_{s+1}}{C_s} < 1 - \frac{p}{S}(1 - \Delta_м). \qquad (40)$$

Очевидно, что рекурсивный процесс деления, выполняемый в рамках алгоритма деления, будет сходящимся, если интервал изменения величин C_{s+1}/C_s ($s = \overline{1,r}$), определяемый неравенством (40), включается в некоторый промежуток $[-K_{сх}; K_{сх}] \subset (-1; 1)$, где $K_{сх}$ – выбранный надлежащим образом ограничительный порог, $K_{сх} < 1$. Указанное требование приводит к следующему дополнительному ограничению (см. (37), рис., *а*) на выбор масштаба *S*:

$$\frac{1-K_{сх}}{1-\Delta_м} < \frac{p}{S} < \frac{1+K_{сх}}{1+\Delta_м} (\Delta_м \le K_{сх} < 1)$$

или в эквивалентной форме

$$\frac{1+\Delta_м}{1+K_{сх}} < \frac{S}{p} < \frac{1-\Delta_м}{1-K_{сх}} (\Delta_м \le K_{сх} < 1). \quad (41)$$

Область изменения отношения $d = S/p = \tilde{p}/p$, которая определяется неравенством (41), ограничена участками гипербол $d = d'_{\min}(K_{сх}) = (1+\Delta_м)/(1+K_{сх})$, $d = d'_{\max}(K_{сх}) = (1-\Delta_м)/(1-K_{сх})$ и прямой $K_{сх} = 1$ (рис., *б*). Значению $K_{сх} = 1$ отвечают $S/p \ge (1+\Delta_м)/2$. С уменьшением $K_{сх}$ промежуток изменения отношения *S*/*p* сужается.

Пересечением областей изменения $d = S/p$, показанных на рис., *а* и *б*, является область плоскости *xOd* (рис. 1, *в*), ограниченная снизу участком гиперболы $d = d'_{\min}(x) = (1+\Delta_м)/(1+x)$, сверху – прямой $d = d_{\max}(x) = (1+x)/(1+\Delta_м)$ и справа – прямой $x = 1$.

Таким образом, искомым ограничительным условием на выбор масштаба $S = \tilde{p}$ (см. (12)) служит неравенство

$$\frac{1+\Delta_м}{1+K_{сх}} < \frac{S}{p} < \frac{1+\Delta_д}{1+\Delta_м} \ (K_{сх}, \Delta_д \in [\Delta_м; 1)). \quad (42)$$

В выражении (42) пороги $K_{сх}$ и $\Delta_д$ могут принимать значения из промежутка $[\Delta_м; 1)$ независимо друг от друга. При $\Delta_д = K_{сх} = 1$ (42) дает

$$(1+\Delta_м)/2 < S/p < 2/(1+\Delta_м), \quad (43)$$

а при $K_{сх} = \Delta_д = (1+\Delta_м)/2$ – неравенство

$$\frac{2(1+\Delta_м)}{3+\Delta_м} < \frac{S}{p} < \frac{3+\Delta_м}{2(1+\Delta_м)}. \quad (44)$$

Пусть, например, $\Delta_м = 0{,}6$. Тогда (43) и (44) соответственно дают $0{,}8 < S/p < 1{,}25$ и $8/9 < S/p < 9/8$.

Приведенные математические выкладки доказывают справедливость следующей теоремы.

Теорема 3. *Пусть модуль p аппроксимирован целым числом* $\tilde{p}$ (см. (12)), *удовлетворяющим условию*

$$\frac{1+\Delta_м}{1+K_{сх}} < \frac{\tilde{p}}{p} < \frac{1+\Delta_д}{1+\Delta_м}, \quad (45)$$

где $\Delta_м = (m+k)/(2m)$ – верхний порог абсолютной погрешности процедуры масштабирования *SC* (*m* – вспомогательный модуль, $m \ge k$); $K_{сх}$, $\Delta_д \in [\Delta_м; 1)$.

Тогда процесс деления в МИМСС с основаниями $m_1, m_2, \ldots, m_k$ и динамическим диапазоном $\mathbf{D} = \mathbf{Z}_{2M}^{-}$ $\left(M > (p-1)^2\right)$ числа $C \in \mathbf{D}$ на p, реализуемый по алгоритму деления при помощи процедуры масштабирования SC с применением масштаба $S = \widetilde{p}$, является сходящимся с коэффициентом сходимости $K_{\text{сх}}$. При этом достигаемая абсолютная погрешность операции деления $\Delta(C) = |C/p - \hat{Q}|$ ($\hat{Q}$ – «неполное частное» (см. (27)) ограничена сверху порогом $\Delta_{\text{д}} < 1$.

