



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА «ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ЭВМ И ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ» (ИУ7)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Информатика и вычислительная техника

ОТЧЕТ

по лабораторной работе № 6

Название: Муравьиный алгоритм

Дисциплина: Анализ алгоритмов

Студент

ИУ7-52Б

(Группа)

(Подпись, дата)

Короткая В. М.

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

Волкова Л.Л.

(И.О. Фамилия)

Москва, 2021

Содержание

Введение	3
1 Аналитическая часть	5
1.1 Задача коммивояжера	5
1.2 Решение полным перебором	5
1.3 Решение муравьиным алгоритмом	6
1.4 Вывод	8
2 Конструкторская часть	9
2.1 Разработка алгоритмов	9
2.2 Вывод	11
3 Технологическая часть	12
3.1 Выбор ЯП	12
3.2 Сведения о модулях программы	12
3.3 Вывод	12
4 Экспериментальная часть	13
4.1 Сравнение времени работы муравьиного алгоритма и полного перебора	13
4.2 Результат работы программы	13
4.3 Параметризация муравьиного алгоритма	13
4.4 Вывод	13
Заключение	14
Список литературы	15

Введение

Муравья нельзя назвать сообразительным. Отдельный муравей не в состоянии принять ни малейшего решения. Дело в том, что он устроен крайне примитивно: все его действия сводятся к элементарным реакциям на окружающую обстановку и своих собратьев. Муравей не способен анализировать, делать выводы и искать решения.

Эти факты, однако, никак не согласуются с успешностью муравьев как вида. Они существуют на планете более 100 миллионов лет, строят огромные жилища, обеспечивают их всем необходимым и даже ведут настоящие войны. В сравнении с полной беспомощностью отдельных особей, достижения муравьев кажутся немыслимыми.

Добиться таких успехов муравьи способны благодаря своей социальности. Они живут только в коллективах – колониях. Все муравьи колонии формируют так называемый роевой интеллект. Особи, составляющие колонию, не должны быть умными: они должны лишь взаимодействовать по определенным – крайне простым – правилам, и тогда колония целиком будет эффективна.

В колонии нет доминирующих особей, нет начальников и подчиненных, нет лидеров, которые раздают указания и координируют действия. Колония является полностью самоорганизующейся. Каждый из муравьев обладает информацией только о локальной обстановке, не один из них не имеет представления обо всей ситуации в целом – только о том, что узнал сам или от своих сородичей, явно или неявно. На неявных взаимодействиях муравьев, называемых стигмергией, основаны механизмы поиска кратчайшего пути от муравейника до источника пищи.

Каждый раз проходя от муравейника до пищи и обратно, муравьи оставляют за собой дорожку феромонов. Другие муравьи, почувствовав такие следы на земле, будут инстинктивно устремляться к нему. Поскольку эти муравьи тоже оставляют за собой дорожки феромонов, то

чем больше муравьев проходит по определенному пути, тем более привлекательным он становится для их сородичей. При этом, чем короче путь до источника пищи, тем меньше времени требуется муравьям на него – а следовательно, тем быстрее оставленные на нем следы становятся заметными.

В 1992 году в своей диссертации Марко Дориго (Marco Dorigo) предложил заимствовать описанный природный механизм для решения задач оптимизации. Имитируя поведение колонии муравьев в природе, муравьиные алгоритмы используют многоагентные системы, агенты которых функционируют по крайне простым правилам. Они крайне эффективны при решении сложных комбинаторных задач – таких, например, как задача коммивояжера, первая из решенных с использованием данного типа алгоритмов.

Целью данной работы является изучение и реализация двух алгоритмов:

- полный перебор;
- муравьиный алгоритм.

В рамках выполнения работы необходимо решить следующие задачи:

- изучить два, описанных выше алгоритма для решения задачи коммивояжера;
- применить изученные знания для реализации двух алгоритмов;
- составить схемы алгоритмов;
- провести параметризацию муравьиного алгоритма;
- провести сравнительный анализ скорости работы реализованных алгоритмов;
- выбрать и обосновать выбор языка программирования, для решения поставленной задачи.

1 | Аналитическая часть

1.1 Задача коммивояжера

В 19-м и 20-м веке по городам ездили коммивояжеры (сейчас их называют «торговые представители»). Они ходили по домам и предлагали людям купить разные товары. Тактика была такой: коммивояжёр приезжал в город, обходил большинство домов и отправлялся в следующий город. Города были небольшими, поэтому обойти всё было вполне реально.

Чем больше городов посетит коммивояжёр, тем больше домов он сможет обойти и больше заработать с продаж.

