

Санкт-Петербургский государственный университет
информационных технологий, механики и оптики
Факультет информационных технологий и программирования
Кафедра «Компьютерные технологии»

А. Л. Новохатько, Т. М. Инюшин

**Проектирование и исследование автоматов для
решения задачи управления упрощенной моделью
автомобиля**

Санкт-Петербург
2011

Оглавление

Введение	3
1. Постановка задачи	4
1.1. Использование автоматов в векторных задачах	4
1.2. Задача об автомобиле	5
2. Реализация	6
2.1. Представление автоматов	6
2.2. Редактор автоматов.....	6
2.3. Панель настроек автомобиля	7
3. Построение и исследование автоматов.....	9
3.1. Автомат из двух состояний.....	9
3.2. Автомат из четырех состояний.....	11
Заключение.....	12
Источники	14
Приложение.....	15
Конфигурационные файлы	15
Исходные коды программ.....	15

Введение

В процессе работы с конечными автоматами возник интерес к тому, возможно ли их применение для решения задач в векторном пространстве – не в дискретном мире. И целесообразно ли будет использование автоматов для управления объектами в таком пространстве. В курсовой работе представлена попытка создания и применения автоматов для этих целей, а также выполнен анализ этого вопроса.

1. Постановка задачи

Задача курсовой работы – разработать среду для проектирования конечных автоматов и построить в ней несколько конечных автоматов для управления упрощенной моделью автомобиля в векторном двумерном пространстве, а также исследовать каждый из этих автоматов при разных параметрах модели.

1.1. Использование автоматов в векторных задачах

Основная проблема, связанная с использованием конечных автоматов в векторном пространстве, связана с тем, что у объектов в таком пространстве, как правило, бесконечное число входных данных, которыми они могут оперировать, – у автомата, который будет управлять такими объектами, будет бесконечное число входных воздействий. Разберем данную ситуацию на примере упрощенной модели автомобиля и ее окружения. Упростим автомобиль, считая его материальной точкой, обладающей координатами, скоростью, ускорением. Окружение представляет собой множество векторных объектов – линий и окружностей, являющихся для автомобиля непроходимыми преградами. Возникает вопрос: что будет входными воздействиями для автомата, который будет управлять автомобилем? Если это будут координаты объектов окружения, то каких? И в каком виде их представлять? Ясно, что входных воздействий не должно быть слишком много, иначе спроектировать такой автомат будет очень сложно. Поэтому встает вопрос о выборе набора наиболее важных входных данных для автомобиля, и о критерии данного выбора. Также необходимо составить набор действий, которые автомат может совершать над автомобилем. Очевидно, что весь процесс будет аппроксимироваться конечным числом шагов, за каждый из которых автомат выполняет над автомобилем какое-либо действие из набора. Далее рассмотрим все это уже на более конкретном примере.

1.2. Задача об автомобиле

Окружение – битовая карта любого размера (для поставленной задачи это почти ничем не отличается от задания окружения набором векторных объектов).

Автомобиль – материальная точка, обладающая такими характеристиками как:

- мощность (ускорение в пикселах);
- маневренность (скорость поворота в углах);
- максимальная скорость (в пикселах);
- качество тормозов (скорость торможения в пикселах).

За шаг автомобиль может выполнить следующие действия:

- повернуть налево;
- повернуть направо;
- увеличить скорость (нажать на газ);
- уменьшить скорость (затормозить).

Вопрос о выборе набора входных данных решен так: из автомобиля строятся лучи под углом 45 градусов к фронтальному направлению рис. 1 и вычисляется, в каком направлении из этих двух препятствие находится ближе к автомобилю.

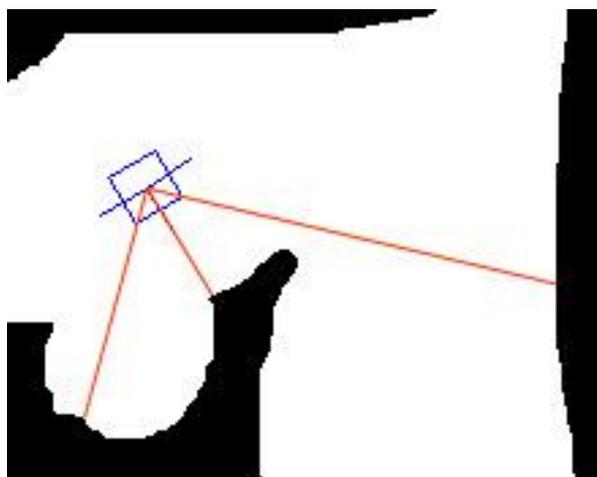


Рис. 1. Модель входных данных автомобиля

Далее строится еще один луч вперед от автомобиля и выясняется, есть ли помеха спереди. Задается значение расстояния в пикселях, так что если помеха спереди находится дальше этого расстояния, то считается, что путь вперед свободен (и наоборот). Таким образом, получается два булевых значения, которые и становятся входными воздействиями для будущего автомата.

