

<1; 6; 3>	<4; 8; 2>	<5; 9; 7>
-----------	-----------	-----------

При координатном представлении элементы, размещаемые в  $n$ -мерном пространстве не повторяются. Предлагается использовать векторную хромосому, в которой локус соответствует номеру элемента, а аллель – кортежу вида  $\langle x_1; x_2; \dots; x_n \rangle$  – определяющему “координаты” элемента. Тогда получим:

<1; 3>	<3; 2>	<3; 3>	<1; 2>	<1; 1>	<2; 3>	<3; 1>	<2; 2>	<2; 1>
--------	--------	--------	--------	--------	--------	--------	--------	--------

Отметим, что кроме рассмотренных существуют и иные методы кодирования информации.

**Заключение.** Математически доказано, что не существует идеальной структуры представления решений в генетических алгоритмах. Поэтому для получения эффективного решения оптимизационных задач необходимо сочетать все способы, которые позволят кодировать и декодировать любое возможное решение и производить его оценку. Способ представления различных типов данных в виде кода хромосомы определяет содержание отдельных операторов и генетического алгоритма в целом. Кроме того, правильный выбор представления, способов кодирования и декодирования обуславливает не только пространственную и временную сложность алгоритма, но и его эффективность.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Гринченко С.Н.* Метод «проб и ошибок» и поисковая оптимизация: анализ, классификация, трактовка понятия «Естественный отбор» <http://zhurnal.ape.relarn.ru/articles/2003/104.pdf>.
2. *Рутковская Д.* Нейронные сети, генетические алгоритмы и нечеткие системы. – М.: Горячая линия – Телеком, 2007. – 452 с.
3. *Бакало М.А.* Варианты представления различных типов данных, их кодирование и декодирование в виде хромосом // Известия ТРТУ. Тематический выпуск «Интеллектуальные САПР». – Таганрог: Изд-во ТРТУ, 2007. – С. 70-75.
4. *Гладков Л.А., Курейчик В.М., Курейчик В.В.* Генетические алгоритмы. – М.: Физматлит, 2006.

УДК 658.512

**Б.К. Лебедев, В.Б. Лебедев, О.Б. Лебедев**

#### ЭВОЛЮЦИОННЫЕ МЕХАНИЗМЫ ТРАССИРОВКИ В КАНАЛЕ\*

Работа посвящена адаптивным поисковым алгоритмам детальной трассировки с использованием интеллектуальных процедур.

Рассматривается метод канальной трассировки (КТ), работающий с использованием процедур снижения пространственной сложности задачи трассировки.

При КТ каждая цепь, представляется в виде набора горизонтальных и вертикальных фрагментов. На взаимное расположение горизонтальных фрагментов накладываются ограничения, задаваемые с помощью графа вертикальных ограничений (ГВО)  $G_V=(X,U)$  и графа горизонтальных ограничений (ГГО)  $G_H=(X,W)$ . Задача трассировки в канале рассматривается как задача распределения фиксированно-

\* Работа выполнена при поддержке РФФИ (грант № 07-01-00174).

го множества горизонтальных участков  $F = \{f_i | i=1,2,\dots,N\}$  в множестве магистралей  $M = \{m_j | j=1,2,\dots,L\}$ .

Вначале, на основе совместного учета ГГО и ГВО, для каждого участка  $f_i$  определяются граничные, сверху и снизу канала, номера  $\delta_i^t$  и  $\delta_i^b$  магистралей, в которые он может быть помещен. На основе совместного анализа  $\delta_i^t$  и  $\delta_i^b$ , ГГО и ГВО, по впервые разработанной методике, рассчитывается оценка минимальной ширины канала  $\xi_m$  [1]. Для каждого  $f_i$  формируется множество  $M_i = \{m_j\}$  – набор разрешенных магистралей (НРМ), в которых  $f_i$  может быть размещен при ширине канала, равной  $\xi_m$ .

На основе исследования и анализа ряда закономерностей, учитывающих специфику задачи трассировки в канале, в работе [1] сформулированы 6 правил «отсечки» (ПО), заключающиеся в сужении НРМ без потери оптимальности, и правило проверки (ПП) реализуемости трассировки соединений в канале с заданной шириной.

На базе ПО и ПП разработана адаптивная процедура получения решения в усеченном пространстве решений.

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

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

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  $F_{k+1}$ .

