

УДК 519.68

Л.С. ВЕЛЕНТ<sup>1</sup>, С.И. КИРКОРОВ<sup>2</sup>

<sup>1</sup>Научно-производственное предприятие «МедиаСкан», г. Минск, Республика Беларусь

<sup>2</sup>Высший государственный колледж связи, г. Минск, Республика Беларусь

## КОМПОЗИЦИИ ПАРАЛЛЕЛЬНЫХ АЛГОРИТМОВ ЯЗЫКА ADA ДЛЯ СРЕДСТВ СВЯЗИ

### Введение

Есть множество задач, которые по различным причинам должны быть решены быстро. Поэтому во многом делается упор на компьютеры с несколькими ядрами, в частности – в средствах связи обработки речевой и видео информации. Параллельные вычисления (параллельная обработка) – это использование нескольких или многих вычислительных устройств для одновременного выполнения разных частей одной программы (одного проекта) [1]. Основная цель параллельных вычислений – это уменьшение времени решения задачи. Единственным универсальным языком программирования имеющим международный стандарт и поддерживающим параллельные алгоритмы строго стандартными средствами является Ada. Выбор языка Ada – оптимальный подход для приложений критичных к безопасности и выполнения требований информационной безопасности [2,3].

В данной работе рассмотрен следующий подход к композиции параллельных алгоритмов и улучшению локальности алгоритмов: каждое гнездо циклов (каждый алгоритм) отображается на параллельный компьютер сам по себе. Получены достаточные условия на эти отображения, которые обеспечивают хорошую локальность «стыковочных» данных. Исследована локальность данных, определяемых в одном, а используемых в другом гнезде циклов.

### 1. Тайлинг

Одним из основных способов уменьшения накладных расходов на использование иерархической памяти является тайлинг [4,5]. При тайлинге операции алгоритма разбиваются на тайлы, т.е. на множества операций, выполняемых атомарно, как одна единица вычислений.

*Обобщенный тайлинг является допустимым, если для любой такой зависимости выполняются следующие условия:*

$$j \geq i_k, \quad k_1 \leq k \leq c_{\alpha,\beta}, \quad (1.2.3)$$

$$\vartheta^\beta \geq \vartheta^\alpha \quad (1.2.4)$$

*При отсутствии указанных зависимостей тайлинг допустим [5].*

### 2. Улучшение локальности композиции параллельных алгоритмов

Локальность – это вычислительное свойство алгоритма, отражающее степень использования памяти с быстрым доступом. рассматривается композиция параллельных алгоритмов. Подход для нее следующий: каждый алгоритм (каждое гнездо циклов) отображается на параллельный компьютер сам по себе. Наша задача – получить условия на эти отображения, обеспечивающие хорошую локальность «стыковочных» данных

Пусть у нас имеется зависимость  $S_\alpha(I) \rightarrow S_\beta(J)$ ,  $I \in V_{j\vartheta^\alpha}$ ,  $J \in V_{j\vartheta^\beta}$ , которую описывает функция зависимостей вида:

$$\bar{\Phi}_{\alpha,\beta}(J) = \Phi_{\alpha,\beta}J + \Psi_{\alpha,\beta}N - \varphi^{(\alpha,\beta)}. \quad (2.3.1)$$

Так как данная функция зависимостей отображает  $\bar{\Phi}_{\alpha,\beta}: V_{j\vartheta^\beta} \rightarrow V_{j\vartheta^\alpha}$ , то будем считать, что координаты виртуальных процессоров, выполняющих операции  $S_\beta(J)$  многомерного цикла, задаются функциями вида  $I = \bar{\Phi}_{\alpha,\beta}(J)$ ,  $I = \Phi_{\alpha,\beta}J + \Psi_{\alpha,\beta}N - \varphi^{(\alpha,\beta)}$ .

Для каждого  $\vartheta$ ,  $1 \leq \vartheta \leq \theta$  зафиксируем один из параметров циклов  $j_{\zeta_{\vartheta\beta}}$ . Обозначим  $e_{\zeta_{\vartheta\beta}}$  – вектор размера  $n_\beta$ , у которого координата с номером  $\zeta_{\vartheta\beta}$  равна 1. Обозначим через  $(\Phi_{\alpha,\beta})_{\zeta_{\vartheta\beta}}$  и  $(\Psi_{\alpha,\beta})_{\zeta_{\vartheta\beta}}$  строки матриц с номером  $\zeta_{\vartheta\beta}$ ; если  $\zeta_{\vartheta\beta} > n_\beta$ , то примем  $(\Phi_{\alpha,\beta})_{\zeta_{\vartheta\beta_1}} = 0$ ,  $(\Psi_{\alpha,\beta})_{\zeta_{\vartheta\beta}} = 0$ ,  $\phi_{\zeta_{\vartheta\beta}}^{(\alpha,\beta)} = 0$ .

