

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

А. С. Охитин

**Отчет по лабораторной работе
«Использование генетических алгоритмов для
построения управляющих автоматов»**

Вариант №21

Санкт-Петербург
2012

Оглавление

1. Введение.....	3
1.1. Постановка задачи.....	3
1.2. Задача о роботе, обходящем препятствия.....	3
2. Реализация.....	4
2.1. Функция приспособленности.....	4
2.2. Автомат Мили.....	4
2.3. Оператор мутации и оператор скрещивания.....	5
2.4. Операторы селекции.....	5
2.5. Элитизм.....	5
3. Результаты.....	6
4. Выводы.....	8
5. Источники.....	9

1. Введение

Цель лабораторной работы – сравнить эффективность работы генетического алгоритма, строящего автомат Мили для решения задачи о роботе, обходящим препятствия, при использовании в качестве оператора селекции рангового отбора и турнирного отбора. При выполнении лабораторной работы использовался фреймворк Watchmaker [1], весь исходный код написан на языке Java.

1.1. Постановка задачи

Задачей данной лабораторной работы является изучение эффективности работы генетического алгоритма при использовании рангового и турнирного отборов в качестве операторов селекции. Генетический алгоритм должен строить автомат Мили для решения задачи о роботе, обходящем препятствия.

1.2. Задача о роботе, обходящем препятствия

Дано поле 32×32 клетки, представляющее собой фиксированный лабиринт с препятствиями. Робот видит только клетку впереди себя.

За один ход робот может совершить одно из действий:

- пойти вперед (если впереди препятствие, то ничего не произойдет);
- повернуть налево;
- повернуть направо;
- ничего не делать.

Задача робота – добраться до цели за число ходов, не превышающее двухсот.

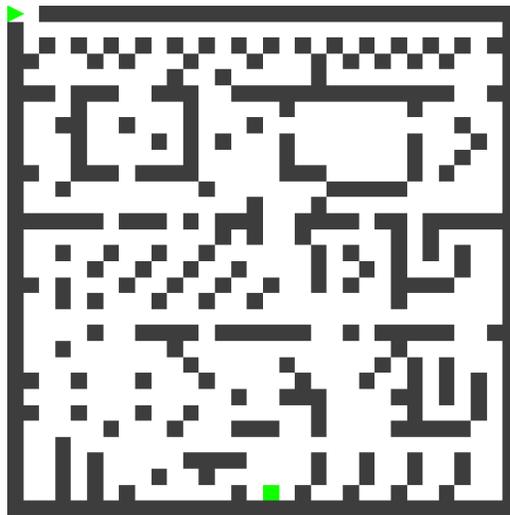


Рис. 1 – Лабиринт. Треугольником обозначено начальное положение робота, зеленым квадратом – финиш.

2. Реализация

2.1. Функция приспособленности

Как показал анализ задачи, использование расстояния до финиша в вычислении функции приспособленности оказывает не лучшее влияние на алгоритм: возможны ситуации, при которых геометрическое расстояние до финиша невелико, но при этом роботу необходимо сделать большое число шагов. При этом, тестирование показало, что большинство особей, способных дойти до финиша, делают это за число шагов, не превышающее 800.

Следует отметить, что из большого числа случайных особей доходят до финиша порядка 0,1%. На рисунке 1 представлены результаты всех дошедших до финиша за разумное время автоматов из случайно сгенерированных 100 000.

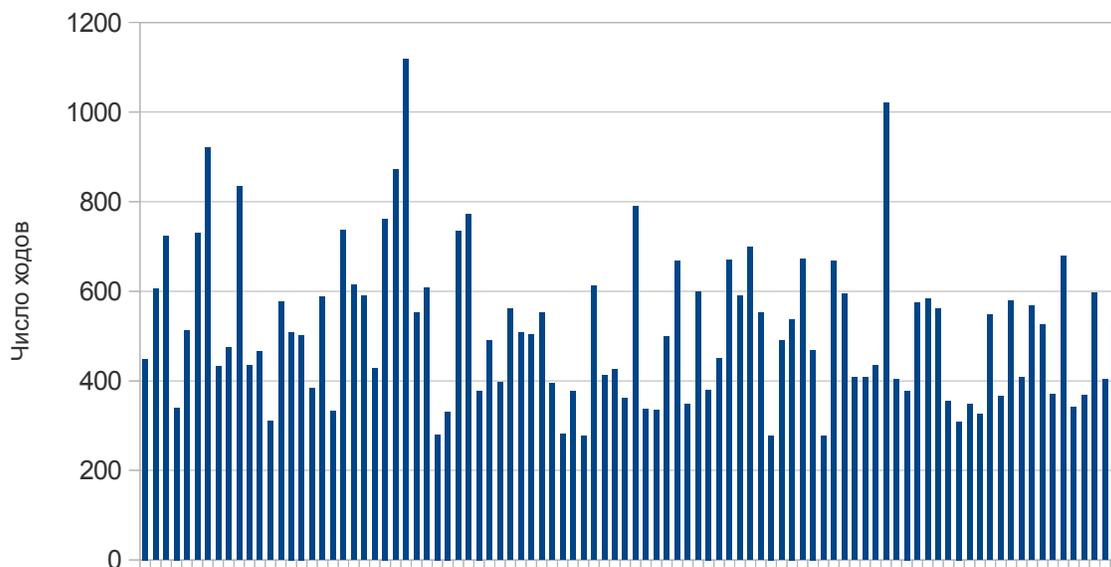


Рис. 2 – Число ходов, потребовавшееся всем дошедшим до финиша автоматам среди 100 000 случайных.

