

УДК 681.3.06

А.Е. Лисовцова

НЕЧЕТКИЕ ГЕНЕТИЧЕСКИЕ АЛГОРИТМЫ В ЗАДАЧАХ РАЗМЕЩЕНИЯ*

В статье рассматривается решение задачи трассировки с использованием нечетких генетических алгоритмов. Этап трассировки является заключительным в структуре проектирования. С математической точки зрения этап трассировки считается самым сложным и трудоемким в вычислении пути оптимальности.

Трассировка; оптимальность пути; нечеткие генетические алгоритмы; мягкие вычисления.

A.E. Lisovcova

FUZZY GENETIC ALGORITHM IN THE PROBLEM LAYING

Problem is considered in article problem laying with using fuzzy genetic algorithm. Step laying present floodout stage in the structures design. From mathematical viewpoint stage laying consider very complex and labor-consuming in the path calculus optimal solution.

Laying; optimal solution; fuzzy genetic algorithm; soft computing.

Современный этап развития цивилизации, характеризующийся высокими темпами роста развития науки и техники, нелегко представить без использования изделий микроэлектронной промышленности в повседневной жизни.

Использование различных достижений микроэлектроники при производстве интегральных схем, больших интегральных схем, сверхбольших интегральных схем, суперкристаллов, обуславливает повышение основных характеристик электронных вычислительных средств, проектируемых на их основе, например, снижаются массогабаритные показатели, уменьшаются потребляемая и рассеиваемая мощности, повышаются быстродействие и надёжность [1]. Современные электронные вычислительные системы, такие как персональные компьютеры, оборудование вычислительных центров и исследовательских лабораторий, контроллеры и другие управляющие устройства и системы содержат в своём составе десятки, а иногда и сотни сверхбольших интегральных схем.

Разнообразие конструктивно-технологических методов создания сверх больших интегральных схем обусловлено стремлением не только улучшить их технико-экономические показатели, но и достичь общих целей: минимизировать длительность процесса проектирования, обеспечить проектирование сверх больших интегральных схем высокой сложности, повысить качество проектирования (главным образом, безошибочность) [1].

В целом проектирование полузаказных сверх больших интегральных схем зависит от возможностей заказчика и изготовителя сверх больших интегральных схем. Возрастание сложности проектов сверх больших интегральных схем и самих электронно-вычислительная машина почти исключает возможность использования традиционных методов для их проектирования. Наибольшее распространение получила концепция базового матричного кристалла, когда при разработке сверх больших интегральных схем используется специальная заготовка – базовый кристалл, представляющий собой матрицу ячеек нескоммутированных транзисторов или базовых логических и аналоговых элементов. При матричном методе сначала

* Работа выполнена при поддержке: РФФИ (грант № 09-07-00318), г/б № 2.1.2.1652.

создается базовый матричный кристалл в виде матриц из N строк и M столбцов (N и M – числа в диапазоне $5...10^3$) логических элементов и более сложных структур. Для реализации проекта необходимо разработать индивидуальную систему межсоединений этих транзисторов, т.е. достаточно спроектировать и изготовить только фотошаблоны для переменных слоев коммутации.

Выбор маршрута топологического проектирования определяется числом элементов и числом выводов элементов. Наиболее распространенным критерием оценки качества полученной топологии является критерий максимума плотности заполнения базового кристалла, равный отношению числа элементов МаБИС к числу ячеек (посадочных мест) кристалла. При плотности заполнения 90...100% применяют традиционное проектирование, при плотности 80...90% – автоматизированное, а при плотности менее 80% – автоматический синтез. Соответственно меняются сроки разработки топологии. На этапе контроля происходит сравнение исходной и восстановленной принципиальных схем, а также контролируется соблюдение конструкторских и технологических правил и норм проектирования. При подготовке технической документации проводится компиляция магнитных лент с информацией о тестах для контрольно-измерительного оборудования и с программами управления электронно-лучевыми или оптическими генераторами изображения.

Исходной информацией для системы является функционально-логическое описание больших интегральных схем в базисе стандартных топологических элементов. Это описание должно находиться в базе данных систем автоматизированного проектирования и включать следующие параметры: число элементов и число цепей, размеры каждого элемента, список номеров и координат внешних выводов для каждого элемента, число внешних выводов больших интегральных схем, списки номеров этих выводов и координаты контактных площадок. Кроме того, необходим учет следующих конструктивно-технологических параметров: размеров межслойных переходов и контактных площадок, числа слоев коммутации, информации о приоритетных слоях, ширины проводников и минимального расстояния между соседними проводниками для каждого коммутационного слоя [4].

Выходной информацией о топологии для системы являются координаты левого нижнего угла каждого элемента схемы и представление о проведенных соединениях в символьном виде. Данные о соединениях получают последовательно для каждого обобщенного фрагмента и записываются через интерфейс прикладных программ в базу данных системы автоматизированного проектирования в том формате, который используется для их описания в конкретной системе автоматизированного проектирования больших интегральных схем.

Трассировка соединений является, как правило, заключительным этапом конструкторского проектирования электрические вычислительные аппараты и состоит в определении линий, соединяющих эквипотенциальные контакты элементов, и компонентов, составляющих проектируемое устройство. Задача трассировки – одна из наиболее трудоемких в общей проблеме автоматизации проектирования электрические вычислительные аппараты. С математической точки зрения трассировка – сложнейшая задача выбора из огромного числа вариантов оптимального решения. Задача трассировки может решиться при использовании четко поставленных исходных данных, но в этих данных может иметься небольшая погрешность, основанная на влиянии какого-нибудь фактора извне. Поэтому, чтобы получить точные результаты трассировки с учетом влияния каких-либо факторов для решения, начали применять нечеткие генетические алгоритмы.

История эволюционных вычислений началась с разработки ряда различных независимых моделей. Основными из них были генетические алгоритмы и класси-

фикационные системы Голланда (Holland), опубликованные в начале 60-х годов и получившие всеобщее признание после выхода в свет книги, ставшей классикой в этой области, – "Адаптация в естественных и искусственных системах".

Для работы генетических алгоритмов выбирают множество натуральных параметров оптимизационной проблемы и кодируют их в последовательность конечной длины в некотором алфавите. Они работают до тех пор, пока не будет выполнено заданное число генераций (итераций алгоритма) или на некоторой генерации будет получено решение определенного качества, или, когда найден локальный оптимум, возникновение преждевременной сходимости и алгоритм не может найти выход из этого состояния. Генетический алгоритм, как правило, анализирует различные области пространства решений одновременно и поэтому они более приспособлены к нахождению новых областей с лучшими значениями целевой функции [2].

Мягкие вычисления – сложная компьютерная методология, основанная на нечеткой логике, генетических вычислениях, нейрокомпьютинге и вероятностных вычислениях. Составные части не конкурируют, но создают эффект взаимного усиления для достижения робастности, низкой цены решения, повышения эффективности приложений. Четыре составные части мягких вычислений включают в себя:

- ◆ нечеткую логику (приближенные вычисления, грануляция информации, вычисление на словах);
- ◆ нейрокомпьютинг (обучение, адаптация, классификация, системное моделирование и идентификация);
- ◆ генетические вычисления (синтез, настройка и оптимизация с помощью систематизированного случайного поиска и эволюции);
- ◆ вероятностные вычисления (управление неопределенностью, сети доверия, хаотические системы, предсказание).

Математический аппарат теории нечетких систем используется в данном случае для кодирования, подбора оптимальных параметров генетических алгоритмов, значений вероятности генетических операторов, выбора функции пригодности, создания нечетких генетических операторов. Есть два преимущества нечеткого кодирования:

- 1) кодовые последовательности могут быть неоднородными и ориентироваться на отдельные многообещающие области поиска, что позволит сократить область поиска и соответственно вычислительные затраты, в закодированную последовательность может быть неявным образом включена функция пригодности;
- 2) нечеткое кодирование позволяет выполнять, так называемое, слабое кодирование оптимизируемых структур.

При решении задач оптимизации функций в непрерывных областях поиска, большое значение имеет обеспечение высокой точности получаемых решений, для этого предложены генетические операторы, адаптированные для хромосом с вещественным кодированием. Такие операторы позволяют выполнять локальную настройку решений.

Настройка функций принадлежности с помощью нейронной сети или генетическим алгоритмом устраняет принципиальную слабость теории нечетких систем – субъективность функций принадлежности. Применение нейронных сетей к настройке функций принадлежности позволяет рассматривать окончательную форму функции как аппроксимацию обучающей выборки. Такую же задачу можно решить с помощью генетического алгоритма как метода стохастической оптимизации. Генетической нечеткой системой называют нечеткую систему, функции принадлежности и база правил которой спроектированы с помощью генетического алгоритма [4].

Применить генетический алгоритм – значит выбрать единицу кодирования, т.е. хромосому; уточнить эволюционные операторы рекомбинации, мутации и селекции; сформировать функцию адаптивности (fitness function, performance index). В настоящее время генетический алгоритм используют либо для настройки функций принадлежности (базы данных), либо для формирования базы правил, либо для одновременного формирования и функций принадлежности, и правил (базы знаний). В соответствии с объектами оптимизации выбирают единицу кодирования – хромосому. Для настройки функций принадлежности за хромосому выбирают одно правило (Мичиганский подход), для настройки базы правил за хромосому выбирают вариант базы правил (Питсбургский подход, подход итеративного обучения правил). В соответствии с кодированием уточняются правила генерации новых хромосом. Функция адаптивности представляет собой механизм нечеткого вывода, который для каждого варианта базы правил строит либо управление для нечеткого контроллера, либо экспертное заключение для диагностической экспертизы. Все нечеткие правила вносят вклад в окончательный результат, т.е. правила сотрудничают. Но при отборе правил (хромосом) для генерации новых правил генетического алгоритма накапливаются правила, внесшие максимальный вклад в общий результат, т.е. хромосомы конкурируют. Говорят о проблеме «конкуренции и кооперации» в генетических нечетких системах (a competition vs. a cooperation). Решение проблемы в каждом конкретном случае системы с генетической настройкой строится эвристически, например, при подходе итеративного обучения правил используют два этапа оптимизации. На первом шаге правила конкурируют за право войти в базу правил, а на втором взаимодействуют при формировании общего результата [2].

С увеличением сложности изделий растет и число уровней абстракции, на которых приходится проектировать эти системы. На каждом уровне абстракции имеется свое понятие системы и элементов системы, в базисе которых ведется проектирование. То, что было системой на предыдущем уровне, на следующем, более высоком уровне, становится элементом. Существуют две концепции проектирования любого изделия: сверху вниз и снизу вверх. Практическая реализация каждой отдельной концепции невозможна. В первом случае это обусловлено тем, что для каждого изделия необходимо разрабатывать индивидуальный реальный элементный базис низшего уровня абстракции, что противоречит требованиям времени проектирования, возможности серийного производства и разумной стоимости проектируемого изделия. Во втором случае нереализуемость концепции определяется отсутствием обоснованных требований к функциональному составу и параметрам структурных компонентов любого более высокого уровня. Поэтому на практике используется разумное сочетание этих концепций. Довольно распространенным представлением процесса проектирования является следующее: на каждом уровне представления проекта решаются две задачи – синтез и анализ, причем под синтезом понимается получение структуры системы в виде соединенных между собой структурных компонентов данного уровня, а под анализом – составление и решение математической модели этой структуры. Например, при схемотехническом проектировании структура системы представляется списком цепей, соединяющих транзисторы, резисторы, емкости и другие компоненты, а анализ – составлением математической модели схемы в виде системы обыкновенных дифференциальных уравнений и ее численного решения, которое и позволяет рассчитать функциональные параметры схемы. Ошибочность данного представления состоит в следующем: при недостаточно обоснованном выборе параметров математических моделей структурных компонентов может быть забракована хорошая структура системы. Чтобы избежать подобную ситуацию, необходимо в обязательном порядке

выполнить процедуру параметрической оптимизации. Но и этого недостаточно для оценки правильности выбора структуры. Чтобы параметры выбранной структуры системы удовлетворяли предъявляемым требованиям при разбросе параметров структурных компонентов, необходимо проведение статистического анализа. Учитывая, что процедуры параметрической оптимизации и статистического анализа реализуются соответствующим образом организованным многократным решением математических моделей, теоретически идеальное проектирование требует колоссальных вычислительных ресурсов, несмотря на огромный прогресс в разработке современных средств автоматизации проектирования.

При проектировании сложных технических систем (в т.ч. сверхбольших интегральных схем и других изделий микроэлектроники) используется блочно-иерархический подход. При блочно-иерархическом подходе процесс проектирования разделяется на уровни (этапы). На высшем уровне используется наименее детализированное представление, отражающее только самые общие черты проектируемой системы. На каждом низшем последовательном уровне проектирования степень подробности рассмотрения возрастает, при этом система рассматривается не в целом, а отдельными блоками, как правило, функционально законченными. Такой подход позволяет на каждом этапе формулировать и решать задачи приемлемой сложности, поддающиеся пониманию одним человеком.

