

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

А. П. Минюк

**Отчет по лабораторной работе
«Автоматное программирование и генетические алгоритмы на
примере задачи об «Умном муравье»**

Вариант № 2

Санкт-Петербург
2009

Введение	3
1. Постановка задачи	4
1.1. Задача об «Умном муравье»	4
1.2. Автомат Мура	4
2. Реализация	6
2.1. Способ представления автоматов	6
2.2. Метод скрещивания	6
2.3. Метод мутации	6
2.4. Метод генерации очередного поколения	6
2.5. Описание вычисления фитнес функции	6
3. Результаты работы	6
3.1. График роста максимального значения фитнес функции ..	Error! Bookmark not defined.
3.2. Таблица переходов полученного автомата	Error! Bookmark not defined.
Заключение	9
Источники	10

Введение

Цель лабораторной работы – построить с помощью генетических алгоритмов конечный автомат Мура, решающий задачу об «Умном муравье».

При выполнении работы использовался программный комплекс для изучения методов глобальной оптимизации *GlOpt* [1], разработанный студентами кафедры «Компьютерные технологии» СПбГУ ИТМО. Данный комплекс позволяет реализовывать генетические алгоритмы в виде исполняемых плагинов и изучать процесс их выполнения.

1. Постановка задачи

Задачей лабораторной работы является построение с помощью генетических алгоритмов конечного автомата Мура, решающего задачу об «Умном муравье». При этом автомат должен быть представлен в виде битовых строк, должен быть использован классический алгоритм и метод «рулетки» для генерации очередного поколения.

1.1. Задача об «Умном муравье»

Игра происходит на поверхности тора размером 32x32 клетки. В некоторых клетках находятся яблоки, всего их 89 штук. За ход муравей может выполнить следующие действия:

- повернуть налево;
- повернуть направо;
- сделать шаг вперед и, если в новой клетке есть яблоко, съесть его;
- ничего не делать.

Игра длится 200 ходов. Цель игры – найти муравья, который за минимальное число ходов ест как можно больше яблок. Пример игрового поля представлен на рис. 1.

