ПРИБЛИЖЕННАЯ ТОНКАЯ ИНФОРМАЦИОННАЯ СТРУКТУРА АЛГОРИТМОВ

Н. А. Лиходед

Введение

Любые процедуры распараллеливания должны опираться на знание информационной структура рассматриваемого алгоритма, т.е. на знание зависимостей (информационных связей) на уровне фрагментов, макроопераций или отдельных операций алгоритма. На практике широко используются графовые и аналитические способы представления зависимостей: граф вызовов, граф управления, граф алгоритма, редуцированный граф зависимостей, уровни зависимостей, векторы зависимостей, многогранники зависимостей, функции зависимостей.

Особое значение для методов распараллеливания имеют функции зависимостей. Они являются удобным математическим аппаратом для описания тонкой информационной структуры алгоритма, т.е. для описания информационных связей на уровне отдельных операций алгоритма. Другие названия этих функций — покрывающие функции графа алгоритма [1, 2], h-преобразования [3, 4]. Доказано (теорема В.В. Воеводина об информационном покрытии [1, 2]), что все зависимости алгоритмов, представленных довольно широким линейным классом программ, можно задать функциями, линейно зависящими от параметров циклов и внешних переменных; каждая из функций определена на линейном многограннике; число функций не зависит от внешних переменных и для реальных линейных программ пропорционально числу операторов присваивания; коэффициент пропорциональности не превосходит нескольких единиц.

Во многих случаях получение информационной структуры алгоритма является трудной задачей и требует применения специальных методов. Несмотря на достигнутые успехи в автоматизации поиска тонкой информационной структуры алгоритма [5–7], получение функций зависимостей в приемлемом для практического использования виде остается актуальной задачей.

В этой работе разработан метод получения функций, задающих зависимости между операциями алгоритмов, представленных многомерными циклами произвольной структуры вложенности. Метод параметрически учитывает внешние переменные циклов и позволяет получать функции зависимостей в случае, когда между парой вхождений массивов в операторы может быть только одна зависимость (это наиболее часто встречаемый в приложениях случай). Получаемая информационная структура алгоритма является приближенной в том смысле, что функции зависимостей могут иметь несколько избыточную область определения, и теоретически возможны потери при обнаружении потенциального параллелизма. По сравнению с точными методами получения тонкой информационной структуры алгоритмов, новый метод лучше приспособлен для автоматизации, что существенно для приложений.

Предварительные сведения

Пусть алгоритм задан аффинным гнездом циклов: индексные выражения переменных и границы изменения параметров циклов алгоритма являются аффинными функциями от параметров циклов и внешних переменных.

Пусть в гнезде циклов имеется K операторов S_β и используется L массивов a_l. Простые переменные будем считать массивами размерности 0. Пусть $V_\beta \subset \mathbf{Z}^{n_\beta}$ — область изменения параметров гнезда циклов для оператора S_β, где n_β — число циклов, окружающих оператор S_β; $W_l \subset \mathbf{Z}^{\nu_l}$ — область изменения индексов l-го массива, где ν_l — размерность l-го массива; $N \in \mathbf{Z}^e$ — вектор внешних переменных алгоритма, где e — число внешних переменных. Реализацию оператора S_β при конкретном значении вектора параметров циклов J будем называть операцией и обозначать $S_\beta(J)$. Выполнение всех операций, зависящих от J, называется J-й итерацией.

Вхождением (l, β, q) будем называть q-е вхождение массива a_l в оператор S_β. Для наглядности будем наряду с (l, β, q) использовать также обозначение (a_l, S_β, q). Пусть индексы элементов l-го массива, связанных с вхождением (l, β, q), выражаются аффинной функцией $\overline{F}_{l,\beta,q} : V_\beta \to W_l$ вида

$$\overline{F}_{l,\beta,q}(J) = F_{l,\beta,q}J + G_{l,\beta,q}N + f^{(l,\beta,q)},$$
$$J \in V_\beta,\ N \in \mathbf{Z}^e,\ F_{l,\beta,q} \in \mathbf{Z}^{\nu_l \times n_\beta},\ G_{l,\beta,q} \in \mathbf{Z}^{\nu_l \times e},\ f^{(l,\beta,q)} \in \mathbf{Z}^{\nu_l}.$$

Операция $S_\beta(J),\ J \in V_\beta$, зависит от операции $S_\alpha(I),\ I \in V_\alpha$, если:

1) $S_\alpha(I)$ выполняется раньше $S_\beta(J)$;

2) $S_\alpha(I)$ и $S_\beta(J)$ используют один и тот же элемент какого-либо массива (т. е. $\overline{F}_{l,\alpha,p}(I) = \overline{F}_{l,\beta,q}(J)$ для некоторых l, p, q), и, по крайней, мере одно из использований есть переопределение (изменение) элемента;

3) между операциями $S_\alpha(I)$ и $S_\beta(J)$ этот элемент не переопределяется. Зависимость операции $S_\beta(J)$ от операции $S_\alpha(I)$, порождаемая вхождениями (l, α, p) и (l, β, q), будем обозначать $S_\alpha(I) \stackrel{l,p,q}{\to} S_\beta(J)$.

