

Размещение элементов СБИС на основе моделей роевого интеллекта

Б.К. Лебедев, О.Б. Лебедев, А.А. Жиглатый

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

Аннотация — Разработана архитектура гибридного алгоритма размещения на основе интеграции методов роя частиц и пчелиной колонии. Представлена модифицированная парадигма роя частиц, отличающаяся от канонической, возможностью использования в аффинном пространстве позиций с целочисленными значениями параметров. Предложена модифицированная структура алгоритма пчел. Ключевой операцией алгоритма является формирование и исследование перспективных позиций, лежащих в окрестностях базовых позиций. Тестовые испытания доказали, что при интеграции моделей поведения роя пчел и роя частиц, результаты нового гибридного алгоритма получаются на 11-18% лучше, чем у каждого алгоритма по отдельности.

Ключевые слова — СБИС, размещение, роевой интеллект, пчелиный алгоритм, рой частиц, гибридизация, многоагентная система, аффинное пространство поиска, направленная мутация, окрестность базовых позиций бионический поиск.

I. ВВЕДЕНИЕ

Сверхбольшие интегральные схемы на основе базовых стандартных библиотечных элементов отличаются использованием заранее спроектированных библиотечных элементов и макроблоков. Размещение блоков и трассировка межсоединений определяется макрокомпоновкой топологии полузаказной СБИС. Цель размещения состоит в нахождении для каждого элемента конкретной позиции на топологии.

Задача размещения элементов СБИС [1]-[3] относятся к классу NP-трудных задач комбинаторной оптимизации. Она достаточно неплохо изучена и имеется большое количество алгоритмов, позволяющих ее решать. В настоящее время, всё чаще применяются методы на основе искусственного интеллекта [4], [5]. Такие методы опираются на моделирование коллективного интеллекта [6]-[8] и к ним относится метод пчелиной колонии [9]-[12].

Сообщества этих насекомых обладают коллективным интеллектом. Организация поведения дает возможность этим сообществам выполнять задачи, которые не могут быть выполнены каждым насекомым в отдельности. Это достигается при помощи совместных (коллективных) действиях и не сложного взаимодействия между членами сообщества. Такие сообщества обладают способностями к самоорганизации и адаптации.

Алгоритмы, построенные на основе таких способностей сообществ заключаются в перемещении членов коллектива (агентов) в пространстве поиска. Движение агентов осуществляется по позициям. От определенных агентами позиций зависит значение целевой функции. Множество частиц и пчел представляется как многоагентная система, в которой каждая частица или пчела выполняют движение самостоятельно, по тривиальным правилам.

Проанализировав известные подходы, применяемые для решения «сложных» задач, можно сделать вывод, что использование какого-либо одного алгоритма не дает гарантии получения качественного решения. В связи с этим, в настоящее время один из способов улучшения эффективности методов нахождения глобального оптимума решаемых задач, состоит в гибридизации алгоритмов [13]. Особенностью гибридного алгоритма является то, что достоинства одного алгоритма могут возместить недостатки другого. Объединение различных методик поисковых алгоритмов, дает возможность определения большей области допустимых решений и нахождение оптимального решения.

В работе описан разработанный алгоритм решения задачи размещения элементов СБИС, объединяющий процедуры поведения коллектива пчел и роя частиц.

II. ПОСТАНОВКА ЗАДАЧИ И КОНЦЕПЦИЯ РАЗМЕЩЕНИЯ ЭЛЕМЕНТОВ МЕТОДОМ ПАРЫ ПОСЛЕДОВАТЕЛЬНОСТЕЙ

Характеристиками, определяющими суть постановки задачи размещения, являются: модель представления размещаемых (конструктивных) элементов как геометрических объектов; модель монтажного пространства (пространство позиций); модель представления электрической принципиальной схемы; характер целевой функции для оценки размещения [3], [14]. Необходимо разместить элементы на коммутационном поле с оптимизацией некоторых критериев качества.

Основными, известными критериями при размещении [1], [3], [14] являются: суммарная длина связи, длина самой длинной связи, число возможных пересечений, число изгибов соединений, площадь кристалла и др.

Рассмотрим концепцию размещения элементов методом пары упорядоченных списков A_1 и A_2 одного и

того же множества элементов. $|A_1|=|A_2|=n$ [15]. Для заданной пары списков осуществляется формирование графа ограничений путем последовательного построения мета-сетки с наклоном в 45 градусов.

Пусть имеется пара последовательностей $(A_1=4,3,1,6,2,5; A_2=6,3,5,4,1,2)$. Формирование графа ограничений для заданной пары списков осуществляется путем последовательного построения мета-сетки с наклоном в 45 градусов, рис. 1.

Для каждого элемента плоскость делится двумя наклонными прямыми на четыре сектора, как показано на рис. 1 для элемента 1. Последовательная пара задает расположение некоторой пары элементов a_i и a_j по секторам, в зависимости от их взаимного расположения этих списках:

Если $(A_1=\langle \dots, a_i, \dots, a_j, \dots \rangle, A_2=\langle \dots, a_i, \dots, a_j, \dots \rangle)$, то a_j *правее* a_i ;

Если $(A_1=\langle \dots, a_i, \dots, a_j, \dots \rangle, A_2=\langle \dots, a_j, \dots, a_i, \dots \rangle)$, то a_j *ниже* a_i .

