

УДК 004.272

К.В. Кротов, доцент, канд. техн. наук

Севастопольский национальный технический университет

Студгородок, г. Севастополь, Украина, 99053

E-mail: K_krotov@mail.ru

**ФОРМАЛИЗАЦИЯ МЕТОДА РАСПАРАЛЛЕЛИВАНИЯ ПОСЛЕДОВАТЕЛЬНЫХ ПРОГРАММ,
ИСПОЛЗУЮЩЕГО ПОНЯТИЯ ДВУМЕРНОЙ МОДЕЛИ ВЫЧИСЛИТЕЛЬНОГО ПРОЦЕССА**

Рассматривается формализация метода построения ярусно-параллельного вида последовательной программы, позволяющего автоматически с помощью средств вычислительной техники формировать схему параллельного решения задач.

Повышение производительности вычислений является актуальной задачей, связанной с возрастающей сложностью решения задач. Возможным путем решения данной проблемы является увеличение вычислительной мощности систем обработки информации, требующее значительных материальных затрат. Альтернативной возможностью увеличения производительности вычислений является реализация их параллельного выполнения. При этом для реализации алгоритма задачи необходимо использовать предварительно сформированную схему ее параллельного выполнения. Большинство методов формирования параллельной схемы выполнения программы [1] предполагают построение данной схемы вручную. Для сложных последовательных программ такой способ формирования их параллельных алгоритмов требует значительных временных затрат. Поэтому актуальной является разработка таких методов распараллеливания программ, которые позволяли бы формировать указанную схему параллельного выполнения автоматически, с использованием средств вычислительной техники. Анализируя существующие методы, возможно выполнить их совершенствование для решения задачи автоматического формирования схем (алгоритмов) параллельного выполнения программ.

Целью данной работы является дальнейшее развитие и уточнение математической формализации метода распараллеливания последовательных программ, предложенного в [1], позволяющее выполнять автоматическое построение их ярусно-параллельных форм.

Предложенный в [1] метод распараллеливания линейных (с отсутствием ветвлений) программ предусматривает совместное выполнение операций, входящих в разные арифметические выражения программы (либо линейного участка программы). При этом суть данного метода предусматривает формирование множеств операций программы – не включившихся, го-

товых к включению, включившихся, работающих, выключившихся, – для каждого временного шага реализации вычислительного процесса. Общая формализация двумерной модели вычислительного процесса представлена в [1]. Основная идея реализации метода предполагает формирование множества операций, готовых к выполнению на каждом временном шаге, с использованием введенных формализаций.

Начальная формализация метода предполагает, что каждая j -ая операция, входящая в линейную программу, характеризуется набором параметров:

$$a_j = \{j, \theta_j, \text{In } a_j, \text{Out } a_j, W_j, |\text{In } a_j|, |W_j|\}, \quad (1)$$

где θ_j – тип операции a_j (характеризует ход вычислительного процесса и определяет количество входных операндов), $\text{In } a_j$ – множество входных операндов j -ой операции (операнды, являющиеся входными для операции a_j), $\text{Out } a_j$ – множество выходных переменных j -ой операции (переменные, являющиеся результатами выполнения этой операции; при этом при реализации рассматриваемого алгоритма на каждом шаге вычисления должно выполняться условие уникального именования результатов), W_j – множество номеров операций, использующих результаты j -ой операции, $|\text{In } a_j|, |W_j|$ – мощности соответствующих множеств.

Так как каждая операция линейной программы идентифицируется номером j , то $W_j = \{j_1, \dots, j_k\}$, где $j_1, \dots, j_k$ – номера операций, использующих результаты операции j .

Для реализации алгоритма распараллеливания строится таблица операций, каждая строка которой представляет собой набор параметров вида (1). Построение ярусно-параллельной формы (ЯПФ) программы выполняется в соответствии с этой таблицей при учете выполняемости условия готовности операции к активизации, сформулированном в [1]. Данное условие требует для запускаемой на выполнение операции того, чтобы все операции, формирующие для нее входные данные, к этому времени закончили свое функционирование.

