

## Раздел I. Модели и методы обработки информации

УДК 519.712.2

DOI 10.18522/2311-3103-2022-5-6-17

**М.Д. Ясир, Л.А. Гладков, Н.В. Гладкова**

### РЕШЕНИЕ ЗАДАЧИ РАЗМЕЩЕНИЯ ЭЛЕМЕНТОВ ЭВА НА ОСНОВЕ ИНТЕГРАЦИИ МЕТОДОВ ЭВОЛЮЦИОННОГО ПОИСКА И НЕЧЕТКОГО УПРАВЛЕНИЯ

*Рассматривается задача размещения элементов электронно-вычислительной аппаратуры (ЭВА). Проведен анализ современного состояния исследований по данной теме, отмечена актуальность рассматриваемой задачи. Подчеркнута важность разработки новых эффективных методов решения задач конструкторского проектирования. Отмечена перспективность разработки и использования гибридных подходов и моделей для решения сложных слабоформализованных задач проектирования и оптимизации. Приведена постановка задачи размещения элементов схем ЭВА. Отмечена важность качественного решения задачи размещения с точки зрения успешного выполнения последующих этапов проектирования. Проведен анализ различных подходов и алгоритмов решения задачи размещения. Приведены варианты выбора различных критериев оценки качества размещения. Предложен комплексный аддитивный критерий для оценки качества размещения. Приведена целевая функция и ограничения рассматриваемой задачи размещения как задачи оптимизации. Предложен гибридный подход к решению задачи размещения. Для повышения эффективности и сокращения времени работы алгоритма, предложена модель параллельного многопопуляционного генетического алгоритма. Для синхронизации эволюционных процессов в многопопуляционном генетическом алгоритме разработан модифицированный оператор миграции. Выполнен анализ эффективности работы предложенного оператора миграции и сформулированы рекомендации по его использованию. С целью увеличения быстродействия алгоритма решения задачи размещения предложена модель организации параллельных эволюционных вычислений за счёт использования многопоточности на локальном уровне. Описаны принципы работы модуля нечеткого управления. Описана процедура логического вывода с использованием базы правил. Предложена структура многослойной нейронной сети, реализующей функцию Гаусса. Предложена модель нечеткого логического контроллера для динамического изменения значений управляющих параметров генетического алгоритма. Определены управляющие параметры нечеткого логического контроллера. Предлагаемый гибридный алгоритм реализован в виде прикладной программы. Были проведены серии вычислительных экспериментов для определения эффективности разработанного алгоритма и выбора оптимальных значений управляющих параметров.*

*Автоматизация проектирования; задача размещения элементов ЭВА; биоинспирированные алгоритмы; гибридные методы; параллельные генетические алгоритмы; нечеткое управление.*

**M.J. Yaser, L.A. Gladkov, N.V. Gladkova**

### SOLUTION OF THE ELEMENTS LOCATION PROBLEM IN DIGITAL COMPUTER EQUIPMENT ON THE BASIS OF INTEGRATION OF EVOLUTIONARY SEARCH AND FUZZY CONTROL METHODS

*The problem of elements placement in digital computing equipment are consider in the article. The analysis of the current state of research on this topic is carried out, the relevance of the problem under consideration is noted. The importance of developing new effective methods for solving engineering design problems is emphasized. The prospects of developing and using hybrid*

*approaches and models for solving complex semi-formalized design and optimization problems are noted. The statement of the problem of placement of circuit elements of digital computing equipment is given. The importance of a qualitative solution of the placement problem from the point of view of the successful implementation of the subsequent stages of design is noted. The analysis of various approaches and algorithms for solving the placement problem is carried out. Options for choosing various criteria for assessing the quality of placement are given. A complex additive criterion for assessing the quality of placement is proposed. The objective function and limitations of the considered placement problem as an optimization problem are given. A hybrid approach to solving the placement problem is proposed. To increase the efficiency and reduce the running time of the algorithm, a model of a parallel multipopulation genetic algorithm is proposed. To synchronize evolutionary processes in a multipopulation genetic algorithm, a modified migration operator has been developed. An analysis of the efficiency of the proposed migration operator has been carried out and recommendations for its use have been formulated. In order to increase the speed of the algorithm for solving the placement problem, a model for organizing parallel evolutionary computations through the use of multithreading at the local level is proposed. The principles of operation of the fuzzy control module are described. The procedure of logical inference using the rule base is described. The structure of a multilayer neural network that implements the Gaussian function is proposed. A model of a fuzzy logic controller for dynamically changing the values of control parameters of a genetic algorithm is proposed. The control parameters of the fuzzy logic controller are determined. The proposed hybrid algorithm is implemented as an application program. A series of computational experiments were carried out to determine the efficiency of the developed algorithm and to select the optimal values of the control parameters.*

*Automation of digital computer design; element placement problem; bioinspired algorithms; hybrid methods; parallel genetic algorithms; fuzzy control.*

**Введение.** В настоящее время процесс проектирования невозможно представить без применения вычислительной техники. Сложность проектируемых устройств, число элементов и связей между ними непрерывно растет. На данный момент идет об устройствах содержащих порядка 10<sup>4</sup> элементов. Решать возникающие задачи такого уровня сложности путем полного перебора вариантов пока невозможно [1]. В некоторых случаях отсутствуют детерминированные вычислительные методы, что не позволяет решить задачу выполнения проектирования в полностью автоматическом режиме. Поэтому актуальной остается задача разработки эвристических методов оптимизации, позволяющих получать приемлемые по качеству решения за полиномиальное время.

