

ДИСКРЕТНЫЕ ФУНКЦИИ И АВТОМАТЫ

DISCRETE FUNCTION AND AUTOMATONS

Научная статья

УДК 53.083

doi: 10.17223/19988605/60/12

Об одном подходе к приближенному синтезу дискретных устройств с памятью

Дмитрий Васильевич Сперанский

Российский университет транспорта (МИИТ), Москва, Россия, speranskiy.dv@gmail.com

Аннотация. Рассматриваются теоретические аспекты проблемы приближенного синтеза дискретных устройств с памятью, представленных моделью конечного автомата. Предложен подход к решению проблемы, состоящий из двух этапов. На первом этапе осуществляется подготовка предварительной информации о функционировании заданного автомата-прототипа. На втором этапе она используется для синтеза автомата, аппроксимирующего работу прототипа, осуществляемого с использованием предлагаемого генетического алгоритма.

Ключевые слова: дискретные устройства с памятью (ДУ); приближенный синтез ДУ; генетические алгоритмы

Для цитирования: Сперанский Д.В. Об одном подходе к приближенному синтезу дискретных устройств с памятью // Вестник Томского государственного университета. Управление, вычислительная техника и информатика. 2022. № 60. С. 111–119. doi: 10.17223/19988605/60/12

Original article

doi: 10.17223/19988605/60/12

On one approach to approximate synthesis of discrete devices with memory

Dmitriy V. Speranskiy

Russian University of Transport (MIIT), Moscow, Russian Federation, speranskiy.dv@gmail.com

Abstract. The problem of approximate synthesis of discrete memory devices (DM) is considered. In the last decade a similar problem for combinational circuits (CC) has been actively studied. Interest in them is caused by the possibility of obtaining compact and reliable devices at the expense of relaxing requirements for the accuracy of their functions. The resulting flexibility in operation can be used to reduce chip area, circuit delays, and increase fault tolerance. Changes in the devices result in mismatches with the outputs of the prototype circuit. In a number of areas they may be quite acceptable in practice. The problem of approximate synthesis of discrete memory devices in the proposed article was investigated in the formulation, which has not been previously considered, which explains the absence of relevant publications. In previously known publications on the approximate synthesis of automata-type systems, where the closeness of the obtained solution is understood as obtaining an approximate (in quality) solution to the optimal one of the problem for which the sought simpler device is created. In this case, the quality of the solution can be evaluated according to various criteria. The corresponding methods are not applicable to the problem we are considering, since they are based on the use of the specifics of such systems.

In the paper the object of research is any discrete memory device (DM) represented by a finite state machine model described by its transition-exit table (TET). This is the most general and widespread mathematical model of such DMs. For practical applications the following problem is of interest: for a given DM of interest it is required to synthesize an approximating device (ADM) that minimizes the number of mismatches (conflicts) with DM outputs

on an arbitrary input sequence and that should be less complex than the prototype DM. The proposed article is devoted to the approach of its solution.

This approach implies two stages. The first stage consists in obtaining vectors of characteristic features of outputs of a given automaton. Their obtaining is based on the use of the well-developed apparatus of the theory of sampling research. These vectors are formed by the TET of a given DM using the frequency principle. Their components correspond to DM outputs and are equal to 0 or 1, or to the symbol *. The values 0 or 1 indicate that they are with high probability almost stabilized at the corresponding DM output, while the symbol * indicates the absence of stabilization. The feature vectors are used at the second stage to form the initial generation for the operation of the GA proposed in the article.

The second stage consists in obtaining the TET of the sought device with memory, which is an approximate model of the given automaton (prototype). The search for such a table is reduced to a problem of combinatorial type and for its solution the use of so-called simple GA is proposed. It is known that they are very effective for solving problems of combinatorial type.

The article gives a generalized description of GA. The introduced target function in the form of a weighted sum of two summands is justified since the problem under consideration is multicriteria. The peculiarities of the implementation of the GA, in particular the execution of the recombination (crossing) operator, are given.