2. Реализация

2.1. Представление автоматов

Автомат представляется в программе как набор состояний, набор связей (для каждой связи указаны идентификаторы начального и конечного состояний, входное воздействие, при котором эта связь используется, действие) и активное состояние. Начальное состояние автомата не указывается, но его можно выбирать и изменять в процессе работы автомата. На число связей, исходящих из одного состояния, ограничений не накладывается. Автомат детерминирован, так как при наличии двух связей из одного состояния с одинаковыми входными воздействиями выберется та из них, которая была раньше добавлена.

2.2. Редактор автоматов

Среда для проектирования автоматов, разработанная для этой программы, поддерживает такие действия как добавление и удаление состояний, добавление и удаление связей, полная настройка связей с указанием входных воздействий и прочих параметров. Ее внешний вид представлен на рис. 2.

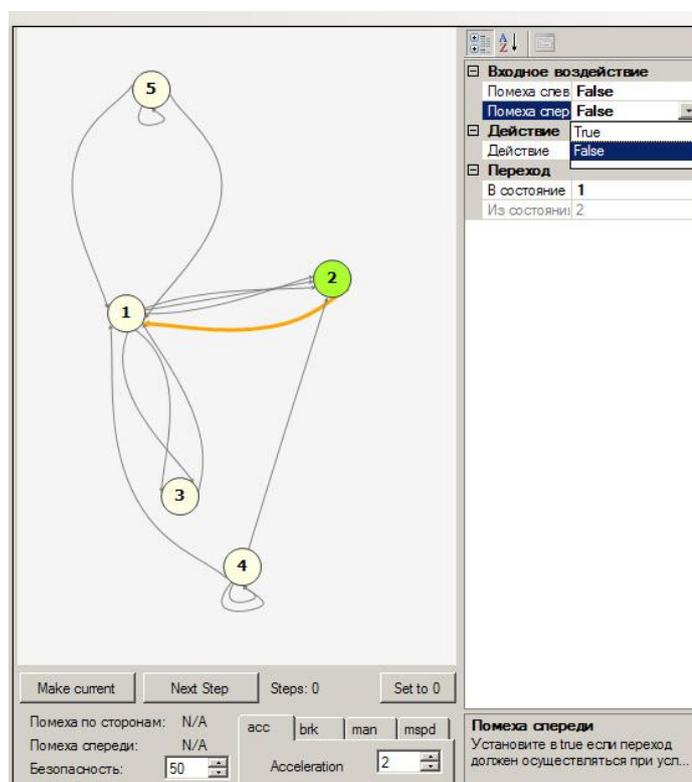


Рис. 2. Редактор автоматов

Пояснение пользовательского интерфейса приведено в разд. 2.3. Спроектированный автомат можно сохранить и загрузить. При этом данные автомата сериализуются в *xml*-файле, что существенно повышает совместимость с другими программами. Ниже описана панель настроек модели автомобиля.

2.3. Панель настроек автомобиля

Как было сказано, модель автомобиля характеризуется некоторым набором параметров. Их можно модифицировать посредством пользовательского интерфейса программы: панели настроек автомобиля, которая представлена на рис. 3. Параметр «Безопасность» – критерий того, будет ли считаться объект спереди автомобиля за помеху.

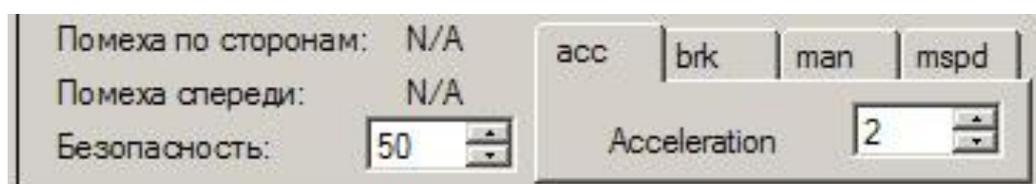


Рис. 3. Панель настроек автомобиля

Пояснение пользовательского интерфейса представлено в табл. 1. Параметры модели автомобиля подробнее рассматривались в разд. 1.2.

