

Министерство образования и науки Российской Федерации
Российская академия наук
Российский фонд фундаментальных исследований
Южный федеральный университет
Институт проблем управления им. В.А. Трапезникова РАН
ОАО «Концерн «ЦНИИ «Электроприбор»
Научно-исследовательский институт многопроцессорных вычислительных систем имени академика А.В. Каляева Южного федерального университета
Российская Ассоциация искусственного интеллекта
Журнал «Вестник компьютерных и информационных технологий»
Журнал «Мехатроника, автоматизация, управление»
Журнал «Проблемы управления»

ШЕСТАЯ ВСЕРОССИЙСКАЯ МУЛЬТИКОНФЕРЕНЦИЯ ПО ПРОБЛЕМАМ УПРАВЛЕНИЯ

МКПУ-2013

Материалы
6-й Всероссийской мультikonференции
30 сентября – 5 октября 2013 г.
с. Дивноморское, Геленджик, Россия

ТОМ 1

Ростов-на-Дону
2013

УДК 004.451.25:004.7(063)+ 004.89(063)
ББК 32.973
Ш 514

Ш514 6-я Всероссийская мультikonференция по проблемам
управления // Материалы 6-й Всероссийской мультikonференции.
Т.1. – Ростов-на-Дону: Изд-во ЮФУ, 2013. – 178 с.

ISBN

В первом томе материалов Шестой Всероссийской мультikonференции по проблемам управления (МКПУ-2013) представлены тезисы докладов локальной научно-технической конференции «Управление в интеллектуальных, эргатических и организационных системах» (УИнтЭргОС-2013) по направлению «Интеллектуальные системы».

Издание осуществлено при финансовой поддержке Российского фонда фундаментальных исследований, проект № 13-08-06045-г, и Южного федерального университета

М $\frac{14020700000}{6КО(03) - 2013}$ без объявл. ББК 32.973

ISBN

© Авторы докладов
© Институт проблем управления
им. В.А. Трапезникова РАН,
составление, 2013
© Научно-исследовательский институт
многопроцессорных вычислительных
систем имени академика А.В. Каляева
Южного федерального университета,
составление, оформление, 2013

РАЗДЕЛ 1

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

1.1. Теоретические вопросы искусственного интеллекта

И.П. Бужинский, В.И. Ульянов, Ф.Н. Царев

ПОСТРОЕНИЕ УПРАВЛЯЮЩИХ КОНЕЧНЫХ АВТОМАТОВ С ВЕЩЕСТВЕННЫМИ ПЕРЕМЕННЫМИ ПО ОБУЧАЮЩИМ ПРИМЕРАМ

*Национальный исследовательский университет информационных технологий, механики и оптики, г. Санкт-Петербург
{buzhinsky, ulyantsev, tsarev}@rain.ifmo.ru*

Введение

Управляющим конечным автоматом [1] называется шестерка $(S, s_0, E, A, \delta, \lambda)$, где S – конечное множество состояний, $s_0 \in S$ – начальное состояние, E – множество событий, A – множество выходных воздействий, $\delta: S \times E \rightarrow S$ – функция переходов, $\lambda: S \times E \rightarrow A$ – функция выходов. Автоматные программы являются наглядным способом представления систем управления, а также могут быть легко верифицированы [2] методом *Model Checking*. Ранее были разработаны несколько методов построения управляющих автоматов по спецификации их поведения: методы построения автоматов по тестам [3], по тестам и формулам темпоральной логики [4], по сценариям работы [5]. Эти методы применимы, когда события и выходные воздействия автоматов являются дискретными.

Более интересный случай, когда у объекта управления есть вещественные параметры, рассмотрен в [6], где в качестве объекта управления используется модель самолета. В рамках предложенного в этой работе подхода входные параметры объекта управления преобразуются в набор значений предикатов – булевых переменных. Однако такого рода дискретизация ограничивает множество реализуемых автоматами законов управления.

В настоящей работе предлагается метод, позволяющий строить по обучающим примерам автоматы, использующие для выработки выходных воздействий произвольные вещественные переменные от входных воздействий. Как будет показано, метод превосходит своего предшественника [6] по качеству генерируемых автоматов.

Постановка задачи

Рассматриваемая задача заключается в построении управляющего конечного автомата по набору из N обучающих примеров, или тестов, которые представляют собой примеры управления некоторым объектом с вещественными входными и выходными параметрами. В качестве объекта управления в настоящей работе, как и в работе [6], рассматривается модель самолета. Для моделирования работы автоматов используется авиасимулятор *FlightGear* [7].