Совершенствование технологий изготовления привело к пересмотру некоторых принципов, на которых основан процесс физического проектирования [2]. Это в свою очередь привело к пересмотру актуальных критериев оценки качества размещения. Использувавшиеся ранее критерии, построенные на модели оценки временных задержек пропорционально суммарной длине соединений стали неактуальными, изменилась концепция процесса физического проектирования вычислительных устройств.

Одним из основных трендов современной науки является гибридизация, т.е. создание новых систем, моделей и методов, подразумевающих интеграцию научных подходов различной физической природы. Хорошие результаты позволяет получить использование биоинспирированных алгоритмов в качестве инструмента поиска оптимальных решений. В настоящее время предложено большое число разновидностей биоинспирированных алгоритмов для решения различных задач оптимизации [3, 4]. Особенно эффективно использование таких систем для решения сложных слабо формализованных задач, в том числе задач проектирования и оптимизации [5].

В данной статье предлагается гибридный подход к решению задачи размещения элементов ЭВА на основе интеграции различных подходов, таких как, эволюционные алгоритмы поиска, нейронные сети, нечеткие методы управления параметрами алгоритма и механизмы распараллеливания вычислений [6].

**Постановка задачи.** После решения задачи разбиения электрических цепей можно выполнить оценочный расчет площади, занимаемой каждым блоком и определить необходимое число контактов (выводов). Перед началом трассировки, проектировщику нужно расположить блоки на поверхности кристалла и соединить их выводы между собой в соответствии с заданным списком соединений. Расположение блоков осуществляется на этапе размещения, а соединение осуществляется на этапе трассировки [5].

На этапе размещения элементы должны быть расположены на рабочей поверхности так, чтобы любые два блока не имели взаимного перекрытия, и, при этом, оставалось достаточно места для проведения соединений. Элементы должны быть размещены таким образом, чтобы минимизировать занимаемую площадь.

Для простоты предполагается, что все блоки имеют прямоугольную форму. Как правило, для определения формы блока используют соотношение сторон, т.е. отношение высоты блока к его ширине. Для определения допустимых размеров задают верхнюю и нижнюю границы соотношения сторон.

Алгоритмы размещения на уровне платы нацелены на минимизацию числа слоев при трассировке при соблюдении требований по производительности. Длина критических связей в высокопроизводительных цепях очень важна, следовательно, элементы таких цепей должны размещаться максимально близко друг к другу. Другой важнейшей проблемой является соблюдение температурного режима. Теплоотвод в схеме должен быть равномерным, значит блоки, выделяющие наибольшее количество тепла, не должны располагаться близко друг к другу. Эти и другие требования могут противоречить друг другу.

Пусть задано множество элементов  $E_1, E_2, \dots, E_n$ , которые должны быть размещены на рабочей поверхности. Известны габаритные размеры (высота  $h_i$  и ширина  $w_i$ ) каждого элемента  $E_i$ ,  $1 \leq i \leq n$ . Также задано множество цепей  $N = \{N_1, N_2, N_3, \dots, N_m\}$ , множество свободных позиций  $R = \{R_1, R_2, \dots, R_n\}$  и множество прямоугольных областей  $Q = \{Q_1, Q_2, \dots, Q_k\}$ , в которых будет выполняться трассировка. Оценочную длину цепи  $N_i$ ,  $1 \leq i \leq m$  обозначим как  $L_i$ .

Тогда задача размещения состоит в том, чтобы найти для каждого элемента множества  $E$  подходящую по размеру позицию на рабочей поверхности так, чтобы соблюдались следующие условия:

1. Любой элемент множества  $E_i$  можно поместить в одну из свободных позиций множества  $R_i$ .
2. Никакие две позиции не перекрываются, то есть  $R_i \cap R_j = \emptyset$ ,  $1 \leq i, j \leq n$ .
3. Размещение является трассируемым, т.е. площади  $Q_i$ ,  $1 \leq j \leq k$ , достаточно для трассировки всех цепей.
4. Общая площадь прямоугольника, ограничивающего области  $R$  и  $Q$ , является минимальной.

5. Общая длина соединений  $\sum_{i=1}^m L_i$  минимизируется. При наличии в схеме вы-

сокопроизводительных цепей длина самой длинной цепи  $\max\{L_i \mid i = 1, \dots, m\}$  минимизируется.

В качестве критерия оптимизации используется аддитивная функция:

$$F = \min_{z_j \in Z} (k \cdot O(L(z_j)) + P(S_{\text{общ}}(z_j))),$$

где  $z_j$  – текущий вариант размещения;  $k$  – весовой коэффициент;  $S_{\text{общ}}$  – суммарная площадь перекрытия элементов;  $O(L(z_j))$  – оценка длины соединений;  $P(S_{\text{общ}}(z_j))$  – штраф за перекрытие площадей.

**Описание гибридного алгоритма.** Для решения задачи размещения предложен параллельный многопопуляционный генетический алгоритм. Он предполагает параллельное выполнение эволюционных процессов на нескольких популяциях. Для обмена особями используется островная модель параллельного генетического алгоритма [7].

В островной модели синхронизация параллельных процессов осуществляется в точках миграции. Миграция осуществляется при наступлении определённых событий, которые могут произойти в любой популяции. При наступлении события в одной из популяций, выполнение эволюции приостанавливается. Миграция осуществляется между двумя соседними популяциями [8].