В задаче коммивояжера рассматривается n городов и матрица попарных расстояний между ними. Требуется найти такой порядок посещения городов, чтобы суммарное пройденное расстояние было минимальным, каждый город посещался ровно один раз и коммивояжер вернулся в тот город, с которого начал свой маршрут. Другими словами, во взвешенном полном графе требуется найти **гамильтонов цикл минимального веса**.

1.2 Решение полным перебором

Эту задачу возможно решить полным перебором т. е. разобрать все возможные варианты и выбрать оптимальный. Но проблема такого решения в том, что с увеличением количества городов, время выполнения будет расти.

Хотя такой подход и гарантирует точное решение задачи, уже при небольшом числе городов решение задачи за допустимое время не возможно.

1.3 Решение муравьиным алгоритмом

В то время как простой метод перебора всех вариантов чрезвычайно неэффективен при большом количестве городов, эффективными признаются решения, гарантирующие получение ответа за время, ограниченное полиномом от размерности задачи.

В основе алгоритма лежит поведение муравьиной колонии – маркировка более удачных путей большим количеством феромона.

Каждый муравей хранит в памяти список пройденных им узлов. Этот список называют списком запретов (tabu list) или просто памятью муравья. Выбирая узел для следующего шага, муравей «помнит» об уже пройденных узлах и не рассматривает их в качестве возможных для перехода. На каждом шаге список запретов пополняется новым узлом, а перед новой итерацией алгоритма – то есть перед тем, как муравей вновь проходит путь – он опустошается.

Кроме списка запретов, при выборе узла для перехода муравей руководствуется «привлекательностью» ребер, которые он может пройти. Она зависит, во-первых, от расстояния между узлами (то есть от веса ребра), а во-вторых, от следов феромонов, оставленных на ребре прошедшими по нему ранее муравьями. Естественно, что в отличие от весов ребер, которые являются константными, следы феромонов обновляются на каждой итерации алгоритма: как и в природе, со временем следы испаряются, а проходящие муравьи, напротив, усиливают их.

Процесс поиска кратчайшего пути от колонии до источника питания (рис 1.1):

1. первый муравей находит источник пищи (F) любым способом (a), а затем возвращается к гнезду (N), оставив за собой тропу из феромонов (b);
2. затем муравьи выбирают один из четырёх возможных путей, затем укрепляют его и делают привлекательным;
3. муравьи выбирают кратчайший маршрут, так как феромоны с более длинных путей быстрее испаряются.

Вероятность перехода из вершины i в вершину j определяется по формуле 1.1.

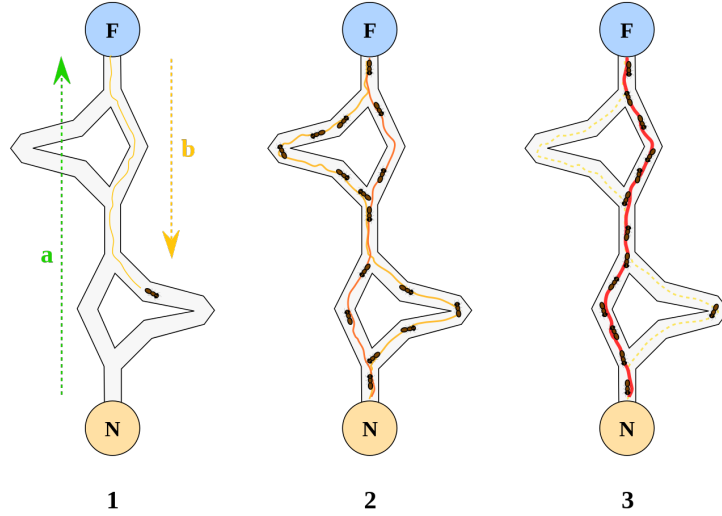


Рис. 1.1: Работа муравьиной колонии

$$p_{i,j} = \frac{(\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)}{\sum (\tau_{i,j}^\alpha)(\eta_{i,j}^\beta)} \quad (1.1)$$

где $\tau_{i,j}$ — расстояние от города i до j ;

$\eta_{i,j}$ — количество феромонов на ребре ij ;

α — параметр влияния длины пути;

β — параметр влияния феромона.

Уровень феромона обновляется в соответствии с формулой 1.2

$$\tau_{i,j} = (1 - \rho)\tau_{i,j} + \Delta\tau_{i,j}, \quad (1.2)$$

где ρ — доля феромона, которая испарится;

$\tau_{i,j}$ — количество феромона на дуге ij ;

$\Delta\tau_{i,j}$ — количество отложенного феромона, вычисляется по формуле

1.3.

$$\Delta\tau_{i,j} = \tau_{i,j}^0 + \tau_{i,j}^1 + \dots + \tau_{i,j}^k \quad (1.3)$$

где k — количество муравьев в вершине графа с индексами i и j .

