УДК 681.3
Э.И. Ватутин, канд. техн. наук, доцент, кафедра вычислительной техники, ЮЗГУ (e-mail: evatutin@rambler.ru)
С.А. Мосин, студент кафедры вычислительной техники, ЮЗГУ (e-mail: ibn_hottab33@mail.ru)
В.С. Титов, д-р техн. наук, профессор, зав. кафедрой вычислительной техники, ЮЗГУ (e-mail: titov-kstu@rambler.ru)

ИСПОЛЬЗОВАНИЕ ВЕНГЕРСКОГО АЛГОРИТМА ПРИ АНАЛИЗЕ ТАБЛИЦ ВКЛЮЧЕНИЙ В РАМКАХ ПАРАЛЛЕЛЬНО-ПОСЛЕДОВАТЕЛЬНОГО ПОДХОДА К СИНТЕЗУ РАЗБИЕНИЙ

Приведено описание метода и алгоритма анализа таблиц включений с использованием Венгерского алгоритма при параллельно-последовательном синтезе разбиений граф-схем параллельных алгоритмов. Показано, что необходимые затраты времени возрастают на величину порядка степени параллелизма граф-схемы алгоритма $\omega_{\max}$.

Ключевые слова: дискретная комбинаторная оптимизация, Венгерский алгоритм, параллельно-последовательный метод, граф-схемы параллельных алгоритмов, разбиения

***

Одной из задач дискретной комбинаторной оптимизации, возникающей при проектировании однородных многомодульных систем логического управления в базисе логических мультиконтроллеров, является задача поиска разбиения заданной параллельной граф-схемы алгоритма на множество частных алгоритмов (блоков) в соответствии с рядом структурных и функциональных ограничений [1–3]. Поставленная задача относится к классу *NP* и не может быть решена точно за приемлемое время для абсолютного большинства практически важных случаев, поэтому при ее решении довольствуются субоптимальными решениями, получаемыми с использованием различных эвристических методов [4–9]. Одним из методов, применяемых для ее решения, является метод параллельно-последовательной декомпозиции [3–5], основой которого является построение разбиения множества вершин A граф-схемы алгоритма $G=\langle A,V\rangle$ на множество сечений $\Re=\left\{\Omega_1,\Omega_2,...,\Omega_{N_\Re}\right\}$ с использованием математического аппарата для преобразования конструктивных подмножеств вершин (*R*-выражений) [3, 10, 11], для каждого из которых впоследствии производится разбиение на субсечения, которые, в свою очередь, подлежат распределения по блокам разбиения на основании анализа таблиц включений [1–3, 12, 13]. Подобная параллельно-последовательная стратегия синтеза разбиения по сравнению, например, с жадными подходами [6–9] имеет преимущество, при ее использовании ожидается получение более близких к оптимуму решений (разбиений) с меньшими значениями частных показателей качества. Проведенные вычислительные эксперименты [13–17] показывают, что преимущество параллельно-последовательного подхода особенно выражено в области очень сильных технологических ограничений, однако в областях слабых и отсутствующих ограничений метод несколько проигрывает жадным подходам. Также по некоторым показателям качества (например, сложность сети межблочных связей Z_α и интенсивность межблочных взаимодействий Z_δ) метод отстает от жадных подходов для различных значений технологических ограничений, за исключением узкой области очень сильных ограничений. Результирующее качество решений, получаемых с использованием метода параллельно-последовательной декомпозиции, во многом определяется именно этапом анализа таблиц включений, поэтому поиск оптимального соответствия субсечений и блоков разбиения является важной подзадачей при синтезе разбиения. Ранее уже были предприняты попытки модификации этого этапа [15, 18], результатом которых стало некоторое улучшение качества получаемых решений. В данной статье приводится описание подхода к анализу таблиц включений путем сведения задачи поиска оптимального соответствия субсечений и блоков разбиения к задаче о назначениях, для решения которой возможно получение оптимальных решений за полиномиальное время с использованием Венгерского алгоритма [19] с некоторыми отличиями от классической постановки задачи.

