

Принятие решений в задаче размещения компонентов СБИС на основе модели поведения стаи волков

Э. В. Кулиев¹, В. В. Курейчик, И. О. Курситыс

Южный федеральный университет

¹ekuliev@sfedu.ru

Аннотация. Статья посвящена решению ключевой задачи этапа автоматизированного проектирования – размещения компонентов сверхбольших интегральных схем. В настоящее время все чаще исследуются возможности разработки и применения алгоритмов, основанных на адаптивном поведении биологических систем, для качественного решения в задачах автоматизированного проектирования. Основной проблемой является конфликт между требованиями принятия эффективных решений в реальном времени и сложностью САПР. Одним из подходов к решению возникших проблем, могут служить новые интеллектуальные технологии на стыке бионики, информатики и автоматизации проектирования. В связи с вышесказанным, авторами рассмотрен биоинспирированный алгоритм, основанный на поведении стаи волков в природе. Представлена постановка задачи размещения компонентов СБИС. Описана модель поведения волков в живой природе, а также основные принципы её работы адаптированы под решение задачи размещения. Проведены экспериментальные исследования. Основная цель проведенных экспериментальных исследований заключалась в оценке возможностей применения методов инспирированных природными системами к решению конструкторских задач.

Ключевые слова: автоматизация проектирования; стая волков; размещение; принятие решений; сверхбольшие интегральные схемы

I. ВВЕДЕНИЕ

В настоящее время для проектирования сложных систем необходимы большие трудозатраты, состоящие как из человеческих ресурсов, так и времени. В связи с этим необходимо применять различные средства автоматизированного проектирования, где электронно-вычислительная машина является как самим объектом, так и инструментом проектирования [1].

Автоматизация проектирования зависит от совершенства вычислительной техники и её элементной базы.

Распространенной задачей связанной с автоматизированным проектированием, является задача планирования сверхбольших интегральных схем (СБИС).

Работа выполнена при финансовой поддержке гранта Президента Российской Федерации для государственной поддержки молодых российских ученых - кандидатов наук № МК-1480.2018.9

Основная задача планирования заключается в размещении на поле кристалла блоков, полученных на предыдущем этапе проектирования, не имеющих фиксированных размеров и имеющих заданную площадь. Цель задачи размещения заключается в определении места на поле кристалла для каждого блока и элемента [1].

Основными характеристиками при размещении являются временные задержки, длина связи, площадь и размеры кристалла, распределение блоков по кристаллу. Целью решения задачи размещения является получения наилучших условий для последующих этапов конструкторского проектирования. Для этого вводятся критерии и оценки, оптимизация которых приводит к наилучшим условиям размещения.

II. ПОСТАНОВКА ЗАДАЧИ

Постановка задачи размещения компонентов сверхбольших интегральных схем может быть сформулирована так: задано коммутационное поле, на котором могут быть размещены множество элементов с расположенными на них контактами (выводами). Заданы элементарные цепи, связывающие контакты элементов. Необходимо разместить элементы на коммутационном поле с оптимизацией заданных критериев качества. Основная цель задачи размещения заключается в минимизации общей площади кристалла, минимизации общей суммарной длины связей ($L(G) \rightarrow \min$) [3] и создания оптимальных условия для дальнейших этапов конструкторского проектирования [1, 2].

В общем виде задача размещения может быть сформулирована так: на монтажном пространстве заданную область разбивают на множество позиций, количество позиций в которой должно быть больше равно числу размещаемых элементов. Каждый из элементов может занимать одно посадочное место, расстояние между которыми описывается матрицей расстояний.

Имеющееся множество элементов, связанных между собой множеством электрических цепей необходимо таким образом отобразить на множестве P , чтобы обеспечивался экстремум целевой функции качества размещения. Задача размещения сводится к отображению заданного графа схемы в решетку таким образом, чтобы вершины множества размещались в узлах решетки и, например,

суммарная длина связей была наименьшей для возможных способов отождествления вершин графа и узлов решетки. При этом на исходную плоскость наносится, декартова система координат с осями s и t [3, 4]. В каждую ячейку плоскости может быть размещен элемент коммутационной схемы.

