

УНИВЕРСИТЕТ ИТМО  
Факультет прикладной информатики

Квантовые подходы к обработке информации и искусственному интеллекту

Лабораторная работа №4  
Вариант 22

«Задача максимального разреза (*Max-Cut Problem*)»

Выполнил:  
Проскуряков Роман Владимирович, 409413

Номер группы:  
К3339

Преподаватель:  
Чуруксаев Иван Валерьевич

Санкт-Петербург, 2026

## ЦЕЛЬ РАБОТЫ

Целью данной лабораторной работы является исследование и сравнение эффективности классических оптимизационных алгоритмов для решения NP-трудной задачи Max-Cut. В рамках работы реализованы и протестированы два подхода: жадный алгоритм с локальными улучшениями (baseline) и метод имитации отжига (Simulated Annealing). Экспериментальная часть направлена на анализ влияния параметров алгоритма имитации отжига на качество получаемых решений и скорость сходимости, а также на сравнение производительности двух методов на взвешенном графе из 20 вершин.

## ХОД ВЫПОЛНЕНИЯ

### 1 Формулировка задачи моего варианта

**Дано:**

- Неориентированный граф  $G = (V, E)$
- Множество вершин:  $V = \{0, 1, \dots, n - 1\}$
- Множество рёбер:  $E \subseteq V \times V$
- Веса рёбер:  $w_{ij} \geq 0$  для каждого  $(i, j) \in E$  (по умолчанию  $w_{ij} = 1$ )

**Требуется найти:**

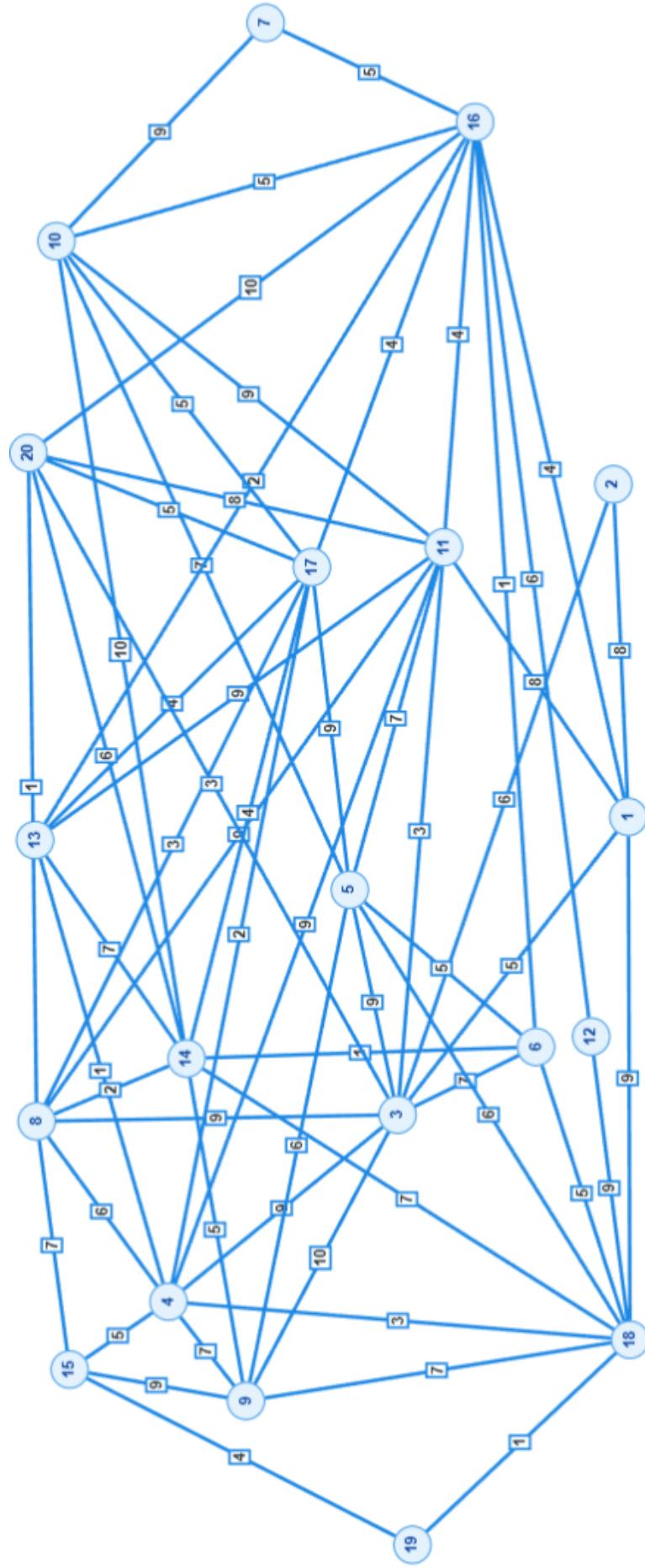
Разбиение вершин графа на два непересекающихся множества  $S$  и  $\bar{S} = V \setminus S$  такое, что вес разреза максимален:

