УДК 004.4(043.2)

Гамаюн В. П., д-р техн. наук

МОДЕЛИРОВАНИЕ МНОГОРАЗРЯДНЫХ КОМПЬЮТЕРНЫХ СРЕДСТВ

Институт компьютерных технологий Национального авиационного университета

Предлагаются методы, арифметико-алгоритмический базис построения моделей для исследования многоразрядных компьютерных средств. Альтернативность подхода заключается в применении структур данных и соответствующих методов, алгоритмов обработки на основе разрядно-логарифмического кодирования информации.

Введение и постановка задачи. Точность преобразования информации (данных) при использовании компьютерных средств является инвариантной характеристикой последних. Числа, которые обрабатываются на компьютере за одну регистровую операцию, один операционный цикл, имеют ограниченную длину. Однако для большого числа современных задач, в основном связанных с защитой информации в компьютерных сетях, решением задач, чувствительных к ошибкам вычислений и др. существует проблема обработки многоразрядных чисел (операндов). Например, в криптографических системах защиты информации производится преобразование операндов с разрядной сеткой в несколько тысяч бит. Такие задачи требуют создания как новых методов компьютерной обработки и компьютерных средств для их реализации, так и соответствующих методов моделирования.

Основы разрядно-логарифмического кодирования.

В основе высокоточных вычислений для чисел большого диапазона применяются методы с применением специальных арифметик и алгебр, на основе которых создаются соответствующие компьютерные математики. В настоящее время известны следующие целочисленные арифметики, применяемые для обработки многоразрядных чисел [1, 2, 3, 4]:

- одномодульная арифметика вычетов
- многомодульная арифметика вычетов
- конечноразрядная p-адическая арифметика
- «фибоначчиева» арифметика

Каждая из перечисленных арифметик применяется для обработки многоразрядных чисел, однако рассматриваемая проблема еще не разрешена в комплексе, так как изменяются требования и условия по решению соответствующих задач.

Разрядно-логарифмическое (РЛ) представление (кодирование) разрабатывалось с учетом двоичной технологической базы для двоичной арифметики. Приведем основные положения РЛ структур данных [5, 6]. Известно, что любое положительное число A может быть представлено в виде конечной или бесконечной дроби

$$A = a_m * p^m + a_{m-1} * p^{m-1} + a_{m-2} * p^{m-2} + \dots$$
$$+ a_1 * p^1 + a_0 * p^0 + a_{-1} * p^{-1} \dots$$

или

$$A = \sum_{i=-\infty}^{m} a_i * p^i ,$$

где a_i – цифры числа A (набор цифр зависит от выбранной системы счисления), p –основание системы счисления.

При РЛ представлении каждый двоичный разряд a_j, который не равен нулю (значащий разряд), изображается как код N_i, который равен двоичному логарифму от количества, определяемого этим разрядом:

$$N_i = \log_2 a_i 2^i = i \ (a_i \neq 0).$$

Число A будет записано как набор кодов $N_1N_2N_3\ldots N_i\ldots$. Двоичное число с фиксированной запятой $A = \overset{12\,11\,10\,9\,8\,7\,6\,5\,4\,3\,2\,1\,0\,-1\,-2\,-3}{1\,0\,0\,0\,1\,1\,0\,1\,0\,0\,1\,0\,1\,.\,1\,0\,1}$ в РЛ представлении будет иметь следующий вид $A_{ÐË} = 12.8.7.5.2.0.-1.-3..$

Структура РЛ операнда включает поле знака числа, поле количества значащих единиц и поле набора кодов значащих единиц в упорядоченном или неупорядоченном виде:

$$A_{ÐË} = signA.Q_A.N_1N_2N_3\ldots N_{Q_A},$$

где $signA$ – поле знака числа A (0 – положительное число или 1 – отрицательное число), Q_A – поле количество значащих единиц, $N_1N_2N_3\ldots N_{Q_A}$ – поле РЛ кодов. Для приведенного выше примера числа РЛ структура будет следующая $A_{ÐË} = 0.8.12.8.7.5.2.0.-1.-3.$

Формат РЛ числа представлен на рис. 1, где Q – разрядность РЛ числа, N_i – код ненулевой единицы.

