

Многослойная глобальная трассировка методом коллективной адаптации *

Б.К. Лебедев, О.Б. Лебедев

Таганрогский технологический институт Южного федерального университета,

lbk@tsure.ru

Аннотация — Описываются новый подход, алгоритмы и методики на основе механизмов коллективной альтернативной адаптации для задачи многослойной глобальной трассировки. По сравнению с существующими алгоритмами достигнуто улучшение результатов.

Ключевые слова — многослойная глобальная трассировка; коллективная альтернативная адаптация, оптимизация.

I. ВВЕДЕНИЕ

Одной из важнейших задач при построении интеллектуальных систем автоматизированного проектирования (САПР) сверхбольших интегральных схем (СБИС), является трассировка соединений, которая во многом определяет качество топологии СБИС [1]. В связи с последними успехами в области КМОП технологий наблюдается резкое уменьшение физических размеров элементов СБИС. При этом задача трассировки становится все более сложной. Традиционно эта проблема разбивается на две части – глобальную [2-5] и детальную трассировку [6-8].

В работе рассматривается глобальная трассировка.

Входными данными для глобальной трассировки являются: информация о размещении, включающая местоположение блоков (элементов), выводов на границах блоков, расположения выводов на границах чипа (коммутационного поля) и т.д.

Задача многослойной глобальной трассировки в общем случае состоит из предварительного этапа, заключающегося в разбиении общей области коммутационного поля (КП) каждого слоя на отдельные области, и основного этапа, заключающегося в распределении соединений по областям и определении наборов цепей на границах каждой области. Пропускная способность границы является ресурсом, характеризующим максимально возможную суммарную ширину

цепей, пересекающих эту границу [9]. Итак, на первом этапе формируется множество областей $A = \{a_i | i = 1, 2, \dots\}$ и множество границ между областями $B = \{b_j | j = 1, 2, \dots\}$. При распределении соединений по областям рассматриваются критерии, учитывающие возможную длину соединений, распределение ресурсов с учетом пропускных способностей границ, факторы, влияющие на величину временной задержки в цепях.

Для решения задачи распределения соединений по областям в качестве модели коммутационной среды (КС) используется l -слойный граф $G^l = (X^l, U^l)$, где $l \geq 2$ – положительное целое число, определяющее количество доступных слоев. Предполагается, что слои металлизации нумеруются как 1, 2, ..., l , начиная с нижнего слоя. Вершины графа $x_i \in X$ соответствуют областям $a_i \in A$. Множество ребер U^l можно разделить на два непересекающихся подмножества U_b^l и U_v^l . $U_b^l \cup U_v^l = U^l$. Если в одном и том же слое две области a_i и a_j имеют общую границу b_k , то вершины x_i и x_j , соответствующие этим областям, связываются ребром $u_k^l \in U_b^l$. Каждое ребро в U^l , которое соединяет две смежные вершины на одном и том же уровне (слое), называется граничным ребром и принадлежит U_b^l . Если две области a_i и a_j расположены в двух смежных слоях и их проекции пересекаются, то вершины x_i и x_j , соответствующие этим областям, связываются ребром $u_k^l \in U_v^l$, которое называется межслойным ребром. Для каждого ребра u_k^l , связывающего вершины x_i и x_j , задается вес α_k , равный пропускной способности общей границы между областями, соответствующими вершинам x_i и x_j . Координатам вершины присваивается значения координат центра соответствующей области. Если области имеют один и тот же размер, то граф G^l представляет собой многослойную ортогональную решетку. Предполагается, что каждое граничное ребро име-

* Работа выполнена при финансовой поддержке программы развития научного потенциала высшей школы РНП.2.1.2.1652 и РФФИ (гранты № 09-01-00509, № 10-07-00055).

ет одну и ту же длину проводника, межслойные ребра тоже имеют одну длину проводника. Однако длина для граничного и межслойного проводника может быть разной. Пусть задано множество цепей $T = \{t_s / s = 1, 2, \dots, n_s\}$. Для каждой цепи определяется множество областей КС, в которых существуют контакты, связываемые этой цепью. На графе G^l множеству областей, связываемых цепью $t_s \in T$, соответствует множество вершин $X_s \in X$. Распределить цепь t_s по областям - это значит построить в графе G^l на множестве вершин X_s связывающую сеть. Каждая цепь t_s после ее реализации, т.е. распределения по областям, потребляет определенную часть ресурсов пересекаемых ее границ [10].

