

УДК 658.512

Б.К. Лебедев, О.Б. Лебедев**ЭВОЛЮЦИОННЫЕ ПРОЦЕДУРЫ РАЗБИЕНИЯ СБИС***

Введение. Одной из широко востребованных задач целочисленного программирования является задача разбиения, рассматриваемой в комбинаторном направлении теории графов. Современная СБИС может содержать десятки миллионов транзисторов, поэтому в связи с ограниченными возможностями вычислительных средств (память, скорость) не может быть спроектирована топология всей схемы в целом. Нормальным является разбиение схемы группированием компонентов в блоки. В результате разбиения формируется множества блоков и межсоединений между блоками. В очень больших схемах используется иерархическая структура разбиения.

Большинство разработанных к настоящему времени алгоритмов разбиения используют в качестве модели схемы граф или гиперграф. Процедура разбиения графа входит в состав большого числа алгоритмов, решающих различные задачи. Весьма популярными алгоритмами разбиения графа $G(X, U)$ на части являются итеративные алгоритмы, требующие для своей работы некоторого начального разбиения [1, 2]. Итеративные алгоритмы делятся на детерминированные и вероятностные.

В детерминированных алгоритмах изменение разбиения (решения) реализуется на основе четкой, детерминированной зависимости от изменяемого решения. Однако при экспериментальном исследовании указанных алгоритмов был выявлен ряд недостатков, ограничивающих их практическое применение. Установлено, что они довольно чувствительны к начальному разбиению, что сказывается на времени работы алгоритма. Экспериментальные исследования подтвердили тот факт, что время работы алгоритмов [1-3] практически не зависит от того, насколько исходное разбиение близко к оптимальному. Недостатком является частое попадание в локальный оптимум («локальную яму»).

В вероятностных алгоритмах переход к новому решению осуществляется случайным образом. Недостатком алгоритмов, реализующих чисто случайный поиск, является значительная трудоемкость. Один из путей преодоления этой трудности состоит в группировке сильно связанных подсхем в кластеры и дальнейшем объединении этих кластеров в узлы (вершины) перед выполнением алгоритма. Размерность данной задачи, таким образом, значительно уменьшается, что ведет к снижению времени поиска [4]. Спектральные методы [5, 6] разбиения схем выполняют следующие шаги: геометрическое вписывание графа в одно или многомерное пространство; разбиение графа на необходимое число подграфов, используя геометрическую информацию, полученную на предыдущем шаге, не нарушая при этом ограничений. Экспериментальные результаты показывают, что спектральные методы находят намного лучшее решение чем, улучшенные итерационные методы. Как показывает опыт, спектральные алгоритмы хоть и применимы к любым графовым задачам но, в отличие от алгоритма Фидучиа-Маттеуз (FM), имеют намного более высокую ВСА [6]. Поэтому спектральные алгоритмы целесообразнее использовать на небольших графах.

* Работа выполнена при поддержке: РФФИ (гранты № 06-01-00272, № 08-01-00473), РНП 2.1.2.3193, РНП 2.1.2.2238.

Недавно был разработан новый класс алгоритмов разбиения гиперграфа [7-10], которые основаны на многоуровневой парадигме. В этих алгоритмах создан упорядоченный набор последовательно уменьшающихся гиперграфов. Находится разбиение самого маленького гиперграфа, затем это разбиение последовательно проектируется на следующий уровень, и на каждом уровне используется итерационный алгоритм для улучшения качества решения. Эксперименты, представленные в [7-9], показывают, что многоуровневые алгоритмы разбиения могут находить существенно лучшие решения за приемлемое время, чем те, которые находятся итерационными и спектральными алгоритмами.

Но следует отметить, что многоуровневые алгоритмы не решают сами по себе проблему разделения графа. Под этим названием скрывается общий подход, основанный на переходе от задачи разбиения большого графа к задаче разбиения намного меньшего графа, для которого гораздо дешевле, с точки зрения затраченного времени, использовать любой итерационный метод решения задачи разбиения.

