

Распределение соединений по слоям при многослойной глобальной трассировке

Б.К. Лебедев¹, О.Б. Лебедев¹, А.Н. Щелоков²

¹Южный федеральный университет, г. Таганрог, lebedev.ob@mail.ru

²Институт проблем проектирования в микроэлектронике РАН, г. Москва, schan@ipprm.ru

Аннотация – В работе рассматривается алгоритм распределения соединений по слоям в задаче многослойной глобальной трассировки, базирующейся на бионспирированных методах поиска. Решение задачи глобальной трассировки базируется на подходе, включающем две стадии. На первой стадии решается задача однослойной трассировки. На второй – выполняется «распределение соединений по слоям». Разработанные алгоритмы реализованы в виде двух программ: эвристического распределения соединений по слоям и разнесения соединений по слоям методом муравьиной колонии. Решения, полученные в процессе выполнения глобальной трассировки предложенным алгоритмом, на 2-5% лучше, в сравнении с достигнутыми результатами известных трассировщиков. Общая оценка временной сложности лежит в пределах $O(n^2)$ - $O(n^3)$.

Ключевые слова – многослойная глобальная трассировка, распределение соединений по слоям, муравьиный алгоритм, оптимизация.

I. ВВЕДЕНИЕ

Задача многослойной глобальной трассировки (ГТ) в общем случае состоит из предварительного этапа, заключающегося в разбиении общего трассируемого пространства коммутационной среды (КС) на отдельные прямоугольные области (глобальные ячейки) каждого слоя, и основного этапа, заключающегося в распределении соединений по ячейкам и определении наборов цепей на границах каждой ячейки [1].

Оценкой *трассируемости* является соотношение между ресурсами КС и ресурсами, требуемыми для реализации соединений. Ресурсы КС определяются структурой и размерами КС. Ресурсы, требуемые для реализации соединений, определяются размерами, конфигурацией и областью распространения соединений в КС [1]-[2]. Для решения задачи распределения соединений по ячейкам, в модели коммутационной среды (МКС), используется l -слойный граф $G^l=(X^l, U^l)$, где $l \geq 2$ – положительное целое число, определяющее количество доступных слоев (рис. 1). Слои металлизации нумеруются, начиная с нижнего слоя. Вершины графа $x_i \in X^l$ соответствуют глобальным ячейкам МКС $a_i \in A$, $|X^l|=n_x$. Множество ребер U^l можно разделить на два

непересекающихся подмножества U_b^l и U_v^l . $U_b^l \cup U_v^l = U^l$, $|U^l|=n_u$. Если в одном и том же слое две ячейки a_i и a_j смежны (имеют общую границу b_k), то вершины x_i и x_j , соответствующие этим ячейкам, связываются ребром $u_{ik}^l \in U_b^l$. Каждое ребро в U_b^l , которое соединяет две смежные вершины на одном и том же уровне (слое), называется граничным ребром. Если ячейки a_i и a_j расположены в двух смежных слоях и их проекции пересекаются, то вершины x_i и x_j , соответствующие этим ячейкам, связываются межслойным ребром $u_{jk}^l \in U_v^l$. Для каждого ребра u_{jk}^l , связывающего вершины x_i и x_j , задается вес α_k , равный пропускной способности общей границы между ячейками, соответствующими вершинам x_i и x_j .



Рис. 1. Трехслойный граф G^3 , используемый для моделирования процесса глобальной трассировки

В настоящее время к решению задачи многослойной ГТ существуют два основных подхода. При первом подходе все цепи трассируются непосредственно в многослойном пространстве решений [1]-[2]. Однако это метод может оказаться слишком затратным при большом размере задачи. Второй подход заключается в следующем. На первом этапе сначала граф «сжимается» в однослойный граф G^1 , рис. 2.

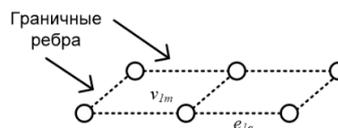


Рис. 2. Однослойный граф G^1 , полученный после «сжатия» трехслойного графа G^3

Каждая вершина, ребро и контакт многослойного графа проецируются на нижний слой. При этом емкости лежащих друг над другом граничных ребер складываются, определяя, таким образом, емкость

граничного ребра на графе G^l . 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 глобальный однослойный трассировщик на однослойном графе G^l формирует решение задачи однослойной трассировки [3]-[4]-[5].

На втором этапе выполняется «распределение соединений по слоям». Каждая цепь распределяется на ребрах исходного многослойного графа G^l . В работе решение задачи ГТ базируется на втором подходе.

Разработанный алгоритм распределения соединений по слоям в задаче многослойной глобальной трассировки, базируется на биоинспирированных методах поиска [5]. С учетом особенностей задачи ГТ разработаны модифицированные механизмы поведения муравьев и структура пространства решений, в рамках которого организован поисковый процесс, базирующийся на моделировании адаптивного поведения муравьиной колонии. Цель состоит в том, чтобы найти такое распределение соединений по слоям, при котором удовлетворяются ограничения на переполнение, а общая стоимость МП минимизирована. Алгоритм гарантирует генерацию распределения по слоям, удовлетворяющую данным ограничениям на перегрузку [5].

II. ПОСТАНОВКА ЗАДАЧИ РАСПРЕДЕЛЕНИЯ СОЕДИНЕНИЙ ПО СЛОЯМ ПРИ МНОГОСЛОЙНОЙ ГЛОБАЛЬНОЙ ТРАССИРОВКЕ

Пусть задано множество цепей $T = \{t_s | s=1, 2, \dots, n_s\}$. Для каждой цепи определяется множество ячеек КС, в которых существуют контакты, связываемые этой цепью. На графе G^l множество ячеек, с размещенными в них контактами, связываемых цепью $t_s \in T$, соответствует множество вершин $X_s \in X^l$. Распределить цепь t_s по ячейкам – значит построить в графе G^l на множестве вершин X_s связывающую сеть es^l .

Межслойные переходы могут располагаться в вершинах дерева Штейнера [6]-[7]. В результате однослойной трассировки для каждой цепи T_s^l на графе $G^l = (X^l, U^l)$, строится сеть es^l в виде дерева Штейнера.

