С. В. ЗУДИН, к. т. н. С. Ю. ЛУЗИН, О. Б. ПОЛУБАСОВ

Россия, г. С.-Петербург, Гос. ун-т телекоммуникаций
им. М. А. Бонч-Бруевича, НИИ "Звезда"
E-mail:luzin@spb.cityline.ru

Дата поступления в редакцию
05.06 2001 г.
Оппонент *к. т. н. В. В. СИБИРЯКОВ*

ВЫБОР КОНФИГУРАЦИИ СОЕДИНЕНИЯ КОМПОНЕНТОВ ЭЛЕКТРОННЫХ СХЕМ ДО ЭТАПА РАЗМЕЩЕНИЯ

Предложен метод построения деревьев соединений компонентов в многозвенных цепях с целью повышения качества автоматического размещения в САПР печатных плат и БИС.

Одной из основных причин сравнительно невысокой эффективности известных систем конструкторского синтеза радиоэлектронных средств является несогласованность критериев на этапах размещения элементов и трассировки их соединений. Эта несогласованность сказывается особенно ощутимо, если в проектируемой монтажной схеме есть многозвенные цепи (соединяющие более двух элементов). В таких случаях количество связей между элементами, учитываемое в традиционных алгоритмах размещения, обычно значительно превышает число соединений, действительно требующих учета для обеспечения максимально полной разводки трасс.

Обычно задача размещения предшествует задаче трассировки, и на этом этапе порядок соединения элементов в цепях еще не определен. Поэтому на практике, как правило, схему моделируют мультиграфом, где отдельные электрические цепи представляются полными подграфами (все эквипотенциальные контакты попарно связаны), а качество размещения оценивается либо по суммарной длине ребер таких подграфов, либо по сумме полупериметров прямоугольников, охватывающих контакты, связанные цепью. Такой мультиграф дает в общем случае существенно завышенную оценку реальной длины межсоединений, которая в дальнейшем входит и в оценку качества размещения элементов.

Учет связей, которые на самом деле трассировать будет не нужно, и попытка на этапе размещения создать условия для их оптимальной прокладки неизбежно ухудшают условия для прокладки реальных связей.

Несогласованность принятых формализованных решений задачи размещения с условием полной разводки трасс может быть устранена на стадии построения теоретико-графовой модели принципиальной электрической схемы проектируемого узла, используемой при размещении. Для этого необходимо разработать такие методы устранения избыточных связей, которые улучшают полезные для трассировки критерии (минимум среднеквадратичного разброса степеней вершин, минимум числа ребер графа, минимум толщины скелетного графа, моделирующего схему, и т. д.).

В работе [1, с. 95] была отмечена вышеописанная несогласованность и предпринята попытка ее преодоления. Предлагается из множества конфигураций деревьев выбрать те, которые в объединении дают граф с минимальным числом ребер. Однако описанный в [1] метод слишком трудоемок.

В данной работе предложен альтернативный подход, решающий ту же задачу, но обладающий существенно меньшей комбинаторной сложностью и потому пригодный для практического использования.

Постановка задачи [1]

Из множества конфигураций деревьев выбрать те, которые в объединении дают граф с минимальным числом ребер.

Пусть $X=\{x_1, x_2, \ldots, x_n\}$ — множество вершин графа схемы, $G=\{G_1, G_2, \ldots, G_l\}$ — множество подмножеств X, соответствующих каждой цепи принципиальной электрической схемы.

Рассмотрим n-вершинный граф, в котором смежна любая пара вершин, инцидентных некоторой цепи G_i.

Множество вершин графа есть множество X, а множество ребер $u=\{u_1, u_2, \ldots, u_m\}$. Описанный граф содержит все варианты деревьев для любой из цепей.

Отождествим символы u_i с псевдобулевыми переменными:

$$u_i=\begin{cases}1, \text{ если соответствующее ребро принадлежит графу схемы;}\\ 0 \text{ в противном случае.}\end{cases}$$

Псевдобулевыми переменные названы потому, что на самом деле $u_i=0$ использоваться не будет. Следовательно, при осуществлении преобразований функций используются только два правила: $a\cdot a=a$ и $a\vee a\cdot b=a$.