Замечание 2. С точки зрения реализации существенной особенностью представленного метода деления является свойство невязок (26) принимать как положительные, так и отрицательные значения. Это требует применения МИМСС с симметричным диапазоном $\mathbf{D} = \mathbf{Z}_{2M}^{-}$. Знак невязки C_{s+1} ($s = \overline{1,r}$) зависит прежде всего от того, с недостатком или с избытком аппроксимирован модуль p масштабом $S = \widetilde{p}$ (см. замечание 1). Пусть, например, $p < S$ и $C_s > 0$. Тогда $C_s/p > C_s/S$ и, следовательно, $C_s/p > \hat{C}_s + \Delta_{\text{i},s}$ ($\Delta_{\text{м},s} \in (-\Delta_{\text{м}}; \Delta_{\text{м}})$) или $C_{s+1} \geq \left| p\Delta_{\text{м},s} \right|$. Отсюда видно, что в рассматриваемом случае все невязки, кроме r-й, положительны, но при $s = r$ возможна ситуация, когда $\Delta_{\text{м},r} < 0$ и $C_{r+1} < 0$. Если же $p > S$, то, как нетрудно проверить, последовательность невязок $C_1 = C > 0$, $C_2, \ldots, C_r$ является знакопеременной, а невязка C_{r+1} также может быть как положительной, так и отрицательной.

Замечание 3. Согласно (26), (28), (30) и теореме 3

$$\Delta(C) = \left|\Delta_r\right| = \left|C_{r+1}/p\right| = \left|\hat{\gamma}/p\right| < \Delta_{\text{д}} < 1.$$

Поэтому

$$-p < \hat{\gamma} < p. \tag{46}$$

Отсюда следует, что неполное частное $Q = \lfloor C/p \rfloor$ и остаток $\gamma = |C|_p$ при делении C на p связаны со своими аналогами $\hat{Q}$ и $\hat{\gamma}$, получаемыми алгоритмом деления, соотношениями

$$Q = \begin{cases} \hat{Q} - 1, & \text{åñëè } \hat{\gamma} < 0, \\ \hat{Q}, & \text{åñëè } \hat{\gamma} \geq 0; \end{cases} \tag{47}$$

и

$$\gamma = \begin{cases} \hat{\gamma} + p, & \text{åñëè } \hat{\gamma} < 0, \\ \hat{\gamma}, & \text{åñëè } \hat{\gamma} \geq 0. \end{cases} \tag{48}$$

Реализация корректирующих равенств (47) и (48) требует определения знака числа $C_{r+1} = \hat{\gamma}$. Эта операция может быть выполнена с помощью процедуры формирования ИХМК или путем преобразования МИМК числа $\hat{\gamma}$ в позиционный код [24].

4. Оценка эффективности МИМА-процедуры умножения по модулю

Умножение по модулю p, реализуемое посредством алгоритма деления, на мультипроцессорной или многомашинной системе модулярной обработки информации (СМОИ), включающей независимые друг от друга тракты по вспомогательному модулю $m \geq k$, основным модулям $m_1, m_2, \ldots, m_k$, а также тракт для формирования интервально-индексной характеристики, занимает время

$$t_{\text{у,СМОИ}} = (r+1)t_{\text{му}} + r(k+3)t_{\text{сл}}, \tag{49}$$

где r – количество итераций алгоритма деления; k – число оснований базовой МИМСС; $t_{\text{му}}$ и $t_{\text{сл}}$ – времена выполнения операций модульного умножения и сложения ЦЧ соответственно. Предполагается, что суммирование вычетов по модулям $m_1, m_2, \ldots, m_k$ в рамках процедуры масштабирования *SC* производится по аккумулятивно-табличному методу [25]. Процессы вычисления модульных сумм в трактах СМОИ не распараллеливаются. Извлечение очередных вычетов из таблиц совмещается во времени с соответствующими текущими операциями сложения.