Возникшие потребности в решении задач большой и очень большой размерности является побудительным мотивом исследований и разработок новых эффективных алгоритмов. Анализ литературы показывает, что наиболее успешными в этих условиях являются методы, основанные на моделировании эволюционных процессов [11-13].

Особый интерес представляет поисковая адаптация, основанная на использовании обучающихся автоматов, моделирующих поведение объекта адаптации в среде. Трудности использования такого подхода связаны, в первую очередь, с проблемой представления исходной формулировки задачи в виде адаптивной системы. Достоинством является повышенная целенаправленность и сходимостью алгоритма [14].

В работе излагаются: методика символьного представления решения задачи разбиения на базе матрицы смежности графа и адаптивные механизмы видоизменения матрицы смежности, а также рассматривается структура процесса эволюционной модификации матрицы смежности для решения задачи нахождения разбиения.

1. Основные положения. Задача разбиения гиперграфа с взвешенными вершинами и ребрами формулируется следующим образом.

Пусть задан гиперграф $H = (X, E)$, где $X = \{x_i / i=1,2,\dots,n\}$ – множество вершин, а $E = \{e_j / e_j \subset X, j=1,2,\dots,m\}$ – множество ребер (каждое ребро – подмножество связываемых им вершин). Вес вершин задается множеством $\Phi = \{\varphi_i / i=1,2,\dots,n\}$, а вес ребер – множеством $\Psi = \{\psi_j / j=1,2,\dots,m\}$. Необходимо сформировать K -узлов, т.е. множество X разбить на K непустых и непересекающихся подмножеств X_v , $X = \cup X_v$, $(\forall i, j) [X_i \cap X_j = \emptyset]$, $X_v \neq \emptyset$.

На формируемые узлы накладываются ограничения. С помощью вектора $P = \{p_v / v=1,2,\dots,k\}$ задается максимально допустимый суммарный вес вершин, назначенных в v -ый узел, а с помощью вектора $N = \{n_v / v=1,2,\dots,k\}$ – максимально допустимое число вершин назначенных в v -ый узел.

Ограничения на вместимость имеют вид:

$$\sum_{i \in I} \varphi_i \leq p_v, \quad I = \{i / x_i \in X_v\}, \quad v=1,2,\dots,k. \quad (1)$$

$$|X_v| \leq n_v, \quad v=1,2,\dots,k. \quad (2)$$

Выражение (1) является ограничением на максимальный вес узла, а выражение (2) – на максимальное число вершин в узле.

Иногда задано допустимое число выводов γ_{max} для узлов. Ограничение для узлов, на число выводов γ , имеет вид:

$$\gamma_v \leq \gamma_{max}, v=1,2,\dots,k$$

$$\gamma_v = |E_v|, E_v = \{e_j \mid (e_j \cap X_v \neq \emptyset) \ \& \ (e_j \cap X_v \neq e_j)\},$$
 (3)

где E_v – множество ребер, связывающих множество вершин X_v с вершинами остальных узлов.

Основным критерием является F_1 – суммарная стоимость ребер в разрезе.

$$F_1 = \sum_{j \in J} \psi_j$$
 (4)

где $J = \{j \mid e_j \in C\}$, $C = \{e_j \mid (\forall v) [e_j \cap X_v \neq e_j]\}$ – множество ребер в разрезе.

Вторым часто используемым критерием является F_2 – суммарное число выводов.

$$F_2 = \sum_{v=1} \gamma_v$$
 (5)

Возможно использования критерия F , являющегося аддитивной сверткой критериев F_1 и F_2 . $F = k_1 \cdot F_1 + k_2 \cdot F_2$.

Рассмотрим пример. Пусть задан граф $G=(X,U)$, представленный на рис. 1.

