

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

Кириллова А.А.

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

Вариант №9

Санкт-Петербург
2011 г.

Оглавление

Введение	3
1. Постановка задачи	4
1.1. Задача о роботе, обходящем препятствия	4
1.2. Автомат Мили	5
2. Алгоритм имитации отжига	6
2.1. Сверхбыстрый отжиг	6
2.2. Метод Xin Yao	7
3. Реализация	8
3.1. Функция приспособленности	8
3.2. Генерация новой точки	8
4. Результаты работы	9
4.1. Начальная температура для сверхбыстрого отжига	9
4.2. Начальная температура для метода Xin Yao	10
4.3. Сравнение экспериментальных данных методов	11
5. Лучший полученный автомат	12
6. Заключение	13
7. Источники	14

Введение

В данной лабораторной работе требуется сравнить эффективность работы метода имитации отжига при использовании сверхбыстрого отжига и метода Xin Yao на примере построения автомата Мили, решающего задачу о «Роботе, обходящем препятствия». Для решения поставленной задачи был проведен ряд исследований. Для каждого из методов была найдена начальная температура, при которой достигался наилучший результат работы. И в конечном итоге было проведено сравнение результатов работы каждого из методов, положив начальные значения температур равными соответствующим найденным.

Для необходимых исследований была написана программа на языке C++, строящая автоматы Мили, которые решают задачу о «Роботе, обходящем препятствия» с использованием заданных методов. В результате серии запусков этой программы были получены результаты, на основе которых построены графики, позволяющие сравнить эффективность работы методов.

1. Постановка задачи

Задача лабораторной работы – сравнить алгоритмы сверхбыстрого отжига и метода Хип Яо, при помощи которых строится автомат Мили из восьми состояний, решающий задачу о «Роботе, обходящем препятствия» [1]. Критерий оценки автомата заключается в числе действий, за которое робот доходит до цели – необходимо минимизировать эту величину.

1.1. Задача о «Роботе, обходящем препятствия»

Имеется игровое поле 32 на 32 клеток с препятствиями (рис.1). В верхней левой свободной клетке находится робот, который может видеть только клетку прямо перед собой. Робот может совершать следующие действия:

- пойти вперед (если впереди препятствие, то он останется на месте);
- повернуть налево;
- повернуть направо.

Задача робота – добраться до цели за наименьшее число действий, не превышающее 200.