Keywords: discrete memory devices (DM); approximate DM synthesis; genetic algorithms

For citation: Speranskiy, D.V. (2022) On one approach to approximate synthesis of discrete devices with memory. *Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naja tehnika i informatika* – Tomsk State University Journal of Control and Computer Science. 60. pp. 111–119. doi: 10.17223/19988605/60/12

В последнее десятилетие проявляется большой интерес к проблеме приближенного синтеза комбинационных схем (КС) [1–4]. Решение этой проблемы позволяет получить компактные высокоскоростные и надежные устройства. Такой результат достигается за счет менее жестких требований к точности реализации схемой ее функций.

Ослабление требований к точности функционирования схемы, в результате которого возникают отклонения от выходов прототипа, в некоторых случаях оказывается вполне приемлемым на практике.

Появляющаяся в результате некоторая гибкость в функционировании устройства может быть использована с целью уменьшения площади кристалла, задержек в схеме, повышения отказоустойчивости и полноты обнаружения неисправностей (при тестировании). Это является весомым аргументом в пользу применения приближенного синтеза. Можно в качестве примера указать довольно широкий круг приложений, где некоторые пределы отклонений (рассогласований выходов) не являются критичными. К числу таких областей относятся беспроводные коммуникации, картография, графические объекты, аудио- и видеозаписи и т.д.

Названная проблема представляет интерес и для устройств с памятью, но публикации по их приближенному синтезу в точно такой же постановке, как и для КС, отсутствуют. В случае КС приближенный синтез состоит в построении более простой схемы по возможности с минимальным числом рассогласований по выходам с заданной КС. Однако имеется много публикаций, посвященных приближенному синтезу устройств автоматного типа, где такой синтез понимается как получение приближенного (по качеству) решения проблемы, для которого синтезируется более простое устройство. При этом под качеством понимается «близость» приближенного решения к оптимальному, которая может оцениваться по различным критериям (но не так, как в случае КС). В качестве примера назовем публикации [5, 6], посвященные приближенному синтезу систем автоматного типа, относящихся к классу специализированных систем оптимального управления. Соответствующие методы неприменимы к рассматриваемой нами задаче, поскольку они базируются на использовании специфики таких систем. Аналогичная ситуация имеет место и во многих иных публикациях, в которых синтезируемые устройства должны решать специальные задачи из разных областей.

Для решения задач приближенного синтеза КС разработаны различные подходы, основанные на использовании логической регрессии [3], реконфигурации КС и их логической оптимизации, вероятностных и эволюционных методах [2] и др. Возможно, что для решения аналогичной задачи для устройств с памятью полезным может оказаться использование некоторых идей из названных подходов.

Далее рассматривается задача приближенного синтеза любых дискретных устройств с памятью, функционирование которых описывается моделью конечного автомата [7]. Для ее решения в статье предлагается подход, предполагающий выполнение двух этапов. Первый этап состоит в получении векторов характеристических признаков выходов заданного автомата. Получение таких признаков будет основано на использовании аппарата теории выборочных исследований. Второй этап состоит в получении таблицы переходов-выходов (ТПВ) искомого устройства с памятью, являющегося аппроксимированной моделью заданного автомата (прототипа). Получение такой таблицы сводится к задаче комбинаторного типа, и для ее решения предлагается использование так называемого простого генетического алгоритма (ПГА), подробно описанного в ряде публикаций, например в [8]. Далее предполагается знакомство с его структурой, типовым набором стандартных операторов (репродукции, кроссинговера, мутации). Известно, что генетические алгоритмы (ГА) весьма эффективны для решения задач комбинаторного типа.

Методы приближенного синтеза должны быть применимы в практических приложениях к устройствам с достаточно большим числом входов. Поэтому в случае КС для описания их функционирования отказываются от таблиц истинности из-за большой их размерности, а вместо них используется «черный ящик» (например, как в [3]). Внутренняя структура черного ящика считается неизвестной. В процессе решения задачи на него можно подавать любые входные сигналы и наблюдать при этом выходные, т.е. получать любое необходимое множество пар слов «вход–выход» КС. При решении рассматриваемой ниже задачи будет использоваться ТПВ автомата-прототипа. Такая таблица выполняет роль аналога черного ящика.

1. Постановка задачи

Объектом исследования является дискретное устройство с памятью (ДУ), представленное моделью конечного автомата, описываемого его ТПВ.