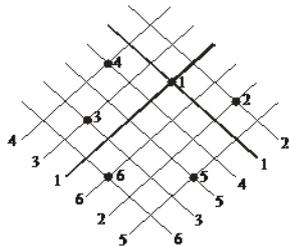


Рис. 1. Взаимное расположение элементов на мета-сетке

Рассмотрим расположение элементов относительно элемента 1. Для нашей пары последовательностей элемент 2 находится в правом секторе от элемента 1 (рис. 2), так как в обеих последовательностях 2 правее 1. Элементы 6 и 5 находятся в нижнем секторе от элемента 1 и т.д.

Размещение элементов, задаваемое парой представлено на рис. 2.

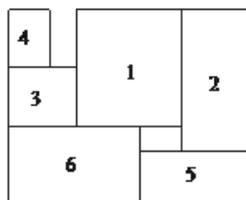


Рис. 2. Размещение элементов

Следовательно, имея пару упорядоченных списков (A_1, A_2) , задающую горизонтальное отношение между элементами, можно построить граф горизонтальных ограничений $R_h(V_h, E_h)$ следующим образом:

$$V_h = \{s_h\} \cup \{t_h\} \cup \{v_i | i=1..n\},$$

$$E_h = \{(s_h, v_i) | i=1..n\} \cup \{(v_i, t_h) | i=1..n\} \cup \{(v_i, v_j) | a_i \text{ левее } a_j\},$$

где v_i соответствует элементу, s_h – исходная вершина, представляющая левую границу, t_h – конечная вершина, представляющая правую границу. Вес вершины v_i равен

ширине элемента a_i . Вес вершин s_h, t_h , равен нулю. Граф вертикальных ограничений $R_v(V_v, E_v)$ строится аналогично.

Пример: на рис. 3 изображены оба графа для размещения, представленного на рис. 2.

Оба графа – с взвешенными вершинами, направлены и ациклически, то есть мы можем применить алгоритм нахождения длиннейшего пути для определения XU координат для каждого элемента. За координату элемента примем координату его нижнего левого угла.

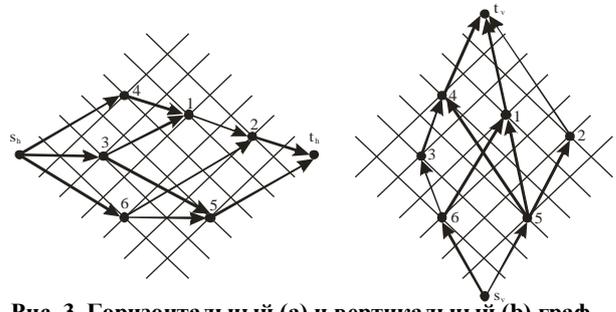


Рис. 3. Горизонтальный (а) и вертикальный (б) граф ограничений

Процесс перехода от пары упорядоченных списков к размещению реализуется в два этапа. Сначала на базе ограничений или упорядоченное дерево (УД) ограничений, покрывающее граф ограничений, а затем по дереву ограничений тривиальным алгоритмом строится план или размещение элементов.

III. ПОДХОД К ПРЕДСТАВЛЕНИЮ РЕШЕНИЙ В АЛГОРИТМЕ НА ОСНОВЕ РОЕВОГО ИНТЕЛЛЕКТА

В методе оптимизации роем частиц агентами являются частицы в пространстве параметров задачи оптимизации. Каждая частица связана со всем роем, может взаимодействовать со всем роем и она тяготеет к лучшему решению роя. В каждый момент времени (на каждой итерации) частицы занимают в этом пространстве некоторые позиции. Для каждой позиции 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 свою позицию в пространстве поиска [7], [8].

Основная идея парадигмы пчелиного роя [8]-[13] состоит в выполнении двухуровневой стратегии поиска. На первом уровне в пространстве поиска генерируется множество базовых позиций. На втором уровне в пространстве поиска исследуются окрестности базовых позиций. Лучшие позиции, найденные в каждой окрестности базовых позиций на итерации t , входят в состав множества базовых позиций, используемых на итерации $(t+1)$.

Цель сообщества пчел найти позицию, с наилучшим значением целевой функции.

В отличие от канонической парадигмы роя частиц гибридные поисковые алгоритмы в качестве моделей для представления решений используют широкий диапазон графовых структур (маршрут, дерево, двудольный граф, парасочетание, внутренне устойчивое множество и т.д.) [14]-[22]. Это не позволяет напрямую использовать каноническую парадигму роя частиц (например, задача направленной мутации одного дерева в направлении другого, с формальной точки зрения, является весьма нетривиальной). В связи с этим актуальной является разработка модернизированной структуры пространства поиска, структуры данных для представления решений и позиций, модернизированных механизмов перемещения частиц в пространстве поиска.

Предлагается подход к построению модифицированной парадигмы роя частиц, обеспечивающей возможность одновременного использования хромосом с целочисленными значениями параметров в пчелином алгоритме и в алгоритме на основе роя частиц.

IV. АФФИННОЕ ПРОСТРАНСТВО И ПОИСК МЕТОДОМ РОЯ ЧАСТИЦ

Пусть имеется линейное векторное пространство (ЛВП), элементами которого являются n -мерные точки. Каждым любым двум точкам p и q этого пространства однозначным образом сопоставим единственную упорядоченную пару этих точек, которую в дальнейшем будем называть геометрическим вектором (вектором). $p, q \in V(p, q)$ – геометрический вектор (упорядоченная пара). Совокупность всех точек ЛВП, пополненная геометрическими векторами, называют точечно-векторным или аффинным пространством. Аффинное пространство является n -мерным, если соответствующее ЛВП так же является n -мерным.