Рассмотрим процедуру разбиения  $F$  на  $F_k$ . Обозначим через  $l_i$  и  $r_i$  левый и правый концы горизонтального участка  $f_i$ . Каждый  $l_i$  и  $r_i$  имеют свою абсциссу. Сформируем упорядоченный массив  $A$ , состоящий из концов отрезков  $f_i \in F$ . Концы в массиве  $A$  располагаются по возрастанию их абсцисс. Если  $l_i$  и  $r_j$  имеют одну и ту же абсциссу, то  $r_j$  в  $A$  располагается за  $l_i$ .

**Алгоритм А1.**

1.  $k=1$  ( $k$  – номер подмножества).

2. Выбирается очередной, начиная с первого, элемент массива  $A$ . Если все элементы рассмотрены, то переход к п. 5.

3. Если выбранный элемент является левым концом  $l_i$ , то  $f_i$  включается в  $F_k$  и переход к п. 2, иначе – к п. 4.

4. Если выбранный элемент является правым концом  $r_i$  и  $f_i$  входит в  $F_k$ , то  $k=k+1$ . Переход к п. 2.

5. Конец.

Алгоритм разбиения  $F$  на  $F_k$  имеет линейную трудоемкость.

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

Для отображения решения задачи канальной трассировки (ЗКТ) формируется матрица  $D$ , число столбцов которой равно числу подмножеств  $F_k$ , а число строк равно  $\xi_m$  (рис. 2). В каждый  $k$ -й столбец заносятся элементы множества  $F_k$ . Построенная таким образом матрица  $D$  используется для символического представления решения ЗКТ. Таким образом, если считать, что элементы матрицы  $D$ , размещенные в одной строке, размещаются в одной магистрали, то матрица  $D$  будет решением ЗКТ, если между элементами будут отсутствовать вертикальные и горизонтальные конфликты, т.е. отсутствуют нарушения ограничений, задаваемых ГВО и ГГО.

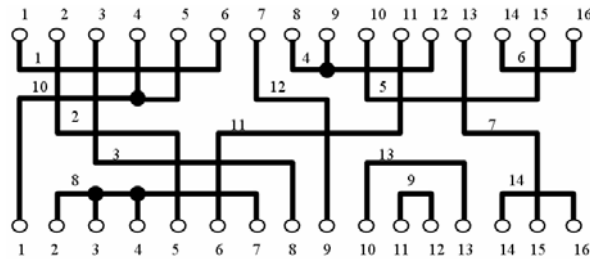


Рис. 1. Пример трассировки в канале

Формирование решения ЗКТ осуществляется в процессе эволюционной модификации матрицы  $D$ . Вначале в каждый  $k$ -й столбец элементы множества  $F_k$  заносятся случайным образом, но в соответствии с НРМ, построенным для каждого

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

Рис. 2. Символьное представление решения задачи канальной трассировки

элемента. На первом этапе эволюционная модификация матрицы  $D$  производится путём выборочных групповых парных перестановок соседних элементов в столбцах. Адаптивный процесс состоит из повторяющихся шагов, каждый из которых представляет собой переход от одного решения (состояния матрицы  $D$ ) к другому – лучшему [2], что обеспечивает направленное последовательное перемещение элементов в столбцах матрицы  $D$ . Глобальная цель адаптивного процесса – ликвидация вертикальных и горизонтальных конфликтов между горизонтальными фрагментами.

На каждом шаге анализируются пары элементов в столбцах матрицы  $D$ . Анализ осуществляется за четыре такта.

