Р. А.Осмонов*

МЕТОД РАСПАРАЛЛЕЛИВАНИЯ АЛГОРИТМОВ УНИМОДУЛЯРНЫМИ ПРЕОБРАЗОВАНИЯМИ

1. ВВЕДЕНИЕ

Применение оптимизирующих преобразований направлено на повышение рабочих характеристик программы. Сохранение смысла программы, т.е. эквивалентность исходной программе, является необходимым условием применения оптимизирующих преобразований.

Система оптимизирующих преобразований — это набор преобразований в совокупности со стратегией их применения.

Наибольший эффект от оптимизации достигается при применении ее к участкам повторяемости, таким как циклы, определяющие время работы программы.

В качестве программной модели берется гнездо из двух циклов с постоянными границами цикла. Нам дан фиксированный набор экземпляров тела цикла, которые следует выполнять последовательно в определенном порядке. Целью оптимизационных преобразований является определение группы экземпляров, которые могут быть распараллелены.

Метод волнового фронта (Wavefront Method) используется для выполнения гнезда цикла на параллельных и векторных компьютерах в том случае, когда ни один из циклов не может быть выполнен в векторной форме. Метод волнового фронта создает «волну», которая проходит через пространство итераций. Все итерации на линии волнового фронта исполняются параллельно. Этот метод выявляет максимум параллелизма, при этом сохраняя все отношения зависимости по данным [1].

* rafhat_iis@gorodok.net

Работа выполнена при финансовой поддержке Министерства образования РФ (научная программа «Университеты России», грант № УР.04.01.201).и Российского фонда фундаментальных исследований (грант № 04-07-90441).

2. ОПРЕДЕЛЕНИЕ УНИМОДУЛЯРНЫХ ПРЕОБРАЗОВАНИЙ

В случае с целочисленными матрицами операции сложения, вычитания, умножения, скалярного умножения на целое число и транспонирования всегда приводят к целочисленной матрице. Однако вычисление обратной матрицы для несингулярной матрицы не всегда приводит к целочисленной матрице. Важным же свойством унимодулярной матрицы является то, что обратная к ней — также целочисленна и унимодулярна [2].

Определение 1. Квадратная целочисленная матрица U — *унимодулярна*, если $|det(U)| = 1$.

Важный момент использования унимодулярных матриц состоит в том, что, переходя к новым индексным переменным в цикле, мы получаем целочисленный вектор тогда, когда исходный вектор индексных переменных целочисленный, и — наоборот. Преобразованный цикл будет состоять из тех же экземпляров операторов, но экземпляры в новом цикле будут исполняться в порядке возрастания новых индексных переменных.

Унимодулярные преобразования входят как в класс *оптимизирующих преобразований*, так и в класс *реструктурирующих преобразований* — преобразований, увеличивающих степень параллелизма в программе за счет изменения структуры программы [3].

Другим классом преобразований программ являются *несингулярные преобразования,* включающие перестановку, скашивание, обращение и масштабирование цикла. Несингулярные матрицы включают унимодулярные матрицы как частный случай. Преимуществом использования несингулярных матриц является то, что их проще сгенерировать и работать с ними, чем с унимодулярными матрицами.

Сгенерированная унимодулярная матрица может представлять собой как явное преобразование, так и совокупность преобразований в силу данных ей свойств. В случае, когда $det(U)$ элементарной унимодулярной матрицы равен –1, то она соответствует преобразованию — обращению цикла.

Определение 2. Между итерациями в гнезде существует зависимость в том случае, если есть итерации $S(i_1, i_2)$ и $S(j_1, j_2)$, такие что:

- итерация $S(i_1, i_2)$ выполняется раньше $S(j_1, j_2)$, экземпляр $H(i_1, i_2)$ исполняется до экземпляра $H(j_1, j_2)$ тогда и только тогда, когда $(i_1, i_2) < (j_1, j_2)$;
- обе итерации обращаются к одной и той же ячейке памяти, и по крайней мере, одно из обращений есть запись;

- между итерациями в ячейку памяти не заносится никакая информация [4].

Определение 3. *Индексный вектор* гнезда — это упорядоченная пара (I_1, I_2).

