УДК 004.222.3

В.С. Бабков (канд.техн.наук, доц.), **Е.В. Пехотин** (магистрант)
Донецкий национальный технический университет
Victor.Babkov@gmail.com

МЕТОДЫ ВЫПОЛНЕНИЯ ОПЕРАЦИЙ С СОХРАНЕНИЕМ ТОЧНОСТИ НАД МНОГОРАЗРЯДНЫМИ ЧИСЛАМИ В ЭФФЕКТИВНО-ПРЕДСТАВЛЯЮЩЕМ ФОРМАТЕ

В работе предложен подход к обработке многоразрядных чисел в формате с плавающей точкой с минимизацией потери точности. Результатом работы является теоретическое обоснование предложенных форматов представления данных, получение характеристик точности при выполнении базовых арифметических операций в предложенном формате.

плавающая точка, интервальные вычисления, точность, эффективно-представляющий формат, пакетная обработка

Введение

На сегодняшний день в математике существует множество методов решения задач в различных областях человеческой деятельности. Но, в отличии от теоретических изысканий, непосредственное применение этих методов в реальности затруднительно, ибо на практике можно использовать только числа, имеющие конечное представление, т.е. представленные в виде конечной цепочки цифр, в то время, как большинство этих методов работают с произвольными вещественными числами. Естественно, подавляющее большинство вещественных чисел невозможно выразить конечным представлением (и это легко доказать, ибо для любого конечного представления мы будем иметь возможность представить всего $N = az$ чисел, где a – основание с.с. представления, а z – число разрядов представления, а как известно, мощность множества вещественных чисел равна ∞ и естественно, что $N < \infty$). А т.к. данное ограничение касается как входных, так и выходных и промежуточных данных, то в области реальных вычислений встает задача как повышения точности расчетов, так и разработки устойчивых методов расчета (т.е. таких методов, в которых теоретически оценено или (что то же) контролируется влияние точности промежуточных вычислений на результат).

Задача выполнения вычислений с высокой точностью возникает в инженерных и научных расчетах в различных областях [1]: моделирование в физике, химии, астрономии и т.д. На практике наиболее часто возникающей проблемой является выбор между диапазоном представления величин и точностью вычислений (т.е. потерей точности и накоплением ошибок при выполнении вычислительных операций).

Многие современные математические пакеты, например, Mathematica, уже содержат встроенные средства для управления точностью представления результата и разрядностью при выполнении операций, что говорит об актуальности данного исследования.

Итак, суммируя вышесказанное, можно сказать о существовании на сегодняшний день проблемы соотнесения вычисленного приближенного результата с истинным абстрактным решением поставленной задачи.

Существует два основных подхода к решению данной проблемы:

- точечный или аппроксимационный подход – использовать вместо вещественных чисел их приближение рациональными, т.е. имеющими представление в виде конечной цепочки цифр. Общепринятым стандартом для представления чисел в формате с плавающей точкой и выполнения операций над ними является стандарт IEEE-754 [2];

- интервальный подход – представление вещественных чисел в виде конечно-представимого множества чисел, содержащих в себе искомое число, в форме достоверного интервала–пары рациональных чисел [A; B] (достоверный в данном случае означает, что истинный абстрактный результат гарантированно находится в интервале, локализуемом указанными границами).

А т.к. данное ограничение касается как входных, так и выходных и промежуточных данных, то в области реальных вычислений встает задача как повышения точности расчетов, так и разработки устойчивых методов расчета [1-5].

Данная работа является продолжением работы, опубликованной в [6].

Общее введение в теорию рациональных чисел

Итак, рациональное число есть действительное число в виде конечной цепочки цифр. В системе счисления с основанием *a* такое число A можно представить, как

$$A = \pm \sum_{i=-m}^{n} z_i a^i = \pm z_n a^n + z_{n-1} a^{n-1} + \ldots + z_0 a^0 + z_{-1} a^{-1} + \ldots + z_{-m} a^{-m} \qquad (1)$$

где z_i ◎ [0; a-1] – значение i-го разряда (цифра) числа A в с.с. с основанием *a*,

N = n+m+1 – количество разрядов, отведенное под рациональное число A в с.с. *a*.