Аффинно-релаксационная модель (АРМ) роя частиц – это граф, вершины которого соответствуют позициям роя частиц, а дуги соответствуют аффинным связям между позициями (точками) в аффинном пространстве. Мера аффинности – степень похожести (близости) двух позиций, например, какое-либо расстояние между их векторами характеристик. Аффинность определяет целевую функцию, которая показывает, насколько данные две позиции схожи (близки) друг другу. Используются функции расстояния (например, евклидова) между двумя точками n -мерного пространства.

На каждой итерации каждый агент p_i переходит в аффинном пространстве в новое состояние (позицию), при котором вес аффинной связи между агентом p_i и базовым (лучшим) агентом p^* уменьшается. Переход агента p_i в новую позицию $x_i(t+1)$ из $x_i(t)$ осуществляется с помощью релаксационной процедуры. Специальная релаксационная процедура перехода зависит от вида структуры данных

(хромосомы): вектор, матрица, дерево и их совокупности, являющейся интерпретацией решений.

Лучшие частицы с точки зрения целевой функции объявляются «центром притяжения». Векторы перемещения всех частиц в аффинном пространстве устремляются к этим центрам. Переход возможен с учетом степени близости к одному базовому элементу либо к группе соседних элементов и с учетом вероятности перехода в новое состояние.

В эвристических алгоритмах роевого интеллекта многомерное пространство поиска населяется роем частиц [7], [8]. В процессе поиска методом роя частиц каждая частица перемещается в новую позицию. Новая позиция в канонической парадигме роя частиц определяется в соответствии с методикой, описанной в работе [9].

В отличие от канонического метода роя частиц, в работе в качестве позиции частицы служит хромосома с целочисленными значениями параметров. Пусть имеется рой частиц $P = \{p_k | k = 1, 2, \dots, n_k\}$. Каждая частица p_k на шаге t размещена в позиции $X_k(t)$. Поскольку размещение задается парой упорядоченных списков A_1 и A_2 позиция $X_k(t)$, соответствующая частице p_k , определяется набором из 2 хромосом, соответствующих паре упорядоченных списков A_1 и A_2 $X_k(t) = \{H_{k1}(t), H_{k2}(t)\}$.

Хромосома $H_{ki}(t) = \{g_{kil} | l = 1, 2, \dots, n_l\}$ представляет собой совокупность n генов g_{kil} . Значение гена g_{kil} равно значению соответствующего элемента списка A_i .

Пространство поиска *одной хромосомы* включает число осей X_{il} равное числу генов в хромосоме $H_{ki}(t)$. Каждому гену в хромосоме $H_{ki}(t)$ соответствует ось (номер оси). Точками отсчета на каждой оси X_{il} являются целые числа в интервале от 0 до n .

Пусть в качестве базового списка используется список $B = \langle 1, 7, 21, 4, 8, 18 \rangle$. $|B| = 6$. Пространство поиска X включает 6 осей: $X_1 - X_6$, в соответствии с числом элементов списка. Каждая ось соответствует позиции списка. Точками отсчета x_{ij} на оси $X_i = \langle x_{ij} | j = 1, 2, \dots, 6 \rangle$ являются упорядоченные значения элементов списка B : $X_i = \langle x_{i1} = 1, x_{i2} = 7, x_{i3} = 21, x_{i4} = 4, x_{i5} = 8, x_{i6} = 18 \rangle = \langle 1, 7, 21, 4, 8, 18 \rangle$.

Например:

Список $M_i = \langle 21, 8, 7, 1, 8, 4, 18 \rangle$ представляется в виде позиции $H_i = \{x_{13}, x_{25}, x_{32}, x_{41}, x_{54}, x_{66}\}$.

В алгоритме роя частиц на каждой итерации частицы роя перемещаются в новые позиции.

Авторами была разработана *процедура перемещения*, которая реализуется при помощи операции направленной мутации (ОНМ), суть которого заключается в изменения взаимного расположения элементов в списке. Частица p_i перемещается в направлении позиции $H_c(t)$ из позиции $H_i(t)$ в новую позицию $H_i(t+1)$ с новым взаимным расположением элементов в списке. Работа операции направленной мутации подробно описана в работе [9].

Введем характеристику $S_{iz}(t)$, отражающую степень различия между позициями $H_i(t)$ и $H_z(t)$. Для этого произведем сравнение взаимного расположения элементов у всех возможных пар в сравниваемых позициях $H_i(t)$ и $H_z(t)$. Пусть $S_{iz}(t)$ – число пар, у которых взаимное расположение элементов в сравниваемых позициях $H_i(t)$ и $H_z(t)$ не совпадает. Модификация позиции $H_i(t)$, то есть переход к позиции $H_i(t+1)$, производится путём выборочных групповых парных перестановок соседних элементов в позиции $H_i(t)$. Модификация позиции $H_i(t)$ выполняется за два такта.

На первом такте формируется множество D_1 непересекающихся пар элементов в позиции $H_i(t)$ таких, у которых индекс l – нечетное число: $P_1 = \{(g_{il}(t), g_{i,l+1}(t)) | l=1,3,5,\dots\}$. На втором такте формируется множество D_2 непересекающихся пар элементов в позиции $H_i(t)$, у которых индекс l – четное число: $D_2 = \{(g_{il}(t), g_{i,l+1}(t)) | l=2,4,6,\dots\}$. Подсчитывается число пар $S_{iz}(t)$ в множестве $D = D_1 \cup D_2$, у которых взаимное расположение элементов пары в позиции $H_i(t)$ не совпадает с взаимным расположением этих элементов в позиции $H_z(t)$. Принимается решение о перестановке элементов каждой такой пары в позиции $H_i(t)$, вычисляются показатели $S_{ik}(t)$ и формируется позиция $H_i(t+1)$.