Определение 4. *Пространство итераций* гнезда — это набор всех возможных значений (I_1, I_2), т.е. множество точек с целочисленными координатами.

Определение 5. Если существуют две итерации (i_1, i_2) и (j_1, j_2), удовлетворяющие вышеприведенным условиям, то говорят, что в гнезде существует зависимость с *дистанционным вектором* (D_1, D_2), определяющимся как $(D_1, D_2) = (j_1 - i_1, j_2 - i_2)$.

Определение 6. В гнезде существует зависимость с *вектором направления* (s_1, s_2), если есть зависимость с любым дистанционным вектором (D_1, D_2), где $s_1 = sig(D_1)$ и $s_2 = sig(D_2)$. Обычно удобно говорить, что гнездо имеет определенный дистанционный вектор зависимости или вектор направления.

Определение 7. *Зависимость на уровне k* — это зависимость с вектором направления соответствующего вида:

- *зависимость на уровне* 1, в случае вектора направления вида $(1, s_2)$;
- *зависимость на уровне* 2, в случае вектора направления (0, 1).

Структура унимодулярного преобразования включает в себя три базовых преобразования: перестановка, скос и обращение цикла.

Перестановка циклов в гнезде — это метод выявления параллелизма, являющийся также мощным средством векторной оптимизации. Применение преобразования перестановки циклов к гнезду из двух циклов соответствует изменению порядка обхода двумерного пространства итераций, а так как между итерациями могут существовать зависимости по данным, то перестановка циклов легальна, если не будут нарушены эти зависимости [4].

Скашивание цикла — это модификация формы итерационного пространства цикла, т.е. преобразование границ цикла. Применение скашивания цикла может изменять векторы направлений, предотвращающие перестановку циклов. Цикл может быть скошен фактором различной величины. Скашивание цикла в комбинации с перестановкой цикла порождает метод волнового фронта.

Обращение цикла меняет первоначальное направление обхода циклом пространства итераций на противоположное. Обращение приводит к изменению заголовка цикла с уменьшением итерационной переменной до значения нижней границы.

3. ПРОЦЕДУРА ПОСТРОЕНИЯ УНИМОДУЛЯРНЫХ ПРЕОБРАЗОВАНИЙ

Если взять в качестве элементарной матрицы единичную матрицу, также являющуюся унимодулярной, то можно получить элементарные матрицы унимодулярных преобразований. Существуют следующие классы унимодулярных матриц размерностью $2{\times}2$, которые непосредственно используются в преобразованиях:

– $\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}$ и $\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ — обращающие матрицы;

– $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ — перестановочная матрица;

– $\begin{bmatrix} 1 & p \\ 0 & 1 \end{bmatrix}$ — множество верхних скашивающих матриц и $\begin{bmatrix} 1 & 0 \\ p & 1 \end{bmatrix}$ — множество нижних скашивающих матриц, где p — целое число, называемое фактором искажения.

Унимодулярная матрица может быть выражена как произведение обращающей, перестановочной и верхней скашивающей матриц.

Определение 8. Гнездом циклов является множество циклов, вложенных один в другой. Гнездо называется *совершенным*, если тело каждого следующего цикла, отличного от самого внутреннего, состоит только из следующего цикла гнезда [3].

Тело цикла содержит ряд экземпляров операторов $H(I_1, I_2)$, которые исполняются в направлении увеличения индексных переменных (I_1, I_2). Каждый такой экземпляр при преобразовании описывается уникальным значением новых переменных (K_1, K_2). Преобразование двойного цикла (L_1, L_2) под действием унимодулярной матрицы U приводит к новому циклу, который состоит из тех же экземпляров H, как в (L_1, L_2), но где экземпляры исполняются в порядке возрастания (K_1, K_2).

Если взять в качестве U элементарную матрицу, то получим специальное унимодулярное преобразование. Преобразование $(L_1, L_2) \to (L_1^U, L_2^U)$ называется:

- обращением цикла, если U — обращающая матрица;
- перестановкой цикла, если U — перестановочная матрица;
- скашиванием цикла, если U — скашивающая матрица.

