

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

Кирилл Николаев

**Отчет по лабораторной работе  
«Построение управляющих автоматов  
с помощью генетических алгоритмов»**

Санкт-Петербург  
2011

## Содержание

Введение.....	3
Постановка задачи.....	4
Задача об «Умном муравье-3».....	4
Автоматы Мили.....	6
Способ представления автомата.....	6
Реализация.....	7
Функция приспособленности.....	7
Метод мутации.....	7
Метод скрещивания.....	7
Генерация очередного поколения.....	8
Результаты работы.....	9
Графики максимального и среднего значения функции приспособленности.....	9
Заключение.....	10
Источники.....	11

## Введение

В лабораторной работе изучается применение генетических алгоритмов для построения управляющих автоматов. В качестве примера рассматривается задача об «Умном муравье-3». Задача об «Умном муравье-3» является модификацией более ранней задачи об «Умном муравье». Последняя была предложена в 1992 году Д. Джефферсоном и другими в работе [1].

Целью работы было построение автомата Мили (Mealy), управляющего поведением муравья, таким образом, чтобы задача решалась наиболее эффективным образом. Применимость генетических алгоритмов для решения задачи об умном муравье была описана Ф. Царевым и А. Шалыто в работе [3].

При выполнении работы использовался программный модуль «Виртуальная лаборатория обучения генетическому программированию для генерации управляющих автоматов», разработанный А. Давыдовым и Д. Соколовым и описанный в работе [2]. Эта система позволяет разделять разработку генетических алгоритмов на отдельные модули (особь, фитнес-функция, функция мутации, функция кроссинговера и т. п.) и оформлять их в виде отдельных плагинов.

## Постановка задачи

Задачей лабораторной работы являлось построение с помощью генетических алгоритмов конечного автомата Мили, решающего задачу об «Умном муравье». При этом требовалось использовать представление автоматов с помощью графов переходов, традиционный генетический алгоритм и метод «усечения» (truncation) для отбора особей в очередное поколение.

### Задача об «Умном муравье-3»

Игра происходит на поверхности тора размером  $32 \times 32$  клетки (рис. 1). В некоторых клетках поля находится еда с заданной вероятностью  $\mu$  (эти клетки обозначены на рисунке черным цветом). Изначально он находится в фиксированной стартовой клетке (отмечена на рис. 1).