$$\text{Cut}(S, \bar{S}) = \sum_{\substack{i \in S, j \in \bar{S} \\ (i, j) \in E}} w_{ij}$$

**Целевая функция:**

$$\max_{S \subset V} \text{Cut}(S, \bar{S})$$

Исходный граф моего варианта (n=20)



## 2 Математическая модель (QUBO-формулировка)

### 2.1. Бинарные переменные

$$x_i \in \{0, 1\}, \quad i \in V$$

Интерпретация:

- $x_i = 1$  означает, что вершина  $i$  принадлежит множеству  $S$
- $x_i = 0$  означает, что вершина  $i$  принадлежит множеству  $\bar{S}$

Число переменных:  $n$

### 2.2. Целевая функция

Ребро  $(i, j)$  попадает в разрез, если вершины  $i$  и  $j$  находятся в разных множествах, то есть когда  $x_i \neq x_j$ .

Индикатор разреза для ребра:

$$1_{\text{cut}}(i, j) = x_i(1 - x_j) + x_j(1 - x_i) = x_i + x_j - 2x_i x_j$$

Вес разреза:

$$\text{Cut}(x) = \sum_{(i,j) \in E} w_{ij}(x_i + x_j - 2x_i x_j)$$

Максимизация разреза эквивалентна минимизации:

$$E_{\text{obj}}(x) = - \sum_{(i,j) \in E} w_{ij}(x_i + x_j - 2x_i x_j)$$

Раскрывая и группируя:

$$E_{\text{obj}}(x) = 2 \sum_{(i,j) \in E} w_{ij} x_i x_j - \sum_{i \in V} d_i x_i$$

где  $d_i = \sum_{j:(i,j) \in E} w_{ij}$  — взвешенная степень вершины  $i$ .

### 2.3. Итоговая энергия

$$E_{\text{total}}(x) = E_{\text{obj}}(x)$$

Задача:  $\min_{x \in \{0,1\}^n} E_{\text{total}}(x)$

### 3 Описание реализованных алгоритмов

#### 3.1. Имитация отжига (Simulated Annealing)

**Идея:** Метод глобальной стохастической оптимизации, основанный на аналогии с физическим процессом отжига металлов. Алгоритм начинает с высокой "температуры", при которой допускаются переходы в состояния с большей энергией (для исследования пространства решений и избежания локальных минимумов). Затем температура постепенно снижается, и алгоритм становится более "жадным".

**Основные компоненты:**

1. **Температурный график:** Управляет вероятностью принятия ухудшающих решений

$$T(t) = T_0 \cdot \alpha^t$$

где  $T_0$  — начальная температура,  $\alpha \in (0, 1)$  — коэффициент охлаждения,  $t$  — номер итерации

2. **Критерий принятия решения:** Для перехода из состояния с энергией  $E$  в состояние с энергией  $E'$ :

- Если  $\Delta E = E' - E < 0$  (улучшение) — принять всегда
- Если  $\Delta E \geq 0$  (ухудшение) — принять с вероятностью:

$$P(\text{accept}) = e^{-\Delta E / T}$$

При высокой температуре  $T$  эта вероятность близка к 1 (принимаем почти любые ухудшения). При низкой температуре  $T \rightarrow 0$  эта вероятность стремится к 0 (становимся жадными).