Зависимость называется: истинной, если элемент массива сначала определяется, а затем его значение используется в качестве аргумента; антизависимостью, если значение элемента массива сначала используется как аргумент, а затем переопределяется; зависимостью по выходу, если элемент массива определяется, а затем переопределяется. Если функция $\overline{F}_{l,\alpha,p}(I)$ (см. определение зависимости) встречается в левой части оператора S_α, а $\overline{F}_{l,\beta,q}(J)$ встречается в правой части оператора S_β, то зависимость является истинной.

Если $\overline{F}_{l,\alpha,p}(I)$ встречается в правой части S_α, а $\overline{F}_{l,\beta,q}(J)$ — в левой части S_β, то имеет место антизависимость. Если обе функции $\overline{F}_{l,\alpha,p}(I)$ и $\overline{F}_{l,\beta,q}(J)$ встречаются в левых частях операторов, то зависимость является зависимостью по выходу.

Обозначим $P = \{ ((l,\alpha,p),(l,\beta,q)) \mid \exists\ I \in V_\alpha,\ J \in V_\beta,\ \ S_\alpha(I) \overset{l,p,q}{\to} S_\beta(J) \}$ — множество, которое определяет пары вхождений, порождающих зависимости. Для каждой пары $((l,\alpha,p),(l,\beta,q)) \in P$ обозначим $V^{l,p,q}_{\alpha,\beta} = \{ J \in V_\beta \mid \exists\ S_\alpha(I) \overset{l,p,q}{\to} S_\beta(J) \}$. Пусть зависимости операций алгоритма задаются функциями $\overline{\Phi}^{l,p,q}_{\alpha,\beta} : V^{l,p,q}_{\alpha,\beta} \to V_\alpha$ таким образом, что если $S_\alpha(I) \overset{l,p,q}{\to} S_\beta(J)$, $I \in V_\alpha$, $J \in V^{l,p,q}_{\alpha,\beta} \subseteq V_\beta$, то $I = \overline{\Phi}^{l,p,q}_{\alpha,\beta}(J)$. Будем называть функции $\overline{\Phi}^{l,p,q}_{\alpha,\beta}$ функциями зависимостей и предполагать, что они являются аффинными:

$$\begin{gathered}\overline{\Phi}^{l,p,q}_{\alpha,\beta}(J) = \Phi^{l,p,q}_{\alpha,\beta} J + \Psi^{l,p,q}_{\alpha,\beta} N - \varphi^{\alpha,\beta,l,p,q},\\ J \in V^{l,p,q}_{\alpha,\beta},\ N \in \mathbf{Z}^e,\ \Phi^{l,p,q}_{\alpha,\beta} \in \mathbf{Z}^{n_\alpha \times n_\beta},\ \Psi^{l,p,q}_{\alpha,\beta} \in \mathbf{Z}^{n_\alpha \times e},\ \varphi^{\alpha,\beta,l,p,q} \in \mathbf{Z}^{n_\alpha},\\ ((l,\alpha,p),(l,\beta,q)) \in P.\end{gathered} \tag{1}$$

Основные этапы метода

Нашей задачей является разработать метод получения функций зависимостей в предположении, что любая пара вхождений порождает не более одной функции. Такое предположение обычно не является ограничительным.

Зафиксируем пару вхождений $((l,\alpha,p),(l,\beta,q))$, порождающих зависимость $S_\alpha \overset{l,p,q}{\to} S_\beta$. Представим функции зависимостей (1) в виде

$$\overline{\Phi}^{l,p,q}_{\alpha,\beta}(J) = \left(\Phi^{l,p,q}_{\alpha,\beta} \middle| \Psi^{l,p,q}_{\alpha,\beta} \middle| -\left(\varphi^{\alpha,\beta,l,p,q}\right)^{\mathrm{T}}\right)(J|N|1)^{\mathrm{T}}.$$

Для упрощения записи будем иногда опускать верхние и нижние индексы.