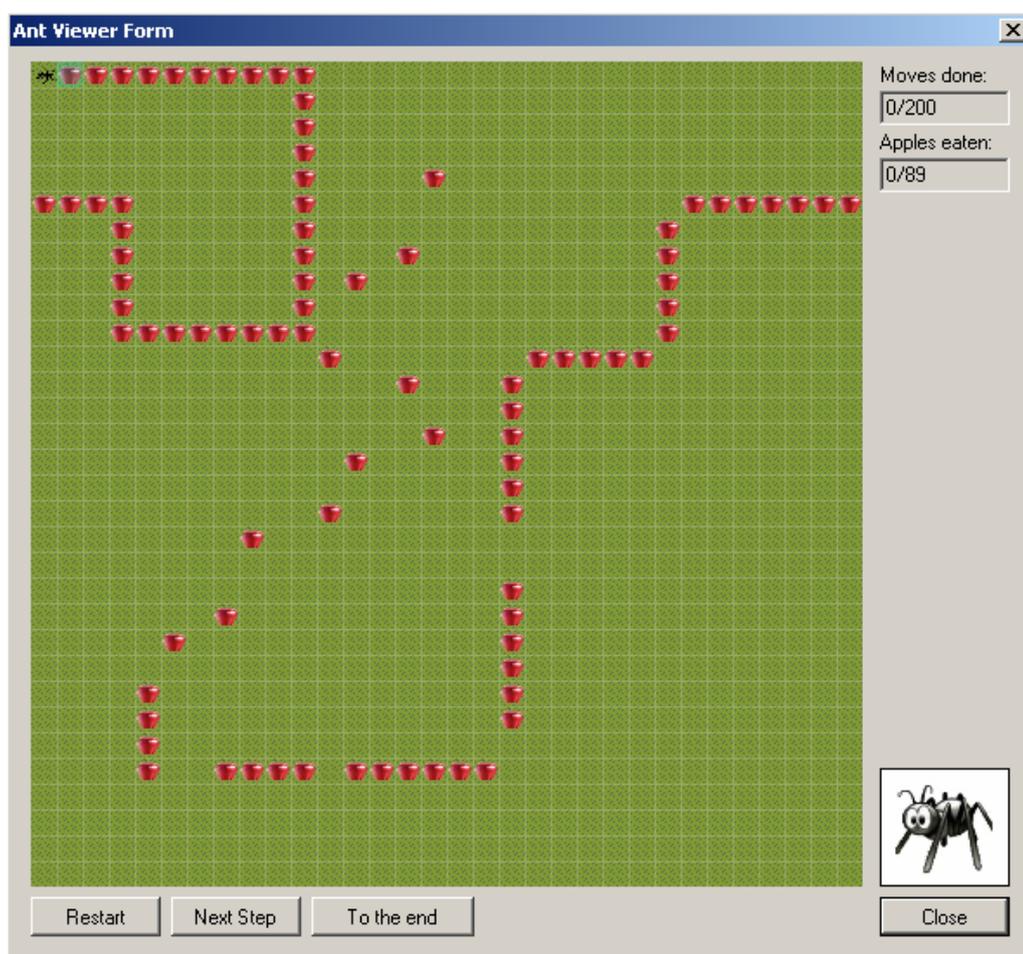


Рис. 1. Игровое поле

1.2 Автомат Мура

Согласно книге «Автоматное программирование» [2], автоматом Мура называется автомат, у которого выходное воздействие зависит только от состояния и не зависит от входного воздействия. Различают два типа автомата Мура: автомат первого рода формирует выходные воздействия на основе текущих значений внутренних переменных, в

то время как автомат второго рода сначала обновляет свое состояние, а затем на основе обновленного состояния формирует выходное воздействие. Для выполнения задания данной лабораторной работы было решено использовать автомат Мура второго рода, пример такого автомата изображен в виде диаграммы переходов на рис. 2.

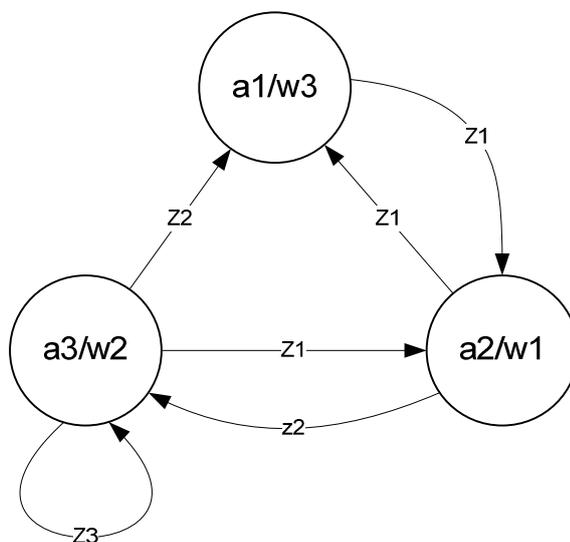


Рис. 2. Автомат Мура: а – состояния, z – входные воздействия, w – действия, соответствующие состояниям

В каждой вершине написано соответствующее ей выходное воздействие, таким образом, оно не зависит от входного воздействия, а зависит только от текущего состояния автомата.

2. Реализация

Для реализации алгоритма с помощью *GlOpt* необходимо реализовать два класса, наследованные соответственно от *SearchOperator* и *GeneticOptimizationAlgorithm*.

2.1. Способ представления автоматов

Автомат, управляющий муравьем, представляется массивом битов. По числу состояний (в данной работе – восемь состояний) определяется необходимое число битов для хранения состояния в двоичном представлении (в данной работе – восемь бит). Битовый массив содержит восемь записей состояний, имеющих следующую структуру: двоичная запись номера выходного воздействия (два бита) и номеров состояний (по три бита), в которые необходимо перейти в случае отсутствия впереди муравья яблока и в случае его наличия.

2.2. Метод скрещивания

При скрещивании два предка порождают два потомка. При скрещивании битовая строка предка делится на две части случайной длины, и одна из частей заменяется на часть строки другой особи такой же длины. Полученная строка и является битовой строкой потомка.

2.3. Метод мутации

Каждый бит в автомате инвертируется с вероятностью 80%.

2.4. Метод генерации очередного поколения

Сначала генерируется 100 случайных особей.

Для каждой особи выполняются следующие действия:

1. Происходит скрещивание с другими пятью случайными особями.
2. К каждой из полученных особей с вероятностью 15% применяется оператор мутации.
3. Методом «рулетки» определяется кандидат на переход в следующее поколение (вероятность перехода пропорциональна значению фитнес-функции индивидуума).
4. Если кандидат обладает лучшим значением фитнес-функции, чем исходный индивидуум, то он занимает его место.

Если фитнес-функция не изменилась в течение 500 поколений, то выполняется большая мутация: остается самая приспособленная особь, остальные особи заменяются на случайные особи.

