

6. *Камерон Хьюз, Трейси Хьюз.* Параллельное и распределенное программирование и использованием С++. –М, СПб, Киев: 2004.
7. *Воеводин В. В., Воеводин Вл. В.* Параллельные вычисления. – СПб.: БХВ – Петербург, 2002.

В.Е.Золотовский, П.В.Савельев

ФОРМАТЫ ДАННЫХ ДЛЯ СИСТЕМЫ СИМВОЛЬНЫХ ВЫЧИСЛЕНИЙ

Базовым слоем любого вычислительного комплекса являются форматы представляемых данных. Для символьной математики вопрос об организации данных, пожалуй, наиболее важен, так как используемые форматы закладывают основу производительности системы, полученной по завершении разработки. Далее будут рассмотрены формы представления чисел произвольной размерности, а также символьных выражений, которые содержат буквенно-числовые обозначения переменных величин.

Обычно длинные числа представляются в виде коэффициентов при различных степенях основания заданной системы счисления. Более формально это показывает

$$X = x_0 + x_1B + x_2B^2 + \dots + x_nB^n,$$

если X является целым числом. В случае, когда X – дробное, формула преобразуется к виду

$$X = x_0 + \frac{x_1}{B} + \frac{x_2}{B^2} + \dots + \frac{x_n}{B^n};$$

$$X = Y \cdot B^e.$$

$$X = x,$$

где B – основание системы счисления, в котором записывается число, x_i – коэффициенты.

Знак числа обычно либо хранится отдельно, либо все $x_i < 0$.

Основание B зависит от максимального размера базового типа данных на вычислительной машине и выбирается исходя из следующих соображений:

- основание B подходит под один из базовых типов данных;
- B должно быть как можно больше, чтобы уменьшить размер представления длинного числа и увеличить скорость выполнения операций с ними, но достаточно малого размера, чтобы все операции с коэффициентами использовали базовый тип данных;
- для удобства можно выбрать степень B равную 10. Это упростит отладку модуля автоматического выполнения арифметических операций с длинными числами, а также позволит воспользоваться очень примитивными средствами вывода информации и отображения операндов и результатов;
- при использовании B равного степени двойки появляется возможность реализовывать быстрые операции на низком уровне. Однако сложной проблемой является переход от основания $B=2m$ к основанию $B=10n$, особенно для больших чисел;
- некоторые задачи, например определение простоты числа, могут использовать реализацию при $B=2m$, так как в качестве вывода используются логические значения ДА или НЕТ.

Кроме того, длинное число может быть представлено в каноническом или неканоническом виде. При каноническом представлении все коэффициенты x_i удовлетворяют неравенству

$$0 \leq x_i < B,$$

что для неканонического представления является необязательным требованием.

Альтернативным способом представления длинных чисел является метод разложения чисел на простые множители. В этом случае длинный операнд будет представляться при помощи набора простых чисел, и для каждого простого числа будет указываться количество его вхождений в данное.

Современные системы математической обработки и структурного моделирования уже предоставляют возможности определения математических выражений в естественном для исследователя-математика виде. Поэтому будем считать, что рассматриваемые символьные выражения представлены в виде строки символов, содержащей числовые коэффициенты, символьные константы и переменные, знаки математических операций, разделители целой и дробной частей, а также скобки.

Данное допущение позволяет построить систему, которая будет легко интегрироваться с существующими вычислительными комплексами. Однако представление символьных выражений просто в виде текстовой строки нерационально с точки зрения производительности модуля математических операций, поскольку на каждом шаге вычислений требуется разбор строки, собственно сами вычисления над частью исходного операнда либо над полным операндом и формирование выходной строки результата. Поэтому необходимо использовать внутреннее представление символьных выражений, которое позволит оптимизировать скорость вычислительного процесса, а следовательно, сократит время моделирования в целом.

