

ПОСТРОЕНИЕ АВТОМАТОВ УПРАВЛЕНИЯ ОБЪЕКТАМИ СО СЛОЖНЫМ ПОВЕДЕНИЕМ ПО ТЕСТАМ С УЧЕТОМ НЕПРЕРЫВНЫХ ВОЗДЕЙСТВИЙ

И.П. Бужинский

*студент кафедры «Компьютерные технологии» НИУ ИТМО,
buzhinsky@rain.ifmo.ru*

В.И. Ульянцев

*студент кафедры «Компьютерные технологии» НИУ ИТМО,
ulyantsev@rain.ifmo.ru*

Аннотация: Предлагается улучшение метода автоматизированного построения конечных управляющих автоматов. Рассматриваемые автоматные модели характеризуются наличием непрерывных (вещественных) выходных воздействий. Улучшение опробовано на задаче построения автомата управления моделью беспилотного самолета.

Введение

Автоматное программирование [1] — парадигма, в рамках которой программы проектируются в виде систем автоматизированных объектов управления, каждый из которых состоит из объекта управления и управляющего конечного автомата. Автоматное программирование применяется для задач управления объектами со сложным поведением — объектами, которые могут демонстрировать различное поведение при одинаковых входных воздействиях. В настоящем исследовании рассматривается задача построения автоматов управления объектами с непрерывными (вещественными) управляющими параметрами по обучающим примерам, или тестам.

В работе [2] для создания автомата с дискретными выходными воздействиями, управляющего моделью беспилотного самолета, был использован генетический алгоритм. При этом функция приспособленности автомата вычислялась на основе моделирования в авиасимуляторе, а полный цикл построения автомата занимал месяц работы персонального компьютера. В работе [3] был также применен генетический алгоритм, но функция приспособленности автомата вычислялась на основе тестов, что позволило сократить время построения автомата до нескольких часов. Другой отличительной особенностью работы [3] является использование непрерывных выходных воздействий. Метод построения автоматов, предложенный в [3], был позже ускорен [4] за счет использования муравьиного алгоритма [5].

Целью настоящего исследования является качественное улучшение метода построения автоматов, предложенного в работе [3]. Предлагается новый способ представления автоматов, позволяющий использовать для задания

функции выходов вещественные переменные — произвольные функции от параметров объекта управления.

Постановка задачи

Рассматриваемая задача заключается в построении конечного управляющего автомата по набору из N тестов, которые описывают желаемое поведение объекта управления, обладающего сложным поведением. В качестве объекта управления в настоящей работе, как и в работах [2–4], рассматривается модель беспилотного самолета. Для моделирования работы автоматов используется авиасимулятор *FlightGear* [6].

Тест состоит из последовательности P -элементных *кортежей входных воздействий* $in[i] = (in[i][t])_{t=1}^{len[i]}$ и последовательности C -элементных *кортежей выходных воздействий* $out[i] = (out[i][t])_{t=1}^{len[i]}$, где $i=1 \dots N$ — номер теста, а $len[i]$ — его длина. Входные воздействия описывают состояние объекта управления в различные моменты времени (интервал между ними в настоящей работе равен 0,1 с). По кортежам входных воздействий вычисляются значения булевых переменных — *предикатов*. Кортежи выходных воздействий представляют собой «эталонные» значения управляющих параметров — положений органов управления самолета. Будем считать, что значения всех управляющих параметров можно задать вещественными числами. Пример теста, записанного с помощью авиасимулятора *FlightGear*, приведен в Таблице 1.

Воздействие	Комментарий	$t=1$...	$t=10$...	$t=235$
$in[i][t][1]$	Угол крена (°)	−0,076	...	−0,076	...	−0,076
$in[i][t][2]$	Угол курса (°)	198,03	...	198,11	...	205,64
$in[i][t][3]$	Скорость (узлы)	251,42	...	252,29	...	289,40
$out[i][t][1]$	Положение элеронов (число от −1 до 1)	0,000	...	0,032	...	−0,003
$out[i][t][2]$	Положение руля высоты (число от −1 до 1)	−0,035	...	−0,039	...	−0,011

Таблица 1. Пример теста ($P=3$, $C=2$, $len[i]=235$)

Базовые положения исследования

Для построения автомата на основе поисковой оптимизации используется незначительно модифицированная *функция приспособленности* (ФП) из работы [3], отражающая близость поведения автомата к поведению, показанному в тестах:

$$f = 1 - \sqrt{\frac{1}{N} \sum_{i=1}^N \frac{1}{len[i]} \sum_{t=1}^{len[i]} \frac{1}{C} \sum_{j=1}^C \left(\frac{ans[i][t][j] - out[i][t][j]}{c_j^{\max} - c_j^{\min}} \right)^2} - P_{\tau}$$

Здесь c_j^{\min} и c_j^{\max} — границы значений j -го управляющего параметра, $\text{ans}[i]$ — последовательность значений управляющих параметров, выработанная автоматом в ответ на $\text{in}[i]$, а P_τ — дополнительный штраф, возрастающий с числом изменений состояния автомата в процессе его работы на тестах. Особью алгоритма поисковой оптимизации является «каркас» автомата — автомат, выходные воздействия которого не определены. Для их определения используется алгоритм, максимизирующий ФП на заданном «каркасе».