С помощью оператора миграции выполняется обмен решениями между популяциями. Хромосомы для миграции отбираются из хромосом популяции, имеющих лучшее значение функции пригодности. Затем выполняется обмен между популяциями с помощью копирования нескольких решений с лучшими значениями функции пригодности. Чтобы размер популяции не изменялся, из популяции одновременно удаляется такое же количество хромосом с худшим значением функции пригодности. На рис. 1 представлена схема предложенной модели параллельного генетического алгоритма, выполняемого на 2-х популяциях [9–14]. На практике количество популяций может быть значительно больше.

Важно правильно подобрать частоту выполнения оператора миграции (интервал времени между миграциями) и число хромосом, участвующих в миграции. При слишком большой частоте возникает опасность перемешивания решений и преждевременной сходимости генетического алгоритма. Такая же ситуация возникает и при малой частоте миграции. Поэтому для определения вероятности выполнения оператора в разработанном алгоритме используется модуль нечёткого управления. Вероятность выполнения оператора миграции, наряду с вероятностями кроссинговера и мутации, определяется на основе данных об эффективности эволюции в каждом процессе.

Увеличение быстродействия алгоритма решения задачи размещения достигается путем организации параллельных эволюционных вычислений, а также за счёт использования многопоточности на локальном уровне, при расчёте значений целевых функций. Вычисление значения целевой функции каждой хромосомы выполняется в отдельном потоке. Популяция делится на  $S/N$  блоков хромосом, где  $S$  – размер популяции,  $N$  – предустановленное количество потоков расчёта целевой функции. Каждый блок содержит  $N$  хромосом. Для каждой хромосомы расчёт значения целевой функции запускается в отдельном потоке. После завершения всех запущенных потоков начинается вычисление значений целевой функции хромосом следующего блока. Процесс продолжается пока не будут найдены значения целевой функции для всех хромосом. Схема процедуры распараллеливания расчёта значений ЦФ представлена на рис. 2.

Модуль нечёткого управления (нечёткий логический контроллер) состоит из четырех взаимосвязанных блоков: блок фаззификации, база правил, блок дефаззификации, блок вывода [15].

База правил состоит из набора нечетких правил [15, 16]. Каждое из нечетких правил записывается в виде логической конструкции: IF ... THEN ... С точки зрения теории множеств нечеткие правила представляют собой нечеткое отношение  $R^{(k)}$ . Функция принадлежности такого нечеткого множества – это нечеткая импликация

$$\mu_{R^{(k)}}(x, y) = \mu_{A^k \rightarrow B^k}(x, y), \quad (1)$$

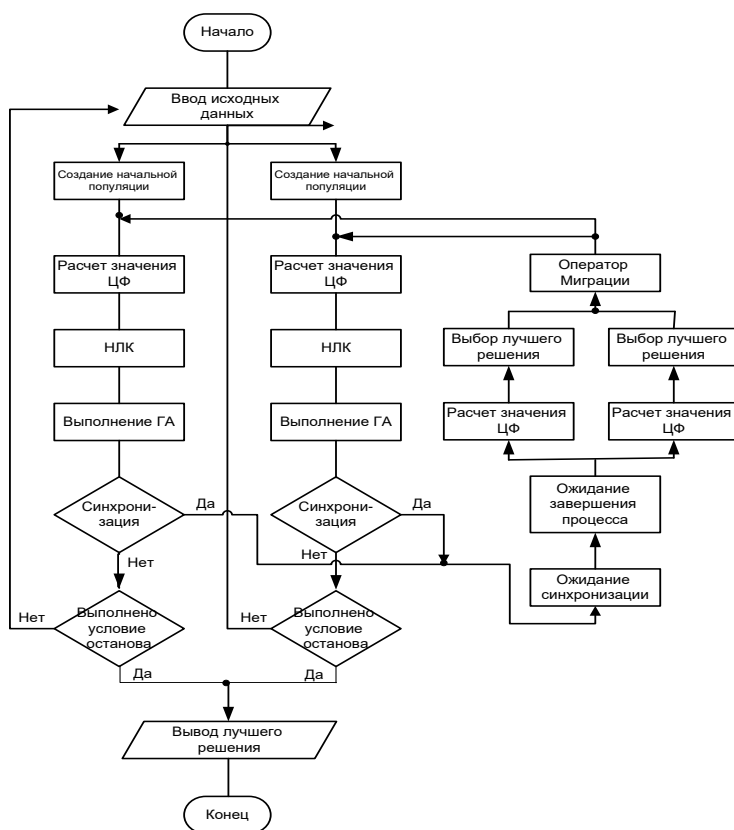


Рис. 1. Схема параллельного алгоритма

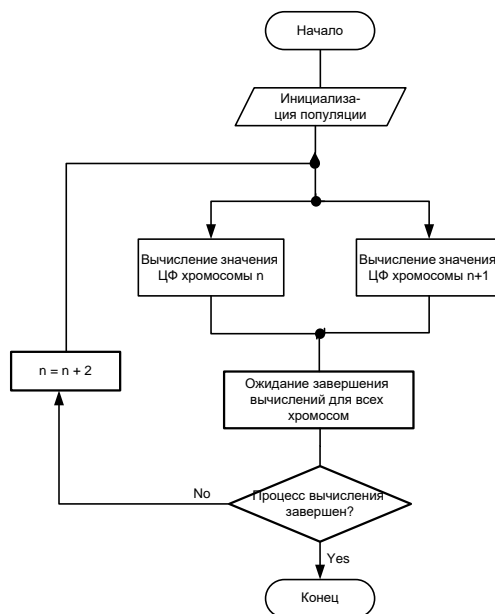


Рис. 2. Схема распараллеливания расчета целевой функции

Блок фаззификации выполняет трансформацию чётких данных в нечеткие множества. Для этой цели часто используют представление нечеткой величины в виде синглтона.

