

## Поиск пути к цели с минимальным Манхэттен-расстоянием

В этой задаче, точка старта робота и цель, которую он должен достичь, являются предварительно заданными.

Задача состоит в том, чтобы найти робота, который достигает цель с минимальным Манхэттен-расстоянием. Примеры на рисунках ниже.

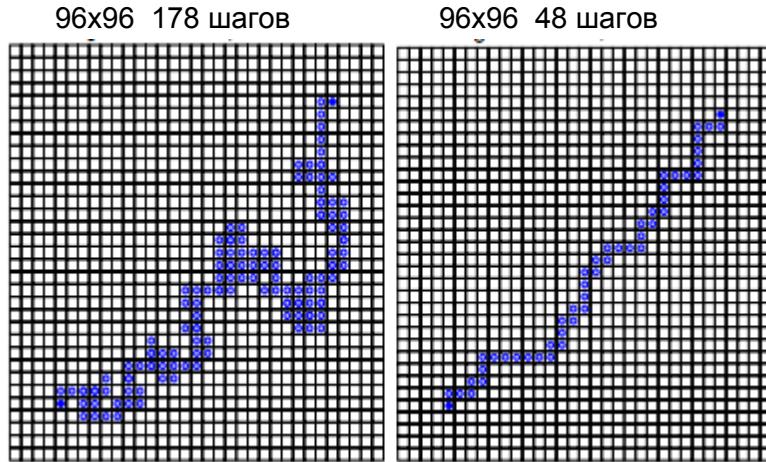


Рис. 4: В сетке 96x96, с начальными координатами (24, 24) робот отправляется к цели с координатами (72, 72), не имея никакой предварительной информации. На рисунке слева минимальный путь среди 100 случайных. Справа - минимальный путь робота, найденный после эволюционного обучения, как показано на Рис.3. (Крайние зоны не указаны.)

## 8 Задача о коммивояжёре (TSP)

Есть  $N$  городов с заданными координатами, задача коммивояжёра (TSP - Traveling Salesperson Problem) состоит в том, чтобы посетить все города один раз, при этом путь должен быть минимальным.

Теперь возьмём 4 города - A, B, C и D в качестве простейшего примера. Предположим что города расположены соответствующим образом, например:

	(x, y)
A	(0.83, 7.79)
B	(3.28, 8.32)
C	(1.52, 4.48)
D	(7.65, 3.46)

Тогда Евклидово расстояние между всеми возможными парами городов вычисляется по формуле:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (2)$$

где  $r_{ij}$  расстояние между городом  $i$  и городом  $j$ , и  $(x_i, y_i)$  и  $(x_j, y_j)$  - координаты городов  $i$  и  $j$  соответственно. Расстояние:

	A	B	C	D
A	0.000	2.505	3.382	8.074
B	2.505	0.000	4.232	6.539
C	3.382	4.232	0.000	6.214
D	8.074	6.539	6.214	0.00

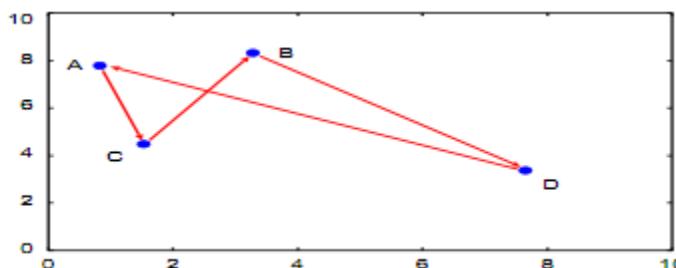


Рис. 5: Пример 4 городов и возможного маршрута

Все возможные маршруты в этом примере:

(A-B-C-D-A), (A-B-D-C-A), (A-C-B-D-A), (A-C-D-B-A), (A-D-B-C-A), и (A-D-C-B-A)

Заметьте, что длина пары маршрутов одинакова, например (A-B-C-D-A) и (A-D-C-B-A). То есть у нас есть  $3!/2 = 3$  маршрута в данном примере.

Рассмотрим теперь маршрут A-C-B-D-A, он показан на Рис.5.

Длина маршрута на рисунке:

$$r_{A-C-B-D-A} = 3.382 + 4.232 + 6.539 + 8.074 = 22.227$$

Таким же образом мы можем вычислить два других маршрута:

$$r_{A-B-D-C-A} = 2.505 + 6.539 + 6.214 + 3.382 = 15.640$$

$$r_{A-B-C-D-A} = 2.505 + 4.232 + 6.214 + 8.074 = 21.02$$

Итак, маршрут с минимальной длиной пути - это A-B-D-C-A (или A-C-D-B-A).

Но что произойдёт, если мы имеем большее число городов? Теперь вы знаете, даже в случае 10 городов, мы имеем  $9!/2 = 181,440$  возможных различных маршрутов. Вы хотите вычислять длины всех этих маршрутов? Конечно же нет! Более того, как быть в случае 1000 городов?