В работе [3] для представления автомата использовались полные таблицы переходов. Каждому переходу был сопоставлен кортеж выходных воздействий. За такт работы автомата совершалось несколько переходов, при этом кортежи выходных воздействий на всех выполнившихся переходах суммировались. В настоящей работе предлагается совместно с методом построения автоматов, предложенным в [3], использовать новый способ представления автоматов, имеющих непрерывные выходные воздействия.

Для представления функции переходов используются сокращенные таблицы [2]: в каждом состоянии автомата хранится маска значимости предикатов (Таблица 2), а событиями на переходах автомата являются всевозможные комбинации значений предикатов, помеченных единицей в маске значимости. На каждом такте автомат совершает только один переход, что упрощает как его визуализацию, так и понимание роли отдельных его состояний.

Предикат	Значение маски
p_1	1
p_2	0
p_3	1

Таблица 2. Пример маски значимости предикатов для одного из состояний автомата

Ключевым компонентом для представления функции выходов являются *вещественные переменные*, которые, как и предикаты, представляют собой функции от кортежей входных воздействий. Для каждого управляющего параметра $j = 1 \dots C$ задается набор $\{v_{j,i}\}_{i=1}^{n_j}$ из n_j вещественных переменных. В каждом состоянии автомата хранится маска значимости вещественных переменных (Таблица 3): только помеченные единицей в маске переменные влияют на формирование соответствующих им выходных воздействий. Все маски, используемые для представления функций переходов и выходов, являются частью особи алгоритма поисковой оптимизации.

Переменная	Значение маски	Переменная	Значение маски
$v_{1,1}$	1	$v_{2,1}$	0
$v_{1,2}$	1	$v_{2,2}$	1
$v_{1,3}$	0	$v_{2,3}$	1

Таблица 3. Пример маски значимости вещественных переменных для одного из состояний автомата ($C = 2$, $n_1 = n_2 = 3$)

Пусть s — текущее состояние автомата. Тогда изменение Δu_j выходного воздействия для j -го управляющего параметра по отношению к предыдущему такту работы автомата вычисляется как линейная комбинация значений вещественных переменных:

$$\Delta u_j = \sum_{i=1}^{n_j} r_{s,i,j} v_{j,i},$$

где $r_{s,i,j}$ — коэффициенты, вычисляемые алгоритмом расстановки выходных воздействий [3]. Эти коэффициенты заведомо равны нулю для незначимых в состоянии s переменных.

Пример автомата с предлагаемым способом представления приведен на Рис. 1. В каждом состоянии автомата значимыми являются один предикат и две переменные, при этом всего заданы три предиката p_1, p_2, p_3 и три переменные v_1, v_2, v_3 для единственного управляющего параметра u .

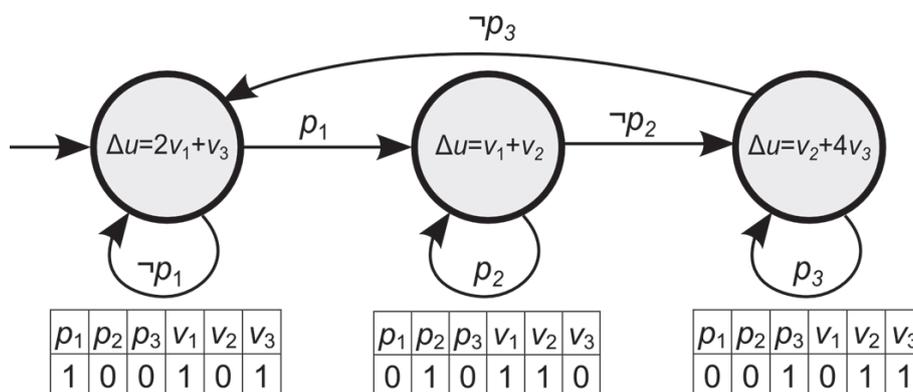


Рис. 1. Пример автомата с предлагаемым способом представления. Внутри состояний приведены законы управления параметром u , внизу — маски значимости переменных и предикатов, соответствующие различным состояниям

Экспериментальное исследование

Сравнение предлагаемого метода представления автоматов с ранее разработанным проведено на трех наборах тестов, описывающих выполнение фигур пилотажа: «мертвой петли», «бочки» и разворота в горизонтальной плоскости на 180° .