III. АЛГОРИТМ СТАИ ВОЛКОВ

В настоящее время становится актуальным применение в системах автоматизированного проектирования методов, основанных на живой природе [5]. Такие методы принято называть биоинспирированными. Биоинспирированные методы основываются на описании природных эволюционных процессов, включающих в себя также генетические алгоритмы. Применяя методы биоинспирированного поиска возможно получать наборы квазиоптимальных решений за полиномиальное время.

Одним из алгоритмов биоинспирированного поиска является алгоритм стаи волков. Впервые данный алгоритм был предложен в 2014 г. [6–8]. Алгоритм основывается на поведении волков в природе, включающий в себя механизмы охоты и иерархии волков.

Иерархия волков включает в себя альфа, бета, дельта и омега волков. Основные этапы охоты у волков заключаются в поиске добычи, окружении и нападении на добычу.

Волки находятся на вершине иерархии хищников, что означает, что они находятся на вершине пищевой цепи.

Волки живут стаями, размер которых составляет 5–12 особей. Следует отметить, что у особей строгая иерархия доминирования. Основными особями – лидерами, являются самцы и самки – Альфа особи. Альфа особь принимает решения о времени подъема стаи, начала охоты, ночлеге и т.д. [6, 9, 10]

Тем не менее, какое-то демократическое поведение также наблюдается, в котором Альфа волки следуют за другими волками в стае. На собрании волков Альфа волка признаёт вся стая путём удерживания хвостов внизу. Альфа-волкам допускается спариваться только в стае. Интересно, что Альфа-это не обязательно самый сильный член стаи, но лучший в плане управления стаей. Это показывает, что организация и дисциплина в стае гораздо важнее, чем его сила.

Следующими в иерархической цепочки стаи стоят бета-волки. Бета-волки находятся в полном подчинении у Альфа и помогают им в любой деятельности стаи. Бета-волк является главным кандидатом на то, чтобы стать альфа волком в случае смерти одного из альфа волков. Бета-волк может управлять только волками, ниже его уровнем [8, 11, 12].

На самом низшем иерархическом уровне находятся омега-волки. Эти волки всегда подчиняются доминирующим волкам. С одной стороны, могло показаться, что данный тип волков не нужен в стае, однако при их отсутствии всегда наблюдались столкновения

между волками. Таким образом, омега – волки позволяют сохранить иерархию волков в стае.

Помимо перечисленных типов волков, выделяют дельта-волков. Дельта-волки стоят в иерархии волков на одну ступень выше, чем Омега-волки. К данному типу волков относятся дозорные волки, разведчики, старейшины, смотрители и охотники.

Волк разведчик отвечает за окрестностью и в случае опасности, предупреждает стаю. Дозорные волки являются охранниками стаи. В волчьей стае старейшинами являются самые опытные волки (альфа и бета волки в прошлом). Волки охотники помогают альфа и бета волкам в охоте.

Для математического моделирования иерархии волков при проектировании, следует считать наиболее приспособленным решением Альфа (α). Тогда, второе и третье решения будут называться соответственно бета (β) и Дельта (δ). Остальные решения принимают значения Омега (ω). В стае волков за оптимизацию решения отвечают α , β и δ .

Для решения задачи размещения компонентов СБИС необходимо адаптировать основные принципы работы алгоритма стаи волков, т.к. изначально, предполагалось, что данная модель предназначена для решения векторных задач.

Для адаптации необходимо начальную популяцию решений представить в виде хромосом. Далее, осуществить их кодирование / декодирование в соответствии со следующим принципом: номер ген в хромосомах – посадочные места, а значения в этих генах – номера элементов, которые будут на этих посадочных местах [13, 14].

Для математического моделирования поведения волков при окружении добычи, введем следующие уравнения:

$$D = |\vec{C} \times \vec{X}_p(t) - \vec{X}(t)|;$$

$$\vec{X}(t + 1) = \vec{X}_p(t) - \vec{A} \times \vec{D},$$

где t обозначает текущую итерацию, \vec{A} и \vec{C} являются векторы-коэффициенты, \vec{X}_p – позиция вектора «добыча», и \vec{X} показывает положение вектора «серый волк».

