

Шаг 3. Осуществляется проход по ним до выхода за пределы окрестности (переход к шагу 7) либо нахождения другой минуции. Также в окрестности ищутся минуции, не связанные с точкой M_0 ;

Шаг 4. Если найденная минуция M_1 , связанная с M_0 , является окончанием, то делается вывод о ложности образования (переход к шагу 6).

Шаг 5. Иначе, для M_1 проводится аналогичное исследование (шаги 2 – 4) с сохранением направлений линий и на основании полученной информации делается вывод об истинности сочетания минуций (переход к шагу 7), типе ложного образования (переход к шагу 6) либо неизвестном сочетании минуций (переход к шагу 8);

Шаг 6. Ложное образование подвергается логической коррекции, а минуции исключаются из результирующего набора;

Шаг 7. Делается вывод об истинности образования, и минуция добавляется в результирующий набор;

Шаг 8. Неизвестное сочетание минуций удаляется из результирующего набора без коррекции изображения. Аналогичным образом обрабатываются и крайние точки изображения.

В результате работы алгоритма устраняются ложные образования известного вида и формируется результирующий набор локальных характеристик, который может быть использован для хранения и сопоставления информации о дактилоскопическом изображении.

Литература

1. Методы, алгоритмы и программное обеспечение гибких информационных технологий для автоматизированных идентификационных систем: сб. научн. ст. – Мн.: БГУ, 1999. – 182 с.
2. *Davide Maltoni, Dario Maio, Anil K. Jain, Salil Prabhakar*. Handbook of fingerprint recognition – New York: Springer-Verlag, 2003. – 361 p.
3. *Гонсалес Р., Вудс Р.* Цифровая обработка изображений /Пер. с англ. – М.: Техносфера, 2005. – 1072 с.

АЛГОРИТМЫ И ПРОГРАММЫ ОБОБЩЕННОГО СЛИЯНИЯ МНОГОМЕРНЫХ ЦИКЛОВ

А. Э. Огейко

ВВЕДЕНИЕ

Слияние циклов – это такое преобразование программы, которое дает возможность объединить несколько циклов в один. Слияние является одним из наиболее широко используемых оптимизирующих преобразований циклов. Применяется в процедурах сжатия и обобщенного сжатия массивов, повышения зернистости вычислений (тайлинг, блокинг),

улучшения локальности (обеспечения повторного использования) данных. Слияние циклов является также полезным предварительным шагом для возможности применения методов распараллеливания программ.

Слияние циклов не всегда допустимо, так как оно может изменить порядок выполнения зависимых операций. Поэтому методы получения слияния должны учитывать информацию о зависимостях. Дополнительные сложности возникают при слиянии циклов с разными границами. Особенно трудной и малоизученной является задача обобщенного слияния – преобразования циклов произвольной структуры вложенности, в результате которого происходит слияние циклов, не обязательно всех, на различных уровнях вложенности. При обобщенном слиянии многомерных циклов могут помимо слияния происходить и другие преобразования одномерных циклов.

В данной работе разрабатывается метод получения векторных таймирующих функций, задающих обобщенное слияние циклов. Метод позволяет параметрически учитывать внешние переменные циклов, допускает преобразования циклов (перестановку, слияние, сдвиг, обращение), приводящие к сравнительно несложной генерации кода.

ФУНКЦИИ, ЗАДАЮЩИЕ ОБОБЩЕННОЕ СЛИЯНИЕ ЦИКЛОВ

Пусть алгоритм задан многомерным циклом произвольной структуры вложенности. Будем считать, что границы изменения параметров циклов задаются неоднородными (в общем случае) формами, линейными по совокупности параметров циклов и внешних переменных. Пусть в многомерном цикле имеется K операторов S_β . Обозначим: V_β – область изменения параметров циклов для оператора S_β , n_β – число циклов, окружающих оператор S_β ; $N \in Z^e$ – вектор внешних переменных алгоритма, e – число внешних переменных. Выполнение оператора S_β при конкретных значениях β и вектора параметров цикла J будем называть операцией и обозначать $S_\beta(J)$.

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 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 $S_\alpha(I) \rightarrow S_\beta(J)$.

Пусть векторная функция $\bar{t}^{(\beta)}(J): V_\beta \rightarrow Z^n$, $1 \leq \beta \leq K$ ставит в соответствие каждой операции $S_\beta(J)$ вектор $(t_1^{(\beta)}, \dots, t_n^{(\beta)})$, $n = \max_{1 \leq \beta \leq K} n_\beta$. Будем полагать, что функции $t_\xi^{(\beta)}$ являются аффинными функциями (для фиксированных β и ξ) вида

$$t_\xi^{(\beta)}(J) = \tau^{(\beta, \xi)} J + b^{(\beta, \xi)} N + a_{\beta, \xi}, \quad J \in V_\beta, \quad \tau^{(\beta, \xi)} \in Z^{n_\beta}, \quad b^{(\beta, \xi)}, \quad N \in Z^e. \quad (1)$$

Функции $\bar{t}^{(1)}, \bar{t}^{(2)}, \dots, \bar{t}^{(K)}$ называются векторными таймирующими функциями, если

$$\bar{t}^{(\beta)}(J) \geq_{lex} \bar{t}^{(\alpha)}(I), \quad J \in V_\beta, \quad I \in V_\alpha, \quad S_\alpha \rightarrow S_\beta, \quad \text{rank } T^{(\beta)} = n_\beta, \quad (2)$$

где матрица $T^{(\beta)}$ состоит из векторов $\tau^{(\beta, \xi)}$.