Каждая сеть es^l представляется в виде набора $es^l = \{f_{sk} | k=1, 2, \dots, n_s\}$ непересекающихся двух терминальных соединений f_{sk} , покрывающих дерево Штейнера [7]. Каждое соединение f_{sk} связывает две основные вершины дерева Штейнера. Для каждой вершины $x_i \in es^l$ рассчитывается локальная степень psi_i . Для каждой сети es^l рассчитывается длина $r(f_{sk}) \in es^l$ каждого соединения и суммарная длина сети $r(es^l) = \sum_k r(f_{sk})$ ($k=1, 2, \dots, n_s$).

В работе используется подход, при котором каждое соединение f_k полностью располагается в одном слое.

Для связи соединений одной цепи, расположенных в разных слоях, вводятся межслойные переходы (МП).

Входные данные задачи разнесения соединений по слоям включают l -слойный граф $G^l = (X^l, U^l)$, T^l – множество непротрассированных цепей на графе G^l , однослойный граф $G^1 = (X^1, U^1)$, и решение задачи однослойной трассировки $E^1 = \{e_s^1 | s=1, 2, \dots, n_s\}$ на (G^1, U^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 каждый фрагмент f_k из e_s^l в слой j , где $1 \leq j \leq l$. Решение E^l должно удовлетворять ограничению на максимальную перегруженность. Отметим, что при разнесении соединений по слоям маршрут соединения f_k в любом слое идентичен маршруту в однослойном графе G^1 .

Каждая цепь t_s после ее распределения по ячейкам КС, потребляет определенную часть ресурсов граничных ребер [7]-[8]-[9]. В качестве исходных данных для каждой цепи t_s задается параметр φ_s , равный ширине цепи плюс расстояние между цепями.

Пусть имеется некоторое решение задачи ГТ в одном слое, 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 граничное ребро $u_k^1 \in U^1$. Обозначим через β_k сумму ресурсов, необходимых множеству связывающих сетей E_k для прохождения через граничное ребро u_k . Другими словами, сумму ресурсов, необходимых цепям множества T_k для пересечения границы b_k :

$$\beta_k = \sum \varphi_s (s | t_s \in T_k). \quad (1)$$

Для каждого граничного ребра $u_k^1 \in U^1$ графа G^1 введен параметр $w_k = \alpha_k - \beta_k$, который назовем резервом ресурсов ребра u_k^1 [7]. Найдем в графе G минимальное значение параметра w_k и обозначим его w_{min} т.е. $\forall_k [w_{min} \leq w_k]$.

В качестве первого критерия оптимизации для задачи распределения соединений по слоям используется величина:

$$\sigma = w_{min}. \quad (2)$$

Цель стандартного глобального трассировщика – максимизация параметра σ . Действительно, чем больше резерв ресурсов на ребрах графа G^l , тем легче реализовать соединения при детальной трассировке, и абсолютно неприемлем результат, когда w_j имеет отрицательное значение. Второй не менее важной целью является минимизация общей стоимости ψ межслойных переходов. Тогда критерий оптимизации примет вид:

$$\zeta = \alpha_1 \cdot \sigma + \alpha_2 / \psi, \quad (3)$$

где a_1 и a_2 – коэффициенты критериев, с помощью которых регулируется влияние каждого из критериев на качество решения, σ – оценка загруженности, ψ – общая стоимость межслойных переходов. Цель оптимизации – максимизация критерия ξ .

III. КОНСТРУКТИВНЫЙ АЛГОРИТМ РАЗНЕСЕНИЯ СОЕДИНЕНИЙ ПО СЛОЯМ

В общем случае каждая сеть e_s^l , связывающая множество контактов K_s , представляется на графе G^l в виде дерева Штейнера $S=(X,U)$ с множеством вершин X и множеством ребер U . Отличительной особенностью дерева Штейнера является то, что кроме основных вершин X_1 , соответствующих связываемым контактам, содержит некоторое множество дополнительных вершин X_2 . $X=X_1 \cup X_2$. Как правило, для некоторого подмножества контактов, расположенных в основных вершинах ДШ, задается фиксированное расположение по слоям.

Для каждой вершины $x_i \in G^l$, включающей МП p_i задается диапазон слоев $d_i = \langle l_n^i, l_e^i \rangle$, $l_n^i < l_e^i$ с которыми связан МП p_i , и контакт k_i с $d_i = \langle l_n^i, l_e^i \rangle$. Если

- $d_i = \langle 0 \rangle$, то это означает, что в вершине x_i нет ни контакта, ни МП;

- $d_i = \langle l_n^i \rangle$, то это означает, что в вершине x_i размещен контакт в слое l_n^i , но нет МП;

- $d_i = \langle l_n^i, l_e^i \rangle$, то это означает, что в вершине x_i размещен контакт k_i в слое l_n^i и МП p_i , связывающий слой начиная со слоя l_n^i и заканчивая слоем l_e^i .

IV. АЛГОРИТМ ФОРМИРОВАНИЯ МЕЖСЛОЙНЫХ ПЕРЕХОДОВ И РАЗНЕСЕНИЯ КОНТАКТОВ ПО СЛОЯМ

Алгоритмом для каждой разносимой по слоям сети e_s осуществляется последовательное по шагам разнесение контактов по слоям и ввод межслойных переходов.

1. Для выбранной на графе $G^l = (X^l, U^l)$ сети e_s фиксируется множество основных вершин $X_s^l \in X^l$.

2. Формируется множество вершин $X_{sk}^l \in X_s^l$, в которые помещены контакты, связываемые сетью e_s^l , с указанием номеров слоев размещения.

3. Для каждой вершины $x_i \in X_{sk}^l$, определяем характеристику $d_i(I)$ – диапазон слоев, с которыми связана вершина x_i через МП, $d_i(I) = \langle l_n^i, l_e^i \rangle$, $l_n^i < l_e^i$.

4. Для каждой вершины $x_i \in X_{sk}^l$, определяем подмножество инцидентных ей соединений $T_i(I) = \{f_{ij}^l\}$. $t=1$. $X_{sk}^l(t) = X_{sk}^l$ (t – номер шага).