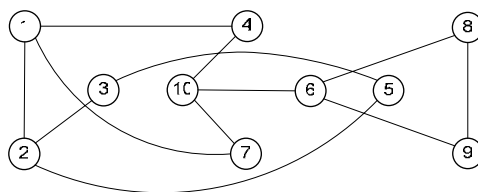


Рис. 1. Начальное разбиение графа G

Необходимо сформировать три узла. Допустимое число вершин назначенных в узлы $n_1=3, n_2=4, n_3=3$. Задано его начальное разбиение $X_1=\{x_1, x_2, x_3\}$, $X_2 = \{x_4, x_{10}, x_7, x_6\}$, $X_3=\{x_8, x_5, x_9\}$. Число связей между узлами – 4. Состояние объекта оптимизации оценивается вектором $S=\{s_i \mid i=1,2,\dots,n\}$. Значением s_i является номер некоторой вершины x_i . Элементы s_i , для которых $1 \leq i \leq n_1$, соответствуют первому узлу X_1 . Элементы s_i , для которых $n_1+1 \leq i \leq n_1+n_2$ соответствуют второму узлу X_2 и т.д. В нашем случае $S_n=\{x_1, x_2, x_3, x_4, x_{10}, x_7, x_6, x_8, x_5, x_9\}$.

На рис. 2, представлено оптимальное разбиение графа G, описываемое вектором $S_o=\{x_2, x_3, x_5, x_1, x_4, x_7, x_{10}, x_6, x_8, x_9\}$.

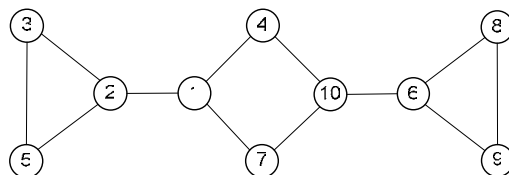


Рис. 2. Оптимальное разбиение графа G

Каждому решению, т.е. каждому вектору S , будем ставить в соответствие матрицу смежности R , у которой столбцы и строки помечены вершинами графа G в том же порядке, в котором соответствующие им вершины расположены в векторе S . Для вектора S_n и S_o соответствующие им матрицы смежности представлены

на рис. 3, 4. Поскольку матрица смежности симметрична относительно главной диагонали, то на рисунках (рис. 3, 4) верхняя часть матрицы не заполнена и не рассматривается. Таким образом, столбцы и строки матрицы с 1 по n_1 помечены номерами вершин, входящих в первый узел, а с $((n_1 + n_2 + \dots + n_i) + 1)$ по $((n_1 + n_2 + \dots + n_i) + n_{i+1})$ – входящих в (n_{i+1}) -ый узел.

	1	2	3	4	10	7	6	8	9	5
1	⊗									
2	1	⊗								
3		1	⊗							
4	1			⊗						
10				1	⊗					
7	1				1	⊗				
6					1		⊗			
8							1	⊗		
9							1	1	⊗	
5		1	1							⊗

Рис. 3. Начальное состояние матрицы смежности

	2	3	5	1	4	7	10	6	8	9
2	⊗									
3	1	⊗								
5	1	1	⊗							
1	1			⊗						
4				1	⊗					
7				1		⊗				
10				1	1		⊗			
6							1	⊗		
8								1	⊗	
9									1	⊗
5									1	1

Рис. 4. Конечное состояние матрицы смежности

Пусть столбцы и строки матрицы с номерами от 1 до $l+m$ помечены элементами, входящими в один узел X_i . Рассмотрим область Q_i матрицы R , образованную на пересечении столбцов и строк матрицы с номерами от 1 до $(l+m)$ симметрично относительно главной диагонали. Элементы области Q_i матрицы R отражают связь между соответствующими вершинами узла X_i . Число h_i отличных от нуля элементов области Q_i матрицы R равно числу внутренних связей между вершинами узла X_i . Число F отличных от нуля элементов матрицы R , не входящих ни в одну из областей Q_i , равно числу внешних связей между узлами X_i . На рисунках (см. рис. 3, 4) области Q_1, Q_2, Q_3 образованы соответственно на пересечении 1-3, 4-7 и 8-10 столбцов и строк матрицы смежности.