Форма записи (1), в которой основание с.с. *a* опускается, значения разрядов z_i записываются справа налево, а позиция десятичной точки запоминается, называется представлением числа A в формате с фиксированной точкой:

$$A = \pm z_n z_{n-1} \ldots z_1 z_0 (.) z_{-1} z_{-2} \ldots z_{-m} \qquad (2)$$

Этот формат имеет серьезный недостаток – малый диапазон представляемых чисел для заданного N [7]. Для улучшения ситуации вводится коэффициент масштаба, на который умножается число, чтобы

получить его истинное значение. Числа такого вида называются числами с плавающей точкой и представляются в с.с. a в виде

$$A = \pm M \cdot a^p$$

где M – мантисса, запись значения числа в форме с фиксированной точкой,

a^p – коэффициент масштаба,

p – порядок числа.

Эффективно-представляющие форматы чисел с плавающей точкой

Теперь можно ввести понятие эффективно-представляющего формата – при данный значениях длин порядка $w = w_0$ и мантиссы $t = t_0$ (соответственно, общий размер числа $N = N_0 = 1 + t_0 + w_0$) формат является **эффективно-представляющим**, если он эффективно использует все w_0 битов порядка и t_0 битов мантиссы, т.е. имеет такое свойство: количество представимых форматом **различных** чисел равно общему числу двоичных чисел разрядностью N_0 (булеану от N_0).

У эффективно–представляющих форматов имеется исходящее из их определения преимущество перед другими: они представляют максимальное количество различных чисел, которые можно представить имеющейся разрядностью.

В [6, с. 14-15] была доказана **лемма 1**: все числа, представимые в формате 1.0 + ε, является различными или (что то же) любое число, представимое в формате 1.0 + ε, представимо в нем единственным образом. Доказательство строилось на том, что любое число можно разложить в вид $a = (1.0 + \varepsilon_a) \cdot 2^{p_a}$ единственным образом благодаря тому, что $0 \le \varepsilon_a < 1.0$, а p_a – целое. У этой леммы имеется важное следствие: алгоритм преобразования из постулата 1 преобразует число a в формат 1.0 + ε единственным образом.

Утверждение 1. Формат 1.0+ε является эффективно-представляющим.

Данное утверждение является заменой эквивалентному в [6, с. 15] – его доказательство дополнено анализом влияния денормализованных чисел.

Утверждение 2. Формат 0.0+ε не является эффективно-представляющим.

Доказательство приведено в [6, с. 15-16] и основывается на том, что либо мы благодаря необходимости задания всех битов мантиссы будем получать для одного числа несколько возможных представлений, либо у нас будут зафиксированы значения некоторых бит мантиссы, что приведет к уменьшению количества доступных для кодирования чисел комбинаций, а оба этих случая противоречат условию ЭП-формата.

Итак, в качестве базового формата представления чисел с плавающей точкой рекомендуется использовать эффективно-представляющий формат 1.0 + ε.

Связь между рациональными, интервальными и метрологическими числами

Теперь можно продемонстрировать взаимосвязь между рациональными, интервальными и метрологическими [8-9] числами через выведение оценок точности арифметических операций. Пускай:

$$x = \bar{x} \pm \Delta x = [a;b] \Rightarrow a = \bar{x} - \Delta x,\ b = \bar{x} + \Delta x$$
$$y = \bar{y} \pm \Delta y = [c;d] \Rightarrow c = \bar{y} - \Delta y,\ d = \bar{y} + \Delta y$$

В такой записи сразу видна связь между метрологическими и интервальными числами; связь же между рациональными и интервальными выводится из связи рациональных и метрологических, где в качестве точности числа ($\Delta x, \Delta y$) используется точность его представления (P_{Δ}^{p}, см. выше).