В [1] выполнена начальная формализация предложенного алгоритма путем задания условий вынесения операции с номером j на k -ый ярус ЯПФ (условие вынесения операции на k -ый ярус соответствует условию ее выполнимости):

1) для унарных операций

$$j \in W_{j'}, \text{ где } W_{j'} \subseteq \{W_{j'}\}_{k-1};$$

2) для бинарных операций

$$j \in \{W_{j'} \cap W_i\}, \text{ где } W_{j'} \subseteq \{W_{j'}\}_{k-1}, W_i \subseteq \bigcup_{k-1} \{W_i\}_{k-1}. \quad (2)$$

Первое соотношение предусматривает принадлежность номера выносимой на k -ярус операции любому из множеств W_j операций с предыдущего яруса. Второе соотношение требует принадлежности операции попарным пересечениям множеств W_j операций предшествующего $(k-1)$ -ого яруса.

Для n -арных операций выражения вида (2) в [1] строятся аналогичным образом. Полученные выводы комментирует рисунок 1.

На рисунке 1,а отображается условие вынесения унарной операции, на рисунке 1,б – бинарной.

Однако значения переменных, являющихся входными для операции a_j , могут формироваться не только на $(k-1)$ -м ярусе, но и на всех ярусах

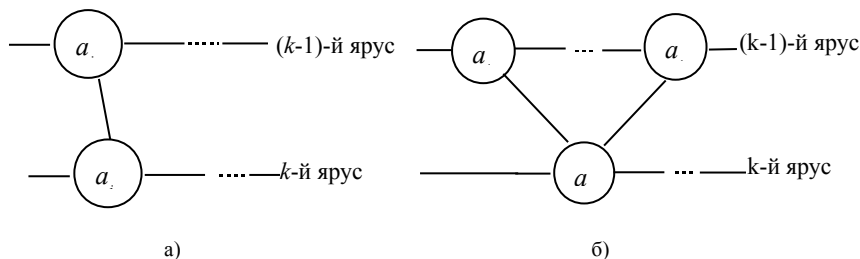


Рисунок 1 – Выполнение условия вынесения операции на k -й ярус ЯПФ соответственно с (2)

ЯПФ, предшествующих ему. Данное утверждение комментирует рисунок 2, на котором представлены условия выполнения бинарной (рисунок 2,а,б) и n -арный, на примере операции с тремя входными операндами, операций (рисунок 2,в,е). В частности, рисунок 2,а определяет, как и в рассмотренном выше случае, условие вынесения бинарной операции на k -ый ярус при принадлежности ее номера попарному пересечению множеств W_j операций одного $(k-1)$ -ого яруса, т.е. данные для этой операции подготовлены вычислениями на предыдущем ярусе. Рисунок 2,б определяет возможность принадлежности номера операции, выносимой на следующий ярус ЯПФ, множествам W_j операций, находящихся на разных ярусах, т.е. подготовку

исходных для этой операции данных на любом уровне, предшествующем $(k-1)$ -ому и на самом $(k-1)$ -ом уровне.

Аналогичные рассуждения могут быть применены и для операций с тремя входными операндами. Рисунок 2,в определяет, что для вынесения операции на k -ый ярус ЯПФ используемые ею данные могут быть подготовлены тремя операциями, выполняющимися на $(k-1)$ -ом ярусе.