Тогда давайте применим наш эволюционный алгоритм. Заметьте, что хромосомы выглядящие как

(B D C)

для маршрута A-B-D-C-A, и

(D C B)

для маршрута A-D-C-B-A, не будут работать, т.к. возможный потомок после одноточечного скрещивания (*one-point-crossover*) будет

(B C B) и (D D C)

что невозможно, потому что, происходит посещение одного города дважды, пренебрегая другим городом вообще.

Тогда возможный вид хромосомы состоит в следующем:

*Шаг-1. Устанавливаем  $i = 1$ .*

*Шаг-2. Если  $i$ -й ген  $n$ , тогда  $n$ -й город в списке городов является посещённым.*

*Шаг-3. Удаляем город из списка городов.*

*Шаг-4. Устанавливаем  $i = i + 1$  и повторяем Шаг-2 - Шаг-4 пока  $i < n$ .*

Например, список городов, кроме начального города А:

{B, C, D}

хромосома: (121) задаёт маршрут:

A-B-D-C-A

Обратите внимание, что гены могут быть любым целым числом и мутация может происходить, просто заменив ген на другое случайное число. Вероятность может быть 1/number-of-genes (вы можете изменить отношение в качестве эксперимента)

#### Задание 4 (TSP)

(1) Создать 14 городов со случайными координатами  $(x_i, y_i)$ .

(2) Подсчитать расстояние между всеми возможными парами городов.

(3) Эволюционировать их пока расстояние маршрута не сойдётся в одно значение.

(4) Повторять (3) до момента, когда фитнесс-значение (=всё расстояние маршрута) не сойдётся в одно значение.

**Результаты, которые вы должны предоставить**

- Координаты всех городов
- Матрица расстояний между любыми двумя городами
- Графическое представление расположения городов и кратчайший маршрут

## 8 Задача о рюкзаке

Предположим, что имеется  $n$  предметов, которые можно положить в рюкзак. Каждый предмет имеет вес  $w_i$  и коэффициент полезности  $p_i$ . Далее для каждого  $i$ -го предмета подбираются неотрицательные целые значения  $x_i$ , где  $i=1,2,\dots,n$ . Цель заключается в поиске максимума для выражения:

$$\sum_{i=1}^n x_i p_i. \quad (3)$$

причём так, чтобы

$$\sum_{i=1}^n x_i w_i < C \quad (4)$$

где  $C$  – максимально возможный вес рюкзака.

Применить генетический алгоритм в данном случае достаточно просто. Наша хромосома задается в форме:

$$(x_1 x_2 x_3 \dots x_n) \quad (5)$$

где  $x_i$  - количество  $i$ -ых предметов, помещенных в рюкзак.

### Удаление непригодных хромосом

Необходимо отметить что, если хромосома не удовлетворяет выражению (4), то такая хромосома удаляется, а процедура генерации хромосомы потомка (крессовер, мутация и т.п.) повторяется снова до тех пор, пока не будет получена подходящая хромосома потомка.

**Задание 5 (Задача о рюкзаке)** Предположим, размер рюкзака - 60.

(1) Создайте 100 предметов, присвойте случайные значения  $p_i$  и  $w_i$  от 0 до 1 для каждого предмета. Например:

i	$p_i$	$w_i$
1st	0.37	0.62
2nd	0.52	0.45
3rd	0.95	0.38
...	...	...
100th	0.72	0.3

(2) Создайте 40 хромосом, каждая из которых включает 100 целочисленных генов, например

$$(5, 7, 13, \dots, 2)$$

что означает 5 первых предметов, 7 вторых, 13 третьих, . . . , 2 сотых.

(3) Попробуйте проверить результат. Замените один предмет на предмет с  $p=0.99$  и  $w=0.01$ . Представьте себе этот предмет как алмаз, маленький и очень ценный. Тогда все предметы должны сойтись на нём. В этом случае вы будете точно знать результаты.

(4) Проведите эволюцию, и нарисуйте график максимальный фитнесс vs. поколение, средний фитнесс vs. поколение.

(5) Визуализируйте содержимое рюкзака.

## 10 Отображение Sammon при помощи генетического алгоритма

В данном разделе рассматривается т.н. Отображение Sammon. Отображение Sammon – это отображение набора точек многомерного пространства в двумерное пространство с тем, чтобы это возможно сохранением соотношения расстояний между элементами исходного пространства. Другими словами, задача состоит в аппроксимации расстояний исходного  $n$ -мерного пространства соответствующими расстояниями в 2-мерном пространстве с минимальными потерями.

Метод был предложен в 1980-х в качестве задачи оптимизации, к которой был применен не самый простой алгоритм Наискорейшего Спуска из области Исследования Операций. С другой стороны применить Эволюционные Вычисления в этой ситуации гораздо проще. Разберемся теперь, что представляет собой Отображение Sammon:

### Алгоритм (Отображение Sammon)

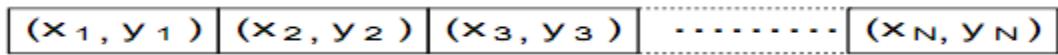
1. Допустим, заданы  $N$  точек в  $n$ -мерном пространстве.
2. Рассчитаем матрицу расстояний  $R$  ( $N \times N$ ), где элемент в позиции  $i-j$  – Евклидово расстояние между  $i$ -ой и  $j$ -ой точкой.
3. Определим также  $N$  точек в двумерном пространстве и для начала распределим их случайным образом.
4. Высчитаем матрицу расстояний  $Q$ , аналогично матрице  $R$ .
5. Рассчитывается матрица ошибок, как  $P = R - Q$ .
6. Осуществляется поиск позиций  $N$  точек в двумерном пространстве таким образом, чтобы минимизировать суммарное значение элементов матрицы  $P$ .

