
Исследование алгоритмов решения задачи к коммивояжеров

Научный руководитель,
проф., д.т.н.

Ю.Л. Костюк

Исполнитель,
аспирант

М.С. Пожидаев

Томский государственный университет. Факультет информатики

Введение

Содержание работы - задача к коммивояжёров.
Цель - минимизация стоимости объезда городов
к коммивояжёрами.

На практике постановка задачи к коммивояжёров
имеет множество вариантов.

В данной работе сформулирована упрощенная
постановка задачи без учета максимальной
загрузки одного коммивояжёра, задержки при
посещении каждого города, возможности
существования нескольких баз и т. д.

Формальная постановка задачи

Среди городов вводится база - город, где начинаются и заканчиваются все пути коммивояжеров. Стоимость переезда из города i в город j будет задаваться матрицей, где нужное значение будет храниться на пересечении i -той строки и j -того столбца. Определим задачу так:

1. На вход алгоритма поступает матрица расстояний между n городами и множество значений расстояний от каждого города до базы.
2. Необходимо найти k замкнутых маршрутов, где $k > 1$, со следующими свойствами:
 - 2.1. Через каждый город должен проходить один и только один маршрут, причём только один раз;
 - 2.2. Все маршруты должны проходить через базу;
 - 2.3. Для любой пары полученных маршрутов количество городов в них не должно отличаться больше чем на единицу.
3. Суммарная длина полученных маршрутов должна быть как можно меньше.

Рассматриваемые алгоритмы

- Оптимальное разрезание общего маршрута по всем городам
- Предварительное разделение на группы для каждого коммивояжёра
 1. Последовательное деление на 2 группы
 - a. Добавление городов на основе минимума расстояний
 - b. Добавление городов на основе разности расстояний
 2. Последовательное деление на 3 группы
- Обменная оптимизация
 - Для всех городов
 - Для ограниченного множества городов

Угловая сортировка городов на ПЛОСКОСТИ

- Является исключением среди рассматриваемых методов, поскольку не удовлетворяет формальной постановке задачи.
- Использует информацию об углах между городами, если они расположены на плоскости.

Алгоритм разрезания "общего" маршрута

- 1. Решить задачу коммивояжёра
- 2. Найти количество городов для каждого коммивояжёра
- 3. В предварительно построенном маршруте выбрать некоторый город в качестве начального.
- 4. Городами для объезда 1-вым коммивояжёром будут первые $N[1]$ городов, начиная с выбранного начального. Последующий отрезок маршрута из $N[2]$ городов будет участком для 2-го коммивояжёра и т. д. В результате этого этапа мы получим k незамкнутых последовательностей нужной длины.
- 5. К каждой такой последовательности добавим базу и замкнём её.
- 6. Запомним полученную длину всех маршрутов и повторим операцию с шага 4, но начальный город сместим на один относительно предыдущего.
- 7. Шаг 6 будем повторять N_{\min} раз, где N_{\min} - это число на один меньше, чем длина самого короткого пути среди коммивояжёров, полученных на шаге 2.

Метод предварительного разделения городов на группы для каждого коммивояжёра

Наиболее эффективным является последовательное разделение на 2 группы. Рассмотрены 2 метода выбора нового города для добавления:

- На основе минимума расстояний.
- На основе разности расстояний. Качество работы выше.