5. Определяем локальные степени $\rho_i(t) = |T_i(I)|$ всех вершин множества X_{sk}^l .

6. Выбирается вершина $x_i \in X_{sk}^l(t)$ с минимальной локальной степенью $\rho_i(t)$, (если $\rho_i(t) = 1$, то вершине инцидентно одно соединение).

Введем оператор θ – принадлежит.

7. Определяются контакты, связываемые соединением $f_{ij}^l \in T_i(t)$, (связывающее контакт $k_i \theta x_i$, с контактом $k_j \theta x_j$).

8. Определяется d_i и d_j для контактов, размещенных в x_i и x_j .

9. Пусть контакт, принадлежащий вершине x_i , лежит в слое l_1 , а у контакта $k_j \theta x_j$, принадлежащий вершине x_j , $d_j = \langle 0 \rangle$, тогда k_2 , тоже лежит в l_1 .

10. Если пересечение диапазонов d_i и d_j не пусто, т.е. $d_i \cap d_j \neq \emptyset$, то d_i и d_j остаются без изменений, а соединение, связывающее x_i и x_j , помещается в слой $l_k \in (d_i \cap d_j)$ с минимальной загрузкой ребер графа G^l , входящих в состав этого соединения.

11. Если пересечение диапазонов d_i и d_j пусто, т.е. $d_i \cap d_j = \emptyset$, то диапазону d_j присваивается новое значение $d_j = d_i \cup d_j$, равно 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.

Возможны два случая: $l_n^i > l_e^j$, $l_n^j < l_e^i$.

Если $l_n^i > l_e^j$ то значения l_n^j , l_e^i выравниваются двумя способами.

1 способ. Путем уменьшения l_n^i на величину $(l_n^i - l_e^j)$. Диапазон d_i расширяется: $l_n^i = l_n^i - (l_n^i - l_e^j)$. Контакт, принадлежащий вершине x_i , помещается в слой в $l_1 = l_n^i - (l_n^i - l_e^j)$, а контакт $k_j \theta x_j$, принадлежащий вершине x_j , помещается в слой $l_2 = l_e^j$.

2 способ. Путем увеличения l_e^j на величину $(l_n^i - l_e^j)$. Диапазон d_j расширяется: $l_e^j = l_e^j + (l_n^i - l_e^j)$. Контакт, принадлежащий вершине x_i , помещается в слой в $l_1 = l_n^i = l_n^i - (l_n^i - l_e^j)$, а контакт $k_j \theta x_j$, принадлежащий вершине x_j , помещается в слой $l_2 = l_e^j$.

В первом случае увеличивается размер МП, помещенного в вершину x_i , во втором случае увеличивается размер МП, помещенного в вершину x_j .

12. Фрагмент цепи f_{ij}^l , связывающий k_i и k_j , помещается в слой l_1 , а в вершине x_j помещается межслойный переход p_j , связывающий контакт k_j с фрагментом цепи f_{ij}^l .

Обозначим диапазон слоев, связываемых межслойным переходом P_j , как $d_i = \langle l_n - l_e \rangle$, $l_n < l_e$. Для контакта k_i диапазон $d_i = \langle l_1 \rangle$.

Обозначим через d_i^* мощность диапазона d_i . Тогда суммарная стоимость межслойных переходов сети e_s^l определится как: $F_s = \sum_i (d_i^* - 1)$.

V. ПРИМЕР РАБОТЫ АЛГОРИТМА

Для подтверждения эффективности разработанного алгоритма рассмотрим пример. На рис. 3 представлена протрассированная в одном слое цепь, связывающая контакты $\{k_1 - k_8\}$. Сеть представляется в виде набора фрагментов $\{f_{12}, f_{23}, f_{36}, f_{28}, f_{53}, f_{46}, f_{67}\}$.

Предварительно рассчитываются локальные степени всех вершин с размещенными в них

контактами. Для каждой вершины x_i однослойного графа $G^l=(X^l, U^l)$ фиксируется локальная степень ρ_i . Для всех контактов k_i в соответствии с принципиальной схемой фиксируется номера слоев l_i , в которых они расположены.

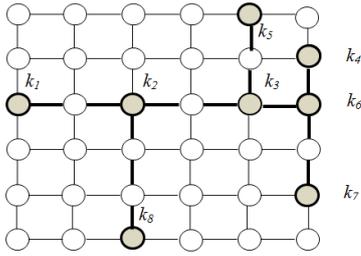


Рис. 3. Формирование межслойных переходов сети

1. Задаются начальные диапазоны слоев контактов: $d_1=<2>$; $d_2=d_3=d_6=<0>$; $d_4=d_8=<3>$; $d_5=<1>$, $d_7=<4>$.

2. Начинаем просматривать вершины с локальной степенью $\rho=1$. В том случае в вершине размещен контакт и вершине инцидентен только один фрагмент цепи.

3. Первой рассмотрим (выбираем) вершину k_1 , связанную с k_2 . Изначально $d_1=<2,0>$, $d_2=<0,0>$. Поскольку $d_1 \cap d_2 = \emptyset$, то d_2 принимает новое значение, $d_2=d_1 \cup d_2 = <2,0>$. $d_1=d_2=<2,0>$. k_1 и k_2 в 2 слое.

4. Далее рассмотрим (выбираем) вершину k_5 . k_5 связана с k_3 . $d_5=<1,0>$. $d_3=<0>$. Поскольку $d_5 \cap d_3 = \emptyset$, то $d_3=d_5 \cup d_3 = <1,0>$. $d_3=d_5=<1,0>$. k_3 и k_5 в 1 слое.

5. Далее рассматривается вершина k_4 , связанная с k_6 . $d_4=<3,0>$, $d_6=<0>$. Поскольку $d_4 \cap d_6 = \emptyset$, то $d_6=d_4 \cup d_6 = <3,0>$. $d_4=d_6=<3,0>$. k_4 и k_6 в 3 слое.

6. Далее выбирается вершина x_7 , определяется смежная ей вершина x_6 . $d_7=<4,0>$. $d_6=<3,0>$. $d_7 \cap d_6 = <3,4>$. $d_6=<3,0>$. $d_7=<3,4>$. k_6 в 3 слое. k_7 и МП с $d_7 = <3,4>$ в 4 слое.

