

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

М. Я. Пинский

Отчет по лабораторной работе
«Применение генетических алгоритмов для решения
задачи об «Умном муравье»

Вариант № 9

2009

Оглавление

1. Введение.....	3
2. Постановка задачи	4
3. Генетический алгоритм.....	5
3.1. Представление автоматов.....	5
3.2. Создание начального поколения.....	5
3.3. Мутация.....	5
3.4. Скрещивание и отбор потомка.....	5
3.5. Функция приспособленности.....	5
4. Результаты работы	6
Заключение.....	8
Источники.....	9

1. Введение

Целью лабораторной работы было исследование генетических алгоритмов для построения конечных автоматов Мили. В качестве примера рассматривается задача об *Умном муравье* [1].

При выполнении лабораторной работы использовалась программа *GLOpt*, написанная студентами кафедры «Компьютерные технологии» СПбГУ ИТМО.

Игра *Умный муравей* происходит на поверхности тора размером 32x32 клетки. За ход муравей может выполнить следующие действия: повернуть налево; повернуть направо; сделать шаг вперед и, если в новой клетке есть еда, съесть её; ничего не делать. Игра длится 200 ходов.

Цель игры – создать муравья с минимальным числом состояний, который за минимальное число ходов ест как можно больше яблок.

Автомат типа Мили – конечный автомат, который генерирует выходные действия в зависимости от текущего состояния и входного сигнала (наличие или отсутствие еды перед муравьем).

2. Постановка задачи

Задачей лабораторной работы являлось построение с помощью генетических алгоритмов конечного автомата Мили, решающего задачу об *Умном муравье*. Использовалось представление автоматов с помощью графов переходов. Для генерации очередного поколения применялся клеточный генетический алгоритм и метод *рулетки* [2].

3. Генетический алгоритм

Разработанный алгоритм состоит из четырех частей:

- создание начального поколения;
- мутация;
- скрещивание каждой особи с «соседями». Выбор из полученных особей потомка;
- вычисление функции приспособленности.

3.1. Представление автоматов

Автомат хранится как двумерный массив, в котором число строк равно числу состояний автомата, а число столбцов равно двум. В ячейках хранится номер состояния, в котором оказывается автомат при данном переходе, и действие, которое совершает при этом муравей. В первом столбце описан переход муравья, если перед ним нет еды, а во втором переходе в случае, когда перед муравьем находится еда. В описание автомата так же входит номер начального состояния.

3.2. Создание начального поколения

Начальное поколение создается случайным образом и хранится в списке в отсортированном по убыванию порядке. Параметром сортировки является значение функции приспособленности. В начале находится «элитная» часть, процент которой задается пользователем.

3.3. Мутация

После создания нового поколения каждый индивид, не находящийся в «элитной» части, с вероятностью заданной пользователем подвергается мутации. При этом с некоторой вероятностью, которая также может быть настроена, или изменяется один из переходов автомата, или мутирует номер начального состояния.

После генерации каждых пятисот поколений все особи, кроме лучшей, меняются на созданные случайно.

3.4. Скрещивание и отбор потомка

Скрещивание происходит следующим образом: по двум данным особям строятся две новые. Для всех номеров состояний первая наследует по одному переходу от каждого родителя. Вторая особь наследует оставшиеся два перехода.

При создании нового поколения каждый муравей, который не находится в «элитной» части, скрещивается с каждым из четырех своих соседей, а также с одним из муравьев в «элитной» части. Автомат, который перейдет в новое поколение выбирается из полученных и исходного методом «рулетки», который состоит в том, что вероятность выпадения каждого автомата пропорциональна его фитнес-функции.

3.5. Функция приспособленности

Функция приспособленности *Fitness* вычисляется по формуле:

$$\text{Fitness} = \text{Apples} - \text{Steps} / 200,$$

где *Apples* – число съеденных яблок, *Steps* – номер хода, на котором муравей съел последнее яблоко.

