

Н. А. ЛИХОДЕД

## СОХРАНЕНИЕ ЗАВИСИМОСТЕЙ МЕЖДУ ОПЕРАЦИЯМИ ПРИ РАСПАРАЛЛЕЛИВАНИИ АЛГОРИТМОВ

### Введение

Требование сохранения зависимостей, т.е. требование сохранения порядка выполнения информационно связанных операций, является обязательным при выполнении преобразования алгоритмов. Поэтому разработка удобных для применения формализованных условий сохранения зависимостей привлекает внимание специалистов в области распараллеливания алгоритмов и программ.

Основные подходы к получению ограничений, задающих условия сохранения зависимостей, можно условно разделить на метод вершин [1–4] и метод, основанный на лемме Фаркаша [4–6]. Первый из этих методов требует поиска параметризованных вершин многогранников, что не всегда удобно на практике. Второй метод решает поставленную задачу только при конкретных значениях параметров алгоритма, и не позволяет переходить к распараллеливанию алгоритма при произвольных значениях внешних переменных. Оба подхода требуют при реализации удовлетворения большого, часто избыточного числа ограничений, выражаемых равенствами и неравенствами. Преодолению некоторых из указанных недостатков посвящены работы [7–12]. Исследования данной работы направлены на дальнейшее развитие метода Фаркаша: предлагаемый способ получения новых условий сохранения зависимостей применим в общем случае и при произвольных, заранее неизвестных значениях внешних переменных.

### Предварительные сведения

Пусть алгоритм задан аффинным гнездом циклов произвольной структуры вложенности. Для таких алгоритмов индексные выражения переменных и границы изменения параметров циклов являются аффинными функциями от параметров циклов и внешних переменных. Пусть в гнезде циклов имеется  $K$  операторов  $S_\beta$ . Область изменения параметров гнезда циклов для оператора  $S_\beta$  будем называть индексной областью и обозначать  $V_\beta$ . Обозначим  $n_\beta$  — число циклов, окружающих оператор  $S_\beta$ ,  $V_\beta \subset \mathbf{Z}^{n_\beta}$ .

Выполнение оператора  $S_\beta$  при конкретных значениях  $\beta$  и вектора параметров цикла  $J$  будем называть операцией и обозначать  $S_\beta(J)$ . Выполнение всех операций, зависящих от  $J$ , называется  $J$ -й итерацией.

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, 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  $S_\alpha(I)$  будем обозначать  $S_\alpha(I) \rightarrow S_\beta(J)$ .

Обозначим  $P = \{(\alpha, \beta) \mid \exists I \in V_\alpha, J \in V_\beta, S_\alpha(I) \rightarrow S_\beta(J)\}$ . Множество  $P$  определяет пары зависимых операторов. Для каждой пары  $(\alpha, \beta) \in P$  обозначим  $V_{\alpha, \beta} = \{J \in V_\beta \mid \exists S_\alpha(I) \rightarrow S_\beta(J)\}$ . Будем считать, что  $V_{\alpha, \beta}$  — выпуклый многогранник в пространстве  $\mathbf{Z}^{n_\beta}$ .

Функции  $\bar{\Phi}_{\alpha, \beta}: V_{\alpha, \beta} \rightarrow V_\alpha$  такие, что если  $S_\alpha(I) \rightarrow S_\beta(J)$ ,  $I \in V_\alpha$ ,  $J \in V_{\alpha, \beta} \subseteq V_\beta$ , то  $I = \bar{\Phi}_{\alpha, \beta}(J)$ , назовем функциями зависимостей. Будем предполагать, что функции зависимостей являются аффинными:

$$\bar{\Phi}_{\alpha,\beta}(J) = \Phi_{\alpha,\beta}J + \Psi_{\alpha,\beta}N - \varphi^{(\alpha,\beta)},$$

где  $J \in V_{\alpha,\beta}$ ,  $(\alpha,\beta) \in P$ ,  $N = (N_1, N_2, \dots, N_e)$  — вектор внешних переменных алгоритма,  $e$  — число таких переменных,  $\Phi_{\alpha,\beta} \in \mathbf{Z}^{n_\alpha \times n_\beta}$ ,  $\Psi_{\alpha,\beta} \in \mathbf{Z}^{n_\alpha \times e}$ ,  $\varphi^{(\alpha,\beta)} \in \mathbf{Z}^{n_\alpha}$ , матрицы  $\Phi_{\alpha,\beta}$ ,  $\Psi_{\alpha,\beta}$  и векторы  $\varphi^{(\alpha,\beta)}$  не зависят от внешних переменных.