2.5. Описание вычисления фитнес функции

Фитнес-функция вычисляется по формуле: $F + \frac{200-S}{S}$. Здесь F – число яблок, съеденных муравьем, S – число шагов, сделанных муравьем до момента съедения последнего яблока. Вычисление числа сделанных шагов и съеденных яблок производится путем эмуляции поведения муравья, управляемого автоматом.

3. Результаты работы

В результате работы генетического алгоритма был получен автомат Мура из восьми состояний, который управляет муравьем, съедающим 86 яблок из 89. Значение функции приспособленности полученного муравья равно 85,03. На рис. 3 изображен график роста максимального значения фитнес-функции при работе алгоритма.

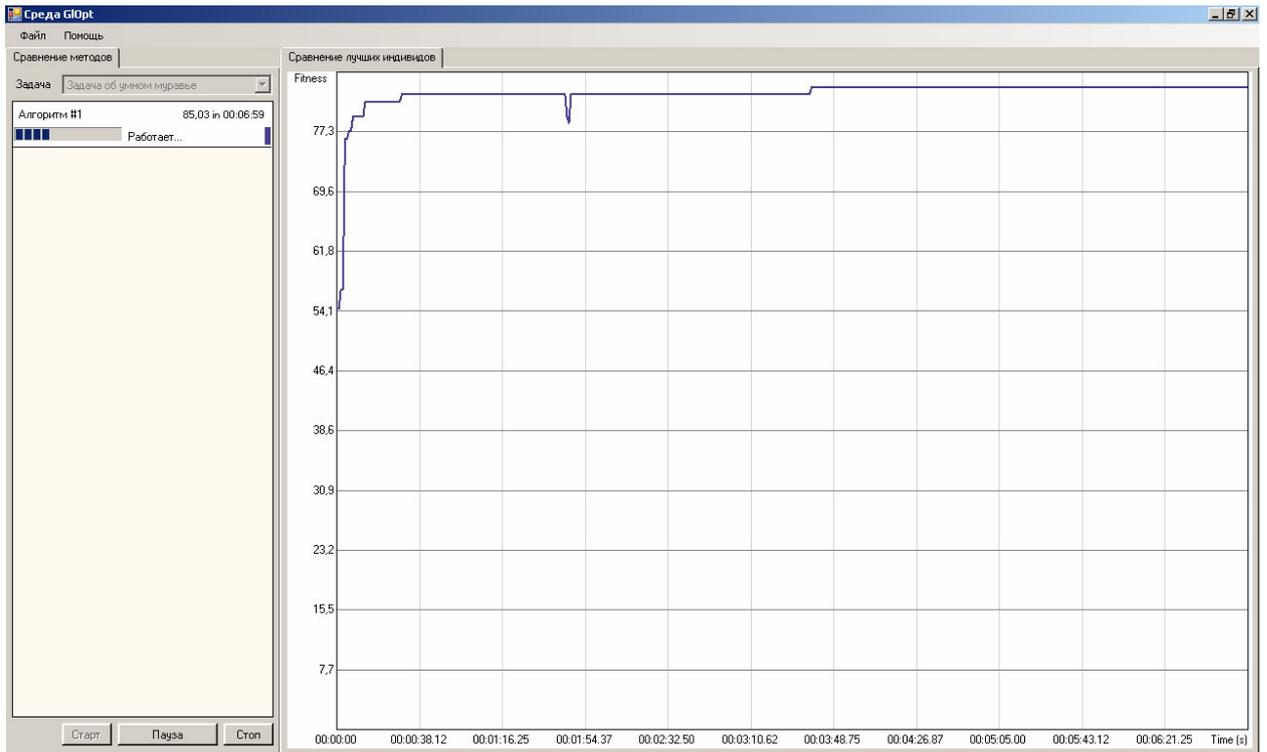


Рис. 3. График роста фитнес функции

В таблице приведен граф переходов построенного автомата.

Состояние		Наличие яблока впереди	
		Присутствует	Отсутствует
№	Действие		
0	Направо	2	1
1	Вперед	4	5
2	Направо	3	1
3	Вперед	6	5
4	Вперед	5	3
5	Вперед	6	4
6	Направо	7	1
7	Направо	0	1

На рис. 4 изображено графическое представление полученного автомата.