7. Далее рассматривается вершина k_8 , связанная с k_2 . $d_8=<3,0>$. $d_2=<2,0>$. $d_8 \cap d_2 = <2,3>$. k_2 в слое 2. k_8 и МП с $d_8 = <2,3>$ в 3 слое.

8. Далее рассматривается вершина k_3 , смежная вершинам k_2 , k_5 , k_6 . $d_2=<2,0>$, $d_5=<1,0>$, $d_6=<3,0>$. Для связи k_3 с k_2 , k_5 , k_6 в вершине x_3 вводится МП p_3 с диапазоном слоев d_3 , равным объединению диапазоном слоев с k_2 , k_5 , k_6 . $d_2=<2,0>$, $d_5=<1,0>$, $d_6=<3,0>$. Отсюда $d_3 = <1,3>$.

Контакты размещены в следующих слоях: k_1 в 2 слое, $d_1 = <2,0>$; k_2 в 2 слое, $d_2 = <2,0>$; k_3 в 1 слое, в x_3 введен МП p_3 с $d_3 = <1,3>$; k_4 в 3 слое, $d_4 = <3,0>$; k_5 в 1 слое, $d_5 = <1,0>$; k_6 в 3 слое, $d_6 = <3,0>$; k_7 в 4 слое, в x_7 введен МП p_7 с $d_7 = <3,4>$; k_8 в 3 слое, в x_8 введен МП p_8 с $d_8 = <2,3>$.

Межслойные переходы введены в 3 вершинах. Введены 3 МП: МП p_7 с $d_7 = <3,4>$ в вершине x_7 ; МП p_8 с $d_8 = <2,3>$ в вершине x_8 ; МП p_3 с диапазоном слоев $d_3 = <1,3>$.

Соединения размещены по слоям, в соответствии с размещением по слоям контактов: f_{12} в 2 слое; f_{23} в 2 слое; f_{36} в 1 слое; f_{28} в 2 слое; f_{53} в 1 слое; f_{46} в 3 слое; f_{67} в 4 слое.

9. Определяется общая стоимость межслойных переходов (мощности диапазонов). $d^*_3=3$; $d^*_7=2$; $d^*_8=2$.

10. Отсюда суммарная стоимость межслойных переходов сети определится как $F_s=7$. Межслойные переходы будут введены в вершинах k_2 , k_4 , k_6 . Суммарная стоимость межслойных переходов для всех сетей определится как: $F = \sum_s F_s$.

Пусть соединение f_{ij} связывает две вершины x_i и x_j , в которых размещены МП p_i с d_i и МП p_j с d_j . Тогда, если $d_i \cap d_j \neq \emptyset$, то соединение, связывающее x_i и x_j , может быть помещено в один из слоев $l_k \in (d_i \cap d_j)$.

VI. МУРАВЬИНЫЙ АЛГОРИТМ РАЗНЕСЕНИЯ СОЕДИНЕНИЙ ПО СЛОЯМ

Метаэвристика муравьиного алгоритма (МА) основывается на комбинации двух техник: общая схема строится на базовом методе, в которую включается встроенная процедура. Базовый метод заключается в реализации итерационной процедуры поиска лучшего решения, на основе механизмов адаптивного поведения муравьиной колонии. Встроенная процедура – это конструктивный алгоритм построения агентом некоторой конкретной интерпретации решения [7]-[9]-[10]-[11].

С помощью МА решается задача нахождения кратчайшего маршрута в специальном графе. Каждый муравей колонии формирует свой маршрут, являющийся интерпретацией решения некоторой задачи на специальном графе поиска решений (ГПР). При движении муравей метит путь феромоном, и эта информация используется другими муравьями для выбора пути [7]-[9].

Постановка задачи, основные обозначения и определения приведены в разделе 2. Для каждой сети e_s^l рассчитываются локальные степени ρ_i всех вершин, связываемых сетью. Для каждой вершины x_i однослойного графа $G^l=(X^l, U^l)$ входящей в состав сети фиксируется локальная степень.

Для каждой сети e_s^l формируется массив $D_s = \{f_k^l / k=1, 2, \dots, n_d\}$ всех двухтерминальных соединений сети e_s^l , протрассированных на однослойном графе $G^l=(X^l, U^l)$. Каждому двухтерминальному соединению f_k^l соответствует маршрут s_c на графе $G^l=(X^l, U^l)$, построенный в процессе однослойной глобальной трассировки. Каждый маршрут s_c представляет совокупность ребер графа $G^l=(X^l, U^l)$, которой в свою очередь соответствует совокупность границ коммутационного поля, пересекаемых соединением f_k^l . Отметим, что при разнесении соединений по слоям маршрут соединения f_k^l в любом слое идентичен маршруту в однослойном графе G^l . Рассчитывается длина $r(f_k^l)$ каждого соединения f_k^l и суммарная длина сети $r(e_s^l)$.

В качестве поиска решений используется граф поиска решений $H=(V\cup O,E)$. Граф H представляется в виде связанной упорядоченной многостадийной графической структуры Структура графа $H=(V\cup O,E)$, представляется в виде линейной цепочки подграфов, соответствующих сетям e_s^l , упорядоченной по убыванию параметра $r(e_s^l)$. В свою очередь, стадии представляются в виде линейной цепочки подграфов, соответствующих двухтерминальным соединениям f_k^s , упорядоченной по убыванию параметра $r(f_k^s)$, (рис. 4).

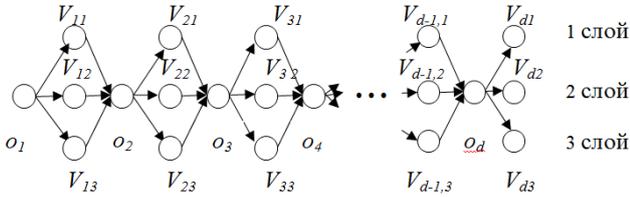


Рис. 4. Граф поиска решений H

Множество вершин $O=\{o_i|i=1,2,\dots,n\}$ соответствует двухтерминальным соединениям f_k^s . Множество вершин $V=\{V_{ij}|i=1,2,\dots,n; j=1,2,\dots,\Theta\}$ соответствует возможным вариантам распределения i -го соединения f_i^s в j -й слой. $V_i=\{V_{ij}|j=1,2,\dots,\Theta\}$ – возможные назначения i -го соединения в слой. Маршрут в графе H имеет вид: $M=o_1V_{12}o_2V_{21}o_3V_{32}o_4V_{43}\dots o_{d-1}V_{d-1,3}o_dV_{d1}$.