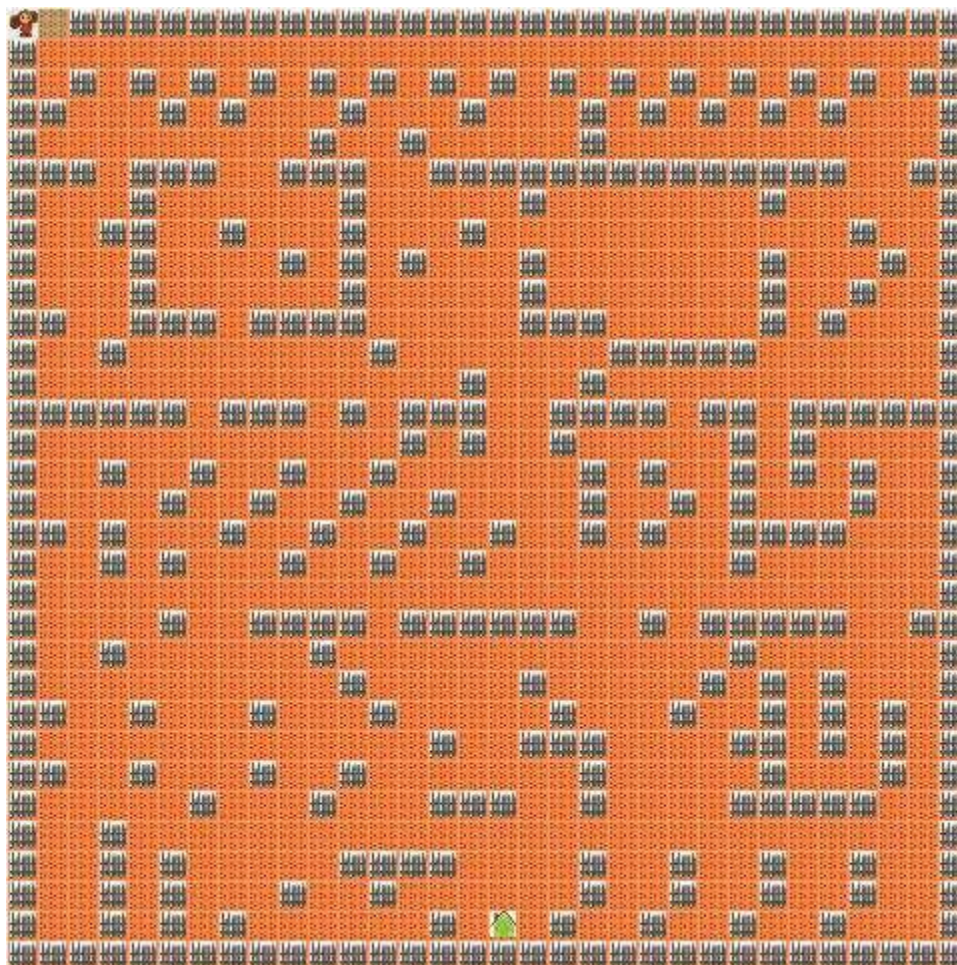


Рис.1. Поле, по которому ходит робот

1.2. Автомат Мили

Автомат Мили [2] – это конечный автомат, генерирующий выходные воздействия в зависимости от текущего состояния автомата и входных воздействий.

Формально, автомат Мили $A = \langle S, q_0 \in S, X, Y, \delta \rangle$, где

- S – множество состояний;
- q_0 – стартовое состояние;
- X – множество входных воздействий;
- Y – множество выходных воздействий;
- $\delta: S \times X \rightarrow S \times Y$ – функция переходов.

Заметим, что в данной задаче входные воздействия – наличие или отсутствие препятствия перед роботом, выходные – те действия, которые может совершать робот.

В памяти компьютера автомат Мили хранится в виде графа переходов. Графически граф переходов в отчете будет изображаться как на рис.2.

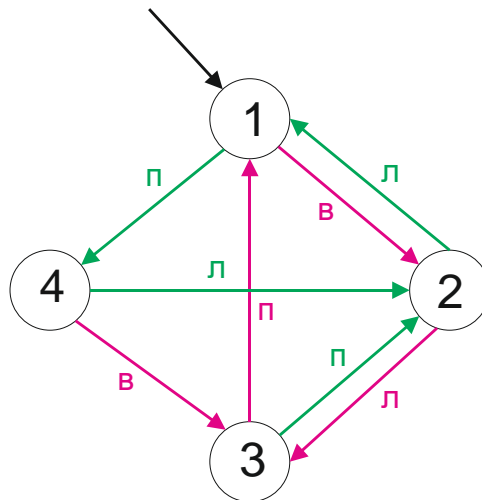


Рис.2. Пример автомата Мили

Начальное состояние обозначается ведущей в него черной стрелкой.

Входные воздействия обозначаются цветом:

- Розовый цвет – впереди робота нет препятствия;
- Зеленый цвет – впереди робота находится препятствие.

Выходные воздействия:

- В – сделать шаг вперед;
- П – повернуть направо;
- Л – повернуть налево.

2. Алгоритм имитации отжига

Метод имитации отжига [3] служит для поиска глобального минимума некоторой функции $f(x)$, заданной для x из некоторого пространства S . Элементы множества S представляют собой состояния воображаемой физической системы, а значения функции f в этих точках используются как энергия системы $E = f(x)$. В каждый момент предполагается, что задана температура системы, как правило, уменьшающаяся с течением времени. Находясь в состоянии при температуре T , следующее состояние системы выбирается в соответствии с заданным порождающим семейством вероятностных распределений $G(x, T)$. После генерации нового состояния x^* , система с некоторой вероятностью $h(\Delta E, T)$ переходит к следующему шагу в состояние x^* , в противном случае процесс генерации x^* повторяется. В данной работе рассматривалась $h(\Delta E, T) = \frac{1}{1 + \exp(\frac{\Delta E}{T})}$.