Процесс построения блоков разбиения в рамках параллельно-последовательного подхода заключается в следующем [2, 3, 12, 13]. Из множества сечений $\Re$ поочередно выбираются сечения Ω_i, $i=\overline{1, N_\Re}$ в каком-либо порядке (что несущественно с точки зрения рассматриваемой подзадачи, однако влияет на результирующее качество разбиений [2, 3, 13]). Для очередного сечения Ω_i находится его разбиение на субсечения $\Upsilon(\Omega_i)=\left\{\tilde{\Omega}_1, \tilde{\Omega}_2, ..., \tilde{\Omega}_{N_\Upsilon(\Omega_i)}\right\}$, такие что $\bigcup\limits_{j=\overline{1, N_\Upsilon(\Omega_i)}} \tilde{\Omega}_j = \Omega_i$, $\tilde{\Omega}_j \neq \varnothing$, $\tilde{\Omega}_j \bigcap \tilde{\Omega}_k = \varnothing$, $j,k=\overline{1, N_\Upsilon(\Omega_i)}$, $j \neq k$, $\neg\left(a_j \omega a_k\right)$, $a_j \in \tilde{\Omega}_l$, $a_k \in \tilde{\Omega}_l$, $l=\overline{1, N_\Upsilon(\Omega_i)}$, где ω – обозначение отношения параллельности вершин $a_i \in A$ граф-схемы алгоритма *G*. Фактически при нахождении множества субсечений $\Upsilon(\Omega_i)$ необходимо отыскание одного из ортогональных разбиений с минимальным числом подмножеств $N_\Upsilon(\Omega_i)$, не содержащего пустых подмножеств, причем в рамках каждого из субсечений не может быть вершин, находящихся в отношении параллельности. В общем случае подобных разбиений может быть несколько, с использованием жадного подхода выбирается любое из них, т.к. более интеллектуальные способы разбиения не оказывают существенного влияния на качество результирующих решений [2, 3, 13]. Далее для полученного множества субсечений производится построение таблицы включений $T\left(\Upsilon(\Omega_{k+1}), \Gamma_k\right)=\left\{T_{ij}\right\}$, $i=\overline{1, N_\Upsilon(\Omega_{k+1})}$, $j=\overline{1, |\Gamma_k|}$, где $\Gamma_k=\left\{A_1, A_2, ..., A_{H_k}\right\}$ – множество блоков разбиения на *k*-м шаге, $H_k=|\Gamma_k|$ – текущее число блоков разбиения. Элементы таблицы включений, показывающие возможность и оптимальность включения субсечения $\tilde{\Omega}_i$ в блок A_j (обозначается как включение $\left(\tilde{\Omega}_i \to A_j\right)$), рассчитываются по следующей формуле:

$$t\left(\tilde{\Omega}_i, A_j\right) = \begin{cases} "-", \exists a_m \in \tilde{\Omega}_i, a_n \in A_j : a_m \omega a_n, \\ "+", \left(W\left(\tilde{\Omega}_i\right) + W\left(A_j\right) > W_{\max}\right) \vee \\ \vee \left(\left|X\left(\tilde{\Omega}_i\right) \cup X\left(A_j\right)\right| > n_{\text{ЛУ}}\right) \vee \left(\left|Y\left(\tilde{\Omega}_i\right) \cup Y\left(A_j\right)\right| > n_{\text{МО}}\right), \\ K_1^Y \left|Y\left(\tilde{\Omega}_i\right) \cap Y\left(A_j\right)\right| - K_2^Y \left|Y\left(\tilde{\Omega}_i\right) \setminus Y\left(A_j\right)\right| + \\ + K_1^X \left|X\left(\tilde{\Omega}_i\right) \cap X\left(A_j\right)\right| - K_2^X \left|X\left(\tilde{\Omega}_i\right) \setminus X\left(A_j\right)\right| - \\ -\left(K_\alpha \Delta Z_\alpha + K_\delta \Delta Z_\delta\right) + K_\psi \psi\left(\tilde{\Omega}_i\right)\left|\Psi\left(A_j\right)\right| \text{ иначе,} \end{cases}$$