Рисунок 2,г поясняет условие, при котором два операнда для операции с тремя входами могут быть сформированы на ярусах, предшествующих $(k-1)$ -ому, и один операнд формируется непосредственно на $(k-1)$ -ом ярусе. Аналогичные рассуждения могут быть применены и для рисунка 2,д. Завершающее условие в цепочке рассуждений, связанных с требова-

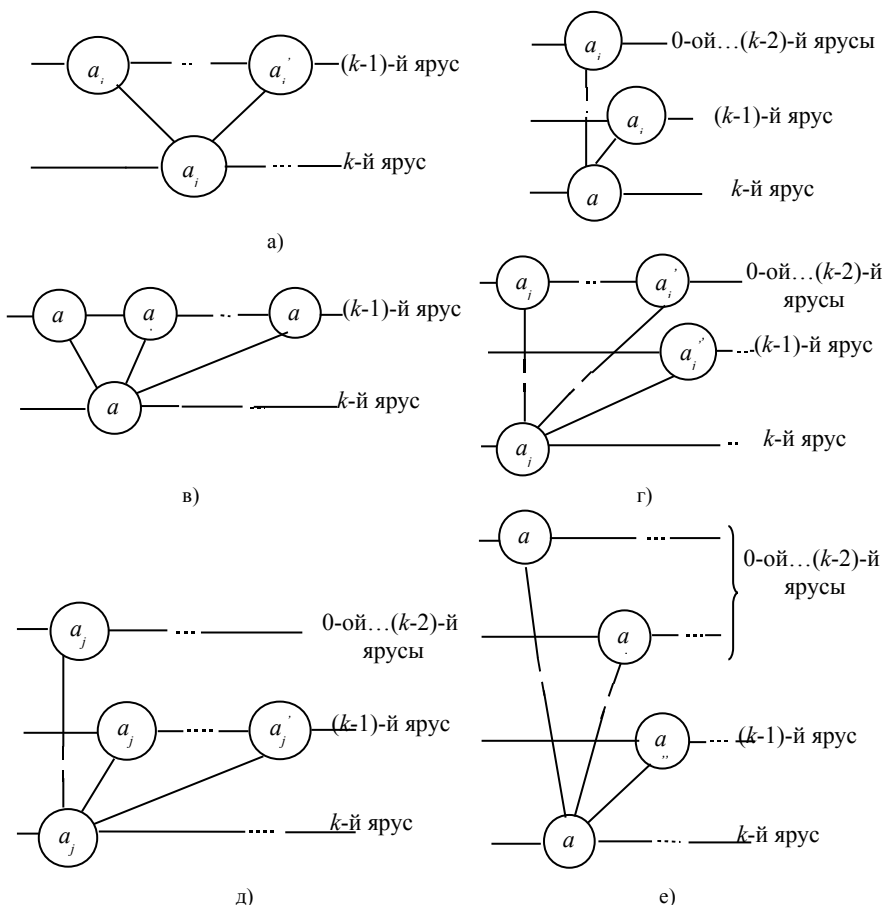


Рисунок 2 – Формирование условий вынесения операции a_j на k -ый ярус ЯПФ

ниями для вынесения операции на k -ый ярус, должно определять подготовку двух операндов на любых различных двух уровнях, предшествующих $(k-1)$ -ому, и одного операнда непосредственно на $(k-1)$ -ом ярусе ЯПФ.

В соответствии с введенными рассуждениями (комментируемыми рисунком 2) формализация условий вынесения операции a_j на k -й ярус ЯПФ выполнена в следующем виде:

1) для унарных операций

$$j \in \left(\bigcup_{i=1}^m W_i \right)_{k-1}, \text{ где } m - \text{число операций на } (k-1)\text{-ом ярусе}; \quad (3)$$

2) для бинарных операций

$$j \in \begin{cases} \bigcup_{i, i'} (W_i \cap W_{i'})_{k-1}, \text{ где } i' = \overline{i+1, m} \text{ при } i = \overline{1, m-1}; \\ \left(\bigcup_{p=0}^{k-2} \left(\bigcup_{i'=1}^{m'} W_i \right)_p \cap \left(\bigcup_{i=1}^m W_i \right)_{k-1} \right), \end{cases} \quad (4)$$