Пусть у объекта управления имеются P входных и C управляющих параметров. Тест i ($i = 1..N$) представляет собой две последовательности длины $\text{len}[i]$. Последовательность значений входных параметров $\text{in}[i]$ состоит из чисел $\text{in}[i][t][k]$, где $t = 1..\text{len}[i]$ – момент времени, $k = 1..P$ – номер входного параметра. Аналогично, последовательность значений управляющих параметров $\text{out}[i]$ состоит из чисел $\text{out}[i][t][j]$, где $j = 1..C$ – номер управляющего параметра. Для модели самолета входными параметрами являются параметры полета (скорость, высота и т. д.), а управляющими – положения органов управления (элеронов, руля высоты и т. д.). Таким образом, тесты описывают состояние самолета и положения его органов управления на протяжении полета. В настоящей работе значения параметров записываются с частотой 10 Гц, и с этой же частотой происходят такты работы автоматов. Пример теста, записанного с помощью авиасимулятора *FlightGear*, приведен в табл. 1.

Пример теста ($P = C = 2$, $\text{len}[i] = 235$)

Таблица 1

Воздействие	Комментарий	$t = 1$...	$t = 10$...	$t = 235$
$\text{in}[i][t][1]$	Угол крена ($^\circ$)	-0,076	...	-0,076	...	-0,076
$\text{in}[i][t][2]$	Скорость (узлы)	251,42	...	252,29	...	289,40
$\text{out}[i][t][1]$	Положение элеронов	0,000	...	0,032	...	-0,003
$\text{out}[i][t][2]$	Положение руля высоты	-0,035	...	-0,039	...	-0,011

Автоматы с вещественными переменными

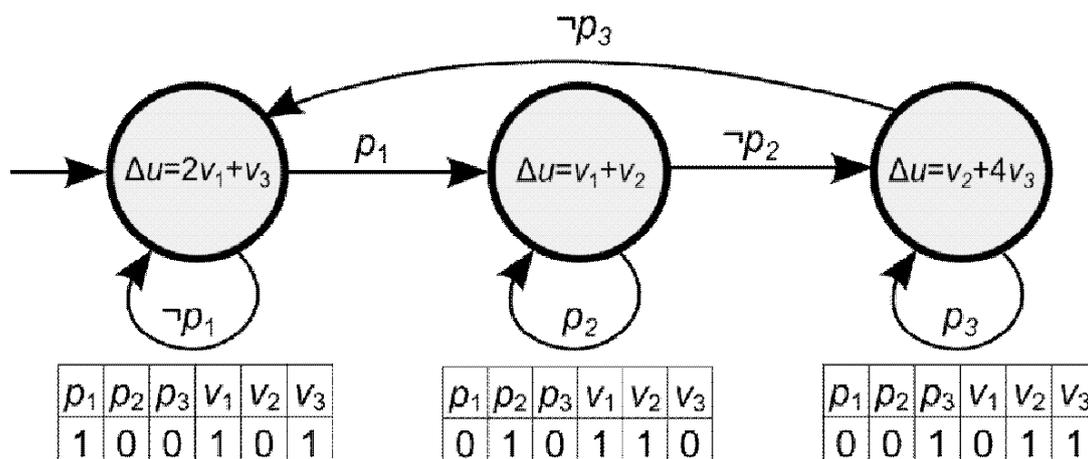
Рассмотрим понятие переменной – величины, зависящей от входных воздействий, переданных автомату на текущем и, возможно, на предыдущих тактах его работы. Переменные бывают булевыми (это предикаты) и вещественными. Как и в [6], значения предикатов используются в качестве событий автомата. Пусть задан набор предикатов p_1, \dots, p_m . Для представления функции переходов используется метод сокращенных таблиц [8]: в каждом состоянии автомата конечные состояния переходов определяются только *значимыми* предикатами.

Перейдем к вещественным переменным, которые ранее не использовались при автоматизированном построении управляющих автоматов.

Для каждого управляющего параметра $j = 1..C$ задается набор вещественных переменных $\{v_{j,i}\}$. Для каждого состояния автомата определены значимые вещественные переменные, по смыслу аналогичные значимым предикатам. Пусть s – текущее состояние автомата. Тогда изменение Δu_j выходного воздействия для j -го управляющего параметра по отношению к предыдущему такту работы автомата будем считать равной линейной комбинации значений вещественных переменных:

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

Коэффициенты $r_{s,i,j}$ заведомо равны нулю для не значимых в состоянии s вещественных переменных. Пример автомата приведен на рисунке. В каждом состоянии автомата значимыми являются один предикат и две вещественные переменные, при этом всего заданы три предиката и три вещественные переменные для единственного управляющего параметра u . Внутри состояний показаны конкретные законы управления вида (1). Под состояниями приведены булевы векторы, показывающие значимость переменных.



Пример автомата с используемым способом представления

Метод построения автоматов

Общая схема разработанного метода совпадает со схемой метода [6]. Построение автоматов происходит на основе поисковой оптимизации (например могут использоваться генетические алгоритмы). Особью алгоритма поисковой оптимизации является каркас автомата – автомат, выходные воздействия которого не определены. Каркас автомата, помимо начального состояния и функции переходов, включает в себя наборы значимых переменных для каждого состояния, представленные булевыми векторами. Используемая *функция приспособленности* (ФП) отражает близость поведения автомата к поведению, показанному в тестах:

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

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 за частую смену состояний автоматом, которая была признана отрицательно сказывающейся на его наглядности.