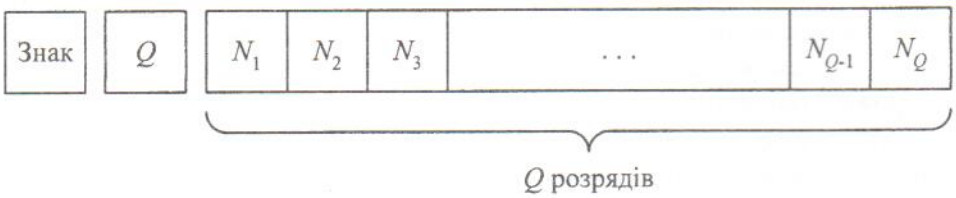

Рис. 1. Формат РЛ числа

РЛ форма представления чисел является единой для целых, дробных и смешанных чисел, т.е. при программировании следует использовать только тип данных «целое». Указанное свойство позволяет упростить программирование и организацию аппаратных средств. Следует также отметить, что исключаются процедуры нормализации и округления.

Диапазон чисел при РЛ кодировании значительно расширяется диапазон представления чисел. Это следует из того, что в разрядной сетке ЦВМ будет представляться не само число A, а коды N_i, которые равны степеням двойки значащих единиц в двоичной форме числа A.

Так при традиционном подходе при n-разрядной сетке максимальное целое знаковое число можно записать как:

$$A_{\max} = 2^{n-1} - 1,$$

а при РЛ кодировании:

$$A_{ÐË\,\max} = 2^{(2^{n-1})} - 1$$

(один разряд выделяется для знака двоичного числа).

В табл. 1 приведены диапазоны двоичного представления чисел и диапазоны РЛ представления при различной разрядности компьютера. В результате расширения диапазона представления данных сокращается влияние на точность вычислений процедуры округления.

Таблица 1

Сравнительная характеристика двоичного и РЛ представлений

Разрядность двоичного кода	Диапазон двоичных чисел	Разрядность РЛ кода	Диапазон РЛ чисел
4	$2^{+3} \le A \le 2^{-3}$	8	$2^{+7} \le A \le 2^{-7}$
8	$2^{+7} \le A \le 2^{-7}$	128	$2^{+127} \le A \le 2^{-127}$
16	$2^{+15} \le A \le 2^{-15}$	32768	$2^{+32767} \le A \le 2^{-32767}$
32	$2^{+31} \le A \le 2^{-31}$	2147483648	$2^{+2147483647} \le A \le 2^{-2147483647}$

Приведенная сравнительная характеристика (табл.1) демонстрирует важную особенность РЛ представления чисел: *не перестраивая архитектуры ЦВМ с короткой разрядной сеткой, РЛ кодирование позволяет моделировать работу ЦВМ со значительно большей разрядностью и, следовательно, большей точностью вычислений.* Компьютеры с малой разрядностью таким образом можно перевести по показателю точности в класс суперкомпьютеров.

Способ представления чисел определяет правила выполнения и алгоритмы выполняемых операций [5, 6, 7, 9]. Основные правила выполнения разрядных операций приведены в табл. 2, где N_i, N_k – ненулевые разряды операндов.

Таблица 2

Правила выполнения разрядных операций в РЛ представлении

Операция	Правила выполнения
Сложение	1) $N_i + N_k = N_i.N_k$, если $N_i > N_k$; 2) $N_i + N_k = N_i + 1$, если $N_i = N_k$;
Вычитание	1) $N_i - N_k = N_{i-1}.N_{i-2}....N_k$, если $N_i > N_k$; 2) $N_i - N_k = 0$, если $N_i = N_k$;
Умножение	$N_i * N_k = N_i + N_k$;
Сдвиг числа на m разрядов	$N_1.N_2....N_n. \rightarrow (N_1 + m).(N_2 + m)....(N_n + m).$;
Логическое И	1) $N_i \bullet N_k = 0$, если $N_i \neq N_k$; 2) $N_i \bullet N_k = N_i$, если $N_i = N_k$;
Логическое ИЛИ	1) $N_i \vee N_k = N_i$, если $N_i = N_k$; 2) $N_i \vee N_k = N_i.N_k$, если $N_i \neq N_k$;

Для организации обработки данных в РЛ представлении следует учитывать равенство номера позиции ненулевого разряда логарифму от веса, определяемого этим разрядом. Такое представление позволяет свести мультипликативные операции к реализации операции сложения-вычитания. Произведение двух чисел при РЛ представлении реализуется как процедура поэлементного сложения двух векторов. Замена умножения-деления сложением-вычитанием при реализации мультипликативных операций позволяет уменьшить аппаратные затраты в операционных структурах и упростить их организацию.

Организация обработки данных в РЛ форме основана на применении алгоритмов целочисленной арифметики. Арифметические операции над РЛ данными включают следующие базовые процедуры [5, 6, 7, 9]:

- объединение структур операндов;
- сравнение структур;
- сложение и вычитание определенных элементов структур;
- сортировка структур;
- приведение подобных элементов в структурах.