где m' – число операций на любом из ярусов ЯПФ, предшествующих $(k-1)$ -му; p – индекс яруса, предшествующего $(k-1)$ -ому;

3) для операций с тремя входными операторами

$$j \in \begin{cases} \bigcup_{i, i', i''} (W_i \cap W_{i'} \cap W_{i'')_{k-1}, \text{ где } i'' = \overline{i'+1, m} \text{ при } i' = \overline{i+1, m-1}, \text{ при } i = \overline{1, m-2}; \\ \bigcup_{p=0}^{k-2} \left(\bigcup_{i, i'} (W_i \cap W_{i'}) \right)_p \cap \left(\bigcup_{i=1}^m W_i \right)_{k-1}, \text{ где } i'' = \overline{i'+1, m'} \text{ при } i' = \overline{1, m'-1}; \\ \bigcup_{p=0}^{k-2} \left(\bigcup_{i'=1}^{m'} W_i \right)_p \cap \left(\bigcup_{i, i'} (W_i \cap W_{i'}) \right)_{k-1}, \text{ где } i' = \overline{i+1, m} \text{ при } i = \overline{1, m-1}; \\ \bigcup_{p, l} \left(\bigcup_{i'=1}^{m'} W_i \right)_p \cap \left(\bigcup_{i''=1}^{m''} W_{i''} \right)_l \cap \left(\bigcup_{i=1}^m W_i \right)_{k-1}, \text{ где } l = \overline{p+1, k-2} \text{ при } p = \overline{0, k-3}. \end{cases} \quad (5)$$

В выражении (5) через m обозначено количество операторов на $(k-1)$ -ом ярусе ЯПФ, через m' – количество операторов на p -ом ярусе ЯПФ, через m'' – количество операторов на l -ом ярусе.

Первое соотношение выражения (4) предполагает объединение полученных пересечений множеств W_j внутри одного $(k-1)$ -ого яруса для операций с номерами i и i' соответственно. Второе соотношение выражения (4) предполагает объединение попарных пересечений множеств W_j операций $(k-1)$ -ого яруса с операциями предшествующих ему ярусов.

Первое соотношение выражения (5) определяет объединение всех подмножеств, полученных в результате трехкратного пересечения мно-

жеств W_j на одном ярусе. Второе соотношение определяет пересечение подмножеств, полученных в результате попарного пересечения множеств W_j на любом ярусе до $(k-1)$ -ого, с множествами W_j на самом $(k-1)$ -ом ярусе (с последующим объединением полученных подмножеств). При этом номер операции, выносимой на следующий ярус, должен входить в объединение указанных пересечений для всех ярусов до $(k-1)$ -ого. Рассуждения, комментирующие третье соотношение, аналогичны. Последнее соотношение выражения (5) определяет условие принадлежности номера операции множествам W_j любых трех уровней ЯПФ (с обязательной принадлежностью множеству W_j на $(k-1)$ -ом ярусе). Полученные в результате подмножества должны быть также объединены для разных уровней.

Выражения в полученных системах (3)–(5) имеют тот же порядок следования, что и порядок схем распараллеливания на рисунке 2.

Полученные выражения позволяют автоматизировать процесс построения ЯПФ последовательных линейных программ, реализовав его средствами вычислительной техники. Выполненная программная реализация полученных соотношений позволяет формировать схему параллельного выполнения программы, при этом анализ программы в ее исходной форме осуществляется на основе таблицы, в которой операции представлены в стандартной форме (1). Перспективы дальнейших исследований в данном направлении связаны с непосредственным (программным) применением разработанной формализации метода к решению задач распараллеливания последовательных программ и оценкой его эффективности.

Библиографический список

1. Параллельная обработка информации; Под ред. А.Н. Свенсона. — Т. 2: Параллельные методы и средства распознавания образов. — Распараллеливание алгоритмов обработки информации. — К.: Наук. думка, 1985. — 280 с.

Поступила в редакцию 6.12.2005 г.