Векторы \vec{A} и \vec{C} рассчитываются следующим образом [12]:

$$\vec{A} = 2\vec{a} \times \vec{r}_1 - \vec{a}; \quad (1)$$

$$\vec{C} = 2 \times \vec{r}_2, \quad (2)$$

где компонент \vec{a} линейно снижается с 2 до 0 в течение итераций и r_1, r_2 являются случайными векторами в $[0,1]$. В нашей задаче в роли параметра a выступает одноточечная мутация между всеми особями, уменьшение которой равномерно распределено на все итерации. Случайные векторы r_1, r_2 разрешают волкам добраться в любую позицию между двух конкретных точек, при этом пространство может быть с N -измерениями, что означает, что серые волки будут перемещаться в гиперкубе вокруг наилучшего решения (рис. 1).

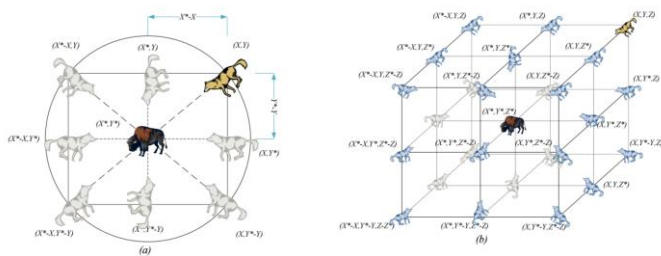


Рис. 1. Обновление позиции волков

С учетом представленных выше уравнений 1 и 2, серый волк в положение (X, Y) может обновить свою позицию в зависимости от положения добычи (X^*, Y^*) . Различные места вокруг лучшего агента могут быть достигнуты по отношению к текущей позиции, регулируя значение векторов \vec{A} и \vec{C} . Например, (X^*-X, Y^*-Y) может быть достигнуто путем установки $\vec{A} = (1, 0)$ и $\vec{C} = (1, 1)$.

Стая волков имеет способность распознавания местонахождения добычи и окружения ее. Охотниками в стае волков являются альфа. Остальные волки иногда также могут принять участие в охоте.

Для математического моделирования охотничьего поведения волка, предположим, что альфа, бета и дельта-волки имеют четкое представление о местонахождении добычи. Тем самым, ЛПР имеет возможность сохранения трех лучших решений, и обязывает других волков, в том числе и Омега-волков обновить свои позиции в зависимости от позиции лучших поисковых агентов [12].

Окончание охоты у волков заключается в нападении на добычу до тех пор, пока она не перестанет двигаться. Для того, чтобы математически смоделировать атаку добычи, мы уменьшаем значение \vec{a} . Важно, что диапазон колебания \vec{A} также снизился на \vec{a} . Другими словами, \vec{A} это случайное значение в интервале $[-2a, 2a]$, где уменьшается от 2 до 0 в течение итераций. Когда случайные значения \vec{A} находятся в интервале $[-1, 1]$, следующая позиция поискового агента может находиться в любом положении между своей текущей позицией и позицией жертвы.

Поиск добычи волками происходит в соответствии с иерархическим положением альфа, бета и дельта волков. Суть поиска заключается в расхождении друг от друга волков с целью поиска добычи и в схождении для атаки добычи. Для математического представления расхождения, применим \vec{A} с случайные значения больше 1 или меньше -1 и принуждаем поискового агента расходиться с целью поиска добычи. Разведка позволяет стае волков осуществлять глобальный поиск. $|A| > 1$ заставляет волков расходиться, надеясь найти след добычи. Еще одним компонентом алгоритма стаи волков, способствующей разведке, является \vec{C} , который содержит случайные значения в диапазоне $[0, 2]$. Этот компонент обеспечивает случайными весами добычу в целях стохастического усиления ($C > 1$) или ослабления ($C < 1$) влияния добычи в определении расстояния [8–10].

Поиск осуществляется путем создания случайных решений (популяции волков). С каждой итерацией иерархия волков оценивает вероятностную позицию добычи. Каждая особь обновляет свою позицию до добычи. Для сужения или увеличения области разведки вводится параметр «а», который путем уменьшения значения от 2 до 0, сужает область поиска допустимых решений (добычи). При $|A|=0$ происходит атака стаи волков на особь. При $|A| > 1$ особи расходятся с добычей, при $|A| < 1$ сходятся в направлении добычи. Алгоритм завершается достижением конечного критерия.