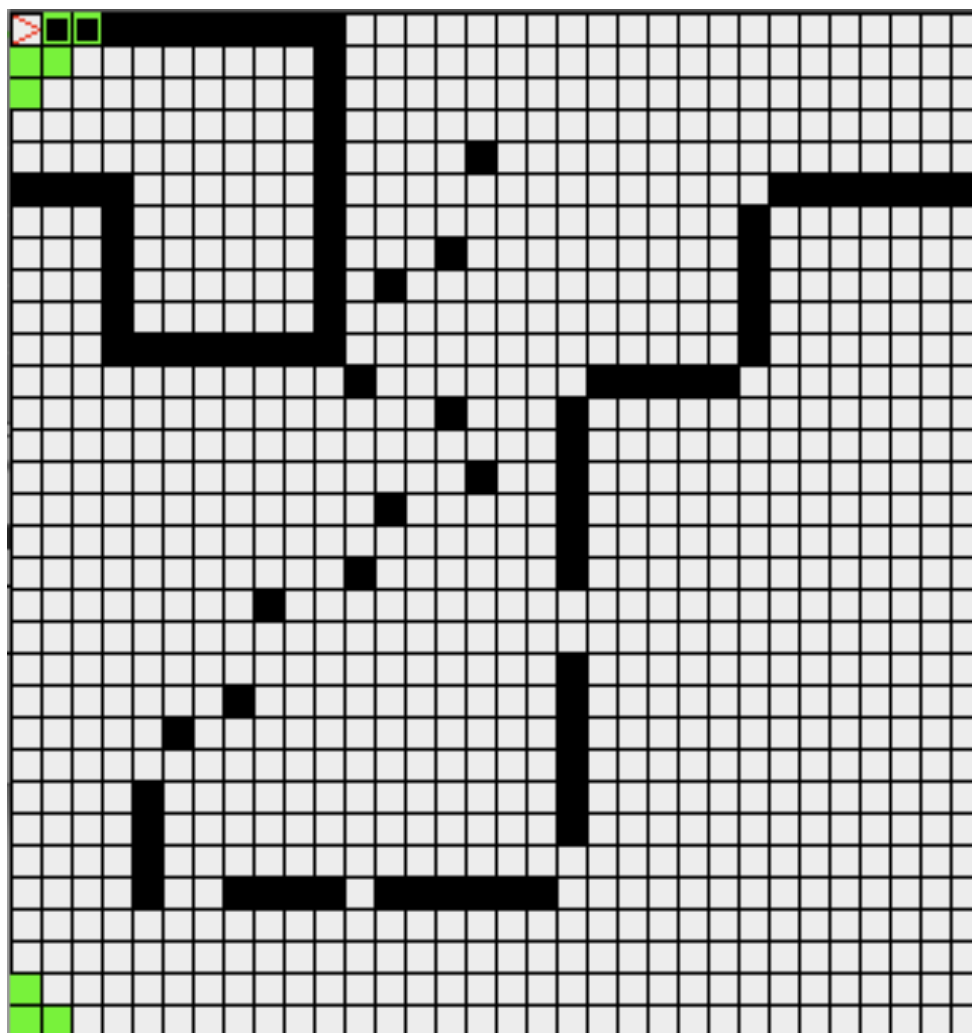


рис. 1. Поле умного муравья

Муравей «видит» восемь клеток вокруг себя и умеет определять, находится ли в них еда. За один ход муравей может совершить одно из четырех действий:

- сделать шаг вперед, съедая еду, если она находится на клетки перед ним,
- повернуть налево,
- повернуть направо.

Игра длится 200 ходов, на каждом из которых муравей совершает одно из перечисленных действий. Результатом игры является количество еды, найденной муравьем. Съеденная еда не восполняется, муравей жив на протяжении всей игры (еда не обязательна для поддержания его жизни).

Задачей является создание конечного автомата, управляющего муравьем так, что в среднем он съедает за 200 ходов максимальное количество еды.

## Автоматы Мили

В данной работе для описания поведения муравья используются автоматы Мили. Автоматом Мили называется совокупность следующих объектов:

- $S$  — множество состояний автомата,
- $X, Y$  — соответственно входной и выходной алфавиты,
- $\delta: S \times X \rightarrow S$  — функция переходов,
- $\mu: S \times X \rightarrow Y$  — функция выходов.

При этом множества связаны следующими функциями:

- $s(t+1) = \delta(s(t), x(t))$ ,
- $y(t+1) = \mu(s(t), x(t))$ .

В задаче об умном муравье входными воздействиями является логический вектор, содержащий информацию о том, есть ли еда в клетках, которые видит муравей. А выходными сигналами являются действия, которые совершает муравей при некотором переходе.

### *Способ представления автомата*

Автомат хранится в виде графа переходов, где вершины — это состояния, а ребра — переходы. Автомат хранится в виде полной таблицы переходов из каждого состояния при всех возможных входных воздействиях. Последних в нашем случае  $2^8$ , так как муравей получает информацию о наличие еды в восьми ячейках поля.

В каждой ячейке таблицы переходов записана информация о переходе из соответствующего ячейке состояния при заданном воздействии: номер состояния и действие, совершаемое муравьем при этом переходе.

## Реализация

В данной работе используется традиционный генетический алгоритм. В роли особей выступают конечные автоматы в вышеописанном представлении.

### *Функция приспособленности*

Функция приспособленности в генетическом алгоритме вычисляется для каждой особи и является мерой эффективности решения этой особью итоговой задачи.

