

Процедуры канальной трассировки, на основе гибридизации роевого интеллекта с генетическим поиском*

Б.К. Лебедев, В.Б. Лебедев

Таганрогский технологический институт Южного федерального университета,

lbk@tsure.ru

Аннотация — Предлагаются новые технологии, принципы и механизмы решения задачи трассировки, использующие математические методы, в которых заложены принципы природных механизмов принятия решений. Используется компактное символьное представление решения задачи трассировки. Поисковый процесс, базирующийся на моделировании адаптивного поведения роя частиц, организован в пространстве решений с неупорядоченными лингвистическими шкалами измерений. Для усиления сходимости алгоритма и способности выхода из локальных оптимумов организация поисковой процедуры произведена на основе гибридизации роевого интеллекта с генетическим поиском. По сравнению с существующими алгоритмами достигнуто улучшение результатов.

Ключевые слова — Канальная трассировка; роевой интеллект; неупорядоченное лингвистическое шкалирование; генетический поиск; 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. Задача канальной трассировки (ЗКТ) [1] является хорошо изученной, и существуют сотни подходов к ее решению: алгоритмы левого конца, “жадные”, графовые алгоритмы, иерархические и т.д. [2]. Однако наличие новых тенденций в проектировании СБИС вызывает потребность в разработке новых подходов к ее реализации. В последнее время для решения различных «сложных» задач, к которым относятся и задачи канальной трассировки, всё чаще используются способы, основанные на применении методов искусственного интеллекта [2,3]. Особенно наблюдается стремительный рост интереса к разработке алгоритмов, инспирированных природными системами [3,4,5]. Од-

ними из новых направлений таких методов являются мультиагентные методы интеллектуальной оптимизации, базирующиеся на моделировании коллективного интеллекта [4-7]. Оптимизация с использованием роя частиц (Particle Swarm Optimization, PSO) - это метод поиска, который базируется на понятии популяции, и моделирует поведение птиц в стае и косяков рыб [8-10]. Рой частиц рассматривается как многоагентная система, в которой каждый агент (частица) функционирует автономно по очень простым правилам. В таких случаях говорят о роевом интеллекте (Swarm intelligence).

В работе излагается метод решения задачи канальной трассировки (КТ) на основе роевого интеллекта и генетической эволюции. Используется компактное символьное представление решения задачи трассировки. Предложена композитная архитектура многоагентной системы бионического поиска.

II. ФОРМИРОВАНИЕ СИМВОЛЬНОГО ПРЕДСТАВЛЕНИЯ РЕШЕНИЯ ЗАДАЧИ КАНАЛЬНОЙ ТРАССИРОВКИ

При КТ каждая цепь, представляется в виде набора горизонтальных и вертикальных фрагментов. Задача трассировки в канале рассматривается как задача распределения фиксированного множества горизонтальных участков $F = \{f_i | i=1, 2, \dots, N\}$ в множестве магистралей $M = \{m_j | j=1, 2, \dots, L\}$. На взаимное расположение горизонтальных фрагментов накладываются ограничения, задаваемые с помощью графа вертикальных ограничений (ГВО) $G_V=(X, U)$ и графа горизонтальных ограничений (ГГО) $G_H=(X, W)$ [1,2]. Вершина x_i в графе G_V и G_H соответствует фрагменту f_i . Наличие дуги (x_i, x_j) в G_V означает, что f_i в канале выше f_j для исключения наложения вертикальных отрезков. Ребро (x_i, x_j) в G_H означает, что x_i и x_j не могут быть размещены в одной магистрали.

* Работа выполнена при финансовой поддержке программы РНП.2.1.2.1652, Грант РФФИ № 09-01-00509.

Цель оптимизации – минимизация числа магистралей NM , необходимых для размещения в них горизонтальных участков с соблюдением ограничений.

Разобьем множество горизонтальных фрагментов F на подмножества F_k , $F = \{F_k | k=1, 2, \dots, V\}$ в соответствии со следующими правилами:

$$1. \cup F_k = F, \quad \forall (ij)[F_i \cap F_j = \emptyset].$$

2. Любые два участка $f_i \in F_k$ и $f_j \in F_k$ находятся в горизонтальном конфликте и не могут быть помещены в одну магистраль.