Процедура получения результата нечеткого вывода с использованием базы правил состоит из следующих шагов [18–20]:

1. Определение уровня срабатывания каждого из правил.
2. Определение результата нечеткого вывода по каждому из правил.
3. Агрегирование индивидуальных результатов нечеткого вывода в общий результат, характерный для всей базы нечетких правил.

Для оценки эффективности эволюционного процесса могут использоваться различные параметры, например, значения функции пригодности лучшего решения в популяции, средняя целевая функция текущей популяции, изменение среднего значения целевой функции популяции и т.д. [15, 22].

При разработке структуры модуля нечеткого управления необходимо также определиться с формой представления нечетких множеств. Для этого можно использовать, например, функцию Гаусса [21]:

$$\mu_{A_j^k}(x_i) = \exp\left(-\left(\frac{x_i - \bar{x}_j^{-k}}{\sigma_j^k}\right)^2\right), \quad (2)$$

где  $\bar{x}_j^{-k}$  – центр, а  $\sigma_j^k$  – ширина Гауссовской кривой.

Результирующая функция модуля нечеткого управления имеет следующий вид:

$$\bar{y} = \frac{\sum_{k=1}^N y^{-k} \left( \prod_{i=1}^n \exp\left(-\left(\frac{x_i - \bar{x}_j^{-k}}{\sigma_j^k}\right)^2\right) \right)}{\sum_{k=1}^N \left( \prod_{i=1}^n \exp\left(-\left(\frac{x_i - \bar{x}_j^{-k}}{\sigma_j^k}\right)^2\right) \right)}. \quad (2)$$

Каждый элемент можно задать в форме функционального блока (сумма, произведение, функция Гаусса). Объединение функциональных блоков дает многослойную нейронную сеть. Для решения поставленной задачи была использована модель нейронной сети, которая имеет 4 слоя (L1 – L4), 4 входных элемента и один выход. Элементы первого слоя реализуют вышеприведенную функцию Гаусса. Элементами нейронной сети являются мультипликаторы, сумматоры и делитель.

Модуль нечеткого управления на основании оценки заданных контрольных значений может изменять значения вероятности выполнения генетических операторов. В качестве контрольных значений могут использоваться различные характеристики популяции, например, разнообразие генотипа и фенотипа популяции, величина лучшего и среднего значения функции пригодности в текущей популяции, динамика изменения лучшего решения и среднего значений функции пригодности [22].

В предложенном алгоритме для оценки текущего состояния были выбраны 4 параметра [23]:

$$e_1(t) = \frac{f_{ave}(t) - f_{best}(t)}{f_{ave}(t)}; \quad e_2(t) = \frac{f_{ave}(t) - f_{best}(t)}{f_{worst}(t) - f_{best}(t)};$$

$$e_3(t) = \frac{f_{best}(t) - f_{best}(t-1)}{f_{best}(t)}; \quad e_4(t) = \frac{f_{ave}(t) - f_{ave}(t-1)}{f_{ave}(t)},$$

где  $t$  – номер текущей популяции,  $f_{best}(t)$ ,  $f_{worst}(t)$ ,  $f_{ave}(t)$  – лучшее, худшее и среднее значение целевой функции на итерации  $t$  соответственно,  $f_{best}(t-1)$ ,  $f_{ave}(t-1)$  – лучшее и среднее значение ЦФ на итерации  $(t-1)$ .

В качестве выходных параметров блока нечеткого управления используются значения вероятностей выполнения основных генетических операторов: кроссинговера  $P_c(t)$ , мутации  $P_m(t)$  и миграции  $P_{mg}(t)$ .

Для хранения данных о топологии печатной платы используется LEF/DEF спецификация [24]. LEF (Library Exchange Format) - это спецификация для представления физической структуры микросхемы в формате ASCII. Она включает правила оформления и абстрактную информации об элементах. LEF используется в сочетании с DEF (Design Exchange Format) спецификацией, которая используется для представления полного размещения элементов микросхемы.

**Программная реализация.** Для создания графического интерфейса использовался фреймворк Qt 5.6. Qt – представляет из себя кросс-платформенный инструментарий разработки прикладного программного обеспечения, широко используемый для создания графических интерфейсов [25]. Он написан на C++ и предоставляет мощные расширения этого языка. Включает в себя все основные классы, которые могут потребоваться при разработке прикладного программного обеспечения, начиная от элементов графического интерфейса и заканчивая классами для работы с сетью, базами данных и XML. Qt является полностью объектно-ориентированным, легко расширяемым и поддерживающим технику компонентного программирования.

Для визуализации и анализа работы нечёткого логического контроллера и алгоритма размещения можно вывести на экран графики изменения входных и выходных параметров контроллера. На рис. 3 показаны графики изменения минимального и среднего значений целевой функции в процессе решения задач размещения и трассировки.