В статье рассматриваются теоретические аспекты решения следующей задачи. Для заданного ДУ требуется синтезировать аппроксимирующее его работу устройство (АДУ), минимизирующее число рассогласований (конфликтов) с выходами ДУ на произвольной входной последовательности, которое должно быть менее сложным, чем ДУ-прототип.

Приведенная постановка задачи пока не является строгой, а носит содержательный характер. Для строгой ее постановки необходимо уточнить сформулированные требования. Первое из них интерпретируется далее как потенциальная возможность в ходе синтеза уменьшать число конфликтов. Условимся, что во втором требовании сложность АДУ будет полагаться равной числу состояний конечного автомата, являющегося его математической моделью, плюс сложность комбинационных схем (например, число входящих в них вентилях) для управления триггерами (памятью) и выходами АДУ.

Понятно, что идеальный случай – это задача синтеза минимального по сложности АДУ с минимальным числом конфликтов с выходами ДУ. Однако при естественных ограничениях на вычислительные ресурсы лучшая АДУ из полученных на предыдущих шагах поиска может не удовлетворять одновременно обоим требованиям.

Пользователю АДУ всегда желательно иметь несколько вариантов его схемной реализации, чтобы он имел возможность выбора с учетом и других критериев. Предлагаемый ниже подход такую возможность может обеспечить. Действительно, предлагаемое в процессе решения использование в нем ГА при повторных его запусках, а также применение нового множества генерируемых выборок будут приводить к новым схемам АДУ.

Отметим, что в рассматриваемой нами задаче приемлемость АДУ задается иначе, чем это принято для КС. Для КС она обычно задается в виде порогового значения вероятности появления конфликтов выходов с эталоном КС, но не оговаривается число конфликтов.

В случае КС многие названные выше подходы к решению задачи приближенного синтеза, как правило, начинают выполнять синтез с чистого листа. Они не пользуются никакой предварительно подготовленной информацией о характере функционирования ДУ-прототипа.

Соответствующие методы синтеза представляют собой итерационные процессы. При этом скорость их сходимости и общая трудоемкость существенно зависят от качества результатов первой итерации. По-видимому, ее качество можно улучшить, получив предварительно некоторую информацию об особенностях функционирования ДУ. При этом возникают естественные вопросы о «цене» получения такой информации и ее влиянии на скорость сходимости процесса. Заметим, что способы формирования упомянутой информации и возможные пути оценки затрат достаточно полно разработаны в теории выборочных исследований, которые подробно изложены, например, в монографии [9]. Поэтому здесь эти вопросы не рассматриваются.

2. Формирование вектора признаков выходов ДУ и начального поколения в ГА

Первый этап предлагаемого подхода к решению рассматриваемой задачи заключается в получении некоторых характеристических признаков выходов заданного ДУ. Опишем, как формируется такая информация. Известно [10], что задачи, решаемые с применением различных эволюционных алгоритмов, могут быть сведены к задачам классификации и кластеризации. Элементы множества V , для которого решаются эти задачи, таковы, что в них закодированы свойства реальных объектов и особенности предметной области. В конкретных задачах множество V есть вектор признаков этих объектов. Значения признаков могут быть получены на основе анализа совокупности отобранных для анализа выборок из генеральной совокупности (ГС) объектов. В нашем случае ГС представляет собой все множество элементов вида:

$$S_i \xrightarrow{a_j/b_j} S_k, \quad (1)$$

указывающих переход из состояния в состояние ДУ, а также вход / выход ДУ при его реализации. Каждая выборка трактуется как упорядоченная последовательность таких элементов, соответствующая траектории движения автомата (ДУ) при подаче заданной входной последовательности.

На основе информации, полученной из используемых выборок, далее выдвигаются гипотезы о структуре геномов элементов выборок и конструируется ГА. Один из возможных путей поиска значений компонентов в векторе признаков для каждого элемента выборки состоит в формальном анализе их состава.

Предполагается, что используемое множество репрезентативных выборок $V = \{V_1, V_2, \dots, V_d\}$ может генерироваться с помощью, например, генераторов случайных чисел, специальных алгоритмов или иными способами. Генерируемые случайные символы 0 и 1 используются затем в качестве входов заданного ДУ. Снимаемые при этом с его выходов данные используются в качестве компонентов векторов признаков, упомянутых выше. Полученная таким образом информация из выборок $V_1, V_2, \dots, V_d$ будет анализироваться, и результаты использоваться для формирования векторов признаков.