Можно более формально записать алгоритм имитации отжига. На первом шаге случайным образом выбирается начальная точка $x = x_0$, $x_0 \in S$. Текущее значение энергии устанавливается в значение $f(x_0)$. Далее, на каждой следующей итерации (с номером k) выполняется следующая последовательность действий:

- генерируется новая точка x^* ;
- вычисляется значение функции в новой точке $E^* = f(x^*)$;
- если $E^* > E$, то устанавливается $x \leftarrow x^*$, $E \leftarrow E^*$ и происходит переход к следующей итерации;
- генерируется случайное число α из промежутка $[0, 1]$;
- если $\alpha < \frac{1}{1 + \exp(\frac{\Delta E}{T(k)})}$, то устанавливается $x \leftarrow x^*$, $E \leftarrow E^*$ и происходит переход к следующей итерации, иначе итерация повторяется заново.

2.1. Сверхбыстрый отжиг

При использовании метода сверхбыстрого отжига [3] пространство S считается состоящим из D – мерных векторов $(x_1 \dots x_D)$, где $x_i \in [A_i, B_i]$. Семейство распределений строится следующим образом. Вводится функция

$$g_{(i,T)}(y_i) = \frac{1}{2(|y_i| + T) \ln(1 + \frac{1}{T})}$$

В качестве y_i используется $\frac{\Delta x_i}{B_i - A_i}$. Таким образом, новое значение x_i^* вычисляется по формуле $x_i^* = x_i + z_i(B_i - A_i)$, где z_i – случайная величина с плотностью $g_{(i,T)}$ на $[-1,1]$. При этом, выходящие за границы интервала значения параметра приравниваются к соответствующим границам. Такая случайная величина моделируется:

$$z_i = \operatorname{sgn}(\alpha - 0.5) T \left(\left(1 + \frac{1}{T}\right)^{|2\alpha-1|} - 1 \right),$$

где α – независимая случайная величина, равномерно распределенная на $[0,1]$.

Закон изменения температуры выражается следующим соотношением:

$$T(k) = T_0 \exp\left(-ck^{\frac{1}{D}}\right), \text{ где } c > 0.$$

2.2. Метод Xin Yao

Алгоритм Xin Yao [4] был получен повторным применением идеи предыдущего алгоритма. В качестве $g_{(i,T)}$ выбирается

$$g_{(i,T)}(y_i) = \frac{1}{2 \left(|y_i| + \frac{1}{\ln\left(\frac{1}{T}\right)} \right) \ln \left(1 + \ln \left(\frac{1}{T} \right) \right)}$$

Тогда z_i можно промоделировать

$$z_i = \operatorname{sgn}(\alpha - 0.5) \frac{1}{\ln\left(\frac{1}{T}\right)} \left(\left(1 + \ln\left(\frac{1}{T}\right)\right)^{|2\alpha-1|} - 1 \right),$$

где α – независимая, случайная величина, равномерно распределенная на $[0,1]$.

Закон изменения температуры выражается следующим соотношением:

$$T(k) = T_0 \exp\left(-\exp\left(bk^{\frac{1}{D}}\right)\right), \text{ где } b > 0.$$

То есть скорость убывания температуры в методе Xin Yao значительно выше, чем в методе сверхбыстрого отжига.

3. Реализация

В данном разделе говорится о деталях реализации. В том числе, о задании функции приспособленности $fitness(A)$, значение которой соответствует значению энергии системы в точке пространства A . Также описывается, как было задано само пространство и как получалась новая точка.

3.1. Функция приспособленности

В работе была использована следующая функция приспособленности:

$$fitness(A) = \begin{cases} 400 - k & , \text{если робот дошел до цели} \\ 200 - l + \frac{steps}{201} & , \text{если робот до цели не дошел} \end{cases}$$

где A – полученный автомат, k – число шагов, за которые робот дошел до цели, l – минимальное расстояние от робота до цели на всем пути, $steps$ – число шагов, за которые было получено минимальное расстояние от робота до цели.

3.2. Генерация новой точки

В данной работе пространство S задавалось 32–мерными векторами. Каждой точке этого пространства соответствовал автомат Мили из восьми состояний, где каждому из состояний соответствовало четыре числа:

- в какую вершину совершается переход, в случае отсутствия препятствия;
- какое действие совершает робот в случае отсутствия препятствия;
- в какую вершину совершается переход, в случае наличия препятствия;
- какое действие совершает робот в случае наличия препятствия.

Каждая координата новой точки получалась из соответствующей координаты старой точки, независимо от других в заданном промежутке.