Исследуем операцию сложения:

$$x + y = [a;b] + [c;d]:$$
$$\left.\begin{array}{l} a \le b \Rightarrow a + c \le b + c,\ a + d \le b + d \\ c \le d \Rightarrow a + c \le a + d,\ b + c \le b + d \end{array}\right\} \Rightarrow a + c \le \left\{\begin{array}{l} b + c \\ a + d \end{array}\right\} \le b + d \Rightarrow$$
$$[a;b] + [c;d] = [a + c; b + d] \equiv \left[\left(\bar{x} - \Delta x\right) + \left(\bar{y} - \Delta y\right); \left(\bar{x} + \Delta x\right) + \left(\bar{y} + \Delta y\right)\right] =$$
$$\left[\left(\bar{x} + \bar{y}\right) - \left(\Delta x + \Delta y\right); \left(\bar{x} + \bar{y}\right) + \left(\Delta x + \Delta y\right)\right] = \left(\bar{x} + \bar{y}\right) \pm \left(\Delta x + \Delta y\right) \Rightarrow$$
$$x + y = z \pm \Delta z,\ \ z = \bar{x} + \bar{y},\ \Delta z = \Delta x + \Delta y$$

Итак, при осуществлении операции сложения для получения корректного результата необходимо сложить как значения чисел, так и их погрешности. Следовательно, при осуществлении сложения точность вычисления падает (соответственно, погрешность вычисления растет).

Теперь исследуем операцию вычитания:

$$x - y = [a;b] - [c;d]:$$
$$\left.\begin{array}{l} a \le b \Rightarrow \begin{array}{l} a - c \le b - c \Rightarrow c - a \ge c - b \\ a - d \le b - d \Rightarrow d - a \ge d - b \end{array} \\ c \le d \Rightarrow \begin{array}{l} c - a \le d - a \Rightarrow a - c \ge a - d \\ c - b \le d - b \Rightarrow b - c \ge b - d \end{array} \end{array}\right. \Rightarrow \begin{array}{l} a - d \le a - c \le b - c \\ a - d \le b - d \le b - c \end{array} \Rightarrow$$
$$\Rightarrow a - d \le \left\{\begin{array}{l} a - c \\ b - d \end{array}\right\} \le b - c$$
$$[a;b] - [c;d] = [a - d; b - c] \equiv \left[\left(\bar{x} - \Delta x\right) - \left(\bar{y} + \Delta y\right); \left(\bar{x} + \Delta x\right) - \left(\bar{y} - \Delta y\right)\right] =$$
$$\left[\left(\bar{x} - \bar{y}\right) - \left(\Delta x + \Delta y\right); \left(\bar{x} - \bar{y}\right) + \left(\Delta x + \Delta y\right)\right] = \left(\bar{x} - \bar{y}\right) \pm \left(\Delta x + \Delta y\right) \Rightarrow$$
$$x - y = z \pm \Delta z,\ \ z = \bar{x} - \bar{y},\ \Delta z = \Delta x + \Delta y$$

Итак, при осуществлении операции вычитания для получения корректного результата необходимо для значений чисел из уменьшаемого вычесть вычитаемого, а их погрешности сложить. Следовательно, при осуществлении вычитания точность вычисления падает (соответственно, погрешность вычисления растет), как и при осуществлении операции сложения.

Теперь исследуем операцию умножения:

$$
\begin{aligned}
&x \cdot y = [a;b] \cdot [c;d]: \\
&\left| \begin{array}{l}
(1) \\
a < 0, b < 0, c < 0, d < 0: \\
\left\{ \begin{array}{l} a \le b \Rightarrow \begin{array}{l} a \cdot c \ge b \cdot c \\ a \cdot d \ge b \cdot d \end{array} \\ c \le d \Rightarrow \begin{array}{l} c \cdot a \ge d \cdot a \\ c \cdot b \ge d \cdot b \end{array} \end{array} \right. \Rightarrow b \cdot d \le \left\{ \begin{array}{l} a \cdot d \\ b \cdot c \end{array} \right\} \le a \cdot c \Rightarrow [a;b] \cdot [c;d] = [b \cdot d; a \cdot c] \equiv \\
\left[\left(\overline{x} + \Delta x \right) \cdot \left(\overline{y} + \Delta y \right), \left(\overline{x} - \Delta x \right) \cdot \left(\overline{y} - \Delta y \right) \right] = \\
\left[\overline{x}\,\overline{y} + \overline{y}\Delta x + \overline{x}\Delta y + \Delta x \Delta y; \overline{x}\,\overline{y} - \overline{x}\Delta y - \overline{y}\Delta x + \Delta x \Delta y \right] = \\
= \left[\overline{x}\,\overline{y} + \left(\overline{x}\Delta y + \overline{y}\Delta x \right) + \Delta x \Delta y; \overline{x}\,\overline{y} - \left(\overline{x}\Delta y + \overline{y}\Delta x \right) + \Delta x \Delta y \right] = \\
= \left(\overline{x}\,\overline{y} + \Delta x \Delta y \right) \pm - \left(\overline{x}\Delta y + \overline{y}\Delta x \right) \Rightarrow \\
z = \left(\overline{x}\,\overline{y} + \Delta x \Delta y \right), \Delta z = - \left(\overline{x}\Delta y + \overline{y}\Delta x \right)
\end{array} \right.
\end{aligned}
$$

Аналогичные выводы можно сделать для всех сочетаний входных параметров.

Собирая все вместе результаты вывода, получаем:

$$
\begin{aligned}
&[1] \left\{ \begin{array}{l} (1): a < 0, b < 0, c < 0, d < 0 \Rightarrow x \cdot y = \left(\overline{x}\,\overline{y} + \Delta x \Delta y \right) \pm - \left(\overline{x}\Delta y + \overline{y}\Delta x \right) \\ (9): a > 0, b > 0, c > 0, d > 0 \Rightarrow x \cdot y = \left(\overline{x}\,\overline{y} + \Delta x \Delta y \right) \pm \left(\overline{x}\Delta y + \overline{y}\Delta x \right) \end{array} \right. \Rightarrow \\
&\Rightarrow \left(\overline{x}\,\overline{y} + \Delta x \Delta y \right) \pm \left| \overline{x}\Delta y + \overline{y}\Delta x \right| \\
&[2] \left\{ \begin{array}{l} (4): a < 0, b < 0, c > 0, d > 0 \Rightarrow x \cdot y = \left(\overline{x}\,\overline{y} - \Delta x \Delta y \right) \pm - \left(\overline{x}\Delta y - \overline{y}\Delta x \right) \\ (6): a > 0, b > 0, c < 0, d < 0 \Rightarrow x \cdot y = \left(\overline{x}\,\overline{y} - \Delta x \Delta y \right) \pm \left(\overline{x}\Delta y - \overline{y}\Delta x \right) \end{array} \right. \Rightarrow \\
&\Rightarrow \left(\overline{x}\,\overline{y} - \Delta x \Delta y \right) \pm \left| \overline{x}\Delta y - \overline{y}\Delta x \right|
\end{aligned}
$$

Операция деления есть по определению операция умножения на обратное и может быть сведена к нему, таким образом, связь между рациональными, метрологическими и интервальными числами для операции деления выводится аналогично.

Математика сохраняющей арифметики чисел с плавающей точкой

Теперь можно рассмотреть, как математически осуществляются арифметические операции над числами форматов $1.0+\varepsilon$ (эффективно-представляющем формате).

Пусть имеются числа $a=(1.0+\varepsilon_a)\cdot 2^{p_a}$ в формате <w_a,t_a> ($N_a = 1+w_a+t_a$) и $b=(1.0+\varepsilon_b)\cdot 2^{p_b}$ в формате <w_b,t_b> ($N_b = 1+w_b+t_b$), и a > b. Из [6] очевидно, что $\Delta a = P_\Delta^{p_a} = G_\Delta^{p_a}/2 = G_\Delta^M \cdot 2^{p_a}/2 = 2^{-t}\cdot 2^{p_a-1} = 2^{p_a-(t+1)}$, $\Delta b = P_\Delta^{p_b} = 2^{p_b-(t+1)}$.