- **Начальная температура  $T_0$ :** Обычно подбирается так, чтобы в начале принималось около 80% ухудшающих переходов. Типичный диапазон:  $T_0 \in [1.0, 3.0]$
- **Коэффициент охлаждения  $\alpha \in [0.93, 0.99]$ :** Чем ближе к 1, тем медленнее охлаждение. Типичные значения: 0.95-0.97.
- **Критерий остановки:** Обычно  $T < T_{min}$  (например,  $T_{min} = 0.001$ ) или фиксированное число итераций.

#### 3.2. Жадный алгоритм + локальные улучшения (Baseline)

**Идея:** Простой жадный алгоритм с последующими локальными улучшениями.

**Алгоритм:**

1. **Жадное построение:**

- Начать с произвольного разбиения (например, случайное)
- Для каждой вершины вычислить выгоду переноса в другое множество:

$$\text{gain}_i = \sum_{j \in \bar{S}_i, (i,j) \in E} w_{ij} - \sum_{j \in S_i, (i,j) \in E} w_{ij}$$

где  $S_i$  — текущее множество вершины  $i$ ,  $\bar{S}_i$  — противоположное

- Переносить вершины с положительной выгодой

**2. Локальные улучшения (2-opt):**

- Повторять:
  - Для каждой вершины  $i$ : вычислить изменение разреза при переносе
  - Если есть улучшение → перенести вершину
- Пока есть улучшения

**3. Возврат:** Итоговое разбиение

## **4 Процесс подбора параметров**

Для алгоритма имитации отжига был проведён систематический поиск оптимальных параметров. Исследовались следующие диапазоны:

- Начальная температура  $T_0 \in \{1.0, 1.5, 2.0, 2.5, 3.0\}$
- Коэффициент охлаждения  $\alpha \in \{0.90, 0.93, 0.96, 0.99\}$

Для каждой из 20 комбинаций параметров было выполнено по 10 независимых запусков алгоритма с различными начальными случайными seed (от 0 до 9). Для каждого запуска фиксировались:

- Вес найденного разреза
- Баланс разбиения (отношение размера меньшего множества к общему числу вершин)
- Время работы
- Траектория изменения лучшего найденного решения во времени

## **5 Проведённые эксперименты**

Выполнялось 10 запусков SA для каждой комбинации  $T_0$  и  $\alpha$ . Полученные для каждой комбинации параметров значения усреднялись и отображались в виде графиков и текст. Так же проводилось сравнение полученных данных с усреднёнными данными работы жадного алгоритма.

## **РЕЗУЛЬТАТЫ**

<https://colab.research.google.com/drive/1Cy1VvmVKpXbAynQlkUa68bJGukZ1C3Cp#scrollTo=9Bs8Hxz7vh7->

## **1 Статистический анализ и сравнение алгоритмов (программный вывод)**

=====

ЭКСПЕРИМЕНТАЛЬНОЕ СРАВНЕНИЕ АЛГОРИТМОВ ДЛЯ MAX-CUT

Начальная температура  $T_0 = 1.0$

Коэффициент охлаждения  $\alpha = 0.9$

## СТАТИСТИЧЕСКИЙ АНАЛИЗ

Имитация отжига (10 запусков):

Средний разрез:  $260.70 \pm 14.10$

Медиана разреза: 256.50

Лучший разрез: 280.00

Худший разрез: 245.00

Среднее время:  $0.002 \pm 0.000$  сек

Среднее время до лучшего: 0.002 сек

Средний баланс:  $0.460 \pm 0.030$

## СРАВНИТЕЛЬНАЯ ТАБЛИЦА

Сравнительная таблица

Метод	Средний разрез	Лучший	Худший	Ср. время (сек)	Balance
SA	$262.50 \pm 17.96$	280.00	234.00	0.0079	$0.465 \pm 0.023$
Baseline	$259.90 \pm 14.14$	280.00	242.00	0.0067	$0.455 \pm 0.027$

## АНАЛИЗ ЛУЧШЕГО РЕШЕНИЯ

Лучшее решение SA:

Вес разреза: 280.00

Баланс: 0.450

Время: 0.003 сек

Время до лучшего: 0.003 сек

Лучшее решение Baseline:

Вес разреза: 280.00

Баланс: 0.450

Время: 0.006 сек

Лучшее разбиение SA:

Множество S (11 вершин): [0, 3, 4, 5, 7, 8, 9, 11, 12, 18, 19]

Множество  $\bar{S}$  (9 вершин): [1, 2, 6, 10, 13, 14, 15, 16, 17]

Теоретическая оценка:

Общий вес рёбер: 346

Достигнуто: 80.9% от общего веса

---

---

## ДОПОЛНИТЕЛЬНАЯ СТАТИСТИКА

---

---

Процент улучшения SA над Baseline для каждого запуска:

Запуск 1: -11.07%

Запуск 2: +6.46%

Запуск 3: +8.68%

Запуск 4: -12.50%

Запуск 5: +6.85%

Запуск 6: -0.40%

Запуск 7: +0.00%

Запуск 8: -1.59%

Запуск 9: +0.00%

Запуск 10: +9.38%

Изменение баланса:

Baseline: 0.455

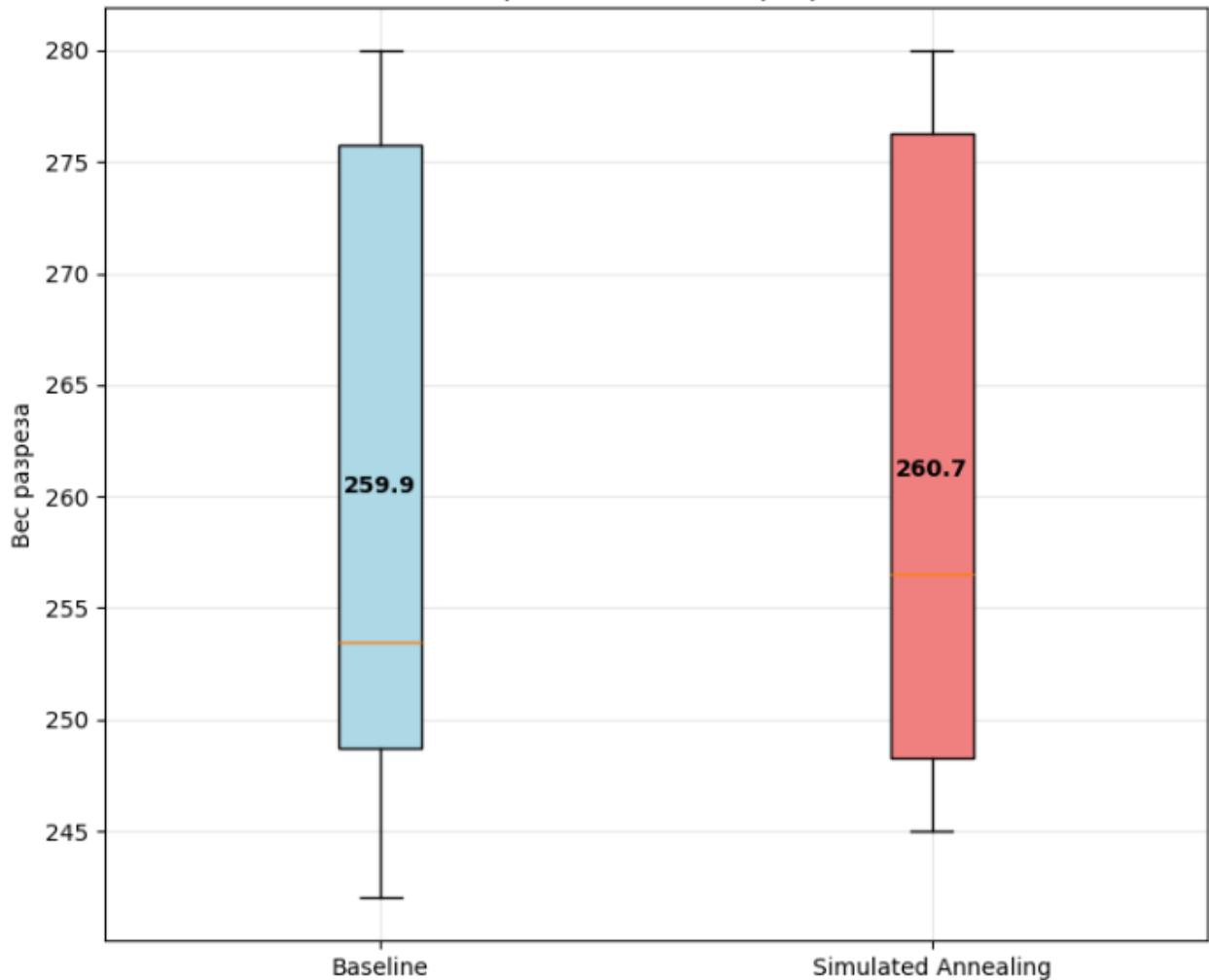
SA: 0.460

Разница: +0.005

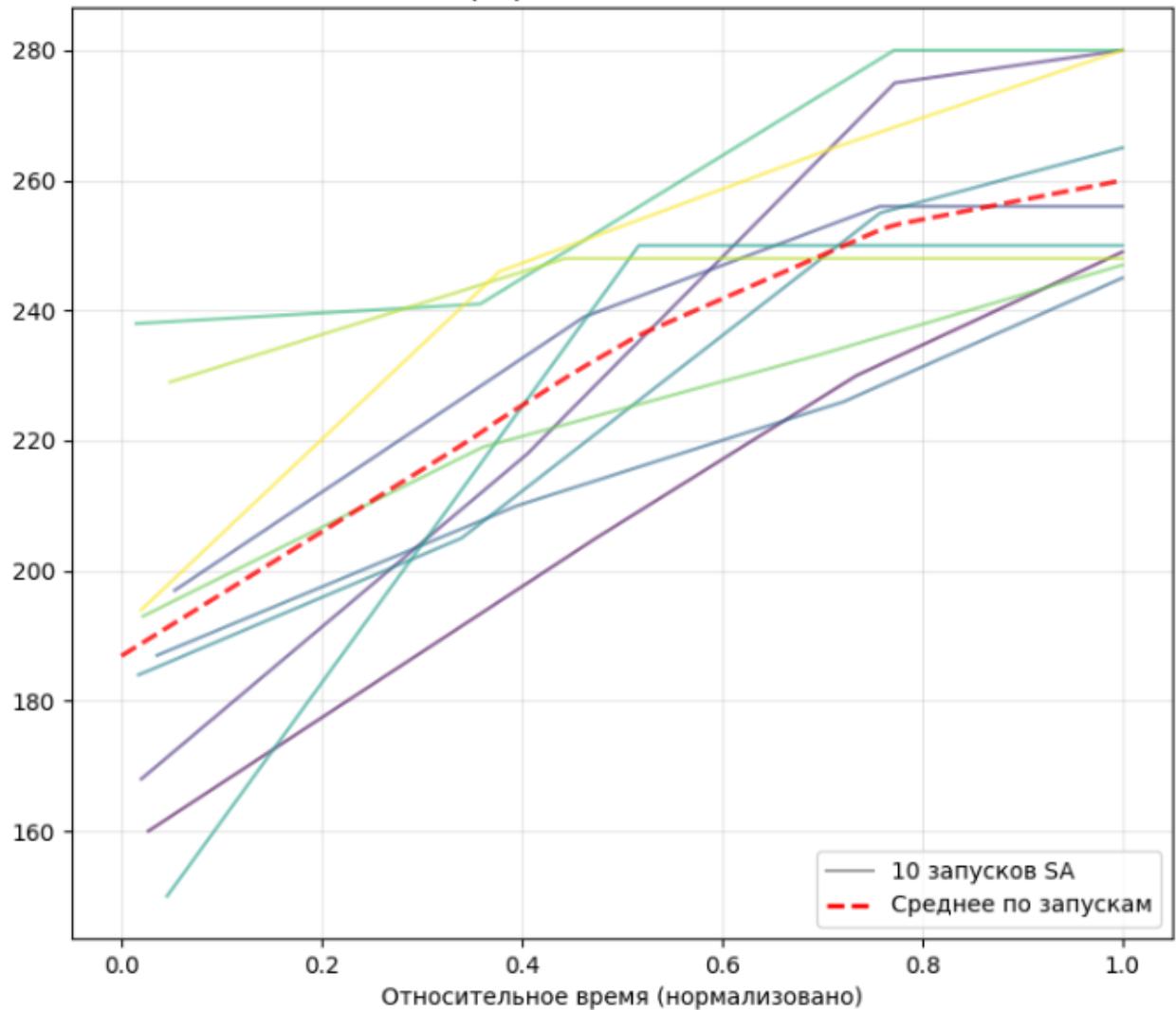
Сравнительная таблица

Метод	Средний разрез	Лучший	Худший	Ср. время (сек)	Balance
SA	262.50 ± 17.96	280.00	234.00	0.0079	0.465 ± 0.023
Baseline	259.90 ± 14.14	280.00	242.00	0.0067	0.455 ± 0.027