Для разбиения маршрута на участки (стадии) соответствующие сетям вводится вектор $O=<o_s|s=1,2,\dots,n>$, значение o_s – номер позиции в M , начиная с которой описывается сеть e_s .

Число вершин равно $n_d n_v$, где n_d – число двухтерминальных соединений всех цепей; $n_v=\Theta$ – число прогнозируемых слоев (вариантов распределения соединения по слоям). Каждая стадия соответствует V_c . Номер группы V_i соответствует номеру двухтерминального соединения f_i^s . Вершины V_{ij} соответствуют номерам слоев, в которые может быть назначено соединение f_i^s . Число вершин в группе V_i равно числу Θ слоев (вариантов распределения соединения по слоям). Между каждой парой соседних групп V_i и V_{i+1} помещается связывающая вершина o_{i+1} . Вершина o_i связана исходящими из нее ориентированными ребрами с каждой вершиной группы V_i . Вершина o_1 является стартовой вершиной. Число вершин o_i равно n_d . Каждая вершина группы V_i связана исходящим ориентированным ребром с вершиной o_{i+1} . Таким образом, каждой вершине o_i (для i от $2\dots n_d$) инцидентно $2n_v$ ребер. Число вершин графа H определится как $n_e=n_d n_v+n_d=(n_d+1)\cdot n_v$. Число ребер графа H определится как $n_r=(n_d-1)\cdot 2n_v+n_v=(2n_d-1)\cdot n_v\cdot f_i^s$.

Необходимо в графе H построить ориентированный маршрут: $M=o_1V_{12}o_2V_{21}o_3V_{32}o_4V_{43}\dots o_{d-1}V_{d-1,3}o_dV_{d1}$, включающий по одной вершине из каждой группы V_i . Построенный маршрут будет определять выбранные варианты распределения всех соединений по слоям. Процесс поиска решений итерационный. Каждая итерация l включает три этапа. На первом этапе каждой итерации каждый агент z_k формирует свое собственное решение,

на втором этапе агенты откладывают на ребрах графа поиска решений H феромон, на третьем этапе осуществляется испарение феромона. В работе используется циклический (ant-cycle) метод муравьиных систем. В этом случае феромон откладывается агентами на ребрах после полного формирования решения [7]-[9]-[12]-[13].

Алгоритм поведения муравьиной колонии.

В соответствии с исходными данными формируется граф поиска решений $H=(V\cup O,E)$.

Агенты обладают памятью. На каждом шаге t в памяти агента z_k имеется:

- количество феромона $f_{iv}(t)$ отложенного на каждом ребре $e_{iv}=(x_i, w_v)$ графа D ;
- список вершин $X_{1k}(t)\in X$, уже включенных в формируемый двудольный подграф $D^k(t)$, и список оставшихся (свободных) вершин $X_{2k}(t)\in X$, $X_{1k}(t)\cup X_{2k}(t)=X$.

1. На начальном этапе на всех вершинах графа H (кроме стартовой) откладывается одинаковое (небольшое) количество феромона, равное Q/n_v , где $n_v=|V|$. Задается число итераций n_l .

2. $l=1$. (l – номер итерации).

3. Задается число муравьев n_a , которые помещаются в стартовую вершину o_1 .

4. (Алгоритм муравья). Каждый муравей a_σ строит на графе H маршрут M_σ , начинающийся с вершины o_1 , и включающий по одной вершине из каждой группы V_c . Рассчитывается оценка F_σ маршрута, т.е. рассмотренные выше критерии оптимизации.

5. После построения всеми муравьями маршрутов, каждый муравей a_σ откладывает на вершинах построенного маршрута M_σ феромон в количестве: $\Delta\tau_\sigma(l)=\varphi(Q/F_\sigma)$, где Q – базовое количество феромона, откладываемое муравьем a_σ на вершинах маршрута M_σ , построенного на итерации l . φ – коэффициент, задаваемый априори (подбирается экспериментально).

6. После того, как каждый агент отложил феромон, происходит общее испарение феромона на вершинах графа H в соответствии с формулой $h_j=h_j(1-\rho)$, где ρ – коэффициент обновления, h_j – суммарное количество феромона, отложенного муравьями на вершине $v_j\in V$.

7. Выбор лучшего решения, полученного на протяжении всех выполненных итераций.

8. Если все итерации выполнены ($l=n_l$), то конец работы алгоритма, в противном случае $l=l+1$ и переход к пункту 3 для выполнения очередной итерации.

Алгоритм муравья (пункт 4 алгоритма поведения муравьиной колонии).

1. $\sigma=1$. (σ – номер агента).

2. $t=0$, $M_\sigma(t)=\emptyset$. $v(t)=x_0$. (t – номер шага, $M_\sigma(t)$ – маршрут на шаге t , $v(t)$ – вершина, вошедшая в маршрут на шаге t).

3. $c=t+1$. Агент a_σ определяет множество $V_\sigma(t+1) \in V_c$ вершин соседней группы V_c , связанных в графе H с вершиной $v(t)$, кандидатов для включения в свой маршрут $M_\sigma(t+1)$.

4. Рассчитывается стоимость CR_z каждой вершины $v_z \in V_\sigma(t+1)$. Пусть вершине $v_z \in V_c$ соответствует номер слоя i , в который помещается двухтерминальное соединение d_c (s_c – маршрут в графе G^i). В маршруте s_c на графе G^i отыскивается ребро с минимальной пропускной способностью α_z . Другими словами α_z – минимальная пропускная способность среди ребер маршрута s_c на графе G^i . По формуле $CR_z = (h_z)^\lambda \cdot (\alpha_z)^\gamma$ при мультипликативной свертке, либо по формуле $CR_z = \lambda \cdot (h_z) + \gamma \cdot (\alpha_z)$, при аддитивной свертке определяется потенциальная стоимость CR_z каждой вершины $v_z \in V_\sigma(t+1)$, где h_z – количество феромона, отложенного на вершине v_z , λ и γ – управляющие параметры, которые подбираются экспериментально.

