Математические вопросы кибернетики и вычислительной техники 19, 1998, 7–9.

Алгоритм трассировки транзитных соединений

И. А. Карапетян

Институт проблем информатики и автоматизации НАН РА и ЕрГУ
E-mail: isko@ipia.sci.am

Аннотация

Работа посвящена разработке полиномиального алгоритма порядка $(n^{5/2})$ проведение транзитных соединений текущего трассированного канала на нижнюю контактную линию следующего канала.

Трассировка соединений цепей является одной из наиболее трудных задач в общей проблеме автоматизации проектирования БИС. Это связано с многообразием способов конструктивно - технологической реализации соединений, каждый из которых обусловливает использование специфических критериев оптимизации при решении задачи трассировки. Одним из многочисленных подходов является разбиение коммутационного поля на горизонтальные и вертикальные каналы. Это особенно удобно при проектировании матричных БИС, где вентили размещены по строкам. Между строками выделяются свободные области для трассировки соединения цепей. Такие области образуют каналы, на границах которых расположены выводы (контакты) вентилей. Этим общая задача трассировки сводится к задаче канальной трассировки. Несмотря на то, что в общей постановке задача канальной трассировки $NP-$полная [1], тем не менее в настоящее время она достаточно изучена и разработано множество эффективных и быстрых алгоритмов [1,2].

В работе представлен точный, полиномиальный алгоритм порядка $O(n^{5/2})$ трассировки транзитных соединений. Для удобства изложения алгоритма, будем считать, что коммутационное поле двуслойное. Первый слой отведен для вертикальных отрезков соединений цепей, а второй –для горизонтальных отрезков соединений. Отрезки одной и той же цепи на разных слоях соединяются с помощью сквозных отверстий, называемых межслойными переходами. Предполагаем также, что выводы вентилей помещены на первом слое. В случае многослойной коммутации предложенный алгоритм имеет аналогичное описание.

Пусть трассировка коммутационного поля проводится снизу вверх по горизонтальным каналам. И предположим также, что на очередном канале K_j завершил работу некий алгоритм трассировки. Фрагменты $i-$ой цепи обозначим через $F_{i_1}, F_{i_2}, F_{i_3}, \ldots$. Фрагмент F_{i_k} назовем продолжающимся, если он не содержит контактов на верхней контактной линии канала K_j, и либо в этом канале, либо в последующих каналах существует фрагмент $i-$ой цепи, отличный от фрагмента F_{i_k}.

Содержательно, задача состоит в том, чтобы от возможно большего числа продолжающихся фрагментов вывести по одному транзитному соединению на свободные узлы, т.е узлы, не являющиеся контактами нижней контактной линии следующего канала K_{j+1}.

Перейдем к описанию алгоритма. Итак, допустим, что на плоскости имеем обыкновенную прямоугольную решетку размера $l \times m$. Удобно эту решетку рассматривать как граф $G = (X, E)$, где множество вершин X совпадает с множеством узлов решетки, а множество ребер E – множество отрезков между соседними узлами. Горизонтальные отрезки будем называть горизонтальными ребрами, а вертикальные отрезки - вертикальными ребрами.

Удалим все ребра из всех фрагментов и те ребра, оба конца которых являются межслойными переходами. Пусть F_{i_j} – непродолжающийся фрагмент, и x_i - некоторая его вершина. Если x_i является контактом, то удалим из графа G оставшееся вертикальное ребро, инцидентное с x_i. А когда x_i является межслойным переходом в фрагменте F_{i_j}, то все инцидентные ему ребра удаляются из графа G. Полученный граф обозначим через G'.

Таким образом, задача проведения транзитных соединений сводится к нахождению наибольшего множества путей $L = \{L_1, L_2, \dots\}$ в графе G', удовлетворяющих следующим условиям:

1. Любая пара путей L_i и L_j, $i \neq j$, не имеет общих ребер;

2. Каждый путь L_i начинается с некоторой вершины некоторого продолжающегося фрагмента и заканчивается на некоторой свободной вершине верхней строки графа G';

3. Число путей из множества L, начинающихся с каждого продолжающегося фрагмента, а также входящих в любую свободную вершину, не больше одного.

Алгоритм решения вышепоставленной задачи проводится в три этапа: построение сети, нахождение максимального потока в сети и нахождение трасс транзитных соединений.

Построение сети. Имея граф G', построим сеть G_0 следующим образом. Введем две новые вершины s и t, и назовем их соответственно источником и стоком. С помощью дуг, входящих в t, все свободные вершины верхней строки G' соединяются с вершиной t. Все продолжающиеся фрагменты, состоящие из одной вершины, соединяются с источником s с помощью дуг исходящих из s. А для продолжающихся фрагментов, имеющих более одной вершины, берется новая, дополнительная вершина и с помощью выходящих из нее дуг соединяется со всеми вершинами соответствующего фрагмента. Сама дополнительная вершина в свою очередь соединяется с общим источником s с помощью дуги, исходящей из s. Будем считать, что пропускные способности всех ребер и добавленных дуг сети G_0 равны 1.