Пусть NQ – множество элементов матрицы R , не входящих ни в одну из областей Q_i . Назовем NQ областью внешних связей. Для матрицы (см. рис. 3) $Q_1=2, Q_2=3, Q_3=1, F=6$, а для матрицы (см. рис. 4) $Q_1=3, Q_2=4, Q_3=3, F=2$. Таким образом, задача разбиения графа на узлы с минимизацией числа соединений между узлами сводится к задаче формирования путем перестановок строк и столбцов в матрице смежности графа области внешних связей NQ с минимальным значением F .

2. Эволюционные механизмы формирования минимальной области внешних связей. Формирование области внешних связей NQ с минимальным значением F в матрице R осуществляется в процессе её эволюционной модификации. Эволюционная модификация матрицы R производится путём выборочных групповых перестановок соседних столбцов и строк, что обеспечивает направленное последовательное перемещение элементов матрицы R с ненулевыми значениями из области внешних связей NQ в области Q_i . Адаптивный процесс состоит из повторяющихся шагов, каждый из которых представляет собой переход от одного решения (состояния матрицы R) к другому – лучшему [15].

На каждом шаге анализируются пары $(i, i+1)$ соседних строк и столбцов матрицы. Анализ осуществляется в два такта. На первом такте анализируются все пары $(i, i+1)$, у которых первый элемент i – нечетное число. На втором такте анализируются пары, у которых первый элемент i – четное число.

Например: пусть $n=10$, тогда на первом такте рассматриваются пары строк и столбцов $\{(1,2),(3,4),(5,6),(7,8),(9,10)\}$. На втором такте – $\{(2,3),(4,5),(6,7),(8,9)\}$.

Пары строк и столбцов анализируются независимо друг от друга. По результатам анализа принимается решение о перестановке соседней пары строк и столбцов. Локальная цель перестановок – перемещение ненулевых элементов мат-

рицы снизу-вверх и справа-налево. Глобальная цель – формирование в матрице R области внешних связей NQ с минимальным значением F , т.е. разбиение графа G с минимальным числом связей между узлами.

Пусть для анализа выбрана пара строк и столбцов $(l, l+1)$ матрицы $R = \|r_{ij}\|$ размером $n \times n$ и пусть строки и столбцы пересекают область Q_k , образованную на пересечении столбцов и строк матрицы с номерами от v до w . По нижеприведенным формулам рассчитываются параметры S_1, S_2, S_3, S_4 :

$$\sum_{j=1}^{v-1} r_{ij} = S_1; \quad \sum_{j=1}^{v-1} r_{l+1,j} = S_2;$$

$$\sum_{i=w+1}^n r_{il} = S_3; \quad \sum_{i=w+1}^n r_{i,l+1} = S_4,$$

где S_1 и S_2 – суммы элементов l -ой и $l+1$ -ой строк матрицы R , не входящих в состав области Q_k ; S_3 и S_4 – суммы элементов l -го и $(l+1)$ -го столбцов матрицы R , не входящих в состав области Q_k .

Если пара строк $(l, l+1)$ матрицы R принадлежат двум соседним областям Q_k и Q_{k+1} , то параметры S_1, S_2, S_3, S_4 рассчитываются по следующим формулам:

$$\sum_{j=1}^{l-1} r_{ij} = S_1; \quad \sum_{j=1}^{l-1} r_{l+1,j} = S_2;$$

$$\sum_{i=l+2}^n r_{il} = S_3; \quad \sum_{i=l+2}^n r_{i,l+1} = S_4.$$

В этом случае в состав сумм S_1 и S_2, S_3 и S_4 входят все элементы l -ой и $l+1$ -ой строк, l -го и $l+1$ -го столбцов треугольной матрицы R , за исключением элемента $r_{l+1,l}$. Суть анализа заключается в определении истинностного значения двух нижеприведенных условий:

- 1) $(S_2 - S_1) + (S_3 - S_4) > 0$;
- 2) $(S_2 - S_1) + (S_3 - S_4) = 0$.

Ответ «да», то есть – переставлять, вырабатывается, если выполняется условие 1. В случае выполнения условия 2 ответ «да» вырабатывается с вероятностью P , задаваемой априорно. В остальных случаях вырабатывается ответ «нет».