Обменная ОПТИМИЗАЦИЯ

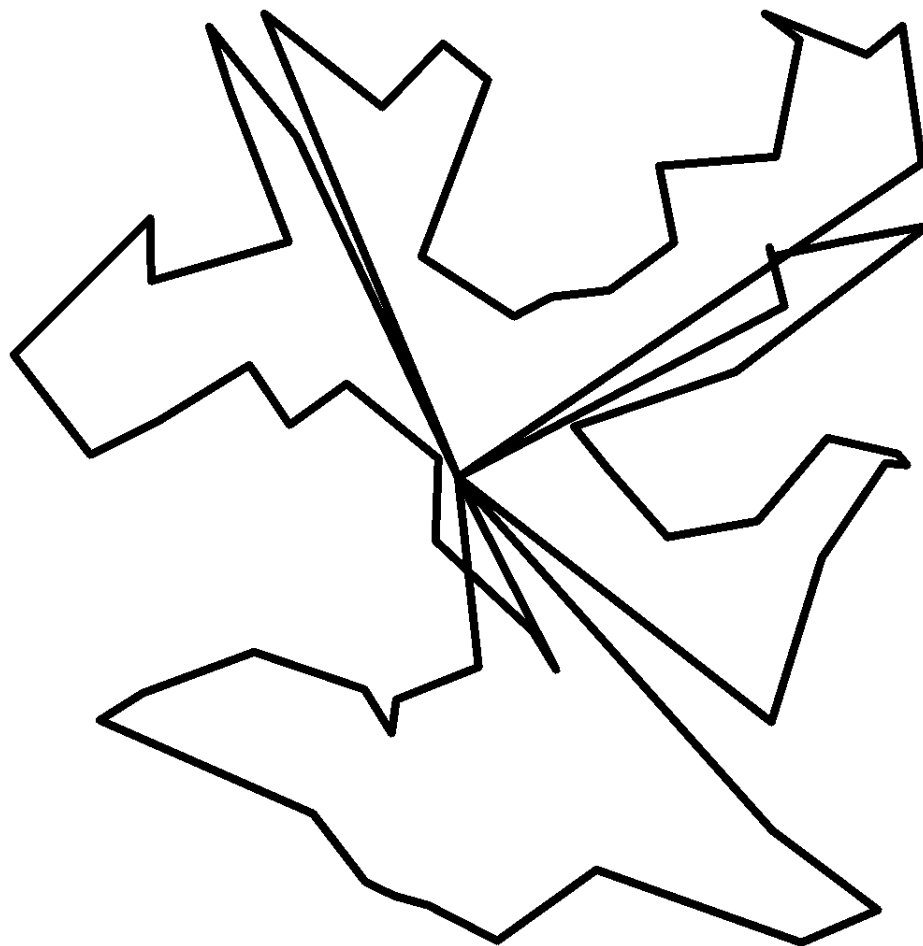
После очередного разделения городов на 2 группы:

- 1. Рассмотрим все возможные пары городов, где первый город будет принадлежать первой, а второй - второй группе;
 - 2. Для каждой такой пары выполним пробный обмен городами. То есть первый город попадёт во вторую группу, а второй - в первую.
 - 3. Выполнив обмен, подсчитаем сумму длин рёбер минимального остова в каждой группе, прибавив длину самого длинного ребра в нём.
 - 4. Найдём оценку из шага 3 для всех возможных пар. Если минимальное значение таких оценок меньше, чем в исходных группах, то совершим обмен.
-
- Оптимизация проводится для всех городов в группах
 - Оптимизация проводится для ограниченного множества городов