При использовании РЛ представления данных возникает вопрос о возможных вариантах использования разрядно-логарифмических структур в компьютерной обработке. Количество кодов N_i в РЛ структуре может быть неограниченно. Исходным двоичным операндам ставятся в соответствие структуры с РЛ кодами. При этом возможно гибкое представление переменной A, соответствующее изменению этой переменной в ходе решения задачи. В общем случае такое представление имеет вид:

$$A_{\text{РЛ}} = \{signA.Q_A.N_1N_2N_3 \ldots N_{Q_A}\}$$

$$(0 \le Q_A < +\infty).$$

Однако такое представление является лишь теоретическим. В действительности диапазон изменения параметра Q_A

приходится ограничивать справа некоторой величиной, ввиду следующего ряда причин:

- конечность памяти ЦЭВМ;
- ограничение на представление РЛ кодов (длина машинной сетки);
- ограничения по времени обработки.

Возможные варианты применения РЛ структур при организации компьютерной обработки:

- обработка данных с фиксированным значением количества значащих единиц Q и фиксированным большим диапазоном;
- обработка данных с адаптивным значением количества значащих единиц Q_A и адаптивным диапазоном.

Модель обработки с фиксированным значением количества значащих единиц в разрядно-логарифмических данных. Одним из вариантов использования разрядно-логарифмических структур в компьютерной обработке является обработка с фиксированным значением количества значащих единиц Q в РЛ данных [7, 8, 9, 10]. При таком варианте вычислений для всех РЛ структур, которые встречаются в ходе решения задачи выбирается некоторая граница изменения числа Q.

Схематически процесс вычислений и встречающиеся в нем РЛ данные с фиксированным значением количества значащих единиц показаны на рис. 2.

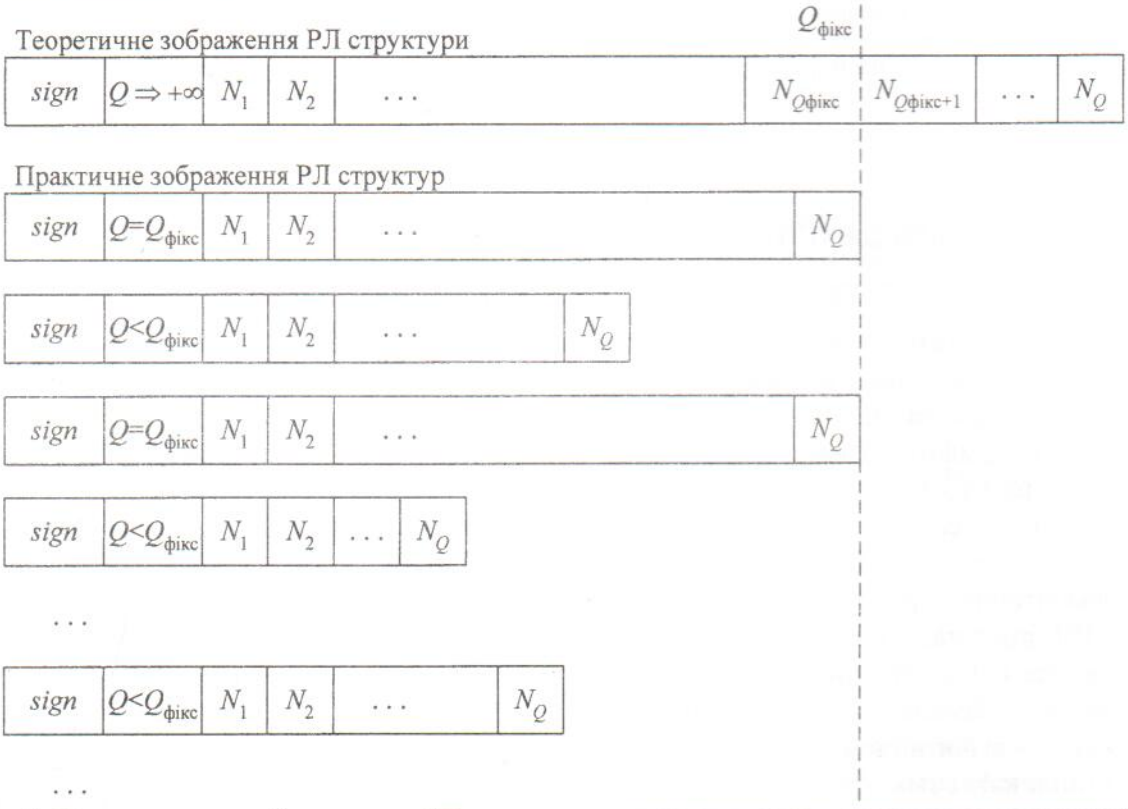

Рис. 2. Процесс изменений структуры РЛ данных при вычислениях с фиксированным значением Q