Таблица 1. Соответствие элементов пользовательского интерфейса и параметров модели автомобиля

Вкладка	Атрибут	Параметр модели автомобиля
acc	Acceleration	мощность
brk	Brakes	качество тормозов
man	Maneuver	маневренность
mspd	Maximum speed	максимальная скорость

Программа предоставляет возможность протестировать спроектированный автомат на задаче об автомобиле. Весь процесс исследования автомата сводится к тому, что задается начальное состояние, выделяя его и нажимая кнопку «Make current». После этого по шагам имеется возможность наблюдать развитие событий, используя кнопку «Next step». Эти кнопки, как и весь процесс работы программы, проиллюстрирован на рис. 4.

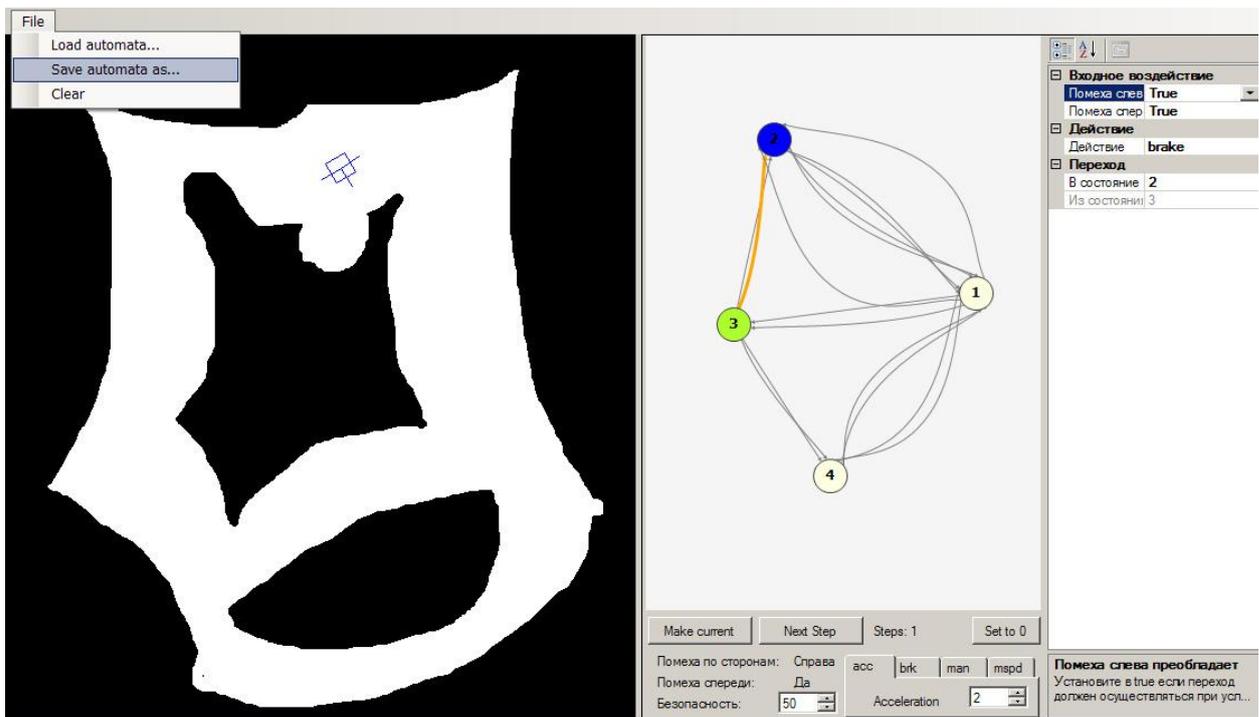


Рис. 4. Процесс работы программы

В программе также реализован счетчик шагов, проделанных автоматом, с возможностью его обнуления. К возможностям программы относится также изменение параметров упрощенной модели автомобиля «на ходу», удаление и добавление состояний и связей в автомат прямо во время работы, а также замена одного автомата на другой в любой момент.

3. Построение и исследование автоматов

В этом разделе рассматривается несколько различных автоматов, построенных для управления моделью автомобиля, и анализируется их эффективность. В качестве трассы для автомобиля используется карта, представленная на рис. 5. Она характеризуется тем, что не имеет острых углов, и легко проходимая даже автомобилями, управляемыми автоматами с небольшим числом состояний.

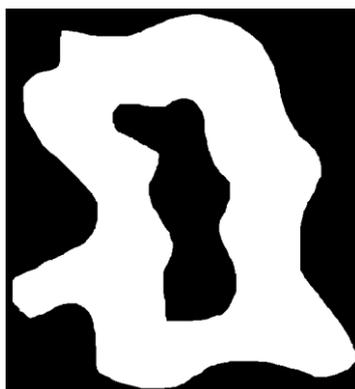


Рис. 5. Карта

3.1. Автомат из двух состояний

На рис. 6 предложен простой автомат из двух состояний с восьмью связями.