Скашивание внутреннего цикла внешним получается, если U — верхняя скашивающая матрица, нижняя скашивающая матрица U вызывает скашивание внешнего цикла внутренним циклом.

Условием распараллеливания может служить наличие зависимости на уровне k.

Определение 9. Внешний цикл может быть распараллелен тогда и только тогда, когда нет зависимости на уровне 1. Внутренний цикл может быть распараллелен тогда и только тогда, когда нет зависимости на уровне 2.

Если ни один из этих случаев не выполняется, тогда необходимо переупорядочить экземпляры путем замены индексных переменных. При наличии унимодулярной матрицы, позволяющей перейти к новым индексным переменным и переупорядочить экземпляры тела цикла, можно сформулировать следующие теоремы.

Теорема 1. Пусть дана 2×2 целочисленная унимодулярная матрица U такая, что преобразованное гнездо двойного цикла (L_1^U, L_2^U) эквивалентно исходному гнезду (L_1, L_2). Внешний цикл $L^U{}_1$ может быть исполнен параллельно тогда и только тогда, когда есть фиксированный целочисленный вектор $(a_1, a_2) \succ 0$ такой, что каждый дистанционный вектор зависимости в (L_1, L_2) имеет вид $\alpha(a_1, a_2)$, где α — положительное целое число.

Доказательство. Предположим, что существует целочисленный вектор $(a_1, a_2) \succ 0$ такой, что любой дистанционный вектор зависимости в (L_1, L_2) имеет вид $\alpha(a_1, a_2)$, где α — положительное целое число. Пусть $g = НОД(a_1, a_2)$. Тогда $g \succ 0$, так как a_1 и a_2 не могут быть оба нулевыми. Определим целочисленную матрицу 2×2

$$U = \begin{bmatrix} u_{11} & u_{12} \\ u_{21} & u_{22} \end{bmatrix},$$

с $u_{11} = a_2/g$, $u_{21} = -a_1/g$ и (u_{12}, u_{22}) равно любой паре целых чисел, для которых $a_1u_{12} + a_2u_{22} = g$.

Эта матрица унимодулярная, так как

$$det(U) = (u_{11}u_{22} - u_{21}u_{12}) = (a_2u_{22} + a_1u_{12}) / g = 1.$$

Для каждого дистанционного вектора $(D_1, D_2) = \alpha(a_1, a_2)$ в (L_1, L_2) мы имеем

$$(D_1, D_2)*U = \alpha(a_1, a_2)*U = \alpha(a_1u_{11} + a_2u_{21}, a_1u_{12} + a_2u_{22}) = (0, \alpha g) >_2 0.$$

Данное выражение показывает отсутствие зависимости на уровне 1, следовательно, внешний цикл может быть распараллелен.

Теорема 2. Всегда существует унимодулярная матрица U 2×2 такая, что преобразованное гнездо $(L^U{}_1, L^U{}_2)$ эквивалентно исходному гнезду (L_1, L_2), и внутренний цикл $L^U{}_2$ может быть исполнен параллельно.

Доказательство. Если есть дистанционный вектор зависимости (D_1, D_2) в (L_1, L_2) такой, что $D_1 > 0$ и $D_2<0$, тогда обозначим через $\mu = \lfloor max\{-D_2/D_1\} + 1\rfloor$, где максимум берется из всех дистанционных векторов (D_1, D_2) в исходном гнезде таких, что $D_1 > 0$ и $D_2<0$. Иначе берем $\mu = 1$.

Возьмем унимодулярную матрицу

$$U = \begin{bmatrix} \mu & 1 \\ 1 & 0 \end{bmatrix}.$$

Рассмотрим знак выражения $(\mu D_1 + D_2)$ для положительного дистанционного вектора (D_1, D_2) в (L_1, L_2). Так как $\mu > 0$, знак очевидно положителен, если $D_1 > 0$ и $D_2 > 0$ или если $D_1 = 0$ и $D_2 > 0$. Знак также положителен, если $D_1 > 0$ и $D_2<0$, так как $\mu > (-D_2/D_1)$. Тогда следует результат

$$(D_1, D_2)*U = (\mu D_1 + D_2, D_1) >_1 0.$$

Отсутствие зависимости для каждого дистанционного вектора на уровне 2, позволяет распараллелить внутренний цикл.