4. Результаты

Программа запускала алгоритм для генерации автоматов из шести, восьми и десяти состояний. Значения функции приспособленности для автоматов из шести состояний в среднем были ниже, чем для автоматов из восьми и десяти состояний. Для автоматов из восьми и десяти состояний значения функции приспособленности в среднем почти не отличались. Однако, для генерации автоматов из десяти состояний требовалось больше времени. Поэтому было принято решение рассматривать в данной работе построение автоматов из восьми состояний.

4.1. Начальная температура для сверхбыстрого отжига

Для подбора оптимальной начальной температуры для сверхбыстрого отжига было сделано по 20 запусков генерации автомата при использовании сверхбыстрого отжига при трех различных температурах: $T_1 = 1$, $T_2 = 1000$, $T_3 = 100000$. Результаты представлены на рис.3.

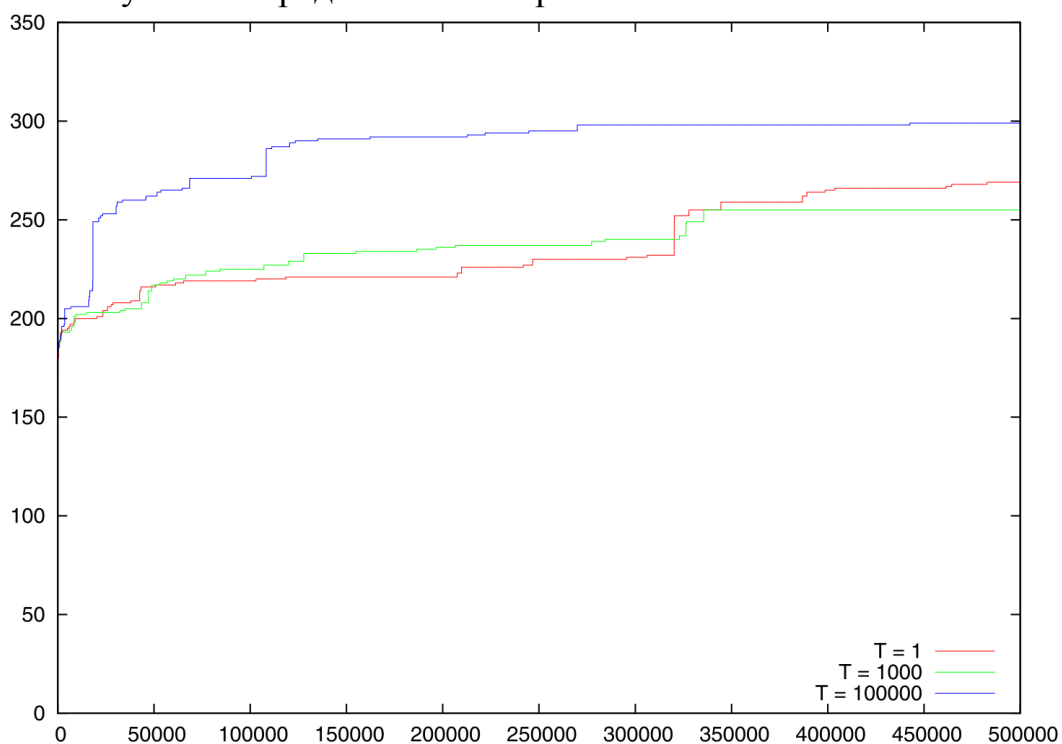


Рис.3. Зависимость эффективности метода сверхбыстрого отжига от начальной температуры

Как видно из графика наилучший результат достигается при температуре $T = 100000$. Далее, для определения более точной оптимальной начальной температуры, были сделаны измерения при температурах $T_1 = 50000$, $T_2 = 90000$, $T_3 = 100000$, $T_4 = 110000$, $T_5 = 150000$. Значения функции

приспособленности в среднем при температурах T_1 , T_5 были ниже, чем при температурах T_2 , T_3 , T_4 , которые, в свою очередь, почти не отличались. Поэтому, в качестве оптимальной температуры рассматривалась температура $T = 100000$.

4.2. Начальная температура для метода Xin Yao

Для подбора оптимальной начальной температуры для метода Xin Yao было сделано по 20 запусков генерации автомата при использовании метода Xin Yao при трех различных температурах: $T_1 = 1$, $T_2 = 10$, $T_3 = 1000$. Результаты представлены на рис.4.