На каждом такте рассматриваются множество непересекающихся пар элементов  $(d_{ij}, d_{i+1,j})$  матрицы  $D$ , каждая из которых расположена в одном  $j$ -м столбце и в двух соседних строках  $i$  и  $i+1$ . Отметим, что первым элементом пары расположен над вторым элементом пары. Будем в дальнейшем первый элемент пары называть верхним, а второй нижним. На первом такте анализируется множество  $P_1$  непересекающихся пар элементов, у которых  $j$ -нечетное число,  $i$ -нечетное число:  $P_1 = \{(d_{ij}, d_{i+1,j}) | i=1,3,5,\dots, j=1,3,5,\dots\}$ ,  $P_1 = \{(d_{11}, d_{21}), (d_{31}, d_{41}), (d_{51}, d_{61}), \dots, (d_{13}, d_{23}), (d_{33}, d_{43}), (d_{53}, d_{63}), \dots, (d_{15}, d_{25}), (d_{35}, d_{45}), (d_{55}, d_{65}), \dots\}$ . На втором такте анализируется множество  $P_2$  непересекающихся пар элементов, у которых  $j$ - четное число,  $i$ -нечетное число:  $P_2 = \{(d_{ij}, d_{i+1,j}) | i=1,3,5,\dots, j=2,4,6,\dots\}$ ,  $P_2 = \{(d_{12}, d_{22}), (d_{32}, d_{42}), (d_{52}, d_{62}), \dots, (d_{14}, d_{24}), (d_{34}, d_{44}), (d_{54}, d_{64}), \dots, (d_{16}, d_{26}), (d_{36}, d_{46}), (d_{56}, d_{66}), \dots\}$ . На третьем такте анализируется множество  $P_3$  непересекающихся пар элементов, у которых  $j$ -нечетное число,  $i$ -четное число:  $P_3 = \{(d_{ij}, d_{i+1,j}) | i=2,4,6,\dots, j=1,3,5,\dots\}$ ,  $P_3 = \{(d_{21}, d_{31}), (d_{41}, d_{51}), (d_{61}, d_{71}), \dots, (d_{23}, d_{33}), (d_{43}, d_{53}), (d_{63}, d_{73}), \dots, (d_{25}, d_{35}), (d_{45}, d_{55}), (d_{65}, d_{75}), \dots\}$ . На четвертом такте анализируется множество непересекающихся пар элементов, у которых  $j$ -четное число,  $i$ -четное число:  $P_4 = \{(d_{ij}, d_{i+1,j}) | i=2,4,6,\dots, j=2,4,6,\dots\}$ ,  $P_4 = \{(d_{22}, d_{32}), (d_{42}, d_{52}), (d_{62}, d_{72}), \dots, (d_{24}, d_{34}), (d_{44}, d_{54}), (d_{64}, d_{74}), \dots, (d_{26}, d_{36}), (d_{46}, d_{56}), (d_{66}, d_{76}), \dots\}$ . На рис. 3 показаны схемы, в соответствии с которыми образуются подмножества пар. Пара элементов в столбце помечена одной цифрой.

Пары элементов на каждом такте анализируются независимо друг от друга. По результатам анализа принимается решение о перестановке элементов каждой

пары. Локальная цель перемещения элемента  $d_{ij}$  в столбце  $j$  матрицы  $D$  – достижение им позиции  $i$ , в которой у элемента  $d_{ij}$  отсутствуют вертикальные и горизонтальные конфликты с остальными элементами матрицы  $D$ . Глобальная цель – формирование решения ЗКТ в минимальном числе магистралей.

1 такт					2 такт					3 такт					4 такт				
1	-	1	-	1	-	1	-	1	-	-	-	-	-	-	-	-	-	-	-
1	-	1	-	1	-	1	-	1	-	1	-	1	-	1	-	1	-	1	-
2	-	2	-	2	-	2	-	2	-	1	-	1	-	1	-	1	-	1	-
2	-	2	-	2	-	2	-	2	-	2	-	2	-	2	-	2	-	2	-
3	-	3	-	3	-	3	-	3	-	2	-	2	-	2	-	2	-	2	-
3	-	3	-	3	-	3	-	3	-	3	-	3	-	3	-	3	-	3	-
-	-	-	-	-	-	-	-	-	-	3	-	3	-	3	-	3	-	3	-