Поставим в соответствие тайлам  $V_{j^{gl}}^{\vartheta}$  функции вида

$$\text{Pr}^{\vartheta}(j^{gl}) = j_{\vartheta}^{gl}. \quad (2.3.2)$$

Нам необходимо определить последовательности зернистых вычислений (т.е. разбитых на тайлы). К одной последовательности отнесём операции тайлов с одинаковыми значениями функции (2.3.2). Данные функции  $\text{Pr}^{\vartheta}$  можно использовать для распределения операций алгоритма между процессорами: организуется вычислительный процесс для выполнения на процессоре с номером  $\text{Pr}^{\vartheta}(j)$  операций тайлов  $V_{j^{gl}}^{\vartheta}$ . Задача выбора функций  $\text{Pr}^{\vartheta}$  – задача распределения операций между процессорами – должна быть согласована с задачей распределения массивов данных между процессорами. От степени согласованности распределения операций и массивов данных зависит локальность параллельных реализаций алгоритмов, одного из важнейших вычислительных свойств алгоритмов.

Для операторов  $S_{\alpha}$  и  $S_{\beta}$  функцию (2.3.2) можно записать следующим образом:

$$\text{Pr}^{\vartheta\alpha}(I^{gl}) = i_{\vartheta\alpha}^{gl}. \quad (2.3.3)$$

$$\text{Pr}^{\vartheta\beta}(J^{gl}) = j_{\vartheta\beta}^{gl}. \quad (2.3.4)$$

**Т е о р е м а 2.3.1.** Пусть элемент массива  $a_l$  определяется на вхождении  $(a_l, S_{\alpha}, 1)$  и используется на вхождении  $(a_l, S_{\beta}, q)$  в правой части оператора  $S_{\beta}$ . Определение и использование данных происходят в одном вычислительном процессе, если выполняются следующие условия:

$$(\Phi_{\alpha,\beta})_{\zeta_{\vartheta\alpha}} = e_{\zeta_{\vartheta\beta}}, \quad (2.3.5)$$

$$r_{\zeta}^{\vartheta\alpha} = r_{\zeta}^{\vartheta\beta}, \quad (2.3.6)$$

$$(\Psi_{\alpha,\beta})_{\zeta_{\vartheta\alpha}} N - \varphi_{\zeta_{\vartheta\alpha}}^{(\alpha,\beta)} - m_{\zeta}^{\vartheta\alpha} = -m_{\zeta}^{\vartheta\beta}. \quad (2.3.7)$$

Можно сформулировать следствие из теоремы 2.3.1 [6].

**С л е д с т в и е 2.3.1.** Пусть элемент массива  $a_l$  определяется на вхождении  $(a_l, S_{\alpha}, 1)$  и используется на вхождении  $(a_l, S_{\beta}, q)$  в правой части оператора  $S_{\beta}$ . Если  $(\Psi_{\alpha,\beta})_{\zeta_{\vartheta\alpha}} = 0$ , то определение и использование данных происходят в одном вычислительном процессе, если выполняются следующие условия:

$$(\Phi_{\alpha,\beta})_{\zeta_{\vartheta\alpha}} = e_{\zeta_{\vartheta\beta}}, \quad (2.3.11)$$

$$r_{\zeta}^{\vartheta\alpha} = r_{\zeta}^{\vartheta\beta}, \quad (2.3.12)$$

$$\varphi_{\zeta_{\vartheta\alpha}}^{(\alpha,\beta)} + m_{\zeta}^{\vartheta\alpha} = m_{\zeta}^{\vartheta\beta}. \quad (2.3.13)$$

Таким образом, в данном случае мы получили условия для любого  $N$ .

## Выводы

Таким образом, в теореме 2.3.1. [6] сформулированы достаточные условия, при выполнении которых мы получаем, что вычисленные данные попадают именно в тот процесс, в котором эти данные нужны, т.е. определение данных и использование данных происходит в одном вычислительном процессе. Подтверждение, что теорема работает - решение, полученное геометрическим путём совпадает с решением, полученным теоретическим путём [6]. Полученные результаты могут использоваться при автоматизации распараллеливания алгоритмов на языке Ada.

## Литература

1. Лиходед, Н.А. Методы распараллеливания гнезд циклов: Курс лекций / Н.А. Лиходед. – Мн.: БГУ. 2007. – 100 с..
2. Ada 95 Language Reference Manual ANSI/ISO 8652.1995-std: [Electronicresource]. – <http://www.adapower.com/rm95/index.html>.
3. Киркоров С. И., Киркорова Л. С. (Велент Л.С.) Параллельные алгоритмы математических моделей: исследование локальности и применение языка Ada. Ж-л: ВІСНИК Харківського національного університету імені В.Н. Каразіна, №863 Серія: Математичне моделювання. Інформаційні технології. Автоматизовані системи управління, Випуск 12, стр. 129-142, Харків, 2009..
4. Лиходед, Н.А. Характеристика локальности параллельных реализаций многомерных циклов / Н.А. Лиходед // Доклады НАН Беларуси. 2010 – Т54. №1 – С.26-32.
5. Лиходед, Н.А. Обобщенный тайлинг / Н.А. Лиходед // Доклады НАН Беларуси. 2011.
6. Киркорова Л.С.(Велент Л.С.) Улучшение локальности композиции параллельных алгоритмов: Магистерская диссертация, Институт подготовки научных кадров НАН Беларуси, 2011.