

Разрезанная модель базовых блоков для оптимизации потоков команд

П.М. Довгалюк

E-mail: Pavel.Dovgaluk@nov.astrosoft.ru

Аннотация. Предлагаемая модель предназначена для описания потоков команд в базовых блоках. Данная модель ориентирована на задачи оптимизации потоков команд по скорости их исполнения. Подобные модели применяются с целью получения кратчайшего по времени расписания команд, поступающих на конвейер процессора.

1. Анализ существующих математических моделей вычислительных процессов в базовых блоках

Существует ряд моделей вычислительных процессов в базовых блоках. Наиболее распространенные из них используют для представления базового блока направленные ациклические графы [3], [4], [5].

Во всех распространенных графовых моделях базовых блоков множество вершин соответствует множеству команд, а наличие дуги между двумя вершинами соответствует наличию зависимости между соответствующими командами (дуга (v, u) показывает, что команда v должна быть выполнена раньше команды u).

Для того чтобы задать протяженность задержки между командами, в наиболее популярной модели, описанной в [3] и [5], используются числовые пометки ребер графа, соответствующие продолжительностям задержек — $D((v, u))$.

На Рис. 1 и Рис. 2 представлен пример содержимого базового блока и его традиционное представление с помощью графа.

```
mov a, b
add c, 1
mul a, c
mov d, c
mul a, d
```

Рис. 1. Пример содержимого базового блока

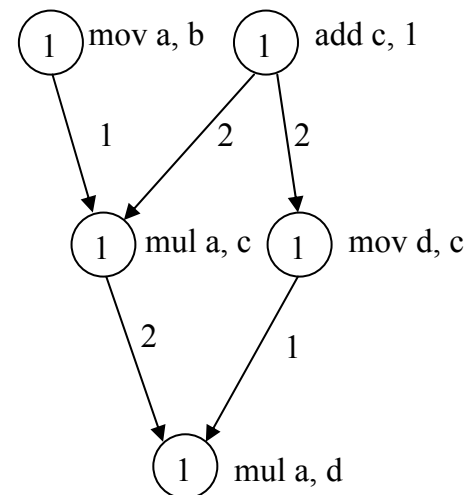


Рис. 2. Традиционное представление базового блока в виде графа

Корректным расписанием S для систем с одним конвейером называется функция $S : (V \rightarrow N \mid \forall (v, u) \in E \Rightarrow S(u) - S(v) > D((v, u)))$. Таким образом, $S(v)$ — позиция вершины v в результирующем расписании. В каждой позиции расписания может находиться либо одна инструкция, либо специальная команда NOP, которая не выполняет никаких действий.

```
mov a, b
add c, 1
mul a, c
nop
mov d, c
mul a, d
```

Рис. 3. Пример корректного расписания для базового блока

Существует множество моделей, построенных на основе описанной выше, отличающихся различными атрибутами вершин и дуг, в зависимости от особенностей архитектуры целевых машин.

В некоторых распространенных архитектурах, например Intel i860 [2], зависимости между командами могут быть ограничены по времени сверху. То есть вторая (зависящая) инструкция должна быть выполнена ровно через определенное количество тактов после первой, иначе результат выполнения первой команды будет утерян. Хотя такие виды зависимостей и описываются существующими моделями [1], [5], но эффективных алгоритмов построения расписания, создающих корректное расписание всегда, когда это возможно, для них не существует. Это объясняется тем, что такие зависимости вводятся в

модель с помощью специального атрибута связей. Данное расширение модели не позволяет эффективно использовать алгоритмы оптимизации, пригодные для моделей без этого атрибута [4], [5]. Эти алгоритмы в процессе работы могут заходить в тупик, генерируя некорректное расписание.

Также ни в одной из наиболее распространенных моделей не учитывается тот факт, что в большинстве архитектур различные команды занимают разное количество тактов конвейера. Например, для RISC-процессоров, где все команды кодируются одним машинным словом, некоторые команды, оперирующие большими константами, могут кодироваться двумя словами.

Кроме того, в традиционных моделях базовых блоков не учитываются команды перехода, имеющие неустраняемые задержки. Такие задержки допустимо заполнять полезными командами, если это не приводит к конфликтам по данным. Так как такое ограничение плохо вписывается в существующие модели, то для решения этой задачи используются специальные алгоритмы [3].

Таким образом, необходимо построить модель базовых блоков, позволяющую оптимизировать вычислительный процесс и в тех случаях, когда существуют жесткие ограничения сверху на продолжительность задержки между командами, а также, если команды кодируются неодинаковым количеством слов. Кроме того, новая модель должна позволять учитывать зависимости между командами из смежных базовых блоков для конвейерной оптимизации команд перехода.

2. Разреженная модель вычислительных процессов в базовых блоках

Традиционная графовая модель базовых блоков использует в качестве узлов отдельные команды целевой машины, из которых состоит базовый блок [5]. Такая модель не отражает загруженности конвейера непроизводительными вычислениями и не позволяет оперировать командами, размер которых больше одного машинного слова.

Поэтому предлагается видоизменить модель базовых блоков следующим образом: *в качестве узлов использовать операции, выполняемые конвейером за один такт.* Такими операциями могут быть – выборка кода команды, либо непроизводительная задержка, в течение которой на конвейер не поступает новых команд. Связывать же эти операции в граф предлагается с помощью связей двух видов: задающих относительный или абсолютный порядок операций, поступающих на конвейер.

