

Анализ и оптимизация циклов с помощью производящих функций

П.М. Довгалюк
(Pavel.Dovgaluk@nov.astrosoft.ru)

Аннотация. В статье рассматривается метод анализа и оптимизации циклов с помощью производящих функций, состоящий в поиске выражений для конечных значений переменных, которые вычисляются в цикле и замене цикла вычислениями по формуле.

1. Введение

Существует множество методов оптимизации циклов. Большинство методов только изменяют структуру цикла, оставляя его в программе. Среди них выделяется метод оптимизации циклов, которые не выполняют никаких действий, кроме изменения некоторых переменных. Для этого используется интерпретация циклов на этапе компиляции, если количество итераций невелико [3]. Если же число итераций превышает некоторое пороговое значение, то данный метод оказывается неэффективным.

В данной статье рассматривается метод оптимизации циклов следующего вида с помощью производящих функций:

$$\text{While } (P) \quad \begin{array}{l} \text{для } \forall v \in V \mid v = v \rightarrow C \\ \text{для } \forall v \in V \mid v = T(v, C) \end{array}$$

где:

V – множество переменных, используемых в цикле, N – мощность этого множества;

C – множество констант;

$V \rightarrow C$ – множество начальных значений переменных, используемых в цикле;

P – предикат, сигнализирующий об окончании цикла, представляющий собой результат применения логических и алгебраических операций к переменным из множества V и константам из множества C ;

$T(V, C)$ – множество функций преобразования элементов множества V , используемых в цикле.

2. Описание метода

Производящая функция последовательности $\{a_n\}$, $n \geq 0$ – формальный степенной ряд:

$$A(z) = a_0 + a_1 z + a_2 z^2 + \dots = \sum_{n \geq 0} a_n z^n.$$

Переход от последовательностей к их производящим функциям позволяет заменить операции над последовательностями операциями над их производящими функциями.

Идея метода производящих функций состоит в том, что рекуррентное соотношение, определяющее последовательность $\{a_n\}$, переписывают как уравнение для ее производящей функции, это уравнение решают и по найденной производящей функции получают зависимость общего члена последовательности $\{a_n\}$ от n .

Для того, чтобы получить алгебраические выражения для значения каждой переменной в зависимости от номера итерации, необходимо преобразовать операции, изменяющие значения переменных, в рекуррентные соотношения вида:

$$\begin{array}{l} v_0 = c \\ v_i = t(V^{i-1}, V^i, C), \end{array}$$

где i – номер итерации;

t – функция преобразования переменной v , принадлежащая множеству T ;

V_{i-1} – значения переменных, полученные на предыдущей итерации;

V_i – значения переменных, полученные на текущей итерации.

Производящие функции (или выражения для v_i) достаточно просто можно получить для присваиваний вида:

$$v_i = f(V^{i-1} - v, V^i, C).$$

В этом случае для получения выражения для v_i достаточно подставить выражения для всех остальных переменных в правую часть или найти производящую функцию для f .

Пользуясь методикой из [2], каждый оператор присваивания вида $v_i = f_0 \cdot v_{i-1} + f(V^{i-1} - v, V^i, C)$ можно записать в виде алгебраического тождества, включающего производящие функции для переменных из множества V :

$$F_v(z) = G(F_V, C, z) + c,$$

где $F_v(z)$ – производящая функция для переменной v ;

F_V – множество производящих функций для переменных, принадлежащих множеству V ;

G – некоторая алгебраическая функция.

Найти алгебраическое выражение (возможно, содержащее в себе другие производящие функции) для производящей функции для f можно, пользуясь

линейностью производящих функций: производящая функция суммы равняется сумме производящих функций.

Для каждого из слагаемых существует следующий выбор:

1. Производящую функцию можно определить непосредственно (например, если слагаемое представляет собой константу, умноженную на переменную) по таблице производящих функций из [2].
2. Производящую функцию нельзя получить непосредственно. Например, $a_i \cdot b_i$. В этом случае необходимо подставить выражения для всех переменных (a_i и b_i), а затем найти производящую функцию для последовательности, которую генерирует это выражение при изменении i от 0 до бесконечности.
3. Производящую функцию нельзя получить непосредственно (как в случае 2), но выражения для a_i и b_i (см. случай 2) невозможно получить без знания выражения для v_{i+1} . В этом случае предлагаемый метод не работает.

Полученные выражения для переменных можно подставить в P , чтобы определить номер итерации, на которой произойдет выход из цикла. Если данную задачу удастся решить, то цикл может быть заменен вычислениями по формуле или просто подстановкой констант в случае, когда все параметры, входящие в выражения для результирующих переменных известны на этапе компиляции.

