

Курейчик Владимир Викторович – Технологический институт федерального государственного автономного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге; e-mail: vkur@tsure.ru; 347928, г. Таганрог, пер. Некрасовский, 44, ГСП 17А; тел.: 88634371651; кафедра систем автоматизированного проектирования; зав. кафедрой; д.т.н.; профессор.

Запорожец Дмитрий Юрьевич – e-mail: elpilasgsm@gmail.com; тел.: 88634371651; кафедра систем автоматизированного проектирования; аспирант.

Заруба Дарья Викторовна – e-mail: daria_zaruba@gmail.com; кафедра систем автоматизированного проектирования; магистрант.

Kureichik Vladimir Victorovich – Taganrog Institute of Technology – Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education “Southern Federal University”; e-mail: vkur@tsure.ru; GSP 17A, 44, Nekrasovskiy, Taganrog, 347928, Russia; phone: +78634371651; the department of computer aided design; head of department; dr. eng. sc.; professor.

Zaporoghetz Dmitri Urievich – e-mail: elpilasgsm@gmail.com; phone: +78634371651; the department of computer aided design; postgraduate student.

Zaruba Daria Victorovna – e-mail: daria_zaruba@gmail.com; the department of computer aided design; master.

УДК 681.3.001.63

Б.К. Лебедев, Е.И. Воронин

ГЕНЕТИЧЕСКИЙ АЛГОРИТМ РАСПРЕДЕЛЕНИЯ СОЕДИНЕНИЙ ПО СЛОЯМ ПРИ МНОГОСЛОЙНОЙ ГЛОБАЛЬНОЙ ТРАССИРОВКЕ СБИС*

Предлагается модифицированный генетический алгоритм распределения соединений по слоям при глобальной трассировке многослойных СБИС. Используется двухэтапный подход. Сначала многослойный граф «сжимается» в однослойный и выполняется однослойная глобальная трассировка. Затем решается задача распределения по слоям, во время которой каждая цепь распределяется на многослойном графе. Критерий оптимизации – общее число межслойных переходов. Ограничения – относительная перегруженность ребер графа. Использование генетического алгоритма позволяет учитывать несколько критериев при поиске оптимального решения. По сравнению с существующими алгоритмами достигнуто улучшение результатов.

СБИС; генетический алгоритм; глобальная трассировка; многослойная трассировка; распределение по слоям; оптимизация.

B.K. Lebedev, E.I. Voronin

GENETIC ALGORITHM FOR LAYER ASSIGNMENT BY MULTILAYER VLSI GLOBAL ROUTING

The genetic algorithm for layer assignment by multilayer VLSI global routing is proposed in this work. The graph models are used to search process simulation. At first multilayered graph "is compressed" in single-layered and single-layered global routing is carried out. Then are solved the distribution problem on layers during which time each net is distributed on the grid graph. Criterion of optimization – via cost. Restrictions – congestion of grid graph edges. The using of genetic algorithm lets perform search process considering some criterions simultaneously. In comparison with existing algorithms improvement of results is reached.

VLSI; multilayer global routing; layer assignment; via minimization; genetic algorithm; optimization.

* Работа выполнена при частичной поддержке РФФИ (проект № 10-07-00055).

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. Трассировка является сложной комбинаторной задачей. В связи с постоянным уменьшением геометрических размеров элементов и, соответственно, увеличением их числа на кристалле, задача все больше усложняется и требует новых методов решения, учитывающих появляющиеся ограничения [1–5].

Большинство известных методов использует двухэтапный подход для решения задачи трассировки, включающий глобальную и детальную трассировку. Основным критерием оптимизации является трассируемость – процент проложенных соединений (в идеале должен быть равен 100 %) и перегрузка [5–8].

Из-за постоянного уменьшения размера элемента СБИС и увеличения числа элементов на кристалле усложняется задача трассировки. Для прокладки всех необходимых соединений недостаточно иметь два слоя металлизации. С увеличением числа слоев металлизации увеличивается и число межслойных переходов (via). В такой ситуации стадия глобальной трассировки становится определяющей при проектировании СБИС. К традиционным критериям трассируемости и общей длины соединений добавляется новый критерий – число межслойных переходов. Существует два основных подхода к решению задачи многослойной глобальной трассировки. Первый – трассировать все цепи непосредственно в многослойном пространстве решений. Зачастую, при большом размере задачи такой подход требует слишком больших временных затрат. Второй подход состоит из двух отдельных задач: глобальной однослойной трассировки и распределения соединений по слоям. Сначала многослойный граф «сжимается» в однослойный и выполняется однослойная глобальная трассировка. Затем решается задача распределения по слоям. При таком подходе задача распределения по слоям является определяющей для качества глобальной трассировки в целом [8].

В данной работе используется модификация генетического алгоритма для решения задачи распределения соединений по слоям при многослойной глобальной трассировке.

Постановка задачи распределения соединений по слоям. Дадим формальное определение задаче распределения по слоям. Пусть $G^k=(V^k, E^k)$ – k -слойный граф, где V^k – множество вершин и E^k – множество ребер. На рис. 1 показан пример графа G^3 для $k = 3$. Граф G используется для моделирования процесса глобальной трассировки. Каждая вершина соответствует области на одном из слоев металлизации. Множество ребер E^k состоит из двух непересекающихся подмножеств E^{kv} и E^{ke} . Множество E^{kv} включает все ребра, которые соединяют вершины разных слоев, это, так называемые, *ребра межслойных переходов* (показаны справа на рис. 1). Множество E^{ke} включает в себя ребра, которые соединяют вершины в одном слое. Такие ребра определяют пропускную способность $c(e^{ke})$ границы между двумя соседними областями трассировки и называются *граничными ребрами*. Пропускная способность или емкость $c(e^{ke})$ определяет максимальное число трасс, которые могут быть проложены через данную границу без нарушения ограничений.

Пусть $G^l(V^l, E^l)$ – однослойный граф глобальной трассировки, полученный путем «сжатия» графа $G^k=(V^k, E^k)$. Граф G^l получается из графа G^k следующим образом. Все слои графа G^k проецируются на первый (нижний) слой. При этом емкости лежащих друг над другом граничных ребер складываются, определяя, таким образом, емкость граничного ребра на графе G^l . Стоит отметить, что при переходе к однослойному графу, все межслойные ребра исчезают. На рис. 2 показан однослойный граф G^l , полученный «сжатием» графа G^3 . Вершины v_{2m} и v_{3m} проециру-

ются в v_{lm} , а ребра e_{3q} и e_{2q} проецируются на ребро e_{1q} . Предположим, что до процесса «сжатия» ребра имели следующую пропускную способность: $c(e_{1q}) = 3$, $c(e_{2q}) = 2$, $c(e_{3q}) = 5$. Тогда, суммарная пропускная способность ребра e_{1q} на графе G^1 будет $c(e_{1q}) = 10$.

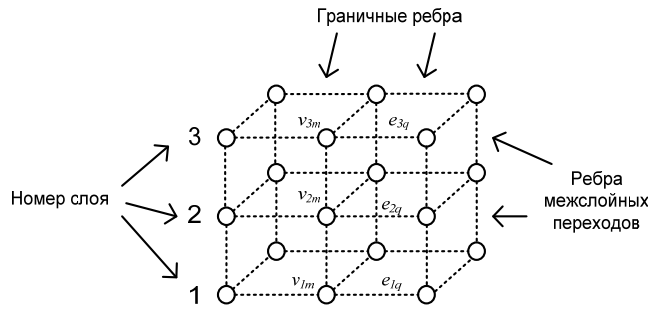


Рис. 1. Трехслойный граф G^3 , используемый для моделирования процесса глобальной трассировки

Пусть имеется некоторое решение задачи глобальной трассировки на графе G^1 , представляющее собой множество протрассированных цепей N^1 . Каждая цепь представляет собой дерево Штейнера S_i , построенное на графе G^1 . Каждое дерево S_i состоит из набора простых соединений $T_i = \{t_{ij} \mid j = 1, 2, \dots, n_j\}$. Необходимо распределить все соединения для каждого T_i на многослойном графе G^k без нарушения ограничений и с учетом заданных критериев оптимизации.

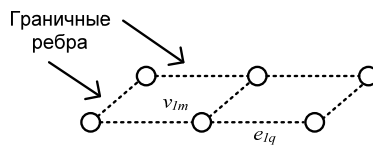


Рис. 2. Однослойный граф G^1 , полученный после «сжатия» трехслойного графа G^3

Определим критерии оптимизации для задачи распределения соединений по слоям. Пусть $d(e^{ke}_q)$ – загруженность ребра e^{ke}_q , т.е. число использованных при трассировке ресурсов. Тогда относительную перегруженность ребра e^{ke}_q определим как $\varphi(e^{ke}_q) = d(e^{ke}_q)/c(e^{ke}_q)$. На величину $\varphi(e^{ke}_q)$ накладывается следующее ограничение:

$$\varphi(e^{ke}_q) \leq 1. \tag{1}$$

Просуммировав относительную перегруженность $\varphi(e^{ke}_q)$ для всех ребер множества E^{ke} , получим первый критерий оптимизации для задачи распределения соединений по слоям:

$$\Omega = \sum_{q=1}^Q \varphi(e^{ke}_q), \tag{2}$$

где Q – число ребер множества E^{ke} . Цель оптимизации – минимизация критерия Ω .

Второй критерий оптимизации определим как общее число межслойных переходов Ψ . Пусть $\mu(e^{kv}_l)$ – число межслойных переходов, назначенных для ребра e^{kv}_l , тогда Ψ рассчитывается по формуле (3):

$$\Psi = \sum_{l=1}^L \mu(e^{kv}_l), \tag{3}$$

где L – число ребер множества E^{kv} . Цель оптимизации – минимизация критерия Ψ . Введем также дополнительное ограничение h на параметр μ :

$$\mu(e^{kv}) \leq 4. \quad (4)$$

Данное ограничение определяет максимальное количество межслойных переходов в пределах одной области трассировки.

Генетический алгоритм распределения соединений по слоям. В работе предлагается генетический алгоритм распределения соединений по слоям, с учетом критериев оптимизации, упомянутых выше.

Целевая функция. Как упоминалось выше, первым этапом при разработке генетического алгоритма является определение целевой функции. Поскольку в нашей задаче необходимо одновременно учитывать два критерия оптимизации, мы будем использовать следующую целевую функцию:

$$F = \alpha_1 \cdot \Omega + \alpha_2 \cdot \Psi, \quad (5)$$

где α_1 и α_2 – коэффициенты критериев, с помощью которых регулируется влияние каждого из критериев на качество решения, Ω – суммарная относительная перегруженность, Ψ – общее число межслойных переходов.

Кодирование и декодирование хромосомы. В качестве исходных данных мы имеем некоторое решение задачи глобальной трассировки на графе G^I , представляющее собой множество протрассированных цепей. Каждая из цепей множества N^I представляет собой множество соединений $T_i = \{t_{ij} \mid j = 1, 2, \dots, n_j\}$.

Структура хромосомы при этом показана на рис. 3.

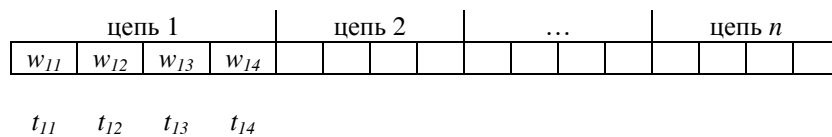


Рис. 3. Структура хромосомы

Ген g_i в хромосоме соответствует соединению, а значением гена является номер слоя, на котором необходимо проложить соответствующее этому гену соединение. В работе используются гомологичные хромосомы, что позволяет нам всегда получать легитимные решения. При декодировании хромосомы для каждой цепи осуществляется распределение соединений по слоям в соответствии со значениями генов.

Выбор метода генерации начальной популяции. Генерация начальной популяции осуществляется случайным образом. Формируется $S_{Пn}$ хромосом, как показано в предыдущем пункте. Последовательно просматриваются все хромосомы. Для каждого гена g_i в хромосоме случайным образом устанавливается значение из диапазона $[0 \dots k]$, определяющее номер слоя для двухтерминального соединения. Таким образом, мы получаем начальную популяцию хромосом $П_n^I$. После того, как все хромосомы сформированы, для каждой из них рассчитывается фитнес. Поскольку формирование популяции осуществляется случайным образом, в результате декодирования некоторых хромосом мы можем получить решения, которые не удовлетворяют ограничениям (некорректные). Часть подобных хромосом необходимо оставить в популяции, чтобы не допустить потерю оптимальных решений. Заранее определяется процент хромосом, которые будут оставлены в популяции (обычно используются значения 30–40 %). Просматриваются все некорректные хромосомы, для каждой из них с помощью (6) рассчитывается вероятность. Затем на основе полученных вероятностей определяется n_{pr}^I хромосом, которые будут оставлены в популяции. В результате получаем начальную популяцию $П_n^I$.

Выбор и разработка генетических операторов. В работе используются такие генетические операторы как кроссинговер и мутация. Поскольку хромосомы, используемые в нашем алгоритме, имеют иерархическую структуру – каждая хромосома разбита не только на гены, но еще и на участки, соответствующие протрассированным цепям – предлагается использовать двухуровневый оператор кроссинговера. Обмен будет производиться, как генами (кроссинговер второго уровня), так и целыми участками (кроссинговер первого уровня).

Кроссинговер первого уровня осуществляется следующим образом. Из текущей популяции выбирается $W1$ родительских пар хромосом ($W1$ является управляющим параметром генетического алгоритма). Вероятность $P(H_s)$ выбора хромосомы H_s в качестве родительской определяется по следующей формуле:

$$P(H_s) = \frac{F_s}{\sum_s F_s}, \tag{6}$$

где F_s – оценка хромосомы H_s . Затем случайным образом определяется множество позиций $R1_z = \{rl_{xz} \mid x = 1, 2, \dots, n1_x\}$ в каждой родительской паре. Здесь параметр $n1 = |N^l|/2$, если n четно, и $n1 = |N^l|/2 + 1$ – в противном случае. Каждая позиция соответствует участку хромосомы, представляющему целую цепь. После того, как позиции определены, производится взаимно-однозначный обмен соответствующими участками между родительскими хромосомами. Далее представлен пример кроссинговера первого уровня для числа цепей $|N^l| = 5$.

Н1	2	3	1	4	1	4	4	2	1	3	2	1	3	2	3	4	1	2	2	3	1
	цепь1				цепь2					цепь3			цепь4				цепь5				
Н2	1	1	3	2	4	3	4	4	1	2	3	1	2	4	3	1	3	2	3	1	4

Пусть были выбраны две родительские хромосомы Н1 и Н2, а также позиции цепей 1 и 4. Тогда в результате кроссинговера получим следующие дочерние хромосомы Нс1 и Нс2:

Нс1	1	1	3	2	1	4	4	2	1	3	2	1	2	4	3	1	3	2	2	3	1
	цепь1				цепь2					цепь3			цепь4				цепь5				
Нс2	2	3	1	4	4	3	4	4	1	2	3	1	3	2	3	4	1	2	3	1	4

Для кроссинговера второго уровня сначала выбирается $W2$ родительских пар хромосом, а также $R2_z$ позиций для каждой пары аналогично тому, как это выполняется в кроссинговере первого уровня. После этого производится вероятностный обмен гомологичными генами внутри каждого из выбранных участков хромосомы. Вероятность обмена P_k выбирается из диапазона 0.4–0.7 перед началом работы алгоритма. Рассмотрим пример кроссинговера второго уровня.

Н1	2	3	1	4	1	4	4	2	1	3	2	1	3	2	3	4	1	2	2	3	1
	цепь1				цепь2					цепь3			цепь4				цепь5				
Н2	1	1	3	2	4	3	4	4	1	2	3	1	2	4	3	1	3	2	3	1	4

В результате вероятностного кроссинговера был произведен обмен генами 1, 4 в первой позиции и генами 1, 2 во второй позиции. То есть, дочерние хромосомы будут выглядеть следующим образом:

Нс1	2	3	1	4	4	4	4	4	1	2	3	1	3	2	3	4	1	2	2	3	1
	цепь1				цепь2					цепь3			цепь4				цепь5				
Нс2	1	1	3	2	1	3	4	2	1	3	2	1	2	4	3	1	3	2	3	1	4

Пусть в качестве родительских хромосом были выбраны те же хромосомы H_1 и H_2 , что и для кроссинговера первого уровня. Предположим, были выбраны позиции, соответствующие цепям 2 и 3.

В результате выполнения оператора кроссинговера формируется дополнительная популяция P_k . P_k объединяется с начальной популяцией P_n , после чего на такой объединенной популяции выполняется оператор мутации.

Операция мутации в нашей работе заключается в случайном изменении значения отдельно взятого гена в хромосоме. Реализация оператора мутации выполняется следующим образом. Последовательно просматриваются все хромосомы из популяции $P_k + P_n$, а в каждой из хромосом все гены. С заранее заданной вероятностью мутации P_m ген мутирует, то есть значение гена меняется на любое другое допустимое значение. В результате выполнения оператора мутации образуется дополнительная популяция P_m .

Селекция и отбор. Заключительной операцией в пределах одного поколения является селекция расширенной популяции $P_n + P_k + P_m$. В результате селекции на основе "принципа рулетки" (в данном случае используется формула (6)) отбирается новая популяция P_n лучших решений, которая является исходной для следующей генерации. Число генераций (поколений) $T_{ген}$, размер популяции M_n и параметры P_k и P_m являются управляющими параметрами, влияющими на эффективность процесса генетического поиска. Ниже представлен упрощенный псевдокод генетического алгоритма распределения соединений по слоям.

