

## 1.1 Постановка задачи

Задача коммивояжера является одной из самых известных задач комбинаторной оптимизации. Суть задачи в том, что имеется  $n$  городов и стоит задача посетить все города и вернуться в начальный. Как правило также стоит условие не посещать дважды один и тот же город, однако есть постановки задачи которые допускают это. Переход между каждыми двумя городами стоит некоторое количество ресурсов: денег, времени, запаса провизии и т.д. Нужно найти оптимальный способ посетить все города. В некоторых постановках, на максимум, наоборот ищется маршрут, который будет стоить максимальное количество ресурсов.

## 1.2 Приложения задачи

У задачи коммивояжера есть множество практических применений.

Самое очевидное - это сама постановка задачи. Посетить некоторое количество городов затратив наименьшее количество ресурсов. Таким образом, например, рассмотрены задачи посещения всех городов с учетом расхода ресурсов линейно зависящего от расстояния между данными городами для одной отдельно взятой страны для некоторых европейских стран. Задача посещения всех европейских стран с учетом стоимости перелета между ними. Однако есть и другие применения задачи, более неочевидные.

Допустим, у нас есть некое устройство, которое одновременно может выполнять только одну задачу. Нам нужно последовательно выполнить на данном устройстве  $n$  задач, причем после каждой задачи нам нужно некое время на подготовку к следующей, причем между похожими задачами время подготовки может быть сокращено, а между разными - наоборот, очень большим. Требуется найти такой порядок выполнения, чтоб было затрачено минимальное количество времени на промежуточные подготовки.

А вот еще одно применение задачи коммивояжера. В сфере радиокommunikаций есть такая задача как назначение частот передатчикам. У нас есть  $n$  передатчиков и некоторый набор доступных частот с некими ограничениями. Эти ограничения могут быть представлены как граф  $G = (V, E)$  где каждая вершина  $i$  это передатчик.  $c_{ij}$  - вес ребра  $(i, j)$  это допустимое отклонение частоты. Пусть  $F = \{0, 1, 2, \dots, R\}$  это набор доступных частот. Задача назначить число  $f(i) \in F$  вершине  $i \in V$  так, чтобы  $|f(i) - f(j)| > c_{ij}$  для всех  $(i, j) \in E$ . Если такое назначение существует, оно называется допустимым. Если  $R$  относительно большое, то допустимое назначение существует всегда. Минимальное число  $R$  для которого существует допустимое назначение для  $G$  называется диапазон  $G$  и обозначается  $Span(G)$ .

Пусть  $G^*$  - полный граф, который получается из  $G$  путем добавления ребер нулевого веса. Пусть  $c'_{ij}$  будет весом ребра  $(i, j)$  в  $G^*$  таким, что  $c'_{ij} = c_{ij} + 1$ . Пусть  $C'(H^*)$  это сумма весов ребер в гамильтоновом цикле наименьшего веса в графе  $G^*$ . Д. Смит и С. Хёрли, доказали, что  $Span(G) \geq C'(H^*)$ . Поэтому

задача коммивояжера имеет здесь применение для расчета нижней границы для данной задачи.

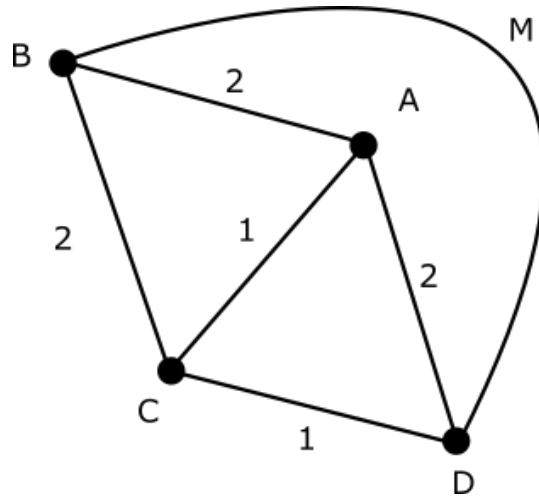
### 1.3 Математическая модель для задачи коммивояжера.

Изначальную задачу можно представить как граф. Вершинами графа будут города, которые нам необходимо посетить. Ребрами же будут пути, пролегающие из города в город. Весом ребра будет стоимость перехода из города в город по этому ребру. В дальнейшем, если заранее не оговорено, мы будем рассматривать такую постановку, в которой нам дан полный взвешенный граф. То есть из любого города в любой есть некий путь.

Цель задачи коммивояжера - найти такой цикл в графе, в котором будут содержаться все вершины, причем вес цикла, который определяется как сумма весов ребер между вершинами, входящими в цикл, должен быть минимальным. Такой цикл, в котором все вершины содержатся ровно один раз называется Гамильтоновым циклом.

Данная задача относится к классу NP-трудных задач. То есть, на данный момент не найден алгоритм, который бы решал задачу за полиномиальное время и не найдено алгоритма, который бы проверял некоторое решение задачи на оптимальность за полиномиальное время. По этой причине для задачи актуально исследование приближенных алгоритмов с целью получения приемлемого решения за более короткий промежуток времени.

Одним из таких алгоритмов, находящих некое решение за приемлемое количество операций, а именно  $O(n^2)$ , где  $n$  - количество вершин, является жадный алгоритм с выбором ближайшей вершины. Суть алгоритма заключается в том, что на каждом шаге мы просматриваем все непосещенные вершины в которые можно попасть из текущей вершины, и выбираем ту, расстояние до которой минимально. Оценку времени работы алгоритма получить нетрудно: на каждом шаге мы просматриваем не более  $n$  вершин и выбираем одну из них. Всего у нас  $n$  вершин, значит будет  $n$  шагов. Получаем оценку  $O(n^2)$ . В общем случае данный алгоритм может работать сколь угодно плохо относительно оптимального решения. На изображении ниже представлен пример на котором жадный алгоритм ИБГ будет работать сколь угодно плохо в зависимости от веса ребра.



Рассмотрим данный граф. При начале работы ИБГ из вершины  $A$  у нас получится путь  $ACDBA$  который зависит от веса  $M$  ребра  $DB$ , которое может быть сколь угодно тяжелым. Оптимальным же маршрутом для данного графа является цикл  $ADCBA$

Однако, если мы наложим дополнительные условия на веса ребер или сам граф мы можем получить более точный результат. Например, если граф метрический, у нас появляются верхняя и нижняя оценки оптимальности работы жадного алгоритма.

Для маршрута коммивояжера с  $n$  вершинами справедлива теорема:

**Теорема 1. Верхняя оценка оптимальности:**

$$\left(\frac{Greedy}{Optimal}\right) \leq \frac{1}{2} \lceil \lg n \rceil + \frac{1}{2} \quad (1)$$

Где  $Greedy$  - вес пути, найденного жадным алгоритмом,  $Optimal$  - вес наиболее оптимального пути.

Доказательство:

**Лемма 1:** Допустим существует отображение вершины  $p$  в число  $l_p$  такое, что:

- 1)  $d(p, q) \geq \min(l_p, l_q) \forall p, q$
- 2)  $l_p \leq \frac{1}{2} Optimal \forall p$

Тогда

$$\sum l_p \leq \frac{1}{2} (\lceil \lg(n) \rceil + 1) Optimal \quad (2)$$

Доказательство. Допустим б.о.о., что  $N$  такое, что  $\{i | 1 \leq i \leq n\}$  и  $l_i \geq l_j$  если  $i \leq j$  Докажем, что

$$Optimal \geq 2 \sum_{i=k+1}^{\min(2k, n)} l_i \quad (3)$$

для любого  $k$  такого что  $l \leq k \leq n$  Пусть  $H$  полный подграф, определенный на множестве узлов

$$\{i | 1 \leq i \leq \min(2k, n)\} \quad (4)$$

Пусть  $T$  - это маршрут коммивояжера в  $H$ , который посещает вершины  $H$  в том же порядке, что и эти узлы посещаются при оптимальном обходе исходного графа. Обозначим длину маршрута  $T$  как  $LENGTH$ . По неравенству треугольника каждое ребро  $(b, c)$  графа  $T$  должно иметь длину меньше или равную длине пути от  $b$  до  $c$ , вычисленного в оптимальном пути коммивояжера. Так как ребра  $T$  суммируются в  $LENGTH$ , а сумма ребер оптимального пути равна  $Optimal$  мы заключаем что

$$Optimal \geq LENGTH \quad (5)$$

По условию 1) Леммы для каждого  $(i, j)$  в пути  $T$   $d(i, j) \geq \min(l_i, l_j)$ . Следовательно,

$$LENGTH \geq \sum_{(i,j) \in T} \min(l_i, l_j) = \sum_{i \in H} a_i l_i \quad (6)$$

где  $a_i$  - количество ребер  $(i, j)$  в  $T$ , для которых  $i > j$  (и, следовательно,  $l_i = \min(l_i, l_j)$ ). Мы хотим получить нижнюю оценку правой части 6. Заметьте, что каждое  $a_i$  не превосходит 2 (потому что  $i$  это конечная точка только двух ребер в маршруте  $T$ ) и что сумма  $a_i$  равна количеству ребер в  $T$ . Поскольку  $k$  составляет не менее половины количества ребер в  $T$ , мы заведомо получим нижнюю оценку правой части 6, если мы предположим, что  $k$  наибольших  $l_i$  имеют  $a_i = 0$ , а остальные  $\min(2k, n) - k$  из  $l_i$  имеют  $a_i = 2$ . По предположению,  $k$  наибольших  $\{l_i | 1 \leq i \leq k\}$ , поэтому оценка нижней границы:

$$\sum_{i \in H} a_i l_i \geq 2 \sum_{i=k+1}^{\min(2k, n)} l_i \quad (7)$$

Таким образом мы доказали неравенство 3

Теперь просуммируем неравенства 3 для всех  $k$  равных степени двойки меньшей  $n$ , то есть:

$$\sum_{j=0}^{\lceil \lg(n) \rceil - 1} Optimal \geq \sum_{j=0}^{\lceil \lg(n) \rceil - 1} 2 * \sum_{i=2^j+1}^{\min(2^{j+1}, n)} l_i \quad (8)$$

Это можно упростить до

$$\lceil \lg(n) \rceil Optimal \geq 2 * \sum_{i=2}^n l_i \quad (9)$$

По условию 2) Леммы

$$Optimal \geq 2 * l_1 \quad (10)$$

Два данных неравенства доказывают лемму

Доказательство Теоремы 1. Для каждой вершины  $p$  положим, что  $l_p$  это длина ребра, выходящего из  $p$  и идущего в вершину, которая выбирается жадным алгоритмом. Мы хотим показать, что  $l_p$  удовлетворяет условиям Леммы 1. Если вершина  $p$  была выбрана жадным алгоритмом до вершины  $q$ , тогда вершина  $q$  была кандидатом на ближайшую невыбранную вершину для вершины  $p$ . Это значит, что ребро  $(p, q)$  не короче чем выбранное ребро, то есть

$$d(p, q) \geq l_p \quad (11)$$

И наоборот, если вершина  $q$  была выбрана до  $p$ , тогда

$$d(p, q) \geq l_q \quad (12)$$

Так как одна из вершин была выбрана раньше другой, одно из двух последних неравенств должно выполняться, вследствие чего условие 1) Леммы 1 выполняется. Для доказательства условия 2) достаточно доказать, что для любого ребра  $(p, q)$

$$d(p, q) \leq \frac{1}{2} Optimal \quad (13)$$

Мы можем рассмотреть оптимальный маршрут как объединение двух частей маршрута, каждый из которых это путь между  $p$  и  $q$ . Из неравенства треугольника получаем, что длина любого пути между  $p$  и  $q$  не может быть меньше, чем  $d(p, q)$ , что доказывает неравенство выше. Так как  $l_p$  это длины всех пар, составляющих маршрут  $T$

$$\sum l_p = GREEDY \quad (14)$$

Данное равенство вместе с Леммой 1 доказывают неравенство из Теоремы 1.

**Теорема 2. Нижняя оценка оптимальности:**

$$\left( \frac{Greedy}{Optimal} \right) > \frac{1}{3} \lg(n+1) + \frac{4}{9} \quad (15)$$

Доказательство: Для каждого  $i \geq 1$  построим неполный взвешенный граф  $F$  с тремя особыми вершинами: левая вершина, центральная вершина и правая вершина. Эти графы строятся рекурсивно как показано на рис.1 (TODO нарисовать и вставить) где левая вершина располагается слева, правая - справа и центральная посередине. Каждый граф  $F$  имеет путь  $P$ , соединяющий левую вершину и центральную, в который входят все вершины графа. Путь  $P$  также строится рекурсивно как на рис.1.

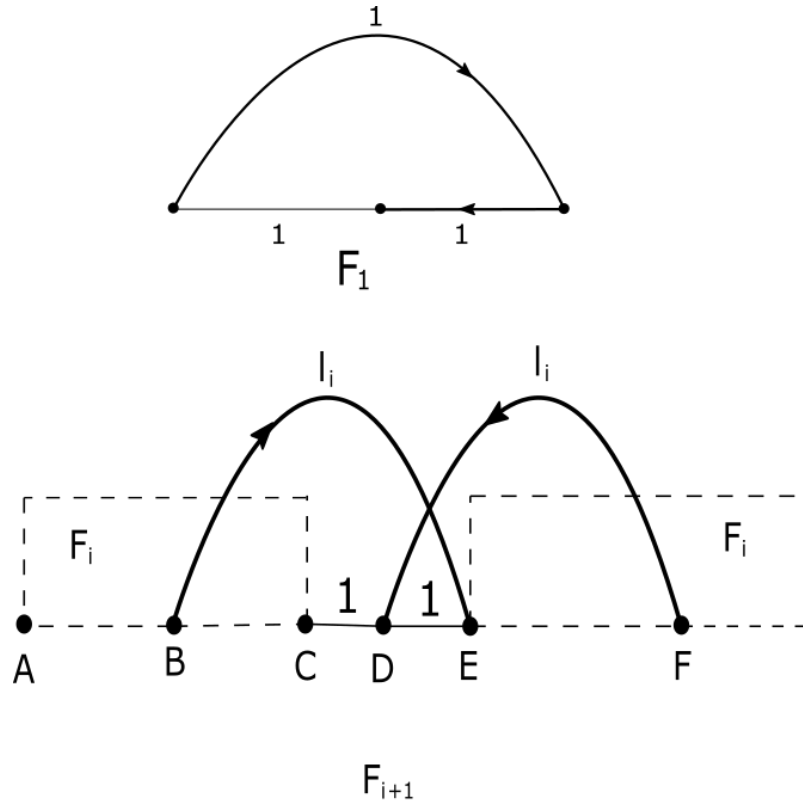


Рис. 1

Граф  $F_1$  состоит из трех вершин, между каждым двумя из которых проведено ребро веса 1. Путь  $P$  состоит из двух ребер: между левой и правой вершиной, между правой и центральной вершиной. Для построения графа  $F_{i+1}$  возьмем две копии графа  $F_i$ , назовем одну копию левой, а другую правой. Добавим дополнительную вершину которая впоследствии станет центральной для  $F_{i+1}$ . На рис.1 эта вершина обозначена  $D$ . Она соединяется с правой вершиной левой копии (вершина  $C$ ) и левой вершиной правой копии (вершина  $E$ ) ребрами длины 1. Дополнительная вершина  $D$  также соединяется с центральной вершиной правой копии (вершина  $F$ ) ребром веса  $l_i$ , который будет определен ниже. Наконец, центральная вершина левой копии (вершина  $B$ ) соединяется с левой вершиной правой копии (вершина  $E$ ) ребром веса  $l_i$ . Левая вершина  $F_{i+1}$  определяется как левая вершина левой копии (вершина  $A$ ), правая вершина  $F_{i+1}$  определяется как правая вершина правой копии (вершина  $G$ ). Путь  $P_{i+1}$  состоит из двух копий пути  $P_i$  и ребер  $(B, E)$ ,  $(F, D)$  длины  $l_i$ . Длина  $l_i$  данных ребер определяется по формуле

$$l_i = \frac{1}{6}(4 * 2^i - (-1)^i + 3) \quad (16)$$

Пусть  $L_i$  - длина пути  $P_i$ . Для длины  $L_i$  есть рекуррентное соотношение:

$$L_{i+1} = 2 * L_i + 2 * l_i \quad (17)$$

Так как  $P_{i+1}$  состоит из двух копий  $P_i$  и двух ребер веса  $l_i$  При условии что  $L_1 = 2$ , решение данного уравнения:

$$L_i = \frac{1}{9}(6i * 2^i + 8 * 2^i + (-1)^i - 9) \quad (18)$$

Для каждого  $F$  мы определяем граф  $G_i$ , который получается из  $F$  путем соединения левой вершины и правой вершины графа ребром веса 1 и соединением центральной вершины с левой вершиной ребром веса  $l_i - 1$ . Левая вершина графа  $F$  считается начальной вершиной графа  $G_i$ . На рисунке изображен граф  $G_4$ . Определим  $\bar{G}_i$  как полный граф на вершинах  $G_i$ . Длина ребер в данном графе будет равна длине наименьшего пути в  $G_i$  между двумя вершинами, которые соединяет ребро. Таким образом  $\bar{G}_i$  будет удовлетворять неравенству треугольника.

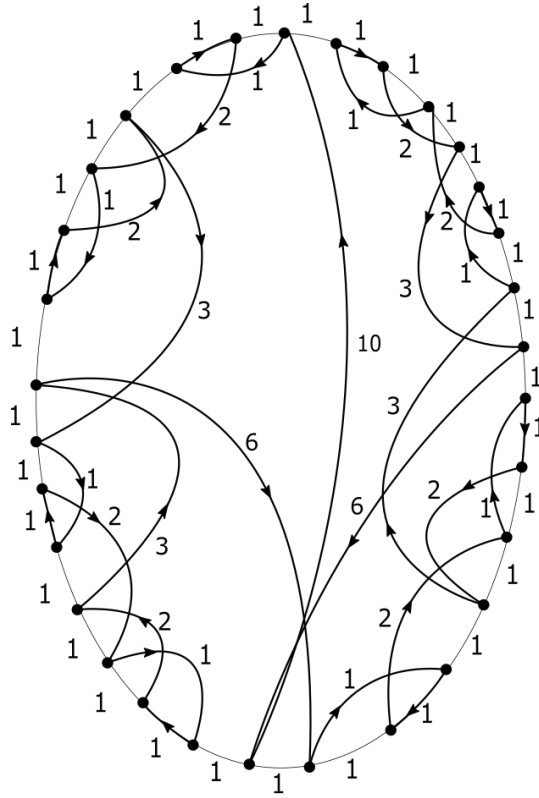


Рис. 2

Граф  $\bar{G}_i$  имеет два важных свойства:

- 1) Ребра графа  $G_i$  имеют в графе  $\bar{G}_i$  такие же длины как и в  $G_i$
- 2) Если жадный алгоритм начинает свою работу с начальной вершины графа  $G_i$  то при подходящем выборе между несколькими возможными путями в алгоритме метод может найти путь  $P_i$ , который будет следовать за ребром длины  $l_i - 1$ , проведенным из центральной вершины (последняя в пути  $P_i$ ) в начальную вершину.

Мы вернемся к доказательству этих свойств после завершения доказательства основной теоремы. Каждый граф  $\bar{G}_i$  имеет оптимальный маршрут, состоящий из ребер единичного веса, с, соответственно весом пути равным  $n$ , где  $n$  - это количество вершин ( $2^{i+1} - 1$ ). Данный путь начинается в начальной вершине и далее посещаются все вершины слева направо, после чего происходит возврат в начальную вершину. Примером, удовлетворяющим теореме является граф  $\bar{G}_{m-1}$ . Соотношение для него:

$$\frac{GREEDY}{Optimal} = (L_i + l_i - 1)/n, i = \lg(n + 1) - 1 \quad (19)$$

Данное соотношение больше чем соотношение в теореме. Остается доказать свойства 1) и 2). Рассмотрим рис.1 Покажем что для каждого  $F_{i+1}$

$$\overline{AB} = \overline{BC} = \overline{EF} = \overline{FG} = l_i - 1 \quad (20)$$

$$\overline{AC} = \overline{EG} = l_{i+1} - 2 \quad (21)$$

$$\overline{BE} = \overline{DF} = l_i \quad (22)$$

$$\overline{AD} = \overline{DG} = l_{i+1} - 1 \quad (23)$$

$$\overline{AG} = l_{i+2} - 2 \quad (24)$$

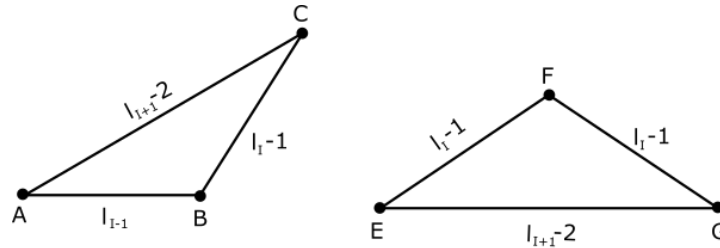


Рис. 3а

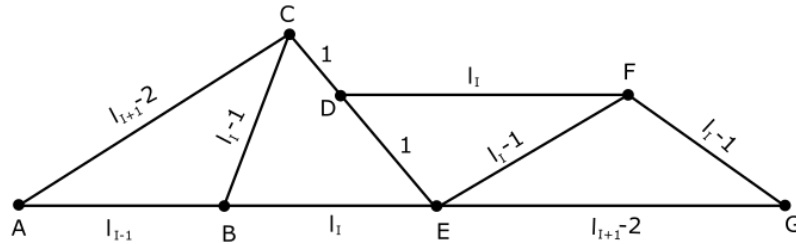


Рис. 3б

Запись  $\overline{XY}$  означает длину кратчайшего пути между X и Y в графе  $F_{i+1}$ . Данные равенства тривиально доказываются для  $i=1$ . Далее будем вести доказательство по индукции. Предположим что равенства верны для  $i \leq I - 1$ , например для  $F_I$ . На рис.3а изображены связанные вершины графа  $F_{I+1}$  до объединения двух копий  $F_I$  в  $F_{I+1}$ . Веса ребер графа равны кратчайшему пути между ними в  $F_I$ . Эти веса определены в предположении индукции. Например, ребро (A,B) на рис.3а соединяет левую и центральную вершины  $F_I$  и, как следует из 23, длина кратчайшего пути между этими вершинами равна  $l_i - 1$ . На рис.3б мы можем увидеть фигуры с рис.3а с ребрами, которые были добавлены при построении  $F_{I+1}$ . Так как каждый вес ребра между двумя вершинами на рис.3а равен кратчайшему пути между этими вершинами, то применяя формулу 16 для  $l_I$  ко всем возможным путям в  $F_{i+1}$  мы можем найти, что вес каждого ребра на рис.3б действительно является длиной кратчайшего пути между двумя вершинами, которые соединяет ребро. Это доказывает равенства 20-22 для  $F_{I+1}$ . Равенства 23, 24 доказываются аналогичными рассуждениями для всех путей на рис.3б. Путем длины  $l_{i+1} - 2$  из A в G является путь  $ABEG$ .



Также стоит отметить, что 20 - 23 выполняются и когда из  $F_{l+1}$  строится  $G_{l+1}$ . Это можно доказать соединив  $A$  и  $G$  на рис. 3б ребром веса  $l_{l+1} - 1$  и снова проверив пути. Кратчайшим путем из  $A$  в  $D$  является ребро  $(A, D)$ .

Теперь вернемся к свойству а). Уравнение (22) показывает, что для каждого построенного  $F_{i+1}$  ребра, добавленные при построении содержатся в кратчайшем пути. Все расстояния между точками в  $F_i$  уже вычислены на момент построения  $F_{i+1}$  потому, что расстояния между особыми точками уже вычислены. (сравним (20) и (23), (21) и (24))

Мы уже заметили, что последнее добавленное ребро при построении  $G_i$  является кратчайшим путем и, очевидно, путь состоящий из ребер длины 1 является кратчайшим. Таким образом, свойство 1) доказано.

Свойство 2) доказывается тем фактом, что центральная вершина графа  $F_i$  достигается только после посещения всех остальных вершин, и вершина в конце ребра длины  $l_i$  хотя бы настолько же близка как любая другая вершина, достигаемая через левую или правую вершины. Эти вершины уже на расстоянии  $l_{i-1}$  от центральной вершины и как минимум на расстоянии 1 от еще не выбранной вершины.

Объектом исследований данной работы являются оценки оптимальности решения задачи, найденного жадным алгоритмом при случае когда веса ребер являются случайными величинами и распределены по некоторому закону распределения.

## 2. Некоторые определения и вспомогательные теоремы

### 2.1 Определения

Алгоритм  $\mathcal{A}$  будем называть асимптотически оптимальным если существуют такие  $\epsilon_n \rightarrow 0, \delta_n \rightarrow 0$  при  $n \rightarrow \infty$  что применение алгоритма  $\mathcal{A}$  дает значение  $\mathcal{Z}_{\mathcal{A}}$ , удовлетворяющее неравенству

$$P\{\mathcal{Z}_{\mathcal{A}} \leq (1 + \epsilon_n)\mathcal{Z}\} \geq 1 - \delta_n \quad (25)$$

где  $\mathcal{Z}_{\mathcal{A}}$  - решение, найденное алгоритмом,  $\mathcal{Z}$  - минимальное решение. Это определение при  $\epsilon_n \equiv 0$  совпадает с понятием алгоритма, который почти всегда приводит к точному решению.

### 2.2 Неравенство Чебышева

Пусть случайная величина  $X: \Omega \rightarrow \mathbb{R}$  определена на вероятностном пространстве  $(\Omega, \mathcal{F}, \mathbb{P})$ , а её математическое ожидание  $\mu$  и дисперсия  $\sigma^2$  конечны. Тогда

$$\mathbb{P}(|X - \mu| \geq a) \leq \frac{\sigma^2}{a^2} \quad (26)$$

, где  $a > 0$ .

## 2.3 Теорема Петрова

Пусть  $X_1, \dots, X_n$  — независимые случайные величины и существуют положительные постоянные  $g_1, \dots, g_n$  и  $T$  такие, что для всех  $0 \leq t \leq T$

$$\mathbb{E} e^{tX_k} \leq e^{\frac{1}{2}g_k t^2} \quad (27)$$

Положим  $S = \sum_{k=1}^n X_k$  и  $G = \sum_{k=1}^n g_k$ . Тогда

$$P\{S > x\} \leq \begin{cases} \exp(-\frac{x^2}{2G}), & 0 \leq x \leq GT \\ \exp(-\frac{Tx}{2}), & x \geq GT \end{cases} \quad (28)$$

## 3. Условия асимптотической оптимальности жадного алгоритма

Найдем условия, при выполнении которых жадный алгоритм является асимптотически оптимальным, то есть

$$P\{\mathcal{Z}_{\mathcal{A}'} > (1 + \epsilon_n)\mathcal{Z}\} \leq \delta_n \quad (29)$$

где  $\epsilon_n \rightarrow 0, \delta_n \rightarrow 0, n \rightarrow \infty$

Вес маршрута коммивояжера, полученного применением жадного алгоритма обозначим как  $\mathcal{Z}_{\mathcal{A}'}$ . Он определяется формулой

$$\mathcal{Z}_{\mathcal{A}'} = \sum_{k=1}^n a_{i_k i_{k+1}} \quad (30)$$

Вес маршрута оптимального решения задачи обозначим как  $\mathcal{Z}$ . Он определяется формулой:

$$\mathcal{Z} = \min_{\{\pi\}} \sum_{k=1}^n a_{i_k i_{k+1}} \quad (31)$$

Оценим сверху левую часть (29). Так как  $a > 0$ , то  $\mathcal{Z} \geq na$  и

$$P\{\mathcal{Z}_{\mathcal{A}'} > (1 + \epsilon_n)\mathcal{Z}\} \leq P\{\mathcal{Z}_{\mathcal{A}'} > (1 + \epsilon_n)na\} \quad (32)$$

Обозначим через  $\mathcal{Z}_{\mathcal{A}'}^*$  и  $\mathcal{D}_{\mathcal{A}'}^*$  верхние оценки соответственно математического ожидания  $E(\mathcal{Z}_{\mathcal{A}'})$  и дисперсии  $D(\mathcal{Z}_{\mathcal{A}'})$  случайной величины  $\mathcal{Z}_{\mathcal{A}'}$ . Обозначим  $\Delta_n = (1 + \epsilon_n)na - \mathcal{Z}_{\mathcal{A}'}^*$ . Получаем:

$$P\{\mathcal{Z}_{\mathcal{A}'} > (1 + \epsilon_n)na\} = P\{\mathcal{Z}_{\mathcal{A}'} > \mathcal{Z}_{\mathcal{A}'}^* + [(1 + \epsilon_n)na - \mathcal{Z}_{\mathcal{A}'}^*]\} \leq P\{\mathcal{Z}_{\mathcal{A}'} > E(\mathcal{Z}_{\mathcal{A}'})) + \Delta_n\} \quad (33)$$

Пусть  $\epsilon_n = K(\frac{\mathcal{Z}_{\mathcal{A}'}^*}{na} - 1)$ ,  $K > 1$ . Тогда  $\Delta_n = (K - 1)(\mathcal{Z}_{\mathcal{A}'}^* - na) \geq 0$ . Продолжим неравенство 33 применив неравенство Чебышева.

$$P\{\mathcal{Z}_{\mathcal{A}'} > E(\mathcal{Z}_{\mathcal{A}'}) + \Delta_n\} \leq P\{|\mathcal{Z}_{\mathcal{A}'} - E(\mathcal{Z}_{\mathcal{A}'})| \geq \Delta_n\} \leq \frac{D(\mathcal{Z}_{\mathcal{A}'})}{\Delta_n^2} \leq \frac{\mathcal{D}_{\mathcal{A}'}^*}{\Delta_n^2} = \frac{\mathcal{D}_{\mathcal{A}'}^*}{(K - 1)^2(\mathcal{Z}_{\mathcal{A}'}^* - na)^2} \quad (34)$$

Так как  $K$  - константа,  $K > 1$ , то из данной цепочки неравенств следует, что условие ассимптотической оптимальности жадного алгоритма будет выполнено если мы покажем, что  $\epsilon_n = K(\frac{\mathcal{Z}_{\mathcal{A}'}^*}{na} - 1) \rightarrow 0$  и  $\delta_n = \frac{\mathcal{D}_{\mathcal{A}'}^*}{(K-1)^2(\mathcal{Z}_{\mathcal{A}'}^* - na)^2} \rightarrow 0$  при  $n \rightarrow \infty$

### Вычисление верхних оценок $\mathcal{Z}_{\mathcal{A}'}^*$ и $\mathcal{D}_{\mathcal{A}'}^*$

Математическое ожидание  $E(\mathcal{Z}_{\mathcal{A}'})$  равно сумме матожиданий величин  $a_{i_k i_{k+1}}$  минимальных длин ребер, ведущих в еще непосещенную вершину на  $k$ -м шаге алгоритма  $\mathcal{A}'$ . В целях удобства дальнейших вычислений нормируем случайную величину  $\xi$  значений элементов  $a_{ij}$  длин ребер графа, положив  $\xi' = \frac{\xi - a}{b - a}$ . Обозначим через  $l_k$  значение матожидания нормированной случайной величины  $l_k = a'_{i_k i_{k+1}}$ . На  $k$ -ом шаге алгоритма выбирается минимум из  $n - k$  элементов. В силу независимости этих элементов вероятность  $\Phi_k(x)$  того, что величина  $l_k$  минимального из этих элементов не превышает величины  $x$  равна

$$\Phi_k(x) = P\{l_k \leq x\} = 1 - (1 - F(x))^{n-k} \quad (35)$$

, где

$$F_k(x) = P\{\xi' \leq x\}, 0 \leq x \leq 1 \quad (36)$$

Тогда величина  $E(l_k)$  равна

$$E(l_k) = \begin{cases} \int_0^1 x d\Phi_k(x), & k = 1, 2, \dots, n-1 \\ E(l_{n-1}), & k = n \end{cases} \quad (37)$$

откуда получим

$$\begin{aligned} E(l_k) &= x\Phi_k(x)|_0^1 - \int_0^1 \Phi_k(x) dx = \\ &= 1 - \int_0^1 [1 - (1 - F(x))^{n-k}] dx = \int_0^1 (1 - F(x))^{n-k} dx \\ &\quad k = \overline{1, n-1} \end{aligned} \quad (38)$$

В силу нормировки минимальный элемент  $a_{i_{k-1} i_k}$  связан с величиной  $l_k$  соотношением  $a_{i_k i_{k+1}} = a + (b - a)l_k$ . Поэтому

$$E(\mathcal{Z}_{\mathcal{A}'}) = \sum_{k=1}^n [a + (b - a)E(l_k)] \quad (39)$$

. Откуда с учетом 38 имеем:

$$\begin{aligned}
E(\mathcal{Z}_{\mathcal{A}'}) &= na + (b - a) \left[ \int_0^1 \sum_{k=1}^{n-1} (1 - F(x))^{n-k} dx + \int_0^1 (1 - F(x)) dx \right] = \\
&= na + (b - a) \int_0^1 \frac{[1 - F(x)][1 - (1 - F(x))^{n-1} + F(x)]}{F(x)} dx \leq \\
&\leq na + (b - a) \int_0^1 \frac{1 - (1 - F(x))^n}{F(x)} dx = \\
&= na + (b - a) \left[ \int_0^{\gamma_n} \frac{1 - (1 - F(x))^n}{F(x)} dx + \int_{\gamma_n}^1 \frac{1 - (1 - F(x))^n}{F(x)} dx \right]
\end{aligned} \tag{40}$$

где  $\gamma_n$  - корень уравнения  $F(\gamma) = \frac{1}{n}$ , то есть  $\gamma_n = F^{-1}(\frac{1}{n})$ . Учитывая что при  $0 \leq Z \leq 1$  справедливо неравенство  $\frac{1 - (1 - Z)^n}{Z} \leq n$ , оценку  $E(\mathcal{Z}_{\mathcal{A}'})$  можем продолжить следующим образом

$$E(\mathcal{Z}_{\mathcal{A}'}) \leq \mathcal{Z}_{\mathcal{A}'}^* = na + (b - a) \left[ \gamma_n * n + \int_{\gamma_n}^1 \frac{dx}{F(x)} \right] \tag{41}$$

Перейдем к вычислению верхней оценки  $\mathcal{D}_{\mathcal{A}'}^*$ . Дисперсия  $d_k$  случайной нормированной величины  $l_k$  на  $k$ -ом шаге равна

$$d_k = \int_0^1 (x - E(l_k))^2 d\Phi_k(x) = \int_0^1 x^2 d\Phi_k(x) - (E(l_k))^2 < \int_0^1 x d\Phi_k(x) = E(l_k) \tag{42}$$

Тогда с учетом того, что дисперсия минимального элемента  $a_{i_k i_{k+1}}$  равна  $(b - a)^2 d_k$ , дисперсия случайной величины  $\mathcal{Z}_{\mathcal{A}'}$  с учетом 39 оценивается следующим образом:

$$\begin{aligned}
\mathcal{D}(\mathcal{Z}_{\mathcal{A}'}) &= \sum_{k=1}^n (b - a)^2 d_k < (b - a)^2 \sum_{k=1}^n E(l_k) = \\
&= (b - a)^2 * \frac{E(\mathcal{Z}_{\mathcal{A}'}) - na}{(b - a)} \leq (b - a)(\mathcal{Z}_{\mathcal{A}'}^* - na)
\end{aligned} \tag{43}$$

Окончательно с учетом 41 получаем верхнюю оценку для  $\mathcal{D}(\mathcal{Z}_{\mathcal{A}'})$

$$\mathcal{D}(\mathcal{Z}_{\mathcal{A}'}) < \mathcal{D}_{\mathcal{A}'}^* = (b - a)^2 \left[ \gamma_n * n + \int_{\gamma_n}^1 \frac{dx}{F(x)} \right] \tag{44}$$

Вернемся к неравенству 34. Имея в виду полученные оценки 41 и 44, выражения для  $\epsilon_n$  и  $\delta_n$  можно записать в следующем виде:

$$\epsilon_n = K \left( \frac{b}{a} - 1 \right) \left[ \gamma_n + \frac{1}{n} \int_{\gamma_n}^1 \frac{dx}{F(x)} \right] \tag{45}$$

$$\delta_n = \frac{1}{(K-1)^2[\gamma_n * n + \int_{\gamma_n}^1 \frac{dx}{F(x)}]} \quad (46)$$

Обозначим  $\mathcal{I}_n = \int_{\gamma_n}^1 \frac{dx}{F(x)}$ ,  $\Psi(n)$  - произвольная растущая от  $n$  функция,  $\lim_{n \rightarrow \infty} \Psi(n) = \infty$

### Теорема 1

Алгоритм  $\mathcal{A}'$  является асимптотически оптимальным при выполнении условий  $\lim_{n \rightarrow \infty} \mathcal{I}_n = \infty$  и

$$\frac{b}{a} \leq \frac{1}{\Psi(n)} \min\left\{\frac{1}{\gamma_n}, \frac{n}{\mathcal{I}_n}\right\} \quad (47)$$

Доказательство. Покажем, что при выполнении условий теоремы  $\epsilon_n \rightarrow 0$  и  $\delta_n \rightarrow 0$  с ростом  $n$ . Действительно, при  $K > 1$  имеем:

$$\begin{aligned} \delta_n &= \frac{1}{(\gamma_n n + \mathcal{I}_n)(K-1)^2} \leq \frac{1}{\mathcal{I}_n * (K-1)^2} \rightarrow 0 \\ \epsilon_n &= K\left(\frac{b}{a} - 1\right)\left(\gamma_n + \frac{1}{n}\mathcal{I}_n\right) \leq K * \frac{b}{a}\left(\gamma_n + \frac{1}{n}\mathcal{I}_n\right) \leq \\ &\leq \frac{K}{\Psi(n)} \min\left(\frac{1}{\gamma_n}, \frac{n}{\mathcal{I}_n}\right)\left(\gamma_n + \frac{1}{n}\mathcal{I}_n\right) = \\ &= \frac{K}{\Psi(n)} \min\left(1 + \frac{\mathcal{I}_n}{n\gamma_n}, 1 + \frac{n\gamma_n}{\mathcal{I}_n}\right) \leq \frac{2K}{\Psi(n)} \rightarrow 0 \end{aligned} \quad (48)$$

при  $n \rightarrow \infty$ . Теорема 1 доказана.

Замечание. Как было показано выше, для своей работы алгоритм  $\mathcal{A}'$  требует  $O(n^2)$  операций, что сравнимо с трудоемкостью записи исходной информации о задаче коммивояжера. Отсюда получаем, что при выполнении условий теоремы 1 алгоритм  $\mathcal{A}'$  является статистически эффективным.

### Равномерное распределение.

Определим условия асимптотической оптимальности алгоритма  $\mathcal{A}'$  для случая, когда веса ребер графа  $a_{ij}$  могут быть выбраны равновероятно из отрезка  $[a, b]$ ,  $a > 0$ . В этом случае нормированная интегральная функция распределения имеет вид  $F(x) = x$ ,  $0 \leq x \leq 1$ ,  $\gamma_n = F\left(\frac{1}{n}\right) = \frac{1}{n}$  и

$$\mathcal{I}_n = \int_{\gamma_n}^1 \frac{dx}{F(x)} = \int_{\frac{1}{n}}^1 \frac{dx}{x} = \ln(n) \quad (49)$$

Тогда из теоремы 1 непосредственно получаем результат, который может быть сформулирован как:

### Теорема 2

Если элементы  $a_{ij}$  матрицы  $A$  принимают значения равновероятно из отрезка  $[a, b]$ , то алгоритм  $\mathcal{A}'$  является асимптотически оптимальным при выполнении следующего условия:

$$\frac{b}{a} \leq \frac{n}{\ln(n)} \frac{1}{\Psi(n)} \quad (50)$$

Представляет интерес оценить величины  $\epsilon_n$  и  $\delta_n$ , фигурирующие в соотношении (25).

Учитывая специфику равномерного распределения можно получить более точные оценки для этих величин по сравнению с общим случаем. Выведем условия асимптотической оптимальности алгоритма  $\mathcal{A}'$  в случае равномерного распределения, проведя в сокращенном виде вычисления оценок для  $E(\mathcal{Z}_{\mathcal{A}'})$  и  $\mathcal{D}(\mathcal{Z}_{\mathcal{A}'})$  и  $P\{\mathcal{Z}_{\mathcal{A}'} \leq (1 + \epsilon_n)\mathcal{Z}\}$ . Согласно 38

$$E(l_k) = \int_0^1 (1-x)^{n-k} dx = \int_0^1 x^{n-k} dx = \frac{1}{n-k+1} \quad (51)$$

$$k = 1, 2, \dots, n-k; E(l_n) = E(l_n - 1) = \frac{1}{2}$$

С учетом (39)

$$\begin{aligned} E(\mathcal{Z}_{\mathcal{A}'}) &= \sum_{k=1}^n [a + (b-a)E(l_k)] = na + (b-a)\left(\frac{1}{2} + \sum_{k=1}^{n-1} \frac{1}{n-k+1}\right) \leq \\ &\leq na + (b-a)\left(\frac{1}{2} + \ln(n)\right) = \mathcal{Z}_{\mathcal{A}'}^* \end{aligned} \quad (52)$$

Оценим дисперсию  $d_k = \int_0^1 (x - E(l_k))^2 d\Phi_k(x)$  случайной величины  $l_k$  с учетом того, что для равномерного распределения  $\Phi_k(x) = 1 - (1-x)^{n-k}$

Используя (51), имеем

$$\begin{aligned} d_k &= \int_0^1 x^2 d\Phi_k(x) - E(l_k)^2 = x^2 \Phi_k(x) \Big|_0^1 - 2 \int_0^1 x \Phi_k(x) dx - E(l_k)^2 = \\ &= 1 - 2 \int_0^1 x [1 - (1-x)^{n-k}] dx - E(l_k)^2 = \\ &= \frac{2}{n-k+1} - \frac{2}{n-k+2} - \frac{1}{(n-k+1)^2}, \\ &k = 1, 2, \dots, n-1; d_n = d_{n-1} = \frac{1}{12} \end{aligned} \quad (53)$$

Отсюда с учетом определения дисперсии величины  $\mathcal{Z}_{\mathcal{A}'}$ , получим

$$\begin{aligned} \frac{1}{(b-a)^2} D(\mathcal{Z}_{\mathcal{A}'}) &= \sum_{k=1}^n d_k = \frac{1}{12} + \sum_{k=1}^{n-1} \left( \frac{2}{n-k+1} - \frac{2}{n-k+2} - \frac{1}{(n-k+1)^2} \right) = \\ &= \frac{1}{12} - \frac{2}{n+1} + 1 - \sum_{k=1}^{n-1} \frac{1}{(n-k+1)^2} \leq \frac{13}{12} - \frac{2}{n+1} - \frac{1}{4} - \int_3^{n+1} \frac{dx}{x^2} = \\ &= \frac{13}{12} - \frac{2}{n+1} - \frac{1}{4} + \frac{1}{n+1} - \frac{1}{3} < 0.417 = \frac{\mathcal{D}_{\mathcal{A}'}^*}{(b-a)^2} \end{aligned} \quad (54)$$

Приведем оценку вероятности невыполнения соотношения (25) для случая равномерного распределения:

$$\begin{aligned}
P\{\mathcal{Z}_{\mathcal{A}'} > (1 + \epsilon_n)\mathcal{Z}\} &\leq P\{\mathcal{Z}_{\mathcal{A}'} > (1 + \epsilon_n)na\} \leq \\
&\leq P\{\mathcal{Z}_{\mathcal{A}'} + \mathcal{Z}_{\mathcal{A}'}^* - E(\mathcal{Z}_{\mathcal{A}'}) > (1 + \epsilon_n)na\} = \\
&= P\{\mathcal{Z}_{\mathcal{A}'} - E(\mathcal{Z}_{\mathcal{A}'}) > (1 + \epsilon_n)na - \mathcal{Z}_{\mathcal{A}'}^*\} \leq \\
&\leq P\{|\mathcal{Z}_{\mathcal{A}'} - E(\mathcal{Z}_{\mathcal{A}'})| > (1 + \epsilon_n)na - \mathcal{Z}_{\mathcal{A}'}^*\} \leq \\
&\leq \frac{D(\mathcal{Z}_{\mathcal{A}'})}{[(1 + \epsilon_n)na + \mathcal{Z}_{\mathcal{A}'}^*]^2} \leq \\
&\leq \frac{0.417(b - a)^2}{[1 + \epsilon_n)na - (nu + (b - a)(\frac{1}{2} + \ln(n)))]^2} = \\
&= \frac{0.417}{[\frac{n\epsilon_n}{\frac{b}{a}-1} - \ln(n) - \frac{1}{2}]^2}
\end{aligned} \tag{55}$$

Положим  $\epsilon_n = \frac{c}{\Psi(n)}$ , константа  $c > 1$ , и пусть  $\frac{a}{b} \leq \frac{n}{\ln(n)} \frac{1}{\Psi(n)}$ . Тогда (55) может быть продолжено следующим образом:

$$\frac{0.417}{[\frac{n\frac{c}{\Psi(n)}}{(\frac{b}{a}-1)} - \frac{1}{2} - \ln(n)]} \leq \frac{0.417}{[\frac{c\frac{b}{a}\ln(n)}{(\frac{b}{a}-1)} - \frac{1}{2} - \ln(n)]^2} = \frac{0.417}{[(c - 1)\ln(n) - \frac{1}{2}]^2} = \delta_n \tag{56}$$

Таким образом, окончательная оценка для вероятности выполнения соотношения (25) примет вид:

$$P\{\mathcal{Z}_{\mathcal{A}'} \leq (1 + \epsilon_n)\mathcal{Z}\} \geq 1 - \frac{0.417}{[(c - 1)\ln(n) - \frac{1}{2}]^2} \tag{57}$$

Нетрудно заметить, что эта величина, характеризующая точность получаемого решения, улучшается с ростом  $\Psi(n)$ , но при этом ухудшается оценка для величины  $\frac{b}{a}$ . Выберем функцию  $\Psi(n)$  таким образом, чтобы произведение верхних оценок для  $\epsilon_n$  и  $\frac{b}{a}$  стремилось к (25) с ростом  $n$ . Такому условию отвечает функция  $\Psi(n) = \sqrt{\frac{cn}{\ln(n)}}$ . При этом оценки для  $\epsilon_n$  и  $\frac{b}{a}$  принимают вид:

$$\frac{b}{a} \leq \sqrt{\frac{n}{c\ln(n)}} \tag{58}$$

$$\epsilon_n \leq \sqrt{\frac{c\ln(n)}{n}} \tag{59}$$

Анализ соотношений 57-59, полученных для равномерного распределения показывает, что уменьшение константы  $c$  "улучшает" оценки для  $\frac{b}{a}$  и  $\epsilon_n$  и "ухудшает" оценку вероятности  $P\{\mathcal{Z}_{\mathcal{A}'} \leq (1 + \epsilon_n)\mathcal{Z}\}$

$\beta$  - распределение. Очень часто в опытно-конструкторских разработках в промышленности и научно-исследовательских проектах длительности  $a_{ij}$  отдельных операций предполагается распределенными по следующему закону /  $\beta$  - распределение /:

$$f(\xi) = \begin{cases} \frac{(\xi-a)^{\alpha} (b-\xi)^{\gamma}}{(b-a)^{\alpha+\gamma+1} \beta(\alpha+1, \gamma+1)}, & \xi \in [a, b] \\ 0, & \xi \notin [a, b] \end{cases} \quad (60)$$

где

$$\beta(\alpha, \gamma) = \int_0^1 x^{\alpha-1} (1-x)^{\gamma-1} dx = \frac{\Gamma(\alpha)\Gamma(\gamma)}{\Gamma(\alpha+\gamma)} \quad (61)$$

где  $\alpha$  и  $\gamma$  - параметры распределения. В нормированном виде функция плотности  $f(\xi)$  запишется следующим образом:

$$f(\xi') = \begin{cases} \frac{(\xi')^{\alpha} (1-\xi')^{\gamma}}{\beta(\alpha+1, \gamma+1)}, & \xi' \in [0, 1] \\ 0, & \xi' \notin [0, 1] \end{cases} \quad (62)$$

Рассмотрим частный случай  $\beta$  - распределения, когда  $\gamma = 0$ ,  $\alpha > 0$ . В этом случае интегральная функция распределения равна

$$F(x) = \int_0^x f(\xi') d\xi' = \int_0^x \frac{x}{\beta(\alpha+1, 1)} = x^{\alpha+1} \quad (63)$$

Вычислим величину  $\gamma_n$ , фигурирующую в условиях теоремы 1:

$$\gamma_n = F^{-1}\left(\frac{1}{n}\right) = \frac{1}{\alpha+1 \sqrt[n]{n}} \quad (64)$$

Тогда первое из условий теоремы 1 выполняется в силу  $\alpha > 0$

$$\mathcal{I}_n = \int_{\gamma_n}^1 \frac{dx}{F(x)} = \int_{\gamma_n}^1 \frac{dx}{x^{\alpha+1}} = \frac{1}{\alpha} (n^{\frac{\alpha}{\alpha+1}} - 1) \rightarrow \infty, \quad n \rightarrow \infty \quad (65)$$

Второе условие принимает вид:

$$\begin{aligned} \frac{b}{a} &\leq \frac{1}{\Psi(n)} \min\left(\alpha+1 \sqrt[n]{n}, \frac{\alpha n}{n^{\frac{\alpha}{\alpha+1}} - 1}\right) = \\ &= \frac{\alpha+1 \sqrt[n]{n}}{\Psi(n)} \min\left(1, \frac{\alpha}{1 - n^{-\frac{\alpha}{\alpha+1}}}\right) \end{aligned} \quad (66)$$

Учитывая, что

$$\min\left(1, \frac{\alpha}{1 - n^{-\frac{\alpha}{\alpha+1}}}\right) \geq \min(1, \alpha) \quad (67)$$

Получаем следующее условие асимптотической оптимальности алгоритма  $\mathcal{A}'$  для  $\beta$  - распределения в частном случае  $\gamma = 0$ ,  $\alpha > 0$ :

$$\frac{b}{a} \leq \frac{\alpha+1 \sqrt[n]{n}}{\Psi(n)} \min(1, \alpha) \quad (68)$$



4. Оптимальность жадного алгоритма для графов с случайным распределением весов ребер.

4.1. Граф с равномерным распределением весов ребер на промежутке. Оценка оптимальности жадного алгоритма.

## **Список использованных источников и литературы.**

### **Статьи из журнала:**

1. Э. Х. Гимади, А. Ле Галлу, А. В. Шахшнейдер, “Вероятностный анализ одного алгоритма приближённого решения задачи коммивояжёра на неограниченных сверху входных данных”, Дискретн. анализ и исслед. опер., 15:1 (2008), 23–43; J. Appl. Industr. Math., 3:2 (2009), 207–221

2. Rosenkrantz, Daniel J.; Stearns, Richard E.; Lewis, II, Philip M – An Analysis of Several Heuristics for the Traveling Salesman Problem. September 1977 SIAM Journal on Computing 6(3):563-581

3. Э. Х. Гимади, В. А. Перепелица, “Асимптотический подход к решению задачи коммивояжера”, Управляемые системы, 1974, № 12, 35–45

### **Книги:**

1. Gregory Gutin, Abraham P. Punnen, The travelling salesman problem and its variations. - 2006