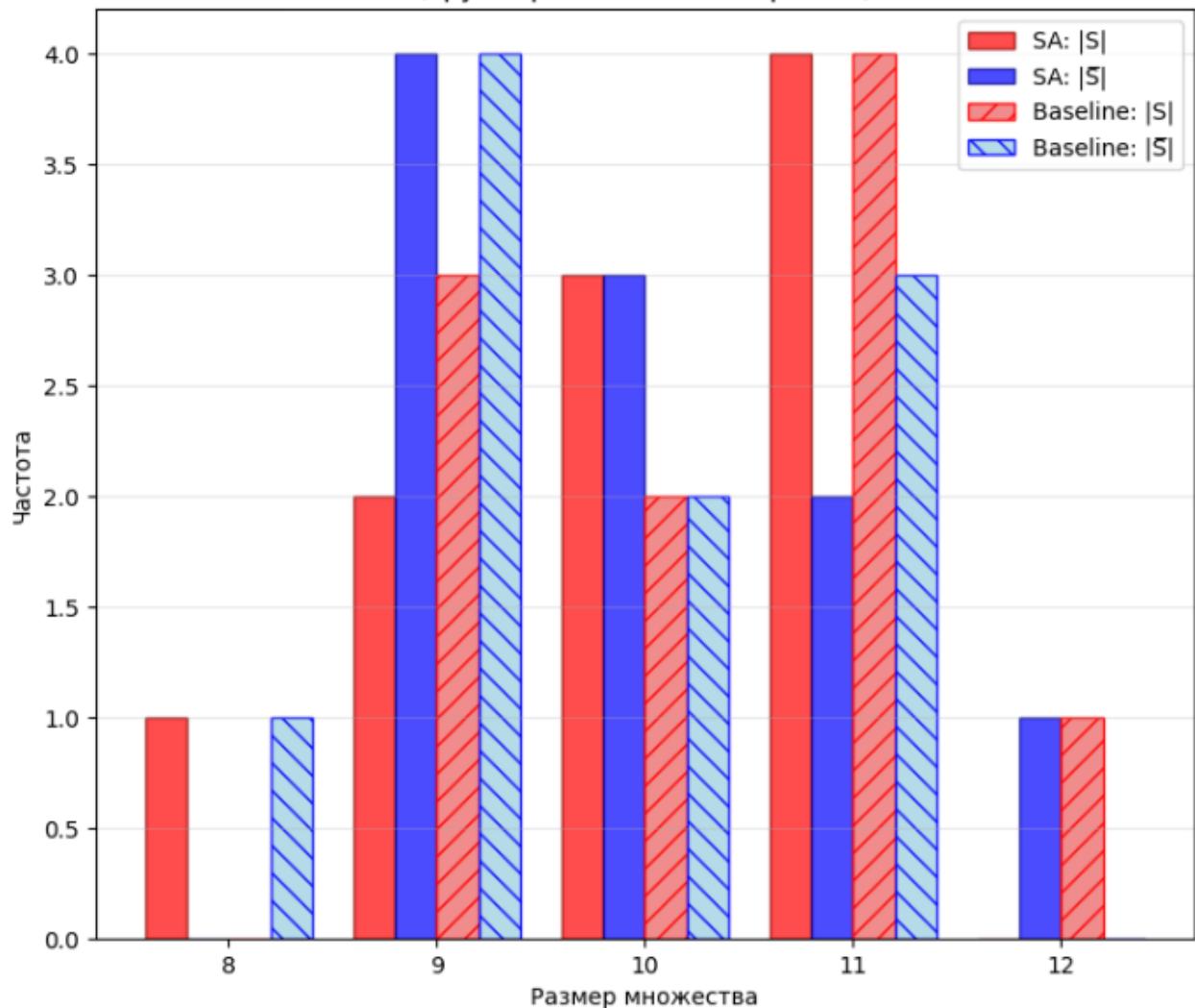
Распределение весов разреза