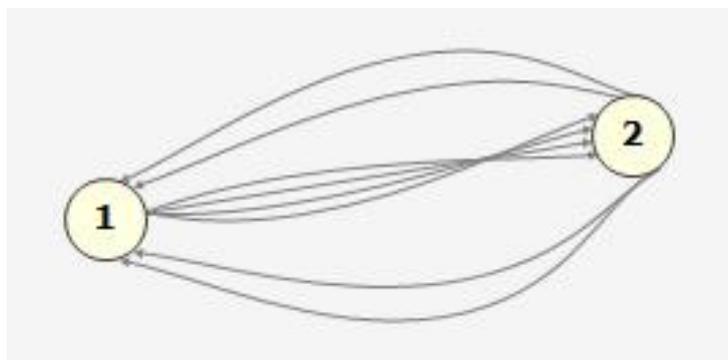


Рис. 6. Автомат из двух состояний

Ниже изображена таблица переходов данного автомата (табл. 2).

Таблица 2. Таблица переходов автомата из двух состояний

Текущее состояние	Входное воздействие	Действие	Следующее состояние
1	(left, false)	turn right	2
1	(right, false)	turn left	2
1	(left, true)	turn right	2
1	(right, true)	turn left	2
2	(left, false)	accelerate	1
2	(right, false)	accelerate	1
2	(left, true)	brake	1
2	(right, true)	brake	1

Таблица представляет собой список связей, для каждой из которых указаны текущее состояние, входное воздействие, действие и следующее состояние. Входное воздействие представлено в виде (left или right, true или false). Left – преобладает помеха слева, right – преобладает помеха справа. True – присутствует помеха спереди, false – отсутствует помеха спереди. Действия: turn left – повернуть налево, turn right – повернуть направо, accelerate – ускориться, brake – затормозить.

Связи, которые идут из состояния 1 в состояние 2, выполняют поворот автомобиля в сторону, где нет бокового препятствия при всех других возможных входных воздействиях (в данном случае независимо от препятствия спереди), а так как их всего два, то число этих связей равно четырем. Обратные связи из состояния 2 в состояние 1 выполняют торможение или ускорение автомобиля в зависимости от наличия переднего препятствия опять же при всех возможных других входных воздействиях (в данном случае независимо от помех слева и справа). Данный автомат хорошо работает при правильно подобранном соотношении параметров «безопасность» и «максимальная скорость» автомобиля, и проходит приведенную выше карту в среднем за 144 шага.

Данный автомат был протестирован при следующих параметрах модели автомобиля (подробнее параметры описаны в разд. 2.3):

- безопасность = 50;
- мощность = 2;
- максимальная скорость = 10;
- качество тормозов = 3;
- маневренность = 10.

Отметим, что при повышении маневренности до 20 градусов, автомат проходит ту же карту в среднем за 131 шаг.

3.2. Автомат из четырех состояний

На рис. 7 предложен более сложный автомат из четырех состояний с большим числом связей.

