2^m импульсов счетчик 4 отрабатывает код z_i в режиме вычитания, так как потенциал верхнего уровня с единичного триггера 12 поступает на вход вычитания счетчика 4. После отработки интервала T_i третий импульс переполнения переключает триггер 11 в "1" и поступает через элемент И 14, элемент ИЛИ 18 на выход генератора и, задержавшись на длительность импульсов переполнения t_n формирователем импульсов 10, сбрасывает триггеры 11 и 12 в "0".

Таким образом, реверсивный счетчик 4 при выдаче генератором интервала, относящегося к поддиапазону $T_0 - T_M$, переполняется дважды, а в поддиапазоне $T_M - T_{max}$ трижды. Поэтому к начальному интервалу T_0 прибавляется дополнительный интервал T_0, в первом случае равный $2t_n$, а во втором - $3t_n$. Чтобы скомпенсировать разницу в одну длительность импульса введен формирователь импульсов 17, вносящий задержку на t_n.

Для несимметричных функций плотности для обеспечения нормальной работы генератора необходимо на нулевой вход триггера 12 подать потенциал, запрещающий счет трех импульсов и работу реверсивного счетчика в режиме вычитания.

Результаты экспериментальных исследований показали, что предложенный способ и генератор позволяют повысить точность воспроизведения заданных функций распределения на порядок в сравнении с известными.

ЛИТЕРАТУРА

1. *В.Н.Четвериков, Э.А.Баканович.* Стохастические вычислительные устройства систем моделирования. М.: Машиностроение, 1989. - 272 с.
2. *Е.В.Батырев, В.А.Майлон и др.* Генератор случайных временных интервалов. Авт. свид. № 440662, к. G06F 1/02. Áþë. ¹ 31, 1974.
3. *А.В.Маргелов и др.* Генератор случайных временных интервалов. Авт. свид. № 74682, кл. G06F 1/02. Бюл. № 25, 1980 и авт. свид. № 1553973, кл. G06F 7/58. Бюл. № 12, 1990.

УДК 658.512

А.М. Никифоров

ОЦЕНКА КАЧЕСТВА РАЗМЕЩЕНИЯ

Введение

Проблема размещения элементов СБИС на кристалле (СБИС на плате) является определяющей для этапа трассировки. Разработано большое число различных алгоритмов размещения элементов по посадочным местам с минимизацией суммарной длины и внутрисхемных пересечений. Все алгоритмы условно можно разделить на две группы: непрерывно-дискретные и дискретные. К первой группе относятся алгоритмы, основанные на построении механических аналогов, градиентные методы и др. Ко второй группе относятся итерационные, последовательные, смешанные, генетические, а также основанные на методах ветвей и границ.[1,2]

Последовательные алгоритмы (часто встречается название "конструктивные с начальным размещением") заключаются в выборе первоначально размещаемого элемента или группы элементов с последующим подсоединением неразмещённых элементов. После размещения элементы уже не перемещаются. Последовательные алгоритмы размещения требуют небольших временных затрат и относятся к классу полиномиальных алгоритмов. Их сложность O(n). Эти алгоритмы широко применяются в практике проектирования БИС. Правила выбора и расстановки элементов зависят от конкретных методов. К сожалению, эти алгоритмы не дают хорошего решения для СБИС более высокой степени интеграции.

Алгоритмы, основанные на идеях ветвей и границ, относятся к точным. При этом множество всех допустимых решений разбивается на меньшие по мощности подмножества, в которых производится поиск оптимального размещения. Метод сопровождается вычислением низших границ, и поиск оптимального размещения прекращается, когда граничное значение начинает превышать значение при найденном допустимом размещении. Процесс продолжается до тех пор, пока не будет закончен поиск в каждом подмножестве или не будет найдено оптимальное размещение.

Итерационные алгоритмы размещения с улучшением качества работают в итеративном режиме. Для изменения позиций выбираются одиночные элементы или группы элементов. Затем по выбранным правилам производится переразмещение элементов. Каждый новый этап должен приводить к размещению с улучшенными характеристиками (уменьшение суммарной длины связей и т.п.). Итерационные алгоритмы позволяют получать более качественные результаты, но за счёт больших временных затрат. Они также относятся к классу полиномиальных алгоритмов, но их сложность порядка $O(n^2)$-$O(n^4)$. К итерационным алгоритмам относятся: метод назначений Штейнберга, метод парного обмена, метод соседнего обмена, метод силонаправленной релаксации, стохастический метод, метод генетических алгоритмов.