4. Результаты работы

В результате работы генетического алгоритма был получен автомат Мили, который решает задачу об Умном муравье. Автомат имеет восемь состояний, значение функции приспособленности равно 84,035. Муравей, управляемый этим автоматом, съедает 85 яблок за 193 шага.

При построении были использованы следующие параметры:

- процент «элитных» особей (ElitePart) = 0,1;
- размер поколения (GenerationSize) = 200;
- вероятность мутации (MutationProbability) = 0,2;
- число состояний (StatesCount) = 8;
- вероятность мутации состояния (StateMutationProbability) = 0,1.

Сгенерированный автомат Мили приведен на рис. 1.

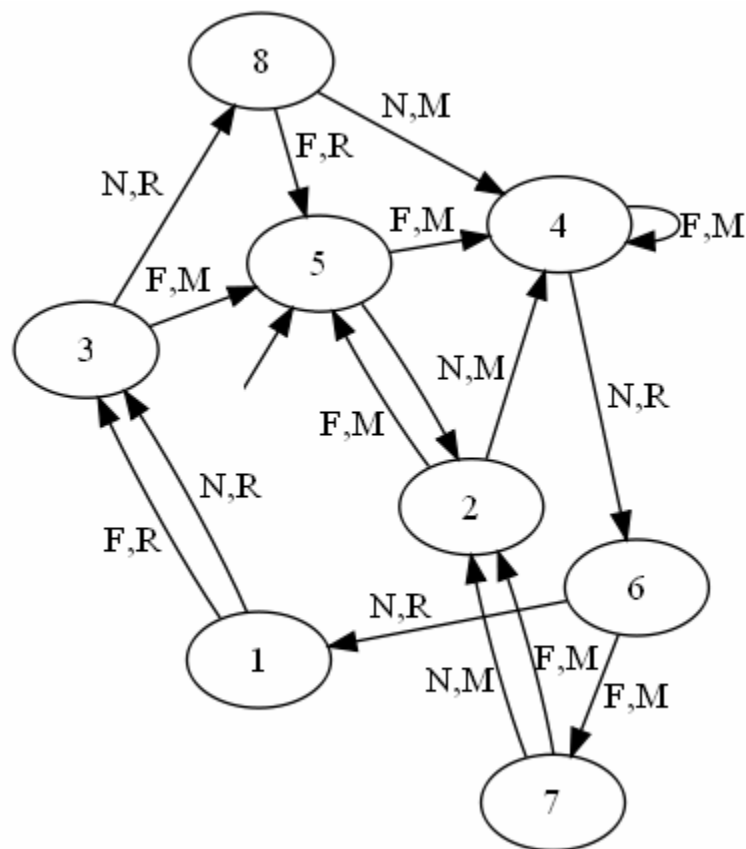


Рис. 1. Сгенерированный автомат Мили

На каждом переходе обозначены две буквы. Первая: F (Food) или N (Nothing) – наличие или отсутствие еды перед муравьем. Вторая буква обозначает действие, которое совершает муравей: M (Move) – шаг вперед, R (Right) – повернуть направо, L (Left) – повернуть налево. Начальное состояние – 5.

График функции приспособленности лучшего индивида от времени приведен на рис. 2.