Основное достоинство блочно-иерархического подхода состоит в том, что сложная задача большой размерности разбивается на последовательно решаемые задачи меньшей размерности. При блочно-иерархическом подходе на каждом уровне абстракции имеются свои представления о системе и элементах. То, что на одном уровне называлось элементом, становится системой на следующем уровне. При проектировании сложных систем используются так называемые нисходящее или восходящее проектирование или их комбинации. Суть нисходящего – от общих требований к системе к технологическому процессу изготовления, а при восходящем – наоборот.

Разнообразие конструктивно-технологических методов создания сверхбольших интегральных схем обусловлено стремлением не только улучшить их технико-экономические показатели, но и достичь общих целей: минимизировать длительность процесса проектирования, обеспечить проектирование сверхбольших интегральных схем высокой сложности, повысить качество проектирования (главным образом, безошибочность) [3].

Главная трудность с возможностью построения вычислительных систем, основанных на принципах естественного отбора и применением этих систем в прикладных задачах, состоит в том, что природные системы достаточно хаотичны, а все наши действия, фактически, носят четкую направленность. В настоящее время компьютер используется как инструмент для решения определенных задач, которые нами самими и формулируются.

При использовании нечетких генетических алгоритмов, при нахождении оптимального пути трассировки, не обязательно теперь задание конкретные параметры ширины проводников и минимальные расстояния между ними, примерные значения и факторы, влияющие на них.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Гладков Л.А., Курейчик В.В., Курейчик В.М.* Генетические алгоритмы / Учебное пособие. Под ред. В.М. Курейчика. – Ростов-на-Дону: ООО «Ростиздат», 2004. – 400 с.
2. *Казёнов Г.Г.* Основы проектирования интегральных схем и систем – М.: БИНОМ. Лаборатория знаний, 2005. – 295 с.

3. *Dubois D., Prade H. Fuzzy Real Algebra: Some Results / Dubois D., Prade H. /Fuzzy Sets and Systems. 1979. V 2. P. 327-348. Dubois D., Prade H. Fuzzy Sets and Systems. – N.Y.: Academic Press, 1980.*
4. *Ярушикина Н. Г. Основы теории нечетких и гибридных систем. – М.: Финансы и статистика, 2004. – 320 с.*

Лисовцова Анастасия Евгеньевна

Технологический институт федерального государственного образовательного учреждения высшего профессионального образования «Южный федеральный университет» в г. Таганроге.

E-mail: russiafluffy@bk.ru.

347928, г. Таганрог, пер. Некрасовский, 44.

Тел.: 8(8634)371-651.

Кафедра систем автоматизированного проектирования; инженер.

Lisovcova Anastasiy Evgenievna

Taganrog Institute of Technology – Federal State-Owned Educational Establishment of Higher Vocational Education “Southern Federal University”.

E-mail: russiafluffy@bk.ru.

44, Nekrasovskiy, Taganrog, 347928, Russia.

Phone: 8(8634)371-651.

Department of Computer Aided Design; engineer.

УДК 681.3

Е.Е. Курносова, А.А. Полупанов

МЕТОДЫ ПОВЫШЕНИЯ КАЧЕСТВА РЕШЕНИЙ В ЭВОЛЮЦИОННО-ГЕНЕТИЧЕСКИХ АЛГОРИТМАХ*

В работе рассматривается набор новых эвристик отбора, вероятности их выбора в операторах мутации и принципах формирования начального поколения, глубина локального поиска, размер микро- и макро- мутаций, а также характер расположения в хромосоме мутируемых генов.

Эволюционно-генетический алгоритм; генетические операторы; локальное улучшение; принудительная микромутация; глубина локального поиска; эвристики отбора.

E.E. Kurnosova, A.A. Polupanov

METHODS OF SOLUTIONS QUANTITY INCREASING IN EVOLUTIONARY-GENETIC ALGORITHMS

In this work the new selection heuristics set, probability of their choose in mutation and principles of start generation, depth of local search, micro- and macro- mutation size, and also location of mutated genes in chromosome are considered.

Evolutionary-genetic algorithm; genetic operators; local improvement; compulsory mutation; local search depth; selection heuristics.

Введение. В последние годы непрерывно разрабатываются новые методы поиска оптимальных решений для задач конструкторского уровня проектирования САПР [1]. В этой связи, широкое распространение получили методы эволюционно-генетического моделирования, заимствующие основные принципы своей рабо-

* Работа выполнена при поддержке: РФФИ (грант № 07-01-00174), г/б № 2.1.2.1652.