УДК 004.9

Особенности постбинарного кодирования на примере интервального представления результатов вычислений по формуле Бэйли-Боруэйна-Плаффа

А.Я. Аноприенко, С.В. Иваница
Донецкий национальный технический университет
anoprien@cs.dgtu.donetsk.ua

Abstract

Anoprienko A., Ivanitsa S. Features of Postbinary Coding on the Example of Interval Representation of the Calculation Results Obtained Using Bailey-Borwein-Plouffe's Formula. *We have considered the possibilities of expanding the standard arithmetic of floating point numbers, in particular the transition to machine interval arithmetic. It is possible to represent a numerical interval by means of tetracodes. One of the methods of mutual transition from the decimal representation of interval boundaries to the tetracode one is shown with the example of calculating* p *using Bailey-Borwein-Plouffe's formula.*

Введение

Современные ЭВМ практически полностью базируются на двоичной логике и арифметике, и до недавнего времени обеспечивали практически все потребности компьютерных вычислений. Однако в конце прошлого века произошли качественные изменения, как в развитии логических основ, так и в области компьютерных технологий, которые обусловили актуальность соответствующих изменений в кодо-логическом базисе современных компьютерных технологий [1].

Одним из вариантов усовершенствования традиционной двоичной логики является переход к многозначной логике, т. е. двумерное логическое пространство может быть продуктивно расширено до трехмерного путем введения третьего измерения, соответствующего возможной недостоверности и «вариабельности» логических значений двумерного пространства. В качестве простейших частных случаев использования вариабельности могут рассматриваться тетралогика, позволяющая реализовать свойство адаптивности в рамках подходов, характерных для традиционной бинарной логики.

В рамках данной статьи рассматривается идея представления данных с помощью тетракодов, в частности при представлении вещественных чисел в виде интервалов (интервальных чисел). Идея такого представления интервальных чисел заключается в кодировании границ интервала одним тетракодовым числом, что способствует, во-первых, возможности представления этих границ в памяти ЭВМ не двумя, а одним машинным числом, и, во-вторых, созданию предпосылок для разработки концептуально нового математического аппарата, работающего с «тетракодовыми интервалами».

Особенности компьютерных вычислений с возможностью расширения стандартной арифметики чисел с плавающей точкой

В настоящее время общепринятым средством для выполнения научных и инженерных вычислений на компьютерах является арифметика чисел, представленных в формате с плавающей точкой. Данный формат разработан ассоциацией IEEE и используется для представления действительных чисел в двоичном коде [2]. На большинстве компьютеров каждая отдельная операция с плавающей точкой обеспечивает результат максимальной точности в том смысле, что получаемый округленный результат отличается от точного вещественного значения не более, чем на единицу последнего разряда мантиссы. Однако результат двух или нескольких последовательных операций может не содержать уже ни одной верной цифры. Поскольку современные суперкомпьютеры преодолели «петафлопсный» рубеж [3], то достоверность вычисляемых результатов становится предметом особого внимания.

Стремительное развитие вычислительной техники явилось стимулом к возрастанию масштабов как самих численных задач, так и требований к надежности получаемых решений. В последние годы для многих задач и приложений численного анализа были разработаны методы решения, обеспечивающие необходимую арифметическую и вычислительную надежность, что дает возможность, во-первых, получения высокой точности результатов вычислений, и, во-вторых, автоматической проверки их корректности. Поиск возможностей, направленных на повышение надежности вычислений, положил начало развитию обширной области исследований – интервальных вычислений (см., например, библиографии в [4, 5]).

Состояние современной компьютерной арифметики можно существенно улучшить, т. е. сделать ее более «интеллектуальной», путем новых средств и методов аппаратно-программной поддержки. Например, в работе [6, с.9-10] высказана возможность создания в арифметическом устройстве (или моделирования с помощью программных средств) специального регистра с фиксированной точкой, покрывающего своей разрядностью весь диапазон чисел с плавающей точкой. Причем возникающая при этом потеря скорости вычислений вполне компенсируется существенным возрастанием их надежности. Однако для обеспечения автоматического контроля погрешностей в существующих компьютерных архитектурах, машинную арифметику необходимо, как минимум, обеспечить возможностью направленного округления результата, т. е. округления до ближайшего машинного числа с недостатком или избытком. Данная концепция представления действительных чисел, а также арифметические операции с составленными из них векторами и матрицами позволяют построить машинную интервальную арифметику [7], в которой интервалы представляются в виде компьютерных континуальных объектов и открывают для численного анализа совершенно новую перспективу. В памяти ЭВМ интервал записывается парой чисел в формате с плавающей точкой, определяемых как границы этого интервала. Такой интервал фиксирует все множество вещественных чисел, заключенных между двумя хранимыми в ЭВМ машинными числами. Арифметические операции с такими парами чисел (т. е. с границами интервалов-операндов) являются составляющими арифметических операций над интервалами, результатом которых также является пара машинных чисел, представляющая границы результирующего интервала.