где $K_1^X, K_2^X, K_1^Y, K_2^Y, K_\alpha, K_\delta, K_\psi$ – весовые коэффициенты; $X\left(\tilde{A}\right), Y\left(\tilde{A}\right), W\left(\tilde{A}\right)$ – соответственно множество сигналов логических условий, множество сигналов микроопераций и суммарное число микрокоманд (вес) для вершин подмножества $\tilde{A} \subset A$; $n_{ЛУ} = X_{\max}, n_{МО} = Y_{\max}, W_{\max}$ – соответственно ограничения на максимальное число логических сигналов, сигналов микроопераций и емкость памяти контроллера, ΔZ_α – приращение сложности сети межблочных связей; ΔZ_δ – приращение интенсивности межблочных взаимодействий; $\Psi\left(A\right) \subseteq A$ – подмножество условных вершин в составе множества вершин *A*, $\psi\left(A\right)$ – функция наличия в множестве *A* условных вершин

$$\psi\left(A\right) = \begin{cases} 1, & \Psi\left(A\right) \neq \varnothing, \\ 0, & \text{иначе.} \end{cases}$$

При нарушении структурных или функциональных ограничений для включения $\left(\tilde{\Omega}_i \to A_j\right)$ соответствующий элемент таблицы $T_{ij} \in \left\{"+", "-"\right\}$ и распределение указанного субсечения в блок невозможно. При анализе таблицы $T\left(\Upsilon\left(\Omega_{k+1}\right), \Gamma_k\right)$ требуется отыскать такое взаимно однозначное соответствие субсечений и блоков разбиения

$$\Theta\left(\Upsilon\left(\Omega_{k+1}\right), \Gamma_k\right) = \begin{bmatrix} \tilde{\Omega}_1 & \tilde{\Omega}_2 & \ldots & \tilde{\Omega}_{N_\Upsilon(\Omega_{k+1})} \\ A_{i_1} & A_{i_2} & \ldots & A_{i_{N_\Upsilon(\Omega_{k+1})}} \end{bmatrix},$$

что $\sum\limits_{\substack{\left(\tilde{\Omega}_j,\, A_{i_j}\right)\in \\ \in\Theta\left(\Upsilon(\Omega_{k+1}),\, \Gamma_k\right)}} t\left(\tilde{\Omega}_j, A_{i_j}\right) \to \max$. Включения, используемые при максимизации указанной суммы, будем отмечать символом «+». Несколько примеров таблиц включений и определение оптимальных включений для них приведены на рис. 1.

а)

	$\tilde{\Omega}_1$	$\tilde{\Omega}_2$	$\tilde{\Omega}_3$
A_1	$0{,}1^+$	–	–
A_2	–	–	–
A_3	–	$0{,}2^+$	–
A_4	–	–	$0{,}3^+$

б)

	$\tilde{\Omega}_1$	$\tilde{\Omega}_2$	$\tilde{\Omega}_3$
A_1	$0{,}1$	–	–
A_2	$0{,}2^+$	–	–
A_3	–	$0{,}3^+$	–
A_4	–	–	$0{,}4^+$

в)

	$\tilde{\Omega}_1$	$\tilde{\Omega}_2$	$\tilde{\Omega}_3$
A_1	$0{,}1^+$	$0{,}4$	–
A_2	–	–	–
A_3	–	–	$0{,}3^+$
A_4	–	$0{,}2^+$	–

г)

	$\tilde{\Omega}_1$	$\tilde{\Omega}_2$	$\tilde{\Omega}_3$
A_1	$0{,}1$	$0{,}2^+$	–
A_2	–	–	–
A_3	–	–	–
A_4	–	–	$0{,}3^+$