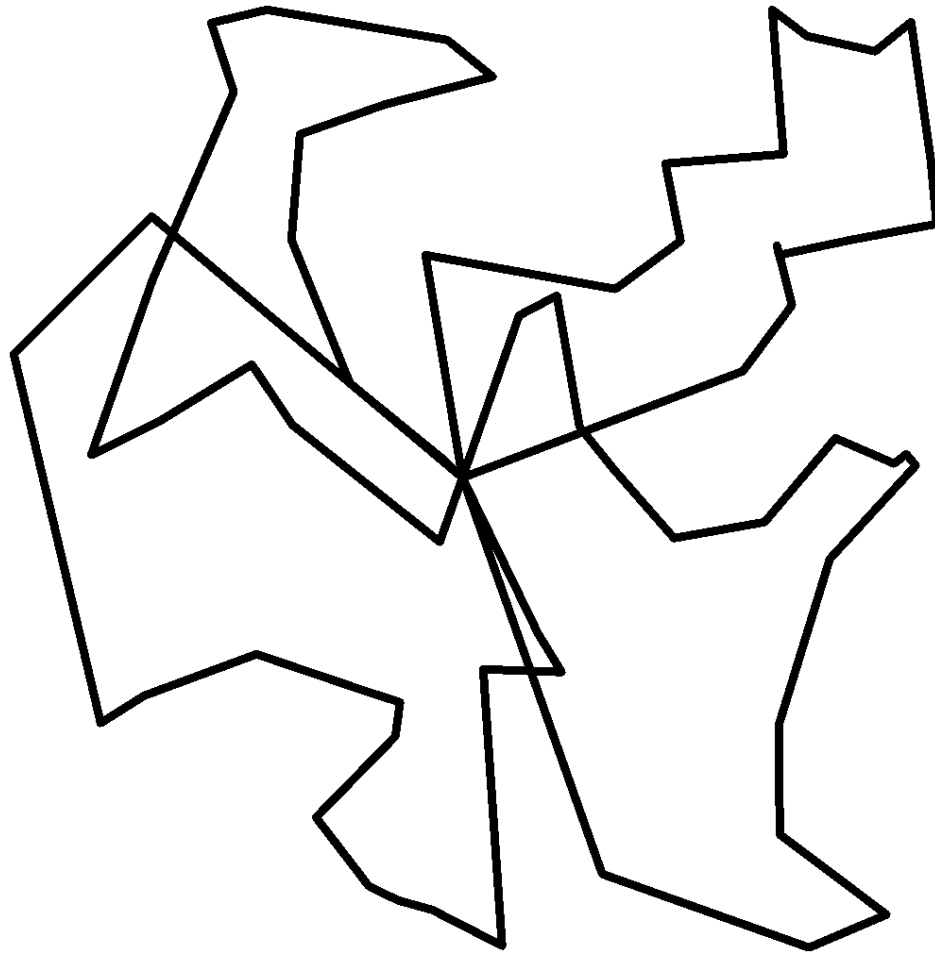
Угловая сортировка городов на ПЛОСКОСТИ

- 1. Для каждого города вычисляются углы между ними и осью X.
- 2. Множество городов сортируются по возрастанию углов.
- 3. Вычисляется количество городов для каждой будущей группы. По результатам вычислений получаем массив N_i .
- 4. Первые N_1 городов из отсортированного списка отнесём к первой группе, последующие N_2 городов - ко второй и. т. д., к каждой группе добавим по базе.
- 5. Для каждой полученной группы вычисляется минимальная оценка возможного решения.
- 6. Повторяем действия на шаге 4, но при построении отступаем на один город и находим оценку, описанную на шаге 5.
- 7. Выбирается то разбиение, где оценка минимальна и выполняется поиск окончательных маршрутов решением простой ЗК внутри каждой группы.

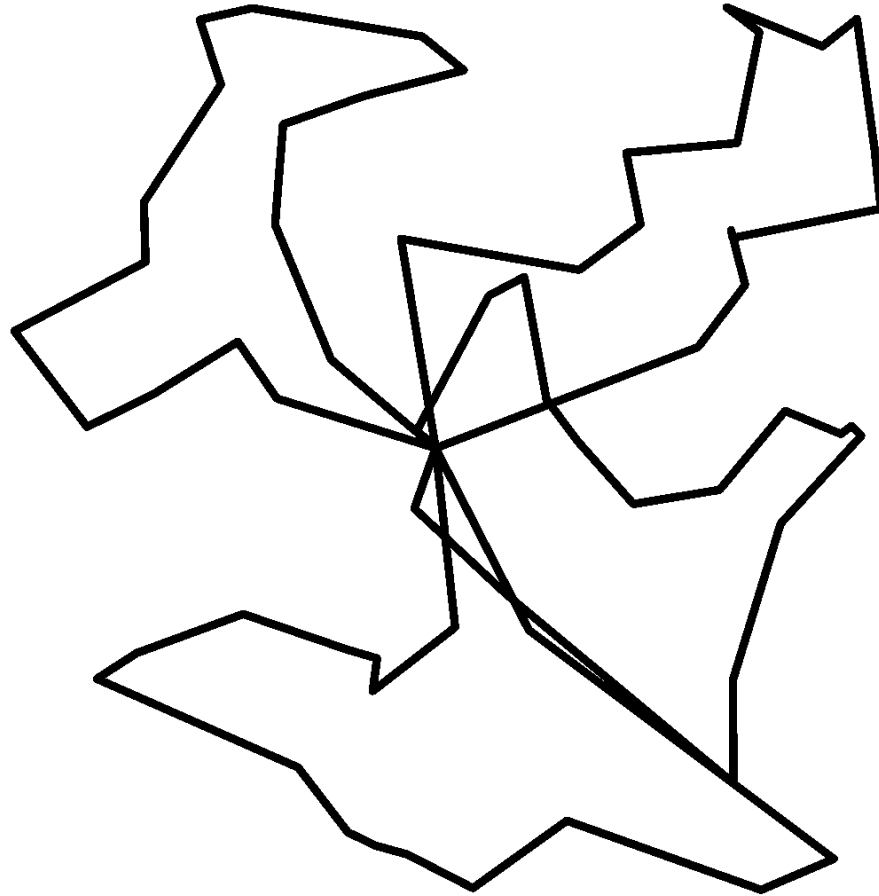
Алгоритм "разрезание готового маршрута"