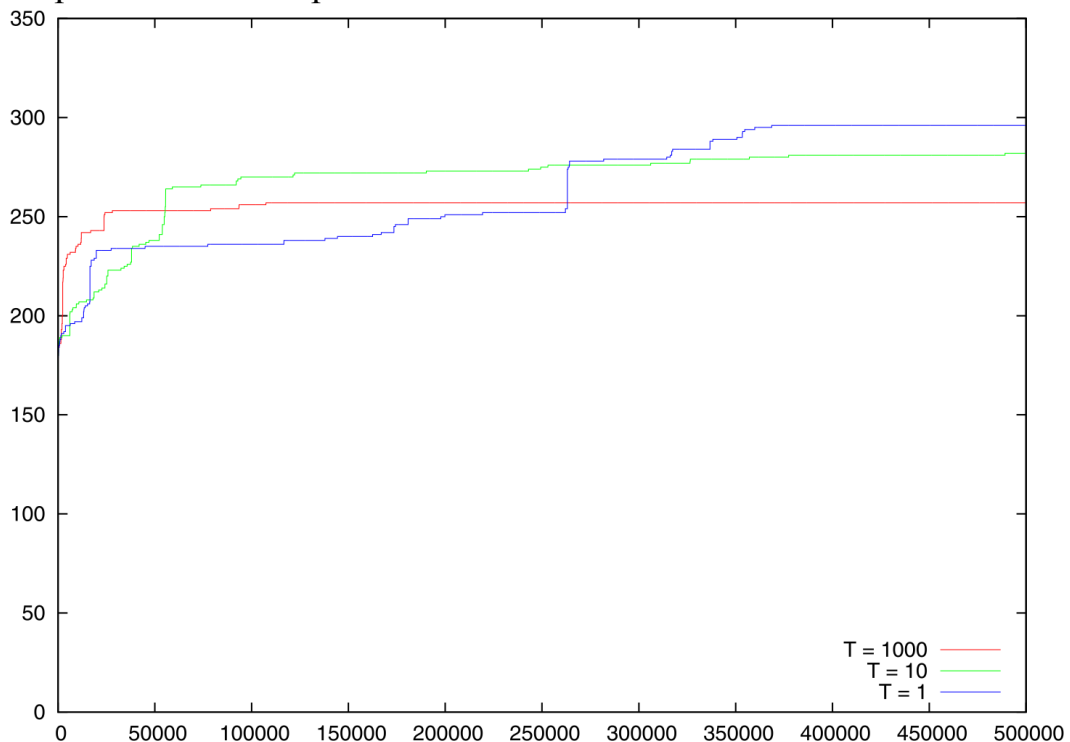


Рис.4. Зависимость эффективности метода Xin Yao от температуры.

Как видно из графика наилучший результат достигается при температуре $T = 1$. Далее, для определения более точной оптимальной начальной температуры, были сделаны измерения при температурах $T_1 = 0.5$, $T_2 = 1$, $T_3 = 1.5$, $T_4 = 2$. Значения функции приспособленности в среднем при температурах T_3 , T_4 были ниже, чем при температурах T_1 , T_2 , которые, в свою очередь, почти не отличались. Поэтому, в качестве оптимальной температуры, рассматривалась температура $T = 1$.

4.3. Сравнение экспериментальных данных методов

Программа произвела 50 запусков генерации автомата при помощи сверхбыстрого отжига при температуре $T = 100000$ и 50 запусков при помощи метода Xin Yao при температуре $T = 1$.

Для каждого метода было посчитано среднее арифметическое значений функций приспособленности в зависимости от числа итераций и был построен соответствующий график. Полученные графики представлены на рис.5.