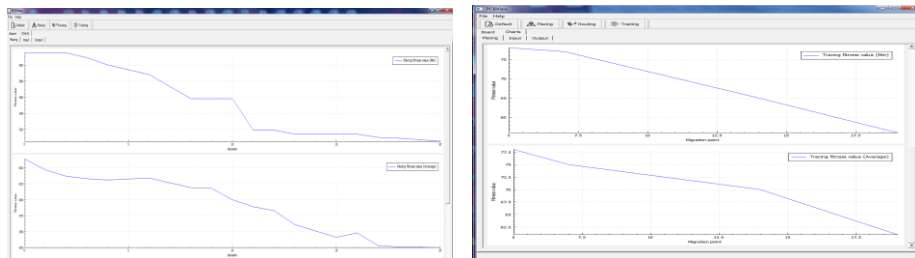


Рис. 3. Графики изменения минимального и среднего значений целевой функции

На рис. 4 приведены графики изменения значений управляющих параметров логического контроллера.

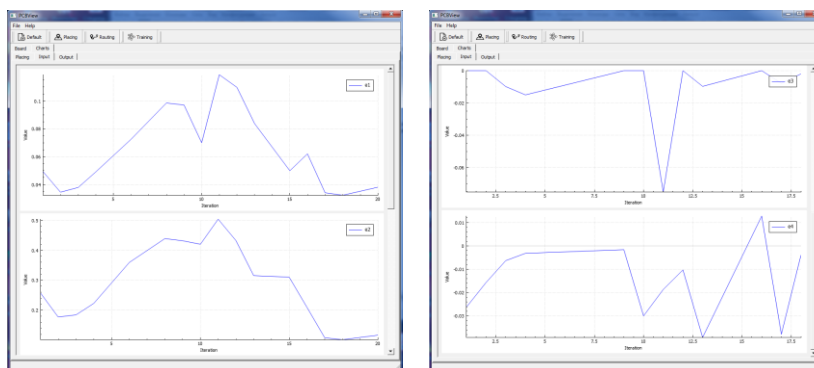


Рис. 4. Графики значений параметров НЛК

На рис. 5 показаны графики изменения значений контрольных параметров и вероятностей выполнения генетических операторов: кроссинговера, мутации и миграции.

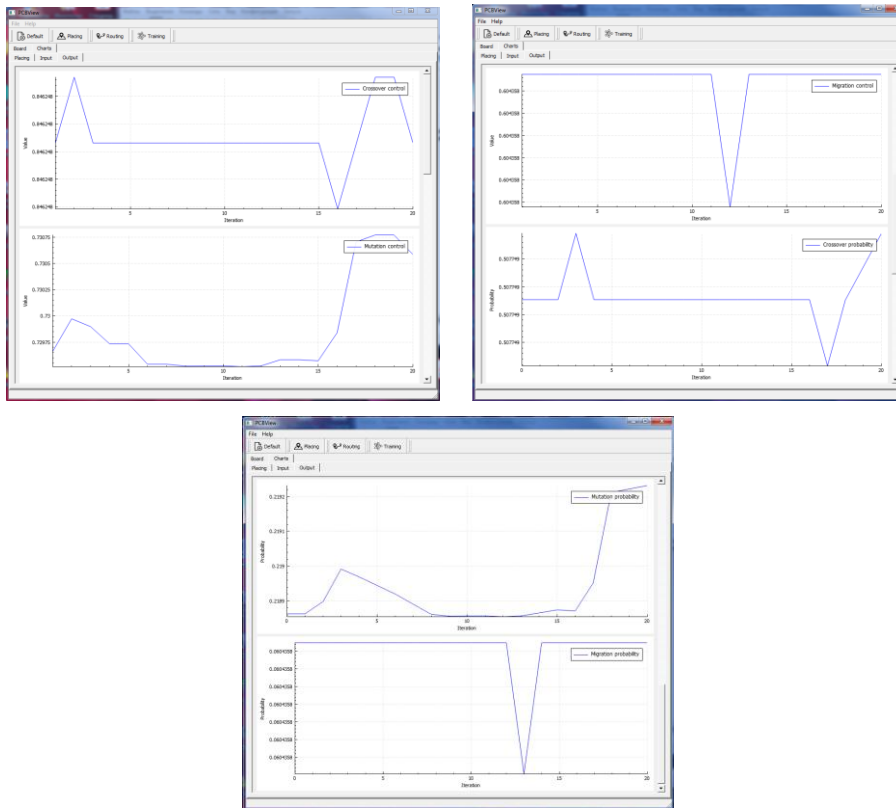


Рис. 5. Изменение вероятности выполнения генетических операторов в процессе работы алгоритма

В ходе выполнения вычислительных экспериментов проводилось исследование временных характеристик разработанного алгоритма. Была выполнена экспериментальная оценка временной сложности разработанного алгоритма, а также сравнение теоретических и эмпирических оценок. Результаты проведенных экспериментов приведены на рис. 6.

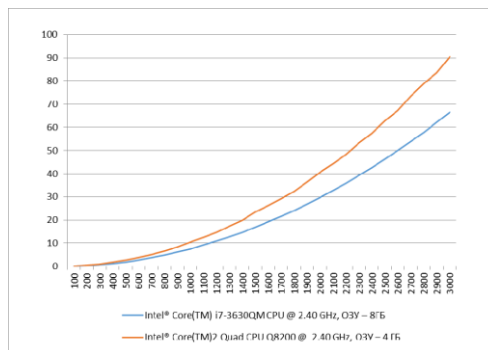


Рис. 6. Зависимость времени выполнения алгоритма от числа элементов



Также был проведен ряд экспериментов с целью определить эффективность использования нечеткого контроллера в работе предложенного гибридного алгоритма решения задачи размещения. Исследование эффективности алгоритма проводилось для решения задачи размещения 300 случайно сгенерированных элементов. Алгоритм тестировался на одном и том же наборе исходных данных при различном числе итераций. Размер популяции – 20 хромосом. Результаты экспериментов представлены на рис. 7.