Условимся считать, что каждая выборка из множества $V = \{V_1, V_2, \dots, V_d\}$ состоит из последовательности элементов вида (1) фиксированной длины k . Значение этого параметра определяется методами теории выборочных исследований.

Излагаемое ниже будет поясняться на примере ДУ, задаваемого в виде ТПВ, для которого требуется синтезировать АДУ. Для простоты изложения рассмотрим автомат, заданный табл. 1, у которого множество состояний $S = \{s_1, s_1, \dots, s_9\}$, входной алфавит $X = \{a_1, a_2, a_3\}$, выходной – $Y = \{0, 1\}$ (автомат имеет один выходной канал). Если число выходных каналов ДУ больше одного, то значения 0 и 1 на каждом канале фиксируются и указываются для каждого канала в отдельности.

Вектор характеристических признаков выходов ДУ предназначен для формирования начального поколения АДУ, необходимого для работы ПГА. Его описание использует терминологию генетики, которую будем пояснять применительно к рассматриваемой задаче. Условимся считать, что ДУ, задаваемое ТПВ, имеет m входных и n выходных каналов. Используя оговоренную терминологию, в качестве особи будем считать состояние ДУ, которому соответствует битовая строка из m хромосом. Каждая хромосома содержит n (бит) генов и определяет значение на каждом выходном канале

в строке ТПВ ДУ, соответствующей каждому состоянию ДУ. Хромосома, содержащая в своих генах конкретные их значения, называется генотипом. Таким образом, ТПВ содержит в себе информацию о генотипах всех состояний ДУ. Так, в нашем примере каждому состоянию ДУ сопоставляется битовая строка из трех хромосом a_1, a_2, a_3 (по одной для каждого входного сигнала), каждая из которых содержит по одному биту (значению выхода ДУ при переходе в следующее состояние). То есть функционирование ДУ, заданное табл. 1, описывается девятью хромосомами, содержащими в общем 3×9 генов (бит).

Таблица 1

Автомат с множеством состояний S и входов X

$S \backslash X$	a_1	a_2	a_3
s_1	$s_2/1$	$s_2/0$	$s_5/0$
s_2	$s_1/0$	$s_4/0$	$s_4/1$
s_3	$s_2/1$	$s_2/0$	$s_5/1$
s_4	$s_3/0$	$s_2/1$	$s_2/1$
s_5	$s_6/1$	$s_4/0$	$s_3/0$
s_6	$s_8/0$	$s_9/1$	$s_6/1$
s_7	$s_6/1$	$s_2/0$	$s_8/1$
s_8	$s_4/1$	$s_4/1$	$s_7/0$
s_9	$s_7/0$	$s_9/1$	$s_7/1$

На втором этапе предлагаемого подхода поиск подходящих генотипов хромосом будет производиться с использованием ГА [8]. Начальное (нулевое) поколение для ГА (строится на основе ТПВ исходного автомата) представляет собой множество из N групп хромосом, где N – число состояний автомата. Каждая группа хромосом взаимно однозначно сопоставляется особи (состоянию ДУ). Гены этих хромосом есть значения ДУ на соответствующих выходных каналах автомата. За счет варьирования значений генов в хромосомах при выполнении ГА будет происходить переход от одного поколения особей к следующему.

Опишем процесс формирования начального поколения особей на основе ТПВ заданного автомата. С этой целью в ТПВ ДУ вычисляются частоты значений символов 0 (1) в каждом из генов z ($1 \leq z \leq n$) всех хромосом состояний ДУ. Так, в примере автомата, ТПВ которого представлена табл. 1, первый столбец ее выходов для всех состояний (соответствующих входному символу a_1) содержит пять значений 1 и четыре значения 0. Поэтому полагаем частоту выходного символа 0 (1) соответственно равной 4 (5) на этом фрагменте. Аналогично подсчитываются частоты появления символов 0 и 1 во всех остальных столбцах (для входных символов a_2 и a_3).