5. Для каждой вершины $v_z \in V_\sigma(t+1)$ рассчитывается вероятность P_z её включения в формируемый маршрут $M_\sigma(t+1)$. $P_z = CR_z / \sum_z CR_z$ для ($z/v_z \in V_\sigma(t+1)$).

6. Случайным образом в соответствии с рассчитанными вероятностями выбирается вершина $v_z \in V_\sigma(t+1)$, которая включается в маршрут $M_\sigma(t+1)$.

7. Если маршрут $M_\sigma(t+1)$ построен, то переход к пункту 8. Иначе $t=t+1$, фиксируется «хвост» маршрута $M_\sigma(t+1)$: $v(t)=v_z$ и переход к пункту 3.

8. Осуществляется распределение соединений по областям в соответствии с вариантами, задаваемыми построенным маршрутом M_σ . Рассчитывается значение критерия F_3 , которое и будет оценкой F_σ маршрута M_σ .

9. Если $\sigma < (n_\sigma - 1)$, то $\sigma = \sigma + 1$ и переход к пункту 2, иначе переход к пункту 10.

10. Окончание этапа построения муравьями маршрутов.

VII. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Целью экспериментальных исследований является выявление зависимостей и влияния на качество глобальной трассировки различных комбинаций управляющих параметров и структур. Разработанные алгоритмы реализованы в виде двух программ:

- Эвристического распределения соединений по слоям.

- Разнесения соединений по слоям методом муравьиной колонии.

Для нахождения наилучшего сочетания таких параметров, как объем популяции M и количество итераций T , использовались точные решения в прямоугольной метрике для тестовых наборов estein OR-library [14], приведенные в работе D. Warne [15]. Данный тестовый набор представляет собой

совокупность задач с размерностью 10, 20, 30, 40, 50, 60, 70, 90, 100, 250 и 500 точек (по 15 задач на каждую размерность). Результаты исследований приведены в таблице 1. На основе обработки экспериментальных данных была построена средняя зависимость степени качества от числа итераций (рис. 5).

Таблица 1

Результаты исследований

Число вершин	Качество решения			Число генераций T		
	Опт.	< 0,1%	> 0,1%	Min	Max	Средн.
10	8	1	1	20	100	80
20	14	4	2	20	115	90
30	15	5	0	30	200	110
40	5	4	1	32	150	120
50	6	4	0	40	175	150
60	6	4	0	70	200	150
80	68	6	6	60	150	145
100	82	10	8	55	135	120
250	200	20	30	50	140	125
500	460	400	60	60	135	110

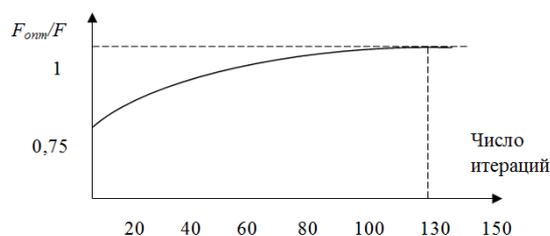


Рис. 5. Зависимость степени качества алгоритма от числа итераций

В результате экспериментов установлено, что при объеме популяции $M=100$ алгоритм сходится на 130 итерации. В среднем результат, полученный предложенным алгоритмом на 130-ой итерации, отличается от оптимального на 0.15 процента.

Сравнение производилось по величине отклонения в % значения критерия оптимизации от оптимального по методике, изложенной в [16]-[17]-[18]. Испытания разработанного алгоритма производился на бенчмарках для шестислойных схем. Полученные результаты испытаний сравнивались с результатами трассировщиков MaizeRouter, VoxRouter и FGR, показавших лучшие результаты на этих бенчмарках. Сравнение производилось по показателям переполнения γ на ребрах модели коммутационной среды (l -слойный граф $G^l=(X^l, U^l)$): total – суммарное переполнения γ на ребрах многослойной коммутационной среды (МКС); max – максимальное переполнения среди ребер МКС и показателю суммарной длины соединений. Соединение между соседними ячейками на одном слое является одной единицей wirelength. Соединение между соседними слоями (т.е. сквозное) представляет собой три единицы wirelength. В таблице 2 приведены результаты сравнительных экспериментов.

Содержимое таблицы включает: имя каждой тестовой задачи – (benchmarks); значение переполнения OF (overflow): суммарное – (total) и в скобках максимальное среди ребер МКС – (max):

суммарная длина соединений WL (wirelength) в условных единицах.

Таблица 2

Сравнительная оценка работы алгоритмов

Bench	MaizeRouter + COLA		BoxRouter + COLA		FGR + COLA		BoxRouter 2.0 + COLA		FGR 1.1 + COLA		ОГТА + РСМК	
	OF	WL	OF	WL	OF	WL	OF	WL	OF	WL	OF	WL
adaptc1	0 (0)	116.48	0 (0)	111.08	54 (2)	108.89	0 (0)	91.29	0 (0)	87.90	0 (0)	87.12
adaptc2	0 (0)	109.93	0 (0)	107.44	36 (2)	105.70	0 (0)	92.62	0 (0)	89.54	0 (0)	89.21
adaptc3	0 (0)	248.96	0 (0)	251.76	0 (0)	246.10	0 (0)	206.46	0 (0)	199.8	0 (0)	191.6
adaptc4	0 (0)	225.99	0 (0)	255.69	0 (0)	223.94	0 (0)	185.82	0 (0)	182.40	0 (0)	180.4
adaptc5	2 (2)	327.89	0 (0)	310.85	2156	304.89	0 (0)	265.70	0 (0)	258.23	0 (0)	248.3
newblue1	1348	107.01	400 (2)	104.55	2296	102.14	394	92.50	236 (2)	89.38	228 (2)	85.46
newblue2	0 (0)	169.49	0 (0)	168.27	0 (0)	166.44	0 (0)	134.05	0 (0)	131.61	0 (0)	124.6
newblue3	32564 (208)	224.69	38958 (364)	218.56	50720 (366)	213.17	38958 (364)	168.65	38372 (400)	162.65	32584 (206)	152.4

На каждой из бенчмарков в среднем результаты были лучше на 2-5%. При сравнении с улучшенными версиями BoxRouter 2.0 and FGR 1.1 [18]-[19] результаты были лучше на 2%. При сравнении с трассировщиком COLA [18]-[19], были получены решения на 1- 2% лучше. Общая оценка временной сложности лежит в пределах $O(n^2)$ - $O(n^3)$.

VIII. ЗАКЛЮЧЕНИЕ

В работе предложен подход к глобальной трассировке, при котором на первом этапе все цепи трассируются непосредственно в однослойном пространстве решений. На втором этапе выполняется стадия «распределения соединений по слоям». Такой подход позволяет снизить комбинаторную сложность алгоритма глобальной трассировки. Для решения задачи распределения соединений по слоям разработан алгоритм основе метода муравьиной колонии, заключающемся в декомпозиции связывающей сети и представлении ее в виде совокупности двухтерминальных соединений. В отличие от канонической парадигмы МК, предложена модифицированная жадная стратегия построения ориентированного маршрута на графовой модели представления решения. Эксперименты показали, что решения, по сравнению с существующими алгоритмами, получались лучше в среднем на 0,5-1%. Задача глобальной трассировки решена путем последовательной гибридизации двух алгоритмов, функционирующих последовательно один за другим, что позволило улучшить результаты решения эвристическим и муравьиным алгоритмом распределения соединений по слоям на 3,2 %.

ПОДДЕРЖКА

Работа выполнена при финансовой поддержке гранта РФФИ № 20-07-00260 а.

ЛИТЕРАТУРА

[1] Немудров В. Системы-на-кристалле. Проектирование и развитие. М: Техносфера, 2004. 157 с.
 [2] Казеннов Г.Г. Основы проектирования интегральных схем и систем. М.: Лаборатория знаний, 2005. 295 с.

[3] Alpert C.J., Mehta D.P., Sapatnekar S.S., Handbook of Algorithms for Physical Design Automation. Boston, MA: Auerbach, 2009. 245 p.
 [4] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие. М: Издательство МГТУ им. Н.Э. Баумана, 2014. 448 с.
 [5] Лебедев Б.К., Лебедев В.Б. Глобальная трассировка на основе роевого интеллекта // Известия ЮФУ. Изд-во ТТИ ЮФУ. 2010. №7. С. 32-39.
 [6] Pan M., Chu C. FastRoute 2.0: A high-quality and efficient global router // Proc. Asia South Pacific Des. Autom. Conf. 2007. pp. 250–255.
 [7] Лебедев О.Б. Модели адаптивного поведения муравьиной колонии в задачах проектирования. Таганрог: Изд-во ЮФУ, 2013. 199 с.
 [8] Cho M., Pan D.Z. BoxRouter: A new global router based on box expansion and progressive ILP // Proc. Des. Autom. Conf. 2006. pp. 351–356.
 [9] Лебедев О.Б. Трассировка в канале методом муравьиной колонии // Известия ЮФУ. Изд-во ТТИ ЮФУ. 2009. №2. С. 46-52.
 [10] Roy J.A., Markov I.L. High performance routing at the nanometer scale // Proc. Int. Conf. Comput.-Aided Des. 2007. pp. 496–502.
 [11] Лебедев В.Б., Лебедев О.Б. Генетический алгоритм глобальной трассировки на основе иерархических многохромосомных представлений // Интеллектуальные системы. М.: Физматлит. 2009. С. 88-105.
 [12] Гаврилов С.В., Железников Д.А., Хватов В.М. Решение задач трассировки межсоединений с ресинтезом для реконфигурируемых систем на кристалле // Изв. вузов. Электроника. 2017. Т.22. №3. С. 266-275.
 [13] Заплетина М.А., Железников Д.А., Гаврилов С.В. Иерархический подход к трассировке реконфигурируемой системы на кристалле островного типа // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2020. Выпуск 3. С. 16-21. doi:10.31114/2078-7707-2020-3-16-21
 [14] Лебедев Б.К., Лебедев О.Б., Лебедева Е.О. Параллельно-последовательный биоинспирированный алгоритм построения минимального дерева Штейнера // Известия ЮФУ. Технические науки. Ростов-на-Дону: Изд-во ЮФУ. 2019. № 4. С. 174-186.
 [15] Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Трассировка соединений в канале на основе моделей адаптивного поведения муравьиной колонии // Известия РАН. Теория и Системы Управления. 2015. № 2. С. 84–101.
 [16] Рыженко Н.В., Быков С.А., Сорокин А.А. Трассировка битовых элементов памяти с автоматическим построением ограничений на границах ячеек // Проблемы разработки перспективных микро- и наноэлектронных систем (МЭС). 2016. № 1. С. 144-150.
 [17] Cho M., Lu K., Yuan K. BoxRouter 2.0: Architecture and implementation of a hybrid and robust global router // Proc. Int. Conf. Comput.-Aided Des. 2007. pp. 503–508.
 [18] Ozdal M., Wong M. Archer: A history-driven global routing algorithm // Proc. Int. Conf. Comput.-Aided Des. 2007 pp. 488–495.
 [19] Moffitt M. MaizeRouter: Engineering an effective global router // Proc. Asia South Pacific Des. Autom. Conf. 2008, pp. 226–231.