Так, подмножество X множества всех действительных (вещественных) чисел $\mathbb{R}$, такое что

$$X = [x_1; x_2] = \{r \mid x_1 \le r \le x_2,\ x_1 \in \mathbb{R},\ x_2 \in \mathbb{R}\},$$

называют замкнутым вещественным интервалом (или просто интервалом), причем каждый такой интервал является элементом множества всех замкнутых вещественных интервалов $I(\mathbb{R})$. Однако всякое вещественное число a из $\mathbb{R}$ также является элементом $I(\mathbb{R})$, поскольку может быть представлено в виде интервала $[a; a]$, который называют точечным.

Интервальная математика предполагает выполнение унарных и бинарных операций над интервалами. Если $\bullet \in \{+, -, \cdot, :\}$ – бинарная операция на множестве $\mathbb{R}$ и $X, Y \in I(\mathbb{R})$, то

$$X \bullet Y = \{z = x \bullet y \mid x \in X, y \in Y\}$$

определяет бинарную операцию на $I(\mathbb{R})$. Если $s(x)$ – непрерывная унарная операция на $\mathbb{R}$, то

$$s(X) = [\min_{x \in X} s(x);\ \max_{x \in X} s(x)]$$

определяет соответствующую ей бинарную операцию на $I(\mathbb{R})$.

При вычислении нижней (левой) границы интервала-результата часто прибегают к округлению с недостатком, а при вычислении верхней (правой) границы – с избытком. Таким образом, в результате выполнения интервальных операций получается результирующий интервал, который гарантированно содержит все результаты применения данной арифметической операции к любым парам чисел, взятых из любого интервала-операнда. Однако стоит заметить, что интервальное вычисление арифметического выражения требует примерно вдвое большего числа операций по сравнению с вычислением такого же выражения в стандартной арифметике с плавающей точкой. Очевидно, что границы возможных значений результирующего интервала и его ширина интервала является естественной мерой неопределенности (неоднозначности).

Интервальная арифметика является основным инструментом интервального анализа, который в качестве математической дисциплины изучает задачи с интервальными неопределенностями в данных, а также методы их решения. При этом важно иметь в виду, что в данном контексте термин «неопределенность» означает не полное незнание (например, невозможность или крайняя затруднительность проведения исследования), а состояние частичного знания, предполагающего наличие какой-либо информации об исследуемой величине. Простейшей и наиболее распространенной ситуацией описания не известной точно величины является задание множества ее возможных значений [8]. Например, величина φ, представляющая неопределенность значений на числовой прямой между 1 и 5 может быть задана как $\varphi \in \{1, 2, 3, 4, 5\}$ или $1 \le \varphi \le 5$ или, в более обобщенном виде, как $\varphi \in \mathbb{R}$.

Потребность такого представления неопределенности возникает во множестве самых разнообразных ситуаций, в частности при представлении в машинных кодах бесконечных десятичных дробей. Например, число π – бесконечную десятичную дробь можно представить как некоторое приближение вещественного числа, имеющее конечное число десятичных или двоичных знаков: $\pi \approx 3{,}14159$. Однако при оперировании таким приближением числа π уже в начале вычислений допускается неизбежная ошибка, поскольку не известна информация о том, каким именно приближением, с недостатком или с избытком, является указываемое значение. Очевидно, что при более корректном представлении данных нужно явным образом указывать границу этой ошибки, например, путем уточнения того, что ошибка представления не превосходит половины единицы последнего разряда. Существует более предпочтительный способ указания ошибки представления, который состоит в том, чтобы предоставить наиболее узкие точечно-представимые границы (нижнюю и верхнюю) для задания какой-либо неоднозначной величины. Так, для числа π:

$$\pi \in [3{,}14159;\ 3.14160].$$

Частным случаем ошибок представления, имеющим инженерный характер, являются ошибки перевода из одной системы счисления в другую. Многие конечные десятичные дроби не имеют точного конечного представления среди двоичных чисел, с которыми оперируют современные компьютерные системы. Но при введении подобных чисел в ЭВМ, они заменяются некоторым конечным рядом по степеням двойки, что, как следствие, вносит ошибку в машинное представление числа.