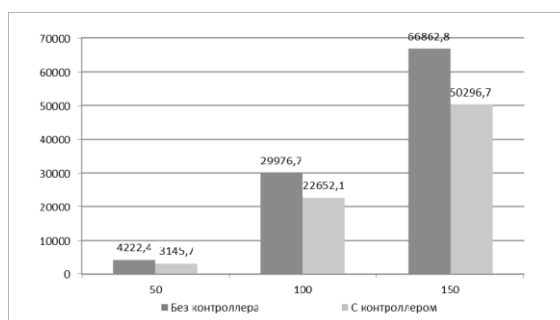


Рис. 7. Сравнение результатов полученных с использованием НЛК и без него

**Заключение.** Разработана архитектура параллельного многопопуляционного алгоритма генетического поиска на основе модели островов. Предложен гибридный оператор миграции, вероятность применения которого регулируется модулем нечёткого управления. С целью увеличения быстродействия алгоритма решения задачи размещения предложена схема расчёте значений целевой функции с использованием многопоточности на локальном уровне для организации параллельных эволюционных вычислений. Многопопуляционный алгоритм интегрирован с элементами глобального однопопуляционного алгоритма путём реализации механизма распределения вычисления целевой функции между несколькими потоками.

Разработана математическая модель модуля нечёткого управления, который предназначен для динамического изменения управляющих параметров параллельного генетического алгоритма. Предложен вариант интеграции генетического алгоритма и модуля нечёткого управления. Предложена структура гибридного алгоритма решения задачи размещения разногабаритных элементов, разработаны модифицированные генетические операторы, а также компоненты гибридного алгоритма.

Выполнена оценка временной сложности разработанных алгоритмов. Проанализированы результаты применения нейро-нечёткого логического контроллера и блока обучения контроллера, который позволяет получить оптимальные параметры управления. Проведена оценка оптимального количества потоков параллельного генетического алгоритма.

Разработан программно-алгоритмический комплекс для решения задачи размещения. В настоящее время проводятся серии вычислительных экспериментов для определения оптимальных параметров алгоритма и изучения эффективности предложенного подхода.

Анализ полученных экспериментальных данных позволяет сделать вывод, что включение в модель гибридного алгоритма нечеткого контроллера позволяет улучшить результат решение задачи при одинаковом количестве итераций в среднем на 25%. Эффективность использования контроллера повышается после введения блока обучения на основе модели искусственной нейронной сети.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Shervani N.* Algorithms for VLSI physical design automation. – USA, Kluwer Academy Publisher, 1995.
2. *Charles J. Alpert, Dinesh P. Mehta, Sachin S. Sapatnekar.* Handbook of algorithms for physical design automation. – CRC Press, New York, USA, 2009.
3. *Гладков Л.А., Курейчик В.М., Курейчик В.В., Сороколетов П.В.* Биоинспирированные методы в оптимизации. – М.: Физматлит, 2009.
4. *Карпенко А.П.* Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. – М.: Изд-во МГТУ им. Баумана, 2016.
5. *Cohon J.P., Karro J., Lienig J.* Evolutionary Algorithms for the Physical Design of VLSI Circuits. Advances in Evolutionary Computing: Theory and Applications, Ghosh A., Tsutsui S. (eds.). – Springer Verlag, London, 2003. – P. 683-712.
6. *Гладков Л.А.* О некоторых подходах к построению гибридных интеллектуальных систем для решения графовых задач // Новости искусственного интеллекта. – 2000. – № 3. – С. 71-90.
7. *Alba E., Tomassini M.* Parallelism and evolutionary algorithms // IEEE T. Evolut. Comput. – 2002. – Vol. 6. – P. 443-461.
8. *Zhongyang X., Zhang Y., Zhang L., Niu S.* A parallel classification algorithm based on hybrid genetic algorithm // Proc. of the 6th World Congress on Intelligent Control and Automation, Dalian, China. – 2006. – P. 3237-3240.
9. *Гладков Л.А., Курейчик В.В., Курейчик В.М.* Генетические алгоритмы. – М.: Физматлит, 2010.
10. *Редько В.Г.* Моделирование когнитивной эволюции. На пути к теории эволюционного происхождения мышления. – М.: Изд-во УРСС, 2015.
11. *Kar A.K.* Bio Inspired Computing - A Review of Algorithms and Scope of Applications // Expert Systems with Applications. – 2016. – Vol. 59. – P. 20-32.
12. *Baqais A.A.B.* A Multi-view Comparison of Various Metaheuristic and Soft Computing Algorithms // I.J. Mathematical Sciences and Computing. – 2017. – Vol. 1.3 (4). – P. 8-19.
13. *Roy S., Biswas S., Chaudhuri S.S.* Nature-Inspired Swarm Intelligence and Its Applications // I.J. Modern Education and Computer Science. – 2014. – No. 12. – P. 55-65.
14. Nature-Inspired Computation and Swarm Intelligence: Algorithms, Theory and Applications. – Academic Press, 2020. – ISBN 978-0-12-819714-1. – <https://doi.org/10.1016/C2019-0-00628-0>.
15. *Michael A., Takagi H.* Dynamic control of genetic algorithms using fuzzy logic techniques // Proc. of the 5th International Conference on Genetic Algorithms. – Morgan Kaufmann, 1993. – P. 76-83.
16. *Herrera F., Lozano M.* Adaptation of genetic algorithm parameters based on fuzzy logic controllers. In: F. Herrera, J.L. Verdegay (eds.) // Genetic Algorithms and Soft Computing. – Physica-Verlag, Heidelberg, 1996. – P. 95-124.
17. *Herrera F., Lozano M.* Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions // Soft Computing. – Springer-Verlag, 2003. – No. 7. – P. 545-562.
18. Нечеткие гибридные системы. Теория и практика / под ред. Н.Г. Ярушкиной. – М.: Физматлит, 2007.
19. *Ярушкина Н.Г.* Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004.
20. *Борисов В.В., Круглов В.В., Федюлов А.С.* Нечеткие модели и сети. – М.: Горячая линия – Телеком, 2007.
21. *Рутковская Д., Пилиньский М., Рутковский Л.* Нейронные сети, генетические алгоритмы и нечеткие системы. – М.: Горячая линия-Телеком, 2004.
22. *Gladkov L.A., Gladkova N.V., Gusev N.Y., Semushina N.S.* Integrated approach to the solution of computer-aided design problems. // Proceedings of the 4th International Scientific Conference “Intelligent Information Technologies for Industry” (ITI'19). Advances in Intelligent Systems and Computing. – Springer, Cham, 2020. – Vol. 1156. – P. 465-476.
23. *Gladkov L.A., Gladkova N.V., Leiba S.N., Strakhov N.E.* Development and research of the hybrid approach to the solution of optimization design problems. // Advances in Intelligent Systems and Computing. Vol. 875. International Conference on Intelligent Information Technologies for Industry ITI'18. – Springer Nature Switzerland AG, 2019. – Vol. 2. – P. 246-257.
24. *Library Exchange Format.* University of Maryland, Baltimore County, 2011.
25. *Qt Documentation.* – <http://doc.qt.io/qt-5/reference-overview.html>.