Чтобы проверить корректность применения унимодулярного преобразования, необходимо знать дистанционные вектора исходного цикла, поскольку информации о векторах направления недостаточно в этом случае. Однако еще остается вопрос о выборе корректных преобразований, способных привести к распараллеливанию цикла, и о последовательности их применения.

Приведем некоторые свойства преобразований, позволяющие сделать такой выбор.

Перестановка циклов используется во многих случаях, например, для векторизации внутреннего цикла, если внешний цикл должен исполняться последовательно.

Применение скашивания цикла всегда корректно, поскольку оно не влияет на численные результаты программы. Скашивание также не меняет порядок выполнения итераций, итерации в скошенном пространстве итераций исполняются в том же самом порядке, как и соответствующие итерации в исходном пространстве итераций. Однако это преобразование изменяет вектора направлений: при скашивании внутреннего цикла элементы дистанционных векторов изменяются, причем положительный вектор всегда остается положительным.

Пространство итераций представляется на плоскости в виде точек, координаты которых определяют соответствующие итерации. Так, обращение внешнего цикла есть отражение индексного пространства относительно вертикальной оси координат, а обращение внутреннего цикла — отражение индексного пространства относительно горизонтальной оси координат. Значения элементов дистанционных векторов остаются неизменными по абсолютному значению, меняются только вектора направлений [5].

Цикл в преобразованном гнезде может быть исполнен параллельно тогда и только тогда, когда не существует дистанционного вектора в исходном цикле, удовлетворяющего условию $dU >_k 0$, где k — уровень вложенности цикла. Унимодулярное преобразование гнезда цикла может быть выполнено путем последовательного применения конечной последовательности элементарных преобразований, соответствующих элементарным унимодулярным матрицам.

Задачей нижеприведенного алгоритма является распараллеливание внутреннего или внешнего цикла посредством получения унимодулярной матрицы. С помощью полученной унимодулярной матрицы можно получить преобразованный цикл и проверить эквивалентность исходному. В случае эквивалентности можно определить, какой из циклов в преобразованном гнезде может быть исполнен параллельно. Важно, чтобы интервал распараллеливаемого цикла был максимальным.

4. ЭКСПЕРИМЕНТЫ И РЕЗУЛЬТАТЫ

Для экспериментов выбирались совершенные гнезда циклов двойной вложенности. Исходный код программы представлен на языке С + +. Форма записи цикла представляется выражением

«$for\ (I_1 = n_1;\ I_1 < N_1;\ I_1 + +)$
$for(I_2 = n_2;\ I_2 < N_2;\ I_2 + +)$
$H(I_1, I_2);$»,

где i, j — индексные переменные, n_1, N_1, n_2, N_2 — постоянные границы цикла. Алгоритм состоит из четырех основных шагов: нахождение индексных переменных и границ исходного гнезда цикла; вычисление дистанционных векторов; определение векторов направлений и зависимостей на уровне; генерация унимодулярной матрицы для распараллеливания внешнего или внутреннего цикла.

Алгоритм направлен на выявление параллелизма внешнего или внутреннего цикла. Каждый из случаев предполагает наличие соответствующих дистанционных векторов, допускающих данное распараллеливание. Множество дистанционных векторов определяется по индексам массивов, входящих в тело цикла, и записывается в виде матрицы D. Для распараллеливания внешнего цикла достаточно наличие дистанционных векторов, для распараллеливания же внутреннего цикла необходимо определить вектора направлений по множеству дистанционных векторов. Зависимость на уровне k может быть определена как по множеству векторов направления, так и по множеству дистанционных векторов. Определение зависимости на уровне k может однозначно указать, какой из циклов может быть распараллелен. Однако, если стоит задача о распараллеливании цикла, первоначально не подлежащего параллельному исполнению, тогда мы прибегаем к системе преобразований.