3. Подмножества F_k сформированы и пронумерованы так, что все левые концы участков F_k расположены левее всех левых концов участков F_{k+1} .

Алгоритм разбиения F на F_k имеет линейную трудоемкость [6].

Пусть задан канал (рис.1), произведено разбиение цепи на фрагменты и построены ГВО и ГГО.

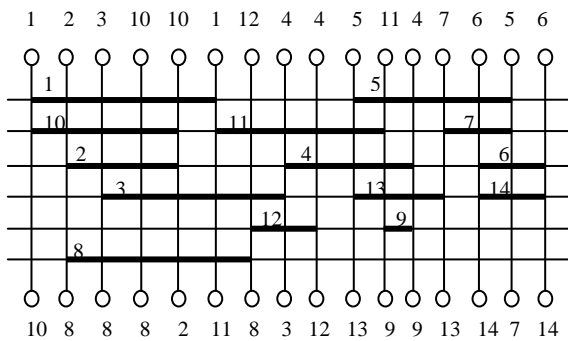


Рис. 1. Множество горизонтальных фрагментов

Множество горизонтальных фрагментов F , в соответствии с приведенными правилами, разбивается на подмножества: $F_1 = \{f_1, f_{10}, f_2, f_8, f_3\}$; $F_2 = \{f_{11}, f_{12}, f_4\}$; $F_3 = \{f_5, f_{13}, f_9\}$; $F_4 = \{f_7, f_6, f_{14}\}$. Обозначим через α_k – мощность подмножества F_k . Среди α_k отыскивается α_{max} с максимальным значением, $\forall k[\alpha_{max} \geq \alpha_k]$. Произведем нормирование подмножеств F_k , путем добавления в них $(\alpha_{max} - \alpha_k)$ нулей в случайной последовательности. Получим $F_1 = \{f_1, f_{10}, f_2, f_8, f_3\}$; $F_2 = \{f_{11}, 0, f_{12}, 0, f_4\}$; $F_3 = \{f_5, 0, f_{13}, f_9, 0\}$; $F_4 = \{f_7, 0, f_6, f_{14}, 0\}$.

Для отображения решения задачи канальной трассировки (ЗКТ) формируется матрица $D = \{d_{ik} | i=1, 2, \dots, \alpha_{max}; k=1, 2, \dots, V\}$, число столбцов которой равно числу V подмножеств F_k , а число строк равно α_{max} . В каждый k -й столбец D_k в случайной последовательности заносятся элементы множества F_k . Для удобства на рис.2 элементам матрицы D присвоены значения индексов фрагментов. Процедура получения решения ЗКТ по матрице D осуществляется следующим образом. Магистрали заполняются последовательно, начиная с первой. Последовательно по строкам, начиная с первой, а в пределах строки слева направо просматриваются элементы матрицы D . Для каждого выбранного элемента матрицы (фрагмента) определяется

возможность его размещения в текущей магистрали в соответствии с G_V и G_H . Если возможно, то фрагмент помещается в текущую магистраль и удаляется из D . По окончании просмотра матрицы D осуществляется сжатие всех столбцов, из которых удалялись фрагменты, снизу-вверх. После этого осуществляется переход к заполнению следующей магистрали. Процесс последовательного заполнения магистралей $m_1 - m_3$ показан на рис.2.

| | | | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| d_1 | d_2 | d_3 | d_4 | d_1 | d_2 | d_3 | d_4 | d_1 | d_2 | d_3 | d_4 |
| 1 | 11 | 5 | 7 | 10 | 11 | 0 | 7 | 2 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 2 | 0 | 13 | 0 | 8 | 12 | 13 | 6 |
| 2 | 12 | 13 | 6 | 8 | 12 | 9 | 6 | 3 | 0 | 9 | 14 |
| 8 | 0 | 9 | 14 | 3 | 0 | 0 | 14 | - | 4 | 0 | 0 |
| 3 | 4 | 0 | 0 | - | 4 | - | 0 | - | - | - | - |

$\{1,5\} \rightarrow m_1$ $\{10,11,7\} \rightarrow m_2$ $\{2,13,6\} \rightarrow m_3$

Рис. 2. Процесс заполнения магистралей

В результате применения вышеуказанной процедуры к матрице D магистрали будут заполнены следующим образом: $m_1 = \{1,5\}$; $m_2 = \{10,11,7\}$; $m_3 = \{2,13,6\}$; $m_4 = \{9,14\}$; $m_5 = \{4\}$; $m_6 = \{3\}$; $m_7 = \{12\}$; $m_8 = \{8\}$.