## REFERENCES

1. *Shervani N.* Algorithms for VLSI physical design automation. USA, Kluwer Academy Publisher, 1995.
2. *Charles J. Alpert, Dinesh P. Mehta, Sachin S. Sapatnekar.* Handbook of algorithms for physical design automation. CRC Press, New York, USA, 2009.
3. *Gladkov L.A., Kureychik V.M., Kureychik V.V., Sorokoletov P.V.* Bioinspirirovannyye metody v optimizatsii [Bioinspired methods in optimization]. Moscow: Fizmatlit, 2009.
4. *Karpenko A.P.* Sovremennyye algoritmy poiskovoy optimizatsii. Algoritmy, vdokhnovlennyye prirodoy [Modern search engine optimization algorithms. Algorithms inspired by nature]. Moscow: Izd-vo MGTU im. Bauman, 2016.
5. *Cohoon J.P., Karro J., Lienig J.* Evolutionary Algorithms for the Physical Design of VLSI Circuits. Advances in Evolutionary Computing: Theory and Applications, Ghosh, A., Tsutsui, S. (eds.). Springer Verlag, London, 2003, pp. 683-712.
6. *Gladkov L.A.* O nekotorykh podkhodakh k postroeniyu gibridnykh intellektual'nykh sistem dlya resheniya grafovyykh zadach [About some approaches to the construction of hybrid intelligent systems for solving graph problems], *Novosti iskusstvennogo intellekta* [Artificial Intelligence News], 2000, No. 3, pp. 71-90.
7. *Alba E., Tomassini M.* Parallelism and evolutionary algorithms, *IEEE T. Evolut. Comput.*, 2002, Vol. 6, pp. 443-461.
8. *Zhongyang X., Zhang Y., Zhang L., Niu S.* A parallel classification algorithm based on hybrid genetic algorithm, *Proc. of the 6th World Congress on Intelligent Control and Automation, Dalian, China, 2006*, pp. 3237-3240.
9. *Gladkov L.A., Kureychik V.V., Kureychik V.M.* Geneticheskie algoritmy [Genetic algorithms]. Moscow: Fizmatlit, 2010.
10. *Red'ko V.G.* Modelirovaniye kognitivnoy evolyutsii. Na puti k teorii evolyutsionnogo proiskhozhdeniya myshleniya [Modeling cognitive evolution. On the way to the theory of the evolutionary origin of thinking]. Moscow: Izd-vo URSS, 2015.
11. *Kar A.K.* Bio Inspired Computing - A Review of Algorithms and Scope of Applications, *Expert Systems with Applications*, 2016, Vol. 59, pp. 20-32.
12. *Baqais A.A.B.* A Multi-view Comparison of Various Metaheuristic and Soft Computing Algorithms, *I.J. Mathematical Sciences and Computing*, 2017, Vol. 1.3 (4), pp. 8-19.
13. *Roy S., Biswas S., Chaudhuri S.S.* Nature-Inspired Swarm Intelligence and Its Applications, *I.J. Modern Education and Computer Science*, 2014, No. 12, pp. 55-65.
14. *Nature-Inspired Computation and Swarm Intelligence: Algorithms, Theory and Applications.* – Academic Press, 2020. ISBN 978-0-12-819714-1. Available at: <https://doi.org/10.1016/C2019-0-00628-0>.
15. *Michael A., Takagi H.* Dynamic control of genetic algorithms using fuzzy logic techniques, *Proc. of the 5th International Conference on Genetic Algorithms*. Morgan Kaufmann, 1993, pp. 76-83.
16. *Herrera F., Lozano M.* Adaptation of genetic algorithm parameters based on fuzzy logic controllers, In: F. Herrera, J.L. Verdegay (eds.), *Genetic Algorithms and Soft Computing*. Physica-Verlag, Heidelberg, 1996, pp. 95-124.
17. *Herrera F., Lozano M.* Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions, *Soft Computing*. Springer-Verlag, 2003, No. 7, pp. 545-562.
18. *Nechetkie gibridnye sistemy. Teoriya i praktika* [Fuzzy hybrid systems. Theory and practice], ed. by N.G. YArushkinoy. Moscow: Fizmatlit, 2007.
19. *Yarushkina N.G.* Osnovy teorii nechetkikh i gibridnykh system [Fundamentals of the theory of fuzzy and hybrid systems]. Moscow: Finansy i stati-stika, 2004.
20. *Borisov V.V., Kruglov V.V., Fedulov A.S.* Nechetkie modeli i seti [Fuzzy models and networks]. Moscow: Goryachaya liniya – Telekom, 2007.
21. *Rutkovskaya D., Pilin'skiy M., Rutkovskiy L.* Neyronnye seti, geneticheskie algoritmy i nechetkie sistemy [Neural networks, genetic algorithms and non-discrete systems]. Moscow: Goryachaya liniya-Telekom, 2004.
22. *Gladkov L.A., Gladkova N.V., Gusev N.Y., Semushina N.S.* Integrated approach to the solution of computer-aided design problems, *Proceedings of the 4th International Scientific Conference "Intelligent Information Technologies for Industry" (IITI'19). Advances in Intelligent Systems and Computing*. Springer, Cham, 2020, Vol. 1156, pp. 465-476.