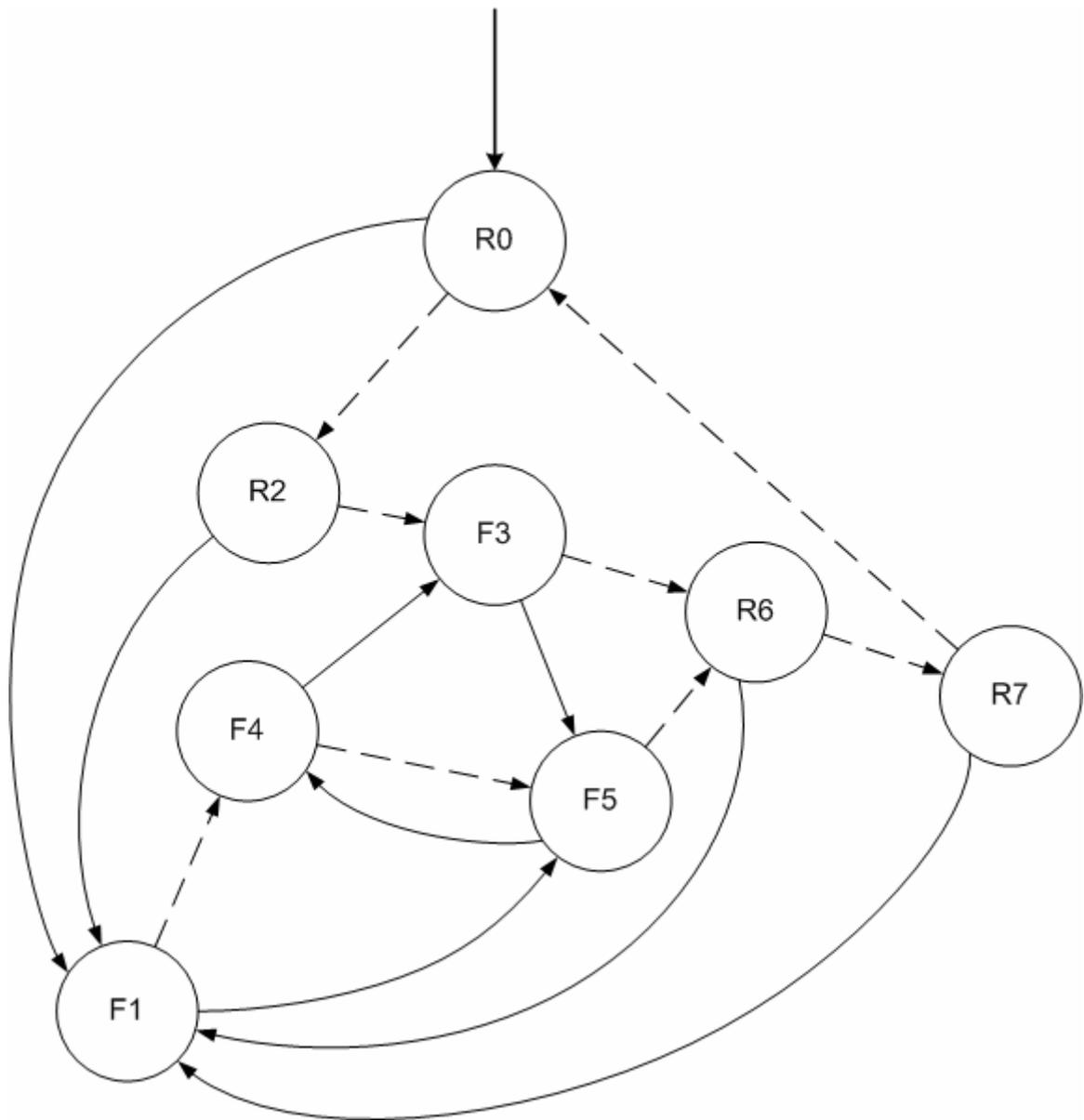


Рис. 4. Полученный автомат

- > Переход, если перед муравьем есть еда.
 - - - - -> Переход, если перед муравьем нет еды.
 F – Движение вперед. R – Поворот на право

Заключение

В результате лабораторной работы был получен автомат Мура из восьми состояний, кодирующий муравья, который успевает съесть 86 яблок из 89. Данный результат был получен за три с половиной минуты генетического поиска, но после большой мутации, что подтверждает скорость выбранного метода в поиске локальных по оптимальности решений и необходимость дополнительных эвристик для нахождения более качественного решения.

Источники

1. Документация к комплексу для изучения методов глобальной оптимизации «GIOpt». http://is.ifmo.ru/courses/_giopt-src.rar
2. Поликарпова Н. И., Шальто А. А. Автоматное программирование. СПб.: Питер, 2009.