В качестве исходных данных для каждой цепи t_s задается параметр φ_s , равный ширине цепи плюс расстояние между цепями. Иногда для одной цепи задаются два параметра - φ_s^1 при распространении цепи по горизонтали, φ_s^2 - по вертикали.

Пусть имеется некоторое решение задачи глобальной трассировки, в соответствии с которым построено множество связывающих сетей E .

Пусть $E_k \in E$ - множество связывающих сетей, построенных для множества цепей $T_k \in T$, в состав которых входит ребро $u_k \in U^l$. Обозначим через β_k сумму ресурсов, необходимых множеству связывающих сетей E_k для прохождения через ребро u_k . Другими словами, сумму ресурсов, необходимых цепям множества T_k для пересечения границы b_k :

$$\beta_k = \sum \varphi_s (s / t_s \in T_k).$$

Для каждого ребра $u_k^l \in U_b^l$ графа G^l введен параметр $w_k = \alpha_k - \beta_k$, который назовем остатком ресурсов [10]. Найдем в графе G минимальное значение параметра w_k и обозначим его w_{min} , т.е.

$$w_{min} \rightarrow \forall_k [w_{min} \leq w_k].$$

Цель стандартного глобального трассировщика - максимизация параметра w_{min} . Действительно, чем больше остаток ресурсов, тем легче реализовать соединения при детальной трассировке, и абсолютно неприемлем результат, когда w_j имеет отрицательное значение.

Возьмем параметр w_{min} с обратным знаком, обозначим как $\gamma = -w_{min}$ и назовем максимальным переполнением. Тогда целью глобального трассировщика становится минимизация максимального переполнения γ или его ограничение

$$\gamma \leq 0.$$

Межслойные переходы для соединения участков цепи, распределенных по разным слоям при многослойной трассировке должны использоваться только тогда, когда они действительно необходимы, так как они не только ухудшают надежность и качество проекта, но также повышают стоимость производства.

Следовательно, необходимо минимизировать стоимостную оценку межслойных переходов при многослойной трассировке.

В настоящее время к решению задачи многослойной глобальной трассировки существуют два основных подхода. При первом подходе все цепи трассируются непосредственно в многослойном пространстве решений [11,12]. Поскольку этот подход непосредственно ищет решение для многослойной глобальной трассировки, он может учитывать количество межслойных переходов во время процедуры поиска.

Однако этот метод может оказаться слишком затратным при большом размере задачи.

Второй подход заключается в следующем [13-16]. На первом этапе сначала граф «сжимается» в однослойный граф. Каждая вершина, ребро и контакт многослойного графа проецируются на нижний слой. Ребра, соответствующие межслойным переходам (МП), исчезают при переходе к однослойному графу. Контакты, находящиеся на одной вертикали, проецируются в один контакт на нижнем слое. Пропускная способность каждого ребра на однослойном графе получается путем сложения соответствующих пропускных способностей ребер с многослойного графа. Затем используется глобальный однослойный трассировщик для решения задачи однослойной трассировки,

На втором этапе выполняется стадия «распределения соединений по слоям». Каждая цепь распределяется на ребрах исходного многослойного графа. Такой подход может использовать преимущества многих существующих однослойных трассировщиков, однако, необходима дополнительная стадия «распределения по слоям». По сути, распределение по слоям сохраняет длину цепей и топологию начального решения однослойной трассировки и предоставляет гибкие возможности для оптимизации задержек, числа межслойных переходов и т.д. Следовательно, распределение по слоям - это один из ключевых шагов при таком подходе.

В данной работе решается задача распределения соединений по слоям с минимальной стоимостью межслойных переходов. При работе с данной задачей, используются ограничения на перегрузку: максимальное «переполнение». При этом цель состоит в том, чтобы найти такое распределение по слоям, при котором удовлетворяются ограничения на переполнение и стоимость МП минимизирована. Для решения данной задачи предлагается алгоритм, который основывается на однослойном результате трассировки, и затем решает задачу распределения по слоям.

Этот алгоритм гарантирует генерацию распределения по слоям, удовлетворяющую данным ограничениям на перегрузку.