Обозначим через $g_{s,a_j}(0)$ ($g_{s,a_j}(1)$) число выходов 0 (1) в строке состояния s и столбца a_j табл. 1, где $s = 1, 2, \dots, 9$, $j = 1, 2, 3$. Здесь возможны следующие ситуации:

- 1) $g_{s,a_j}(0)$ значительно больше $g_{s,a_j}(1)$;
- 2) $g_{s,a_j}(1)$ значительно больше $g_{s,a_j}(0)$;
- 3) $g_{s,a_j}(0)$ и $g_{s,a_j}(1)$ отличаются не очень существенно.

В первом случае значение разряда (гена) y ($1 \leq y \leq n$) в хромосоме полагается равным 0, во втором случае – 1, в третьем случае замещается символом *. Результатом является структура геномов для вектора признаков, используемая в первой итерации ПГА в процессе синтеза АДУ.

Для формирования хромосом начального поколения состояний АДУ выполним следующие действия. Значения в позициях генов, являющихся 0 или 1 (стабилизированными в полученной структуре), полагаются равными этим значениям во всех хромосомах. Остальные гены в хромосоме каждой конкретной особи (состояния) замещаются случайным образом значениями 0 или 1.

В нашем примере структура начальных генотипов хромосом всех особей (состояний) АДУ будет иметь следующий вид:

a_1	a_2	a_3
*	*	1

Поэтому значения в гене a_3 в каждой хромосоме начального поколения полагаются равными 1, а гены a_1 и a_2 замещаются девятью случайными комбинациями символов 0 и 1.

Полученные описанным способом геномы далее используются в качестве исходной информации для ГА. Способ формирования состава хромосом начального поколения диктуется следующими содержательными соображениями. В каждом закрепленном гене хромосомы (он соответствует конкретному выходному каналу ДУ) содержится наиболее часто встречающееся при работе ДУ значение на этом выходе у всех его состояний. Интуитивно понятно, что поиск подходящих геномов хромосом, осуществляемый ГА, в этом случае с большей вероятностью потребует меньше итераций, чем при другом выборе начальных генотипов.

3. Обобщенное описание ГА

Изобретатель ГА J. Holland в [11] доказал shemata theorem, содержательный смысл которой состоит в следующем. С ростом числа итераций ГА вероятность приближения к экстремуму его целевой функции (фитнес-функции) возрастает. Это означает, что от поколения к поколению происходящая при выполнении ГА селекция постепенно стабилизирует значения группы разрядов в геноме. Кроме того, при продолжении выполнения итерационного процесса разрушения стабилизации не происходит. Эта теорема указывает на перспективность пути определения стабильных разрядов в начальном поколении ГА, что можно реализовать, например, так, как описано в предыдущем разделе статьи.

Перед началом работы предлагаемого ГА создается множество репрезентативных выборок V , и на его основе далее формируется информация, необходимая для его работы.

Начнем с описания целевой функции (ЦФ) ГА. Из постановки задачи понятно, что она представляет собой многокритериальную задачу оптимизации. Для построения ЦФ воспользуемся популярным направлением развития классических методов оптимизации, когда новая «общая» ЦФ строится из отдельных компонентов ЦФ в виде взвешенной суммы.

Вначале введем ЦФ для оценки отдельного варианта A схемы АДУ (для конкретной выборки $V_i \in V$):

$$G(A) = \sum_{i=1}^2 w_i f_i(A),$$

где веса $w_i \in [0,1]$ и $\sum_{i=1}^2 w_i = 1$. Выбор весов w_i для каждой «частной» ЦФ должен выполняться пользователем. В нашем случае $G(A)$ имеет два слагаемых, где $f_1(A)$ – число конфликтов выходов оцениваемого АДУ, $f_2(A)$ – его сложность. На каждой выборке множества $V = \{V_1, V_2, \dots, V_d\}$ выберем «лучшие» АДУ, образующие множество $\tilde{A} = \{A_1, A_2, \dots, A_d\}$. Каждая из них имеет наименьшее значение ЦФ $G(A_i)$ на выборке, на которой она получена. Определим ЦФ для ГА на всем множестве выборок V (мощности d) следующим образом:

$$F(V) = \frac{1}{d} \sum_{i=1}^d G(A_i).$$

Целью ГА является минимизация функции $F(V)$, являющейся средним арифметическим ЦФ «лучших» АДУ, полученных на всем множестве V .