Пусть функции  $t^{(\beta)}: V_\beta \rightarrow \mathbf{Z}$ ,  $1 \leq \beta \leq K$ , ставят в соответствие каждой операции  $S_\beta(J)$  алгоритма целое число  $t^{(\beta)}(J)$ . Пусть функции  $t^{(\beta)}$  являются таймирующими функциями, т.е.

$$t^{(\beta)}(J) \geq t^{(\alpha)}(\Phi_{\alpha,\beta}J + \Psi_{\alpha,\beta}N - \varphi^{(\alpha,\beta)}), J \in V_{\alpha,\beta}, (\alpha,\beta) \in P. \quad (1)$$

Условия (1) называются условиями сохранения зависимостей и означают, что если  $S_\alpha(I) \rightarrow S_\beta(J)$ .  $I = \Phi_{\alpha,\beta}J + \Psi_{\alpha,\beta}N - \varphi^{(\alpha,\beta)}$ , то операция  $S_\alpha(I)$  должна выполняться не позже операции  $S_\beta(J)$ . Будем предполагать, что функции  $t^{(\beta)}(J)$  являются аффинными:

$$t^{(\beta)}(J) = \tau^{(\beta)}J + b^{(\beta)}N + a_\beta,$$

где  $1 \leq \beta \leq K$ ,  $J \in V_\beta$ ,  $\tau^{(\beta)} \in \mathbf{Z}^{n_\beta}$ ,  $b^{(\beta)}$ ,  $N \in \mathbf{Z}^e$ ,  $a_\beta \in \mathbf{Z}$ . Предполагается, что  $\tau^{(\beta)}$ ,  $b^{(\beta)}$ ,  $a_\beta$  не зависят от  $N$ .

Таймирующие функции используются для распараллеливания и других преобразований алгоритмов [1–12]. Удовлетворение условий сохранения зависимостей является обязательным при получении функций таймирования.

Далее нам понадобится понятие эрмитовой нормальной формы матрицы.

Пусть  $D$  есть матрица с вещественными элементами,  $\rho$  — ранг матрицы  $D$ . Тогда элементарными строчными преобразованиями матрицу  $D$  можно свести к ее эрмитовой нормальной форме — к матрице  $H$  следующей структуры:

- первые  $\rho$  строк матрицы  $H$  ненулевые, все элементы остальных строк равны нулю;
- первый ненулевой элемент в  $i$ -й ( $i=1, 2, \dots, \rho$ ) строке матрицы  $H$  равен 1; пусть этот элемент входит в столбец с номером  $c_i$ ;
- $c_1 < c_2 < \dots < c_\rho$ ;
- в столбце с номером  $c_i$  все элементы, кроме входящего в  $i$ -ю строку, равны нулю.

## Постановка задачи

Введем обозначения:

$$\sigma_0 = 0, \sigma_j = \sum_{i=1}^j n_i, 1 \leq j \leq K, \sigma = \sigma_K + Ke + K;$$

$\tilde{\tau} = (\tau^{(1)}, \dots, \tau^{(K)}, b^{(1)}, \dots, b^{(K)}, a_1, \dots, a_K)$  — вектор размера  $\sigma$ , составленный из параметров функций  $t^{(\beta)}$ ;

$0^{i \times j}$  — нулевая матрица размера  $i \times j$ ;

$E^{(i)}$  — единичная матрица порядка  $i$ ;

$0^{(i)}$  — нулевой вектор-столбец размера  $i$ ;

$e_j^{(i)}$  — вектор размера  $i$ , у которого координата с номером  $j$  равна 1, а остальные координаты нулевые;

$$\Phi_{\alpha,\beta} = \begin{pmatrix} 0^{\sigma_{\beta-1} \times n_\beta} \\ E^{(n_\beta)} \\ 0^{(\sigma - \sigma_\beta) \times n_\beta} \end{pmatrix} - \begin{pmatrix} 0^{\sigma_{\alpha-1} \times n_\beta} \\ \Phi_{\alpha,\beta} \\ 0^{(\sigma - \sigma_\alpha) \times n_\beta} \end{pmatrix} \text{ — матрица размера } \sigma \times n_\beta;$$

$$\Psi_{\alpha,\beta} = \begin{pmatrix} 0^{(\sigma_K + (\beta-1)e) \times e} \\ E^{(e)} \\ 0^{((K-\beta)e + K) \times e} \end{pmatrix} - \begin{pmatrix} 0^{\sigma_{\alpha-1} \times e} \\ \Psi_{\alpha,\beta} \\ 0^{(\sigma - \sigma_{\alpha}) \times e} \end{pmatrix} - \begin{pmatrix} 0^{(\sigma_K + (\alpha-1)e) \times e} \\ E^{(e)} \\ 0^{((K-\alpha)e + K) \times e} \end{pmatrix} \text{ — матрица размера } \sigma \times e;$$

$$\varphi^{(\alpha,\beta)} = \begin{pmatrix} 0^{(\sigma_{\alpha-1})} \\ \varphi^{(\alpha,\beta)} \\ 0^{(\sigma - \sigma_{\alpha})} \end{pmatrix} + e_{\sigma-K+\beta}^{(\sigma)} - e_{\sigma-K+\alpha}^{(\sigma)} \text{ — вектор размера } \sigma,$$

Используя введенные обозначения, условия сохранения зависимостей (1) можно записать в виде [12]

$$\tilde{\tau}(\Phi_{\alpha,\beta} | \Psi_{\alpha,\beta})(J, N) + \tilde{\tau}\varphi^{(\alpha,\beta)} \geq 0, \quad J \in V_{\alpha,\beta}, \quad (\alpha, \beta) \in P, \quad (2)$$

где  $(\Phi_{\alpha,\beta} | \Psi_{\alpha,\beta})$  — матрица размера  $\sigma \times (n_{\beta} + e)$ , составленная из матриц  $\Phi_{\alpha,\beta}$  и  $\Psi_{\alpha,\beta}$ ,  $(J, N)$  — вектор размера  $n_{\beta} + e$ , составленный из вектора параметров цикла  $J$  и вектора внешних переменных  $N$ .

Основные трудности практического использования системы (2) связаны с необходимостью удовлетворения большого числа неравенств, количество которых зависит от размера областей  $V_{\alpha,\beta}$ . Нашей задачей является разработка алгоритма получения системы неравенств равносильной системе (2), но с существенно меньшим числом неравенств.

### Алгоритм получения условий сохранения зависимостей

Следующая процедура приводит для каждой пары  $(\alpha, \beta) \in P$  систему неравенств (2) к равносильной системе с числом неравенств, не зависящим от  $J$  и  $N$ .

1. Положить  $x = \tilde{\tau}(\Phi_{\alpha,\beta} | \Psi_{\alpha,\beta})$ ,  $w = (J, N)$ ,  $u = \tilde{\tau}\varphi^{(\alpha,\beta)}$ ,  $h = n_{\beta} + e$ ,  $V = \{(J, N) | J \in V_{\alpha,\beta}\}$ . Представить область  $V$  в виде  $V = \{w \in \mathbf{Z}^h | Aw + b \geq 0\}$ , где  $A \in \mathbf{Z}^{u \times h}$  — матрица,  $b \in \mathbf{Z}^h$  — вектор.

2. Привести матрицу  $A^T$  к эрмитовой нормальной форме:  $H = PA^T$ , где  $P$  — матрица, осуществляющая соответствующие строчные преобразования. Пусть  $c_1, c_2, \dots, c_h$  — номера столбцов матрицы  $H$ , в которых единственный ненулевой элемент равен единице.

3. Рассмотреть систему  $PA\lambda = Px$ ,  $\lambda_0 = u - \lambda b$ ,  $\lambda_0, \lambda \geq 0$ ,  $\lambda \in \mathbf{Z}^u$ ,  $\lambda_0 \in \mathbf{Z}$ . Выразить  $\lambda_{c_i}$ ,  $1 \leq i \leq h$ , через  $\lambda_j$ ,  $j \neq c_i$ , и  $x_k$ ,  $1 \leq k \leq h$ ; выразить  $\lambda_0$ , через  $u$ ,  $\lambda_j$ ,  $j \neq c_i$ , и  $x_k$ ,  $1 \leq k \leq h$ :  $\lambda_{c_i} = f_{c_i}(\lambda_j, x_k)$ ,  $\lambda_0 = f_0(u, \lambda_j, x_k)$ .

4. Последовательно исключить все  $\lambda_j$  из системы неравенств  $f_{c_i}(\lambda_j, x_k) \geq 0$ ,  $1 \leq i \leq h$ ,  $\lambda_j \geq 0$ ,  $j \neq c_i$ ,  $f_0(u, \lambda_j, x_k) \geq 0$ . Для исключения фиксированного  $\lambda_{j_0}$  выполнить следующее:

— взять все пары неравенств с коэффициентами при  $\lambda_{j_0}$  противоположных знаков и для каждой пары получить новое неравенство с исключенным  $\lambda_{j_0}$ ;

— добавить из исходной системы неравенств все неравенства, не содержащие  $\lambda_{j_0}$ .

5. В полученной в результате выполнения предыдущего шага системе неравенств заменить  $u$  на  $\tilde{\tau}\varphi^{(\alpha,\beta)}$ , а  $x_k$  заменить на  $\tilde{\tau}(\Phi_{\alpha,\beta} | \Psi_{\alpha,\beta})_k$ , где  $(\Phi_{\alpha,\beta} | \Psi_{\alpha,\beta})_k$  обозначает  $k$ -й столбец матрицы  $(\Phi_{\alpha,\beta} | \Psi_{\alpha,\beta})$ .

**Обоснование алгоритма.** В принятых обозначениях для фиксированных  $x$  и  $u$  условия (2) примут вид

$$xw + u \geq 0, \quad w \in V \subset \mathbf{Z}^h, \quad (3)$$

Система неравенств (3) справедлива тогда и только тогда, когда выполняются условия [4]

$$x = \lambda A, \quad u = \lambda_0 + \lambda b, \quad \lambda_0, \lambda \geq 0. \quad (4)$$

Предполагается, что в матрице  $A \in \mathbf{Z}^{u \times h}$  число строк больше числа столбцов,  $\text{rank} A = h$  (для практического использования интересен этот случай). Нас интересуют ограничения на  $x$  и  $u$ , поэтому  $\lambda_0$  и  $\lambda$  следует исключить. На шагах 2 и 3 происходит исключение  $\lambda_{c_0}, \lambda_{c_1}, \dots, \lambda_{c_n}$ . На шаге 4 применяется процедура Фурье-Моткина исключения из неравенств оставшихся  $\lambda_j$ .

Проиллюстрируем применение алгоритма на примере.

Рассмотрим основную часть алгоритма перемножения трех квадратных матриц  $A, B, D$  порядка  $N$ :

```

do i = 1, N
  do j = 1, N
    do k = 1, N
      S1: c(i,j) = c(i,j) + a(i,k) b(k,j)
    do i = 1, N
      do j = 1, N
        do k = 1, N
          S2: x(i,j) = x(i,j) + c(i,k) d(k,j)

```

Одна из зависимостей алгоритма является неоднородной и определяется функцией

$$\bar{\Phi}_{1,2}(i, j, k) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} i \\ j \\ k \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} N, \quad V_{1,2} = \{(i, j, k) \in \mathbf{Z}^3 \mid 1 \leq i, j, k \leq N\}.$$

Получим условия сохранения этой зависимости.

Имеем:  $K = 2, n_1 = n_2 = 3, e = 1, \sigma_1 = 3, \sigma_2 = 6, \sigma = 10$ ,

$$\tilde{\tau} = (\tau_1^{(1)}, \tau_2^{(1)}, \tau_3^{(1)}, \tau_1^{(2)}, \tau_2^{(2)}, \tau_3^{(2)}, b^{(1)}, b^{(2)}, a_1, a_2), \quad \Phi_{1,2} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \\ & E^{(3)} & \\ & 0^{4 \times 3} & \end{pmatrix}, \quad \Psi_{1,2} = \begin{pmatrix} 0^{(2)} \\ -1 \\ 0^{(3)} \\ -1 \\ 1 \\ 0^{(2)} \end{pmatrix}.$$

$$\varphi^{(1,2)} = -e_9^{(10)} + e_{10}^{(10)},$$

$$V = \{(i, j, k, N) \in \mathbf{Z}^4 \mid i-1 \geq 0, j-1 \geq 0, k-1 \geq 0, N-i \geq 0, N-j \geq 0, N-k \geq 0\},$$

$$A^T = \begin{pmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}, \quad b = (-1, -1, -1, 0, 0, 0).$$

Для получения нормальной эрмитовой формы матрицы  $A^T$  достаточно к первой строке прибавить четвертую. Выразим  $\lambda_1, \lambda_2, \lambda_3, \lambda_4$  через  $\lambda_5, \lambda_6$  и  $x_k, 1 \leq k \leq 4$ ; выразим  $\lambda_0$ , через  $u, \lambda_5, \lambda_6$  и  $x_k, 1 \leq k \leq 4$ . Придем к системе неравенств  $x_1 + x_4 - \lambda_5 - \lambda_6 \geq 0, x_2 + \lambda_5 \geq 0, x_3 + \lambda_6 \geq 0, x_4 - \lambda_5 - \lambda_6 \geq 0, \lambda_5 \geq 0, \lambda_6 \geq 0, u + x_1 + x_2 + x_3 + x_4 \geq 0$ . После исключения  $\lambda_5$  получим систему  $x_1 + x_2 + x_4 - \lambda_6 \geq 0, x_2 + x_4 - \lambda_6 \geq 0, x_1 + x_4 - \lambda_6 \geq 0, x_4 - \lambda_6 \geq 0, x_3 + \lambda_6 \geq 0, \lambda_6 \geq 0, u + x_1 + x_2 + x_3 + x_4 \geq 0$ . Исключив  $\lambda_6$  придем к системе  $x_1 + x_2 + x_3 + x_4 \geq 0, x_2 + x_3 + x_4 \geq 0, x_1 + x_3 + x_4 \geq 0, x_3 + x_4 \geq 0, x_1 + x_2 + x_4 \geq 0, x_2 + x_4 \geq 0, x_1 + x_4 \geq 0, x_4 \geq 0, u + x_1 + x_2 + x_3 + x_4 \geq 0$ . Осталось в этой системе неравенств заменить  $u$  на  $\tilde{\tau}^{(\alpha, \beta)}$ , а  $x_k$  заменить на  $\tilde{\tau}(\Phi_{\alpha, \beta} | \Psi_{\alpha, \beta})_k$ .

Эти условия совпадают с необходимыми и достаточными условиями сохранения зависимостей,

полученными в работе [12] другим, менее общим, но более приспособленным для автоматизации способом.

Работа выполнена при поддержке Фонда фундаментальных исследований Республики Беларусь.

#### Summary

Conditions for preservation of dependencies when paralleling of algorithms are obtained. Unlike the other methods which based on Farkash lemma, the method can be applied for arbitrary and unknown beforehand values of external variables

#### Литература

1. Quinton P., Dongen V., V. // J. of VLSI Signal Processing 1989. Vol. 1(2). P. 95-113.
2. Mauras C., Quinton P., Rajopadhye S., Sauter Y. Scheduling affine parameterized recurrences by means of variable dependent timing functions. In Kung S.Y., J. Swartzlander E.E., Fortes J.A.B., Przytula K.W. editors. // Application Specific Array Processors. P. 100-110. IEEE Computer Society Press, September 1990.
3. Воеводин В. В., Воеводин Вл. В. Параллельные вычисления. — СПб.: БХВ-Петербург, 2002. - 600с.
4. Balev S., Quinton P., Rajopadhye S., Risset T. Linear programming models for scheduling systems of affine recurrence equations — a comparative study. 10th Symposium on parallel algorithms and architectures (SPAA), 1998.
5. Feautrier P. // Int. J. of Parallel Programming. 1992. Vol. 21, № 5,6. P. 313–348, P. 389–420.
6. Lim A. W., Lam M. S. // Parallel Computing. 1998. Vol.24, № 3-4. P. 445-475.
7. Darté A., Vivien F. // Int. J. Parallel Programming. 1997. Vol. 25, № 6. P. 447-496.
8. Darté 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. Vol. 105. P. 147–183.
9. Адуцкевич Е. В., Лиходед Н. А. // Весці НАН Беларусі. Сер. фіз.-мат. навук. 2004. № 3. С. 100-105.
10. Адуцкевич Е. В., Лиходед Н. А., Соболевский П. И. // Весці НАН Беларусі. Сер. фіз.-мат. навук. 2005. № 3. С. 105-111.
11. Адуцкевич Е. В. // Весці НАН Беларусі. Сер. фіз.-мат. навук. 2005. № 4. С. 107-112.
12. Адуцкевич Е. В., Лиходед Н. А. // Программирование. 2006. № 3. С. 54-65.

Likhoded N.A.

Preservation of dependencies between operations when paralleling the algorithms.

УДК 519.6+681.3.012

Л и х о д е д Н. А. Сохранение зависимостей между операциями при распараллеливании алгоритмов // Весці НАН Беларусі. Сер. фіз.-мат. навук. ???? №

Разработан метод получения условий сохранения зависимостей распараллеливаемого алгоритма. В отличие от других способов, основанных на лемме Фаркаша, метод применим при произвольных, заранее неизвестных значениях внешних переменных.

Библиогр. — 12 назв.