Адаптивная поисковая процедура продолжается, пока существуют пары, для которых выполняются условия 1 и 2. В результате будет сформирована область внешних связей NQ с минимальным значением F и будет определено разбиение графа с минимальным числом связей между узлами.

Пример. Пусть дан граф G (см. рис. 1). Задано его первоначальное разбиение. Матрица смежности графа G представлена на рисунке (см. рис. 3).

На каждом шаге на первом такте рассматриваются пары $\{(1,2), (3,4), (5,6), (7,8), (9,10)\}$, на втором такте – пары $\{(2,3), (4,5), (6,7), (8,9)\}$. Пара строк и столбцов переставляется, если выполняется одно из двух вышеперечисленных условий. В исходной матрице R столбцы и строки помечены номерами вершин графа G . Перестановка соседней пары строк и столбцов $(i, i+1)$ приводит также к перестановке их меток. Будем в дальнейшем для идентификации строк и столбцов использовать их метки.

Шаг 1, такт 1: $(1,2)$ – «да»; $(3,4)$ – «нет»; $(10,7)$ – «да»; $(6,8)$ – «нет»; $(9,5)$ – «да». Итак, на 1-ом такте 1-го шага осуществляется перестановка пар $(1,2), (10,7), (9,5)$. Модифицированная матрица R представлена на рис. 5.

Шаг 1, такт 2: (1,3) – «да»; (8,5) – «да». Матрица R представлена на рис. 6.

	2	1	3	4	7	10	6	8	5	9
2	⊗									
1	1	⊗								
3	1		⊗							
4		1		⊗						
7		1			⊗					
10				1	1	⊗				
6						1	⊗			
8							1	⊗		
5	1		1						⊗	
9							1	1		⊗

Рис. 5 Шаг 1, такт 1

	2	3	1	4	7	10	6	5	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
7			1		⊗					
10				1	1	⊗				
6						1	⊗			
5	1	1						⊗		
8								1	⊗	
9							1	1		⊗

Рис. 6. Шаг 1, такт 2

Шаг 2, такт 1: (6,5) – «да». Модифицированная матрица R – рис. 7.

Шаг 2, такт 2: (10,5) – «да». Модифицированная матрица R – рис. 8.

Шаг 3, такт 1: (7,5) – «да». Модифицированная матрица R – рис. 9.

Шаг 3, такт 2: (4,5) – «да». Модифицированная матрица R – рис. 10.

Шаг 4, такт 1: (1,5) – «да». Модифицированная матрица R – рис. 4.

Шаг 4, такт 2: нет разрешенных перестановок.

После выполнения четырех шагов в модифицированной матрице сформирована область внешних связей NQ с минимальным значением $F=2$.

	2	3	1	4	7	10	5	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
7			1		⊗					
10				1	1	⊗				
5	1	1					⊗			
6						1		⊗		
8							1		⊗	
9								1	1	⊗

Рис. 7. Шаг 2, такт 1

	2	3	1	4	7	5	10	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
7			1		⊗					
5	1	1				⊗				
10				1	1		⊗			
6						1		⊗		
8							1		⊗	
9								1	1	⊗

Рис. 8. Шаг 2, такт 2

	2	3	1	4	5	7	10	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
4			1	⊗						
5	1	1			⊗					
7			1			⊗				
10				1			⊗			
6						1		⊗		
8							1		⊗	
9								1	1	⊗

Рис. 9. Шаг 3, такт 1

	2	3	1	5	4	7	10	6	8	9
2	⊗									
3	1	⊗								
1	1		⊗							
5	1	1		⊗						
4			1		⊗					
7			1			⊗				
10				1	1		⊗			
6						1		⊗		
8							1		⊗	
9								1	1	⊗

Рис. 10. Шаг 3, такт 2

Для преодоления локального барьера, используются подходы, основанные на сочетании различных видов эволюции.

В первом подходе используются идеи метода моделирования отжига. Если в процессе анализа обнаруживается, что условия (1)-(3) не выполняются, то перестановка осуществляется с вероятностью $P = \exp(-\Delta F/kT)$, где T – температура; ΔF – разница между суммами значений элементов анализируемых строк.