Рис. 3. Схемы, в соответствии с которыми образуются подмножества пар

В процессе анализа для каждого элемента матрицы  $D$  (с ненулевым значением), соответствующего некоторому участку  $f_i$ , определяется его состояние, т.е. наличие или отсутствие горизонтальных и вертикальных конфликтов с остальными элементами матрицы  $D$  в соответствии с их расположением в матрице. Если элемент  $d_{ki}$  находится в горизонтальном конфликте с каким-либо элементом  $d_{kj}$ , расположенным в той же строке, что и  $d_{ki}$ , то оба они помечаются меткой  $\downarrow$ , т.е.  $d_{ki}$  и  $d_{kj}$  необходимо разнести по разным строкам. Если в соответствии с расположением элементов в матрице  $D$  элемент  $d_{ki}$  находится в вертикальном конфликте с каким-либо элементом  $d_{lj}$ , причем для его ликвидации необходимо  $d_{ki}$  поместить выше  $d_{lj}$ , то  $d_{ki}$  помечается меткой  $\uparrow$ , а элемент  $d_{lj}$  меткой  $\downarrow$ . Если элемент  $d_{ki}$  не конфликтует с другими элементами матрицы  $D$ , т.е. для него отсутствуют нарушения ограничений, задаваемых ГВО и ГГО, то элемент  $d_{ki}$  помечается меткой 0. Метки  $\downarrow$  и  $\uparrow$  суммируются. Если число меток  $\downarrow$  больше числа меток  $\uparrow$ , то окончательно элемент помечается одной меткой  $\downarrow$ . И наоборот. Если же число меток  $\downarrow$  равно числу меток  $\uparrow$ , то окончательно элемент случайным образом помечается либо меткой 0, либо меткой  $\uparrow$ . Отметим, что возможна ситуация, когда отдельные элементы могут быть помечены парой меток, одна из которых –  $\downarrow$ , а другая – либо  $\uparrow$ , либо  $\downarrow$ . В этом случае остается только одна из меток – либо  $\uparrow$ , либо  $\downarrow$  соответственно.

Суть анализа каждой пары заключается в анализе состояний элементов рассматриваемой пары, на основании которых принимается решение об их перестановке.

В работе рассматриваются два подхода. При первом подходе задача решается с помощью простой эволюционной процедуры. При втором подходе задача решается методами поисковой адаптации на основе коллективного поведения автоматов адаптации.

При первом подходе в процессе эволюционной модификации матрицы  $D$  в соответствии с вышеприведенной методикой перестановка пары элементов ( $d_{ij}$ ,  $d_{i+1,j}$ ) матрицы  $D$  осуществляется при следующих комбинациях их состояний:  $\downarrow\uparrow$ ;  $\downarrow\downarrow$ ;  $\uparrow\uparrow$ ;  $\uparrow\downarrow$ ;  $\downarrow 0$ ;  $0\uparrow$ ;  $0\downarrow$ ;  $\uparrow 0$ . Для остальных комбинаций –  $\uparrow\downarrow$ ,  $\downarrow\downarrow$ ,  $\uparrow\uparrow$ ,  $\uparrow\downarrow$ ,  $\uparrow\uparrow$ ,  $\uparrow 0$ ,  $0\downarrow$ ,  $00$  перестановки либо не выполняются, либо для преодоления локального барьера выполняются с некоторой вероятностью. При этом комбинации упорядочиваются по степени ухудшения состояния и соответствующие этим комбинациям значения вероятностей перестановок уменьшаются в этом же порядке.

При втором подходе решение оптимизационной задачи трассировки методами поисковой адаптации осуществляется на основе сочетания самообучения, само-

организации и генетического поиска, что позволяет преодолеть барьер локального оптимума. Концептуальная схема решения рассматриваемой проблемы такова. Задача представляется в виде многоагентной системы (МАС), состоящей из простейших реактивных агентов, которые способны достигать поставленных целей, согласовывать индивидуальные цели с общими целями всего коллектива, осуществлять распределение ресурсов, реализовывать процессы саморегулирования.

Для реализации механизма адаптации каждому объекту (элементу  $d_{ij}$  матрицы  $D$ ) сопоставляется два автомата адаптации (АА):  $A^1_{ij}$  и  $A^2_{ij}$ , моделирующих поведение объекта адаптации в среде. Автомат  $A^1_{ij}$  управляет перестановками элемента  $d_{ij}$  матрицы  $D$  в том случае, когда он является верхним элементом пары, а  $A^2_{ij}$  управляет перестановками элемента  $d_{ij}$  матрицы  $D$  в том случае, когда он является нижним элементом пары. Число групп состояний АА равно числу возможных альтернатив при перестановке элемента  $d_{ij}$ .  $A^1_{ij}$ . Таких альтернатив 3: переставлять ( $y$ ), не переставлять ( $n$ ), нейтральное положение ( $e$ ).

Не нарушая общности, рассмотрим принципы функционирования одного АА. АА имеет 3 группы состояний  $\{C^1_{ij}, C^2_{ij}, C^3_{ij}\}$ . Если АА находится в группе  $C^1_{ij}$ , то соответствующий ему элемент  $d_{ij}$  стремится к перестановке со вторым элементом рассматриваемой пары (альтернатива  $V_1$ ).  $C^3_{ij}$  соответствует запрету перестановки для  $d_{ij}$  (альтернатива  $V_3$ ).  $C^2_{ij}$  – нейтральное состояние (альтернатива  $V_2$ ). Граф-схема переходов АА представлена на рис. 4.