Добавление узлов-задержек между командами делает граф более разреженным, что и послужило источником названия модели.

Разреженную модель базовых блоков можно математически описать с помощью следующего ациклического графа:

$$G = (V; E; s; e), \text{ где}$$

- V – множество узлов, соответствующих конвейерным операциям, формирующим базовый блок;
- $E \subset V \times V$ – множество связей, определяющих порядок поступления узлов-операций (команд и задержек) на конвейер процессора;
- $s \in V$ – стартовый (корневой) узел;
- $e \in V$ – последний узел в любом корректном расписании, построенном на основе данного графа.

Узлы в таком графе должны быть помечены соответственно их назначению – являются ли они выборками кода команды из памяти, либо непроизводительными задержками.

Для решения поставленных задач необходимо ввести два вида связей между вершинами.

Введем следующие определения:

Определение 1: Связь называется «жесткой», если две операции, которые она соединяет должны поступать на конвейер строго друг за другом (между ними не должно быть других операций).

Обозначим подмножество жестких связей как H .

Определение 2: Связь называется «гибкой», если она задает лишь относительный порядок поступления операций на конвейер (между ними на конвейер могут поступать другие операции).

Множество задержек введено для моделирования минимального времени между инструкциями, которое традиционно [3] представляется в виде числовой пометки дуги. В предлагаемой модели паузы между инструкциями заполняются с помощью непроизводительных операций.

Формальное описание графа приведенное ниже недостаточно точно описывает модель. Для того чтобы решать задачи оптимизации потока команд с помощью данной модели, граф должен удовлетворять следующим условиям:

- в графе существует только одна корневая вершина – s ;
- в графе существует только один лист – e ;
- граф является слабо связным;
- в графе не существует циклов, так как не могут существовать циклические зависимости по данным между инструкциями в одном базовом блоке.

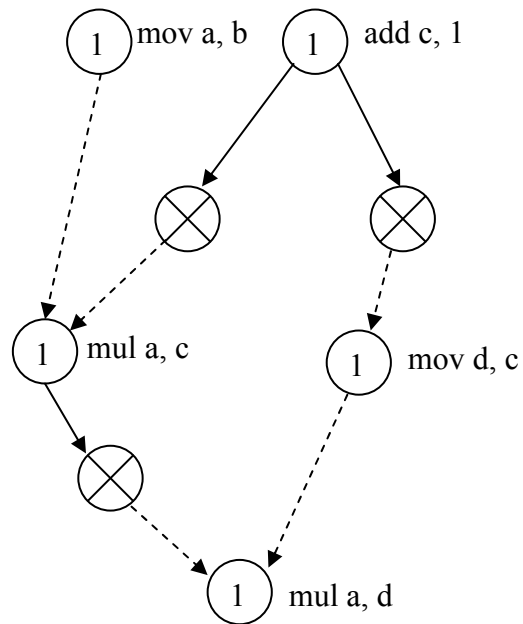


Рис. 4. Представление базового блока с помощью разреженной модели

2.1. Моделирование особенностей архитектуры целевой машины

Такая особенность целевой машины, как инструкции, состоящие из нескольких машинных слов, может быть описана с помощью нескольких последовательных узлов-операций, соединенных жесткими связями.

Команды, продолжительность задержки между которыми строго фиксирована (т.е. время жизни результата выполнения первой из команд ограничено), предлагается моделировать с помощью последовательности, состоящей из двух узлов-операций и нескольких узлов-задержек между ними. Данные узлы соединяются жесткими связями.

Аналогичным образом могут описываться команды переходов с неустраняемыми задержками, только в этом случае вместо второй операции должен использоваться последний узел графа.

3. Выводы

В статье рассмотрен традиционный способ представления базовых блоков с помощью графовой модели. На основе анализа её недостатков введена разреженная модель базовых блоков.

Предлагаемая модель отличается от традиционной специальными видами узлов и связей. В отличие от традиционных моделей, в разреженной модели в качестве узлов используются однопиковые операции конвейера целевой машины. Данные операции объединяются в граф с помощью ребер двух видов – для задания относительного и абсолютного порядка операций.

Разреженная модель позволяет применять единый подход при оптимизации потока команд в базовых блоках при наличии команд из нескольких машинных слов, инструкций перехода с неустраняемыми задержками, а также команд с ограниченным временем жизни результата их выполнения. Эта ее особенность дает возможность оптимизировать потоки команд в базовых блоках в рамках одного универсального алгоритма, что невозможно в традиционных моделях.

Литература

1. Beaty, S. List scheduling: Alone, with foresight, and with lookahead. In Conference on Massively Parallel Computing Systems: the Challenges of General-Purpose and Special-Purpose Computing (Ischia, Italy, May 1994)
2. Intel. i860 64-bit microprocessor programmer's reference manual, 1990.
3. S. Muchnick. Advanced compiler design and implementation, 1997
4. Philip Schielke. Issues in Instruction Scheduling. Rice University, Department of Computer Science. Ph. D. Thesis Proposal
5. Bjorn De Sutter. General-Purpose Architecture Instruction Scheduling Techniques. ELIS Technical Report DG 98-09, November 1998