ГА начинает работу с формирования начального поколения АДУ. Мощность K начальной популяции является важнейшим параметром ГА и обычно принадлежит диапазону [30; 200]. На разных этапах работы ГА оптимальное значение K может быть различным. Формирование начального поколения для решения рассматриваемой задачи осуществляется так, как это описано выше. Известно [8], что в целом эффективность ГА в значительной степени определяется структурой и качеством начальной популяции (поколения).

Следующий шаг в работе ГА состоит в выполнении репродукции, включающей создание родительских пар (пула) из особей текущего поколения. Из многих известных способов отбора на практи-

ке наиболее часто используется селективный отбор (с применением рулетки). Здесь родителями могут стать только особи, значения целевой функции которых не менее среднего значения по популяции при равной вероятности этих кандидатов. Такой путь, как правило, обеспечивает более быструю сходимость ГА и целесообразен для предлагаемого ГА.

Очередной шаг состоит в выполнении оператора рекомбинации (скрещивания, кроссинговера) двух особей. В нашей задаче, имея в виду ее автоматную специфику, целесообразно применение вводимой нами разновидности кроссинговера, отчасти аналогичного однородному [8]. Он радикально отличается от популярных типов одноточечного и многоточечного кроссинговеров. Здесь каждый ген потомка создается сложением по модулю двух хромосом родителей, и в каждой позиции, где стоит 1 после суммирования, производится обмен генами в хромосомах родителей. Это означает, что обмен генами происходит в позициях, где скрещиваемые автоматы имеют разные выходы.

На следующем шаге работы предлагаемого ГА полученные потомки подвергаются мутации (обычно с вероятностью из диапазона $[0,01; 0,05]$), состоящей в инвертировании мутируемого гена. Далее расширенная популяция сокращается до исходного размера, где остаются автоматы с лучшими значениями ЦФ, и проверяются критерии останова работы ГА. Он работает до тех пор, пока в процессе эволюции либо будет создано заданное количество поколений, либо на некоторой итерации будет достигнуто заданное качество, либо будет исчерпан выделенный объем вычислительных ресурсов.

Изменения генотипов моделей АДУ при работе ГА означают изменения значений выходов АДУ, порождающие новые автоматы. В них можно минимизировать число состояний, а при структурном синтезе АДУ по минимизированному автомату можно минимизировать и комбинационные схемы возбуждения триггеров, реализующих память автоматов и схемы управления выходами АДУ. Таким образом, в последовательно получаемых поколениях АДУ может уменьшаться число конфликтов по выходам с ДУ, а также и сложность схемной реализации.

ГА производит последовательную обработку по одному и тому же сценарию каждой выборки из множества V , и после обработки всех выборок будет получено значение ЦФ ГА на использованном множестве выборок V .

Перед началом работы ГА предусматривается случайная генерация множества репрезентативных выборок $V = \{V_1, V_2, \dots, V_d\}$. На каждой из них производится логическое моделирование работы заданного ДУ, стартующего из одного и того же начального состояния. Такое моделирование осуществляется один раз до выполнения ГА, и его выходы используются как эталон для сравнения с выходами синтезируемого АДУ. Это же состояние является начальным и для АДУ при его моделировании в процессе выполнения ГА. По этим данным описанным выше способом формируется начальное поколение АДУ, после чего начинается выполняться унифицированная часть ГА для обработки текущей выборки.

После получения очередного поколения АДУ необходимо вычислить для каждого АДУ его сложность и число конфликтов путем логического моделирования дискретного устройства, а также использования программы синтеза его схемы по автоматной модели. Заметим, что для этого имеются различные готовые пакеты программ.

После получения очередного поколения АДУ и значения ЦФ можно предусмотреть возможность выполнения очередной итерации ГА, рассчитывая на синтез более «качественного» АДУ, чем было получено на предыдущей итерации. Кроме того, с этой же целью можно реализовать синтез АДУ на новом множестве выборок.

Для создания ГА при решении различных задач ныне существуют специальные программные средства, существенно облегчающие решение этой проблемы. В качестве примера назовем программный DEAP-пакет для создания ГА на языке Python.