Предлагается использовать представление в виде графического дерева, вершины которого являются слагаемыми, а ребра показывают отношение суммы. Таким образом, исходный вид строки символьного выражения на начальном этапе вычислений разбирается и формируется особая списковая структура, которая и будет являться операндом внутри модуля.

Для примера покажем разбор следующего выражения в такой граф:

$$7AF^2(2A-3B)(4C+E)^3 - \sqrt{2A}(2C+3A) \equiv$$

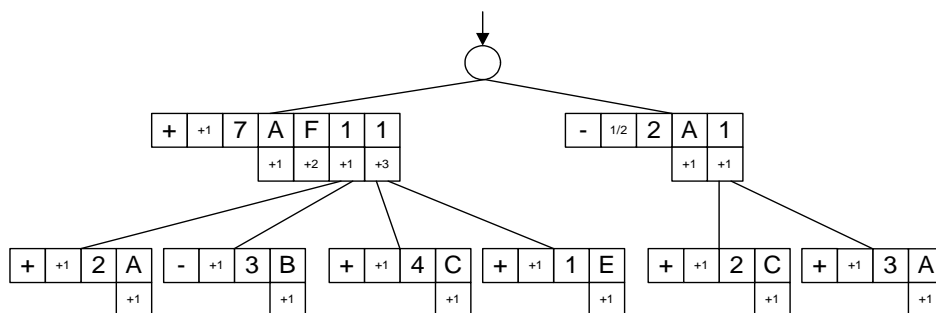


Рис. 1. Дерево символьного выражения

На рис.1 стрелка показывает точку входа в дерево. На ее месте мог бы стоять единичный элемент. Слагаемые собраны в узлы дерева. Каждое слагаемое содержит знак, числовой коэффициент и список переменных с соответствующими степенями. Если какое-то из слагаемых содержит в качестве множителя другое символическое выражение, то узел расширяется на единичный элемент, который хранит только степень. От него ребра идут к слагаемым, составляющим данный множитель.

Здесь наибольший интерес представляет узел дерева. Покажем схематично его структуру в общем виде (рис.2).

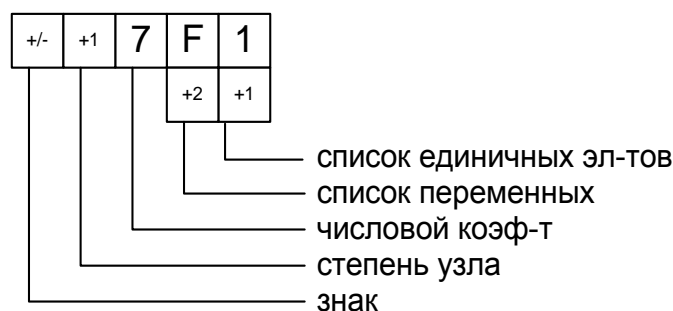


Рис. 2. Структура узла дерева выражения

Слева направо структура узла содержит:

- знак всего слагаемого;
- степень, в которую возводится все слагаемое, включая нижележащие узлы;
- числовой коэффициент, представленный в виде длинного числа, формат которого описан в предыдущем пункте;
- список символьных переменных вместе с их степенями, причем при вычислениях использоваться должны не символьные имена переменных, а их коды из общей таблицы переменных;
- дополнительные единичные элементы, содержащие степень, в которую необходимо возвести сумму узлов, непосредственно соединенных с данным узлом.

В нижних ячейках показаны значения степени. Это знаковая величина с плавающей точкой. Если числовой коэффициент беззнаковый, то требуется еще одна ячейка памяти для хранения знака узла.

Такое представление позволяет легко получать доступ к различным элементам без дополнительного разбора выражения, также с легкостью организовывать выполнение базовых математических операций, из которых состоят более сложные вычислительные процессы моделирования.

Таким образом, система символьных вычислений, построенная на предложенных форматах данных, сможет выполнять все математические преобразования с повышенной точностью либо вовсе без потери точности.