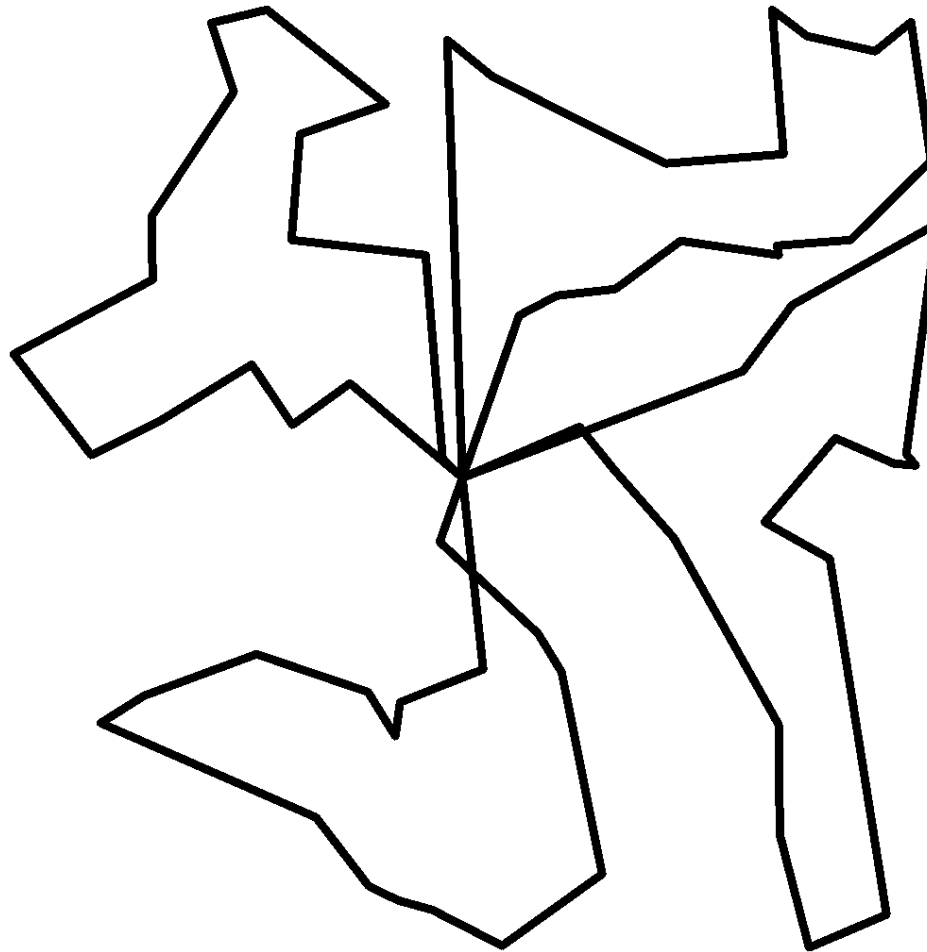
Алгоритм "деление по минимуму"



Алгоритм "деление по разности"