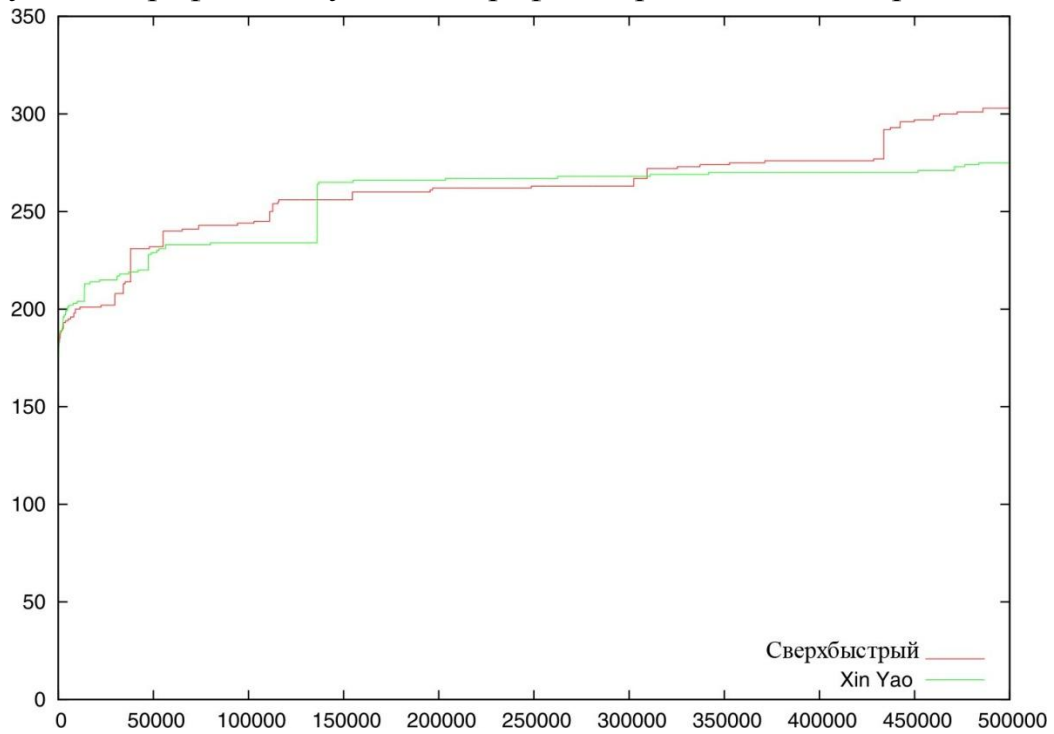


Рис.5. Сравнение функций приспособленности Сверхбыстрого отжига и метода Xin Yao.

5. Лучший полученный автомат

Робот доходит до цели за 81 действие. Автомат представлен на рис.6. Путь, пройденный роботом, приведен на рис.7.

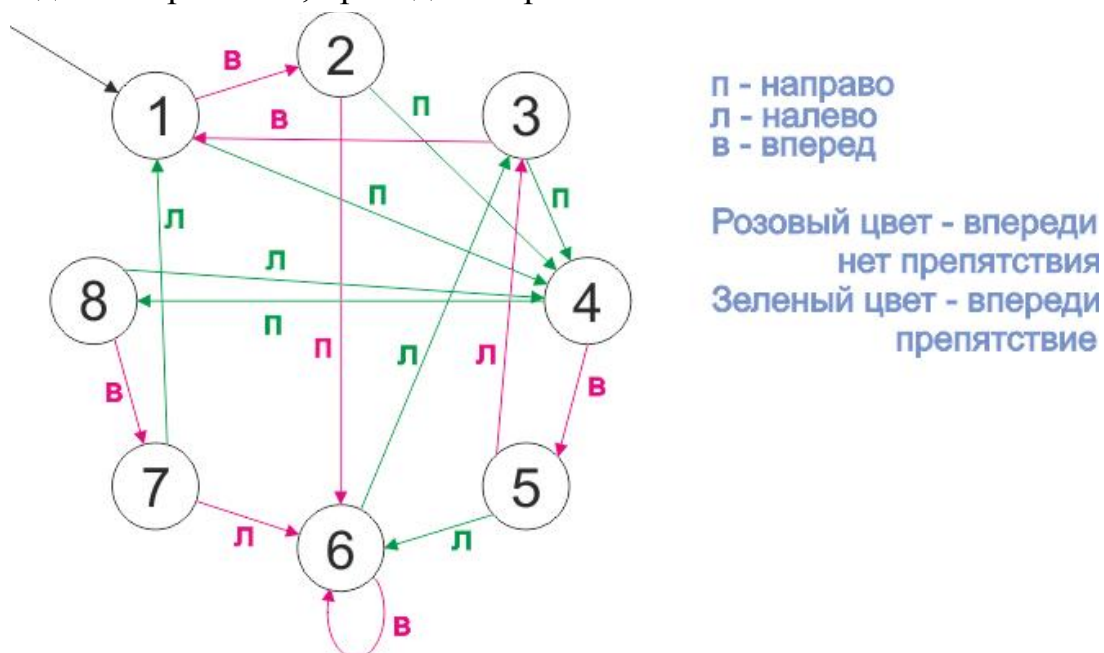


Рис.6. Полученный автомат из восьми состояний.