Distribution of Connections by Layers in Multi-layer Global Routing

B.K. Lebedev¹, O.B. Lebedev¹, A.N. Schelokov²

¹Southern Federal University, Taganrog, Southern Federal University, lebedev.ob@mail.ru

²Institute for automation design systems of RAS, Moscow, schan@ippm.ru

Abstract – The paper considers an algorithm for the distribution of compounds by layers in the problem of multilayer global tracing based on bioinspired search methods. The solution of the global tracing problem is based on an approach that includes two stages. At the first stage, the problem of single-layer tracing is solved. On the second – «distribution of connections by layers» is performed. The main goal of the global tracer is to minimize the maximum overflow γ or limit it. As a result of single-layer tracing for each circuit on the graph, a network is built in the form of a Steiner tree. The vias can be located at the vertices of the Steiner tree. Each network is represented as a set of fragments. Each fragment represents a two-terminal connection connecting two vertices of the Steiner tree x_i and x_j . The task of distributing connections over layers is to transform each network from a given single-layer routing solution to its analogue in an l -layer global routing solution in such a way that, when projected, we obtain the same topology result as for the original network. The solution must satisfy the maximum congestion constraint. At the first stage, independently of each other, each network from the given single-layer routing solution is transformed into its analogue in the l -layer global routing solution with minimization of the cost of interlayer transitions. At the second stage, adaptive search methods are used to redistribute connections across layers to fulfill the maximum congestion limit while minimizing the total number of vias. The solutions obtained in the process of performing global tracing by the proposed algorithm are 2-5% better in comparison with the results achieved by known tracers. The overall time complexity estimate lies within $O(n^2)$ - $O(n^3)$.

Keywords – multilayer global tracing, connection distribution by layers, ant colony algorithm, optimization.

REFERENCES

- [1] Nemudrov V. Sistemy-na-kristalle. Proyektirovaniye i razvitiye (Systems-on-a-chip. Design and development). M: Tekhnosfera, 2004. 157 c.
- [2] Kazennov G.G. Osnovy proyektirovaniya integral'nykh skhem i sistem (Fundamentals of design of integrated circuits and systems). M.: Binom, Laboratoriya znaniy, 2005, 295 c.
- [3] Alpert C.J., Mehta D.P., Sapatnekar S.S., Handbook of Algorithms for Physical Design Automation. Boston, MA: Auerbach, 2009. 245 p.
- [4] Karpenko A.P. Sovremennyye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennyye prirodoy: uchebnoye posobiye (Modern search engine optimization algorithms. Algorithms inspired by nature: a tutorial). M: Izdatel'stvo MSTU N.E. Bauman, 2014. 448 p.
- [5] Lebedev B.K., Lebedev V.B. Global'naya trassirovka na osnove royevogo intellekta (Global tracing based on swarm intelligence) // Izvestiya YUFU. Izd-vo TTI YUFU. 2010. № 7. pp. 32-39.
- [6] Pan M., Chu C. FastRoute 2.0: A high-quality and efficient global router // *Proc. Asia South Pacific Des. Autom. Conf.* 2007. pp. 250–255.
- [7] Lebedev O.B. Modeli adaptivnogo povedeniya murav'inoy kolonii v zadachakh proyektirovaniya (Models of adaptive behavior of an ant colony in design problems). Taganrog: Izd-vo YUFU, 2013. 199 p.
- [8] Cho M., Pan D.Z. BoxRouter: A new global router based on box expansion and progressive ILP // *Proc. Des. Autom. Conf.* 2006. pp. 351–356.
- [9] Lebedev O.B. rassirovka v kanale metodom murav'inoy kolonii (Tracing in the channel by the ant colony method) // *zvestiya YUFU. Izd-vo TTI YUFU.* 2009. № 2. pp. 46-52.
- [10] Roy J.A., Markov I.L. High performance routing at the nanometer scale // *Proc. Int. Conf. Comput.-Aided Des.* 2007. pp. 496–502.
- [11] Lebedev V.B., Lebedev O.B. Geneticheskiy algoritm global'noy trassirovki na osnove iyerarkhicheskikh mnogokhromosomnykh predstavleniy (Genetic algorithm for global tracing based on hierarchical multichromosomal representations) // *Intellektual'nyye sistemy.* M.: Fizmatlit. 2009. pp. 88-105.
- [12] Gavrillov S.V., Zheleznirov D.A., Khvatov V.M. Resheniye zadach trassirovki mezhsoyedeniy s resintezom dlya rekonfiguriruyemykh sistem na kristalle (Solving the problems of tracing interconnections with resynthesis for reconfigurable systems on a chip) // *Izvestiya vuzov. Elektronika.* 2017. V.22. № 3. pp. 266-275.
- [13] Zapletina M.A., Zheleznirov D.A., Gavrillov S.V. The Hierarchical Approach to Island Style Reconfigurable System-on-a-chip Routing // *Problems of Perspective Micro- and Nanoelectronic Systems Development - 2020.* Issue 3. P. 16-21. doi:10.31114/2078-7707-2020-3-16-21
- [14] Lebedev B.K., Lebedev O.B., Lebedeva E.O. Parallelnoposledovatel'nyy bioinspirirovanny algoritm postroyeniya minimal'nogo dereva Shteynera (Parallel-sequential bioinspired algorithm for constructing a minimal Steiner tree) // *Izvestiya YUFU. Tekhnicheskkiye nauki.* Rostov-na-Donu: Izd-vo YUFU. 2019. №4. pp. 174-186.
- [15] Kureichik V.M., Lebedev B.K., Lebedev O.B. Trassirovka soyedineniy v kanale na osnove modeley adaptivnogo povedeniya murav'inoy kolonii (Connection tracing in a channel based on models of adaptive behavior of an ant colony) // *Izvestiya RAN. Teoriya i Sistemy Upravleniya.* 2015. №2. pp. 84–101.
- [16] Ryzhenko N.V., Bykov S.A., Sorokin A.A. Routing of Memory Bits Cells with Automated Construction of Boundary Layout Constraints // *Problems of Perspective Micro- and Nanoelectronic Systems Development - 2016.* Proceedings / edited by A. Stempkovsky, Moscow, IPPM RAS, 2016. Part 1. P. 144-150.
- [17] Cho M., Lu K., Yuan K. BoxRouter 2.0: Architecture and implementation of a hybrid and robust global router // *Proc. Int. Conf. Comput.-Aided Des.* 2007. pp. 503–508.
- [18] Ozdal M., Wong M. Archer: A history-driven global routing algorithm // *Proc. Int. Conf. Comput.-Aided Des.* 2007 pp. 488–495.
- [19] Moffitt M. MaizeRouter: Engineering an effective global router // *Proc. Asia South Pacific Des. Autom. Conf.* 2008, pp. 226–231