Описание поведения муравьев при выборе пути.

- Муравьи имеют собственную «память». Поскольку каждый город может быть посещён только один раз, то у каждого муравья есть список уже посещенных городов - список запретов. Обозначим через J_{ik} список городов, которые необходимо посетить муравью k , находящемуся в городе i .
- Муравьи обладают «зрением» - видимость есть эвристическое желание посетить город j , если муравей находится в городе i . Будем считать, что видимость обратно пропорциональна расстоянию между городами.
- Муравьи обладают «обонянием» - они могут улавливать след феромона, подтверждающий желание посетить город j из города i на основании опыта других муравьёв. Количество феромона на ребре (i, j) в момент времени t обозначим через $\tau_{i,j}(t)$.
- Пройдя ребро (i, j) , муравей откладывает на нём некоторое количество феромона, которое должно быть связано с оптимальностью сделанного выбора. Пусть $T_k(t)$ есть маршрут, пройденный муравьем k к моменту времени t , $L_k(t)$ - длина этого маршрута, а Q - параметр, имеющий значение порядка длины оптимального пути. Тогда откладываемое количество феромона может быть задано формулой 1.4.

$$\Delta\tau_{i,j}^k = \begin{cases} Q/L_k, & \text{если } k\text{-ый муравей прошел по ребру } ij; \\ 0, & \text{иначе} \end{cases} \quad (1.4)$$

где Q - количество феромона, переносимого муравьем.

1.4 Вывод

В данном разделе были рассмотрены основополагающие материалы, которые в дальнейшем потребуются при реализации алгоритмов для решения задачи коммивояжера.

2 | Конструкторская часть

2.1 Разработка алгоритмов

В данном разделе будут рассмотрены схемы.

На рис. 2.1 показана схема алгоритма полного перебора.

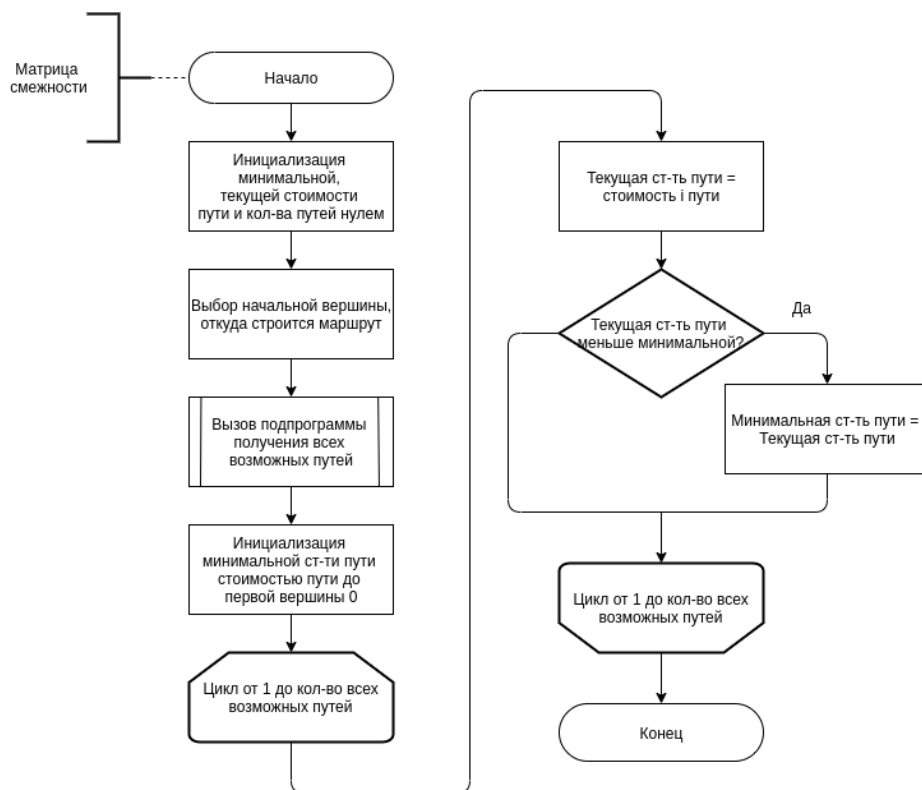


Рис. 2.1: Схема алгоритма полного перебора

На рис. 2.2 показана схема муравьиного алгоритма.