Рассмотрим несколько примеров:

1. Переменная, которая в каждой итерации умножается на константу и к результату прибавляется другая константа:

$$v_0 = c_0$$

$$v_i = c_1 \cdot v_{i-1} + c_2$$

Найдем производящую функцию для данной последовательности:

$$z^i \cdot v_i = [i = 0] \cdot c_0 \cdot z^i + [i > 0] \cdot (c_1 \cdot v_{i-1} + c_2) \cdot z^i$$

$$\sum_{i \geq 0} v_i \cdot z_i = c_0 + c_2 \sum_{i \geq 1} z^i + c_1 \sum_{i \geq 1} v_{i-1} \cdot z^i$$

$$V(z) = c_0 + c_2 \cdot z \cdot \sum_{i \geq 0} z^i + c_1 \cdot z \cdot \sum_{i \geq 0} v_i \cdot z^i$$

$$V(z) = c_0 + \frac{c_2 \cdot z}{1-z} + c_1 \cdot z \cdot V(z)$$

$$V(z) = \frac{c_0}{1-c_1 \cdot z} + \frac{c_2 \cdot z}{(1-z) \cdot (1-c_1 \cdot z)}$$

Отсюда можно получить алгебраическое выражение для v в зависимости от номера итерации:

$$\text{для } c1 = 1: v_i = c_0 + c_2 \cdot i \quad (*)$$

$$\text{для } c1 \neq 1: v_i = c_0 \cdot c_1^i + c_2 \cdot \frac{c_1^i - 1}{c_1 - 1}$$

2. Линейная комбинация переменных:

$$v_0^k = c_0$$

$$v_i^k = \sum_{j=1}^{k-1} v_i^j \cdot c_j + \sum_{j=k}^N v_{i-1}^j \cdot c_j$$

где k – номер переменной (переменные с номерами меньше k изменились в текущей итерации, с номером большим k – будут изменены позднее).

Применяя манипуляции, подобные использованным в предыдущем пункте, можно получить следующий результат:

$$V_k(z) = \sum_{j=1}^{k-1} V_j(z) \cdot c_j + z \cdot \sum_{j=k}^N V_j(z) \cdot c_j + c_0$$

Отсюда можно получить выражение для $V_k(z)$, а из него – выражение для v^k .

3. Многочлен от переменных, выражения для которых имеют вид (*):

$$v_0^k = c_0$$

$$v_i^k = P(v_i^1, \dots, v_i^{k-1}, v_{i-1}^{k+1}, \dots, v_{i-1}^N) + c_1 \cdot v_{i-1}^k$$

Метод, применяемый в предыдущих примерах здесь не сработает, поэтому необходимо выполнить подстановку выражений для v^j в данную формулу, что приведет к появлению многочлена от i :

$$v_i^k = R(i) + c_1 \cdot v_{i-1}^k$$

Производящая функция для i^n не существует в простой форме, однако ее можно выразить через производящие функции для числа сочетаний:

$$\frac{1}{(1-z)^{n+1}} = \sum_{i \geq 0} C_{n+i}^n \cdot z^i = \sum_{i \geq 0} \frac{(i+n) \cdot \Lambda \cdot (i+1)}{n!} \cdot z^i$$

Отсюда можно получить производящие функции для i^n :

$$C_{i+1}^1 = \frac{i+1}{1!}$$

$$i^1: i = C_{i+1}^1 - 1$$

$$I_1(z) = \frac{1}{(1-z)^2} - \frac{1}{1-z}$$

$$C_{i+2}^2 = \frac{(i+2) \cdot (i+1)}{2!}$$

$$i^2: i^2 = 2 \cdot C_{i+2}^2 - 3 \cdot i - 2 = 2 \cdot C_{i+2}^2 - 3 \cdot C_{i+1}^1 + 1$$

$$I_2(z) = \frac{2}{(1-z)^3} - \frac{3}{(1-z)^2} + \frac{1}{1-z}$$

и так далее.

Таким образом, производящая функция для v_i^k будет иметь вид:

$$V_k(z) = \frac{z}{1 - c_1 \cdot z} \cdot \sum_{i \geq 0} r_i \cdot I_i(z) + c_0$$

где r_i – коэффициент при i^n в $R(i)$, а $I_i(z)$ – производящая функция для i^n .

3. Выводы

В статье рассмотрен метод анализа циклов с помощью производящих функций. Несмотря на то, что не все циклы могут быть подвергнуты анализу с помощью данного метода, его можно применить для решения нижеперечисленных задач:

1. Данный метод выполняет задачу, обратную описанной в [3, разд. 14.1.1] – т.е. заменяет инкрементальные вычисления вычислениями по формуле. Кроме того, в процессе анализа цикла с помощью данного метода можно найти все его индуктивные переменные.
2. Компилятор, проделав все вышеприведенные вычисления, может либо заменить результат вычислений на константу (если все параметры являются постоянными), либо использовать вместо цикла соответствующую формулу.
3. Если цикл, подобный вышеприведенному, является вложенным в какой-либо другой, то производящие функции для переменных из V можно использовать описанным выше способом при оптимизации главного цикла.
4. Данный метод, как и многие другие методы алгебраической оптимизации, может быть использован только для целых чисел из-за возможной потери точности или переполнения чисел с плавающей точкой.
5. Полученные выражения для переменных могут быть использованы при анализе зависимостей между переменными, определении псевдонимов, устранении общих подвыражений (в том числе и полученных при выполнении аналогичных действий в различных циклах), при устранении незначимых вычислений, при определении индуктивных переменных.
6. Аналогичный подход может быть применен при оптимизации некоторых видов рекурсивных вычислений.

Литература

1. А. Ахо, Р. Сети, Д. Ульман. Компиляторы: принципы, технологии и инструменты.: Пер. с англ. – М.: Издательский дом «Вильямс», 2001
2. Р. Грэхем, Д. Кнут, О. Паташник. Конкретная математика. Основание информатики: Пер. с англ. – М.: Мир, 1998
3. S. Muchnick. Advanced compiler design and implementation, 1997