Рис. 1. Типовые ситуации, возникающие при анализе таблиц включений

В указанной постановке задача схожа с известной задачей о назначениях [20], однако при этом имеет место ряд принципиальных отличий. Во-первых, целевая функция максимизируется, в то время как в классической постановке задачи о назначениях требуется минимизировать функцию стоимости. Данное отличие можно обойти изменением знака для элементов таблицы включений: $T_{ij} := -T_{ij}$. Во-вторых, в общем случае $N_\Upsilon\left(\Omega_i\right) \neq H_k$, т.е. анализируется таблица, не являющаяся квадратной. При этом, если число субсечений $N_\Upsilon\left(\Omega_i\right) > H_k$, в множество блоков разбиения добавляется $N_\Upsilon\left(\Omega_i\right) - H_k$ пустых блоков разбиения. Если же $N_\Upsilon\left(\Omega_i\right) < H_k$, то это не мешает построению оптимального плана включений. В третьих, не все включения являются допустимыми, что при практической реализации может быть учтено путем занесения в соответствующие элементы таблицы включений заведомо маленького значения $t_{\min}$ (например, можно положить $t_{\min} = -\infty$ или, что удобнее при программной реализации, $t_{\min} = -10^{100}$). Далее задача представляет собой классическую задачу о назначениях и может быть решена с использованием Венгерского алгоритма [19]. При этом, если в ходе составления плана оптимальных включений в нем присутствуют включения, для которых $t\left(\tilde{\Omega}_j, A_{i_j}\right) = t_{\min}$ (т.е. на самом деле данные включения запрещены), то субсечения, входящие в состав подобных включений, должны образовать собой новые блоки разбиения. При подобном подходе отпадает необходимость в анализе предельных случаев [2, 3, 15, 18], в некоторых случаях приводивших ранее к образованию лишних блоков в разбиениях. Алгоритм анализа таблиц включений с использованием Венгерского алгоритма, базирующийся на алгоритме, описание которого приведено в [2, 3, 15, 18], может быть представлен в следующем виде:

1. (инициализация) Положить $\Gamma := \varnothing$. Пометить все вершины граф-схемы алгоритма управления, которые необходимо распределить по блокам, как доступные, положив для множества уже рассмотренных вершин $A^* := \varnothing$.
2. Выбрать еще не рассмотренное сечение $\Omega_i \in \Re$. Если все сечения рассмотрены, перейти к п. 11.
3. Исключить из состава сечения Ω_i уже распределенные по блокам вершины: $\Omega_i := \Omega_i \setminus A^*$.
4. Представить сечение Ω_i в виде множества субсечений $\Upsilon(\Omega_i) = \left\{\tilde{\Omega}_1, \tilde{\Omega}_2, ..., \tilde{\Omega}_{N_\Upsilon(\Omega_i)}\right\}$.
5. Сформировать таблицу включений $T\left(\Upsilon(\Omega_i), \Gamma\right)$. Если $N_\Upsilon(\Omega_i) > |\Gamma|$, положить $\Gamma := \Gamma \cup \underbrace{\{\varnothing, \varnothing, ..., \varnothing\}}_{\substack{N_\Upsilon(\Omega_i) - |\Gamma| \\ \text{блоков}}}$, для вновь добавленных блоков положить $T_{ij} := -t_{\min}$.
6. Инвертировать все элементы таблицы включений: $T_{ij} := -T_{ij}$. Определить оптимальный план включений $\Theta\left(\Upsilon(\Omega_i), \Gamma\right)$ с использованием Венгерского алгоритма.
7. Распределить субсечения по блокам в соответствии с планом включений $\Theta\left(\Upsilon(\Omega_i), \Gamma\right)$ для тех включений, у которых $t\left(\tilde{\Omega}_j, A_{i_j}\right) \neq t_{\min}$.
8. Для всех включений $\left(\tilde{\Omega}_j \to A_{i_j}\right) \in \Theta\left(\Upsilon(\Omega_i), \Gamma\right)$, у которых $\left(t\left(\tilde{\Omega}_j, A_{i_j}\right) = t_{\min}\right) \wedge \left(\left|\tilde{\Omega}_j\right| = 1\right)$, сформировать новые блоки разбиения $\Gamma := \Gamma \cup \left\{\tilde{\Omega}_j\right\}$.
9. Для всех включений $\left(\tilde{\Omega}_j \to A_{i_j}\right) \in \Theta\left(\Upsilon(\Omega_i), \Gamma\right)$, у которых $\left(t\left(\tilde{\Omega}_j, A_{i_j}\right) = t_{\min}\right) \wedge \left(\left|\tilde{\Omega}_j\right| > 1\right) \wedge \left(|\Gamma| > 0\right)$, произвести разбиение субсечений на отдельные вершины, сформировать для них отдельную таблицу включений T', распределить вершины по блокам на основании анализа таблицы T'.
10. Пометить вершины, входящие в состав сечения Ω_i как рассмотренные: $A^* := A^* \cup \Omega_i$. Перейти к п. 2.
11. Конец алгоритма.