Очевидно, что необходимым и достаточным условием реализации некоторой цепи i является существование в графе схемы подмножества ребер, образующих дерево на множестве вершин G_i.

Определим булеву функцию связности $F_i\{u_1, u_2, \ldots, u_m\}$ как функцию, принимающую значение 1 тогда и только тогда, когда граф H_i (подграф графа K_n, порожденный множеством вершин G_i) является связным:

$$F_i=\bigvee_{j=1}^{k(i)} u_{j1}\cdot u_{j2}\cdot\ldots\cdot u_{jr(i)}\,. \qquad (1)$$

Члены выражения (1), объединенные дизъюнкцией, представляют собой конъюнкцию переменных, соответствующих ребрам, доставляющим связность H_i. Поскольку из всякого связного графа, содержащего циклы, после удаления некоторого непустого множества ребер можно получить дерево, то термы в выражении (1), соответствующие графам H_i с циклами, поглощаются термами, соответствующими деревьям. Таким образом, можно считать, что F_i соответствует дизъюнкции всевозможных деревьев на G_i.

Так как условием реализации исходной схемы является связность всех графов H_i, то его можно записать в виде

$$\mathop{\&}_{i=1}^{l} F_i = 1. \qquad (2)$$

Последовательно выполняя в выражении (2) операции конъюнкции и пользуясь правилом поглощения, приведем левую часть к виду

$$\bigvee_{j=1}^{p} u_{j1} u_{i2} \ldots u_{jr(j)}. \qquad (3)$$

Выбирая в выражении (3) терм с минимальным числом букв, определим совокупность ребер искомого графа схемы.

К сожалению, такой способ решения практически нереализуем для схем, содержащих более 20 компонентов.

Предлагаемый метод

Поскольку конечной целью является получение терма минимальной длины, то, по-видимому, нет необходимости получать все термы, выполняя огромное количество операций. В ряде случаев на основе анализа исходных данных можно установить эквивалентность частичных решений с точки зрения выбранного критерия и выбрать один из вариантов. В других случаях тот же анализ исходных данных позволяет осуществить декомпозицию цепей. И в тех, и в других случаях существенно снижается размерность задачи.

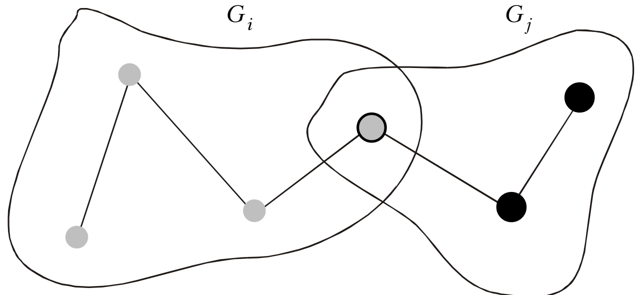
Без ограничения общности можно считать, что внутренние соединения компонента (фрагменты цепей, соединяющие контакты одного компонента) отсутствуют.

1. Пусть некоторая цепь G_k соединяет только два элемента — x_i и x_j, тогда ребро, соединяющее элементы x_i и x_j, обязательно войдет в результирующий граф.

2. Пусть для пары цепей i и j ($1 \le i \ne j \le n$)

$$|G_i \cap G_j| \le 1,$$

тогда в графе $G' = G_i \cup G_j$ все решения, содержащие по одному дереву для каждой из цепей, эквивалентны по числу ребер. В подобных случаях нет необходимости ни генерировать все деревья для каждой из цепей, ни попарно перемножать термы, соответствующие деревьям. Достаточно произвольно выбрать по одному дереву для каждой из цепей.

