

Telpukhov Dmitry Vladimirovich – Institute for Design Problems in Microelectronics of Russian Academy of Sciences, e-mail: nofrost@inbox.ru; Moscow, Zelenograd, Russia; dr. of eng. sc.; head of department of integrated circuits design methodology.

Zhukova Tatyana Dmitrievna – e-mail: zhukova_t@ippm.ru; junior researcher.

Schelokov Albert Nikolaevich – e-mail: schan@ippm.ru; cand. of phis.-math. sc.; senior researcher.

Kretinina Polina Denisovna – e-mail: kretinapolina@gmail.com; research engineer.

УДК 519.712.2

DOI 10.18522/2311-3103-2021-4-231-243

Л.А. Гладков, М.Д. Ясер, Н.В. Гладкова

ГИБРИДНЫЙ МЕТОД РЕШЕНИЯ ЗАДАЧИ РАЗМЕЩЕНИЯ ЭЛЕМЕНТОВ ЦИФРОВЫХ ВЫЧИСЛИТЕЛЬНЫХ УСТРОЙСТВ*

Рассматривается задача размещения элементов цифровой вычислительной техники. Проведен анализ современного состояния исследований по данной теме, отмечена актуальность рассматриваемой задачи. Подчеркнута важность разработки новых эффективных методов решения подобных задач. Показано место задачи размещения в общем цикле конструкторского этапа проектирования. Отмечена важность качественного решения задачи размещения с точки зрения успешного выполнения последующих этапов проектирования. Отмечена важность минимизации задержек соединений в процессе проектирования устройств большой размерности. Проведен обзор и анализ различных моделей и критериев оценки решения задачи размещения. Подчеркнуто, что важнейшим критерием является длина соединений, она оказывает существенное влияние на применяемые при проектировании технологии. Выполнена комплексная математическая постановка задачи размещения элементов цифровой вычислительной техники. Приведена целевая функция и ограничения рассматриваемой задачи размещения как задачи оптимизации. Проанализированы перспективные подходы к решению задач проектирования, описаны гибридные методы и модели решения сложных многокритериальных задач оптимизации и проектирования. Описаны принципы работы и модель нечеткого логического контроллера. Приведено описание используемой схемы нечеткого управления. Определены функции различных блоков нечеткого логического контроллера. Предложена структура многослойной нейронной сети, реализующей функцию Гаусса. Описано взаимодействие блоков нечеткого генетического алгоритма. Предложена модель гибридного алгоритма решения задачи размещения. 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; задачи оптимизации; биоинспирированные алгоритмы; гибридные методы; генетические алгоритмы; нечеткое управление.

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

HYBRID METHOD FOR SOLVING THE PROBLEM OF PLACEMENT OF DIGITAL COMPUTER DEVICES

The problem of placing elements of digital computing technology is considered 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 such problems are highlighted. The place of the placement problem in the general cycle of

* Исследование выполнено при поддержке РФФИ в рамках научного проекта № 19-01-00715.

the design stage is shown. The importance of a high-quality solution to the placement problem from the point of view of the successful implementation of subsequent design stages is noted. The importance of minimizing connection delays in the design process of large-scale devices is noted. A review and analysis of various models and criteria for evaluating the solution to the placement problem is carried out. It was emphasized that the most important criterion is the length of the joints, it has a significant impact on the technologies used in the design. A complex mathematical formulation of the problem of placing elements of digital computing equipment has been completed. Perspective approaches to solving design problems are analyzed, hybrid methods and models for solving complex multicriteria optimization and design problems are described. The principles of operation and the model of a fuzzy logic controller are described. The description of the used fuzzy control scheme is given. The functions of various blocks of a fuzzy logic controller are determined. The structure of a multilayer neural network that implements the Gaussian function is proposed. The interaction of blocks of a fuzzy genetic algorithm is described. A model of a hybrid algorithm for solving the placement problem 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 to determine the effectiveness of the developed algorithm and select the optimal values of the control parameters were carried out.

Design automation; the problem of placing elements of digital computing devices; optimization problems; bioinspired algorithms; hybrid methods; genetic algorithms; fuzzy control.

Введение. Цикл проектирования элементов цифровой вычислительной техники включает следующие этапы: спецификация системы, функциональное проектирование, логическое проектирование, схемное проектирование, конструкторское проектирование, изготовление, сборка, тестирование и контроль [1–3]. В свою очередь конструкторский этап проектирования включает задачи: компоновки, размещения, трассировки и т.д.