Время выполнения разработанного МИМА-алгоритма умножения по модулю p с применением только одной ПЭВМ составляет

$$t_{\text{у,ПЭВМ}} = k(r+1)t_{\text{му}} + rk(k+6)t_{\text{сл}}. \qquad (50)$$

Пусть, например, в качестве инструментальной платформы используются процессоры типа Intel Pentium 4 (3 ГГц), для которых $t_{\text{сл}}$=2 нс, время извлечения элемента таблицы $t_{\text{ч}}$=1,14 нс. Предположим, что для реализации операций умножения по модулям МИМСС применяется индексный метод [25]. Тогда $t_{\text{му}} = 3t_{\text{ч}} + t_{\text{сл}} = 5{,}42$ нс. В табл. приведены рассчитанные согласно (49) и (50) времена выполнения операции умножения по модулю p с помощью алгоритма деления для некоторых значений разрядности $\lceil \log_2 p \rceil$ модуля p и отвечающих ему значений числа k оснований МИМСС и количества r итераций применяемой рекурсивной схемы деления. Основания $m_1, m_2, \ldots, m_k$ МИМСС выбираются из промежутка $(2^{15}; 2^{16})$.

Отдельное большое число в МИМСС представляется кодом длиной $L_{\text{МИМК}} = 2k$ байтов. Каждая цифра занимает два байта. Если, например, p имеет разрядность 2462 бита, то число оснований базовой МИМСС составляет $k = 308$ и в этом случае $L_{\text{МИМК}} = 616$ байтов. В рамках развиваемого табличного подхода к созданию алгоритмов умножения по большим простым модулям главная часть требуемой памяти приходится на таблицы, прежде всего таблицы масштабирования и таблицы интервального индекса. Для их размещения необходима соответственно память $M_{TSC} = 2(k+1)\sum_{i=1}^{k} m_i$ и $M_{TII} = 2\sum_{i=1}^{k} m_i$ байтов. В случае $\lceil \log_2 p \rceil = 2462$ суммарный объем памяти для размещения указанных таблиц не превышает 11,935 Гб. Таблица *TCOSC* поправок для операции масштабирования занимает не более 94,864 байта.

Времена выполнения МИМА-алгоритма умножения по модулю на основе рекурсивной схемы спуска Ферма с использованием процессоров Intel Pentium 4 (3 ГГц)

Значения базовых параметров			Времена выполнения алгоритма, нс	
$\lceil \log_2 p \rceil$	k	r	$t_{\text{у, ПЭВМ}}$	$t_{\text{у, СМОИ}}$
64	9	4	1323,9	123,1
64	9	5	1642,68	152,52
128	17	5	4462,84	232,52
128	17	6	5336,98	277,94
256	33	6	16696,02	469,94
256	33	7	19448,88	547,36
512	66	5	49666,32	722,52
512	66	8	79251,48	1152,78
1024	133	4	151500,3	1115,1
2462	308	4	782042,8	2515,1

Предложенная табличная конфигурация МИМА-алгоритма умножения позволяет свести все вычисления к операциям извлечения из табличной памяти наборов вычетов и их суммирования на 32-битовом сумматоре. Что касается известных МА-алгоритмов умножения по моду-

лю p [11], то они используют основания разрядностью 32 бита. Это затрудняет применение табличного метода, что ведет к снижению производительности. Для наиболее близкого модулярного аналога к синтезированному алгоритму временные затраты при реализации на однопроцессорной ЭВМ можно оценить по формуле

$$t'_{\text{у,ПЭВМ}} = n((2n+7)t'_{\text{му}} + (8n+10)t_{\text{сл}} + 2(n+1)t_{\text{ч}}), \qquad (51)$$

где n – число оснований каждой из двух применяемых в аналоге [11] МСС; $t'_{\text{му}}$ – длительность операции умножения по модулям МСС, выполняемой на позиционном умножителе с приведением произведения 32-битовых чисел к остатку по соответствующему модулю. Принятая в качестве основы инструментальная платформа обеспечивает $t'_{\text{му}}$ = 84 нс, $t_{\text{сл}}$ = 2 нс, $t_{\text{ч}}$ =1,14 нс. С учетом этого при 2462-битовых p, которым отвечает n = 80, выражение (51) дает $t'_{\text{у,ПЭВМ}}$ = 1 241 014,4 нс. Аналогичная характеристика для синтезированного МИМА-алгоритма принимает значение $t_{\text{у,ПЭВМ}}$ = 782 042,8 нс (см. табл.). Таким образом, достигается 1,5869-кратное сокращение временных затрат. При этом, однако, классический алгоритм требует значительно меньшего объема табличной памяти.