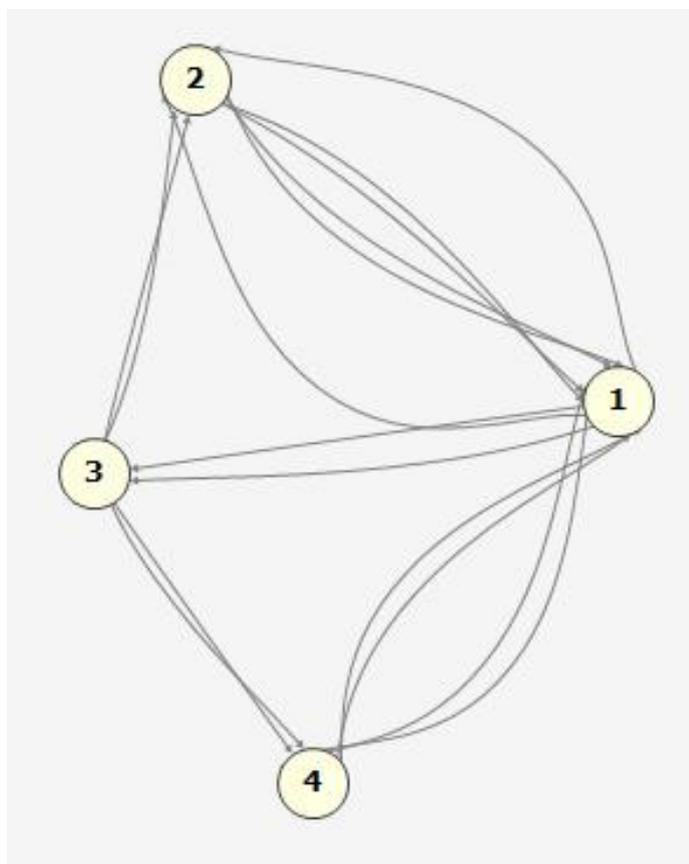


Рис. 7. Автомат из четырех состояний

Ниже изображена таблица переходов данного автомата (табл. 3). Таблица представляет собой список связей, для каждой из которых указаны текущее состояние, входное воздействие, действие и следующее состояние. Входное воздействие представлено в виде (left или right, true или false). Left – преобладает помеха слева, right – преобладает помеха справа. True – присутствует помеха спереди, false – отсутствует помеха спереди. Действия: turn left – повернуть налево, turn right – повернуть направо, accelerate – ускориться, brake – затормозить.

Таблица 3. Таблица переходов автомата из четырех состояний

Текущее состояние	Входное воздействие	Действие	Следующее состояние
1	(right, false)	accelerate	2
1	(left, false)	accelerate	2
1	(left, true)	brake	3
1	(right, true)	brake	3
2	(left, false)	turn right	1
2	(right, false)	turn left	1
2	(left, true)	turn right	1
2	(right, true)	turn left	1
3	(right, false)	turn left	2
3	(left, false)	turn left	2
3	(left, true)	turn right	4
3	(right, true)	turn right	4
4	(right, false)	turn left	1
4	(left, false)	turn left	1
4	(left, true)	turn right	1
4	(right, true)	turn right	1

Данный автомат состоит из двух циклов – цикла разгона и цикла торможения, в каждом цикле также присутствуют связи, выполняющие поворот автомобиля в сторону наименьшей помехи. Данный автомат работает хорошо при правильно подобранном соотношении параметров автомобиля, и проходит рассматриваемую карту в среднем за 135 шагов, что лучше, чем в случае автомата с двумя состояниями. Параметры:

- безопасность = 50;
- мощность = 2;
- максимальная скорость = 10;
- качество тормозов = 3;
- маневренность = 10.

При маневренности равной 20 рассматриваемая карта проходится автоматом за 130 шагов.

Заключение

В процессе исследования было обнаружено, что при повышении маневренности автомобиля при работе с автоматом из четырех состояний (рис. 7) до 20 градусов, он проходит рассматриваемую карту в среднем за 130 шагов, что примерно соответствует тому, как проходит ту же карту более простой автомат из двух состояний (рис. 6). Это свидетельствует о том, что чем выше «качество» автомобиля, тем меньше на его управление влияет сложность автомата. Грамотно спроектированный автомат поможет избежать столкновений. Иллюстрацию столкновения несложно получить, установив

счетчик максимальной скорости на 50, а ускорение на 10. При таких настройках автомобиль не вписывается в первый же поворот. Однако вполне возможно создать такой автомат, который бы безопасно управлял автомобилем даже на высоких скоростях. Здесь встает вопрос о выборе набора входных данных и структуре самого автомата.

Применение конечных автоматов в векторных задачах вполне возможно, более того, процесс проектирования можно автоматизировать, используя генетические алгоритмы. Использование таких алгоритмов также позволит создать расширенный набор входных воздействий, что (если говорить о задаче про автомобиль) увеличит качество управления им. Не говоря уже о том, что спроектировать автомат с большим числом входных воздействий человеку будет не под силу.

Источники

1. Шалыто А. А. Автоматное программирование
http://is.ifmo.ru/download/shalyto_doklad_v_saratove.ppt
2. Бедный Ю. Д., Шалыто А. А. Применение генетических алгоритмов для построения автоматов в задаче «Умный муравей».
http://is.ifmo.ru/works/_ant.pdf
3. Шалыто А. А. Технология автоматного программирования
<http://www.softcraft.ru/auto/switch/aptech.shtml>

Приложение

Конфигурационные файлы

К проекту прилагается папка **auto** с примерами спроектированных автоматов – автомата из двух состояний (auto2.xml) и автомата из четырех (auto4.xml) состояний. В этой же папке размещены автоматы с большим числом состояний. В папке **images** содержатся разные варианты поля для задачи об автомобиле.

Исходные коды программ

Исходные коды и проектные файлы содержатся в папке **_src**. Программа реализована на языке C# в среде проектирования *MS Visual C# 2008 express edition*.