Чтобы оценить, насколько хорошо полученный автомат управляет действиями муравья, необходимо смоделировать поведение агента и оценить результаты. Таким образом, запустив муравья на имеющемся поле, можно получить следующие показатели:

- $a$  — количество съеденной за 200 ходов пищи,
- $t$  — номер последнего из ходов, на которых была съедена пища.

За значение функции  $F$  приспособленности принимается следующая величина:

$$F = a - t / 200.$$

Эта величина усредняется по итогам нескольких запусков симуляции. Легко видеть, что значение этой функции тем выше, чем эффективней автомат решает задачу.

### *Метод мутации*

Мутация в генетическом алгоритме — это случайное изменение некоторых признаков особи, применяющееся с некоторой вероятностью к случайной особи.

При имеющемся представлении особей у них можно с некоторой вероятностью изменить четыре признака:

- стартовое состояние,
- действие, поставленное в соответствие случайно выбранному состоянию,
- состояние, в которое ведет случайно взятое ребро,

### *Метод скрещивания*

Оператор скрещивания принимает на вход две особи — родителей, и возвращает также две особи — потомков. Обозначим первых как S1 и S2, а вторых — как P1 и P2.

Скрещивание происходит следующим образом:

- 1) В половине случаев особи обмениваются номерами стартового состояния.
- 2) Для каждого состояния особи в половине случаев обмениваются значением выходного сигнала.
- 3) Для каждого состояния особи обмениваются номерами конечных состояний переходов. Если обозначить переход из состояния  $i$  в автомате  $A$  по входному воздействию «вперед есть пища» как  $A(i, 0)$ , а по «вперед нет пищи» - как  $A(i, 1)$ , то с равной вероятностью будет справедливо одно из четырех:

$$\begin{aligned} P1(i, 0) &= S1(i, 0), P1(i, 1) = S1(i, 1), P2(i, 0) = S2(i, 0), P2(i, 1) = S2(i, 1) \\ P1(i, 0) &= S2(i, 0), P1(i, 1) = S2(i, 1), P2(i, 0) = S1(i, 0), P2(i, 1) = S1(i, 1) \\ P1(i, 0) &= S1(i, 0), P1(i, 1) = S2(i, 1), P2(i, 0) = S2(i, 0), P2(i, 1) = S2(i, 1) \\ P1(i, 0) &= S2(i, 0), P1(i, 1) = S1(i, 1), P2(i, 0) = S1(i, 0), P2(i, 1) = S1(i, 1) \end{aligned}$$

### ***Генерация очередного поколения***

Размер поколения, вероятность мутации и доля особей, попадающих в новое поколение из предыдущего, являются настраиваемыми параметрами алгоритма.

Чтобы выбрать тех особей, которые будут использоваться для генерации нового поколения, используется «метод усечения». Данная стратегия использует отсортированную по возрастанию популяцию. Число особей для скрещивания выбирается в соответствии с порогом  $T \in (0, 1)$ . Порог определяет, какая доля особей, начиная с самой первой (то есть самой приспособленной), будет принимать участие в отборе. Среди особей, попавших «под порог», случайным образом выбирается  $N$  особей и записывается в промежуточный массив, используемый для скрещивания.

Новое поколение дополняется до необходимого размера следующим образом. Случайным образом из выбранной элиты выбираются две пары особей, скрещиваются, и их потомки добавляются в новое поколение.

Далее с вероятностью, задаваемой параметром «вероятность мутации», ко всем особям нового поколения применяется оператор мутации.



## ***Результаты работы***

Алгоритм был запущен с параметрами:

- 5 состояний в автомате;
- вероятность нахождения еды в клетке — 0,05;
- вероятность мутации — 0,02;
- размер поколения равен 200 особям.

В результате был получен автомат из пяти состояний, под управлением которого муравей съедает в среднем 38 единиц еды на заданном поле. Это составляет в среднем 75% еды.

## ***Графики максимального и среднего значения функции приспособленности***

На рисунке 2 представлен график среднего значения функции приспособленности среди особей поколения (зеленый) и ее максимального значения в пределах поколения (синий).

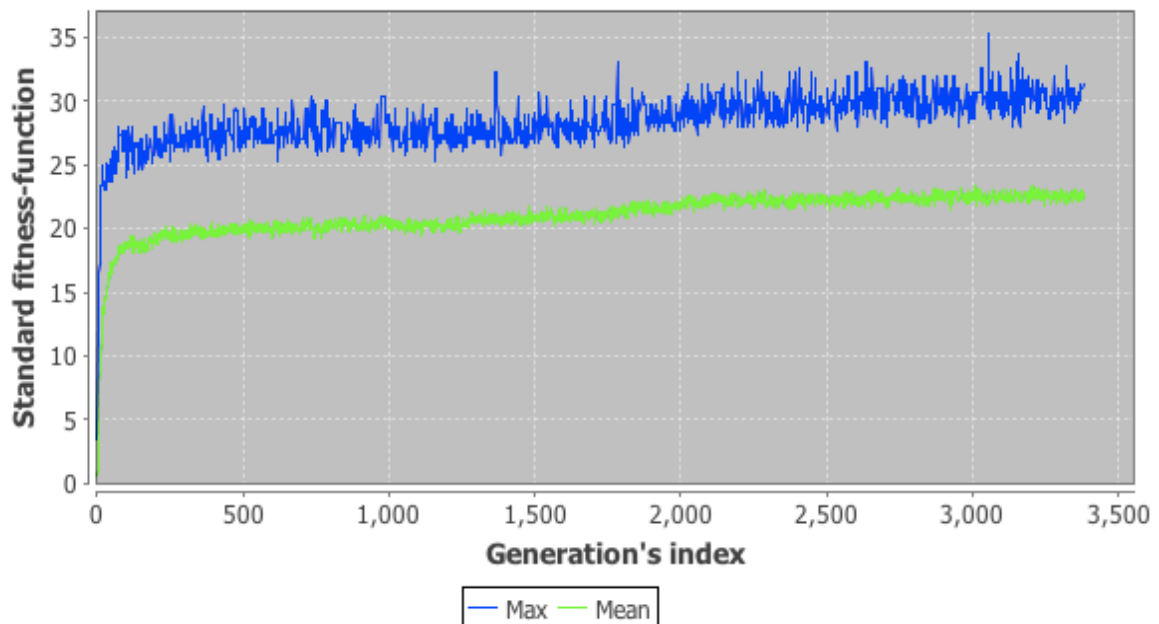


рис. 2: График функции приспособленности

На рисунке виден разброс получающегося значения функции приспособленности, который обусловлен рандомизированными испытаниями на случайном поле и агрессивным применением мутации при создании нового поколения. Однако в целом наблюдается рост максимального значения приспособленности.

## **Заключение**

Результаты лабораторной показывают, что использованный метод имеет достаточно высокую эффективность для решения задачи об «Умном муравье». Всего за 3 минут был построен автомат, который позволяет съесть 75% еды на случайном поле. При этом он содержит пять состояний.

## Источники

1. Jefferson D., Collins R., Cooper C., Dyer M., Flowers M., Korf R., Taylor C., Wang A. The Genesys System. 1992  
[www.cs.ucla.edu/~dyer/Papers/AlifeTracker/Alife91Jefferson.html](http://www.cs.ucla.edu/~dyer/Papers/AlifeTracker/Alife91Jefferson.html)
2. Давыдов А.А., Соколов Д.О., Царев Ф.Н., Шалыто А.А. Виртуальная лаборатория обучения генетическому программированию для генерации управляющих конечных автоматов.  
[http://is.ifmo.ru/works/2\\_93\\_davidov\\_sokolov.pdf](http://is.ifmo.ru/works/2_93_davidov_sokolov.pdf)
3. Царев Ф.Н., Шалыто А.А. Применение генетического программирования для генерации автомата в задаче об «умном муравье»  
[http://is.ifmo.ru/genalg/ant\\_ga.pdf](http://is.ifmo.ru/genalg/ant_ga.pdf)