Заключение

Изложенные в настоящей статье исследования по созданию эффективных методов и алгоритмов умножения по большим простым модулям с использованием МСС показывают, что арифметика МИМСС обеспечивает принципиально новые возможности для оптимизации базовых немодульных процедур. Основные результаты представленной разработки состоят в следующем:

1. На базе быстросходящейся рекурсивной МИМА-схемы деления спуска Ферма и процедуры масштабирования табличного типа на масштаб, адаптируемый на этапе предварительных вычислений к рабочему модулю, разработан новый алгоритм умножения для СКЗИ. В отличие от известных модулярных аналогов, использующих пару операций расширения кода, данный алгоритм обладает абсолютной однородностью модульных трактов, минимизирует количество связей между ними и сокращает примерно в два раза время немодульных сегментов между смежными операциями умножения по модулям МИМСС. Это в значительной мере нейтрализует негативное влияние на общую производительность процесса умножения рекурсивного характера реализуемой вычислительной схемы.

2. В базовой процедуре масштабирования применены точные ИХМК – ИИ и главный ИИ. Это позволило ограничить конечный результат умножения промежутком $(-p;\ p)$ и тем самым исключить свойственный известным алгоритмам умножения по модулю p этап коррекции результирующего «остатка» на аддитивную поправку, кратную p.

3. Предложенный подход к выполнению операции умножения по модулю p предполагает использование отдельного тракта для расчета интервально-индексных характеристик, благодаря чему отпадает необходимость в средствах управления, которые применяются в аналогах для аддитивного расщепления неточной ИХМК – ранга. Наряду с двухшаговой организацией немодульного сегмента (расширения МК) наличие средств управления вносит существенный вклад в неоднородность модульных трактов.

4. Синтез созданного МИМА-алгоритма умножения для систем защиты информации осуществлен с учетом стратегии, которая нацелена на обеспечение максимального повышения производительности вычислительного процесса, реализуемого криптосистемой в реальном времени, за счет увеличения объема предварительных вычислений, главным образом связанных с формированием рабочего комплекта таблиц.

Разработка выполнена в рамках государственной программы научных исследований «Инфотех».

Список литературы

1. Высокоскоростные методы и системы цифровой обработки информации / А.Ф. Чернявский [и др.]. – Минск : Университетское, 1996. – 376 с.

2. Bajart, J.-C. An RNS montgomery modular multiplication algorithm / J.-C. Bajart, L.-S. Didier, P. Kornerup // IEEE Trans. Comput. – 1998. – Vol. 47, № 7. – P. 766–776.

3. Hiasat, A.A. New efficient structure for a modular multiplier for RNS / A.A. Hiasat // IEEE Trans. Comput. – 2000. – Vol. 49, № 2. – P. 170–174.

4. Alia, G. Fast modular exponentiation of large number with large exponents / G. Alia, E. Martinelli // J. Syst. Archit. – 2002. – Vol. 47, № 14–15. – P. 1079–1088.

5. Lee, K.-J. Systolic multiplier for Montgomery's algorithm / K.-J. Lee, K.-J. Yoo // Integration. – 2002. – Vol. 32, № 1–2. – P. 99–109.

6. RSA speedup with residue number system immune against hardware fault cryptanalysis / S.- M. Yen [et al.] // Lect. Notes Comput. Sci. – 2002. – Vol. 2288. – P. 297–413.

7. Hiasat, A.A. Semi-custo VLSI design an implementation of a new efficient RNS division algorithm / A.A. Hiasat, Z.H. Abdelati // Comput.1. – 1999. – Vol. 42, № 3. – P. 232–240.

8. Talahmeh, S. Arithmetic division in RNS using Galois field GF(p) / S. Talahmeh, P. Siy // Comput. Arc. Math. Appl. – 2000. – Vol. 39, № 5–6. – P. 227–238.

9. Posch, K.S. Modulo reduction in residue number system / K.S. Posch, R. Posch // IEEE Trans. on parallel and distributed syst. – 1995. – Vol. 6, № 5. – P. 449–454.