Из a > b:

$$a > b \overset{def}{\Leftrightarrow} (1.0+\varepsilon_a)\cdot 2^{p_a} > (1.0+\varepsilon_b)\cdot 2^{p_b} \Leftrightarrow \left\{\begin{array}{l} p_a > p_b \\ p_a = p_b, \varepsilon_a > \varepsilon_b \end{array}\right\} \Rightarrow p_a \geq p_b \Rightarrow$$

$$2^{p_a} \geq 2^{p_b} \Rightarrow 2^{p_a-(t+1)} \geq 2^{p_b-(t+1)} \Rightarrow \Delta a \geq \Delta b \Rightarrow \Delta a = \max[\Delta a, \Delta b]$$

Рассмотрим подробно для операции сложения a + b:

$$z = a + b = (1.0+\varepsilon_a)\cdot 2^{p_a} + (1.0+\varepsilon_b)\cdot 2^{p_b} = \left\{\begin{array}{l} a > b \Rightarrow p_a \geq p_b \\ p_b = p_a - p_\varphi, \\ p_\varphi = p_a - p_b \geq 0 \end{array}\right\} =$$

$$(1.0+\varepsilon_a)\cdot 2^{p_a} + (1.0+\varepsilon_b)\cdot 2^{p_a-p_\varphi} == \left((1.0+\varepsilon_a) + \frac{(1.0+\varepsilon_b)}{2^{p_\varphi}}\right)\cdot 2^{p_a} =$$

$$\left((1.0+\varepsilon_a) + (1.0+\varepsilon_b)\cdot 2^{-p_\varphi}\right)\cdot 2^{p_a};$$

$$\left\{\begin{array}{l} 0 \leq \varepsilon_a < 1 \\ 0 \leq \varepsilon_b < 1 \end{array}\right\} \Rightarrow \left\{\begin{array}{l} 1 \leq 1.0+\varepsilon_a < 2 \\ 1 \leq 1.0+\varepsilon_b < 2 \end{array}\right\} \Rightarrow \left\{\begin{array}{l} 2^{p_a} \leq (1.0+\varepsilon_a)\cdot 2^{p_a} < 2^{p_a+1} \\ 2^{p_b} \leq (1.0+\varepsilon_a)\cdot 2^{p_b} < 2^{p_b+1} \end{array}\right\} \Rightarrow$$

$$2^{p_a} + 2^{p_b} \leq \left((1.0+\varepsilon_a) + (1.0+\varepsilon_b)\cdot 2^{-p_\varphi}\right)\cdot 2^{p_a} < 2^{p_a+1} + 2^{p_b+1} \Rightarrow$$

$$\frac{2^{p_a} + 2^{p_b}}{2^{p_a}} \leq (1.0+\varepsilon_a) + (1.0+\varepsilon_b)\cdot 2^{-p_\varphi} < \frac{2^{p_a+1} + 2^{p_b+1}}{2^{p_a}} \Rightarrow$$

$$1 + 2^{p_b-p_a} \leq (1.0+\varepsilon_a) + (1.0+\varepsilon_b)\cdot 2^{-p_\varphi} < 2 + 2^{p_b+1-p_a} \Rightarrow$$

$$1 + 2^{-p_\varphi} \leq (1.0+\varepsilon_a) + (1.0+\varepsilon_b)\cdot 2^{-p_\varphi} < 2 + 2^{-p_\varphi+1}, \left\{\begin{array}{l} p_\varphi \geq 0 \Rightarrow \\ 0 \leq 2^{-p_\varphi} \leq 1 \\ 0 \leq 2^{-p_\varphi+1} \leq 2 \end{array}\right\} \Rightarrow$$

$$1 \leq (1.0+\varepsilon_a) + (1.0+\varepsilon_b)\cdot 2^{-p_\varphi} < 4 \Rightarrow 0 \leq \varepsilon_a + (1.0+\varepsilon_b)\cdot 2^{-p_\varphi} < 3 \Rightarrow$$