Очевидно, что порядок распределения фрагментов по магистралям и их число полностью определяется порядком расположения элементов в D и существует некоторая D_{opt} , которой соответствует оптимальное решение. На рис.3 представлена такая матрица D_{opt} , и соответствующее ей решение.

$$D_{opt} =$$

| | | | |
|-------|-------|-------|-------|
| d_1 | d_2 | d_3 | d_4 |
| 1 | 4 | 0 | 6 |
| 10 | 12 | 5 | 0 |
| 2 | 11 | 0 | 7 |
| 3 | 0 | 13 | 0 |
| 8 | 0 | 9 | 14 |

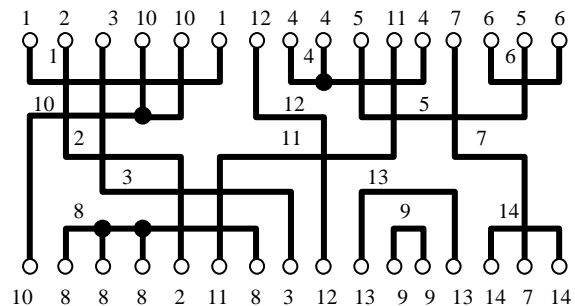


Рис. 3. Оптимальное решение

Таким образом, матрица D является символьным представлением решения задачи канальной трассировки. В работе пространство решений представляется множеством матриц D . Поиск решения сводится к поиску такой матрицы D , т.е. к поиску такого порядка

расположения элементов в D , который оптимизируют показатель качества (критерий).

III. МЕХАНИЗМЫ ТРАССИРОВКИ НА ОСНОВЕ РОЕВОГО ИНТЕЛЛЕКТА

В методе оптимизации роем частиц агентами являются частицы в пространстве параметров задачи оптимизации. Основу поведения роя частиц составляет самоорганизация, обеспечивающая достижение общих целей роя на основе низкоуровневого взаимодействия. Каждая частица связана со всем роем, может взаимодействовать со всем роем и тяготеет к лучшему решению роя. Процесс поиска решений итерационный. На каждой итерации каждая частица перемещается в новую позицию. В каждый момент времени (на каждой итерации) частицы имеют в этом пространстве некоторое положение и вектор скорости. Для каждого положения частицы вычисляется соответствующее значение целевой функции. При определении следующего положения частицы учитывается информация о наилучшей частице из числа «соседей» данной частицы, а также информация о данной частице на той итерации, когда этой частице соответствовало наилучшее значение целевой функции, и на этой основе по определенным правилам частица меняет свое положение и скорость в пространстве поиска [6].

В каноническом методе роя частиц [7], используемом метрическое шкалирование, новая позиция частицы i определяется как

$$x_i(t+1) = x_i(t) + v_i(t+1), \quad (3)$$

где $v_i(t+1)$ скорость перемещения частицы i из позиции $x_i(t)$ в позицию $x_i(t+1)$. Начальное состояние определяется, как $x_i(0)$, $v_i(0)$.

Введем обозначения:

- $f_i(t)$ - текущее значение целевой функции частицы i в позиции $x_i(t)$ (решение $D_i(t)$);
- $f^*(t)$ - лучшее значение целевой функции среди частиц роя в момент времени t , а $x^*(t)$ - позиция с этим значением (решение $D(t)$);
- $x_i^*(t)$ - лучшая позиция частицы i , которую она посещала с начала первой итерации (решение $D_i^*(t)$), а $f_i^*(t)$ - значение целевой функции в этой позиции - лучшее значение частицы i .

Лучшие частицы, с точки зрения целевой функции, объявляются «центром притяжения». Векторы скоростей всех частиц устремляются к этим центрам.

В работе в отличие от стандартной парадигмы метода роя частиц, вместо метрической (числовой) шкалы используются *неупорядоченные лингвистические шкалы измерений*.