Таким образом, представляется разумным определить функцию приспособленности так:

$$F(A) = 600 - s,$$

где A – оцениваемая особь, s – необходимое для нее число шагов до финиша. Успешными считаются особи, для которых значение функции приспособленности не меньше 400.

2.2. Автомат Мили

Все особи представляют собой автоматы Мили, характерной особенностью которых является обусловленность действий переходами между состояниями. В данной работе у всех автоматов семь состояний и четыре предусмотренных действия, соответствующие действиям роботов. В памяти компьютера автомат представлен четырьмя массивами, в которых для состояния с номером i в i -тых ячейках хранятся:

- номер состояния, в которое переходит автомат, если перед роботом стена;

- номер состояния, в которое переходит автомат, если перед роботом стены нет;
- действие, совершаемое роботом, если впереди стена;
- действие, совершаемое роботом, если впереди стены нет.

2.3. Оператор мутации и оператор скрещивания

В результате действия оператора мутации, у автомата меняются действия, переходы и начальное состояние на случайные с вероятностью 0,1.

Оператор скрещивания представляет собой одноточечный кроссовер [2], в результате его применения к двум автоматам порождаются два автомата-потомка, у одного из которых состояния, номера которых не превышают некоторое случайное число, соответствуют состояниям первого предка, остальные – второго; у другого потомка – наоборот.

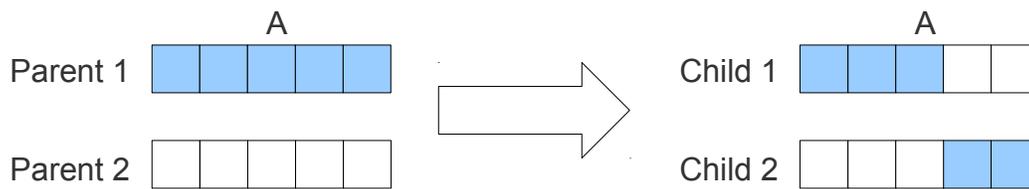


Рис. 3 – Схема одноточечного кроссовера.

2.4. Операторы селекции

В лабораторной работе сравниваются результаты использования различных механизмов отбора [3], а именно рангового и турнирного.

Ранговый отбор характерен тем, что для каждой особи вероятность ее попадания в промежуточную популяцию пропорциональна ее порядковому номеру в отсортированной по возрастанию приспособленности данной популяции.

Турнирный отбор производится следующим образом: случайным образом выбираются две особи, из которых лучшая попадает в промежуточную популяцию. Процесс продолжается, пока промежуточная популяция не окажется заполненной.

2.5. Элитизм

В алгоритме используется элитизм [3]: 5% особей с наилучшими значениями функции приспособленности, минуя отбор, попадают в промежуточную популяцию.

3. Результаты

Было произведено 100 запусков генетического алгоритма с использованием рангового отбора и 100 запусков генетического алгоритма с турнирным отбором. При каждом запуске было сгенерировано 15000 поколений, состоящих из 100 особей. Были вычислены средние арифметические функции приспособленности лучших особей для каждого поколения.

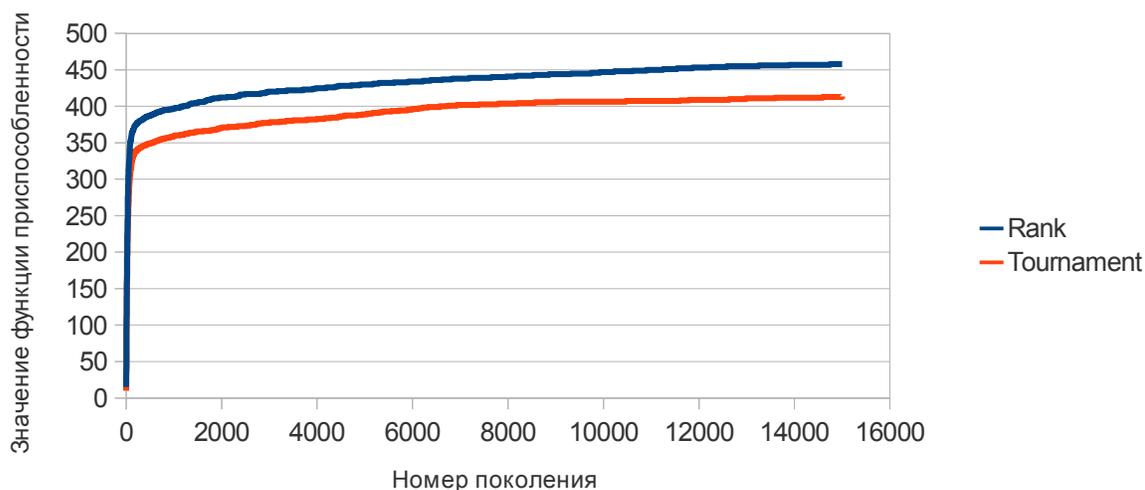


Рис. 4 – Средние арифметические функций приспособленности после 100 запусков в зависимости от номера поколения.