$$z = (1.0+\varepsilon_z)\cdot 2^{p_z}, \{\varepsilon_z, p_z\} = \left\{\begin{array}{l} \Sigma = \varepsilon_a + (1.0+\varepsilon_b)\cdot 2^{p_b-p_a} \\ \{\Sigma, p_a\}, \Sigma < 1 \\ \left\{\frac{\Sigma - 1.0}{2}, p_a + 1\right\}, 1 \leq \Sigma < 3 \\ \Uparrow 2\cdot\frac{(1.0+\Sigma)}{2}\cdot 2^{p_a} = \left(1.0 + \frac{\Sigma}{2} - 0.5\right)\cdot 2^{p_a+1} \end{array}\right.$$

Итак, сложение многоразрядных чисел выполняется путем комбинации сдвигов и сложений над их мантиссами (ибо для двоичного представление числа умножение на любую двойку в степени эквивалентно сдвигу вправо или влево). Теперь можно указать как границы необходимой разрядности для представления, так и точную разрядность результата операции:

$$p_z = \begin{cases} p_a \\ p_a + 1 \end{cases} \Rightarrow \left\{ \begin{array}{l} p_a \le p_z \le p_a + 1 \\ p_{\min} \le p_a \le p_{\max} \\ p_{\max} - p_{\min} = 2^{w_a} - 1 \\ bias = -p_{\min} \end{array} \right\} \Rightarrow \left\{ \begin{array}{l} p_{\min} - p_{\min} \le p_a - p_{\min} \le p_{\max} - p_{\min} \Rightarrow \\ \left\{ \begin{array}{l} 0 \le p_a + bias \le 2^{w_a} - 1 \\ 1 \le p_a + 1 + bias \le 2^{w_a} \\ p_a + bias \le p_z + bias \le p_a + 1 + bias \end{array} \right. \Rightarrow \\ 0 \le q_z \le 2^{w_a} \end{array} \right\}$$

$$w_z = \lceil \log_2 q_z \rceil \equiv \omega : 2^{\omega} \ge q_z \Rightarrow w_z = \left\{ \begin{array}{l} w_a, p_z = p_a \Leftarrow \Sigma < 1 \\ w_a + 1, \left\{ \begin{array}{l} p_a = p_{\max} \Leftrightarrow q_a = 2^{w_a} - 1 \Leftrightarrow \\ q_a + 1 = 2^{w_a} \Leftrightarrow \\ q_a \,\&\, (q_a + 1) = 0 \end{array} \right\} \Leftarrow \Sigma > 1 \\ w_a, p_a < p_{\max} \Leftrightarrow q_a \,\&\, (q_a + 1) \ne 0 \end{array} \right\}$$

$$\varepsilon_z = \begin{cases} \Sigma, \Sigma < 1 \\ \dfrac{\Sigma - 1.0}{2}, \Sigma > 1 \end{cases} : \begin{cases} \varepsilon_a + (1.0 + \varepsilon_b) \cdot 2^{-p_\varphi}, \varepsilon_a + (1.0 + \varepsilon_b) \cdot 2^{-p_\varphi} < 1 \\ \dfrac{\varepsilon_a + (1.0 + \varepsilon_b) \cdot 2^{-p_\varphi} - 1.0}{2}, \varepsilon_a + (1.0 + \varepsilon_b) \cdot 2^{-p_\varphi} \ge 1 \end{cases}$$

$$\Sigma < 1 : \varepsilon_z = \varepsilon_a + (1.0 + \varepsilon_b) \cdot 2^{-p_\varphi} \overset{(1),(3)}{=} \sum_{i=-1}^{-L[\varepsilon_a]} \varepsilon_a^{(i)} \cdot 2^i + \left(1.0 + \sum_{i=-1}^{-L[\varepsilon_b]} \varepsilon_b^{(i)} \cdot 2^i \right) \cdot 2^{p_b - p_a} =$$

$$\sum_{i=-1}^{-L[\varepsilon_a]} \varepsilon_a^{(i)} \cdot 2^i + 2^{p_b - p_a} + \sum_{i=-1}^{-L[\varepsilon_b]} \varepsilon_b^{(i)} \cdot 2^{i + p_b - p_a} = \left\{ \begin{array}{l} \overline{\varepsilon_b^{(i)}} = \left\{ \begin{array}{l} \\ \varepsilon_b^{(i+)} \end{array} \right\} \\ \end{array} \right.$$

$$, \begin{cases} L[\varepsilon_a] \le t_a \\ L[\varepsilon_b] \le t_b \end{cases}$$

Аналогичным образом вывод осуществляется для остальных операций (вычитание, умножение, деление).