Выходные воздействия автоматов, рассматриваемых в настоящей работе, определяются только состояниями (используются автоматы Мура). Для их определения на фиксированном каркасе автомата достаточно задать числа $r_{s,i,j}$. Линейность выражения (1) позволяет путем решения S систем линейных уравнений сделать это так, чтобы ФП достигла максимума на заданном каркасе.

Экспериментальное исследование

Экспериментальная проверка разработанного метода построения автоматов, а также его сравнение с методом [6] были проведены на трех наборах тестов, описывающих выполнение моделью самолета различных фигур пилотажа: «мертвой петли», «бочки» и разворота в горизонтальной плоскости на 180° . Подбор наборов переменных осуществлялся вручную, при этом если качество выполнения фигуры пилотажа построенным автоматом являлось неудовлетворительным, то набор корректировался. При использовании метода [6] не удалось подобрать предикаты так, чтобы автомат смог выполнить разворот, в то время как для предлагаемого метода подбор переменных не оказался трудной задачей. Далее описаны результаты сравнения методов на наборах тестов, описывающих «мертвую петлю» и «бочку».

Для построения автоматов применялись генетический и муравьиный [9] алгоритмы, при этом последний показал несколько лучшую производительность и поэтому использовался в сравнении. Для каждого набора тестов и числа состояний автомата от трех до пяти были проведены 50 запусков каждого метода. Критерием остановки запуска была стагнация в течение 5000 вычислений ФП, при этом продолжительность запусков методов для различных конфигураций составляла от 9000 до 14000 вычислений ФП. В табл. 2 приведены средние значения ФП, достигнутые по окончании запусков методов. Результаты запусков метода [6] приведены перед косой чертой, результаты запусков предложенного метода – после.

Как видно из таблицы, использование предложенного метода позволяет получать автоматы с более высоким значением ФП. Качество построенных автоматов также измерялось в моделировании. Автоматы запускались при условиях, близким к условиям записи тестов. Среднее отклонение углов крена и тангажа от значений, записанных в тестах, оказалось меньше для автоматов, построенных с помощью предложенного метода, что говорит о его качестве. Время работы метода на персо-

нальном компьютере с четырехъядерным процессором *Intel Core 2 Quad Q9400* составляло порядка десяти-двадцати минут.

Средние значения ФП, достигнутые на различных наборах тестов при использовании двух методов построения автоматов

Таблица 2

Число состояний	«Мертвая петля»	«Бочка»
3	0,9814 / 0,9855	0,9832 / 0,9851
4	0,9834 / 0,9866	0,9854 / 0,9862
5	0,9838 / 0,9873	0,9856 / 0,9865

Заключение

Предложен метод построения управляющих конечных автоматов, использующих для выработки выходных воздействий вещественные переменные. Метод позволил построить автоматы более высокого качества, чем более ранний метод [6]. Метод также оказался работоспособным на наборе тестов, описывающих разворот самолета, на котором метод [6] не удалось заставить работать.

1. Поликарпова Н.И., Шалыто А.А. Автоматное программирование. СПб.: Питер, 2010.
2. Вельдер С.Э., Лукин М.А., Шалыто А.А., Яминов Б.Р. Верификация автоматных программ. СПб: Наука, 2011. – 242 с.
3. Царев Ф.Н. Метод построения управляющих конечных автоматов на основе тестовых примеров с помощью генетического программирования. // Информационно-управляющие системы, 2010, № 5, С. 31–36.
4. Егоров К. В., Царев Ф. Н., Шалыто А. А. Построение автоматов управления системами со сложным поведением на основе верификации и сценариев работы / Всероссийская научная конференция по проблемам информатики (СПИСОК-2012). СПб.: ВВМ. СПбГУ. 2012, С. 411–414.
5. Ulyantsev V., Tsarev F. Extended Finite-State Machine Induction using SAT-Solver / Proceedings of the 14th IFAC Symposium «Information Control Problems in Manufacturing - INCOM'12». IFAC, 2012, P. 512-517.
6. Александров А.В., Казаков С.В., Сергушичев А.А., Царев Ф.Н., Шалыто А.А. Применение эволюционного программирования на основе обучающих примеров для генерации конечных автоматов, управляющих объектами со сложным поведением // Известия РАН. Теория и системы управления. 2013. № 3, С. 85–100.
7. FlightGear [Электронный ресурс]. Режим доступа <http://www.flightgear.org/> свободный. Яз. англ. (дата обращения 05.06.13).
8. 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.
9. Chivilikhin D., Ulyantsev V. Learning Finite-State Machines with Ant Colony Optimization // Lecture Notes in Computer Science. 2012. V. 7461/2012. P. 268–275.