Зависимые итерации I и J связаны соотношениями $(\Phi\,|\Psi| - (\varphi)^{\mathrm{T}})\,(J|N|1)^{\mathrm{T}} = I^{\mathrm{T}}$, или

$$(J|N|1)\,(\Phi\,|\Psi| - (\varphi)^{\mathrm{T}})^{\mathrm{T}} = I. \tag{2}$$

Возьмем некоторое число M различных наборов $J^{(m)}$, $N^{(m)}$, $I^{(m)}$, где $I^{(m)}$ и $J^{(m)}$ — связанные зависимостью итерации. Наборы зависимых итераций $I^{(m)}$ и $J^{(m)}$ можно получить раскруткой циклов при фиксированных значениях внешних параметров $N^{(m)}$ и анализом общих элементов массивов.

При получении зависимых итераций можно анализировать особенности области определения функции $\overline{\Phi}^{l,p,q}_{\alpha,\beta}$ — множества $V^{l,p,q}_{\alpha,\beta}$. Имеет место включение $V^{l,p,q}_{\alpha,\beta} \subseteq V_\beta$. На практике довольно часто $V^{l,p,q}_{\alpha,\beta}$ отличается от V_β фиксированным значением некоторой координаты. Поэтому, если для зависимости $S_\alpha(I) \overset{l,p,q}{\to} S_\beta(J)$ значение какой-либо координаты J есть величина постоянная, то определим $V_{\alpha,\beta}$ как V_β с соответствующим фиксированным значением этой координаты. Такая редукция V_β часто является хорошим приближением множества $V_{\alpha,\beta}$ и не приводит к потере параллелизма. Отметим еще, что из определения функции зависимостей следует $V^{l,p,q}_{\alpha,\beta} \subseteq \{\, J \in V_\beta \mid \overline{\Phi}^{l,p,q}_{\alpha,\beta}(J) \in V_\alpha \}$; множество $V^{l,p,q}_{\alpha,\beta}$ не шире множества точек с целочисленными координатами, принадлежащих многограннику, задаваемому системой неравенств $J \in V_\beta, \overline{\Phi}^{l,p,q}_{\alpha,\beta}(J) \in V_\alpha$.

Составим матрицы из наборов зависимых итераций:

$$\tilde{J} = \begin{pmatrix} J^{(1)} & N^{(1)} & 1 \\ J^{(2)} & N^{(2)} & 1 \\ \dots & \dots & \dots \\ J^{(M)} & N^{(M)} & 1 \end{pmatrix}, \quad \tilde{I} = \begin{pmatrix} I^{(1)} \\ I^{(2)} \\ \dots \\ I^{(M)} \end{pmatrix}.$$

Исходя из соотношений (2) получим относительно $(\Phi\,|\Psi| - (\varphi)^{\mathrm{T}})^{\mathrm{T}}$ матричное уравнение

$$\tilde{J}\,(\Phi\,|\Psi| - (\varphi)^{\mathrm{T}})^{\mathrm{T}} = \tilde{I}. \tag{3}$$

Утверждение. *Пусть вхождения (l,α,p) и (l,β,q) порождают ровно одну функцию зависимостей. Если ранг матрицы $\tilde{J}$ равен $n_\beta + e + 1$, то параметры функции зависимостей (1) определяют единственное решение уравнения (3).*

Действительно, для нахождения n_α столбцов матрицы $(\Phi\,|\Psi| - (\varphi)^{\mathrm{T}})^{\mathrm{T}}$ требуется решить n_α систем линейных алгебраических уравнений (с $n_\beta+e+1$ неизвестными в каждой системе) с одной и той же матрицей $\tilde{J}$, но с разными правыми частями. В матрице $\tilde{J}$ существенное значение имеют только такие строки $J^{(m)}\ N^{(m)}\ 1$, вычеркивание которых уменьшает ранг матрицы. Так как по условию матрица $\tilde{J}$ имеет полный столбцовый ранг $n_\beta + e + 1$, то решение каждой системы существует и единственно. Согласно теореме В.В. Воеводина об информационном покрытии, зависимости, порождаемые любой парой вхождений, задаются функциями вида (1). По условию зависимость описывается ровно одной функцией, поэтому параметры функции зависимостей определяются единственным решением матричного уравнения (3).

В вырожденных случаях ранг матрицы $\tilde{J}$ может оказаться меньше $n_\beta+e+1$ при любом числе строк M. Тогда параметры функции зависимостей определяются неоднозначно, так что одна (на заданной области определения) функция может иметь разные представления; этот случай будет продемонстрирован в примере в конце раздела.

