Содержание

Оглавление

[Введение 1](#_Toc82729502)

[1. История вопроса 1](#_Toc82729503)

[2. Метод ветвей и границ 3](#_Toc82729504)

[2.1. Особенности реализации на С++ 3](#_Toc82729505)

[2.2. Особенности реализации в математическом пакете (Wolfram Mathematica) 6](#_Toc82729506)

[3. Поиск с запретами 6](#_Toc82729507)

[Заключение 7](#_Toc82729508)

[Литература 7](#_Toc82729509)

# Введение

Основной целью ознакомительной практики 4-го семестра, входящей в учебный план подготовки бакалавров по направлению 01.03.04 – Прикладная математика, является знакомство с особенностями деятельности в рамках выбранного направления подготовки и получение навыков применения теоретических знаний в практической деятельности.

# История вопроса

В рамках прикладных проблем, которыми занимается «Комбинаторная оптимизация» (область теоретической математики), особой известностью пользуется «Задача коммивояжёра» (или **TSP** от англ. *Travelling salesman problem*), ее вариация «Задача о ходе коня» известна еще по работе Леонарда Эйлера 1759 г.

В общем виде «Задача коммивояжера» может быть сформулирована так:

*Найти самый выгодный маршрут, проходящий через указанные города хотя бы по одному разу с последующим возвратом в исходный город.*

Критерии «выгодности» могут определяться различно, например: минимизация длинны всего пути, времени в пути, стоимости в пути.

Для формального определения задачи используется математическое определение «модели на графе». Граф определен множеством вершин, и множеством пар вершин – рёбер, где вершины графа соответствуют городам, а рёбра между вершинами — пути сообщения между этими городами. Каждому ребру ставится в соответствие коэффициент выгодности.

Для решения этой задачи могут использоваться алгоритмы разной сложности:

Простые, но не сильно эффективные:

1. полный перебор,
2. случайный перебор,
3. метод ближайшего соседа,
4. метод включения ближайшего города,
5. метод самого дешёвого включения,
6. метод минимального остовного дерева,
7. метод имитации отжига.

Эвристические, более эффективные:

* метод ветвей и границ,
* метод генетических алгоритмов,
* алгоритм муравьиной колонии.

Рассмотрим подробно два метода:

* 1. «Метод ветвей и границ», предложенный в 1960г. Алисой Лэнд и Элисон Дойг для решения комбинаторных задач, и применённый группой авторов к решению задачи Коммивояжёра в 1963г (Дж. Литл, К. Мурти, Д. Суини, К. Кэрол).
  2. «Поиск с запретами», мета-алгоритм поиска, созданный Фредом У. Гловером в 1986 (формализованный им в 1989).

# Метод ветвей и границ

Метод ветвей и границ является эвристическим методом, который в отличии от полного перебора отсекает прохождение заведомо неоптимальных ветвей. Последовательно осуществляется деление множества допустимых решений на две части, из которых выбирается одна из «ветвей», каждая из ветвей анализируется для принятия решения о включении или исключении ее в «маршрут».

Состав шагов алгоритма решения задачи коммивояжера с использованием метода ветвей и границ следующий:

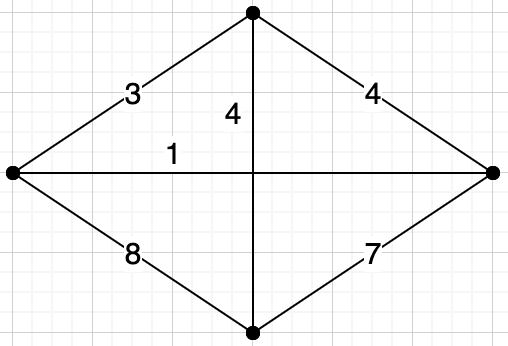
1. Конвертация графа исходной задачи в матрицу данных.
2. Обработка строк:
   1. Нахождение минимумов для каждой строки.
   2. Редукция матрицы по строкам.
3. Обработка столбцов.
   1. Нахождение минимума для каждого столбца.
   2. Редукция матрицы по столбцам.
4. Вычисление оценок нулевых клеток.
5. Редукция всей матрицы.
6. Проверка маршрута на окончание (продолжение цикла с п. 2, или переход к завершению).
7. Вычисление по последовательности шагов маршрута его итоговой длины.

## Особенности реализации на С++

Будем рассматривать в нашей задаче модельный полносвязный n-вершинный ориентированный граф, у которого между каждой (i, j) парой вершин существует 2 дуги с разной стоимостью проезда и одностороннее (Сij ≠ Cji) движение.

Решением задачи коммивояжёра является отыскание на нашем орграфе маршрута, проходящего однократно через все (n) вершин, при наименьшей его стоимости.