Таким образом, мы имеем дело с задачей оптимизации, которая, как нам известно, может быть легко решена при помощи Эволюционных Вычислений. Наша задача заключается в поиске  $N$  точек в 2-мерном пространстве, которые соответствовали бы  $N$  точкам в  $n$ -мерном пространстве. Причем при отображении необходимо сохранить, насколько это возможно, все соотношения расстояний между точками исходного  $n$ -мерного пространства, т.е. выполнить аппроксимацию с минимальной ошибкой.

При реализации генетического алгоритма для решения поставленной задачи хромосомы должны включать  $n$  генов, каждый из которых соответствует искомой координате  $x-y$  точки в 2-мерном пространстве. Применяется операция равномерного скрещивания, а время от времени – мутация, замещающая один ген случайной координатой  $x-y$  (см. Рисунок 2, см. Рисунок представленный ниже).

Пример для  $49^2 = 2401$  мерного пространства:

Хромосома:



Равномерное скрещивание:

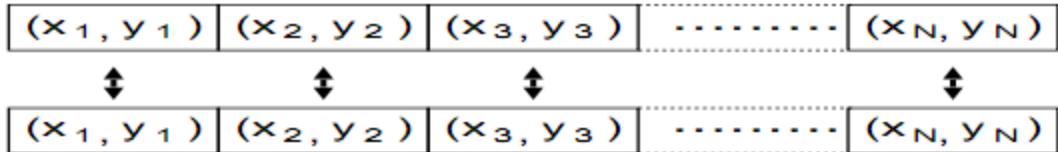


Рис. 6: Представление хромосомы и равномерное скрещивание.

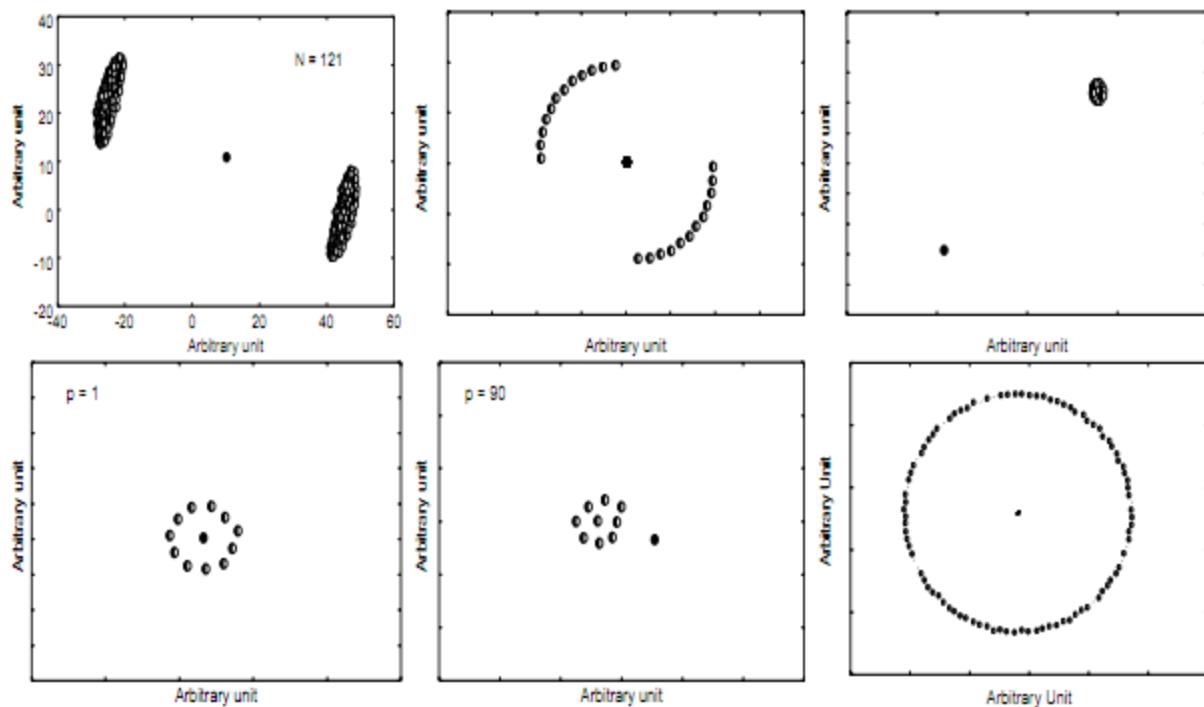


Рис. 7: Шесть примеров Отображения из 2401-мерного пространства в 2-мерное пространство. Разъяснения приведены в тексте.