23. Gladkov L.A., Gladkova N.V., Leiba S.N., Strakhov N.E. Development and research of the hybrid approach to the solution of optimization design problems, *Advances in Intelligent Systems and Computing*. Vol. 875. *International Conference on Intelligent Information Technologies for Industry IIT'18*. Springer Nature Switzerland AG, 2019, Vol. 2, pp. 246-257.
24. *Library Exchange Format*. University of Maryland, Baltimore County, 2011.
25. *Qt Documentation*. Available at: <http://doc.qt.io/qt-5/reference-overview.html>.

Статью рекомендовала к опубликованию д.т.н., профессор Л.С. Лисицына.

**Ясир Муханад Джаббар Ясир** – Южный федеральный университет; e-mail: yasir\_82@mail.ru; г. Таганрог, Россия; тел.: 88634371625; кафедра САПР; аспирант.

**Гладков Леонид Анатольевич** – e-mail: lagladkov@sfnedu.ru; тел.: 88634371625; кафедра САПР; к.т.н.; доцент.

**Гладкова Надежда Викторовна** – e-mail: nvgladkova@sfnedu.ru; тел.: 88634393260; кафедра САПР; старший преподаватель.

**Yasir Mukhanad Dzhabbar Yasir** – Southern Federal University; e-mail: yasir\_82@mail.ru; Taganrog, Russia; phone: +78634371625; the department of CAD; postgraduate student.

**Gladkov Leonid Anatol'evich** – e-mail: lagladkov@sfnedu.ru; phone: +78634371625; the department of CAD; cand. of eng. sc.; associate professor.

**Gladkova Nadezda Viktorovna** — e-mail: nvgladkova@sfnedu.ru; phone: +788634393260; the department of CAD; senior teacher.

УДК 004.056

DOI 10.18522/2311-3103-2022-5-17-29

**Д.М. Зарубин, В.П. Добрица, Е.А. Титенко**

### **МЕТОД ОБЕСПЕЧЕНИЯ ЗАЩИТЫ ПЕРЕДАВАЕМЫХ СООБЩЕНИЙ В СИСТЕМЕ ADS-B С ИСПОЛЬЗОВАНИЕМ АППАРАТА КЛЕТОЧНЫХ АВТОМАТОВ**

*Цель исследования – разработка метода кодирования передаваемых ADS-B сообщений между воздушными судами в процессе полета. Открытый формат 1090ES передаваемых данных является критическим в плане проведения различных типов атак, которые могут привести к нарушению безопасности полетов воздушных судов. Работа направлена на применение средств кодирования и декодирования сообщений с закрытым ключом. Методы исследования основаны на применении и развитии потокового шифрования данных с использованием одномерных клеточных автоматов. Они работают в режиме генератора псевдослучайных последовательностей, преобразующих элементарных состояний ячейки одномерного клеточного автомата на основе типовых аппаратно-ориентированных операций. В основу процессов кодирования и декодирования полей данных положена аналитическое выражение, использующее типовые логические операции (дизъюнкция, сумма по модулю два). Это свойство позволяет вести параллельную обработку полей данных сообщения и создавать неповторяющиеся последовательности кодов. Результаты – создан метод обеспечения защиты передаваемых данных, дополнительно кодирующий на передаче и декодирующий на приеме сообщения. Отличительная особенность метода – сохранение формата протокола. Выполнена оценка вычислительной сложности работы клеточного автомата. Метод использует одномерный клеточный автомат, который выполняет кодирование и декодирование целевых полей (координаты, курс и др.) с использованием генератора псевдослучайных чисел. Разработанный метод относится к классу аппаратно-ориентированных методов. Критические для кодирования и декодирования свойства периодичности полей данных и длины ключа устраняются путем выбора начального иррационального значения и организации «потоковой» работы кодировщика. Если кодирующий автомат работает в потоковом режиме, текущее значение зависит от предыстории некоторой глубины, определение длины «автоматического ключа» из ADS-B сообщения*