Размещение – одна из важнейших задач этапа физического проектирования в процессе решения которой выполняется построение макета проектируемой схемы на основе заданного списка соединений, а также оценка времени прохождения сигналов и возможных задержек межсоединений. Важность данной задачи определяется тем, что на данном этапе строится пространственная модель расположения элементов, которая является основой выполнения всех последующих задач проектирования. В современных электронных устройствах задержки межсоединений становятся определяющим фактором, а поскольку взаимное расположение элементов схемы определяется на этапе размещения, это оказывает существенное влияние на качество проектирования [1–3].

В процессе решения задачи размещения выполняется преобразование существующего на уровне блоков / затворов / транзисторов списка цепей в фактическую схему за конечное время. Формируются основные строительные блоки на основе логического списка цепей, и после определения точного местоположения элементов схемы в каждой области кристалла выполняется общая оценка временных характеристик проектируемого объекта. В современных СБИС сложность и размерность проектируемых схем непрерывно увеличивается, необходимая тактовая частота продолжает расти из-за более высокой производительности и более сложных функциональных требований для одного чипа. Более того, в условиях активного масштабирования технологий в эпоху субмикронных значений, задержки между соединениями становятся доминирующим фактором для общей производительности чипа. Поскольку расположение элементов схемы и соответствующие задержки межсоединений определяются на этапе размещения, это оказывает существенное влияние на конечные характеристики проекта. Более того, если задача размещения решена плохо, практически невозможно выполнить синхронизацию, независимо от того, насколько успешно применяются другие физические методы синтеза и оптимизации трассировки. Следовательно, размещение рассматривается как один из наиболее важных и эффективных методов оптимизации в

процессе физического синтеза. Большинство существующих логических и физических алгоритмов оптимизации должны взаимодействовать с алгоритмом задачи размещения, чтобы улучшить сроки проектирования и гарантировать легальное решение по задаче размещению после проведения оптимизации. Следовательно, большинство программных и алгоритмических инструментов физического проектирования нацелены на повышение эффективности решения задачи размещения.

После того, как основные блоки зафиксированы в месте и определены границы областей для размещения остальных ячеек выполняется глобальное размещение, а за тем детальное размещение для внесения локальных улучшений. В процессе решения задачи размещения ячеек на рабочем поле производится назначение наиболее длинных соединений, которые могут увеличивать задержки в проблемных областях. Эти задержки затем пытаются уменьшить с помощью применения методов буферизации и определения размеров соединений. При этом решение задачи размещения элементов и синтеза рабочего варианта прокладки соединений с учетом имеющихся временных ограничений в современных системах проектирования происходит в едином процессе, называемом физическим синтезом. Физический синтез включает в себя практически все традиционные процессы физического проектирования: планирование кристалла, размещение, глобальную и детальную трассировку, при этом добавляется возможность учета требований по срокам проектирования. Безусловно, плохое качество решения задач планирования и размещения автоматически отражается на качестве физического синтеза, поэтому проектировщики выполняют этот процесс в виде отдельных итераций, чтобы своевременно выявлять и исправлять возможные проблемы, возникающие при решении задач на отдельных этапах проектирования.

Успешное завершение процесса физического синтеза по-прежнему требует своевременного исправления допущенных ошибок и учета проблем с зашумленностью, изменчивостью и технологичностью. К сожалению, для внесения таких исправлений разработчику иногда приходится вернуться к более ранним этапам процесса.

Постановка задачи. Стандартная цель решения задачи размещения состоит в задании такого расположения элементов схемы на рабочем поле, которое приводит к минимизации общей длины соединений схемы. Это связано с тем, что длина соединений может быть легко смоделирована и служит хорошим приближением первого порядка к реальным целевым функциям, таким как синхронизация, мощность и управляемость проекта. Также существуют различные формы оценки длины соединений. Наиболее популярными моделями оценки являются квадратичная длина соединений, линейная длина соединений или некоторое приближение линейной длины соединений, которые используются во многих алгоритмах размещения. В последнее время для оценки длины соединений активно используется модель задачи Штейнера, которая считается наиболее точной оценкой проложенной длины проводов, также использовалась в качестве целевой функции размещения в некоторых научных исследованиях. Независимо от используемой формы длины соединений, возможность получения минимальной длины соединений размещения имеет решающее значение для проектирования современных изделий микроэлектроники, поскольку длина соединений напрямую влияет на задержки электрических сигналов. Длина соединений также влияет на качество решения задачи трассировки. Трассировка выполняется сразу после размещения, и нет смысла создавать решение для размещения без учета условий для последующей трассировки. Длина соединений также оказывает существенное влияние на применяемые при проектировании технологии, что является еще одним важным аспектом физического синтеза.