IV. ЭКСПЕРИМЕНТЫ

Были проведены экспериментальные исследования на примере различных графов. Одна из основных задач, решение которой находится при использовании биоинспирированных алгоритмов – это поиск подходящих характеристик, при которых алгоритм демонстрирует более действенные заключения.

Для определения временной сложности алгоритма (ВСА) был проведен ряд опытов, в процессе которых были получены результирующие итоги среднего значения времени работы алгоритма. Они показаны на рис. 2.

График зависимости времени работы от числа составляющих элементов подтверждает выдвинутые предположения о временной сложности разрабатываемого алгоритма, основывающийся на поведении стаи волков.

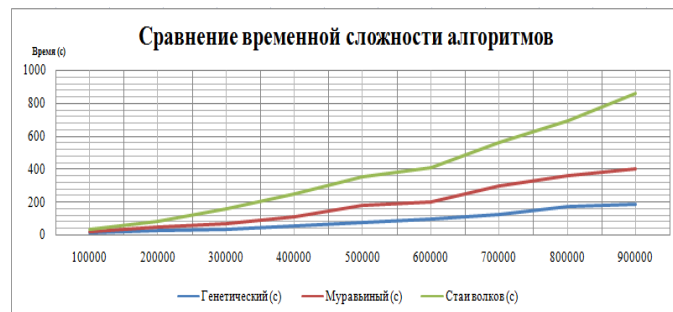


Рис. 2. График зависимости времени работы от числа элементов

С целью установления эффективности разработанного алгоритма проводились исследования качества решений на нескольких испытательных вариантах. Под эффективностью алгоритма понимается качество решений, получаемых с его помощью. На рис. 3 приведена гистограмма сравнения качества решений.

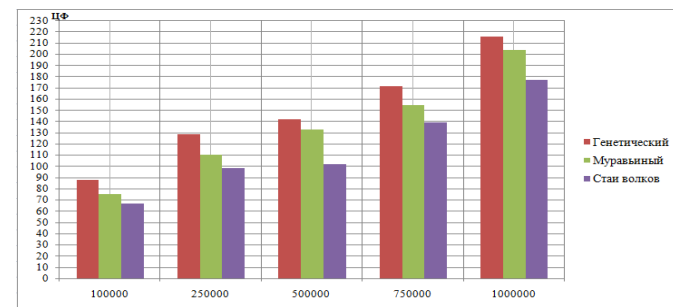


Рис. 3. Гистограмма сравнения качества решений

Следуя из итогов экспериментальных исследований можно заметить, что наиболее эффективным является разработанный алгоритм, базирующийся на поведении стаи волков. Он в среднем на 12% эффективнее генетического алгоритма [15] и на 8% муравьиного алгоритма [16].

V. ЗАКЛЮЧЕНИЕ

В статье рассмотрено решение задачи размещения компонентов сверхбольших интегральных схем на основе алгоритма поведения стаи волков. Алгоритм поведения стаи волков в полной мере можно считать достаточно эффективным, но, он не является универсальным алгоритмом для решения абсолютно всех оптимизационных задач.

Рассмотрена иерархия волков, включающая в себя альфа, бета, дельта и омега волков. Основные этапы охоты которых заключаются в поиске добычи, окружении и нападении. Следует отметить, что у особей строгая иерархия доминирования. Основными особями – лидерами, являются самцы и самки – Альфа особи. Альфа особь принимает решения о времени подъема стаи, начала охоты, ночлеге и т.д.

Для моделирования охотничьего поведения волка, было представлено, что альфа, бета и дельта-волки имеют четкое представление о местонахождении добычи. Тем самым, ЛПР имеет возможность сохранения трех лучших решений, и обязывает других волков, в том числе и Омега-волков обновить свои позиции в зависимости от позиции лучших поисковых агентов. Поиск добычи волками происходит в соответствии с иерархическим положением альфа, бета и дельта волков. Суть поиска заключается в расхождении друг от друга волков с целью поиска добычи и схождения, для атаки добычи.

Для математического представления иерархии волков при проектировании, авторы считали, что наиболее приспособленным решением является Альфа (α). Тогда, вторым и третьим решениями соответственно бета (β) и Дельта (δ). Остальные решения принимали значения Омега (ω). В стае волков за оптимизацию решения отвечают α , β и δ .