II. ФОРМУЛИРОВКА ЗАДАЧИ МНОГОСЛОЙНОЙ ГЛОБАЛЬНОЙ ТРАССИРОВКИ

Определим формально задачу распределения по слоям следующим образом. Задача имеет следующие входные данные: l -слойный граф $G^l = (X^l, U^l)$, T^l – множество непротрассированных цепей на графе G^l , однослойный граф $G^1 = (X^1, U^1)$, полученный сжатием графа G^l , множество цепей T^1 , полученных проецированием T^l на G^1 и решение задачи однослойной трассировки $E^1 = \{e_s^1 \mid s=1,2,\dots,n_s\}$ на (G^1, T^1) . Задача состоит в том, чтобы преобразовать каждую сеть $e_s^1 \in E^1$ из данного решения однослойной трассировки E^1 в ее аналог e_s^l в l -слойном решении глобальной трассировки E^l таким образом, что при проецировании e_s^l на G^1 получаем результат топологии такой же, как для e_s^1 . Другими словами, задача состоит в том, чтобы назначить каждое ребро из e_s^l в слой j , где $1 \leq j \leq l$. Решение E^l должно удовлетворять ограничению на максимальную перегруженность

Поставленная задача в работе решается на двух этапах. На первом этапе независимо друг от друга каждая сеть $e_s^1 \in E^1$ из данного решения однослойной трассировки E^1 преобразовывается в ее аналог e_s^l в l -слойном решении глобальной трассировки E^l с минимизацией стоимости межслойных переходов. Другими словами осуществляется первоначальное распределение соединений всех сетей по слоям. При этом допускаются переполнения отдельных ребер графа G^l .

На втором этапе методами адаптивного поиска осуществляется перераспределение соединений по слоям для выполнения ограничения на максимальную перегруженность с минимизацией общего количества МП.

III. РАСПРЕДЕЛЕНИЕ ПО СЛОЯМ СОЕДИНЕНИЙ КАЖДОЙ СЕТИ

В результате однослойной трассировки для каждой цепи T_s^1 на графе $G^1 = (X^1, U^1)$, строится сеть e_s^1 в виде дерева Штейнера. Межслойные переходы могут располагаться в вершинах дерева Штейнера. Каждая сеть e_s^1 представляется в виде набора фрагментов. Каждый фрагмент представляет двухтерминальное соединение f_{ij} , связывающее две вершины дерева Штейнера x_i и x_j . Отметим, что набор фрагментов покрывает дерево Штейнера, а пересечение двух любых фрагментов из набора является пустым. Введем для каждой вершины x_i дерева Штейнера характеристику d_i - диапазон слоев, с которыми связана вершина x_i через МП. Изначально для каждой вершины x_i , включающей контакт, задается диапазон d_i , равный номеру слоя, в котором лежит контакт. Для вершин, не содержащих контакта, изначально диапазон d_i пустой. Определяем на графе $G^1 = (X^1, U^1)$ локальные степени всех вершин выбранной для расслоения сети $e_s^1 \in E^1$. Вершины рассматриваются в порядке убывания локальных степеней. Пусть x_i вершина с локальной степенью, равной еди-

нице. Определяется вершина x_j , смежная x_i . Пусть контакт k_i , принадлежащий вершине x_i , лежит в слое l_1 , контакт k_j , принадлежащий вершине x_j , лежит в слое l_2 , $l_1 \neq l_2$. Фрагмент цепи f_{ij} , связывающий k_i и k_j , помещается в слой l_1 , а в вершине x_j помещается межслойный переход v_j , связывающий контакт k_j с фрагментом цепи f_{ij} . Обозначим диапазон слоев, связываемых межслойным переходом v_j , как $d_j = \langle l_1 - l_2 \rangle$. Для контакта k_i диапазон $d_i = \langle l_1 \rangle$.

Далее последовательно на каждом шаге выбирается контакт, у которого на данный момент среди инцидентных ему фрагментов цепей только один фрагмент цепи еще не назначен ни в один из слоев. Пусть выбрана вершина x_i , и определена смежная ей вершина x_j .

Если пересечение диапазонов d_i и d_j не пусто, то есть $d_i \cap d_j \neq \emptyset$, то d_i и d_j остаются без изменений, а фрагмент цепи, связывающий x_i и x_j , помещается в слой $l_k \in (d_i \cap d_j)$ с минимальной загрузкой ребер, входящих в состав этого фрагмента.