Формулировка задачи. Анализ критериев размещения

Целью данной работы является обзор и анализ существующих характеристик размещения элементов СБИС на кристалле (СБИС на плате), а также рекомендации по выбору критериев для разработки генетического алгоритма.

Для работы таких алгоритмов, как генетический, стохастический, метода ветвей и границ вопрос оценки качества полученного решения является основополагающим. Наиболее общими являются такие критерии качества, как суммарная длина соединений и количество внутрисхемных пересечений.

Рассмотрим более подробно все эти критерии.

Для вычисления *длины соединения* между элементами существует три метода:

1. Вычисление реальной длины связи. Этот метод не вызывает трудностей, но он может быть применён только после этапа трассировки, когда линия связи уже существует.
2. Вычисление расстояния между элементами по формуле Пифагора $r=\sqrt{[(s_1-s_2)^2+(t_1-t_2)^2]}$. Этот метод даёт точный результат, но вычисление квадратного корня требует больших вычислительных затрат, а получение результата в вещественном виде затрудняет обработку данных, поэтому предложен третий вариант.
3. Вычисление расстояния между элементами по формуле $r=|s_1-s_2|+|t_1-t_2|$. Этот метод свободен от недостатков предыдущего метода, а поскольку мы производим только предварительную оценку размещения элементов, то точность вычислений нас вполне устраивает.

Необходимо заметить, что критерий минимальности суммарной длины связей сам по себе не является достаточным показателем качества полученного решения, поскольку при достаточно малом расстоянии между элементами резко возрастает количество пересечений линий связи. Поэтому этот критерий следует применять совместно с критерием числа пересечений.

Идея нахождения *числа пересечений* p(G) рёбер графа G, отображённого в сетку G_r, заключается в выделении некоторых подмножеств вершин $X_k, X_l \subset X$. Рассмотрим произвольное ребро $u_{p,g}$ и выделим относительно его подмножества вершин X_k и X_l, которые назовём "подмножествами пересечений". Функция пересечений двух рёбер $u_{i,j}$ и $u_{p,g}$ может принимать два значения:

$$p(u_{i,j}, u_{p,g})= \begin{cases} 1, \text{ если рёбра } u_{i,j} \text{ и } u_{p,g} \text{ пересекаются;} \\ 0 \text{ в противном случае.} \end{cases} \quad (1)$$

Тогда если $(\forall u_{i,j} \in U_{k,l})$ $p(u_{i,j}, u_{p,g})=1$, то подмножества вершин X_k и X_l будут подмножествами пересечений. Здесь $U_{k,l}$ – множество рёбер, инцидентных как X_k, так и X_l.Очевидно, что $X_k \cap X_l = \varnothing$, а $X_k \cup X_l = H$, где $H \subset X$. Разобьём всё множество рёбер графа G на три подмножества U_1, U_2, U_3 $(U_1 \cup U_2 \cup U_3 = U)$ и выделим относительно каждого из них подмножества пересечений. Рассмотрим произвольное ребро $u_{i,j}$. Пусть вершина x_i находится в узле сетки с координатами $s=\pi$, $t=q$, а x_j – вершина в узле с координатами $s=k$, $t=l$. Будем считать, что ребро

$$u_{i,j} \in \begin{cases} U_1, \text{ если } q = l; \\ U_2, \text{ если } \pi \le k; \\ U_3, \text{ если } \pi > k. \end{cases} \quad (2)$$

Подсчёт функции пересечения начнём с рёбер, инцидентных x_1. После определения числа пересечений рёбер, инцидентных x_1, вершина x_1 отбрасывается и рассматривается вершина x_2 и т.д.[1]

В качестве критерия размещения используется также число цепей, пересекающих заданные линии коммутационного поля, что позволяет реально учитывать распределение ресурсов коммутационного поля.[2]

Надо сказать, что все перечисленные выше критерии дают не точную оценку размещения (она может быть получена только после этапа трассировки), а лишь предварительную оценку решения, которой мы и будем пользоваться в дальнейшем.

Анализ критериев завершения работы алгоритмов

Работа итерационных алгоритмов размещения завершается, когда значение целевой функции будет находиться в допустимых пределах некоторой минимальной (максимальной) оценки.

Для определения *нижней границы суммарной длины соединений* между элементами цепи схемы заменяются покрывающими деревьями, строится матрица геометрии D_r, а затем определяется нижняя оценка одним из трёх существующих способов:

1. С помощью треугольной матрицы геометрии D_r. В этом случае в каждой строке D_r выбираются элементы с минимальным весом и определяется их сумма. Затем из D_r выбираются нерассмотренные элементы $d_{i,j}$, и для них определяется наименьшие возможные значения расстояний при их размещении в линейке. Общая сумма
$$L(G)=\sum_{i,j} \min d_{i,j} + \delta, \quad (3)$$
где δ – суммарная длина элементов $d_{i,j}$, выбранных на втором шаге.
2. С помощью построения стандартного графа G_Δ. В этом случае нижняя оценка равна $L(G_\Delta)$ при размещении стандартного графа в линейку.
3. В третьем способе последовательно выбираются все вершины графа с инцидентными им рёбрами и располагаются по очереди в линейке таким образом, чтобы суммарная длина связей была минимальна. Сумма длин связей звёздных подграфов и определит нижнюю оценку минимальной суммарной длины.

Заметим, что проще всего определять первую оценку, хотя она и является заниженной. Вторая и третья оценки точнее, но процесс их вычисления сложнее.

Для определения *нижней границы количества пересечений* необходимо воспользоваться понятием гамильтонова цикла. Поскольку рёбра гамильтонова цикла не образуют пересечений, то общее число пересечений определяется выражением

$$L(G)=\sum_{k=2}^{n-2} P_k\text{, где } P_k=\sum_{j=k+2}^{n} p_{k,j} \text{ – число пересечений рёбер, инцидентных вершине } x_k \quad (4)$$

Непосредственный подсчёт по формуле (4) затруднителен, поэтому предлагается использовать треугольную матрицу геометрии.

При использовании в качестве критерия размещения числа цепей, пересекающих заданные линии коммутационного поля, для определения нижней границы необходимо задаться каким либо априорным числом, либо в качестве признака завершения работы алгоритма использовать число итераций, считая что алгоритм позволяет за данное число шагов достигнуть оптимального значения.

Заключение.

При использовании в качестве критерия размещения только одного параметра, полученное решение может и не являться оптимальным, поэтому при реализации алгоритмов размещения необходимо определять несколько различных критериев. Наиболее приемлемым предлагается вариант, когда в качестве основного параметра вводится суммарная длина связей, а количество пересечений используется как ограничение.

ЛИТЕРАТУРА

1. *Морозов К.К., Одиноков В.Г Курейчик В.М.* Автоматизированное проектирование конструкций радиоэлектронной аппаратуры. М.: Радио и связь, 1983, 280с.

УДК 007.52.611.81(07.58)

Д.П. Пономарев

РАСПОЗНАВАНИЕ ИЗОМОРФИЗМА ГРАФОВ С ПОМОЩЬЮ НЕЙРОННЫХ СЕТЕЙ

Проблема распознавания изоморфизма графов (РИГ) является достаточно актуальной задачей. Графы очень часто используются для описания различных систем, имеющих некие связи между элементами. В этой статье рассматриваются, в основном, графы, описывающие структурные, функциональные и принципиальные электронные схемы, которые являются, как правило, направленными и/или взвешенными. В своей структуре они содержат дополнительную информацию, позволяющую повысить вероятность правильного распознавания. Изоморфные графы не несут никакой новой информации, поэтому желательно их исключать из рассмотрения. РИГ относится к так называемым NP-полным задачам, т.е. не решаемых за полиномиальное время, требующих для своего решения значительных вычислительных ресурсов, а, следовательно, и значительного времени, сокращая сложность анализируемых систем. Поэтому разработка эффективного алгоритма РИГ является важной задачей вычислительной математики и систем искусственного интеллекта.

В настоящей статье предлагается метод распознавания изоморфизма, основанный на применении аппарата нейронных сетей. Такой подход связан достаточной проработанностью математического аппарата, с возможностью применения аппаратных средств, для ускорения обработки возбуждения, возможности легкого распараллеливания вычислений. Распознавание основано на генерации уникальных картин возбуждения или векторов состояний нейронных сетей от неизоморфных графов. Использование нейронных сетей связано с близостью, а иногда, и полной адекватностью, графа и нейронной сети, которая является, по сути дела, направленным и взвешенным графом. Для анализа в качестве исходной информации используются матрицы инценденций. Входное возбуждение подается на опорный нейрон, степень вершины которого наименее часто встречается в векторе степеней графа S[1]. По этим матрицам генерируются нейронные сети и проводится анализ их выходных сигналов в