В нашем случае позиция $x_i(t)$ соответствует i -ому решению, задаваемому матрицей $D_i(t)$. Число осей в пространстве решений равно числу столбцов $D_{ik}(t)$ в матрице $D_i(t) = \{D_{ik}(t) \mid k=1, 2, \dots, V\}$. Точками отсчета на каждой оси k являются значения лингвистической

переменной $x_{ik}(t)$, неупорядоченные друг относительно друга. Значением $x_{ik}(t)$ является некоторая комбинация взаимного расположения элементов в k -м столбце $D_{ik}(t)$ матрицы $D_i(t)$. Позиция $x_i(t)$, то есть матрица $D_i(t)$ является решением, а скорость $v_i(t+1)$ рассматривается как средство изменения решения. В отличие от канонического метода роя частиц, в нашем случае скорость $v_i(t+1)$ не может быть представленной в виде аналитического выражения. В качестве скорости $v_i(t+1)$ выступает *процедура перемещения*, суть которой заключается в изменения взаимного расположения элементов в столбцах матрицы $D_i(t)$. Частица i перемещается в новую позицию с новыми значениями лингвистических переменных $x_{ik}(t)$.

Пусть «центром притяжения» частицы i , размещенной в позиции $x_i(t)$, является позиция $x_0(t)$. Рассмотрим работу процедуры перемещения, приводящую к модификации решения $D_i(t)$. Введем характеристику, отражающую степень различия между матрицами $D_i(t)$ и $D_0(t)$. Для этого произведем сравнение взаимного расположения пары элементов в сравниваемых матрицах $D_i(t)$ и $D_0(t)$ у всех возможных пар, таких, что оба элемента пары принадлежат одному столбцу. Пусть $S_{ik}(t)$ - число пар, у которых взаимное расположение элементов в k -м столбце матриц $D_i(t)$ и $D_0(t)$ не совпадает. Общее число несовпадений определится как

$$W_i(t) = \sum_k S_{ik}(t).$$

Чем больше $W_i(t)$, тем больше различие между $D_i(t)$ и $D_0(t)$, и наоборот, чем меньше $W_i(t)$, тем различие между $D_i(t)$ и $D_0(t)$ меньше. Тяготение частицы в позиции $x_i(t)$ к позиции $x_0(t)$ выражается в стремлении уменьшения различия между $D_i(t)$ и $D_0(t)$, то есть уменьшения показателя $W_i(t)$. Будем считать, чем меньше различие между $D_i(t)$ и $D_0(t)$, тем ближе позиция $x_i(t)$ к позиции $x_0(t)$.

Модификация матрицы $D_i(t)$, то есть переход к матрице $D_i(t+1)$, производится путём выборочных групповых парных перестановок соседних элементов в столбцах матрицы $D_i(t)$.

На каждом шаге за два такта анализируются пары элементов в столбцах матрицы $D_i(t)$ и осуществляется ее модификация, приводящая к уменьшению показателя $W_i(t)$.

На каждом такте рассматриваются множество непересекающихся пар элементов $(d_{lk}, d_{l+1,k})$ матрицы $D_i(t)$, каждая из которых расположена в одном столбце $D_{ik}(t)$ и в двух соседних строках l и $l+1$. Будем в дальнейшем первый элемент пары называть верхним, а второй нижним. На первом такте в каждом столбце $D_{ik}(t)$ анализируется множество P_1 непересекающихся пар элементов, у которых l -нечетное число: $P_1 = \{(d_{1k}, d_{1+1,k}) \mid l=1, 3, 5, \dots, j=1, 2, 3, \dots\}$, $P_1 = \{(d_{1k}, d_{2k}), (d_{3k}, d_{4k}), (d_{5k}, d_{6k}), \dots\}$. На втором такте анализируется множество P_2 непересекающихся пар элементов, у которых l -

четное число: $P_2 = \{(d_{lk}, d_{l+1,k}) \mid l=2,4,6,\dots, j=1,2,3,\dots\}$, $P_2 = \{(d_{21}, d_{31}), (d_{41}, d_{51}), (d_{61}, d_{71}), \dots\}$. Пары элементов на каждом такте анализируются независимо друг от друга. По результатам анализа принимается решение о перестановке элементов каждой пары, вычисляются показатели $S_{ik}(t)$ и осуществляется модификация $D_i(t)$.

Если взаимное расположение рассматриваемой пары отличных от нуля элементов в столбце $D_{ik}(t)$ матрицы $D_i(t)$ не совпадает с взаимным расположением этих элементов в столбце $D_{0k}(t)$ матрицы $D_0(t)$, производится парная перестановка этих элементов в $D_i(t)$.

При совпадении взаимного расположения элементов рассматриваемой пары в матрицах $D_i(t)$ и $D_0(t)$, перестановка этих элементов не производится.

Если один из элементов рассматриваемой пары имеет нулевое значение, то рассчитывается число нулей n_i^0 , размещенных в столбце $D_{ik}(t)$ над вторым (ненулевым) элементом и число нулей n^0 , размещенных над таким же элементом в столбце $D_{0k}(t)$. Если перестановка такой пары элементов в $D_{ik}(t)$ приводит к уменьшению величины $|n_i^0 - n^0|$, то эти элементы переставляются. В противном случае перестановка этих элементов не производится. Таким способом осуществляется выравнивание числа нулей, расположенных выше и ниже рассматриваемого элемента в столбце $D_{ik}(t)$ по отношению к столбцу $D_{0k}(t)$.

Отметим, что изменение взаимного расположения пары соседних элементов в столбце $D_{ik}(t)$ не приводит к изменению относительного расположения каждого элемента пары с остальными элементами столбца, а также к изменению взаимного расположения остальных элементов друг относительно друга.

Локальная цель перемещения частицы i - достижение позиции с наилучшим значением целевой функции. Глобальная цель роя частиц - формирование решения ЗКТ в минимальном числе магистралей.

Рассмотрим пример работы процедуры перемещения. На рис.4 представлена матрица $D_0(t)$, матрица $D_i(t)$, матрица $D_i^l(t+1)$, полученная после первого такта, матрица $D_i(t+1)$, полученная после второго такта.

| | | | | | | | |
|--------------|-------|-------|-------|------------|-------|-------|-------|
| $D_0(t)$ | | | | $D_i(t)$ | | | |
| d_1 | d_2 | d_3 | d_4 | d_1 | d_2 | d_3 | d_4 |
| 1 | 12 | 0 | 6 | 1 | 0 | 5 | 6 |
| 10 | 0 | 5 | 14 | 3 | 0 | 0 | 0 |
| 2 | 11 | 0 | 7 | 2 | 11 | 9 | 7 |
| 3 | 0 | 9 | 0 | 10 | 4 | 13 | 14 |
| 8 | 4 | 13 | 0 | 8 | 12 | 0 | 0 |
| $D_i^l(t+1)$ | | | | $D_i(t+1)$ | | | |
| d_1 | d_2 | d_3 | d_4 | d_1 | d_2 | d_3 | d_4 |
| 1 | 0 | 0 | 6 | 1 | 0 | 0 | 6 |
| 3 | 0 | 5 | 0 | 10 | 11 | 5 | 14 |
| 10 | 11 | 9 | 14 | 3 | 0 | 9 | 0 |
| 2 | 4 | 13 | 7 | 2 | 12 | 0 | 7 |
| 8 | 12 | 0 | 0 | 8 | 4 | 13 | 0 |

Рис. 4. Работа процедуры перемещения

Рассмотрим характерные ситуации. На первом такте в первом столбце матрицы $D_i(t)$ рассматриваются пары $\langle 1,3 \rangle$ и $\langle 2,10 \rangle$. Расположение элементов в паре $\langle 2,10 \rangle$ не совпадает с относительным расположением этих же элементов в первом столбце матрицы $D_0(t)$, поэтому производится парная перестановка этих элементов. В третьем столбце матрицы $D_i(t)$ рассматриваются пары $\langle 5,0 \rangle$ и $\langle 9,13 \rangle$. Значение второго элемента пары $\langle 5,0 \rangle$ равно нулю. Рассчитывается число нулей $n_i^0 = 0$, размещенных в столбце над вторым (ненулевым) элементом пары $\langle 5,0 \rangle$ со значением 5 в $D_i(t)$ и число нулей $n^0 = 1$, размещенных в столбце над таким же элементом в $D_0(t)$. Перестановка пары элементов $\langle 5,0 \rangle$ в $D_i(t)$ приводит к уменьшению величины $|n_i^0 - n^0|$, поэтому элементы пары $\langle 5,0 \rangle$ переставляются. На втором такте в первом столбце матрицы $D_i^l(t+1)$ рассматриваются пары $\langle 3,10 \rangle$ и $\langle 2,8 \rangle$. Расположение элементов в паре $\langle 3,10 \rangle$ не совпадает с расположением этих же элементов в первом столбце матрицы $D_0(t)$, поэтому производится парная перестановка этих элементов. Аналогичные действия производятся во всех столбцах рассматриваемых матриц.

IV. РОЕВОЙ АЛГОРИТМ КАНАЛЬНОЙ ТРАССИРОВКИ

Для учета одновременного тяготения частицы i к позициям $x_i^*(t)$ и $x_i^s(t)$ формируется центр притяжения $x_i^c(t)$ этой частицы. Среди позиций $x_i^*(t)$ и $x_i^s(t)$ определяется худшая $-x_i^b(t)$ и лучшая $-x_i^g(t)$. Формирование $x_i^c(t)$ осуществляется путем применения модифицированной процедуры перемещения к худшей позиции $x_i^b(t)$ для перемещения из нее в направлении к лучшей $x_i^g(t)$. Отличие модифицированной процедуры перемещения состоит в том, что обязательные в процедуре перемещения перестановки в модифицированной процедуре перемещения осуществляются с заданной вероятностью ρ . После определения центра притяжения $x_i^c(t)$ частица i с помощью процедуры перемещения сдвигается в направлении $x_i^c(t)$ из позиции $x_i(t)$ в позицию $x_i(t+1)$.

Схема работы роевого алгоритма канальной трассировки включает следующие шаги:

1. В соответствии с постановкой задачи трассировки и исходными данными формируется множество горизонтальных участков $F = \{f_i \mid i=1,2,\dots,N\}$.

2. Множество горизонтальных фрагментов F разбивается на подмножества F_k , $F = \{F_k \mid k=1,2,\dots,V\}$. Формируется пространство решений, каждая ось которого соответствует подмножеству F_k .

3. Создается исходная «случайная» популяция частиц, $t=0$. Для каждой частицы случайным образом задается начальная позиция $x_i(0)$. С этой целью формируется матрица $D_i(0)$, число столбцов которой рав-

но числу подмножеств F_k , а число строк равно a_{max} . В каждый k -й столбец в случайной последовательности заносятся элементы множества F_k .

4. Шаги 5-11 итерационно повторяются заданное число раз.

5. Рассчитывается целевая функция $f_i(t)$ для каждой частицы.

6. Определяются лучшая позиция роя $x^*(t)$ на шаге t и значение целевой функции $f^*(t)$ в этой позиции.

7. Для каждой частицы определяются лучшая позиция $x_i^*(t)$, которую она посещала с начала первой итерации, и значение целевой функции $f_i^*(t)$ в этой позиции.

8. Для каждой частицы определяется центр притяжения $x_i^c(t)$ этой частицы.

9. Рассчитываются новые позиции частиц в пространстве решений. С помощью процедуры перемещения каждая частица перемещается из позиции $x_i(t)$ в позицию $x_i(t+1)$.

10. $t=t+1$.

11. Запоминается лучшее решение, найденное роем к моменту t .

V. ГИБРИДИЗАЦИЯ РОЕВОГО ИНТЕЛЛЕКТА С ГЕНЕТИЧЕСКИМ ПОИСКОМ

Отдельные попытки [2,11,12] применения эволюционного программирования к задаче канальной трассировки были достаточно успешными. Однако предложенные структуры ГА фактически являются “слепыми” поисковыми структурами с присущими им недостатками: генерация решений с нарушениями, что требует дополнительного контроля; генерация большого количества подобных решений; генерация большого количества “плохих” решений.

По аналогии с эволюционными стратегиями, рой можно трактовать как популяцию, а частицу как индивида (хромосому). Это дает возможность построения гибридной структуры поиска решения, основанную на сочетании генетического поиска с методами роевого интеллекта. Связующим звеном такого подхода является структура данных, описывающая в виде хромосомы решение задачи. Если в качестве частицы используется хромосома, то число параметров, определяющих положение частицы в пространстве решений должно быть равно числу генов в хромосоме. В работе предлагается подход к построению структур и принципов кодирования хромосом, обеспечивающих их гомологичность и возможность одновременного использования в генетическом алгоритме, и в алгоритме на основе роя частиц.

Для отображения i -ого решения задачи канальной трассировки (ЗКТ) используется рассмотренная выше матрица $D_i = \{D_{ik} | k=1, 2, \dots, V\}$. Представим i -ое решение в виде хромосомы. Хромосома H_i является упорядоченной совокупностью генов g_{ik} . Ген g_{ik} является од-

ним из вариантов вектора D_{ik} матрицы D_i , т.е. значением g_{ik} является некоторый вектор D_{ik} .

Гены g_{ik} и g_{jk} хромосом H_i и H_j гомологичны, они одинаковы по составу элементов, соответствуют одному и тому же подмножеству фрагментов F_k , но отличаются порядком расположения элементов. Таким образом, хромосоме H_i соответствует матрица $D_i = \{D_{ik} | k=1, 2, \dots, V\}$. Декодирование хромосомы и определение магистралей для фрагментов осуществляется с помощью рассмотренной выше процедуры.

Основными генетическими операторами являются кроссинговер, мутация, селекция и редукция.

Реализация кроссинговера осуществляется следующим образом. У предварительно выбранной родительской пары хромосом (на основе использования «принципа рулетки») последовательно просматриваются гомологичные пары генов и с вероятностью P_k осуществляется обмен генами.

Реализация оператора мутации осуществляется следующим образом. Последовательно просматриваются локусы хромосом, и с вероятностью P_M осуществляется мутация гена в рассматриваемом локусе. Операция мутации гена заключается во взаимно-однозначной перестановке двух случайным образом выбранных в гене элементов.

Селекция и редукция популяции после выполнения операций кроссинговера и мутации осуществляется с использованием “принципа рулетки”.

Предлагается композитная архитектура многоагентной системы бионического поиска для решения задачи трассировки на основе роевого интеллекта и генетической эволюции. Рассмотрены два подхода.

Первый и наиболее простой подход к гибридации заключается в следующем. Сначала поиск решения осуществляется генетическим алгоритмом. Затем на основе популяции, полученной на последней итерации генетического поиска, формируется популяция для роевого алгоритма. В формируемую популяцию включаются лучшие, но отличные друг от друга хромосомы. При необходимости полученная популяция доукомплектовывается новыми индивидами. После этого дальнейший поиск решения осуществляется роевым алгоритмом.

При втором подходе метод роя частиц используется в процессе генетического поиска и играет роль аналогичную генетическим операторам. В этом случае на каждой итерации генетического алгоритма синтез новых хромосом с одной стороны осуществляется с помощью кроссинговера и мутации, а с другой стороны с помощью операторов роевого метода.

Оценка временной сложности операторов роя частиц не превышает оценки временной сложности генетических операторов. В связи с этим общая оценка временной сложности при любом подходе к гибридации не превышает оценки временной сложности

генетического алгоритма и лежит в пределах $O(n^2)$ - $O(n^3)$.

VI. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Экспериментальные исследования проводились на IBM PC. Было установлено, что нахождение лучшего решения обеспечивается при значении $P_K=0,4$, $P_M=0,1$, $M=50$, $T=130$, где M – объем популяции, T – число итераций.

Эксперименты показали, что целесообразно число строк матриц $D_i(t)$ априори увеличивать на 1-2 по сравнению с величиной α_{max} .

Наибольшую эффективность показал второй подход к гибридизации.

Для анализа точности получаемых решений был синтезирован ряд примеров с априори известным оптимальным значением целевой функции. Исследования подвергались примеры, содержащие до 1000 цепей. При совместной работе алгоритмов в рамках второго подхода вероятность получения оптимального решения составила 0.9, а оценки локально оптимальных решений, отличались от глобального оптимума не более чем на одну магистраль.

Сравнительный анализ с другими алгоритмами [1,2,11,12] трассировки производился на стандартных тестовых примерах и схемах (бенчмарках): Ex1, Ex3a, Ex3c, Ex4b, Ex5, De. Представленный роевой алгоритм канальной трассировки в сочетании с генетическим алгоритмом находит решения для задач большой размерности, не уступающие по качеству, а иногда и превосходящие своих аналогов с меньшими временными затратами, кроме того, он оптимизирует длину соединений.

Исследования трудоемкости гибридного алгоритма показали, что при фиксированных значениях P_M , P_K , M , T она на одной итерации имеет линейную зависимость и пропорциональна $O(N)$, где M – объем популяции, T – число итераций, N – число связываемых контактов.

Общая оценка временной сложности при любом подходе к гибридизации лежит в пределах $O(n^2)$ - $O(n^3)$.

VII. ЗАКЛЮЧЕНИЕ

На основе сравнительного анализа существующих подходов и методов для решения задачи канальной трассировки использованы мультиагентные методы интеллектуальной оптимизации. Для усиления сходимости и способности выхода из локальных оптимумов поисковые процедуры канальной трассировки, базирующиеся на моделировании адаптивного поведения роя частиц, интегрированы с процедурами генетического поиска. Связующим звеном такого подхода

является структура данных, описывающая в виде хромосомы решение задачи.

Повышения эффективности роевого алгоритма можно добиться путем адаптивного управления параметрами процедуры перемещения, такими как число тактов, методы сравнения и т.д.

Предложенный подход полностью применим для “бессеточной” трассировки соединений разной ширины. В этом случае задача трассировки сводится не к распределению фрагментов по магистралям, а к упаковке фрагментов. Модернизированная процедура декодирования будет последовательно размещать горизонтальные фрагменты, “прижимая” их на допустимую величину к ранее размещенным.

Рассмотренный роевой алгоритм может быть использован как базовая процедура при интеграции с генетическим алгоритмом трассировки в коммутационном блоке, рассмотренным в работе [13].

ЛИТЕРАТУРА

- [1] Деньдобренко Б.П., Малика А.С. Автоматизация проектирования радиоэлектронной аппаратуры. М.: Высш. шк., 2002.
- [2] Mazumder P., Rudnick E. Genetic Algorithm For VLSI Design, Layout & Test Automation. India, Pearson Education, 2003.
- [3] МакКоннелл Дж. Основы современных алгоритмов. М.: Техносфера, 2004.
- [4] G. Di Caro, F. Ducatelle, L. M. Gambardella. AntHocNet: An adaptive nature-inspired algorithm for routing in mobile ad hoc networks // *European Transactions on Telecommunications*. 2005. V. 16(5). С. 443-455.
- [5] Курейчик В.М., Лебедев Б.К., Лебедев О.Б. Поисковая адаптация: теория и практика. М.: Физматлит, 2006.
- [6] 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.
- [7] Лебедев О.Б. Трассировка в канале методом муравьиной колонии // *Известия ЮФУ*. Изд-во ТТИ ЮФУ, 2009. №2. С. 46-52.
- [8] A. P. Engelbrecht. *Fundamentals of Computational Swarm Intelligence*. John Wiley & Sons, Chichester, UK, 2005.
- [9] M. Clerc. *Particle Swarm Optimization*. ISTE, London, UK, 2006.
- [10] R. Poli. Analysis of the publications on the applications of particle swarm optimisation // *Journal of Artificial Evolution and Applications*, Article ID 685175, 10 pages. 2008.
- [11] Taniguch N., Liu X., Sakamoto A. & Shimomoto T. An Approach to channel routing using genetic algorithms // *Bulletin of Faculty of Engineering, Tokyshima University*, 1993. № 38. P. 99-112.
- [12] Rahmani, A.T. and Ono N. A Genetic Algorithm for Channel Routing Problem // *Proc. 5th Intl. Conf. on GAs*. 1993. P. 494-498.
- [13] Курейчик В.М., Лебедев Б.К. Генетический алгоритм трассировки в коммутационном блоке // *Известия вузов. Электроника*. 2002, №2. С. 55-66.