Перспективы компьютерной реализации постбинарных интервальных вычислений

Подход к решению различных типов задач, при котором использование машинной интервальной арифметики уменьшает (или вовсе устраняет) проблемы соотнесения вычисленного приближенного результата с истинным (абстрактным) решением, получил название «интервальный», поскольку интервал, представляемый парой рациональных чисел-границ, является простейшим видом конечно-представимого множества, локализующего простейший абстрактный объект – вещественное число. Основная идея интервального подхода состоит в том, что вещественное число представляется в памяти вычислительной машины не одним, а двумя машинными числами – нижней и верхней оценкой, образующих интервальное число. Таким образом, в рамках интервального подхода исходные данные и промежуточные результаты представляются граничными значениями, над которыми и производятся все операции. При этом сами операции (прежде всего арифметические) определяются таким образом, чтобы результат соответствующей операции обязательно располагался внутри вычисляемых границ. Это, в конечном счете, приводит к возможности автоматически учитывать погрешности в задании исходных чисел и погрешности, вызываемые машинным округлением.

Исторически сложившийся переход от бинарного (дикодового) этапа в развитии компьютерных технологий к следующему (гиперкодовому) этапу позволил осуществить реализацию постбинарных интервальных вычислений, в которых все цифровые комбинации задаются с помощью гиперкодов, а реализация интервальных операций и алгоритмов – средствами гиперлогики. Причем гиперлогикой названы логики третьего и более высоких порядков, число вариантов Q_{K} которых для каждой из логик К-го порядка определяется количеством К сочетаний из n различных значений, заданных в логическом пространстве [9, с.27-32]:

$$Q_{\text{K}} = \frac{n!}{\text{K}!(n-\text{K})!}.$$

Аналогично определению гиперлогики, гиперкодами названы все системы кодирования третьего и более высоких порядков.

В интервальных вычислениях переход к тетралогике (т. е. логике четвертого порядка) прежде всего обусловлен возможностью построения различных вариантов логических значений, включающих кроме классических 1 («истина») и 0 («ложь») также и различных парных комбинаций, в частности, таких как А («неопределенность», «неоднозначность») и М («множественность», «многозначность»). Кроме того, система кодирования количественной информации, построенная на тетралогике обладает по сравнению с традиционными рядом качественных преимуществ [10].

Рассмотрим, например, процесс перехода от десятичного интервала к тетракодовому на примере вычисления иррационального числа π с помощью формулы Бэйли-Боруэйна-Плаффа [11]:

$$\pi = f(n) = \sum_{k=0}^{n} \frac{1}{16^k} \cdot \left(\frac{4}{8k+1} - \frac{2}{8k+4} - \frac{1}{8k+5} - \frac{1}{8k+6} \right);$$

Десятичный интервал $X = [\overline{x};\ \underline{x}]$ ($X \in I(\mathbb{R})$, $\overline{x} \in \mathbb{R}$, $\underline{x} \in \mathbb{R}$) составим из чисел, полученных на 1-м и 100-м шагах вычислений:

$$\overline{x} = f(1) = 3{,}14142;\quad \underline{x} = f(100) = 3{,}14159.$$

Поскольку полученные значения различаются, начиная с одной десятитысячной, то при использовании этого интервала в процессе вычислений, позволяющих производить автоматический учет всех видов погрешностей, гарантируется точность четырех первых цифр самого результата, т. е. в данном случае, числа π.

Данные числа, представленные в двоичном формате (для наглядности использован 32-битный двоичный формат для представления дробной части чисел), позволяют выразить данный интервал одним тетракодом $T \in \{t \mid t \in \mathbb{H}\}$, где $\mathbb{H} = \{0, 1, A, M\}$:

$$3{,}14142_{10} =$$
$$= 11.001001000011010000001100111100011_2,$$
$$3{,}14159_{10} =$$
$$= 11.001001000011111100111111000000011_2,$$

откуда $T =$

$$=11.001001000011MMMMMMAAAAAAAAAAAAAA.$$

Количество разрядов «многозначности» M обусловлено обеспечением необходимой точности границ интервала, такой, чтобы гарантировать «захват» исходных чисел. В частности, k-я позиция последнего разряда «многозначности» дробной части тетракодового числа может быть определена по формуле

$$k = \left]\frac{\ln(10^n)}{\ln(2)}\right[+ 1,$$

где n – количество цифр дробной части исходного числа в его десятичном представлении. Для двух исходных чисел с неравным количеством цифр дробной части n_1 и n_2, результирующее значение n выбирается из условия $n = \max(n_1, n_2)$.

Так, для нашего случая, $n = 5$ (оба десятичных числа имеют 5 цифр после запятой). Следовательно, $k = 18$, т. е. в тетракодовом числе разряды дробной части, начиная с 19-го, являются несущественными и принимают значение неопределенности A.

Обратный же переход к десятичным числам достигается абсолютной минимизацией/максимизацией M для получения левой/правой (нижней/верхней) границ интервала.