В качестве эксперимента на основе упрощенной программной версии ГА были проделаны расчеты для автомата, представленного табл. 1. К примеру, сложность АДУ оценивалась только числом его состояний. В результате варьирования длины и состава множества выборок, мощности начального поколения и других параметров был получен АДУ, у которого строки выходов ТПВ (хромосомы)

изменились в состоянии s_2 на $(0,1,1)$, в состоянии s_3 на $(1,0,0)$, в состоянии s_7 на $(1,0,0)$, в состоянии s_8 на $(1,0,0)$. В остальных строках ТПВ исходного ДУ выходы остались прежними. Указанные в скобках триады выходных сигналов и есть решения, полученные в результате упомянутых экспериментов. После минимизации этого автомата получен автомат с пятью состояниями (вместо девяти состояний в исходном; табл. 2). В этой таблице «новые» состояния представляют собой классы эквивалентных состояний исходного автомата: $I = \{1, 3, 8\}$, $II = \{2, 4\}$, $III = \{5, 7\}$, $IV = \{6\}$, $V = \{9\}$.

Таблица 2

Автомат с 5 состояниями

$\begin{matrix} X \\ S \end{matrix}$	a_1	a_2	a_3
I	II/1	II/1	III/0
II	I/0	II/1	II/1
III	IV/1	II/0	I/0
IV	I/0	V/1	IV/1
V	II/0	V/1	III/1

При проведении экспериментов на использованных выборках число конфликтов по выходам не превышало нескольких процентов от длины выборок. Понятно, что из этого примера делать какие-либо выводы о предлагаемом подходе некорректно. Для этого требуется накопление достаточного объема экспериментальных данных, но мотивацией полезности исследований предлагаемого подхода данный пример служить может.

Заключение

Известные ранее методы приближенного синтеза ДУ с памятью существенно базировались на специфике ДУ и именно тех проблем, для решения которых они разрабатывались. Эти методы оказываются неприменимыми к ДУ с памятью произвольной природы и назначения. Модель конечного автомата для них является наиболее общей, поскольку не предполагает при синтезе АДУ использования какой-либо специфики ДУ и решаемых ими проблем. В этом предлагаемый подход обладает в некотором смысле универсальностью.

Показано, что исследуемая задача может быть сведена к задаче комбинаторного типа. Имеющийся ныне опыт эволюционных вычислений доказал, что такие задачи эффективно решаются с использованием ГА. В статье в применении к рассматриваемой задаче приводится обобщенное описание одного из возможных вариантов ГА. Перед началом его работы предполагается получение предварительной информации о функционировании ДУ в виде векторов признаков его выходов. Далее они применяются для повышения быстродействия и эффективности ГА. Такие данные формируются на основе достаточно хорошо разработанной ныне теории выборочных исследований с использованием частотного принципа.

Таким образом, предложенный подход к решению задачи приближенного синтеза ДУ с памятью произвольной природы заполняет пока свободную нишу, поскольку публикации с описанием иных общих подходов отсутствуют. Получить объективную оценку предлагаемого подхода можно только при наличии достаточного объема экспериментальных данных, которых пока не имеется. Однако обоснованием целесообразности работы в этом направлении являются актуальность задачи, использование универсальной математической модели ДУ с памятью и приведенные обоснования принципов, заложенных при конструировании ГА.

Список источников

1. Shin D., Gupta S. Approximate logic synthesis for error tolerant // Design, Automation & Test in Europe Conference & Exhibition (DATE 2010). 2010. P. 957–960.
2. Sanches-Clemente A.J., Entrena L., Hrbacek R., Sekanina L. Error Mitigation Using Approximate Logic Circuits : a Comparison of Probabilistic and Evolution Approaches // IEEE Trans. on Reliability. 2016. V. 65, № 4. P. 1871–1883.