Перестановка выполняется с вероятностью $P = \alpha \cdot S_{iz}(t) / n$, где n – число пар в множестве D .

Главная цель передвижения частицы p_k – нахождение ею позиции с наилучшей оценкой. Общая цель роя частиц – получение оптимального решения задачи размещения.

V. АДАПТИВНОЕ ПОВЕДЕНИЕ ПЧЕЛИНОЙ КОЛОНИИ

Пусть имеется сообщество пчел $P = \{p_k | k=1,2,\dots,n_k\}$. На первой итерации ($t=1$) пчелы-разведчики случайным образом размещаются в области поиска решений. Каждая пчела разведчик p_k на шаге t выбирает позицию $X_k(t)$. Каждая позиция является аналогией места нектара и моделью решения задачи размещения. Объем нектара является значением критерия в этой точке.

Так же, как и в алгоритме роя частиц, позиция $X_k(t)$, выбранная пчелой определяется набором из 2 хромосом, соответствующих паре упорядоченных списков A_1 и A_2 . $X_k(t) = \{H_{k1}(t), H_{k2}(t)\}$.

Хромосома $H_{ki}(t) = \{g_{kil} | l=1,2,\dots,n_i\}$ представляет собой упорядоченную совокупность n генов g_{kil} . Значение гена g_{kil} равно значению соответствующего элемента списка A_i .

Первая операция выполняемая роем пчел заключается в генерации случайным образом множества отличающихся друг от друга (позиций) решений $X(t) = \{X_k(t) | k=1,2,\dots,n_r\}$. Для каждого решения $X_k(t)$ вычисляется значение целевой функции F_k . В множестве $X(t)$ выбираются n_δ лучших решений, которые включаются в множество базовых решений (позиций) $X^\delta(t)$. При этом реализуется вероятностный способ формирования $X^\delta(t)$. Вероятность $p(X_k)$

включения агентом фуражиром позиции $X_k \in X(t)$, $X_k(t) = \{H_{k1}(t), H_{k2}(t)\}$ в состав множества X_k базовых позиций пропорциональна значению целевой функции F_k в этой позиции и определяется как:

$$p(X_k(t)) = F_k(t) / \sum_k (F_k(t)).$$

Для каждой базовой позиции $X^\delta_k(t) \in X^\delta(t)$, $X^\delta_k(t) = \{H^\delta_{k1}(t), H^\delta_{k2}(t)\}$ реализуется вероятностный выбор набора позиций $O^\delta_k(t+1)$, расположенных в θ – окрестности базовой позиции $X^\delta_k(t)$.

Формирование новой позиции $X^\delta_k(t+1)$, лежащей в θ – окрестности базовой позиции $X^\delta_k(t)$ производится путём δ выборочных (случайных) парных перестановок соседних элементов в хромосомах $\{H^\delta_{k1}(t), H^\delta_{k2}(t)\}$ позиции $X^\delta_k(t)$. Будем считать, что позиция $X^\delta_k(t+1)$ лежит в θ – окрестности позиции $X^\delta_k(t)$, если $X^\delta_k(t+1)$ получено путем θ случайных парных перестановок соседних элементов в хромосомах $\{H^\delta_{k1}(t), H^\delta_{k2}(t)\}$ позиции $X^\delta_k(t)$.

Обозначим множество позиций, выбранных в θ – окрестности позиции $X^\delta_k(t)$, как $O^\delta_k(t+1)$. Рассчитывается оценка каждой позиции множества $O^\delta_k(t+1)$. В каждой θ – окрестности $O^\delta_k(t+1)$ выбирается лучшая позиция. Лучшие позиции θ – окрестностей образуют новое множество базовых позиций $X^\delta(t+1)$.

На каждой итерации осуществляется переход от одной популяции базовых позиций к другой.

Лучшее решение (позиция) множества $X^\delta(t+1)$ сохраняется, а затем происходит переход к следующей итерации. В начале второй и на последующих итерациях, прежде всего, формируется множество базовых позиций $X^\delta(t)$ ($t=2,3,\dots,L$), составленное из двух частей $X^{\delta 1}(t)$ и $X^{\delta 2}(t)$, $X^{\delta 1}(t) \cup X^{\delta 2}(t) = X^\delta(t)$. В первую часть $X^{\delta 1}(t)$ включаются $n_{\delta 1}$ лучших позиций, среди позиций $X^*_k(t-1)$, найденных агентами в каждой из областей, сформированных на предыдущей итерации. Вторая часть $X^{\delta 2}(t)$ формируется агентами разведчиками также, как и на первой итерации. Далее выполняются действия, аналогичные действиям, рассмотренным на первой итерации.

VI. ГИБРИДИЗАЦИЯ СТРУКТУРЫ РОЕВОГО ИНТЕЛЛЕКТА

Алгоритмы на основе коллективного поведения, содержат область поиска решений, в которой находится рой агентов. Местонахождение каждого агента является каким-то решением. Нахождение решения состоит в передвижении агентов в области допустимых значений. В нашем случае рой рассматривается как множество решений, а пчела (частица) как агент, что позволяет сформировать гибридные процедуры нахождения решений, путем объединения моделей роя частиц и сообщества пчел на основе коллективной адаптации.