Во втором подходе используется одна из структур генетического поиска [16]. Популяция представляет собой множество матриц смежности (закодированных в

виде хромосом). Декодирование, т.е. получение решения, осуществляется с помощью вышеописанной адаптивной процедуры.

Временная сложность адаптивной процедуры на одном шаге – $O(n)$. Сравнение с известными алгоритмами показало, что при меньшем времени работы новый алгоритм дает более качественные решения.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Naveed Sherwani*. Algorithms for VLSI Physical Design Automation. Kluwer academic publishers. Boston /Dordrecht/ London. 1995.
2. *M. Sarrafzadeh and C. K. Wong*. An Introduction to VLSI Physical Design. New York: McGraw Hill. 1996.
3. *Деньдобренко Б.П., Малика А.С.* Автоматизация проектирования радиоэлектронной аппаратуры. – М., Высш. шк., 2002.
4. *J. Cong, C. Wu*, ‘Global Clustering-Based Performance-Driven Circuit Partitioning’, Proc. ISPD, 2002.
5. *L. Hagen and A. B. Kahng*, "Fast Spectral Methods for Ratio Cut Partitioning and Clustering," Proc. IEEE Intl. Conf. on Computer-Aided Design, 2001, pp. 10-13.
6. *Jan-Yang Chang, Yu-Chen Liu, and Ting-Chi Wang* “Faster and Better Spectral Algorithms for Multi-Way Partitioning”, ASPDAC 1999.
7. *G. Karypis*. *Multilevel hypergraph partitioning*. In J. Cong and J. Shinnerl, editors, Multilevel Optimization Methods for VLSI, chapter 6. Kluwer Academic Publishers, Boston, MA, 2002.
8. *Yongseok Cheon, Seokjin Lee, Martin D. F. Wong*, “Stable Multiway Circuit Partitioning for ECO”, 2003
9. *Navaratnasothie Selvakkumaran and George Karypis*, “Multi-Objective Hypergraph Partitioning Algorithms for Cut and Maximum Subdomain Degree Minimization”, ICCAD 2003.
10. *Cristinel Ababei, Navaratnasothie Selvakkumaran, Kia Bazargan, George Karypis*, “Multi-objective Circuit Partitioning for Cutsizes and Path-Based Delay Minimization”, ICCAD 2002.
11. *Курейчик В.М., Курейчик В.В.* Генетический алгоритм разбиения графа. Известия Академии наук. Теория и системы управления, №4, 1999.
12. *C. Alpert and A. Kahng*. A hybrid multilevel/genetic approach for circuit partitioning. In Proceedings of the Fifth ACM/SIGDA Physical Design Workshop, pages 100-105, 2002.
13. *Курейчик В.М., Лебедев Б.К., Лебедев О.Б.* Поисковая адаптация: теория и практика. – М.: Физматлит, 2006.
14. *Mazumder P., Rudnick E.* Genetic Algorithm For VLSI Design, Layout & Test Automation. India, Pearson Education, 2003
15. *Лебедев Б.К.* Адаптация в САПР: Монография. – Таганрог: Изд-во ТРТУ, 1999.
16. *Лебедев Б.К.* Методы поисковой адаптации в задачах автоматизированного проектирования СБИС: Монография. – Таганрог: Изд-во ТРТУ, 2000.

УДК 681.3.001.63+007.52:611.81

А.Н. Дуккардт, Б.К. Лебедев

КОМПЛЕКСНЫЙ ГИБРИДНЫЙ ГЕНЕТИЧЕСКИЙ АЛГОРИТМ РАЗБИЕНИЯ СХЕМ*

Задача разбиения. Неуклонное повышение степени интеграции СБИС привело к тому, что в них более 60% общей временной задержки сигнала приходится на задержки в межсоединениях. Рост размера области, отводимой для межсоеди-

* Работа выполнена при поддержке: РФФИ (гранты № 06-01-00272, № 08-01-00473), РНП 2.1.2.3193, РНП 2.1.2.2238.