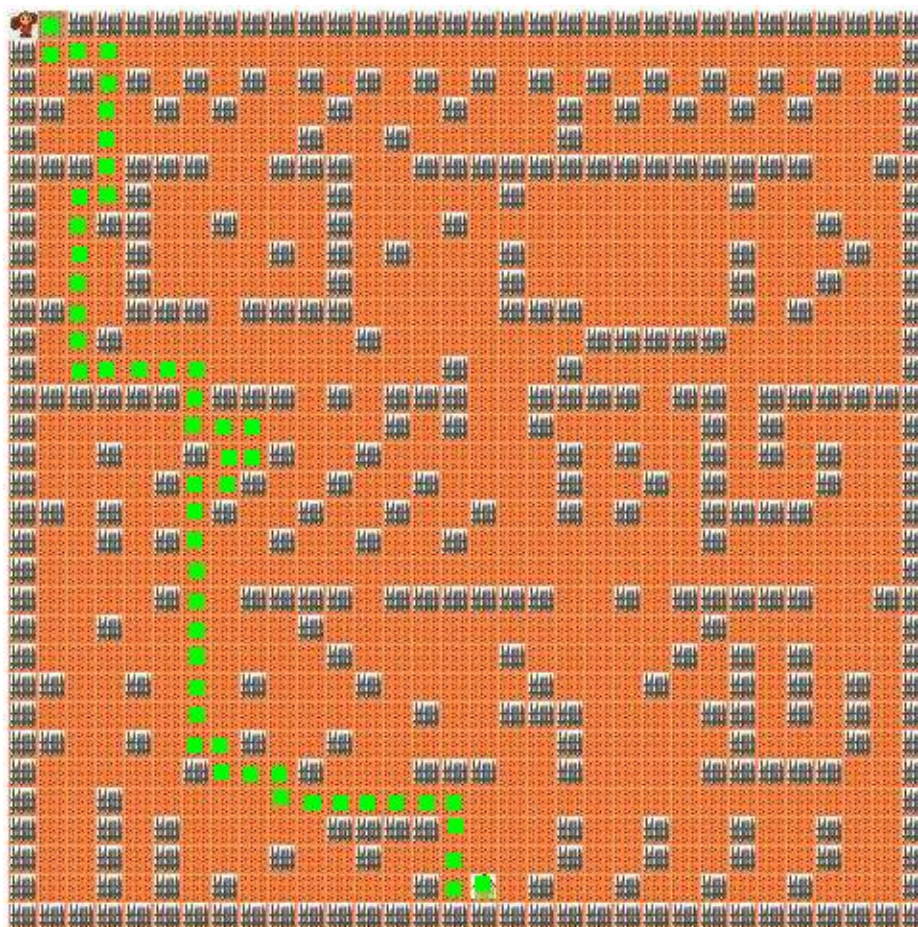


Рис.7. Путь, пройденный роботом.

6. Заключение

Из результатов видно, что эффективность сверхбыстрого отжига в среднем выше, чем метода Xin Yao. Это означает, что увеличение скорости убывания температуры вовсе не означает ускорения в решении задачи. Таким образом, несмотря на то, что этот процесс можно итерировать до бесконечности, получая законы изменения температуры вроде

$$T(k) = T_0 \exp \left(- \exp \left(\exp \left(\dots \exp \left(bk^{\frac{1}{D}} \right) \dots \right) \right) \right)$$

ценность таких «улучшений» представляется сомнительной. Более того, легко видеть, что в пределе это приводит к тривиальному методу случайного поиска, которым является метод отжига при $T = 0$.

7. Источники

- [1] Koza, J.R. (1994), Genetic Programming II: Automatic Discovery of Reusable Programs, MIT Pres
- [2] Поликарпова Н.И., Шалыто А.А. Автоматное программирование. – СПб.: Питер, 2009.
- [3] Метод имитации отжига. Конспект лекций А.Лопатина.
<http://rain.ifmo.ru/~buzdalov/lab-2011/books/annealing.pdf>
- [4] Xin Yao, “A new simulated annealing algorithm”, International Journal of Computer Mathematics, vol.56, pp.161– 168, 1995