Разработанный алгоритм решения задачи размещения элементов СБИС, использует архитектуру бионического поиска, и состоит из объединенных процедур сообщества пчел и роя частиц, что позволяет выходить из «локальных ям» и увеличивает сходимость

алгоритма. Данные представляются в матричном или векторном виде.

Каждая частица p_k на шаге t размещена в позиции $X_k(t)$, поскольку размещение задается парой упорядоченных списков A_1 и A_2 . Позиция $X_k(t)$, соответствующая частице p_k , определяется набором из 2 хромосом, соответствующих паре упорядоченных списков A_1 и A_2 , $X_k(t) = \{H_{k1}(t), H_{k2}(t)\}$.

Хромосома $H_{ki}(t) = \{g_{kil} | l=1, 2, \dots, n_l\}$ представляет собой совокупность n генов g_{kil} . Значение гена g_{kil} равно значению соответствующего элемента списка A_i .

При движении в области допустимых значений сообщество агентов по очередности представляется как сообщество пчел или сообщество частиц, с определенными свойствами адаптивного поведения.

Гибридизация заключается в следующем. Вначале формируется рой местоположений $X(t) = \{X_k(t) | k=1, 2, \dots, n_r\}$, при помощи алгоритма роя частиц. Применяя методику роя частиц, в области допустимых значений при передвижении частиц из места $X(t-1)$ находятся места роя частиц $X(t)$. В соответствии с предлагаемым подходом к интеграции новые позиции рассматриваются как базовые позиции $X^0(t)$, обнаруженные роем пчел разведчиков. Далее, в соответствии с механизмами пчелиной колонии пчелами фуражирами исследуются θ – окрестности каждой из базовых позиций множества $X^0(t)$. Ключевой операцией пчелиного алгоритма является исследование перспективных позиций и их θ – окрестностей в пространстве решений. В каждой θ – окрестности $O^0_k(t+1)$ выбирается лучшая позиция. Лучшие позиции θ – окрестностей образуют новое множество базовых позиций $X^0(t+1)$. На последующей $(t+1)$ -ой итерации множество позиций $X^0(t+1)$ рассматривается как множество позиций роя частиц. Общая оценка зависимости времени работы гибридного алгоритма находится в пределах $O(n^2) - O(n^3)$.

VII. ЭКСПЕРИМЕНТАЛЬНЫЕ ИССЛЕДОВАНИЯ

Исследования роевого алгоритма размещения (РАР) состоят в формировании тестовых заданий для задачи размещения с имеющимся оптимальным результатом (РЕКО) [22]. Оптимальные результаты РЕКО имеются в обоих форматах GSRC BookShelf и LEF/DEF, и они доступны в сети [23], [24].

Все цепи в РЕКО являются локальными, т.е. длина проводников каждой цепи имеет минимально возможное значение. Схемы набора РЕКУ состоят из локальных цепей в стиле РЕКО. Для экспериментальных исследований разработанной программы размещения были применены схемы набора РЕКУ с известным оптимумом F_{opt} : Ex.1 на 30 блоков, Ex.2 – 60, Ex.3 – 90, Ex.4 – 120, Ex.5 – 150. Для сравнения были выбраны современные алгоритмы размещения: Dragon v.2.20 [25], Capo v.8. [26], mPL v.2.0 [27], mPG v1.0 [28] и QPlace v.5.1. [29].

Для определения оптимальности достигнутых значений, рассчитывался параметр: длина соединения к

наилучшей длине соединений (для РЕКО) или (для G-РЕКУ и РЕКУ). Это отношение называется «степенью качества». Ни один из алгоритмов размещения при исследовании не достиг значения коэффициента качества близкого к 1.

В табл. 1. приведены полученные значения показателя степень качества ряда известных алгоритмов и алгоритма РАР.

Таблица 1

Значения показателя «степень качества»

Тест	Dragon	Capo	mPL	mPG	Qplace	РАР
Ex.1	0.71	0.73	0.81	0.71	0.73	0.9
Ex.2	0.72	0.72	0.84	0.72	0.78	0.89
Ex.3	0.81	0.8	0.83	0.81	0.82	0.89
Ex.4	0.83	0.84	0.85	0.83	0.81	0.91
Ex.5	0.82	0.81	0.86	0.82	0.84	0.91

Степень качества у разработанной программы РАР на 10% больше, чем у программ Dragon, Capo, mPL, mPG и Qplace. ВСА $O(n^2)$.

Анализ сходимости алгоритма РАР осуществлялся следующим образом. Каждое тестовое задание запускалось на выполнение 10 раз. Для каждого теста определялся номер итерации, после которой далее не происходило улучшения критерия. На рис. 4 представлены результаты эксперимента. Тестирование выявило, что алгоритм определяет лучший результат на 111-131 итерации. Алгоритм сходится на 128 итерации (рис. 4.).

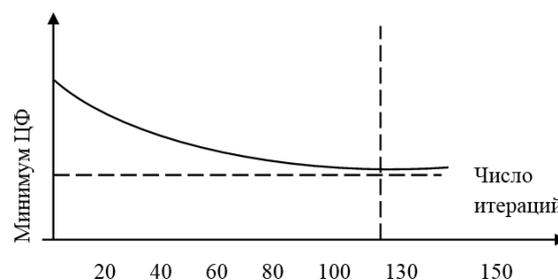


Рис. 4. Оценка качества гибридного алгоритма

Временная сложность алгоритма при фиксированных значениях размера популяции и количества генераций составляет $O(n)$. Общая временная сложность гибридного алгоритма составляет $O(n^2) - O(n^3)$.