Ввиду использования Венгерского алгоритма при составлении оптимального плана распределения субсечений по блокам предложенный выше алгоритм всегда обеспечивает оптимальное распределение вершин, образующих сечение Ω_i, по блокам. При этом предложенный алгоритм поиска оптимального плана включений требует порядка $O\left(n^3\right)$, где $n = \max\left(\left|\overline{\Gamma}\right|, \overline{N}_\Upsilon\right) \simeq \omega_{\max}$, $\left|\overline{\Gamma}\right|$ – среднее число блоков в ходе построения разбиения, $\overline{N}_\Upsilon$ – среднее число субсечений, $\omega_{\max}$ – степень параллелизма граф-схемы алгоритма [3, 21], в то время как алгоритм [15, 18] требует порядка $O\left(n^2\right)$ действий. С учетом того, что время анализа таблиц включений является бутылочным горлышком и составляет величину порядка 80–90% от общего времени синтеза разбиения [22], применение Венгерского алгоритма увеличит временн*ы*е затраты на синтез разбиений приблизительно в $\omega_{\max}$ раз. В перспективе дальнейших исследований необходимо исследование того, насколько меняется качество разбиений и время на их отыскание при различных соотношениях силы ограничений и размера граф-схем параллельных алгоритмов.

Список литературы

1. Организация и синтез микропрограммных мультимикроконтроллеров / И.В. Зотов, В.А. Колосков, В.С. Титов [и др.]. Курск: Изд-во «Курск», 1999. 368 с.
2. Комбинаторно-логические задачи синтеза разбиений параллельных алгоритмов логического управления при проектировании логических мультиконтроллеров / Э.И. Ватутин, И.В. Зотов, М.Ю. Сохен, В.С. Титов; Курский государственный технический университет. Курск, 2010. 200 с.
3. Ватутин Э.И. Проектирование логических мультиконтроллеров. Синтез разбиений параллельных граф-схем алгоритмов. Saarbrucken: Lambert Academic Publishing, 2011 г. 292 с.
4. Ватутин Э.И., Зотов И.В. Метод формирования субоптимальных разбиений параллельных управляющих алгоритмов // Параллельные вычисления и задачи управления (PACO’04). М.: ИПУ РАН, 2004. С. 884–917.
5. Ватутин Э.И., Зотов И.В. Параллельно-последовательный метод формирования субоптимальных разбиений параллельных управляющих алгоритмов // Свидетельство об официальной регистрации программы для ЭВМ № 2005613091 от 28.11.05.
6. Баранов С.И., Журавина Л.Н., Песчанский В.А. Обобщенный метод декомпозиции граф-схем алгоритмов // А и ВТ. 1982. № 5. С. 43–51.
7. Ватутин Э.И. Библиотека функций построения разбиений методом С.И. Баранова с жадным последовательным формированием блоков //