Использование иерархической структуры генетического поиска. В современных СБИС для одного логического слоя (кремниевая подложка) может использоваться восемь, двенадцать и более слоев металлизации. Таким образом, для получения оптимальных или близких к оптимальным решений необходимо модифицировать структуру генетического поиска. В связи с этим в работе предлагается иерархический подход.

Суть метода состоит в последовательном разделении трассировочных ресурсов по принципу дихотомии и параллельном поиске решений в различных популяциях. При этом образуются *суперслои*. Каждый суперслой является некоторой абстракцией, объединяющей один, два или больше слоев и представляющей их как единое целое.

Рассмотрим иерархический подход, использующий последовательное равномерное деление трассировочных ресурсов. Пусть трассировочные ресурсы представлены графом $G^k(V^k, E^k)$, где $k = 8$, т.е. имеется восемь слоев металлизации, которые будут использоваться для распределения соединений. Введем два суперслоя Lr_1 и Lr_2 : суперслой Lr_1 объединяет в себе слои с первого по четвертый, а Lr_2 – слои с пятого по восьмой. Сначала решается задача распределения по слоям с помощью генетического алгоритма для двух суперслоев Lr_1 и Lr_2 . Формируется начальная популяция P_n^1 , гены в хромосомах этой популяции могут принимать значение 0 (соединение будет назначено в Lr_1) или значение 1 (соединение будет назначено в Lr_2). Число генов 0 и 1 примерно равно. После выполнения Ni_1 итераций алгоритма получим решение задачи, т.е. некоторое распределение соединений на Lr_1 и Lr_2 , имеющее лучшую оценку. В результате получаем некоторое число Ψ_1^1 , определяющее число межслойных переходов между Lr_1 и Lr_2 . В соответствии с полученным распределением формируются две новые непересекающиеся популяции P_n^{11} и P_n^{12} . Хромосомы популяции P_n^{11} содержат гены, соответствующие соединениям, попавшим после первого этапа в суперслой Lr_1 , а хромосомы из P_n^{12} – соединениям, попавшим в Lr_2 . Ниже показана хромосома H^1 , полученная в качестве оптимального решения из популяции P_n^1 , а также две новые хромосомы H^{11} и H^{12} из вновь сформированных популяций P_n^{11} и P_n^{12} .

H^1	1	0	1	0	1	0	0	1	0	1	0	1	1	0	1	1	0	0	1
	цепь1				цепь2				цепь3		цепь4			цепь5					

H^{11}					H^{12}														
1	0	1	0	1	0	1	1	0	0	1	1	0	1	0	1				
ц1		ц2		ц3		ц4		ц5		ц1		ц2		ц3		ц4		ц5	

Здесь сверху для каждого соединения показан номер суперслоя, в который оно будет назначено. Как и на первом этапе, гены в хромосомах могут принимать значения 0 и 1. Далее процесс распараллеливается и выполняется поиск решений в двух популяциях. Каждый суперслой, в свою очередь, делится на два суперслоя. В результате получаем две пары суперслоев: Lr_{11} (включает слои 1 и 2) и Lr_{12} (включает слои 3 и 4), Lr_{21} (включает слои 5 и 6) и Lr_{22} (включает слои 7 и 8). Параллельный процесс эволюции в нескольких популяциях предполагает такой генетический оператор как *миграция*. Рассмотрим *миграцию*, предложенную в нашей работе.

Из каждой популяции выбирается по одной хромосоме, при этом не важно, какие именно будут использоваться хромосомы, так как важен только порядок расположения генов, а не их значения. В каждой хромосоме случайным образом выбирается ген g_i^z , который будет мигрировать. После того, как гены определены, с некоторой вероятностью $P_{migration}$ выполняется оператор миграции. Здесь стоит обратить внимание на тот факт, что гены g_i^1 и g_i^2 негомологичны, поскольку мы берем хромосомы из разных популяций. Обмен производится именно целыми генами. После проведения миграции получаем две модифицированные популяции P_M^{11} и P_M^{12} . Новый набор генов будут содержать все хромосомы из P_M^{11} и P_M^{12} .