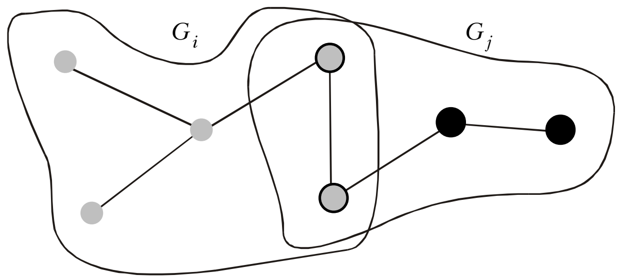


$G_i \cap G_j \le 1$

3. Пусть для пары цепей i и j

$$|G_i \cap G_j| \ge 2,$$

тогда минимальное дерево в $G' = G_i \cup G_j$ будет содержать одно (произвольное) дерево из $G_i \cap G_j$, по одному (произвольному) дереву из $G_i \setminus G_i \cap G_j$ и $G_j \setminus G_i \cap G_j$, а также два произвольных ребра, доставляющих связность трех указанных подграфов.



$G_i \cap G_j \ge 2$

4. "Поглощение":

а) если для пары цепей i и j

$G_i = G_j$,

то одну из цепей можно исключить из рассмотрения (если необходимо, увеличив вес соответствующих ребер);

б) если для некоторой цепи m

$$\bigcup_{k=1}^{P} G_k = G_m, \qquad (4)$$

и при этом

$$\forall i, j \in \overline{1, P} \ \exists \ |G_i \cap G_j| \le 1 \qquad (5)$$

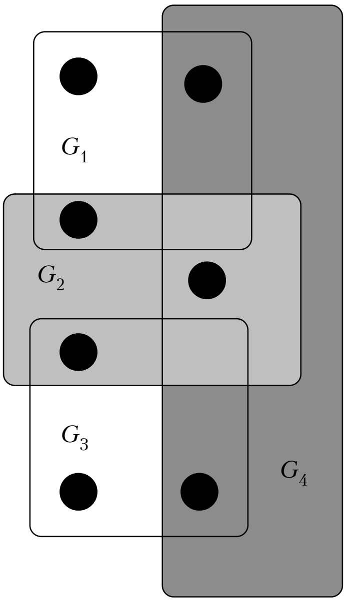
и

$$\sum |G_i \cap G_j| \ge P - 1, \qquad (6)$$

то цепь m можно исключить из рассмотрения, поскольку любая совокупность деревьев для подграфов составит связный граф.

Если же условие (6) не выполняется, и совокупность деревьев для подграфов составит несвязный граф, то цепь m также можно исключить из рассмотрения, добавив в этом случае ребра, доставляющие связность графа.

Ниже приведен рисунок, иллюстрирующий случай, когда при выполнении (5) суммарная мощность пересечения цепей больше их числа.



$$P=4,$$

$$\sum_{i=1}^{4}\sum_{j=i+1}^{4}\left|G_i\cap G_j\right|=5>P-1$$

Таким образом, исследуя пересечения подмножеств, порожденных цепями, можно существенно уменьшить размерность задачи, осуществив сначала максимальную декомпозицию этих подмножеств, а затем выбрав произвольно по одному из деревьев для каждой части цепи и ребра для объединения этих частей в связный граф.

Для максимальной декомпозиции схемы целесообразно выбрать наибольшее число цепей, не имеющих общих ребер.

Чем меньше мощность цепи, тем меньше вариантов реализации деревьев. Очевидно, что начинать нужно с подмножеств наименьшей мощности. Кроме того, целесообразно получить сначала из таких подмножеств максимальное независимое множество, т. е. максимальное множество цепей, попарно не имеющих общих ребер:

$$\forall\, i,j\in\overline{1,P}\ \exists\,|G_i\cap G_j|\le 1 \text{ и } \sum|G_i\cap G_j|\ge P-1,$$

$$\left|\bigcup_{k=1}^{P}G_k\right|=\min,\ P\to\max.$$

Алгоритм.

Начальные установки:

Все G_i упорядочены по возрастанию $|G_i|$, т. е.

$$\forall i\in\overline{1,n}\ \exists\,|G_1|\le|G_2|\le\ldots\le|G_i|\le\ldots\le|G_n|.$$

$$\overline{B}=\{G_i\},\ i\in\overline{1,n}.$$

$B:=G_1$; $\overline{B}:=\overline{B}\setminus G_1$.

Пока в $\overline{B}$ есть G_i, для которых $|B\cap G_i|\le 1$

{

выбрать $|B\cup G_i|=\min$;

(при равенстве оценок предпочтение $B\cap G_i\ne\varnothing$, это обеспечивает связность графа).

$B:=B\cup G_i$; $\overline{B}:=\overline{B}\setminus G_i$.

}

Полученное множество цепей будет образовывать некоторый базис. Лучше всего, если базис представляет собой связный граф, включающий все вершины, т. к. информация о достижимости вершин позволит более обоснованно выбирать деревья.

Декомпозиция цепей базиса.

Пусть X_i — множество элементов, инцидентных цепи G_i.

Пусть выделена подсхема, содержащая множество X_i, и связи между ними, т. е. части G_j' всех цепей G_j, для которых

$$\left|G_i\cap G_j\right|\ge 2.$$

1. Если $\bigcap\limits_{|G_i\cap G_j|\ge 2} G_j'=G_i$, то при критерии минимума ребер графа выбор дерева произволен.

Примечание. Отметим, что критерий минимума ребер далеко не единственный, а возможно, и не главный, поэтому следует обратить внимание на другие критерии, например, на максимум диаметра графа. Чем больше диаметр графа, тем больше степеней свободы у его вершин, тем больше гибкость при укладке его в монтажное пространство, тем больше вариантов укладки, обеспечивающих минимум длины связей. Например, если среди вершин, инцидентных G_i, существует только пара вершин x_j и x_k, для которых выполняются условия

$$G_i\cap G_j=\{x_j\},\ G_i\cap G_k=\{x_k\},$$

т. е.

$$\left|G_i\cap G_j\right|=1,\ \left|G_i\cap G_k\right|=1,$$

то фиксируется простая цепь, в которой x_j и x_k — висячие вершины, а порядок соединения остальных вершин произволен. Если же подобных вершин больше двух, то целесообразно отложить фиксацию ребер и перейти к рассмотрению других цепей, в которых, возможно, выбор будет однозначным, и появится дополнительная информация о степени близости вершин, принадлежащих G_i.

2. Если $\bigcap\limits_{|G_i\cap G_j|\ge 2} G_j'\ne G_i$, то X_i целесообразно разбить на классы, соответствующие пересечениям различной кратности множеств компонентов, инцидентных различным цепям. В этом случае все вышеизложенное можно применить для подсхемы (исключить поглощаемые цепи, найти базис и т. д.) до тех пор, пока для какой-либо цепи (части цепи) не будут фиксированы двухконтактные соединения.

После декомпозиции цепей базиса осуществляется декомпозиция остальных цепей в порядке возрастания их мощности.

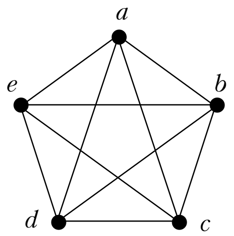
Пример

Продемонстрируем преимущества предлагаемого подхода на примере из [1]. Граф схемы задан матрицей инцидентности “цепи — модули”:

	a	b	c	d	e
k	1	1	0	1	1
l	1	0	1	1	0
m	0	1	1	0	1
n	1	1	0	0	1
p	1	0	0	1	1
q	0	1	1	1	0

Здесь a, b, c, d, e — модули; k, l, m, n, p, q — цепи.

Граф вариантов для такой схемы будет представлять собой полный граф K_5.



Поскольку $G_k = G_n \cup G_p$ и при этом $G_n \cap G_p = \{a, e\}$, то цепь k исключаем — “поглощение” б).

Цепи p и q не могут иметь общих ребер ($G_p \cap G_q = \{d\}$), т. е. в графе $G' = G_p \cup G_q$ для каждой из цепей дерево можно выбрать произвольно. Кроме того, $G_p \cup G_q = G$, т. е. $\{G_p, G_q\}$ — базис. (Отметим, что подобными свойствами обладают пары цепей l и m, n и q.)

Выделим подсхему, содержащую множество компонентов, инцидентных цепи p. Поскольку $G'_k = G_p \cap G_k = G_p$, G'_k исключаем из рассмотрения. Оставшиеся G'_n и G'_l имеют по одному ребру:

$$G'_n = G_p \cap G_n = \{a, e\};$$

$$G'_l = G_p \cap G_l = \{a, d\};$$

	a	d	e
~~k~~	~~1~~	~~1~~	~~1~~
l	1	1	0
n	1	0	1
$\boldsymbol{p}$	**1**	**1**	**1**

Аналогично, для цепи q:

$$G'_m = G_q \cap G_m = \{b, c\};$$

$$G'_l = G_q \cap G_l = \{c, d\};$$

	b	c	d
l	0	1	1
m	1	1	0
$\boldsymbol{q}$	**1**	**1**	**1**

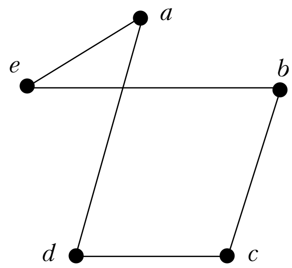
В цепях p, q и l деревья уже определены. Кроме того, зафиксировано по одному ребру в цепях m и n.

Поскольку в цепи m ребро $\{b,c\}$ уже зафиксировано, то остается только ребро $\{b,e\}$:

	b	c	e
k	1	0	1
$\boldsymbol{m}$	**1**	**1**	**1**
n	1	0	1
~~q~~	~~1~~	~~1~~	~~0~~

$$G'_n = G'_k = G_m \cap G_n = \{b, e\}.$$

В результате будет получен граф, представленный ниже.



Таким образом, определена последовательность соединений во всех цепях без построения всех деревьев для каждой из цепей и без сравнения всех возможных конфигураций, содержащих по одному дереву для каждой из цепей.

Заключение

Постановка [1] интересна, но комбинаторная сложность метода слишком высока.

Действительно, число различных деревьев, которые можно построить на n вершинах, равно n^{n-2} [2, с. 173], поэтому число возможных вариантов моделей схемы составит

$$Q = \prod_{i=1}^{m} n_i^{n_i - 2}.$$

Даже для сравнительно небольших схем число Q огромно и исключает возможность построения модели путем прямого перебора.

Для хранения термов необходим объем памяти того же порядка.

Оценим сложность предлагаемого метода.

Для сортировки цепей в порядке возрастания числа инцидентных им компонентов необходимо $O(m\log m)$ операций.

Для формирования базиса требуется определить мощности пересечений цепи-претендента с цепями, входящими в базис. То есть нужно сравнивать списки цепей. Если компоненты в цепях отсортированы в порядке возрастания номеров, то для пары цепей i и j сложность сравнения равна сумме количеств инцидентных им компонентов:

$$Q' = n_i + n_j.$$

Таким образом, комбинаторная сложность построения базиса не выше $O(m\sum_{i=1}^{m} n_i)$.

Умножив эту оценку на количество цепей, получим оценку общей комбинаторной сложности предложенного метода (размерность остальных подзадач существенно меньше):

$$O(m^2 \sum_{i=1}^{m} n_i).$$

ИСПОЛЬЗОВАННЫЕ ИСТОЧНИКИ

1. Шрамченко Б. Л., Абакумов В. Г. Об определении матрицы смежности модулей // В кн.: Автоматизация проектирования в электронике. Вып. 11.— Киев: Техника, 1975.

2. Берж К. Теория графов и ее применения.— М.: ИЛ, 1962.