VIII. ЗАКЛЮЧЕНИЕ

На основе анализа известных методик, применяемых для получения оптимальных решений в комбинаторно-логических задачах, выбраны подходы содержащие модели коллективного поведения, дающие возможность решать «сложные» задачи и при этом достигать оптимальных значений критерия за приемлемое время.

Алгоритм состоит из объединенных процедур сообщества пчел и роя частиц, что позволяет выходить

из «локальных ям» и увеличивает сходимость алгоритма. В предложенном методе данные представляются в матричном или векторном виде.

В работе описывается модифицированная парадигма роя частиц, обеспечивающая, в отличие от канонического метода, возможность поиска решений в аффинном пространстве позиций с целочисленными значениями параметров. Рассмотрены механизмы перемещения частиц в аффинном пространстве для уменьшения веса аффинных связей. Описываются операторы направленной мутации, суть которых заключается в изменения целочисленных значений генов в хромосоме. Предложена модифицированная структура алгоритма пчел. Для каждой базовой позиции реализуется вероятностный выбор набора позиций, расположенных в окрестности базовой позиции.

Улучшить качество работы разработанного алгоритма можно при помощи настройки значений управляющих параметров.

Общая зависимость времени работы гибридного алгоритма составляет $O(n^2) - O(n^3)$.

ПОДДЕРЖКА

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

ЛИТЕРАТУРА

- [1] Норенков И.П. Основы автоматизированного проектирования. М.: Изд-во МГТУ имени Н.Э.Баумана, 2006. 248 с.
- [2] Alpert C.J., Mehta D.P., Sapatnekar S.S., Handbook of Algorithms for Physical Design Automation. Boston, MA: Auerbach, 2009. 245 p.
- [3] Казеннов Г.Г. Основы проектирования интегральных схем и систем. М.: Бином. Лаборатория знаний, 2005. 295 с.
- [4] Wang X. Hybrid nature-inspired computation method for optimization // Doctoral Dissertation. Helsinki University of Technology, TKK Dissertations, Espoo 2009, 161 p.
- [5] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие. М.: Издательство МГТУ им. Н.Э. Баумана, 2014. 448 с.
- [6] Poli R. Analysis of the publications on the applications of particle swarm optimisation. Journal of Artificial Evolution and Applications, Article ID 685175, 2008.
- [7] Clerc M. Particle Swarm Optimization. ISTE, London, UK, 2006. 198 p.
- [8] Kennedy J., Eberhart R.C. Particle swarm optimization // In Proceedings of IEEE International Conference on Neural Networks, 1995. pp. 1942-1948.
- [9] Лебедев Б.К., Лебедев О.Б., Лебедева Е.О., Нагабедян А.А. Гибридный роевой алгоритм глобальной оптимизации в аффинном пространстве поиска // Электронный журнал Программные продукты, системы и алгоритмы. Тверь: Изд-во «Центрпрограммсистем», 2019. № 1.
- [10] Lučić P., Teodorović D. Computing with Bees: Attacking Complex Transportation Engineering Problems // International Journal on Artificial Intelligence Tools. 2003. № 12, pp. 375-394.

- [11] Quijano N., Passino K.M. Honey Bee Social Foraging Algorithms for Resource Allocation: Theory and Application. Columbus: Publishing house of the Ohio State University, 2007. 139 p.
- [12] Лебедев В.Б. Метод пчелиной колонии в комбинаторных задачах на графах // Тринадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2012. Труды конференции. Т. 2. М.: Физматлит, 2012. С. 414-422.
- [13] Лебедев Б.К., Лебедев В.Б. Размещение на основе метода пчелиной колонии // Известия ЮФУ. Изд-во ТТИ ЮФУ, 2010. № 12. С. 12-18.
- [14] Лебедев Б.К., Лебедев В.Б., Лебедев О.Б. Методы, модели и алгоритмы размещения. Ростов-на-Дону: Изд-во ЮФУ, 2015. 181 с.
- [15] Xu J., Guo P.-N., Cheng C.-K. Sequence-Pair Approach for Rectilinear Module Placement // IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems, 1999. № 4. pp. 204-218.
- [16] Лебедев Б.К., Лебедев В.Б. Покрытие методом роя частиц // Материалы VI-й Международной научно-практической конференции «Интегрированные модели и мягкие вычисления в искусственном интеллекте». М.: Физматлит, 2011. С. 611-619.
- [17] Лебедев Б.К., Лебедев О.Б., Лебедева Е.М. Распределение ресурсов на основе гибридных моделей роевого интеллекта // Научно-технический вестник информационных технологий, механики и оптики. Изд-во С.-Петербург, 2017. Т. 17. № 6. С. 1063-1073.
- [18] Лебедев Б.К., Лебедев В.Б. Трассировка на основе метода роя частиц // Двенадцатая национальная конференция по искусственному интеллекту с международным участием КИИ-2010. Труды конференции. Т.2. М.: Физматлит, 2010. С. 414-422.
- [19] Лебедев Б.К., Лебедев О.Б., Лебедев В.Б. Гибридизация роевого интеллекта и генетической эволюции на примере размещения // Электронный журнал Программные продукты, системы и алгоритмы – Тверь: Изд-во «Центрпрограммсистем», 2017. № 4.
- [20] Лебедев Б.К., Лебедев В.Б. Планирование на основе роевого интеллекта и генетической эволюции // Известия ЮФУ. Изд-во ТТИ ЮФУ, 2009. № 2. С. 25-33.
- [21] Лебедев Б.К., Лебедев О.Б., Лебедева Е.О. Роевой алгоритм планирования работы многопроцессорных вычислительных систем // Электронный научный журнал «Инженерный вестник Дона». 2017. № 3.
- [22] Cong J., Romesis M., Xie M. Optimality, Scalability and Stability Study of Partitioning and Placement Algorithms // Proc. of the International Symposium on Physical Design. Monterey, CA, 2003. pp. 88-94.
- [23] Cong J., Romesis M. и Xie M. UCLA Optimality Study Project. <http://cadlab.cs.ucla.edu/pubbench>. 2004.
- [24] MCNC. Electronic and Information Technologies (Online). Available: www.mcnc.org. (28.03.2020).
- [25] Wang M., Yang X. and Sarrafzadeh M. Dragon2000: Standard-cell Placement Tool for Large Industry Circuits // ICCAD 2000. pp. 260-263.
- [26] Caldwell A.E., Kahng A.B. and Markov I.L. Can Recursive Bisection Alone Produce Routable Placements? // DAC 2000. pp. 477-482.
- [27] Chan T., Cong J., Kong T., Shinnerl J. и Sze K. An enhanced multilevel algorithm for circuit placement // In Proceedings of the IEEE International Conference on Computer Aided Design (San Jose, Calif.). IEEE Computer Society Press, Los Alamitos, Calif. 2003.
- [28] Yang X., Choi B.-K. and Sarrafzadeh M. Routability-driven white space allocation for fixed-die standard-cell placement