3. Stempkovsky A., Telpukhov D., Solovyev R. Synthesis of Approximate Combinational Circuits based on Logic Regression Approach // Proc. of IEEE East-West Design &Test Symposium (EWDTS). Varna, 2020. P. 64–68.
4. Gavrilov S.V., Telpukhov D. Automated Evolutionary Design of Fault-Tolerant Logic Circuits // Problemy razrabotki perspektivnyh makro- i nanoelectronnyh system (MES). 2019. № 1. P. 2–6.
5. Бортакровский А.С., Коновалова А.А. Синтез оптимальных дискретных систем автоматного типа при мгновенных многократных переключениях // Известия РАН. Теория и системы управления. 2014. № 5. С. 38–70.
6. Немыченков Г.И. Приближенный синтез оптимальных систем автоматного типа // Труды МАИ. 2016. Вып. 89. С. 1–14.
7. Gill A. Introduction to the theory of finite-state machine. Mc Graw-Hill company, 1962. 207 p.
8. Скобцов Ю.А., Сперанский Д.В. Эволюционные вычисления : учеб. пособие. М.: Нац. Открытый Ун-т «ИНТУИТ», 2015. 326 с.
9. Cochran W. Sampling techniques. New York ; London : John Wiley and Sons, Inc., 1963. 413 p.
10. Королев Л.Н. Эволюционные вычисления, нейросети, генетические алгоритмы // Фундаментальная и прикладная математика. 2009. Т. 15, № 3. С. 119–133.
11. Holland J. Adaptation in Natural and Artificial Systems. Cambridge : MIT Press, 1992. 211 p.

References

1. Shin, D. & Gupta, S. (2010) Approximate logic synthesis for error tolerant. *Design, Automation &Test in Europe Conference & Exhibiton (DATE 2010)*. pp. 957–960. DOI: 10.1109/DATE.2010.5456913
2. Sanches-Clemente, A.J., Entrena, L., Hrbacek, R. & Sekanina, L. (2016) Error Mitigation Using Approximate Logic Circuits : A Comparison of Probabilistic and Evolution Approaches. *IEEE Trans. on Reliability*. 65(4). pp. 1871–1883. DOI: 10.1109/TR.2016.2604918
3. Stempkovsky, A., Telpukhov, D. & Solovyev, R. (2020) Synthesis of Approximate Combinational Circuits based on Logic Regression Approach. *Proceedings of IEEE East-West Design &Test Symposium (EWDTS)*. Varna. pp. 64–68.
4. Gavrilov, S.V. & Telpukhov, D. (2019) Automated Evolutionary Design of Fault-Tolerant Logic Circuits. *Problemy razrabotki perspektivnyh makro- i nanoelectronnyh system (MES) – Problemy razrabotki perspektivnyh makro- i nanoelectronnyh system (MES)*. 1. pp. 2–6.
5. Bortakovsky, A.S. & Konovalova, A.A. (2014) Synthesis of optimal discrete automata-type systems for instantaneous multiple switching. *Izvestiya RAN. Teoriya I sistemy upravleniya*. 5. pp. 38–70. DOI: 10.7868/S0002338814050047
6. Nemychenkov, G.I. (2016) Approximate synthesis of optimal automata-type systems. *Trudy MAI*. 89. pp. 1–16. (in Russia).
7. Gill, A. (1962) *Introduction to the theory of finite-state machine*. Mc GRAW-Hill company.
8. Skobtsov, Yu.A. & Speranskiy, D.V. (2015) *Evolutsionnye vychisleniya* [Evolutionary Computations]. Moscow: National Open University “INTUIT”.
9. Cochran, W. (1963) *Sampling Techniques*. New York; London: John Wiley and Sons.
10. Korolev, L.N. (2009) Evolutionary Computing, Neural Networks, Genetic Algorithms. *Fundamental'naya i prikladnaya matematika – Fundamental and Applied Mathematics*. 15(3). pp. 119–133.
11. Holland, J. (1992) *Adaptation in Natural and Artificial Systems*. Cambridge: MIT Press.

Информация об авторе:

Сперанский Дмитрий Васильевич – профессор, доктор технических наук, профессор кафедры систем управления транспортной инфраструктурой Российского университета транспорта (МИИТ) (Москва, Россия). E-mail: speranskiy.dv@gmail.com

Автор заявляет об отсутствии конфликта интересов.

Information about the author:

Speranskiy Dmitriy V. (Doctor of Technical Sciences, Professor of the Department of Transportation Infrastructure Management Systems of Russian University of Transport (MIIT), Moscow, Russian Federation). E-mail: speranskiy.dv@gmail.com

The author declares no conflicts of interests.

Поступила в редакцию 24.01.2022; принята к публикации 30.08.2022

Received 24.01.2022; accepted for publication 30.08.2022