Как видно из рис. 2 длина поля РЛ кодов во всех данных не может превышать фиксированной величины Q_f. При таком алгоритме обработки операнды в каждой арифметической операции также ограничены величиной Q_f. Результат каждой операции проверяется на превышение длины поля РЛ кодов числа. Если превышения нет, результат остается неизменным, иначе – числу Q присваивается величина Q_f. Задавая параметр Q_f, можно определять объем требуемой памяти для решения задачи в начале процесса обработки.

При определении Q_f следует учитывать следующее:

- при выборе очень большого числа Q_f, увеличивается как точность вычисле-

ний, так и объем используемой памяти и время на обработку;

- при малом числе Q_f точность обработки уменьшается вместе с временем вычислений и объемом памяти, которая необходима для решения задачи. Может возникнуть ситуация, когда величина Q_f будет являться настолько малой, что результат обработки не будет иметь никакой ценности.

Исходя из вышесказанного, задача обеспечения приемлемой точности вычислений с использованием РЛ арифметики при фиксированном Q_f имеет неоднозначное решение. Важным является разработка модели вычислений, которая бы позволяла находить оптимальное значение количества значащих единиц в РЛ структурах для решения задач с требуемой точностью.

Модель обработка с адаптивным значением количества значащих единиц в разрядно-логарифмических данных. Другим вариантом использования разрядно-логарифмических структур в компьютерной обработке является обработка с адаптивным значением количества значащих единиц Q в РЛ данных.

При таком варианте вычислений ограничения на значение Q не накладываются. В процессе обработки используется величина $Q_{АД}$, которая равна максимальному значению Q среди всех РЛ структур. Данное значение не ограничивает разрядность результатов и используется для адаптации модели по точности.

Схематически процесс вычислений и встречающиеся в нем РЛ данные с адаптивным значением количества значащих единиц $Q_{АД}$, показаны на рис. 3.