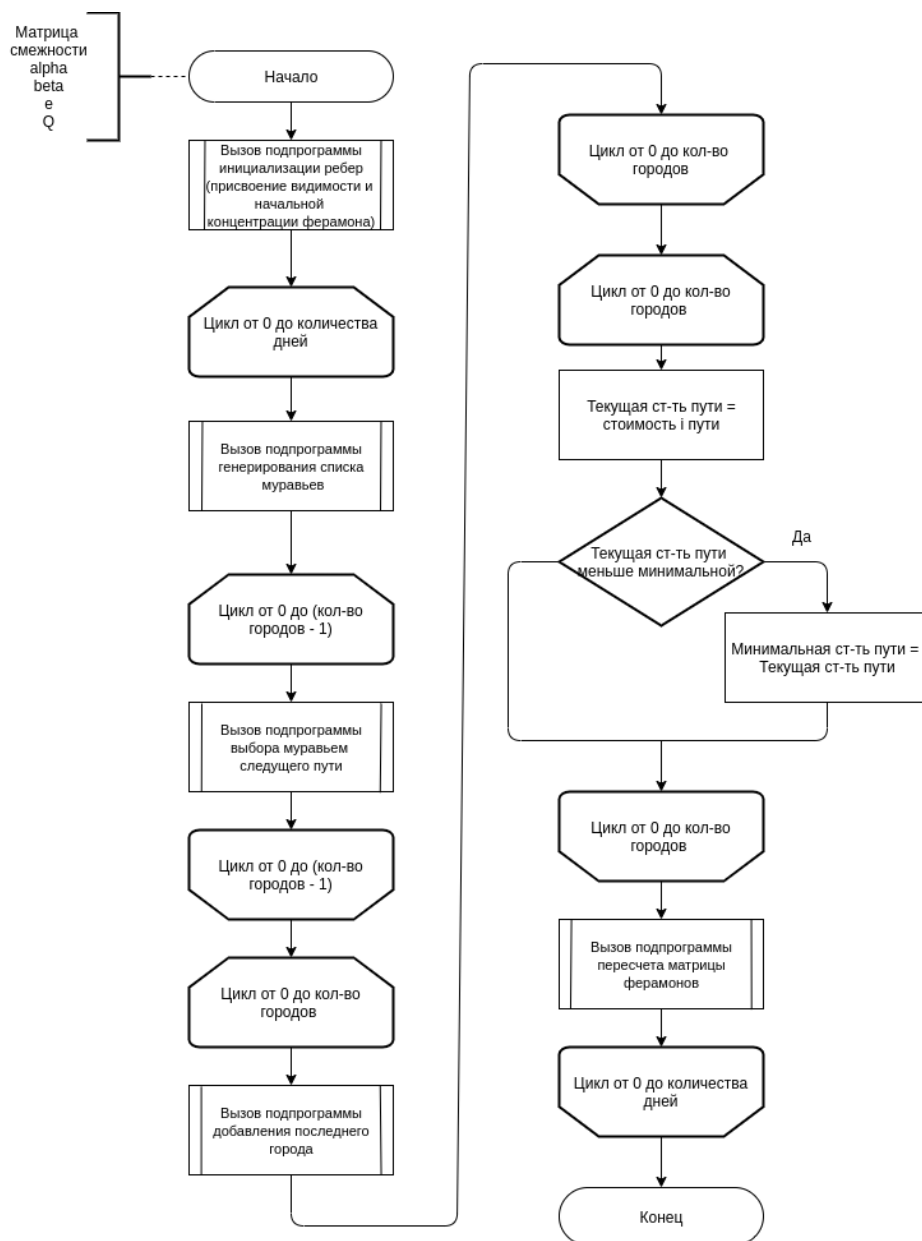


Рис. 2.2: Схема муравьиного алгоритма

2.2 Вывод

В данном разделе были рассмотрены схемы алгоритма полного перебора 2.1 и схема муравьиного алгоритма 2.2.

3 | Технологическая часть

3.1 Выбор ЯП

3.2 Сведения о модулях программы

3.3 Вывод

4 | Экспериментальная часть

- 4.1 Сравнение времени работы муравьиного алгоритма и полного перебора
- 4.2 Результат работы программы
- 4.3 Параметризация муравьиного алгоритма
- 4.4 Вывод

Заключение

В рамках выполнения работы были решены следующие задачи:

- 1.

Литература

- [1] Visual Studio Code [Электронный ресурс], режим доступа: <https://code.visualstudio.com/> (дата обращения: 02.10.2020)
- [2] Windows [Электронный ресурс], режим доступа: <https://www.microsoft.com/ru-ru/windows> (дата обращения: 02.10.2020)
- [3] Linux [Электронный ресурс], режим доступа: <https://www.linux.org.ru/> (дата обращения: 02.10.2020)
- [4] Руководство по языку python [Электронный ресурс], - режим доступа: <https://pythonworld.ru/samouchitel-python> (дата обращения: 26.11.2020)
- [5] Ubuntu 18.04 [Электронный ресурс], режим доступа: <https://releases.ubuntu.com/18.04/> (дата обращения: 02.10.2020)