Placement of VLSI Elements Based on Swarm Intelligence Models

B.K. Lebedev, O.B. Lebedev, A.A. Zhiglaty

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

Abstract — The paper presents the architecture of a multi-agent system for deploying VLSI elements based on the integration of swarm intelligence models. The concept of placing elements by the pair P ($A1, A2$) of ordered lists of the same set of elements is used. The transition from pair P to placement is carried out in two stages. First, a constraint graph is formed on the basis of the pair P , and then a plan or arrangement of elements is constructed from the constraint graph. New structures are proposed for representing the solution to the problem of placing elements of super-large integrated circuits in the form of chromosomes. A modified paradigm of a particle swarm is presented, which differs from the canonical one, with the possibility of using positions with integer parameter values in the affine space. The mechanisms of particle movement in the affine space are considered to reduce the weight of affine bonds using the developed operator directed mutation. The directed mutation operators are described, the essence of which is to change the integer values of the genes in the chromosome. A modified structure of the bee algorithm is proposed. The key operation of the algorithm is the study of promising positions lying in the vicinity of the base positions. The bee paradigm is a two-tier search strategy. At the first level, many base positions are generated in the search space. At the second level, in the search space, the neighborhoods of base positions are investigated. A method of forming positions in the vicinity of base positions is proposed. The best positions found in each neighborhood of the base positions at iteration t are part of the set of base positions used at the iteration $(t + 1)$. The architecture of a hybrid algorithm based on the integration of bee swarm and bee colony methods was developed. Hybridization is as follows. Initially, a swarm of particles is formed. After moving a swarm of particles to new positions, these positions are considered as the basic positions found by the swarm of scout bees. Further, in accordance with the mechanisms of the bee colony, a new set of basic positions is formed, which, in turn, at the next iteration is considered as a swarm of particles. The tests have shown that when integrating the behavior patterns of a swarm of bees and a swarm of particles, the results of the new hybrid algorithm are 11 to 18% better than each algorithm individually.

Keywords - VLSI, placement, swarm intelligence, bee algorithm, swarm of particles, hybridization, multi-agent system, affine search space, directional mutation operator, neighborhood of base positions, bionic search.

REFERENCES

- [1] Norenkov I.P. Osnovy avtomatizirovannogo proyektirovaniya. (Fundamentals of Computer Aided Design). M.: Izd-vo MGTU imeni N.E.Baumana, 2006. 248 p. (In Russian).
- [2] Alpert C.J., Mehta D.P., Sapatnekar S.S., Handbook of Algorithms for Physical Design Automation. Boston, MA: Auerbach, 2009. 245 p.
- [3] Kazennov G.G. Osnovy proyektirovaniya integral'nykh skhem i sistem (Fundamentals of design of integrated circuits and systems). M.: Binom, Laboratoriya znaniy, 2005, 295 p. (In Russian).
- [4] Wang X. Hybrid nature-inspired computation method for optimization // Doctoral Dissertation. Helsinki University of Technology, TKK Dissertations, Espoo 2009, 161 p.
- [5] 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. (In Russian).
- [6] Poli R. Analysis of the publications on the applications of particle swarm optimisation. Journal of Artificial Evolution and Applications, Article ID 685175, 2008.
- [7] Clerc M. Particle Swarm Optimization. ISTE, London, UK, 2006. 198 p.
- [8] Kennedy J., Eberhart R.C. Particle swarm optimization // In Proceedings of IEEE International Conference on Neural Networks, 1995. pp. 1942-1948.
- [9] Lebedev B.K., Lebedev O.B., Lebedeva E.O., Nagabedian A.A. Gibrinny royevoy algoritm global'noy optimizatsii v affinnom prostranstve poiska. (Hybrid swarm global optimization algorithm in the affine search space). Elektronnyy zhurnal Programmnyye produkty, sistemy i algoritmy. Tver': Izd-vo «Tsentrprogrammssystem», 2019. №1. (In Russian).
- [10] Lučić P., Teodorović D. Computing with Bees: Attacking Complex Transportation Engineering Problems // International Journal on Artificial Intelligence Tools. №12, 2003. pp. 375-394.
- [11] Quijano N., Passino K.M. Honey Bee Social Foraging Algorithms for Resource Allocation: Theory and Application. Columbus: Publishing house of the Ohio State University, 2007. 139 p.
- [12] Lebedev V.B. Metod pcheliny kolonii v kombinatornykh zadachakh na grafakh (The bee colony method in combinatorial problems on graphs) // Trinadtsataya natsional'naya konferentsiya po iskusstvennomu intelektu s mezhdunarodnym uchastiyem KII-2012. Trudy konferentsii. T.2. M.: Fizmatlit, 2012. pp. 414-422 (in Russian).
- [13] Lebedev B.K., Lebedev V.B. Razmeshcheniye na osnove metoda pcheliny kolonii // (Accommodation based on the method of bee colony) // Izvestiya YUFU. Izd-vo TTI YUFU, № 12, 2010. pp. 12-18 (in Russian).
- [14] Lebedev B.K., Lebedev V.B., Lebedev O.B. Metody, modeli i algoritmy razmeshcheniya (Methods, models and placement algorithms). Rostov-na-Donu: Izd-vo YUFU, 2015. 181 p. (In Russian).
- [15] Xu J., Guo P.-N., Cheng C.-K. Sequence-Pair Approach for Rectilinear Module Placement // IEEE Trans. Computer-