Для решения задачи размещения компонентов СБИС были адаптированы основные принципы работы алгоритма стаи волков. Для адаптации начальная популяция решений была представлена в виде хромосом. Далее, осуществлено их кодирование / декодирование в соответствии со следующим принципом: номер ген в хромосомах – посадочные места, а значения в этих генах – номера элементов, которые будут на этих посадочных местах.

Разработан программно-алгоритмический комплекс на языке программирования C++. На основе разработанного программно-алгоритмического комплекса были проведены

экспериментальные исследования временной сложности алгоритма и сравнения качества решений на нескольких испытательных вариантах. Следуя из итогов экспериментальных исследований можно заметить, что наиболее эффективным является разработанный алгоритм, базирующийся на поведении стаи волков. Он в среднем на 12% эффективнее генетического алгоритма и на 8% муравьиного алгоритма.

СПИСОК ЛИТЕРАТУРЫ

- [1] Курейчик В.В., Запорожец Д.Ю. Современные проблемы при размещении элементов СБИС // Известия Южного федерального университета. Технические науки. 2011. Т. 120. № 7. С. 68-73.
- [2] Норенков И.П. Эволюционные методы в задачах выбора проектных решений / И.П. Норенков, Н.М. Арутюнян // Электронное научно-техническое издание
- [3] Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой: учебное пособие / А.П. Карпенко. Москва: Издательство МГТУ им. Н.Э. Баумана, 2014. 448 с.
- [4] Кулиев Э.В., Лежебоков А.А. О гибридном алгоритме размещения компонентов СБИС // Известия Южного федерального университета. Технические науки. 2012. Т. 136. № 11 (136). С. 188-192.
- [5] Yurevich Zaporozhets, D., Victorovna Zaruba, D., Kureichik, V.V. Hybrid bionic algorithms for solving problems of parametric optimization// (2013) World Applied Sciences Journal, 23 (8), pp. 1032-1036.
- [6] Rezaei H., Bozorg-Haddad O., Chu X. Grey wolf optimization (GWO) algorithm (2018) Studies in Computational Intelligence, 720, pp. 81-91.
- [7] Zhang X., Kang Q., Cheng, J., Wang X. A novel hybrid algorithm based on Biogeography-Based Optimization and Grey Wolf Optimizer (2018) Applied Soft Computing Journal, 67, pp. 197-214.
- [8] Long W., Jiao J., Liang X., Tang M. Inspired grey wolf optimizer for solving large-scale function optimization problems (2018) Applied Mathematical Modelling, 60, pp. 112-126.
- [9] El Gayyar M., Emary E., Sweilam N.H., Abdelazeem M. A Hybrid Grey Wolf-Bat Algorithm for Global Optimization (2018) Advances in Intelligent Systems and Computing, 723, pp. 3-12.
- [10] Singh, N., Singh, S.B. A novel hybrid GWO-SCA approach for optimization problems (2017) Engineering Science and Technology, an International Journal, 20 (6), pp. 1586-1601.
- [11] Кулиев Э.В., Щеглов С.Н., Пантелюк Е.А., Кулиева Н.В. Адаптивный алгоритм стаи серых волков для решения задач проектирования // Известия ЮФУ. Технические науки. 2017. № 7 (192). С. 28-38.
- [12] Gupta S., Deep K. A novel Random Walk Grey Wolf Optimizer (2018) Swarm and Evolutionary Computation. Article in Press.
- [13] Kureichik V.V., Zaruba D.V. The bioinspired algorithm of electronic computing equipment schemes elements placement // (2015) Advances in Intelligent Systems and Computing, 347, pp. 51-58.
- [14] Zaporozhets D.U., Zaruba D.V., Kureichik V.V. Representation of solutions in genetic VLSI placement algorithms // (2014) Proceedings of IEEE East-West Design and Test Symposium, EWDTS 2014.
- [15] Кулиев Э.В., Лежебоков А.А. Исследование характеристик гибридного алгоритма размещения // Известия ЮФУ. Технические науки. 2013. № 3. с. 255-261.
- [16] Zaruba D., Zaporozhets D., Kureichik V. VLSI placement problem based on ant colony optimization algorithm // (2016) Advances in Intelligent Systems and Computing, 464, pp. 127-133.