Проиллюстрируем решение задачи коммивояжера на простом графе с 4-мя вершинами:



Дерево потенциальных маршрутов для графа на 4-х маршрутах будет выглядеть следующим образом:

Изображение выглядит как провод, металлический

Автоматически созданное описание

На каждом шаге алгоритма будет рассматриваться один из возможных маршрутов и даваться оценка стоимости маршрута, и сравниваться с понятием *базового маршрута*, если оценка выше – то маршрут отбрасывается как не перспективный (и это можно сделать на ранней стадии рассмотрения ветви), то есть

* 1. Сформируем матрицу по графу



* 1. Для определения нижней границы множества выполним *операцию редукции* (операция приведение матрицы) по строкам.
  2. Вычислим минимум для каждой строки (в приписанном справа столбце помещаем эти значения, которые называются *константами приведения*):



* 1. Осуществим редукцию матрицы по строкам



* 1. Так же выполним операцию редукции (операция приведение матрицы) по столбцам.
  2. Вычислим минимум для каждого столбца:



* 1. Осуществим редукцию матрицы по столбцам



Суммируя все константы приведения, получим нижнюю границу:

H = 1 + 3 + 1+ 4 + 0 + 0 + 0 + 1 = 10

1. На следующем шаге проводим для каждого потенциального ребра маршрута оценку его стоимости. Для этого для всех нулевых клеток осуществляем замену на бесконечность и определяем для них константы приведения, подставляем их при выборе анализируемого ребра на это место. Это самый протяженный этап алгоритма.



Для каждого нуля выписываем сумму констант приведения каждого потенциально возможного для рассмотрения ребра. Наибольшая сумма констант приведения для ребра Р(2,4):

Р(1,3) = 2 + 1 = 3;

Р(2,1) = 0 + 0 = 0;

Р(2,4) = 0 + 5 = 5;

Р(3,1) = 3 + 0 = 3;

Р(4,2) = 3 + 2 = 5;

Далее разбиваем множество на два подмножества (2,4) и (2\*,4\*).

*Исключение ребра* Р(2,4) осуществляем заменой элемента матрицы (2,4) на ♾ и приведением полученной матрицы:



Вычисляем нижнюю границу:

H(2\*,4\*) = 10 + 5 = 15

*Включение* *ребра* Р(2,4) осуществляем удалением 2-й строки и 4-го столбца и заменой на ♾ элемента матрицы (4,2):



Вычисляем нижнюю границу:

Н(2,4) = 10 + 5 = 15 == 15

Поскольку нижние границы подмножества (2,4) и подмножества (2\*,4\*) равны, то ребро Р(2,4) включаем в маршрут с новой границей H = 15.

Для последней полученной матрицы повторяем действия текущего шага.

Когда мы дойдем до матрицы 2х2



В результате прохождения алгоритма в машрут будут включены ребра:

Р(2,4), Р(4,3), Р(3,1), Р(1,2),

Стоимость маршрута равна F(Х) = 15.

## Особенности реализации в математическом пакете (Wolfram Mathematica)

Реализация алгоритма для решения задачи коммивояжера с использованием метода ветвей и границ в специализированном математическом пакете представляет меньшую сложность по сравнению с программированием на С++ и позволяет отработать детали алгоритма, что удобно для последующей отладки кода С++.

Однако реализация алгоритма в математическом пакете, в сравнении с его реализацией в коде на С++, уступает в производительности, что радикально скажется при обработке больших графов (матриц).

# Поиск с запретами

## Особенности реализации на С++

# Заключение

По ходу решения поставленных задач, по автоматизации решения задачи коммивояжера, были проанализированы метод ветвей и границ и алгоритм поиска с запретами (tabu-search). Задача коммивояжера помимо теоретического значения имеет практическое значения для транспортной логистики, где при перемещении грузов могут быть использованы не только разные маршруты, но и разные виды транспорта со своими ограничениями по объему груза, стоимости и скорости доставки.

Тем не менее практическое использование реализованных алгоритмов в реальных задачах, вероятно потребует оптимизации из-за сложности оценки критериев «выгодности» маршрута.

# Литература

1. Задача Коммивояжера // Сайт свободная энциклопедия (Википедия),   
   URL: <https://ru.wikipedia.org/wiki/Задача_коммивояжёра#cite_note-3> // (дата обращения: 12.07.2021).
2. Галяутдинов Р.Р. Задача коммивояжера - метод ветвей и границ // Сайт преподавателя экономики. URL: <http://galyautdinov.ru/post/zadacha-kommivoyazhera> (дата обращения: 1.09.2021).
3. Метод ветвей и границ. Задача коммивояжера // Сайт ООО «Хабр», URL: <https://habr.com/ru/post/560468/> // (дата обращения: 16.08.2021).