Для использования способа представления автоматов из работы [3] требуется задать набор предикатов, а для предлагаемого способа помимо этого необходимо определить набор вещественных переменных. Подбор набора переменных осуществляется вручную и является итеративным процессом: если качество выполнения фигуры пилотажа построенным автоматом является неудовлетворительным, набор переменных корректируется. Задачу подбора предикатов не удалось полноценно решить для выполнения разво-

рота при использовании ранее разработанного способа: подавляющая часть построенных автоматов не смогла его выполнить, несмотря на высокое значение ФП.

Поиск автоматов проводился при помощи муравьиного алгоритма [5]. Для каждого набора тестов, способа представления автоматов и числа состояний автомата от трех до пяти были проведены 50 запусков алгоритма. Критерием остановки запуска была стагнация в течение 5000 вычислений ФП, при этом продолжительность запусков муравьиного алгоритма для различных конфигураций составляла от 9000 до 14000 вычислений ФП. В Таблице 4 приведены средние значения ФП, достигнутые по окончании запусков алгоритма. Для каждого набора тестов в левой колонке приведены значения ФП для ранее разработанного метода, в правой — для предлагаемого.

Число состояний	«Мертвая петля»		«Бочка»		Разворот	
	0,9814	0,9855	0,9832	0,9851	0,9891	0,9893
3	0,9814	0,9855	0,9832	0,9851	0,9891	0,9893
4	0,9834	0,9866	0,9854	0,9862	0,9900	0,9899
5	0,9838	0,9873	0,9856	0,9865	0,9900	0,9901

Таблица 4. Средние значения ФП, достигнутые на различных наборах тестов при использовании двух способов представления автомата

Как видно из таблицы, использование предлагаемого способа представления автомата, как правило, позволяет получить автоматы с более высоким значением ФП. Качество сгенерированных автоматов оценивалось при моделировании в авиасимуляторе. Исследования показали, что автоматы с предложенным способом представления, построенные для выполнения «мертвой петли», лучше управляют креном, а автоматы, построенные для «бочки», лучше управляют и креном, и тангажом. На тестах, описывающих разворот, с использованием обоих способов представления удалось построить автоматы с высокими значениями ФП, однако выполнить разворот смогли только автоматы с предлагаемым способом представления.

Для проведения вычислительных экспериментов использовался персональный компьютер с четырехъядерным процессором *Intel Core 2 Quad Q9400*. Время построения автоматов занимало порядка пяти минут при использовании предыдущего и порядка десяти минут при использовании нового способа представления.

Заключение

Предложен новый способ представления автоматов с непрерывными выходными воздействиями, улучшающий метод построения автоматов [3]. По сравнению с ранее разработанным способом он показывает лучшие ре-

зультаты по качеству построенных автоматов, а также делает логику работы автоматов более наглядной.

Литература

1. *Поликарпова Н.И., Шалыто А.А.* Автоматное программирование. СПб.: Питер, 2010.
 2. *Поликарпова Н.И., Точилин В.Н., Шалыто А.А.* Метод сокращенных таблиц для генерации автоматов с большим числом входных переменных на основе генетического программирования // Известия РАН. Теория и системы управления. 2010. №2. С. 100–117.
 3. *Александров А.В., Казаков С.В., Сергушичев А.А., Царев Ф.Н., Шалыто А.А.* Применение генетического программирования на основе обучающих примеров для генерации конечных автоматов, управляющих объектом со сложным поведением // Научно-технический вестник Санкт-Петербургского государственного университета информационных технологий, механики и оптики, 2011, №2 (72). С. 3–11.
 4. *Бужинский И.П., Ульянов В.И.* Применение муравьиных алгоритмов для построения автоматов управления системами со сложным поведением на основе обучающих примеров / Сборник докладов XV Международной конференции по мягким вычислениям и измерениям (SCM`2012). СПб: СПбГЭТУ «ЛЭТИ», 2012. Т. 1, С. 250–253
 5. *Chivilikhin D., Ulyantsev V.* Learning Finite-State Machines with Ant Colony Optimization // Lecture Notes in Computer Science, 2012, Volume 7461/2012, pp. 268–275
 6. FlightGear [Электронный ресурс]. Режим доступа <http://www.flightgear.org/> свободный. Яз. англ. (дата обращения 20.04.13).
-