Параллельное исполнение внешнего цикла проверяется возможностью представления всех дистанционных векторов гнезда цикла в виде $\alpha(a_1, a_2)$, где $\alpha > 0$ и $(a_1, a_2) > 0$. Затем находятся $НОД(a_1, a_2)$ и целые числа (x_0, y_0) такие, что $a_1x_0 + a_2y_0 = g$, по которым и вычисляются элементы унимодулярной матрицы. В случае, если матрица оказывается единичной, мы объявляем внешний цикл параллельным. В противном случае матрица включает в себя одно из преобразований перестановки, скашивания и обращения цикла или их комбинации. Прибегая к преобразованию гнезда цикла полученной унимодулярной матрицей, действуем на индексные переменные, таким способом переходя к новым переменным, при этом меняются и границы циклов. Таким образом, в случае возможности исполнения внешнего цикла параллельно, первый столбец матрицы $D * U$ получаем нулевым, согласно определению зависимости на уровне k.

Рассмотрим следующее гнездо

L$_1$: *for* ($I_1 = 5;\ I_1<100;\ I_1 + +$)
L$_2$: *for*($I_2 = 16;\ I_2<80;\ I_2 + +$)
S: $A[I_1][I_2] = A[I_1\text{-}2][I_2\text{-}4] + A[I_1 - 3][I_2 - 6]$;

Есть два дистанционных вектора в гнезде: (2, 4) и (3, 6). Данные вектора показывают зависимость на уровне 1, следовательно, внешний цикл не может быть распараллелен. Заметим, что (2, 4) = 2(1, 2) и (3,6) = 3(1, 2). Так как дистанционные вектора имеют вид $\alpha(a_1, a_2)$, где $\alpha > 0$ и $(a_1, a_2) > 0$, то существует унимодулярная матрица U такая, что после того как данное гнездо преобразовано матрицей U, внешний цикл в новом гнезде может быть распараллелен (Теорема 1). Найдем $НОД(a_1, a_2) = НОД(1, 2) = 1$, и два целых числа $u_{12} = 1$ и $u_{22} = 0$ такие, что $1 * u_{21} + 2 * u_{22} = 1$. Затем вычисляются целые числа $u_{11} = a_2/НОД(a_1, a_2) = 2$ и $u_{21} = -a_1/НОД(a_1, a_2) = -1$. Получаем матрицу

$$U = \begin{bmatrix} 2 & 1 \\ -1 & 0 \end{bmatrix}.$$

Преобразованный цикл:

for ($K_1 = -70;\ K_1<184;\ K_1 + +$)
for ($K_2 = \lceil max\{5,\ 8 + K_1/2\} \rceil;\ K_2<\lfloor min\{100,\ 40 + K_1/2\} \rfloor;\ K_2 + +$)
S:$A[K_2][-K_1 + 2*K_2] = A[K_2 - 2][-K_1 + 2*K_2 - 4] + A[K_2 - 3][-K_1 + 2*K_2 - 6]$;

Внешний цикл K_1 в новом гнезде может быть исполнен параллельно. Дистанционные вектора (2, 4) и (3, 6) преобразованы в дистанционные вектора (0, 2) и (0, 3).

Если не удается распараллелить внешний цикл, следующим шагом мы попытаемся произвести преобразования, приводящие к параллельному исполнению внутреннего цикла, так, чтобы число итераций внешнего цикла было минимизировано. Для этого находятся все дистанционные вектора и вектора направлений. Унимодулярная матрица генерируется путем выбора вектора направления из начального списка векторов $\{(1, 0)^T, (0, 1)^T, (0, -1)^T, (1, 1)^T, (\mu, 1)^T\}$ в качестве первого столбца унимодулярной матрицы. Условием выбора является $(D_1, D_2) * U >_1 0$. Если лучший выбор для первого столбца есть $(1, 0)^T$, то просто отмечаем внутренний цикл данного гнезда параллельным. В противном случае в качестве второго столбца берем вектор $(1, 0)^T$ и преобразуем исходное гнездо полученной унимодулярной матрицей.