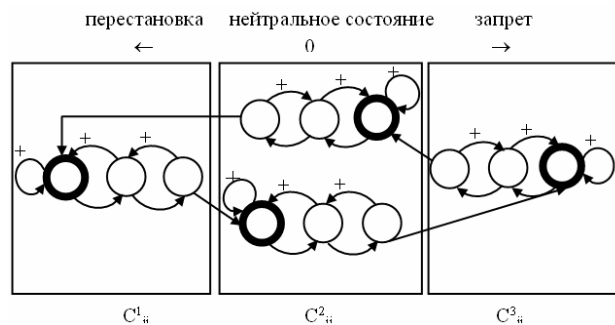


Рис. 4. Граф-схема переходов АА I типа

Число состояний в каждой группе состояний задается параметром  $Q^k_{ij}$ , называемым глубиной памяти или “степенью доверия”. На рис. 4 для каждой группы этот параметр равен 3. Первоначально АА находится в одном из начальных состояний (на рисунке (см. рис.1) эти состояния выделены жирным шрифтом). Особенностью представленного АА является то, что если в группу  $C^2_{ij}$  (нейтральное положение) осуществлен переход из  $C^1_{ij}$ , то выход из  $C^2_{ij}$  возможен только в  $C^3_{ij}$ , и наоборот. При входе в  $C^2_{ij}$  из  $C^3_{ij}$  выход из  $C^2_{ij}$  возможен только в  $C^1_{ij}$ . Это отражает эвристическое соображение, заключающееся в том, что изменение отношения к перестановке осуществляется через нейтральное положение. С другой стороны, это может быть тормозом и замедлять процесс изменения отношения к перестановке в процессе адаптации.

На каждой итерации работа адаптивного алгоритма трассировки осуществляется за четыре такта. На каждом такте рассматривается одно из подмножеств пар  $P_1, P_2, P_3, P_4$ .

На каждом такте выполняется 4 шага.

На 1-м шаге определяется состояние каждого элемента  $d_{ij}$  в соответствии с его расположением в матрице  $D$ . В результате элементы помечаются метками, как указано выше.

На 2-м шаге сравниваются между собой состояния объектов в среде и соответствующих им АА. Последовательно просматриваются все пары рассматриваемого подмножества. Для каждой пары сначала рассматривается верхний элемент, а потом нижний.

Если рассматривается верхний элемент, то сигнал поощрения вырабатывается при следующих комбинациях состояний элемента и АА:  $\downarrow y, \uparrow n, \uparrow y, 0e$ , а сигнал наказания вырабатывается при следующих комбинациях состояний элемента и АА:  $\uparrow y, \uparrow n, \downarrow n$ . Сигнал подается на АА  $A^1_{ij}$ .

Если рассматривается нижний элемент, то сигнал поощрения вырабатывается при следующих комбинациях состояний элемента и АА:  $\uparrow y, \downarrow n, \uparrow y, 0e$ , а сигнал наказания вырабатывается при следующих комбинациях состояний элемента и АА:  $\uparrow n, \uparrow n, \downarrow y$ . Сигнал подается на АА  $A^2_{ij}$ .

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

На 4-м шаге в соответствии с комбинацией состояний АА  $A^1_{ij}$  и  $A^2_{i+1,j}$  осуществляются или нет парные перестановки в каждой паре  $(d_{ij}, d_{i+1,j})$ , принадлежащей рассматриваемому множеству пар. Процесс основан на парном взаимодействии автоматов, при котором они обмениваются информацией о своем текущем состоянии. Для этого предварительно среди всех комбинаций состояний пары АА, соответствующих паре элементов из рассматриваемого множества пар отбираются такие, которые разрешают парную перестановку. Очевидными среди разрешающих являются следующие комбинации состояний пары АА  $A^1_{ij}$  и  $A^2_{i+1,j}$ :  $(yy), (ye), (ey)$ .

Эксперименты показали, что в ряде случаев в это число можно включить и ситуации типа  $(ee)$ , т.к. это позволяло выйти из локального оптимума.

Было определено, что быстрее всего алгоритм сходится при значении параметра  $Q^k_{ij}$  равным  $1 \div 3$ .

Работа адаптивной процедуры завершается в двух случаях. В первом случае работа процедуры завершается после того, как будет достигнуто такое состояние матрицы  $D$ , при котором все ее элементы помечены только символом 0, т.е. соответствующие им фрагменты размещены по магистралям без нарушений ограничений, задаваемых ГВО и ГГО. Во втором случае работа процедуры завершается после выполнения заданного числа шагов (итераций). Если после выполнения заданного числа шагов отдельные элементы матрицы  $D$  размещены с нарушениями горизонтальных и вертикальных ограничений используется стандартная процедура, с помощью которой формируется окончательное решение ЗКТ в виде матрицы  $D^k$ . Матрица  $D^k$  имеет тоже число столбцов, что и матрица  $D$ , а число строк определяется в процессе работы стандартной процедуры.

Строки матрицы  $D^k$  заполняются элементами матрицы  $D$  последовательно, начиная с первой. Заполнение строки матрицы  $D^k$  осуществляется следующим образом. Последовательно по строкам начиная с первой, а пределах строки слева направо просматриваются элементы матрицы  $D$  и определяется возможность их размещения в формируемой строке матрицы  $D^k$  в соответствии с  $G_V$  и  $G_H$ . Если возможно, то элемент помещается в формируемую строку матрицы  $D^k$  и удаляется из  $D$ . По окончании просмотра матрицы  $D$  осуществляется переход к заполнению следующей строки матрицы  $D^k$  и т.д. пока не обнулится матрица  $D$ . Временная сложность адаптивной процедуры на одном шаге –  $O(n)$ . Сравнение с известными

алгоритмами показало, что при меньшем времени работы новый алгоритм дает более качественные решения

Тестирование разработанных алгоритмов КТ и их сравнение с известными производилось на бенчмарках Ex1, Ex3b, Ex3c, Ex4b, Ex5, трудный пример Дойча, По сравнению с существующими алгоритмами достигнуто улучшение результатов на 2-5%.

#### БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Лебедев Б.К.* Интеллектуальные процедуры синтеза топологии СБИС. – Таганрог: Изд-во ТРТУ, 2003.
2. *Лебедев Б.К.* Адаптация в САПР. – Таганрог: Изд-во ТРТУ, 1999.
3. *Лебедев Б.К.* Методы поисковой адаптации в задачах автоматизированного проектирования СБИС. – Таганрог: Изд-во ТРТУ, 2000.

УДК 681.31

**Ю.О. Чернышев, П.Г. Белявский, А.Ю. Полуян**

#### **ЭВОЛЮЦИОННЫЙ ПОДХОД К РЕШЕНИЮ ЗАДАЧИ О НАЗНАЧЕНИИ ЧЕРЕЗ ОПРЕДЕЛЕНИЕ КРАТЧАЙШЕГО ПУТИ**

**Введение.** Задача о назначении является частным случаем транспортной задачи [1]. Поэтому ее можно решать, например, методом потенциалов, алгоритмом Литгла, а также венгерским методом [2, 3]. Указанные алгоритмы определения оптимального назначения не дают заранее временной оценки решения, а некоторые из них не всегда приводят к оптимальным решениям. При практическом увеличении размеров матрицы ограничений применение этих методов также становится нецелесообразным. С целью сокращения вычислительной сложности решения задачи о назначении и получении глобального оптимального решения предлагается свести задачу о назначении к определению кратчайшего пути в графе.

**Математическая модель задачи о назначении.** Для задачи о назначении решение ее представляет собой перестановку  $(p_1, p_2, \dots, p_n)$  чисел  $i=1, \dots, n$ , полученную в результате назначений вида  $i \rightarrow p_i$ . Целью является нахождение на конечном множестве указанных перестановок минимума затрат  $\sum_i c_{ip}, i=1, \dots, n$ .

Любую перестановку можно интерпретировать как точку в  $n^2$  – мерном евклидовом пространстве. При этом ее удобно представить в виде  $n \times n$  – матрицы  $X = //x_{ij}//$ , где  $x_{ij} = 1$  в случае назначения  $i$ -го кандидата на  $j$ -ю должность и  $x_{ij} = 0$  в противном случае. Чтобы не помешать применению программирования, заменим последнее условие другим:  $x_{ij} \geq 0$ . Минимум суммарных затрат запишется:

$$\min F = \sum_i \sum_j C_{ij} x_{ij}, \quad i, j = \overline{1, n}.$$

Тогда целочисленные условия недробления должностей и недопустимости совместительства выразятся: