

УДК 681.518

## ПОСТРОЕНИЕ ИДЕНТИФИЦИРУЮЩИХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ЦИФРОВЫХ СХЕМ С ИСПОЛЬЗОВАНИЕМ СТРАТЕГИИ СИМУЛЯЦИИ ОТЖИГА

<sup>1</sup>Д.Е. Иванов, <sup>2</sup>Р. Зуауи

<sup>1</sup>Институт прикладной математики и механики НАН Украины,

<sup>2</sup>Донецкий Национальный технический университет

В статье предлагается применение стратегии симуляции отжига к задачам построения идентифицирующих последовательностей цифровых схем. Близость идей стратегии к генетическим алгоритмам позволяет эффективно использовать имеющиеся наработки для быстрого построения алгоритмов идентификации, основанных на данном подходе.

### **Введение.**

Задачи построения входных тестовых последовательностей с различными свойствами являются центральными в процессе разработки цифровых устройств. Процедура тестирования по различным оценкам занимает до 80% стоимости всего дизайна.

Для решения такого типа задач используются три основных подхода. Первый подход разрабатывает структурные методы построения входных последовательностей [1]. Он успешно применяется для схем малой и средней размерности. Однако при росте сложности дизайна деревья обходов становятся очень большими, что вызывает переполнения памяти инструментальной ЭВМ. Второй подход основан на символьных преобразованиях выражений, реализующих схему [2] и имеет те же ограничения. Третий подход появился как альтернатива для обработки больших схем. Он основан на том, что необходимая для поиска решения информация получается на основе результатов моделирования работы схем. В зависимости от целей задачи возможно использование исправного и с неисправностями моделирования. Возможность решения задачи для схем большой размерности объясняется тем, что задача моделирования является более простой и для таких схем предложены быстрые алгоритмы её решающие.

Наиболее часто в третьем подходе авторы используют генетические алгоритмы [3-4]. Это объясняется тем, что они являются хорошо изученными и на их основе предложено решение большого количества различных задач. Авторы также применяли генетические алгоритмы в задачах построения входных идентифицирующих последовательностей [5-6].

Однако вне внимания исследователей остаётся ещё одна оптимизирующая эволюционная стратегия «симуляции отжига» (СО). Она впервые

Цель данной работы состоит в построении алгоритмов генерации входных последовательностей цифровых схем на основе стратегии СО с использованием наработок в области ГА, а также сравнение их поисковых качеств.

### **Стратегия симуляции отжига.**

Формально стратегия симуляции отжига относится к вероятностным полиномиальным алгоритмам для решения  $NP$ -полных задач. Введём необходимые определения.

*Конфигурацией*  $K_i$  шага итерации  $i$  называется произвольное потенциальное решение (точка в пространстве решений). Конфигурация в СО аналогична особи в ГА. При этом также встает задача эффективного кодирования точек пространства поиска.

Каждая конфигурация ассоциируется с *функцией стоимости*  $C_i = C(K_i)$ . Эта функция также показывает насколько хорошо конфигурация решает поставленную задачу. Функция стоимости соответствует фитнес-функции в ГА. Сложность вычисления данной функции определит сложность всего алгоритма.

*Окружением* конфигурации  $K_i$  называется множество возможных конфигураций, которые получаются из неё некоторым *возмущением* (обычно небольшим). Операция возмущения в алгоритме симуляции отжига близка по смыслу с мутацией в генетическом алгоритме.

Температура  $T$  в процессе работы алгоритма постепенно уменьшается от некоторого начального значения  $T_0$  до температуры остывания  $T_{кон}$ .

Описание алгоритма симуляции отжига приведено ниже.

- 1) В начале работы формируется начальная конфигурация  $K_0$  и происходит вычисление её функции оценки  $C_0 = C(K_0)$ . Начальная конфигурация принимается в качестве текущей конфигурации  $K_i = K_0$ . Определяется текущая начальная температура  $T_i = T_0$ . Далее итеративно вплоть до выполнения условия остановки выполняются следующие шаги.
- 2) Строится окружение текущей конфигурации  $K_i$  с помощью операции возмущения.
- 3) Для всех конфигураций, входящих в окружение, вычисляется оценка и выбирается наилучшая конфигурация  $K_{ном}$ .
- 4) Вычисляется изменение функции стоимости:

$$\Delta C_i = C(K_{\text{пром}}) - C(K_i).$$

- 5) Произведённые возмущением изменения либо принимаются, либо отвергаются: если изменение функции стоимости отрицательное, то промежуточная конфигурация заменяет текущую. В противном случае такая замена происходит на основании распределения Больцмана:

$$K_{i+1} = \begin{cases} K_{\text{пром}}, & \text{если } \Delta C_i < 0; \\ K_{\text{пром}} & \text{с вероятностью } P = \exp(-\Delta C_i / kT_i), \text{ если } \Delta C_i > 0; \end{cases}$$

где  $k$  - эвристическая константа.

Такой способ приёма изменений конфигурации позволяет при большей температуре чаще принимать плохие решения, чем отбрасывать их. При снижении температуры также снижается вероятность принятия худших решений. В общем случае распределение может быть иным.

- б) Изменяется текущая температура:

$$T_{i+1} = \text{обновить}(T_i).$$

- 7) Переход к шагу 2.

### **Алгоритмы СО построения входных идентифицирующих последовательностей.**

Формальная постановка задач построения инициализирующих и верифицирующих эквивалентность последовательностей дана в [9]. Неформально цели алгоритмов можно сформулировать следующим образом. Алгоритм логической инициализации: построить входную последовательность, которая переводит в известное состояние максимальное число триггеров схемы. Алгоритм верификации: построить входную последовательность, которая различает поведение двух заданных схем. Для решения данных задач с помощью алгоритма СО определим его компоненты.

В качестве конфигурации принимается единичная входная последовательность, в которой число наборов заранее не известно, а длина вектора одного набора равна числу внешних входов схемы [6].

Функция оценки определяет тип строящейся входной последовательности. Для задачи логической инициализации схем она имеет вид:

$$C(s) = f(n_1, n_2, n_3) = (c_1 * n_1 + c_2 * n_2) * c_3^{n_3}, \quad (1)$$

где:  $n_1$  - отношение числа инициированных триггеров к их общему числу;  $n_2$  - активность схемы или число событий моделирования;  $c_1, c_2, c_3$  - нормализующие константы.

Функция оценки для задачи верификации эквивалентности схем:

$$C(s) = f(n_1, n_2, n_3) = n_1 + c_1 \cdot n_2 + c_2 \cdot n_3, \quad (2)$$

где:  $n_1$  - число различных значений на внешних выходах двух анализируемых схем;  $n_2$  - число различных псевдовыходов схем;  $n_3$  - число вентилей двух схем с различными значениями сигналов;  $c_1, c_2$  – нормализующие константы.

Для построения окружения мы используем три типа возмущений, которые соответствуют операциям мутации ГА построения тестов [6]:

- из последовательности удаляется случайный входной набор;
- в случайную позицию в последовательности добавляется случайный вектор;
- случайно изменяется столбец (строка) в конфигурации.

### Экспериментальные данные.

Предложенные алгоритмы СО построения входных последовательностей реализованы программно. Апробация проводилась на схемах из международного каталога ISCAS-89.

Эвристические параметры алгоритмов определялись экспериментальным путём. Алгоритм построения инициализирующих последовательностей:  $T_0=120$ ,  $T_{кон}=1$ , число конфигураций для одной температуры =100, распределение Больцмана, константа Больцмана  $k=0,01$ . Алгоритм верификации эквивалентности:  $T_0=120$ ,  $T_{кон}=1$ , число конфигураций для одной температуры =50, распределение Больцмана, константа Больцмана  $k=0,00001$ .

Результаты машинных экспериментов для некоторых больших схем приведены в табл.1. Для проверки эффективности алгоритма верификации эквивалентности мы строили «почти эквивалентные» схемы в соответствии с методикой в [8]. Для алгоритма верификации колонка «не различено» схем показывает число экспериментов, в которых функция оценки осталась равна нулю. Колонка «?» показывает число экспериментов, в которых функция оценки была больше нуля и

Табл.1. Числовые данные машинных экспериментов для схем ISCAS-89.

схема	число вентилей / триггеров	алгоритм инициализации			алгоритм верификации			
		инициал. триггеров	длина послед.	время, сек.	# экспериментов	различено схем	не различено	?
s3271	1731 / 116	116	11	0	25	25	-	-
s3330	2037 / 132	132	5	0	25	22	2	1
s3384	1940 / 183	187	8	0	25	25	-	-
s4863	2514 / 104	104	12	0	25	25	-	-
s5378	3045 / 179	179	110	40	25	21	-	4
s6669	3460 / 239	239	7	2	25	25	-	-

решение, возможно, будет найдено, если увеличить глубину поиска.

Соответствующие цифровые данные ГА для обоих типов машинных экспериментов приведены в [7-8]. Сравнение числовых данных для алгоритмов СО и ГА показывает, что они очень близки и в оптимизационном смысле оба типа алгоритмов являются практически эквивалентными.

### **Заключение.**

В работе предложено использование стратегии СО к задачам построения входных идентифицирующих последовательностей цифровых схем. На её основе разработаны алгоритмы построения инициализирующих последовательностей и последовательностей для верификации эквивалентности. По результатам машинных экспериментов предложенные алгоритмы показали эффективность не ниже, чем аналогичные алгоритмы, основанные на генетическом подходе.

В качестве дальнейших исследований можно отметить применение стратегии СО к задаче построения тестов, а также разработку параллельных версий алгоритмов СО.

### **Литература**

1. Niermann T., Patel J.H. HITEC: A Test Generation Package for Sequential Circuits // Proc. European Design Automation Conf.- 1991.- p.214-218.
2. Sellers F.F., Hsiao M.Y., Bearnson L.W. Analysing errors with the boolean difference // IEEE Transactions on Computers.- 1967.- №5.- p.675-680.
3. Goldberg D.E., Genetic Algorithm in Search, Optimization, and Machine Learning.- Addison-Wesley.- 1989.
4. Ю.А. Скобцов. Основы эволюционных вычислений.- Донецк: ДонНТУ, 2008.- 326с.
5. Y.A. Skobtsov, D.E. Ivanov, V.Y. Skobtsov Evolutionary distributed test generation methods for digital circuits // Proc. of 8th International Workshop on Boolean Problems, September 18-19, 2008, Freiberg, Germany.- pp.213-218.
6. Skobtsov Y.A., El-Khatib, Ivanov D.E. Distributed Fault Simulation and Genetic Test Generation of Digital Circuits // Proceedings of IEEE East-West Design&Test Workshop(EWDT'06).-2006: Sochi.- p.89-94.
7. Д.Е. Иванов, Ю.А. Скобцов, А.И. Эль-Хатиб Построение инициализирующих последовательностей синхронных цифровых схем с помощью генетических алгоритмов.- Проблеми інформаційних технологій.-2007.-№1.-с.158-164.
8. Д.Е. Иванов Генетический подход проверки эквивалентности последовательностных схем // «Радіоелектроніка. Інформатика. Управління».- Запоріжжя, ЗНТУ.- 2009.- №1(20).- С.118-123.
9. Иванов Д.Е. Генетические алгоритмы построения идентифицирующих последовательностей для цифровых схем с памятью // Наукові праці Донецького національного технічного університету. Серія: “Обчислювальна техніка та автоматизація”. Випуск 14(129).-Донецьк: ДонНТУ. – 2008.- С.97-106

Получено 15.05.2009