Свидетельство о государственной регистрации программы для ЭВМ № 2010612902 от 28.04.10.
8. Ватутин Э.И., Леонов М.Е. Использование смежной окрестности при жадном последовательном формировании блоков разбиения граф-схем параллельных алгоритмов // Известия высших учебных заведений. Приборостроение. 2013. Т. 56. № 6. С. 30-35.
9. Ватутин Э.И., Титов В.С. Библиотека функций для построения разбиений с использованием смежной жадной стратегии и последовательным формированием блоков // Свидетельство о государственной регистрации программы для ЭВМ № 2013619395 от 03.10.13.
10. Ватутин Э.И., Зотов И.В., Титов В.С. Выявление изоморфных вхождений *R*-выражений при построении множества сечений параллельных алгоритмов логического управления // Информационно-измерительные и управляющие системы. № 11, Т. 7. М.: «Радиотехника», 2009. С. 49–56.
11. Ватутин Э.И., Зотов И.В., Титов В.С. Выявления изоморфных вхождений *R*-выражений при построении параллельных алгоритмов логического управления // Известия высших учебных заведений. Приборостроение. 2009. Т. 52, № 2. С. 37–45.
12. Ватутин Э.И., Зотов И.В. Построение блоков разбиения в задаче декомпозиции параллельных управляющих алгоритмов // Материалы и упрочняющие технологии – 2003, Курск. гос. техн. ун-т. Курск, 2003. Т. 2. С. 38–42.
13. Ватутин Э.И., Зотов И.В. Анализ качества блочных разбиений при синтезе логических мультиконтроллеров // Информационно-измерительные и управляющие системы. № 10, Т. 6. М.: «Радиотехника», 2008. С. 32–38.
14. Ватутин Э.И., Волобуев С.В., Зотов И.В. Комплексный сравнительный анализ качества разбиений при синтезе логических мультиконтроллеров в условиях присутствия технологических ограничений // Параллельные вычисления и задачи управления (PACO'08). М.: ИПУ РАН, 2008. С. 643–685.
15. Ватутин Э.И., Волобуев С.В., Зотов И.В. Комплексная сравнительная оценка методов выбора разбиений при проектировании логических мультиконтроллеров // Идентификация систем и задачи управления (SICPRO'08). М.: ИПУ РАН, 2008. С. 1917–1940.
16. Ватутин Э.И., Титов В.С. Сравнение методов синтеза разбиений граф-схем параллельных алгоритмов с использованием двумерных диаграмм // Известия Юго-Западного государственного университета. № 3 (42), 2012. С. 66–74.
17. Ватутин Э.И., Титов В.С. Использование добровольных распределенных вычислений на платформе BOINC для анализа качества разбиений граф-схем параллельных алгоритмов // Параллельные вычисления и задачи управления (PACO'12). М.: ИПУ РАН, 2012. С. 37–54.

18. Ватутин Э.И., Зотов И.В. Повышение качества разбиения алгоритмов при синтезе логических мультиконтроллеров с использованием метода параллельно-последовательной декомпозиции // Перспективы развития систем управления оружием: сборник докладов IV научно-практической конференции, Курск, 19-20 сентября 2007 г. М.: Изд-во «Бедретдинов и Ко», 2007. С. 84–92.
19. Harold W. Kuhn. The Hungarian Method for the assignment problem // Naval Research Logistics Quarterly, 2:83–97, 1955.
20. https://ru.wikipedia.org/wiki/Задача_о_назначениях
21. Ватутин Э.И. Определение степени параллелизма параллельной граф-схемы алгоритма // Интеллектуальные и информационные системы (Интеллект–2009). Тула: изд-во ТулГУ, 2009. С. 24–26.
22. Ватутин Э.И. Анализ узких мест программной реализации метода параллельно-последовательной декомпозиции граф-схем параллельных алгоритмов // Оптико-электронные приборы и устройства в системах распознавания образов, обработки изображений и символьной информации (Распознавание – 2013). Курск, изд-во ЮЗГУ, 2013. С. 235–237.