Заключение. В работе предлагается генетический алгоритм для решения задачи распределения соединений по слоям при многослойной глобальной трассировке. Такой подход использует модифицированные операторы кроссинговера и миграции, что позволяет улучшить качество получаемых решений при трассировке. Временная сложность алгоритма на одной итерации оценивается как $O(n)$. График сходимости алгоритма представлен на рис. 4.

Как показывают экспериментальные исследования, предложенный подход с использованием генетического алгоритма позволяет получать решения, не уступающие по качеству известным многослойным глобальным трассировщикам.



Рис. 4. График сходимости алгоритма

Для испытания нашего алгоритма мы использовали бенчмарки для шестислойных схем, которые использовались на ISPD'07 для конкурсного тестирования. Полученные результаты сравнивались с результатами трассировщиков **MaizeRouter**, **BoxRouter** и **FGR**, показавших лучшие результаты на этих бенчмарках. Кроме того, для тестирования использовались собственные модели, приближенные к реальным СБИС. Сравнение показало улучшение качества решений до 2 %.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Лебедев Б.К., Лебедев В.Б.* Поисквые процедуры канальной трассировки, базирующиеся на моделировании адаптивного поведения роя частиц в пространстве решений с неупорядоченным лингвистическим шкалированием // *Известия ЮФУ. Технические науки.* – 2009. – № 12 (101) – С. 15-22.
2. *Курейчик В.В., Курейчик В.М., Гладков Л.А., Сороколетов П.В.* Бионспирированные методы в оптимизации. – М.: Физмалит, 2009.
3. *Курейчик В.М., Кажаров А.А.* Использование роевого интеллекта в решении NP-трудных задач // *Известия ЮФУ. Технические науки.* – 2011. – № 7 (120). – С. 30-36.
4. *Литвиненко В.А., Ховансков С.А., Норкин О.Р.* Оптимизации мультиагентной системы распределенных вычислений // *Известия ЮФУ. Технические науки.* – 2009. – № 4 (93). – С. 226-235.
5. *Charles J. Alpert, Dinesh P. Mehta, Sachin S. Sapatnekar.* Handbook of algorithms for physical design automation. CRC Press, New York, USA, 2009.
6. *Minsik Cho, Katrina Lu, Kun Yuan, David Z. Pan,* BoxRouter 2.0: A hybrid and robust global router with layer assignment for routability, *ACM Transactions on Design Automation of Electronic Systems*, volume 14 issue 2, March 2009.
7. *Cong J., Fang J., Xie M. and Y. Zhang.* MARS—A multilevel full-chip gridless routing system, *IEEE Trans. Comput.-Aided Design Integr. Syst.*, vol. 24, no. 3, pp. 382–394, Mar. 2005.
8. *Roy J.A.* High-Performance routing at the nanometer scale, *IEEE Trans.Comput.-Aided Design Integr. Syst.* – June 2008. – Vol. 27, № 6. – P. 1066-1077.
9. *Лебедев О.Б.* Глобальная трассировка на основе муравьиного алгоритма // *Известия ЮФУ. Технические науки.* – 2011. – № 7 (120). – С. 94-102.
10. *Лебедев Б.К., Лебедев В.Б.* Глобальная трассировка на основе роевого интеллекта // *Известия ЮФУ. Технические науки.* – 2010. – № 7 (108) – С. 32-39.
11. *Лебедев В.Б.* Построение кратчайших связывающих сетей на основе роевого интеллекта // *Известия ЮФУ. Технические науки.* – 2011. – № 7 (120). – С. 37-44.
12. FGR 1.1. [Online]. Available: <http://vlsicad.eecs.umich.edu/BK/FGR>.

Статью рекомендовал к опубликованию д.т.н., профессор В.П. Карелин.

Лебедев Борис Константинович – Технологический институт федерального государственного автономного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге; e-mail: lbk@tsure.ru; 347928, г. Таганрог, пер. Некрасовский, 44, ГСП 17А; тел.: 88634371743; кафедра систем автоматизированного проектирования; профессор.

Воронин Егор Ильич – e-mail: egorvoronin04@rambler.ru; кафедра систем автоматизированного проектирования; аспирант.

Lebedev Boris Konstantinovich – Taganrog Institute of Technology – Federal State-Owned Autonomy Educational Establishment of Higher Vocational Education “Southern Federal University”; e-mail: lbk@tsure.ru; GSP 17A, 44, Nekrasovsky, Taganrog, 347928, Russia; phone: +78634371743; the department of computer aided design; professor.

Voronin Egor Ijich – e-mail: egorvoronin04@rambler.ru; the department of computer aided design; postgraduate student.