Если пересечение диапазонов d_i и d_j пусто, то есть $d_i \cap d_j = \emptyset$, то диапазону d_j присваивается новое значение $d_j = d_i \cup d_j$, равное объединению диапазонов d_i и d_j , а фрагмент цепи, связывающий x_i и x_j , помещается в один из слоев $l_k \in (d_i \cup d_j)$ с минимальной загрузкой ребер, входящих в состав этого фрагмента.

Обозначим через d_i^* мощность диапазона d_i . Тогда суммарная стоимость межслойных переходов сети e_s^l определится как

$$F_s = \sum_i (d_i^* - 1).$$

Рассмотрим работу алгоритма на примере.

На рис.1 представлена протрассированная в одном слое цепь, связывающая контакты $\{k_1 - k_7\}$. Сеть представляется в виде набора фрагментов $\{f_{12}, f_{23}, f_{34}, f_{28}, f_{56}, f_{64}, f_{47}\}$. Контакты расположены в следующих слоях: k_5 в 1 слое; k_1, k_6, k_7 во 2 слое; k_2, k_3, k_4, k_8 в 3 слое. Устанавливаем первоначальные диапазоны слоев для всех контактов. $d_5 = \langle l_1 \rangle$. $d_1, d_6, d_7 = \langle l_2 \rangle$. $d_2, d_3, d_4, d_8 = \langle l_3 \rangle$. Начинаем просматривать вершины с локальной степенью $\rho = 1$. В этом случае в вершине размещен контакт и вершине инцидентен только один фрагмент цепи. Первой рассмотрим вершину k_1 . k_1 связана с k_2 . Изначально $d_1 = \langle l_2 \rangle$, $d_2 = \langle l_3 \rangle$. Фрагмент цепи f_{12} , связывающий k_1 и k_2 , помещается в слой l_2 . Поскольку $d_1 \cap d_2 = \emptyset$, то в вершину k_2 помещается межслойный переход, связывающий k_2 с f_{12} , а d_2 принимает новое значение, равное $d_1 \cup d_2$, $d_2 = \langle l_2, l_3 \rangle$. Далее рассмотрим вершину k_5 . k_5 связана с k_6 . $d_5 = \langle l_1 \rangle$. $d_6 = \langle l_2 \rangle$. Фрагмент цепи f_{56} , связывающий k_5 и k_6 , помещается в слой l_1 . Поскольку $d_5 \cap d_6 = \emptyset$, то в вершину k_6 помещается переходное отверстие, связывающее k_6 с f_{56} , а d_6 принимает новое значение, равное $d_5 \cup d_6$, $d_6 = \langle l_1, l_2 \rangle$. Далее выбирается вершина k_7 . k_7 связана с k_4 . $d_7 = \langle l_2 \rangle$. $d_4 = \langle l_3 \rangle$. Фрагмент цепи f_{47} , связывающий k_7 и k_4 , по-

мещается в слой l_2 . Поскольку $d_4 \cap d_7 = \emptyset$, то в вершину k_7 помещается переходное отверстие, связывающее k_7 с f_{47} , а d_4 принимает новое значение, равное $d_4 \cup d_7$, $d_4 = \langle l_2, l_3 \rangle$.

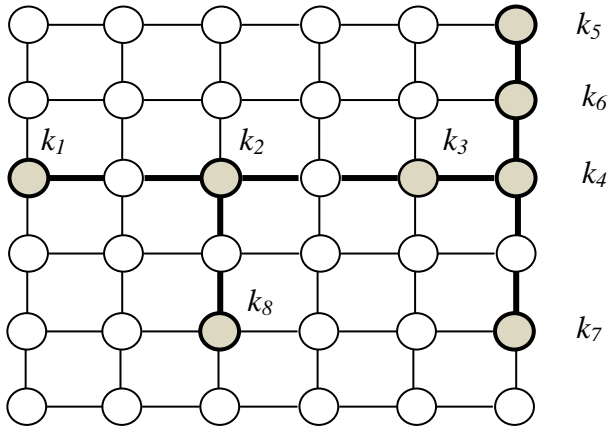


Рис. 1. Формирование межслойных переходов сети

Для вершины k_8 получим. $d_2 = \langle l_2, l_3 \rangle$. $d_8 = \langle l_3 \rangle$. $d_8 \cap d_2 = \langle l_3 \rangle$. Поэтому фрагмент цепи f_{28} , связывающий k_8 и k_2 , помещается в слой l_3 , а d_2 и d_8 остается без изменений.

Далее последовательно на каждом шаге выбирается вершина, у которой на данный момент среди инцидентных ей фрагментов цепей только один фрагмент цепи еще не назначен ни в один из слоев. Пусть выбрана вершина x_6 , и определена смежная ей вершина x_4 . Фрагмент цепи f_{46} еще не назначен ни в один из слоев. $d_6 = \langle l_1, l_2 \rangle$. $d_4 = \langle l_2, l_3 \rangle$. $d_6 \cap d_4 = \langle l_2 \rangle$. Поэтому фрагмент цепи f_{46} помещается в слой l_2 , а d_4 и d_6 остаются без изменений.

Далее выбирается вершина x_4 , определяется смежная ей вершина x_3 . $d_4 = \langle l_2, l_3 \rangle$. $d_3 = \langle l_3 \rangle$. $d_3 \cap d_4 = \langle l_3 \rangle$. Поэтому фрагмент цепи f_{34} помещается в слой l_3 , а d_4 и d_3 остаются без изменений. Определим мощности диапазонов.

$$d^*_1=1; d^*_2=2; d^*_3=1; d^*_4=2; d^*_5=1; d^*_6=2; d^*_7=1; d^*_8=1.$$

Отсюда суммарная стоимость межслойных переходов сети определится как $P=3$. МП будут введены в вершинах k_2, k_4, k_6 .

Суммарная стоимость межслойных переходов для всех сетей определится как

$$F = \sum_s F_s.$$

IV. АДАПТИВНАЯ ПРОЦЕДУРА ПЕРЕРАСПРЕДЕЛЕНИЯ СОЕДИНЕНИЙ ПО СЛОЯМ

На втором этапе осуществляется перераспределение соединений каждой сети $e_s^l \in E^l$ по слоям с минимизацией общего количества МП и с соблюдением ограничений на максимальную перегруженность. В

работе используется комбинаторный подход, основанный на методах поисковой адаптации, реализующих процесс случайного направленного поиска [17]. Представим процесс поиска решения в виде адаптивной системы, работающей на основе моделирования коллективного поведения автоматов адаптации.

Объектами адаптации являются фрагменты цепей. Альтернативным состояниям каждого объекта в среде соответствуют различные варианты его распределения по слоям. Среда - это набор границ областей (ребер графа $G^l = (X^l, U^l)$). Состояние среды определяется распределением всех фрагментов цепей по слоям в соответствии с решением R_l и как следствие набором значений w_j для всех ребер графа G^l и суммарной стоимостью МП.

Состояние объекта адаптации (фрагмента цепи f_{ij}) в среде в соответствии с вариантом его распределения по слоям оценивается набором значений w_j для тех ребер u_j графа G^l , которые входят в фрагмент цепи и диапазонами слоев d_i и d_j , с которыми связан f_{ij} .

Локальная цель объекта адаптации (фрагмента цепи f_{ij}) достичь состояния, при котором выбранный вариант распределения по слоям не содержит ребер u_k с отрицательным значением w_k , а диапазоны слоев d_i и d_j , с которыми связан f_{ij} , минимальны.

Глобальная цель коллектива объектов адаптации достичь такого состояния, при котором оценка w_k для любого ребра u_j графа G^l не имела отрицательного значения, а суммарная стоимость МП минимальна.

Первоначальное решение задачи (первоначальное разнесение каждой сети по слоям) формируется с помощью описанного выше алгоритма. Обозначим как D_{is} и D_{js} полученные после работы первого алгоритма диапазоны слоев для вершин x_i и x_j графа G^l , с которыми связан фрагмент f_{ij} цепи t_s . Назовем их базовыми.

На каждом шаге работы адаптивной системы осуществляется выбор альтернатив для объектов адаптации и их реализация. Выбор альтернатив осуществляется с учетом адаптирующих воздействий, выработанных на основе анализа работы системы на предшествующих шагах.

Для реализации механизма адаптации каждому объекту (фрагменту f_{ij} цепи t_s) сопоставляется автомат адаптации (АА) a_{ijs} , моделирующий поведение объекта адаптации в среде [17,18]. Автомат адаптации имеет две группы состояний: $C_1 = \{c_{1i} | i=1, 2, \dots, g_1\}$ и $C_2 = \{c_{2i} | i=1, 2, \dots, g_2\}$ соответствующие двум альтернативам A_1 и A_2 поведения объекта адаптации в среде. Здесь g_1 и g_2 - глубина памяти (степень доверия). A_1 - остаться в том же слое, A_2 - перераспределение в новый слой. Таким образом выходной алфавит автомата $A = \{A_1, A_2\}$. Входной алфавит $Q = \{+, -\}$ включает возможные отклики среды - «поощрение» (+) и «наказание» (-).

Граф – схема переходов АА показана на рис.2. Знаком (+) помечены переходы под действием сигнала “поощрения”, знаком (–) помечены переходы под действием знака “наказания” (штрафа).

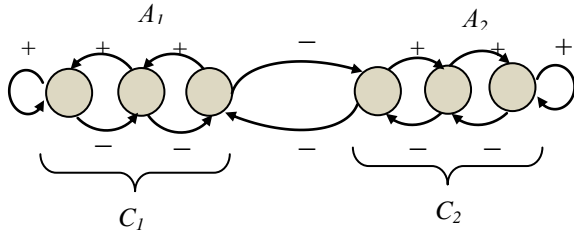


Рис. 2. Граф – схема переходов АА

Отклик среды для АА a_{ijs} в соответствии с состоянием среды и объекта адаптации формируется с учетом двух факторов.

Если после реализации некоторого варианта распределения по слоям фрагмент цепи f_{ij} включает ребро u_k графа G^l с отрицательным значением параметра w_k , учитывающего загрузку ребра u_k , т.е. ребро u_k перенасыщено, то для АА a_{ijs} вырабатывается сигнал “наказание”. Если таких ребер нет, то производится учет второго фактора. В качестве второго фактора рассматриваются текущие диапазоны слоев d_{is} и d_{js} для вершин, связываемых фрагментом цепи f_{ij} .

Если $d_{is} + d_{js} > D_{is} + D_{js}$, то для АА a_{ijs} вырабатывается сигнал “наказание”;

если $d_{is} + d_{js} \leq D_{is} + D_{js}$, то вырабатывается сигнал “поощрение”. (1)

Учет первого фактора связан с выполнением ограничений на максимальную загрузку ребер графа G^l .

Учет второго фактора ориентирован на минимизацию суммарной стоимости межслойных переходов.

Рассмотрим работу алгоритма перераспределения соединений по слоям.

С помощью первого алгоритма получено некоторое начальное решение - распределение всех фрагментов всех сетей по слоям. Для всех вершин, связываемых фрагментами каждой сети, рассчитаны базовые диапазоны слоев D_{is} .

На каждой итерации работа адаптивного алгоритма распределения соединений по слоям осуществляется за четыре такта.

На 1-м такте в соответствии с имеющимся решением для каждого фрагмента f_{ij} каждой сети определяются следующие параметры: диапазоны слоев d_{is} и d_{js} ; значение максимального переполнения γ_{ij}^s среди ребер, входящих в состав фрагмента f_{ij} .

На 2-м такте для каждого АА a_{ijs} вырабатываются отклики среды. Сначала производится учет первого фактора. Если $\gamma_{ij}^s > 0$, то для АА a_{ijs} вырабатывается

сигнал “наказание” и переход к третьему такту работы. Если параметр $\gamma_{ij}^s \leq 0$, то производится учет второго фактора и в соответствии с правилами (1) вырабатывается сигнал “наказание” или “поощрение”.

На 3-м такте по сигналу поощрения или наказания производится переход АА в новое состояние в соответствии с алгоритмом поведения АА.

На 4-м такте анализируется состояние каждого АА a_{ijs} .

Если АА a_{ijs} находится в одном из состояний группы $C_1 = \{c_{1i} | i=1, 2, \dots, g\}$, то фрагмент f_{ij} остается в том же слое.

Если АА a_{ijk} находится в одном из состояний группы $C_2 = \{c_{2i} | i=1, 2, \dots, g\}$, то производится случайный выбор слоя, в который помещается фрагмент f_{ij} . Процесс случайного выбора слоя реализуется следующим образом. Производится пробное размещение фрагмента f_{ij} в каждый из слоев, при этом рассчитываются пробные значения $d_{is}(l)$, $d_{js}(l)$ и $s_{ij}^k(l)$. Рассчитывается оценка пробного размещения фрагмента f_{ij} для каждого слоя l

$$\varepsilon_l = (\delta_l)^\alpha \cdot (\gamma_l)^\beta, \quad (2)$$

где

$$\delta_l = (d_{is} + d_{js}) / (d_{is}(l) + d_{js}(l)),$$

$$\gamma_l = \gamma_{ij}^s / \gamma_{ij}^s(l),$$

α, β - управляющие параметры, которые подбираются экспериментально.

Для каждого слоя рассчитывается вероятность $(P_{ij}^s)_l$ размещения в нем фрагмента f_{ij} .

$$(P_{ij}^s)_l = \varepsilon_l / \sum_l (\varepsilon_l).$$

После расчета вероятностей для всех фрагментов всех сетей фрагменты размещаются по слоям случайным образом, в соответствии с рассчитанными вероятностями.

Таким образом, на каждой итерации после выполнения четырех тактов формируется новое решение. Если оценки нового решения лучше, чем у ранее полученных, то это решение и его оценки запоминаются. Для каждого фрагмента в новом решении производится сравнение диапазонов слоев. Если $d_{ik} + d_{jk} < D_{ik} + D_{jk}$, то параметрам D_{ik} и D_{jk} присваиваются значения параметров d_{ik} и d_{jk} .

Общая оценка временной лежит в пределах $O(n^2)$ - $O(n^3)$.

V. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Экспериментальные исследования проводились на ИВМ РС. Было установлено, что нахождение лучшего решения обеспечивается при значениях $g_1=3, g_2=2$, где g_1 и g_2 - глубина памяти.

Для анализа точности получаемых решений был синтезирован ряд примеров с априори известным оп-

тимальным значением целевой функции. Исследованию подвергались примеры, содержащие до 1000 цепей. При совместной работе алгоритмов в рамках второго подхода вероятность получения оптимального решения составила 0.9.

Сравнительный анализ с другими алгоритмами глобальной трассировки [4, 5, 13, 15] производился на стандартных тестовых примерах и схемах (бенчмарках). Большинство глобальных трассировщиков, использующих второй подход, применяют различные варианты жадной эвристики, или отнимающие много времени методы линейного целочисленного программирования. В работе [16] для решения данной задачи предлагается алгоритм, который, во-первых, генерирует порядок прокладки цепей, основываясь на однослойном результате трассировки, и затем решает задачу распределения по слоям (на каждом шаге выбирается одна цепь), используя динамическое программирование. Слабым звеном подобных подходов является проблема очередности трассируемых цепей.

Для испытания нашего алгоритма мы использовали бенчмарки для шестислойных схем, которые использовались на ISPD'07 для конкурсного тестирования. Полученные результаты сравнивались с результатами трассировщиков **MaizeRouter**, **BoxRouter** и **FGR**, показавших лучшие результаты на этих бенчмарках. Отметим, что в нашей постановке длина соединений не изменялась. Сравнение производилось по критерию суммарной стоимости межслойных переходов с соблюдением ограничения на переполнение. На каждой из бенчмарок в среднем результаты были лучше на 2-5%. При сравнении с улучшенными версиями **BoxRouter 2.0** and **FGR 1.1** [20] результаты были лучше на 2%.

При сравнении с трассировщиком **COLA** [16], показавшем лучшие результаты, в сравнении с вышеупомянутыми трассировщиками, нашим алгоритмом были получены решения на 1-2% лучше.

Общая оценка временной сложности лежит в пределах $O(n^2)$ - $O(n^3)$.

VI. ЗАКЛЮЧЕНИЕ

На основе сравнительного анализа существующих подходов и методов для решения задачи глобальной многослойной трассировки использованы мультиагентные методы интеллектуальной оптимизации. Для усиления сходимости и способности выхода из локальных оптимумов поисковые процедуры базируются на моделировании коллективной адаптации. Приоритет критериев может быть изменен путем несложной модификации откликов среды. В работе рассматривался случай, когда процесс расслоения сводился к распределению по слоям фрагментов цепей, т.е. фрагмент цепи размещался на граничных ребрах графа, лежащих в одном слое. В общем случае можно легко перейти к размещению фрагментов цепей на граничных ребрах графа, лежащих в разных слоях.

