

COMPARISON OF METHODS FOR GETTING SEPARATION OF PARALLEL LOGIC CONTROL ALGORITHMS

E.I. Vatutin, A.D. Jihad, M.H. Nadjadjra, I.V. Zotov

Abstract

The work addresses consideration of the methods for getting [sub]optimal separations of parallel logic control algorithms into a set of concurrent sequential subalgorithms (modules) to be implemented by a parallel logic control system. Comparison of optimizing parameters quality and time costs are given.

Аннотация

Работа посвящена рассмотрению методов построения [суб]оптимальных разбиений параллельных алгоритмов логического управления на последовательные подалгоритмы (модули) ограниченной сложности, реализуемые параллельной системой логического управления. Приводится сравнение качества оптимизации отдельных параметров и временных затрат на построение разбиений.

Одним из перспективных подходов к синтезу систем логического управления, реализующих параллельные алгоритмы управления, является их создание на базе микроконтроллерных сетей, представляющих собой сосредоточенные коллективы параллельно работающих микроконтроллеров, в совокупности решающие возложенную на них задачу управления. При проектировании подобных систем возникает ряд задач, требующих своего решения. Одной из них является задача выбора разбиения заданного параллельного алгоритма управления, характеризующаяся NP-полнотой. Для ее решения могут быть использованы различные методы, отличающиеся как быстродействием, так и качеством получаемых решений.

В таблице приведен перечень методов, которые могут быть использованы для решения поставленной задачи.

Название метода	Критерии сравнения					
	Количество блоков (подалгоритмов)	Распределение микроопераций	Распределение логических условий	Сложность сети межблочных связей	Интенсивность межблочных взаимодействий	Время построения разбиения ¹
Параллельно-последовательный [1]	+	+	+	+	+	~30 мс
С.И. Баранова [2]	-	+	+	-	-	~0,035 мс
А.Д. Закревского [3]	+	-	-	- ²	-	~7 мс
Полного перебора	+	+	+	+	+	10 ² – 10 ³ секунд ³
Случайного перебора	+ ⁴	+	+	+	+	~ ⁵

С целью обеспечения непротиворечивости сравнения методов необходимо обратить внимание на ряд замечаний.

Замечание 1. Оценка времени построения разбиений производилась на алгоритмах размерности ~25 вершин на компьютере с процессором Intel Celeron 850 МГц (ядро Coppermine, CPUID=068Ah).

Замечание 2. Метод А.Д. Закревского непосредственно не оптимизирует сложность сети межблочных связей, однако в следствии малого количества блоков зачастую наблюдается малая сложность сети.

Замечание 3. Время построения разбиения оценено приблизительно и растет очень быстро с увеличением размерности алгоритма.

Замечание 4. Количество блоков оптимизируется при задании маленькой вероятности попадания вершин в новый блок.

Замечание 5. Время нахождения разбиения линейно зависит от количества просмотренных алгоритмов.

На основании представленной таблицы можно провести сравнение методов. Метод С.И. Баранова [2] является самым простым и изначально ориентирован на построение разбиений последовательных алгоритмов (нами рассматривается его обобщение на класс параллельных алгоритмов). Построение блоков разбиения проводится последовательно, при этом учитывается распределение микроопераций, логических условий и «близость» расположения вершин к синтезируемому блоку, однако не оптимизируется ни количество блоков, ни сложность сети связей и интенсивность межблочных взаимодействий. Метод работает быстро в силу своей простоты. Метод А.Д. Закревского [3] сводит решения данной задачи к нахождению минимальной закраски специального графа на языке ПРАЛУ, качество разбиений в целом определяется качеством алгоритма раскраски. Метод при сравнительно низких временных затратах дает близкое к минимальному количество блоков и, как следствие, зачастую приводит к маленькой сложности сети межблочных связей (хотя непосредственно этот критерий не оптимизируется). Однако он не оптимизирует ни распределение микроопераций и логических условий, ни интенсивность межблочных взаимодействий. Параллельно-последовательный метод [1] является наиболее сложным (как в плане трудоемкости программной реализации, так и в плане временных затрат на построение разбиения), однако он оптимизирует все представленные в таблице критерии и имеет поддержку учета технологических ограничений. Метод полного перебора по определению обладает наилучшими характеристиками, однако на практике он обычно не приемлем из-за чрезмерных временных затрат на построение разбиения. Метод случайного перебора, как показывают предварительные исследования, требует перебора большого количества вариантов разбиений для достижения приемлемого качества, что трудновыполнимо на практике ввиду ограниченности временных затрат.

Литература

1. Ватутин Э.И., Зотов И.В. Метод формирования субоптимальных разбиений параллельных управляющих алгоритмов // Труды II международной конференции «Параллельные вычисления и задачи управления» РАСО '04 памяти Е.Г. Сухова. М.: Институт проблем управления им. В.А. Трапезникова РАН, 2004. С. 884–917.

2. Баранов С.И., Журавина Л.Н., Песчанский В.А. Метод представления параллельных граф-схем алгоритмов совокупностями последовательных граф-схем // Автоматика и вычислительная техника. 1984. №5. С. 74–81.
3. Закревский А.Д. Параллельные алгоритмы логического управления. Минск. ИТК НАН Б. 1999. 202 с.