Рассмотрим пример:

L$_1$: *for* $(I_1 = 5; I_1 < 100; I_1 + +)$
L$_2$: *for*$(I_2 = 5; I_2 < 100; I_2 + +)$
S: $A[I_1][I_2] = A[I_1]][I_2 - 1] + A[I_1 - 2][I_2 + 3] + A[I_1 - 3][I_2 + 7];$

Гнездо имеет дистанционные вектора $(0, 1)$, $(2, -3)$ и $(3, -7)$, которые соответствуют наличию зависимости как на уровне 1, так и на уровне 2. Преобразуем его в гнездо, где внутренний цикл может быть исполнен параллельно и количество итераций внешнего цикла минимизировано. Так как гнездо имеет вектор направления $(0, 1)$, то вектора $(1, 0)^T$ и $(0, -1)^T$ отбрасываются как возможные выборы для первого столбца U. Присутствие вектора направления $(1, -1)$ отбрасывает заодно вектора $(0, 1)^T$ и $(1, 1)^T$.

Затем подсчитываем для всех дистанционных векторов

$$\mu \leftarrow \lfloor max\{-D_2/D_1\} + 1\rfloor,$$

где максимум берется из всех дистанционных векторов (D_1, D_2) таких, что $D_1 > 0$ и $D_2 < 0$:

$$\mu = \lfloor max\{-(-3)/2, -(-7)/3\} + 1\rfloor = \lfloor 7/3 + 1\rfloor = 3$$

Добавляем вектор $(\mu, 1)^T$ в начальный список. После проведенных операций оказывается, что вектор $(\mu, 1)^T$ является лучшим выбором в качестве первого столбца унимодулярной матрицы.

Искомая матрица $U = \begin{bmatrix} 3 & 1 \\ 1 & 0 \end{bmatrix}$. Преобразованный цикл имеет вид:

for ($K_1 = 20; K_1 < 400; K_1 + +$)
  for ($K_2 = \lceil max\{5, (K_1 - 100)/3\} \rceil; K_2 < \lfloor min\{100, (K_1 - 5)/3\} \rfloor; K_2 + +$)
  S: $A[K_2][K_1 - 3 * K_2] = A[K_2][K_1 - 3 * K_2 - 1] + A[K_2 - 2][K_1 - 3 * K_2 + 3] +$
    $A[K_2 - 3][K_1 - 3 * K_2 + 7]$;

Внутренний цикл в новом гнезде может быть распараллелен.

Рассматривая сгенерированные матрицы, можно заметить, что при распараллеливании внешнего цикла применялись все три преобразования, а при преобразовании внутреннего — только скашивание и перестановка. Альтернатива параллельного исполнения внешнего или внутреннего цикла может исходить из условий задачи или архитектуры целевой машины. В общем, если параллелизм существует в гнезде, преобразования, рассмотренные в этой статье, найдут его.

5. ЗАКЛЮЧЕНИЕ

Данная работа обобщает простейшие случаи применения унимодулярных преобразований. Представленный метод приводит циклы к более удобному виду для применения и приспособления векторных конструкций. Вопрос о выполнении цикла в параллельной форме сводится к преобразованию, вызванному определенной унимодулярной матрицей, действующей на индексные переменные. Пригодность преобразований обращения, перестановки, скашивания цикла определяются природой проблемы. Исходной информацией о применении того или иного преобразования может явиться множество векторов направления. Однако множество всех дистанционных векторов в гнезде, очевидно, несет больше информации, чем множество всех векторов направления. В процессе преобразования не должна нарушаться структура зависимости данной программы, что и подразумевает принцип преобразования программы: значение дистанционного вектора должно оставаться положительным.

СПИСОК ЛИТЕРАТУРЫ

1. **Bacon D. F., Susan L.** Compiler transformations for High-Performance computing // ACM Computing Surveys. — Vol. 26, N 4. — 1994. — P. 345–420.
2. **Гантмахер Ф. Р.** Теория матриц. — М.: Наука, 1998.

3. **Евстигнеев В. А., Касьянов В. И.** Оптимизирующие преобразования в распараллеливающих компиляторах // Программирование. — №6. — 1996. — С. 12–26.
4. **Векторизация** программ: теория, методы, реализация. Сб. статей / Под ред. Г. Д. Чинина. — М.: Мир, 1991.
5. **Banerjee U.** Loop transformations for restructuring compilers. The foundations. — Kluwer Academic Publishers. — 1993.