- Aided Design of Integrated Circuits and Systems, № 4. 1999. pp. 204-218.
- [16] Lebedev B.K., Lebedev V.B. Pokrytiye metodom roya chastits (Particle swarm coating) // Materialy VI-y Mezhdunarodnoy nauchno-prakticheskoy konferentsii "Integrirrovannyye modeli i myagkiye vychisleniya v iskusstvennom intellekte". M.: Fizmatlit, 2011. pp. 611-619 (in Russian).
- [17] Lebedev B.K., Lebedev O.B., Lebedeva E.M. Raspredeleniye resursov na osnove gibridnykh modeley royeвого intellekta (Resource allocation based on hybrid swarm intelligence models) // Nauchno-tekhnicheskii vestnik informatsionnykh tekhnologiy, mekhaniki i optiki Scientific and Technical Journal of Information Technologies, Mechanics, and Optics. Izd-vo S-Peterburg, T. 17, № 6, 2017. pp. 1063-1073 (in Russian).
- [18] Lebedev B.K., Lebedev V.B. Trassirovka na osnove metoda roya chastits (Tracing based on the particle swarm method) // Dvenadtsataya natsional'naya konferentsiya po iskusstvennomu intellektu s mezhdunarodnym uchastiyem KII-2010. Trudy konferentsii T. 2. M.: Fizmatlit, 2010. pp. 414-422 (in Russian).
- [19] Lebedev B.K., Lebedev O.B., Lebedev V.B. Gibridizatsiya royeвого intellekta i geneticheskoy evolyutsii na primere razmeshcheniya (Hybridization of swarm intelligence and genetic evolution on the example of placement) // Elektronnyy zhurnal Programmnyye produkty, sistemy i algoritmy. Tver: Izd-vo «Tsentrprogrammssystem», 2017, №4. (In Russian).
- [20] Lebedev B.K., Lebedev V.B. Planirovaniye na osnove royeвого intellekta i geneticheskoy evolyutsii (Planning based on swarm intelligence and genetic evolution) // Izvestiya YUFU. Izd-vo TTI YUFU, №2, 2009. pp. 25-33 (in Russian).
- [21] Lebedev B.K., Lebedev O.B. Lebedeva E.O. Royevoy algoritm planirovaniya raboty mnogoprotsessornykh vychislitel'nykh sistem (A swarm algorithm for planning the operation of multiprocessor computing systems) // Elektronnyy nauchnyy zhurnal «Inzhenernyy vestnik Dona, 2017. № 3. (In Russian).
- [22] Cong J., Romesis M., Xie M. Optimality, Scalability and Stability Study of Partitioning and Placement Algorithms // Proc. of the International Symposium on Physical Design. Monterey, CA, 2003. pp. 88-94.
- [23] Cong J., Romesis M. и Xie M. *UCLA Optimality Study Project*. <http://cadlab.cs.ucla.edu/pubbench>. 2004.
- [24] MCNC. Electronic and Information Technologies (Online). Available: www.mcnc.org. (28.03.2020).
- [25] Wang M., Yang X. and Sarrafzadeh M. Dragon2000: Standard-cell Placement Tool for Large Industry Circuits // ICCAD 2000. pp. 260-263.
- [26] Caldwell A.E., Kahng A.B. and Markov I.L. Can Recursive Bisection Alone Produce Routable Placements? // DAC 2000. pp. 477-482.
- [27] Chan T., Cong J., Kong T., Shinnerl J. и Sze K. An enhanced multilevel algorithm for circuit placement // In Proceedings of the IEEE International Conference on Computer Aided Design (San Jose, Calif.). IEEE Computer Society Press, Los Alamitos, Calif. 2003.
- [28] Yang X., Choi B.-K. and Sarrafzadeh M. Routability-driven white space allocation for fixed-die standard-cell placement // IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, 22(4): 2003. pp. 410-419.
- [29] CADENCE DESIGN SYSTEMS, INC. *QPlace version 5.1.55, compiled on 10/25/1999*. Envisia ultra-placer reference.