Условия (2) называются условиями сохранения зависимостей. В случае строгого неравенства в (2), условия называются строгими условиями сохранения зависимостей. Набор векторных таймирующих функций $\bar{t} = \{\bar{t}^{(1)}, \bar{t}^{(2)}, \dots, \bar{t}^{(K)}\}$ называется многомерным таймированием. Этот набор задает преобразование циклов. Это значит, что операция $S_\beta(J)$, выполняемая на итерации J исходного гнезда циклов, будет выполняться на итерации $\bar{t}^{(\beta)}(J)$ преобразованного гнезда циклов. Таким образом, компоненты вектора $\bar{t}^{(\beta)}$ интерпретируются как параметры преобразованного гнезда циклов для оператора S_β : $\bar{t}_1^{(\beta)}$ – параметр самого внешнего цикла, $\bar{t}_n^{(\beta)}$ – параметр самого внутреннего цикла.

Далее будем считать, что только одна координата фиксированного вектора $\tau^{(\beta, \xi)}$ может быть ненулевой, причем она равна 1 или -1. Тем самым мы ограничимся преобразованиями, сравнительно несложными для генерации кода – слияние, перестановка, сдвиг, обращение циклов.

Пусть некоторые операторы S_α и S_β окружены после преобразования циклами уровня вложенности ξ , $\xi+1$, причем цикл j_ξ является общим (параметр цикла уровня вложенности ξ будем обозначать j_ξ , а сам цикл будем называть циклом j_ξ); тогда цикл $j_{\xi+1}$ является общим, если области значений $t_{\xi+1}^{(\alpha)}(J)$, $J \in V_\alpha$ и $t_{\xi+1}^{(\beta)}(J)$, $J \in V_\beta$ имеют пересечение; пересечение должно быть существенным, иначе рассматривается вырожденный случай. В частности если пределы изменения областей V_α и V_β примерно одинаковы, то $b^{(\alpha, \xi+1)} = b^{(\beta, \xi+1)}$.

Введем следующие обозначения:

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

$\tilde{\tau} = (\tau^{(1)}, \dots, \tau^{(K)}, b^{(1)}, \dots, b^{(K)}, \dots, a_1, \dots, a_K)$ – вектор размера σ .

Каждую зависимость между операторами S_α и S_β можно характеризовать некоторой матрицей $D_{\alpha,\beta}^{(0)}$, $D_{\alpha,\beta}^{(0)} = (D_{\alpha,\beta}^\varphi, D_{\alpha,\beta})$.

Условия сохранения зависимостей можно представить в виде

$$\tilde{\tau} D_{\alpha,\beta}^\varphi \geq 0, \tilde{\tau} D_{\alpha,\beta} \geq 0. \quad (3)$$

Если первое неравенство является строгим, то выполняется строгое условие сохранения зависимостей.

Рассмотрим матрицу $(E | D^{(0)})$, где E единичная матрица порядка σ , $D^{(0)} = (D^\varphi | D)$, матрицы D^φ и D составлены из матриц $D_{\alpha,\beta}^\varphi$ и $D_{\alpha,\beta}$ соответственно.

Разобьем первые σ_K строк матрицы $(E | D^{(0)})$ на K множеств $R(1), \dots, R(K)$: $R(1)$ составляют n_1 первых строк матрицы $(E | D^{(0)})$, $R(2)$ – следующие n_2 строк, и т.д., $R(K)$ составляют n_K последних строк. Разобьем, кроме того, первые σ_K строк матрицы $(E | D^{(0)})$ на n множеств $r(1), \dots, r(n)$ согласно следующему требованию: n_β элементов каждого множества $R(\beta)$ распределяются в n_β разных множеств $r(i)$. Таким образом, каждое множество $r(i)$ содержит хотя бы одну строку, любое множество $r(i)$ может включать не более одной строки из одного множества $R(j)$.

Составим матрицу, первая строка которой есть сумма строк матрицы $(E | D^{(0)})$ из множества $r(i)$, а остальные строки – это $Ke + K$ последних строк матрицы $(E | D^{(0)})$. Обозначим эту матрицу $(E(i) | D(i))$. Пусть к матрице $(E(i) | D(i))$ применены некоторые элементарные строчные преобразования. Обозначим полученную матрицу $(P(i) | B(i))$.

Пусть выделены $\xi - 1$ строк разных матриц $(P(i_j) | B(i_j))$, $1 \leq j \leq \xi - 1, \xi \geq 2$. Строку матрицы $P(i_\xi)$, $i_\xi \neq i_j, 1 \leq j \leq \xi - 1$, назовем t_ξ -строкой, если выполняются следующие условия: 1) среди первых σ_K элементов строки имеется ненулевой элемент; 2) если в соответствующую

щей строке матрицы $B(i_\xi)$ на месте какого-либо элемента матрицы $D_{(\alpha,\beta)}^{(0)}$ стоит отрицательный элемент, то хотя бы в одной из выбранных строк матриц $B(i_j)$ в столбцах, соответствующих матрице $D_{\alpha,\beta}^\Phi$, находятся только положительные элементы, а в столбцах, соответствующих матрице $D_{\alpha,\beta}$, – только неотрицательные элементы.

Теорема. Векторная функция, ξ -я компонента которой порождается t_ξ -строкой, является векторной таймирующей функцией.

Для каждой t_ξ -строки матрицы $(P(i) | B(i))$ можно указать зависимости, для которых выполняется строгое условие сохранения зависимостей, т.е. в столбцах, соответствующих матрице $D_{\alpha,\beta}^\Phi$, находятся только положительные элементы, а в столбцах, соответствующих матрице $D_{\alpha,\beta}$, находятся только неотрицательные элементы. Эти зависимости в компонентах векторной таймирующей функции с номерами, большими чем ξ , можно не учитывать, т. к. они удовлетворяются на ранних итерациях.

АЛГОРИТМ

Входные данные: Матрица $D^{(0)} = (D^\Phi | D)$, информация о том, какие столбцы матрицы $D^{(0)}$ соответствуют каждой фиксированной паре (α, β) .

Выходные данные: t_ξ -строки матрицы $(P(i) | B(i))$, а также для каждой такой строки – зависимости, для которых выполняется строгое условие сохранения зависимостей, если использовать эту строку при формировании таймирующих функций.

Алгоритм состоит из n ($n = \max_{1 \leq \beta \leq K} n_\beta$) шагов. На каждом шаге определяется одна компонента векторной таймирующей функции, задающей обобщенное слияние многомерных циклов. Содержание i -го шага алгоритма:

1. Составить матрицу, одна строка которой есть сумма строк матрицы $(E | D^{(i)})$ с номерами из множества $r(i)$, а остальные строки – это $Ke + K$ последних строк матрицы $(E | D^{(i)})$. Обозначим эту матрицу $(E(i) | D(i))$.

2. К матрице $(E(i) | D(i))$ применить алгоритм нахождения максимального числа линейно-независимых решений системы неравенств (3) [1]. Обозначим полученную матрицу $(P(i) | B(i))$.

3. С помощью строк матрицы $(P(i) | B(i))$, таких, что в позициях, соответствующих матрице $B(i)$, а также в первых σ_K столбцах матрицы $P(i)$

находятся только нулевые элементы, получить в остальных строках матрицы $(P(i)|B(i))$ как можно больше нулевых элементов в столбцах с номерами с $\sigma_K + 1$ по $\sigma_K + Ke$, а затем в столбцах с номерами с $\sigma_K + Ke + 1$ по $\sigma_K + Ke + K$.

4. Среди строк матрицы $(P(i)|B(i))$ с ненулевыми элементами в первых σ_K столбцах, выбирается строка, в которой наибольшее число одинаковых элементов в столбцах с номерами с $\sigma_K + 1$ по $\sigma_K + Ke$. Если такая строка не одна, то из них выбирается строка, в которой наибольшее число одинаковых элементов в столбцах с номерами с $\sigma_K + Ke + 1$ по $\sigma_K + Ke + K$.

5. В конце каждого шага из матриц $D^{(i)} = (D^0 | D)$ удаляются столбцы соответствующие зависимостям (α, β) , для которых выполняется строгое условие сохранения зависимостей.

Литература

1. Lim A.W., Lam M.S. Maximizing parallelism and minimizing synchronization with affine partitions // Parallel Computing. 1998. V. 24. – № 3, 4. P. 445–475.

АЛГОРИТМЫ АНАЛИЗА И ПРОГНОЗИРОВАНИЯ ВОЛАТИЛЬНОСТИ ФИНАНСОВЫХ РЫНКОВ НА ОСНОВЕ ЭКОНОМЕТРИЧЕСКИХ МОДЕЛЕЙ

А. А. Петрушко

С финансовыми инвестициями связаны различные виды рисков. Наибольший интерес для различных типов финансовых активов представляет рыночный риск, обусловленный возможным изменением рыночных котировок активов и изменением процентных ставок [1]. Для его измерения в настоящее время широко используется методология *Value-at-Risk* (*VaR*) [2].

Пусть $r \in \mathbb{R}^1$ – случайная величина, соответствующая доходности финансового актива или портфеля активов за один период владения. В соответствии с [2] *VaR* – это величина убытков, связанных с владением финансовым активом в течение рассматриваемого периода времени, которая с заданной вероятностью $1 - p$ не будет превышена, т. е. имеет место:

$$P(r \leq -VaR) = p. \quad (1)$$

На основании (1) для заданного значения $1 - p$ получаем: