2а задача: 2-приближение задачи коммивояжёра

3 модуль, 2 семестр

ФИВТ МФТИ, 2019

Описание by Илья Белов

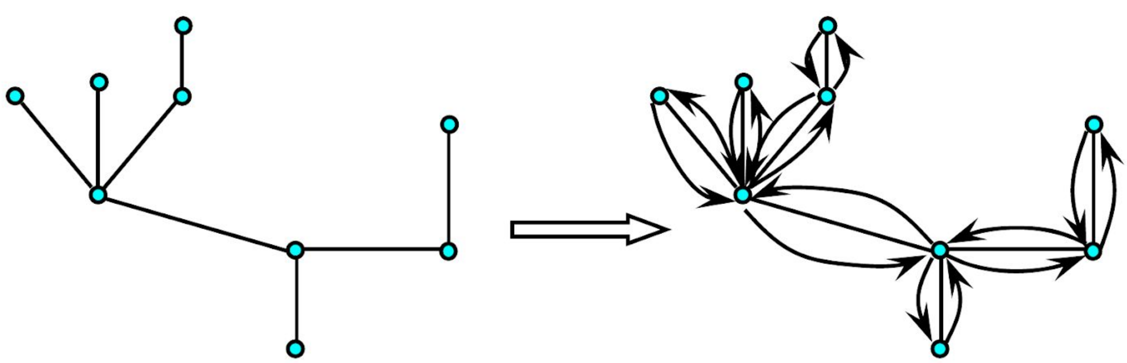
1. Текст задачи

Найдите приближенное решение метрической неориентированной задачи коммивояжера в полном графе (на плоскости) с помощью минимального остовного дерева, построенного в первой задаче. Приближение должно быть не хуже, чем в 2 раза от идеального
Оцените качество приближения на случайном наборе точек, нормально распределенном на плоскости с дисперсией 1. Нормально распределенный набор точек получайте с помощью std::normal_distribution.
При фиксированном N, количестве вершин графа, несколько раз запустите оценку качества приближения. Вычислите среднее значение и среднеквадратичное отклонение качества приближения для данного N.
Запустите данный эксперимент для всех N в некотором диапазоне, например, [2, 10].
Автоматизируйте запуск экспериментов

2. Описание алгоритма[1]

Будем использовать следующие обозначения:
I - пример графа
$Opt(I)$ - истинное значение оптимального пути коммивояжёра
$A_k(I)$ - вес MST
$A_{ST}(I)$ - длина пути, полученная в описываемом алгоритме

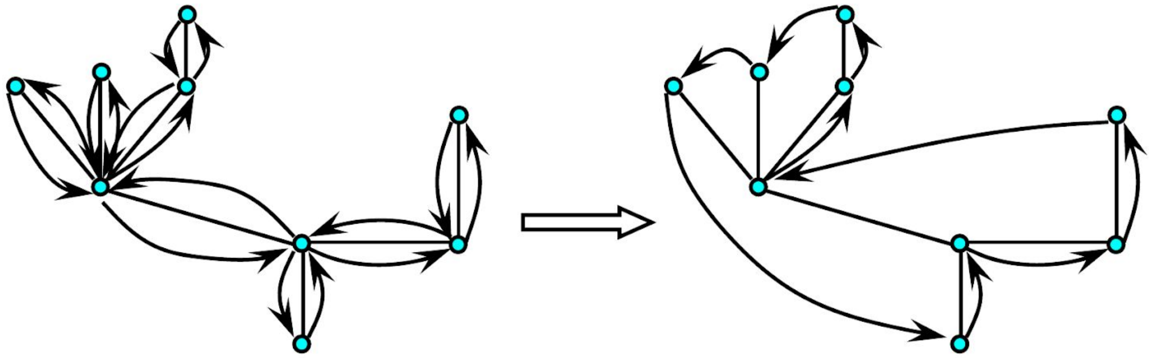
Для полного взвешенного графа задачи коммивояжера построим алгоритмом Краскала A_k остовное дерево минимального веса. Заметим, что $A_k(I) < Opt(I)$ для любого примера, так как удаление ребра из оптимального решения задачи коммивояжера дает остовное дерево. Заменим каждое ребро на два ребра. Получим эйлеров граф каждая вершина имеет четную степень. Построим произвольный эйлеров цикл



Листья посещаются 1 раз, остальные вершины по нескольку раз. Перестроим его так, чтобы получился гамильтонов цикл. Начиная с произвольной вершины, двигаемся вдоль эйлерова цикла и помечаем вершины. Если очередная вершина уже помечена, то пропускаем ее и

[1] www.math.nsc.ru/LBRT/k5/OR-MMF/TSPr.pdf

двигаемся дальше, пока не найдем непомеченную вершину или не вернемся в первую вершину. Цепочку дуг для помеченных вершин заменяем прямой дугой в непомеченную или первую вершин



3. Доказательство корректности[2]

Для любого примера задачи коммивояжера с симметричной матрицей, удовлетворяющей неравенству треугольника, алгоритм A_{ST} получает гамильтонов цикл не более чем в два раза длиннее оптимального, то есть $A_{ST}(I) \le 2Opt(I)$

Доказательство. Для двойного обхода остовного дерева имеем $2A_k(I) < 2Opt(I)$

Пусть новое ребро (i, j) не содержащееся в двойном обходе, заменяет цепочку ребер $(i, k_1), (k_1, k_2), ..., (k_m, j)$. Из неравенства треугольника следует, что $c_{ij} \le c_{ik_1} + ... + c_{k_mj}$.

Следовательно, $A_{ST}(I) \le 2A_k(I) \le 2Opt(I)$

4. Время работы и дополнительная память

$T = O(V^2 logV)$

$M = O(V + E)$

5. Доказательство времени работы и дополнительной памяти

Докажем время работы. Поиск MST занимает $T = O(ElogV)$. Наш граф полный, то есть $E = V(V-1)/2 = O(V^2)$, тогда поиск MST занимает $T = O(V^2 logV)$. После этого с помощью DFS мы строим Гамильтонов цикл в графе. DFS занимает $T = O(V + E) = O(V)$

Итоговое время $T = O(V^2 logV + V) = O(V^2 logV)$

Докажем доп память. $O(V + E)$ необходимо для хранения DSU в алгоритме поиска MST

[2] Аналогично