Максимальный поток. Допустим, имеем сеть $G = (X, E)$ с пропускными способностями дуг $q_{i,j} \geq 0$, источником s и стоком t; $s, t \in X$. Отметим, что множество чисел $a_{i,j} \geq 0$, определенных на дугах $(x_i, x_j) \in E$, называют потоком [3], если выполняются следующие условия:

$$\sum_{x_j \in \Gamma(x_i)} a_{i,j} - \sum_{x_k \in \Gamma^{-1}(x_i)} a_{k,j} = \begin{cases} v, \text{если} & x_i = s; \\ -v, \text{если} & x_i = t; \\ 0, \text{если} & x_i \neq s, t, \end{cases}$$

и $a_{i,j} \leq q_{i,j}$ для всех $(x_i, x_j) \in E$, где $\Gamma(x) = \{x'/x' \in X \ \text{ и } \ (x, x') \in E\}$ и

$$\Gamma^{-1}(x) = \{x'/x' \in X \ \text{ и } \ (x', x) \in E\}.$$

В [3] доказано, что для нахождения максимального потока имеется полиномиальный алгоритм, являющийся простой модификацией алгоритма расстановки пометок. Там же, в частности, доказано, что время работы алгоритма в сетях с единичными пропускными способностями дуг не превосходит $O(|X|^{2/3}|A|)$, где A - списки смежностей, т.е. для каждой вершины $x \in X$ выписывается множество вершин $A(x) \subseteq X$, смежных с x. Для применения этого алгоритма над сетью G_0 необходимо дать ориентацию неориентированным ребрам.

Если вершина x ребра (x, y) является межслойным переходом, то ребру придаем ориентацию от x в y. Остальные ребра (x, x') будем заменять двумя дугами (x', x) и (x, x') с пропускными способностями 1. И, если в конце работы алгоритма нахождения максимального потока по обеим дугам поток равен 1, то будем считать, что поток по ребру (x, x') равен 0. Или же ребрам можно давать ориентацию по ходу работы алгоритма.

Нахождение трасс транзитных соединений. После нахождения максимального потока слева направо поочередно рассматриваются все свободные вершины верхней строки решетки. Для очередной вершины x поступаем следующим образом:

a) Если не существует насыщенной дуги (дуга с потоком 1), входящей в вершину x, то рассматривается следующая свободная вершина верхней строки решетки;

b) Если существует насыщенная дуга (x', x), входящая в вершину x, то ищется насыщенная дуга, входящая в вершину x'. Если таких дуг несколько, то берется та дуга, которая сохраняет направление предыдущей дуги, т. е. если (x', x) вертикальная (горизонтальная) дуга, то берется вертикальная (горизонтальная) дуга входящая в x'. Этот процесс продолжается до тех пор, пока не достигается некоторая вершина некоторого продолжающегося фрагмента. Все найденые дуги будут составлять транзитное соединение соответствущего фрагмента.

Нетрудно заметить, что нахождение максимального потока - самый трудоемкий этап рассматриваемого алгоритма. Поскольку в прямоугольной решетке имеет место неравенство $|E| < 4|X|$, а для любой сети - $|A| = 2|E|$. Поэтому $|A| \leq 8|X|$. Следовательно, сложность вышеприведенного алгоритма не превосходит $O(|X|^{5/3})$.

Замечание. Для ускорения работы алгоритма можно сократить сеть G_0 следующим образом. Удалим из G_0 все висячие вершины, отличные от s и t. Процесс удаления висячих вершин продолжается до тех пор, пока это возможно. Полученную сеть обозначим через G_0'. Заметим, что в сетях G_0 и G_0' продолжающиеся фрагменты и свободные вершины верхней строки совпадают. И поэтому вместо первоначальной сети G_0 можно рассматривать сокращенную сеть G_0'.

Отметим, что вышеописанный алгоритм был успешно применен при трассировке БМК "Т-6т", в котором вся площадь кристалла была разделена на 91 горизонтальный канал, каждый из которых занимал всю длину (1262) площади кристалла.

Литература

[1] Layout Design and Verification, Ed. T. Ohtsuki, North Holland, 1986, 356p.

[2] Абрайтис Л.Б. Автоматизация проектирования топологии цифровых интегральных микросхем, "Сов. Радио", М., 1985, 200с.

[3] Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность. "Мир", М., 1985, 512с.