Выводы

В результате исследования осуществлен вывод формул, связующих рациональное, метрологическое и интервальное представление чисел в формате с плавающей запятой. Полученные формулы позволяют указать как границы необходимой разрядности для представления, так и точную разрядность результата операции. Использование понятия «эффективно-представляющий формат» и критериев, приведенных в [6], позволяет определить оптимальные параметры формата, которые минимизируют потерю точности.

Также осуществлен вывод формул оценки точности базовых арифметических операций (сложение, вычитание, умножение, деление) при использовании эффективно-представляющего формата.

Предложенный математический аппарат может быть использован при построении программных библиотек для расчетов высокой точности, разработка таких библиотек является дальнейшим направлением работы.

Список литературы

1. Wilkinson J. H. Modern error analysis // SIAM Rev. — 1971. — Vol. 13, № 4. — P. 548–568.
2. Moore R. E. Interval analysis. — Englewood Cliffs; Prentice Hall, 1966. — 145 p.
3. Moore R. E. Methods and applications of interval analysis. — Philadelphia; SIAM, 1979. 190 p.
4. Alefeld G., Herzberger J. Introduction to interval computations. — New York etc.; Academic Press, 1983. – XVIII, 333 p.; Рус. перев.; Алефельд Г., Херцбергер Ю. Введение в интервальные вычисления: Пер. с англ. — М.: Мир, 1987. — 356 с.
5. Nickel K. Can we trust the results of our computing? // Mathematics for Computer Science; Proc. Symposium held in Paris, March 16–18, 1982. — S. l.; Association francaise pour la cybernetique et technique (AFCET), 1982. — P. 167–175
6. Бабков В.С. Оценка параметров многоразрядных чисел с плавающей точкой для выполнения операций высокой точности / В.С.Бабков, Е.В. Пехотин // Научные труды Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника. – 2010. – Вып. 12 (165). – С. 12-17.
7. Edmond / HI-TECH. FPU посвящается (часть 1) [электронный ресурс]: wasm.ru/print.php?article=edfpu01

a. Юровицкий В. Третья вычислительная революция / В. Юровицкий - [электронный ресурс]: http://www.yur.ru/science/computer/index.htm
b. Юровицкий В. Метрология для всех Концепция развития метрологии в XXI веке / В. Юровицкий - [электронный ресурс]: http://www.twirpx.com/file/5996/

Надійшла до редакції: 22.10.2010 Рецензент:, канд.техн.наук, проф. Аноприенко А.Я.

В.С. Бабков, Є.В. Пехотін
Донецький національний технічний університет

Методи виконання операцій із збереженням точності над багаторозрядними числами у ефективно-поданому форматі. У роботі запропоновано підхід до обробки багато розрядних чисел у форматі з плаваючою точкою з мінімізацією втрати точності. Результатом роботи є теоретичне обгрунтування запропонованих форматів подання даних, отримання характеристик точності при виконанні базових арифметичних операцій у запропонованому форматі.
плаваюча точка, інтервальні обчислення, точність, ефективно-поданий формат, пакетна обробка

V.S. Babkov, E.V. Pekhotin
Donetsk National Technical University

An Approach to Operations while Preserving the Accuracy over Multi-Bit Numbers in Effectively-Representing Format. In this paper we propose an approach to handling multidigit numbers in a floating point format to minimize the loss of accuracy. As a result the proposed data formats are theoretically justified, the characteristics of precision when performing basic arithmetic operations in the proposed formats have been obtained.
floating point, interval calculations, accuracy, efficiency-representing format, batch processing