Заметна разница между ранговым и турнирным отборами: ранговый отбор быстрее порождает требуемую особь, а при большом числе поколений значения функций приспособленности получаемых особей превосходят аналогичные для турнирного отбора примерно на 10%. Ниже представлен решающий задачу автомат, который был получен с помощью рангового отбора.

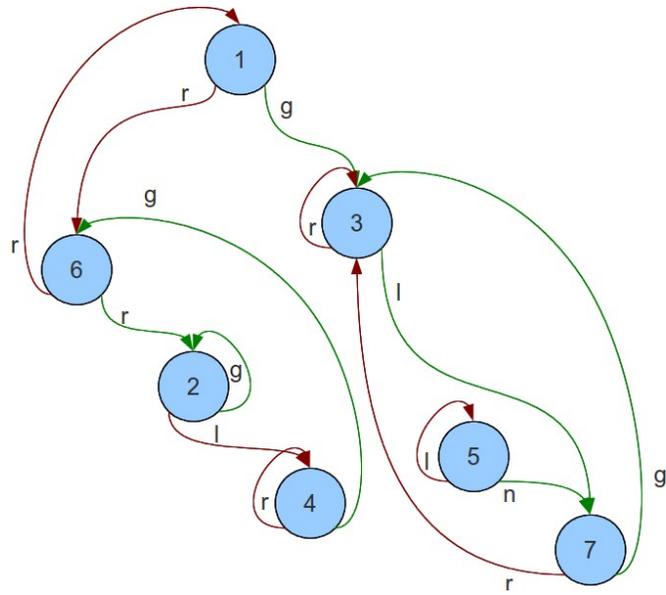


Рис. 5 – Полученный автомат, доходящий до цели за 96 переходов. Зеленые стрелки – переходы для случаев, когда впереди свободно; красные – когда впереди стена.

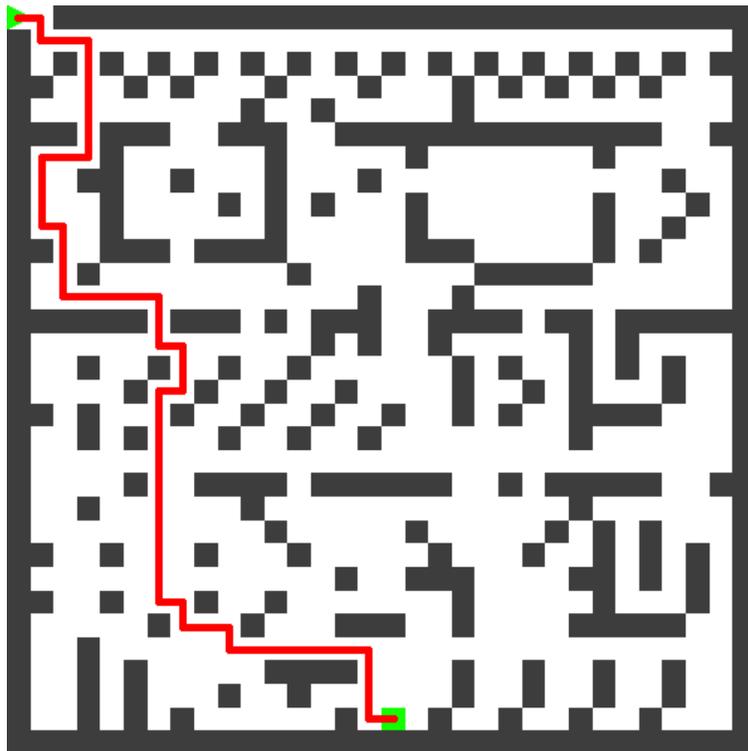


Рис. 6 – Путь, который проходит вышеуказанный автомат.

4. Выводы

На основании тестирования можно сделать вывод о том, что использование как рангового, так и турнирного отборов допустимо при решении данной задачи. Однако, можно рекомендовать использовать ранговый отбор в качестве оператора селекции, как более эффективный.

5. Источники

1. Watchmaker framework for Evolutionary Computation.
<http://watchmaker.uncommons.org/>.
2. Методы представления конечных автоматов в генетических алгоритмах.
<http://is.ifmo.ru/~buzdalov/lab-2011/presentations/automata-representation.pdf>
3. Яминов Б. Генетические алгоритмы.
<http://rain.ifmo.ru/cat/view.php/theory/unsorted/genetic-2005>