10. Schwemmlein, J. RNS-modulo reduction upon a restricted base value set and its applicability to RSA cryptography // J. Schwemmlein, K.S. Posch, R. Posch // Comput. and security. – 1998. – Vol. 17, № 7. – P. 637–650.

11. Cox-Rower architecture for fast parallel Montgomery multiplication / S. Kawamura [et al.] // Eurocrypt 2000, LNCS. – Berlin, 2000. – Vol. 1807. – P. 523–538.

12. Bajard, J.-C. A Full RNS Implementation of RSA / J.-C. Bajard, L. Imbert // IEEE Trans. Comp. – 2004. – Vol. 53, № 6. – P. 769–774.

13. Shenoy, A.P. Residue to Binary Conversion for RNS Arithmetic Using Only Modular Look-up Tables / A.P. Shenoy, R. Kumaresan // IEEE Trans. on Circuit and Systems. – 1988. – Vol. 35, № 9. – P. 1158–1162.

14. Shenoy, A.P. Fast Base Extension Using a Redundant Modulus in RNS / A.P. Shenoy, R. Kumaresan // IEEE Trans. Comput. – 1989. – Vol. 38, № 2. – P. 292–297.

15. Евдокимов, А.А. Метод и алгоритм масштабирования для многомашинных и мультипроцессорных систем модулярной обработки информации / А.А. Евдокимов, Н.А. Коляда, В.В. Ревинский // Весці НАН Беларусі. Сер. фіз.-тэхн. навук. – 2006. – № 1. – С. 104–111.

16. Коляда, А.А. Модулярные структуры конвейерной обработки цифровой информации / А.А. Коляда, И.Т. Пак. – Минск : Университетское, 1992. – 256 с.

17. Устройство для деления чисел в системе остаточных классов : а.с. №1287152 СССР / А.А. Коляда // Открытия. Изобретения. – 1987. – № 4.

18. Устройство для деления чисел : а.с. №1683013 СССР / В.Н. Ахременко, А.А. Коляда, М.Ю. Селянинов // Открытия. Изобретения. – 1991. – № 37.

19. Амербаев, В.М. Теоретические основы машинной арифметики / В.М. Амербаев. – Алма-Ата : Наука, 1976. – 324 с.

20. Устройство для деления чисел в модулярной системе счисления : а.с. №1756887 СССР / В.Н. Ахременко [и др.] // Открытия. Изобретения. – 1992. – № 31.

21. Banerji, D.K. A high-speed division method in residue arithmetic / D.K. Banerji, T.Y. Chueng, V. Ganesan // Proc. 5-th Simp. Comput. Arithmetic. – N.Y., 1981. – P. 158–164.

22. Харин, Ю.С. Компьютерный практикум по математическим методам защиты информации / Ю.С. Харин, С.В. Агиевич. – Минск : БГУ, 2001. – 190 с.

23. Математические и компьютерные основы криптологии / Ю.С. Харин [и др.]. – Минск : Новое знание, 2003. – 382 с.

24. Коляда, А.А. Общая технология вычисления интегральных характеристик модулярного кода / А.А. Коляда, А.Ф. Чернявский // Доклады НАН Беларуси. – 2008. – Т. 52, № 4. – С. 38–44.

25. Функциональные особенности и общие принципы реализации модулярной вычислительной технологии на диапазонах большой мощности / С.М. Завгороднев [и др.] // Электроника инфо. – 2008. – № 12. – С. 50–55.

Поступила 12.08.09

Институт прикладных физических
проблем им. А.Н. Севченко БГУ,
Минск, Курчатова, 7
e-mail: sh_helen@tut.by

A.A. Kolyada, N.A. Kolyada, V.V. Revinsky, A.F. Chernyavsky, H.V. Shabinskaya

MULTIPLICATION ON A LARGE MODULE IN THE MINIMALLY REDUNDANT MODULAR NOTATION USING SCALING OPERATIONS

A new method of multiplication on large simple modules in the minimally redundant modular notation is suggested. It is based on fast-converging recursive scheme of reduction to remainder – the Farm scheme of descent and high-speed algorithm for scaling of the table type. The method correctness is investigated and some results of estimation of its efficiency are reported.

The multiplicative algorithm which in comparison with known methods allows to reduce quantity of tables for the formation of the base integrated characteristics at 35–40 % and reducing time expenses at least for 1.6 times is synthesized.