Отметим, что в данной работе не рассматривается случай порождения одной парой вхождений более одной функции зависимостей. Вообще говоря, такое возможно [1]; в этом случае функции определены на линейных многогранниках, точки которых являются непересекающимися подмножествами множества V_β. Информацию о такой ситуации для исследуемой пары вхождений можно получить, сформировав несколько матриц $\tilde{J}$, используя $J^{(m)}$ из разных нескольких частей области V_β. Наличие различных решений задачи получения параметров функций вида (1) будет говорить о более чем одной функции зависимостей.

Для решения системы (3) используем метод, основанный на приведении матрицы $\tilde{J}$ к эрмитовой целочисленной нормальной форме [8, 9]. Пусть ρ — ранг матрицы $\tilde{J}$. Тогда сложением, вычитанием и перестановкой строк можно привести матрицу к целочисленной матрице H следующей структуры: первые ρ строк матрицы H ненулевые, все элементы остальных строк равны нулю; первый ненулевой элемент в i-й ($i = 1, 2, \ldots, \rho$) строке матрицы H положителен; пусть первый ненулевой элемент в i-й ($i = 1, 2, \ldots, \rho$) строке входит в столбец с номером c_i; тогда $c_1 < c_2 < \ldots < c_\rho$; в столбце с номером c_i все элементы неотрицательны, наибольшим является элемент в i-й строке, в строках с номерами, большими i, находятся нули.

Приведем матрицу $\tilde{J}$ к целочисленной эрмитовой нормальной форме; при этом соответствующие строчные преобразования будем применять и к матрице $\tilde{I}$. Получим матрицу $(H|P\tilde{I})$, где P — матрица элементарных строчных преобразований при получении эрмитовой нормальная форма матрицы $\tilde{J}$. Затем, применяя обратный ход, в каждой из n_α систем линейных алгебраических уравнений (3) последовательно выразим неизвестные. Если ранг матрицы $\tilde{J}$ меньше $n_\beta + e + 1$, то зададим значения координат искомого вектора-решения на местах с номерами, не равными c_i произвольно, а затем выразим остальные координаты.

Таким образом, можно выделить основные этапы получения функций зависимостей:

- выявление зависимостей (определение пар вхождений $((l, \alpha, p), (l, \beta, q))$, порождающих зависимости), определение типа зависимости;
- формирование матриц $\tilde{J}$, $\tilde{I}$;
- получение матриц $\Phi^{l,p,q}_{\alpha,\beta}$, $\Psi^{l,p,q}_{\alpha,\beta}$ и векторов $\varphi^{\alpha,\beta,l,p,q}$ путем решения матричного уравнения (3);
- получение области определения $V^{l,p,q}_{\alpha,\beta}$ редукцией множества V_β или полагая (возможно, избыточно) $V^{l,p,q}_{\alpha,\beta} = \{\, J \in \mathbf{Z}^{n_\beta} \mid J \in V_\beta,\ \overline{\Phi}^{l,p,q}_{\alpha,\beta}(J) \in V_\alpha\}$.

Приведем иллюстрационный пример. Алгоритм умножения матрицы A порядка N на N-мерный вектор b можно записать в виде

$$
\begin{array}{l}
\text{do}\ \ i = 1, N \\
\quad S_1:\ c(i) = 0 \\
\quad \text{do}\ \ j = 1, N \\
\quad\quad S_2:\ c(i) = c(i) + a(i,j)b(j) \\
\quad \text{enddo} \\
\text{enddo}
\end{array}
$$

Рассмотрим зависимости, порождаемые вхождениями $(c, S_1, 1)$ и $(c, S_2, 2)$. С помощью раскрутки циклов, например для итераций $i = 1,\ i = 2$,

$$\begin{array}{ll}
S_1(1): & c(1) = 0 \\
S_2(1,1): & c(1) = c(1) + a(1,1)b(1) \\
S_2(1,2): & c(1) = c(1) + a(1,2)b(2) \\
\cdots\cdots & \cdots\cdots\cdots\cdots \\
S_2(1,N): & c(1) = c(1) + a(1,N)b(N) \\
S_1(2): & c(2) = 0 \\
S_2(2,1): & c(2) = c(2) + a(2,1)b(1) \\
S_2(2,2): & c(2) = c(2) + a(2,2)b(2) \\
\cdots\cdots & \cdots\cdots\cdots\cdots \\
S_2(2,N): & c(2) = c(2) + a(2,N)b(N)
\end{array}$$

и определения истинной зависимости, получим $(J^{(1)},\ N^{(1)},\ I^{(1)}) = ((1,1),\ 2,\ 1)$, $(J^{(2)},\ N^{(2)},\ I^{(2)}) = ((2,1),\ 2,\ 2)$, $(J^{(3)},\ N^{(3)},\ I^{(3)}) = ((1,1),\ 3,\ 1)$.