Основной задачей размещения является определение местоположения элементов схемы в на кристалле. Следовательно, сначала необходимо определить область размещения, как правило, это прямоугольную область, границы которой задаются координатами угловых точек $(x_{low}, y_{low}, x_{high}, y_{high})$. Однако данное условие не является обязательным, и на самом деле в последнее время наблюдается более широкое разнообразие областей размещения, например использование областей размещения L-образной или T-образной формы. Однако для задачи глобального размещения использование прямоугольной области размещения по-прежнему является нормой. Исходными данными для начала решения задачи размещения является список соединений схемы, который может быть задан в виде графа $G = (X, U)$, где множество X - это элементы проектируемой схемы, а множество U - множество соединений элементов схемы [4]. В свою очередь множество вершин X состоит из двух непересекающихся подмножеств: $X = X_1 \cup X_2$, где X_1 - множество элементов, которым еще не назначены позиции, а X_2 - множество элементов, позиции которых уже определены. Местоположение каждого элемента x_i схемы должно находиться в границах заданной области размещения (рабочего поля).

Задача размещения разногабаритных элементов в пространстве может быть задана следующим образом:

1. Задаются ограничения возможной площади, на которой можно выполнять размещение элементов. Область размещения, как правило, задается в виде прямоугольника. При этом можно говорить о задаче размещения одинаковых элементов на поле с кратными габаритами или о решении сложной задачи размещения элементов с различными установочными площадями.

2. Задаются габариты элементов, которые необходимо разместить на рабочем поле, для чего достаточно определить, например, два размера: длину и ширину каждого типа элементов.

Итак, исходными данными задачи являются: a, b - размеры (длина и ширина) рабочего поля; $\{e_1, e_2, \dots, e_n\}$ - множество элементов схемы; U - список связности, отражающий связи элементов.

Необходимо найти такой вариант размещения элементов на рабочем поле, чтобы не происходило взаимного перекрытия элементов и суммарная длина соединений была минимальной

$$E = \{(x_1, y_1), \dots, (x_i, y_i), \dots, (x_n, y_n)\},$$

где (x_i, y_i) - координаты центра тяжести посадочных мест, предназначенных для размещения элементов.

С точки зрения теории оптимизации задача размещения может быть интерпретирована как задача оптимизации аддитивной целевой функции, включающей нормированное значение штрафа за перекрытие площадей размещаемых элементов и оценку суммарной длины соединений:

$$F = \min_{z_j \in Z} (k \cdot O(L(z_j)) + T(S_{sum}(z_j))),$$

где z_j - текущий вариант размещения; k - весовой коэффициент; S_{sum} - суммарная площадь перекрытия элементов; $O(L(z_j))$ - оценка длины соединений; $T(S_{sum}(z_j))$ - штрафа за перекрытие площадей.

Суммарная длина соединений рассчитывается по формуле:

$$L = \sum_{i=1}^n \sum_{j=1}^n d_{ij} c_{ij},$$

где d_{ij} – расстояние между установочными позициями на рабочем поле $d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$; c_{ij} – число связей между элементами i и j .

Для нормирования суммарной длины соединений вычисляют отношение $L(z_j)$ к L_{\max} , где

$$L_{\max} = n^2 \cdot \sqrt{a^2 + b^2};$$

$$O(L(z_j)) = L(z_j) / L_{\max}.$$

Общая площадь перекрытия вычисляется по следующей формуле:

$$S_p = \sum_{i=1}^n \sum_{j=1}^n S_{ij},$$

где S_{ij} – площадь перекрытия элементов z_i и z_j .

$$S_{ij} = [0,5(a_2 + a_1) - |x_2 - x_1|] [0,5(b_2 + b_1) - |y_2 - y_1|].$$

Штраф за перекрытие площадей:

$$P(S_{\text{общ}}) = S_{\text{общ}} / nab.$$

Описание алгоритма. С концептуальной точки зрения, создаваемые интеллектуальные системы автоматизированного проектирования можно классифицировать как смешанные искусственные целеориентированные системы, т.е. системы созданные человеком и объединяющие искусственные и естественные подсистемы, основой функционирования которых является факторы целесообразности [5].

Гибридные системы это гетерогенные системы, состоящие из разнородных компонентов, объединенных для достижения поставленных целей [6, 7]. Интеграция и гибридизация методов и технологий различной физической природы позволяет решать задачи, которые невозможно решить на основе традиционных подходов. Гибридные архитектуры, объединяющие несколько парадигм, помогают обойти недостатки, присущие отдельным методам, при этом наблюдается, так называемый, синергетический эффект, когда достоинства одного метода компенсируют недостатки другого [8].

К сожалению, в отличие от природных, «искусственные» системы, как правило, не обладают возможностями развития, самоорганизации, адаптации к изменяющимся внешним условиям. Следовательно, основной задачей разработчиков является необходимость обеспечить наличие таких свойств в проектируемых системах.

Одним из перспективным инструментом конструирования эффективных алгоритмов проектирования и оптимизации является гибридизация популяционного алгоритма с двумя и более популяционными и/или не популяционными алгоритмами [9 - 12]. Структура гибридного алгоритма позволяет сохранить преимущества популяционных алгоритмов, используя их на начальном этапе для эффективного сужения пространства поиска, а затем применить один из «классических» методов оптимизации для нахождения глобального экстремума.

Одной из гибридных моделей является нечеткий генетический алгоритм [13, 14]. Он сочетает поисковые возможности генетических алгоритмов и возможности систем нечеткого вывода для оценки и изменения управляющих параметров. Для этого используется специальный блок – нечеткий логический контроллер (НЛК), который использует имеющиеся нечеткие правила, оценивает ход процесса эволюции и разнообразие текущей популяции решений и, при необходимости, изменяет значения свободных переменных, таких например, как вероятности выполнения генетических операторов.

Схема НЛК (рис. 1) построена основе управления с обратной связью. НЛК преобразует текущую информацию о ходе эволюции и состоянии популяции решений к нечеткому виду, затем на основе заданных нечетких правил оценивает ее, определяет управляющее воздействие и возвращает скорректированные значения контрольных параметров.

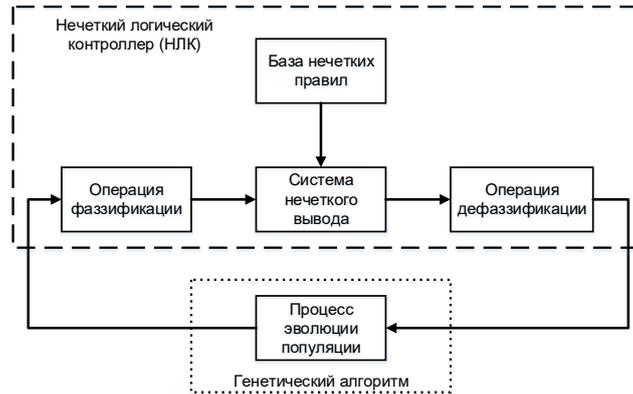


Рис. 1. Структура нечеткого логического контроллера

Основной закон управления можно записать в следующем виде [15–18]:

$$u(t) = f(e(t), e(t-1), \dots, e(t-r), u(t-1), \dots, u(t-r)),$$

где t – дискретное время; e – ошибка между модельным значением величины y^* и реальным значением выходного параметра объекта управления; f – нелинейная функция, которая определяется как отношение между входом и выходом НЛК.

Функционирование НЛК определяется набором лингвистически представляемых правил, основанных на экспертных знаниях, которые записываются в следующей форме: Если IF (множество условий), то THEN (заключение). НЛК оперирует нечеткими множествами. Поэтому входные значения в процессе фаззификации, преобразуются в лингвистические переменные. Они передаются в блок выработки решения, где формируется один или несколько нечетких наборов с соответствующими функциями принадлежности. После этого выполняется операция дефаззификации, т.е. преобразования этих нечетких множеств в управляющее воздействие.

Знания, составляющие основу корректного функционирования модуля нечеткого управления, записываются в виде нечеткого правила:

$$R^k: \text{IF}(x_1 \text{ это } A_1^k \text{ AND } \dots \text{ AND } x_n \text{ это } A_n^k) \text{ THEN}(y \text{ это } B^k).$$

Можно также представить эти правила в виде нечетких множеств с функцией принадлежности, заданной выражением:

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

Следовательно, если в качестве нечеткой импликации будет использоваться операция умножения, то получим формулу:

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

Декартово произведение нечетких множеств можно представить в виде:

$$\mu_{A^k}(x) = \mu_{A_1^k * \dots * A_n^k}(x) = \mu_{A_1^k}(x) \dots \mu_{A_n^k}(x).$$

В результате преобразований получаем следующее выражение для функции принадлежности нечеткого множества \bar{B}^k :

$$\mu_{\bar{B}^k}(y) = \sup_{x_1, \dots, x_n \in X} \left\{ \mu_{B^k}(y) \prod_{i=1}^n \mu_{A_i^k}(x_i) \mu_{A_i^k}(x_i) \right\}.$$

В качестве операции фаззификации используем операцию типа синглетон, пусть

$$A'(x) = \begin{cases} 1, & \text{если } x = \bar{x}, \\ 0, & \text{если } x \neq \bar{x}. \end{cases}$$

Заметим, что супремум в функции принадлежности достигается только в случае, когда $x = \bar{x}$, т.е. для $\mu_{A_i^k}(\bar{x}) = 1$. При этом получаем

$$\mu_{\bar{B}^k}(y) = \mu_{B^k}(y) \prod_{i=1}^n \mu_{A_i^k}(\bar{x}_i).$$

Применим метод дефаззификации «center average defuzzification», в соответствии с которым

$$\bar{y} = \frac{\sum_{k=1}^N \bar{y}^k \mu_{\bar{B}^k}(\bar{y}^k)}{\sum_{k=1}^N \mu_{\bar{B}^k}(\bar{y}^k)}.$$

В приведенной формуле \bar{y}^k – это центр нечеткого множества B_k , т.е. точка, в которой $\mu_{B^k}(y)$ достигает максимального значения.

Исходя из этого, получим равенство:

$$\bar{y} = \frac{\sum_{k=1}^N \bar{y}^k \left(\prod_{i=1}^n \mu_{A_i^k}(\bar{x}_i) \right)}{\sum_{k=1}^N \left(\prod_{i=1}^n \mu_{A_i^k}(\bar{x}_i) \right)}.$$

Завершающий этап в процессе проектирования модуля нечеткого управления – это определение формы представления нечетких множеств A_i^k , $1, \dots, n$; $k = 1, \dots, N$. Например, это может быть функция Гаусса

$$\mu_{A_i^k}(x_i) = \exp \left(- \left(\frac{x_i - \bar{x}_i^k}{\sigma_i^k} \right)^2 \right),$$

где параметры \bar{x}_i^k и σ_i^k имеют физическую интерпретацию: \bar{x}_i^k – это центр, а σ_i^k – ширина гауссовской кривой [19].

Как будет показано ниже, эти параметры могут модифицироваться в процессе обучения, что позволяет изменять положение и структуру нечетких множеств.

Объединим теперь все представленные элементы, и тогда модуль нечеткого управления приобретает окончательный вид:

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

Каждый элемент этого выражения можно задать в форме функционального блока (сумма, произведение, функция Гаусса), что после соответствующего объединения позволяет создать многослойную сеть (рис. 2).

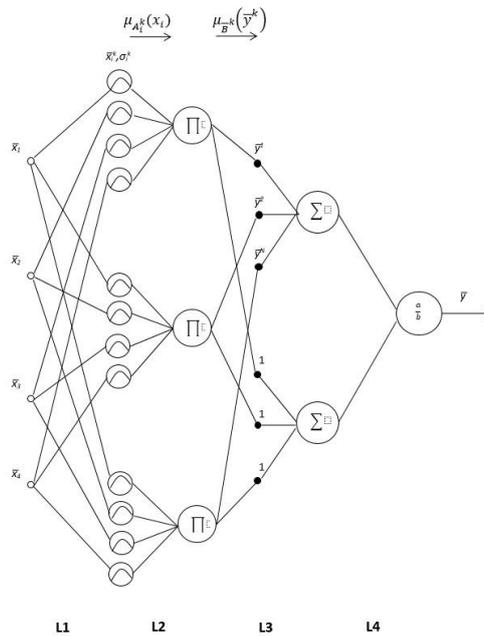


Рис. 2. Структура нейросети

На схеме показан модуль управления с четырьмя входами (n = 4). Слои обозначены символами от L1 до L4 и выделены серым фоном. Элементы, обозначенные символом Π (мультипликаторы), перемножают все входные сигналы, элементы, обозначенные символом Σ (сумматоры) – суммируют их, а элемент (a/b), делит один сигнал на другой. Черные поименованные точки, размещенные на связях, обозначают веса этих связей. Элементы слоя L1 реализуют функцию Гаусса с параметрами \bar{x}_i и σ_{ik} . Выражения и стрелки, размещенные над схемой, определяют направление распространения сигнала и его интерпретацию.

В генетических алгоритмах (ГА) управляющими параметрами, как правило, являются значения вероятности выполнения генетических операторов кроссинговера (Pc) и мутации (Pm), а также размер популяции [12, 20]. Взаимодействие блоков нечеткого генетического алгоритма показано на рис. 3.

Кроме того, очевидно, что эффективная работа генетического алгоритма зависит от верного подбора его параметров, что может быть крайне непростой задачей. Статично заданные ограничения могут привести к пропуску и редукции решений, обладающих высоким потенциалом к дальнейшим преобразованиям, что обязательно скажется на результате работы алгоритма. Чрезмерно высокая или

низкая вероятность применения оператора мутации часто приводит к раннему попаданию алгоритма в состояние локального оптимума. Динамически изменяющиеся ограничения могут помочь избежать этой ситуации, однако создание подобной адаптивной системы – задача очень высокой сложности, так как появляется проблема связанности этих ограничений с предметной областью, и невозможность их применения для широкого класса задач.



Рис. 3. Схема взаимодействия блоков нечеткого генетического алгоритма

Одним из способов реализации гибкой адаптивной системы ограничений является применение механизма нечеткой логики в виде контроллера – модуля, управляющего динамическими переменными в ходе работы системы, и направляющего поток выполнения программной реализации алгоритма.

В данной работе предлагается использование гибридного алгоритма на основе описанных ранее подходов. Общая схема работы предложенного гибридного алгоритма представлена на рис. 4.

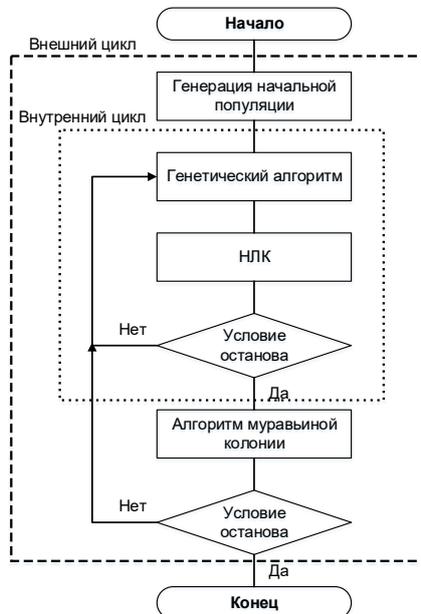


Рис. 4. Общая схема гибридного алгоритма

Гибридный алгоритм начинает работу с задания исходных данных, определения критерия оптимизации и ограничений задачи. После этого происходит создание стартовой популяции решений (начальное размещение) и оценка качества начальной популяции (как отдельных индивидов, так и среднего значения по популяции) с точки зрения выбранного критерия оптимизации.

После этого выполняется генетический алгоритм, происходит создание новой популяции решений и возможная корректировка управляющих параметров алгоритма с помощью нечеткого логического контроллера.

Следующий блок представляет собой модифицированный муравьиный алгоритм. Его задача попытаться улучшить полученное размещение.

В конце происходит проверка выполнения критерия останова алгоритма, условия не выполнены, алгоритм переходит к следующей итерации, в противном случае выполнение алгоритма завершается.

Изменение расположения элементов на рабочем поле продолжается до тех пор, происходит улучшение текущего значения целевой функции.

Таким образом, алгоритм состоит из трех основных модулей – генетического, муравьиной колонии и модуля нечеткого логического контроллера (НЛК).

Результаты компьютерных экспериментов. Для хранения данных о топологии печатной платы используется LEF/DEF спецификация. LEF (Library Exchange Format) – это спецификация для представления физической структуры интегральной схемы в формате ASCII. Она включает правила оформления и абстрактную информацию об элементах. LEF используется в сочетании с DEF (Design Exchange Format) спецификацией, которая используется для представления полного размещения элементов интегральной схемы.

Исследования проводились на двух конфигурации оборудования: Intel® Core(TM) i7-3630QM CPU @ 2.40 GHz, ОЗУ – 8ГБ (конфигурация 1) и Intel® Core(TM)2 Quad CPU Q8200 @ 2.40 GHz, ОЗУ – 4 ГБ (конфигурация 2). Во время исследования было проведено по 5 экспериментов с количеством элементов от 100 до 3000 с шагом 100. При постоянном количестве цепей, равном 50, количестве итераций, равном 50, хромосом популяции, равном 20 и 2 эволюционных процессах. При выполнении алгоритма использовался нечёткий логический контроллер.

Результаты экспериментов показаны на рис. 5. В них представлены зависимости среднего времени выполнения алгоритма от количества размещаемых элементов.

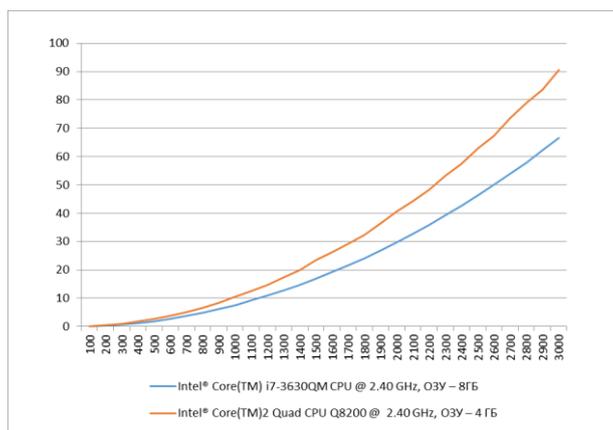


Рис. 5. Зависимость времени выполнения алгоритма от числа элементов

Проводилось сравнение результатов работы предложенного алгоритма с использованием нечеткого логического контроллера и без него. Полученные результаты представлены на рис. 6.

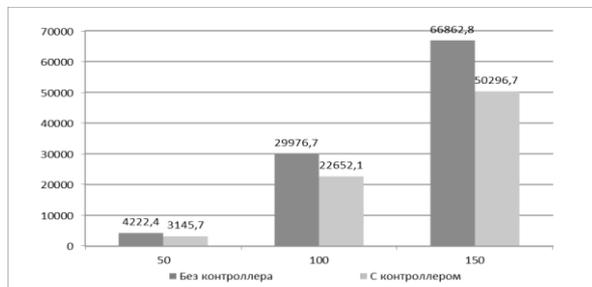


Рис. 6. Сравнение качества решений с контроллером и без его использования

Проанализировав представленные выше данные, можно сделать вывод, что использование НЛК 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.

Заключение. Предложена структура гибридного алгоритма решения задачи размещения разногабаритных элементов, разработаны модифицированные генетические операторы, а также компоненты гибридного алгоритма. Гибридный алгоритм реализован в виде прикладной программы, проведены серии вычислительных экспериментов для определения эффективности разработанного алгоритма и выбора оптимальных значений управляющих параметров.

Необходимо дальнейшее продолжение исследований разработанного алгоритма для выявления имеющихся закономерностей, совершенствования механизма реализации разработанных моделей и подходов.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Charles J. Alpert, Dinesh P. Mehta, Sachin S. Sapatnekar. Handbook of algorithms for physical design automation. – CRC Press, New York, USA, 2009.
2. Shervani N. Algorithms for VLSI physical design automation. – USA, Kluwer Academy Publisher, 1995. – 538 p.
3. 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. – P. 683-712.
4. Гладков Л.А., Курейчик В.В., Курейчик В.М. Дискретная математика. – М.: Физматлит, 2014.
5. Прангишвили И.В. Системный подход и общесистемные закономерности. – М.: СИНТЕГ, 2000.
6. Ярушкіна Н.Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004.
7. Батыршин И.З., Недосекин А.О. и др. Нечеткие гибридные системы. Теория и практика / под ред. Н.Г. Ярушкіной. – М.: Физматлит, 2007.
8. Haken H. Synergetics, an Introduction: Nonequilibrium Phase Transitions and Self-Organization in Physics, Chemistry, and Biology. – New York: Springer-Verlag, 1983.
9. Гладков Л.А., Курейчик В.М., Курейчик В.В., Сороколетов П.В. Биоинспирированные методы в оптимизации. – М.: Физматлит, 2009.
10. Гладков Л.А., Курейчик В.В., Курейчик В.М. Генетические алгоритмы. – М.: Физматлит, 2010.
11. Карпенко А.П. Современные алгоритмы поисковой оптимизации. Алгоритмы, вдохновленные природой. – М.: Изд-во МГТУ им. Баумана, 2016.

12. 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 // Proceedings of the Third International Scientific Conference "Intelligent Information Technologies for Industry" (ITI'18). ITI'18 2018. Advances in Intelligent Systems and Computing. – Vol 875. Springer, Cham. – P. 246-257.
13. Борисов В.В., Круглов В.В., Федулов А.С. Нечеткие модели и сети. – М.: Горячая линия – Телеком, 2007.
14. Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions // Soft Computing. 7(2003). – Springer-Verlag, 2003. – P. 545-562.
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. Lee M.A., Takagi H. Integrating design stages of fuzzy systems using genetic algorithms // Proceedings of the 2nd IEEE International Conference on Fuzzy System. – 1993. – P. 612-617.
17. King R.T.F.A., Radha B., Rughooputh H.C.S. A fuzzy logic controlled genetic algorithm for optimal electrical distribution network reconfiguration // Proceedings of 2004 IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan, 2004. – P. 577-582.
18. 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.
19. Рутковская Д., Пилиньский М., Рутковский Л. Нейронные сети, генетические алгоритмы и нечеткие системы. – М.: Горячая линия – Телеком, 2006.
20. 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. Vol. 875. – Springer, Cham, 2020. – P. 246-257.

REFERENCES

1. Charles J. Alpert, Dinesh P. Mehta, Sachin S. Sapatnekar. Handbook of algorithms for physical design automation. CRC Press, New York, USA, 2009.
2. Shervani N. Algorithms for VLSI physical design automation. USA, Kluwer Academy Publisher, 1995, 538 p.
3. 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.
4. Gladkov L.A., Kureychik V.V., Kureychik V.M. Diskretnaya matematika [Discrete Mathematics]. Moscow: Fizmatlit, 2014.
5. Prangishvili I.V. Sistemnyy podkhod i obshchesistmenye zakonomernosti [A systematic approach and system-wide patterns]. Moscow: SINTEG, 2000.
6. Yarushkina N.G. Osnovy teorii nechetkikh i gibridnykh system [Fundamentals of the theory of fuzzy and hybrid systems]. Moscow: Finansy i statistika, 2004.
7. Batyrshin I.Z., Nedosekin A.O. i dr. Nечеткие гибридные системы. Teoriya i praktika [Fuzzy hybrid systems. Theory and practice], ed. by N.G. Yarushkinoy. Moscow: Fizmatlit, 2007.
8. Haken H. Synergetics, an Introduction: Nonequilibrium Phase Transitions and Self-Organization in Physics, Chemistry, and Biology. New York: Springer-Verlag, 1983.
9. Gladkov L.A., Kureychik V.M., Kureychik V.V., Sorokoletov P.V. Bioinspirovannyye metody v optimizatsii [Bioinspired methods in optimization]. Moscow: Fizmatlit, 2009.
10. Gladkov L.A., Kureychik V.V., Kureychik V.M. Geneticheskie algoritmy [Genetic algorithms]. M.: Fizmatlit, 2010.
11. 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.
12. 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, *Proceedings of the Third International Scientific Conference "Intelligent Information Technologies for Industry" (ITI'18). ITI'18 2018. Advances in Intelligent Systems and Computing. Vol 875. Springer, Cham, pp. 246-257.*

13. Borisov V.V., Kruglov V.V., Fedulov A.S. Nechetkie modeli i seti [Fuzzy models and networks]. Moscow: Goryachaya liniya – Telekom, 2007.
14. Herrera F., Lozano M. Fuzzy Adaptive Genetic Algorithms: design, taxonomy, and future directions, *Soft Computing*. 7(2003). Springer-Verlag, 2003, pp. 545-562.
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. Lee M.A., Takagi H. Integrating design stages of fuzzy systems using genetic algorithms, *Proceedings of the 2nd IEEE International Conference on Fuzzy System*, 1993, pp. 612-617.
17. King R.T.F.A., Radha B., Rughooputh H.C.S. A fuzzy logic controlled genetic algorithm for optimal electrical distribution network reconfiguration, *Proceedings of 2004 IEEE International Conference on Networking, Sensing and Control, Taipei, Taiwan, 2004*, pp. 577-582.
18. 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.
19. Rutkovskaya D., Pilin'skiy M., Rutkovskiy L. Neyronnye seti, geneticheskie algoritmy i nechetkie sistemy [Neural networks, genetic algorithms and fuzzy systems]. Moscow: Goryachaya liniya – Telekom, 2006.
20. 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*. Vol. 875. Springer, Cham, 2020, pp. 246-257.

Статью рекомендовал к опубликованию д.т.н., профессор С.М. Ковалев.

Гладков Леонид Анатольевич – Южный федеральный университет; e-mail: lagladkov@sfedu.ru; г. Таганрог, Россия; тел.: 88634371625; кафедра САПР; к.т.н., доцент.

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

Ясир Муханад Джаббар – e-mail: yasir_82@mail.ru; тел.: 88634371625; кафедра САПР; аспирант.

Gladkov Leonid Anatol'evich – Southern Federal University; e-mail: lagladkov@sfedu.ru; Taganrog, Russia; phone: +78634371625; the department of CAD; cand. of eng. sc.; associate professor.

Gladkova Nadezhda Viktorovna – e-mail: nvgladkova@sfedu.ru; phone: +78634393260; the department of CAD; senior teacher.

Yasir Mukhanad Jabbar – e-mail: yasir_82@mail.ru; phone: +78634371625; the department of CAD; postgraduate student.

УДК 004.7

DOI 10.18522/2311-3103-2021-4-243-255

И.В. Родыгина, И.И. Бузенков, Ю.В. Каханец

КОНЦЕПЦИЯ ПОСТРОЕНИЯ И АРХИТЕКТУРА КРИТИЧЕСКИ ВАЖНЫХ ИНФОРМАЦИОННЫХ СЕГМЕНТОВ РЕГИОНАЛЬНОЙ ИНФОКОММУНИКАЦИОННОЙ СЕТИ

На сегодняшний день, в абсолютном измерении, ни одна отрасль государства не обходится без современных высокотехнологичных средств связи, соединяющих вычислительные системы и удалённые базы данных. Использование новых информационных и коммуникационных технологий в качестве средства разрешения противоречий, а также средства неявно воздействия, на мировой арене становится нарастающей угрозой для безопасности сообщества. В представленной работе рассматриваются не только этапы развития, но и основные тенденции, подходы к построению цифровых информационных систем, а также характерные