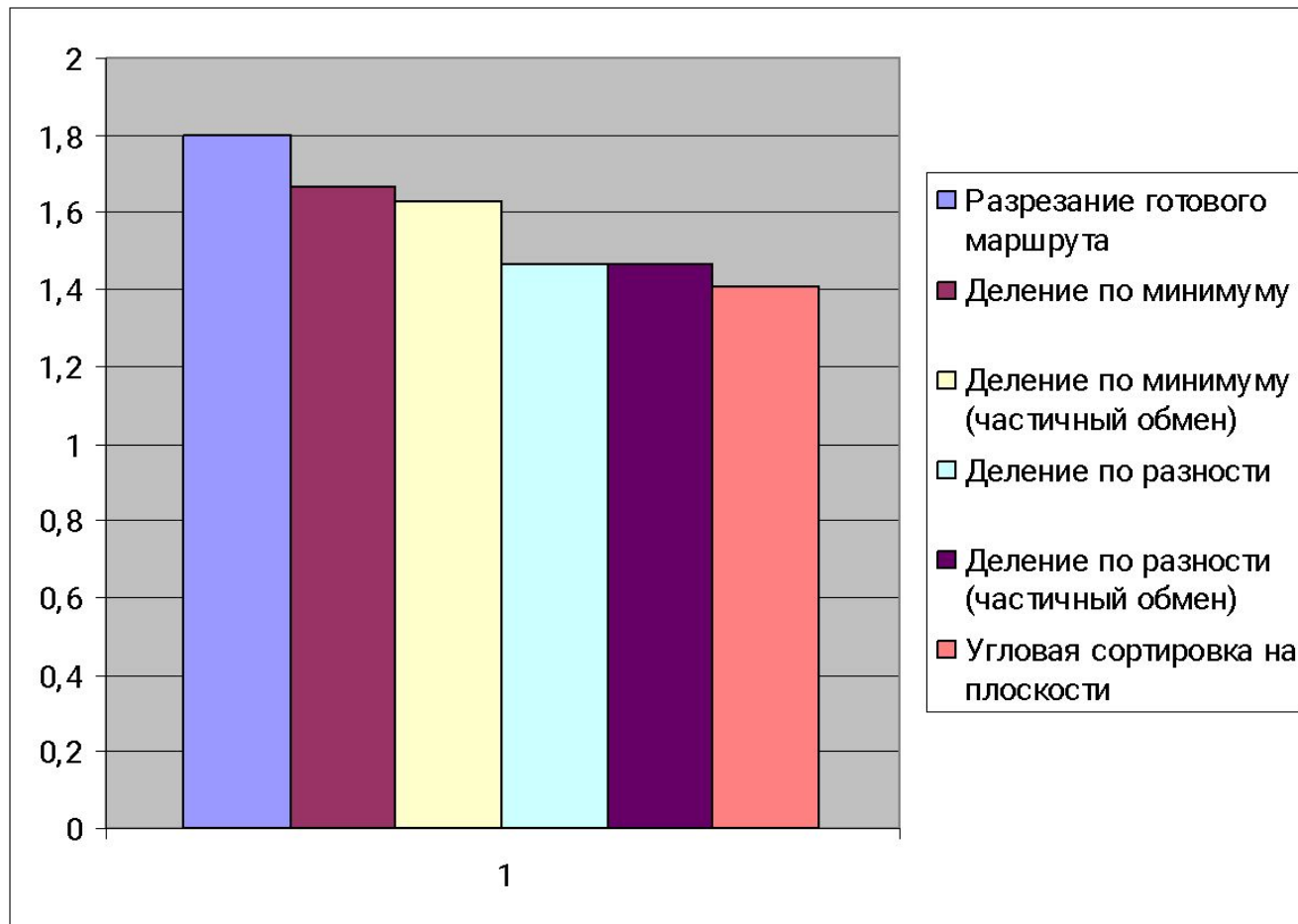
Алгоритм "угловая сортировка на плоскости"



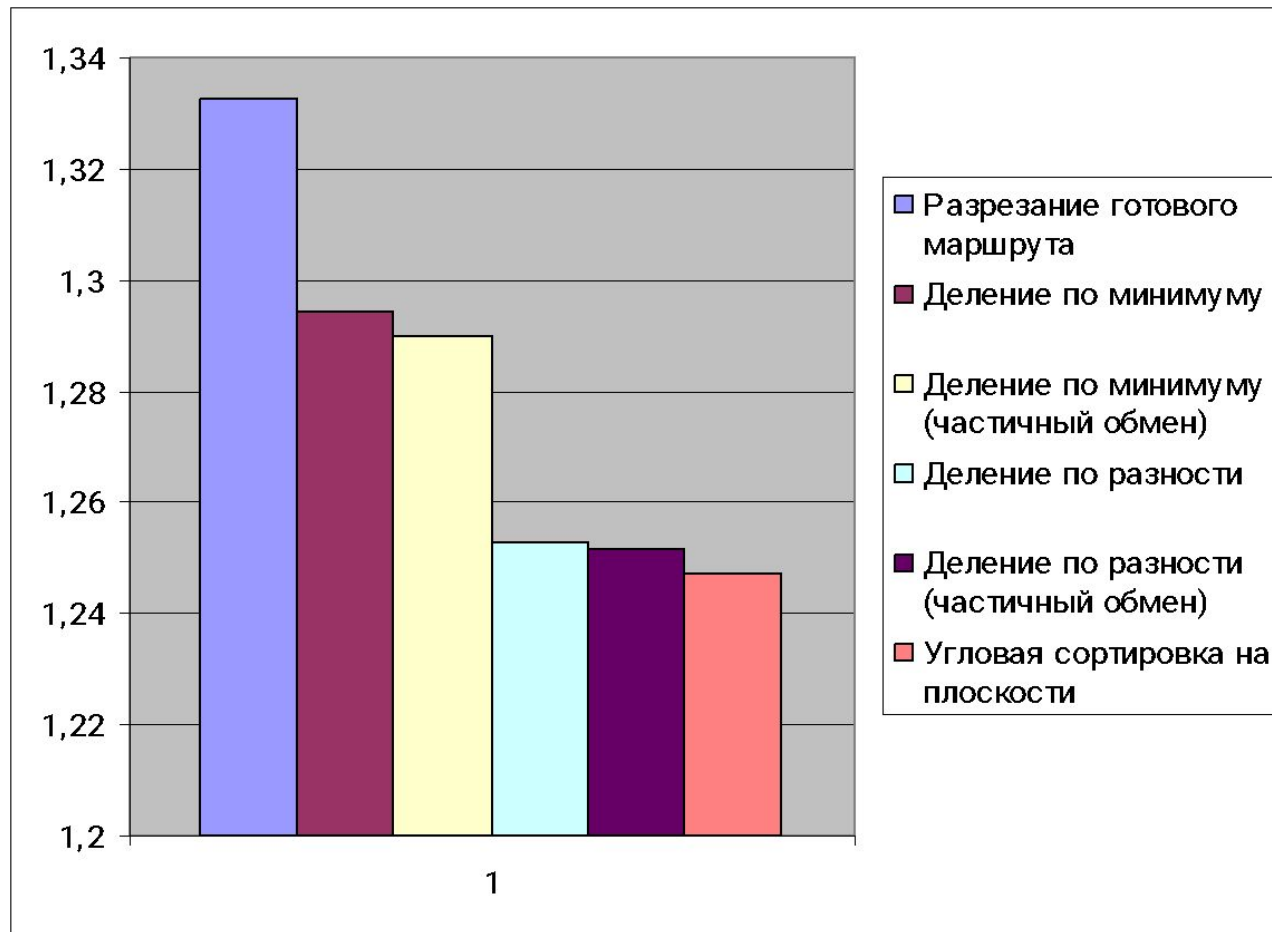
Вычислительный эксперимент

- При тестировании описанных методов выполнялся многократный их запуск на случайно сгенерированном множестве городов. При генерации города располагались на плоскости равномерно в единичном квадрате с независимыми координатами. Результаты приведены в отношении к нижней оценке точного решения задачи коммивояжёра.
- Обменная оптимизация проводилась только в частичном варианте, поскольку многократное нахождение минимального остова занимает слишком много времени.