- [1] Лебедев Б.К. Интеллектуальные процедуры синтеза топологии СБИС. Таганрог: Изд-во ТРТУ, 2003.
- [2] Лебедев Б.К., Лебедев В.Б. Глобальная трассировка на основе роевого интеллекта // Известия ЮФУ. Изд-во ТТИ ЮФУ, 2010, № 7. С. 32-39.
- [3] Лебедев В.Б., Лебедев О.Б. Генетический алгоритм глобальной трассировки на основе иерархических многохромосомных представлений // Интеллектуальные системы. Коллективная монография / Под ред. В.М. Курейчика. М.: Физматлит, 2009. С. 88-105.
- [4] M. Pan and C. Chu FastRoute 2.0: A high-quality and efficient global router // in Proc. Asia South Pacific Des. Autom. Conf. 2007. P. 250-255.
- [5] M. Cho and D.Z. Pan BoxRouter: A new global router based on box expansion and progressive ILP // in Proc. Des. Autom. Conf. 2006.
- [6] Лебедев О.Б. Трассировка в канале методом муравьиной колонии // Известия ЮФУ. Изд-во ТТИ ЮФУ, 2009. № 2. С. 46-52.
- [7] Лебедев Б.К., Лебедев О.Б. Трассировка в канале на основе коллективной адаптации // Международная НТК IEEE AIS'05, CAD 2005. М.: Изд-во Физматлит, 2005. С. 58-63.
- [8] Курейчик В.М., Лебедев Б.К. Генетический алгоритм трассировки в коммутационном блоке // Известия вузов. Электроника. 2002. № 2. С. 55-66.
- [9] Лебедев Б.К. Метод оптимального распределения ресурсов платы // Техническая кибернетика. 1980. № 1. С. 217.
- [10] Лебедев Б.К. Распределение ресурсов коммутационного поля // Автоматизация проектирования электронной аппаратуры. Таганрог: ТРТИ, 1988. Вып. 1. С. 89 - 92.
- [11] L.E. Liu and C. Sechen Multilayer chip-level global routing using an efficient graph-based Steiner tree heuristic // IEEE Trans. Comput.-Aided Design Integr. Circuits Syst. Oct. 1997. Vol. 18. № 10. P. 1442-1451.
- [12] J.A. Roy and I.L. Markov High performance routing at the nanometer scale // in Proc. Int. Conf. Comput.-Aided Des. 2007. P. 496-502.
- [13] M. Cho, K. Lu, K. Yuan, and D.Z. Pan. BoxRouter 2.0: Architecture and implementation of a hybrid and robust global router // in Proc. Int. Conf. Comput.-Aided Des. 2007. P. 503-508.
- [14] M. M. Ozdal and M. D. F. Wong, "Archer: A history-driven global routing algorithm // in Proc. Int. Conf. Comput.-Aided Des. 2007. P. 488-495.
- [15] M.D. Moffitt MaizeRouter: Engineering an effective global router // in Proc. Asia South Pacific Des. Autom. Conf. 2008. P. 226-231.
- [16] T.-H. Lee and T.-C. Wang. Congestion-Constrained Layer Assignment for Via Minimization in Global Routing // IEEE Trans. Comput.-Aided Design Integr. Circuits Syst. 2008. Vol. 27. № 9. P. 1643-1656.
- [17] Курейчик В.М., Лебедев Б.К., Лебедев В.Б. Адаптация в задачах проектирования топологии // Проблемы разработки перспективных микро- и наноэлектронных систем - 2010 // Сборник научных трудов / под ред. А.Л. Стемповского. М.: ИППМ РАН, 2010. С. 170-177.
- [18] Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Поисковая адаптация: Теория и практика. М.: Физматлит, 2006.
- [19] ISPD 2007 Global Routing Contest. [Online]. Available: <http://www.sigda.org/ispd2007/contest.html>
- [20] FGR 1.1. [Online]. Available: <http://vlsicad.eecs.umich.edu/BK/FGR>