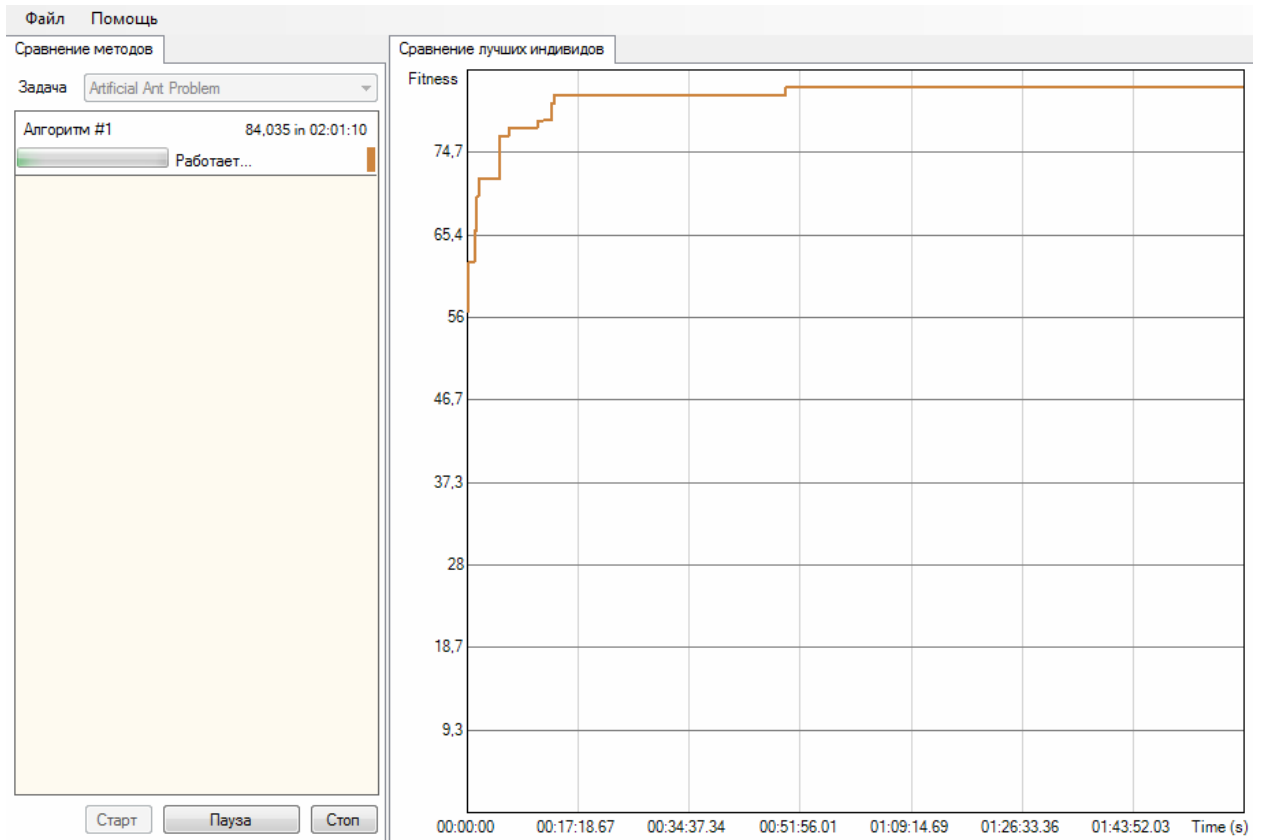


Рис.2. График зависимости максимального значения функции приспособленности от времени

Заключение

Результаты лабораторной работы показали, что с помощью генетических алгоритмов можно быстро сгенерировать автомат Мили с восемью состояниями, который эффективно решает задачу об *Умном муравье*. Достоинством описанного в работе метода является быстрота получения достаточно хорошего результата. В работе [3] был построен автомат, имеющий пять состояний и съедающий только 83 яблока, в то время как описанный выше автомат съедает 85 яблок.

Источники

1. *Бедный Ю. Д., Шалыто А. А.* Применение генетических алгоритмов для построения автоматов в задаче «Умный муравей» http://is.ifmo.ru/works/_ant.pdf
2. *Яминов Б.* Генетические алгоритмы
<http://rain.ifmo.ru/cat/view.php/theory/unsorted/genetic-2005>
3. *Терескин А. А.* Отчет по лабораторной работе «Использование генетических алгоритмов для построения управляющих автоматов» http://is.ifmo.ru/genalg/labs/_tereskin.pdf