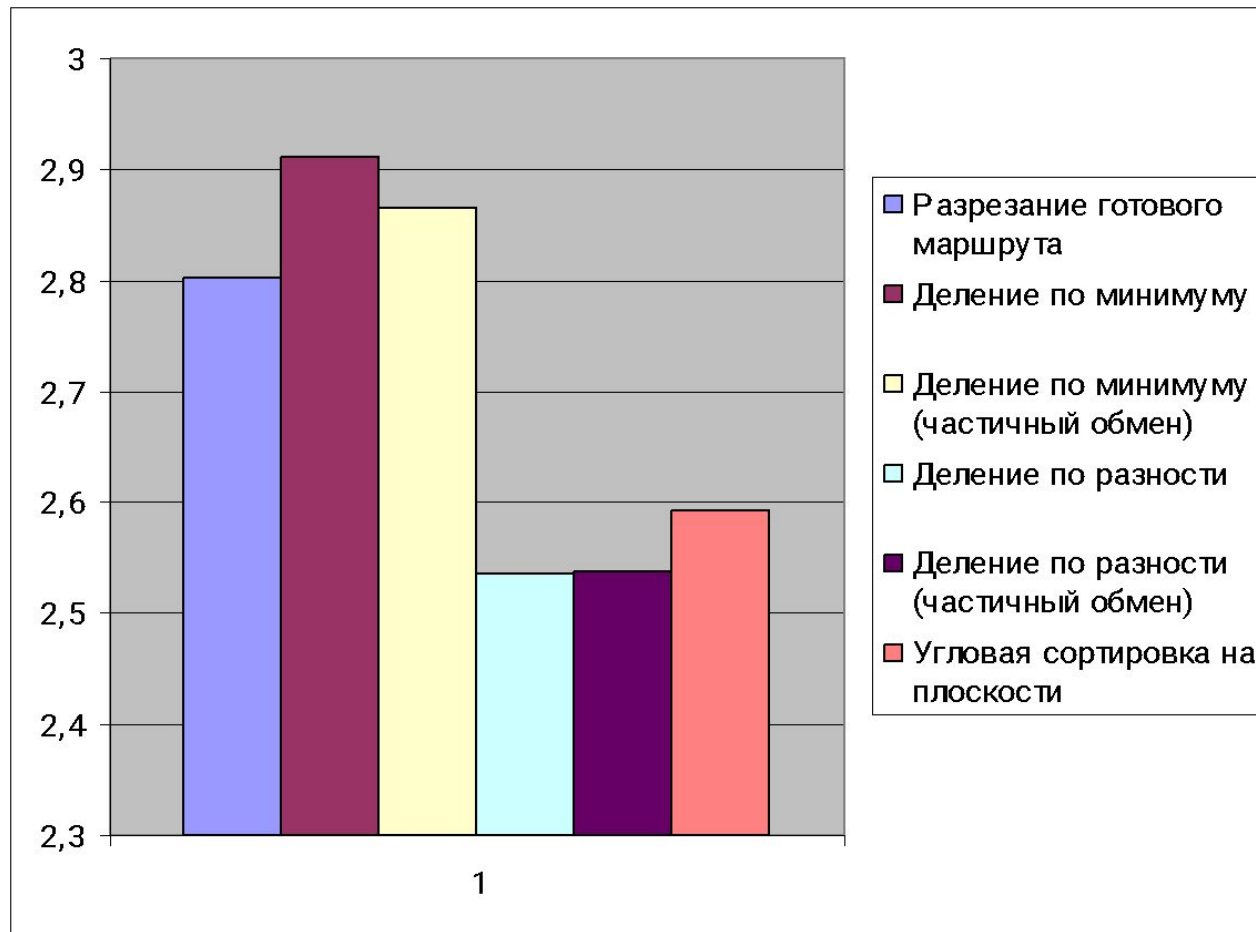
Результаты обработки задачи на 64 города на 4 коммивояжера