Для любых зависимых итераций I и J имеем: если $I = i_0$ $(i_0 \geqslant 1)$, то $J = (i_0, 1)$. Отсюда следует, что область определения функции $\overline{\Phi}^{c,1,2}_{S_1,S_2}$ — множество $V^{c,1,2}_{S_1,S_2}$ — можно получить из множества $V_{S_2} = \{(i,j) \in \mathbf{Z}^2 |\ 1 \leqslant i,j \leqslant N\}$ редукцией $j = 1$. Вырождение двумерной области V_{S_2} в одномерную $V^{c,1,2}_{S_1,S_2}$ приводит к тому, что при любом числе строк M ранг матрицы $\tilde{J}$ меньше числа неизвестных $n_{S_2} + e + 1$ $(n_{S_2} = 2,\ e = 1)$.

Приведем матрицу $\tilde{J}$ к целочисленной эрмитовой нормальной форме; строчные преобразования будем применять к расширенной матрице $(\tilde{J}|\tilde{I}) = \left(\begin{array}{cccc|c} 1 & 1 & 2 & 1 & 1 \\ 2 & 1 & 2 & 1 & 2 \\ 1 & 1 & 3 & 1 & 1 \end{array}\right)$. Вычтем первую строку из второй и третьей строк, затем из первой строки вычтем вторую, дважды третью строку, переставим первую и вторую строки. Получим $(H|P\tilde{I}) = \left(\begin{array}{cccc|c} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{array}\right)$. Из третьей строки получим $\Psi^{c,1,2}_{S_1,S_2} = (0)$. Из первых двух строк, полагая $\varphi^{S_1,S_2 c,1,2} = 0$, получим $\Phi^{c,1,2}_{S_1,S_2} = (1\ 0)$. Получили известную функцию зависимостей $\overline{\Phi}^{c,1,2}_{S_1,S_2}(i,1) = i$, $(i,1) \in V_{S_1,S_2} = \{(i,1) \in \mathbf{Z}^2 |\ 1 \leqslant i \leqslant N\}$.

Таким образом, в работе предложен метод получения функций, задающих тонкую информационную структуру алгоритма (возможно, избыточную). Метод параметрически учитывает внешние переменные алгоритма и хорошо приспособлен для автоматизации.

Литература

1. Воеводин В.В., Воеводин Вл.В. Параллельные вычисления. СПб.: БХВ-Петербург, 2002. - 600 с.

2. Воеводин В.В. Вычислительная математика и структура алгоритмов. Москва: Изд-во МГУ, 2006. - 112 с. http://parallel.ru/info/parallel/voevodin/

3. Feautrier P.Some efficient solutions to the affine scheduling problem. Part 1 // International Journal of Parallel Programming. 1992. Vol. 21/ № 5. P. 313–348

4. Bondhugula U., Baskaran M., Krishnamoorthy S., Ramanujam J., Rountev A., Sadayappan P.Automatic transformations for communication-minimized parallelization and locality optimization in the polyhedral model // International Conference on Compiler Construction (ETAPS CC), Apr 2008, Budapest, Hungary.

5. http://parallel.ru/v-ray/.

6. http://ops.rsu.ru/about.shtml.

7. http://pluto-compiler.sourceforge.net.

8. Маркус М., Минк Х. Обзор по теории матриц и матричных неравенств. Москва: Изд-во Наука, 1972. - 232 с.

9. Darte A.Mathematical tools for loop transformations: from systems of uniform recurrence equations to the polytope model // Algorithms for Parallel Processing, IMA Volumes in Mathematics and its Applications. 1999. V. 105. P. 147-183.

N. A. Likhoded

Approximate fine information structure of algorithms

Summary

A method for obtaining of functions setting data dependencies for sequential programs is presented. The method is suitable for the automatization. In practice it is applicable in most cases.

УДК 519.6+681.3.012

Л и х о д е д Н. А. **Приближенная тонкая информационная структура алгоритмов** // Труды Института математики. 200?. Т. ?. № ?. С. **??**–8.

Разработан метод получения функций, задающих зависимости между операциями алгоритмов, представленных последовательными программами. Метод приспособлен для автоматизации и позволяет получать функции зависимостей в наиболее часто встречаемых в приложениях случаях.

Библиогр. – 9 назв.