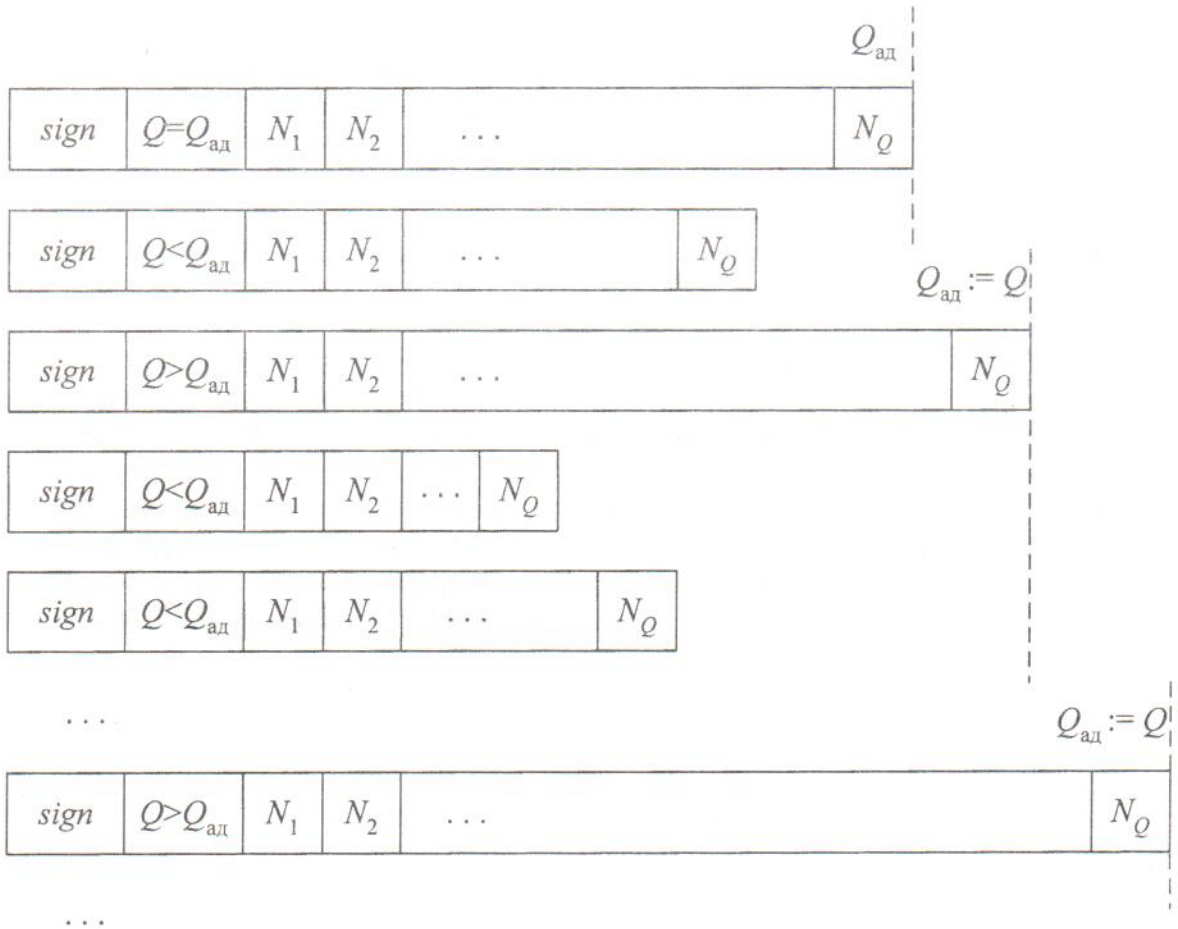

Рис.3. Процесс изменения разрядности при вычислениях с адаптивным значением Q

Как видно из рис. 3 значение $Q_{АД}$ может изменяться в процессе обработки. Этот вариант организации вычислений с РЛ структурами, также как и предыдущий вариант, расширяет возможности ЦЕОМ по показателям точности. Диапазон РЛ данных не ограничивается моделью обработки, а только разрядностью двоичных слов, представляющих РЛ коды (поля N_i). Длина рабочего поля РЛ данных адаптивно настраивается по точности на решаемую вычислительную задачу.

Ограничением для адаптивного способа обработки является ресурс используемой памяти и ресурс времени. Необходимо разработать такую модель вычислений, которая бы позволила при минимальной потере точности результатов РЛ операций сократить затраты памяти и времени на обработку данных.

Для применения модели компьютерной обработки на основе разрядно-логарифмического представления данных разработан интерфейс пользователя, который позволяет при использовании языка программирования высокого уровня работать с рассматриваемыми моделями.

Выводы

Разработанный аппарат моделирования на основе разрядно-логарифмического представления данных реализуется существующими языками программирования и не требует применения новой технологической базы. Свойства такого аппарата позволяют решать и исследовать актуальные прикладные задачи, включая обработку многоразрядных структур данных.

Список литературы

1. *Акушский И. Я., Юдицкий Д. И.* Машинная арифметика в остаточных классах. – М.: Сов. радио, 1968. – 444 с.

2. *Воробьев Н.Н.* Числа Фибоначчи: Популярные лекции по математике. – 5-е изд. – М.: Наука, 1984. – 144 с.

3. *Кнут Д.* Искусство программирования для ЭВМ. Т 2. Получисленные алгоритмы. – М.: Мир, 1977. – 723 с.

4. *Грегори Р., Кришнамурти Е.* Безошибочные вычисления. Методы и приложения: Пер. с англ. – М.: Мир, 1988. – 208 с.

5. *Гамаюн В. П.* Организация вычислений при разрядно-логарифмическом представлении данных. – К., 1996. – 17 с. – (Препр. / НАН Украины; Институт кибернетики им. В.М.Глушкова; 96-9).

6. *Гамаюн В. П.* Макрооператорные методы вычисления многоместных произведений// Микропроцессорные системы и их применение. – К.: Ин-т кибернетики им. В.М. Глушкова АН УССР, 1990. – С. 23-28.

7. *Гамаюн В. П.* Метод многооперандного умножения // УСиМ. – 1994 – N 4/5. – С.57-61.

8. *Форсайт Дж., Малькольм М., Моулер К.* Машинные методы математических вычислений: Пер. с англ. – М.: Мир, 1980. – 386 с.

9. *Гамаюн В. П.* Концепция многооперандной обработки. – К., 1997. – 30 с. – (Препр. / НАН Украины; Институт кибернетики им. В.М.Глушкова; 97-8).

10. *Ивахненко А. Г.* Индуктивный метод самоорганизации моделей сложных систем. – К.: Наук. думка, 1982. – 296 с.