Получим численные значения границ интервала, заданных тетракодом T, причем рассмотрим его наихудшее значение с позиции неопределенности A. Очевидно, что для нижней границы интервала наихудшим будет тот случай, когда все разряды A примут значение 1; для верхней границы интервала наихудшим значением является то, в котором все разряды A примут значение 0:

$$t_{\min} =$$
$$=11.00100100001100000011111111111111;$$
$$t_{\max} =$$
$$=11.00100100001111111100000000000000.$$

Полученные десятичные значения

$$d1 = (t_{\min})_{10} = 3{,}14136|12363\ldots \text{ и}$$
$$d2 = (t_{\max})_{10} = 3{,}14159|77478\ldots$$

фиксируют все значения функции $f(x)$. В этом легко убедиться, поскольку истинное значение $\pi = 3{,}14159|26536\ldots$.

На рис.1 показано поведение функции $f(x)$ на первых 10 шагах вычисления, а также ее «вхождение» в полученный интервал с границами *d1* и *d2*.

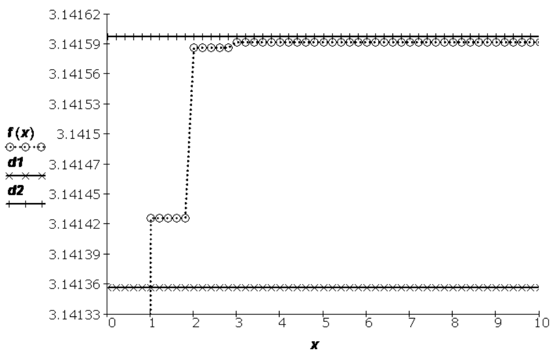

Рисунок 1 – Вычисление числа π с помощью формулы Бэйли-Боруэйна-Плаффа

Выводы

Переход от традиционных бинарных средств и методов компьютерной математики к интервальным вычислениям открывает существенно новые возможности для развития компьютинга и позволит контролировать точность и достоверность получаемых результатов на всех этапах вычислений.

Поскольку интервальный анализ давно вышел за рамки чисто теоретического исследования, и достаточно широко применяется при решении различных численных задач, то использование средств постбинарного компьютинга для реализации методов машинной интервальной арифметики безусловно откроет новые возможности «вычислительного интеллекта» и еще на шаг приблизит его к возможностям человеческого мышления.

Таким образом, представление границ интервалов в виде тетракодов дает возможность гибкого задания практически всех наборов числовых значений, включенных в этот интервал. Хотя по сравнению с обычным бинарным кодом, количество *n* бит, требуемых для кодирования тетракодов увеличивается в 2^n раз, кодирование числового интервала одним тетракодовым словом позволяет «обратиться» ко всем числам числовой оси, лежащих между границами интервала. Повышение степени информативности получаемых за счет этого кодов вполне оправдывает увеличение затрат на кодирование.

Литература

1. Аноприенко А.Я. Эволюция алгоритмического базиса вычислительного моделирования и сложность реального мира /А.Я. Аноприенко // Научные труды Донецкого национального технического университета. Серия: Проблемы моделирования и автоматизации проектирования динамических систем (МАП-2002). – 2002. – Вып. 52. – С. 6-9.

2. IEEE Standard for Floating-Point Arithmetic (IEEE 754) http://en.wikipedia.org/wiki/IEEE_754-2008

3. FLOPS / Материал из Википедии, http://ru.wikipedia.org/wiki/FLOPS.

4. Neumaier A. Interval Methods for Systems of Equations. Cambridge University Press, Cambridge, 1990.

5. Hansen, E., Walster G. Global Optimization Using Interval Analysis. Marcel Dekker, New York, 2004.

6. Hammer R., Hocks M., Kulisch U., Ratz D. Numerical toolbox for verified computing I: Basic numerical problems. – Berlin-Heidelberg: Springer, 1993.

7. Введение и интервальные вычисления / Г. Алефельд и др. – М.:Мир, 1987. — 360 с.

8. Шарый С.П. Конечномерный интервальный анализ. – Режим доступа: http://www.nsc.ru/interval.

9. Аноприенко А. Я. Археомоделирование: модели и инструменты докомпьютерной эпохи / А. Я. Аноприенко. – Донецк: УНИТЕХ, 2007. — 318 с.

10. Аноприенко А. Я. Тетралогика и тетракоды / А. Я. Аноприенко // Сборник трудов факультета вычислительной техники и информатики. – 1996. – Вып.1.

11. Bailey David H., Borwein Peter B., Plouffe Simon On the Rapid Computation of Various Polylogarithmic Constants // Mathematics of Computation. — 1997. — В. 218.Т. 66. — p. 903-913.

Поступила в редакцию 28.03.2010