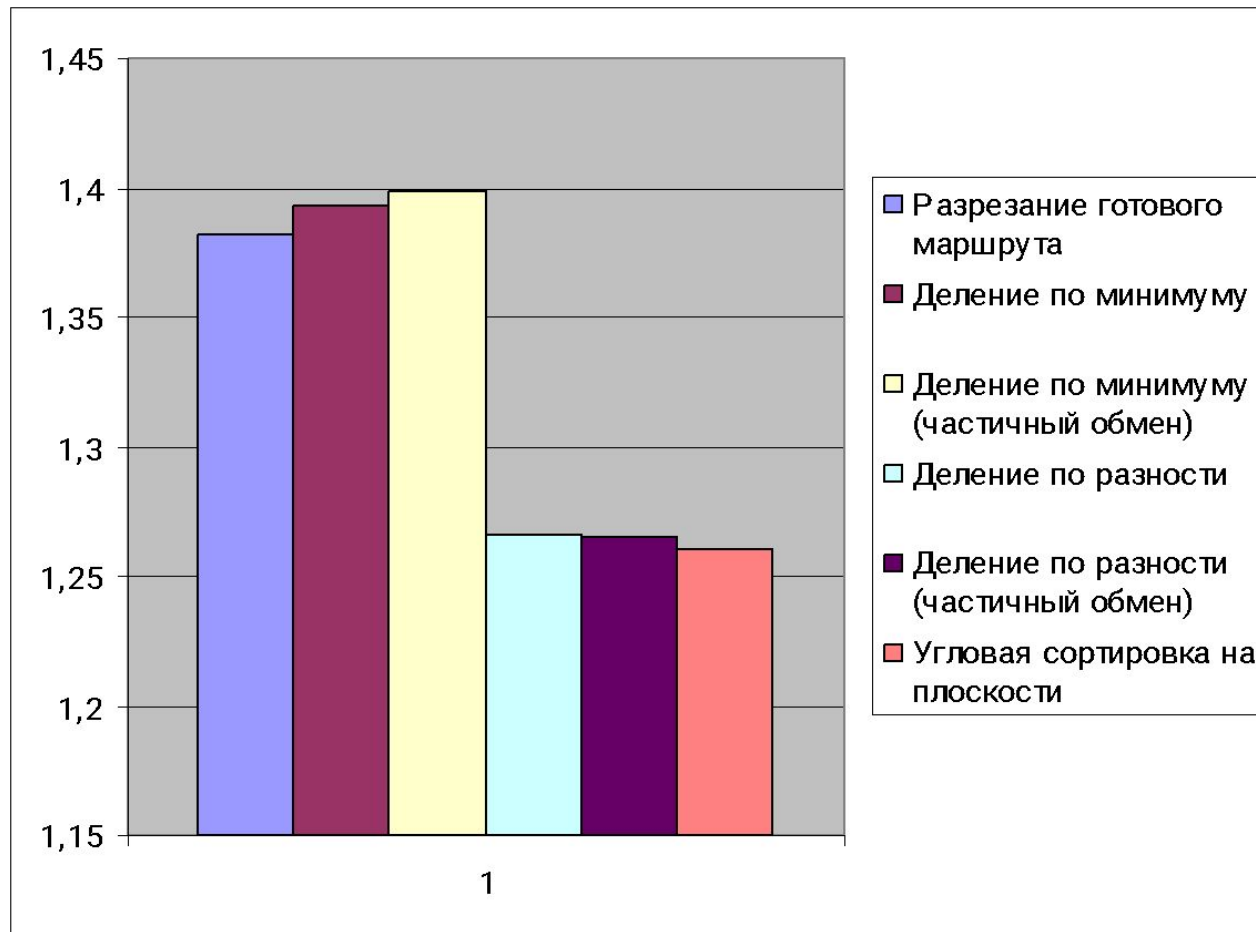
Результаты обработки задачи на 512 городов на 2 коммивояжера



Результаты обработки задачи на 512 городов на 32 коммивояжера



Результаты обработки задачи на 1024 городов на 4 коммивояжера



Заключение

- Сформулирована сбалансированная задача коммивояжеров
- Предложены некоторые методы приближенного её решения
- Поставлен вычислительный эксперимент для сравнения качества работы предложенных алгоритмов и выполнен статистический анализ ранговым критерием Уилкоксона
- Подготовлена программная библиотека

Спасибо за внимание