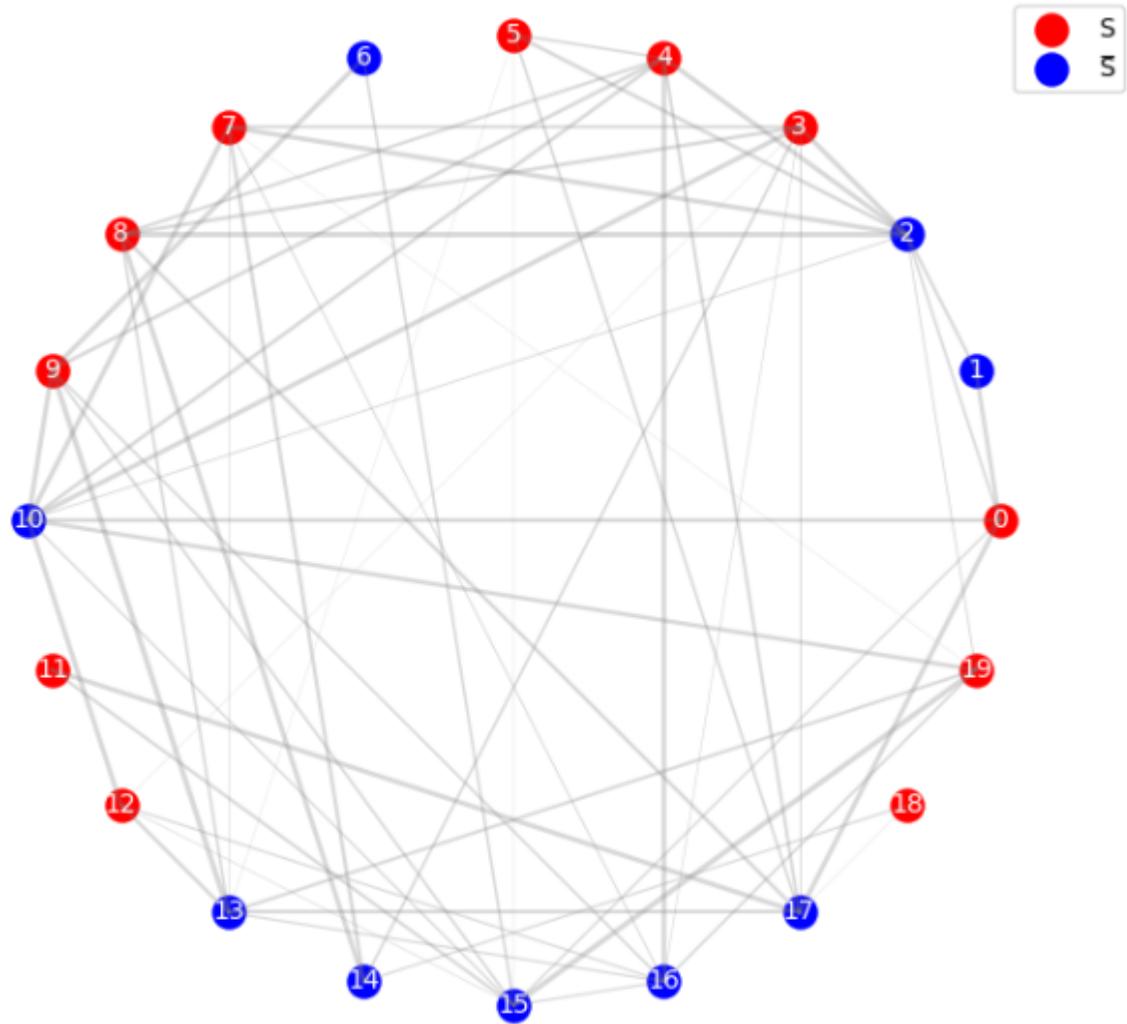
### График сходимости SA



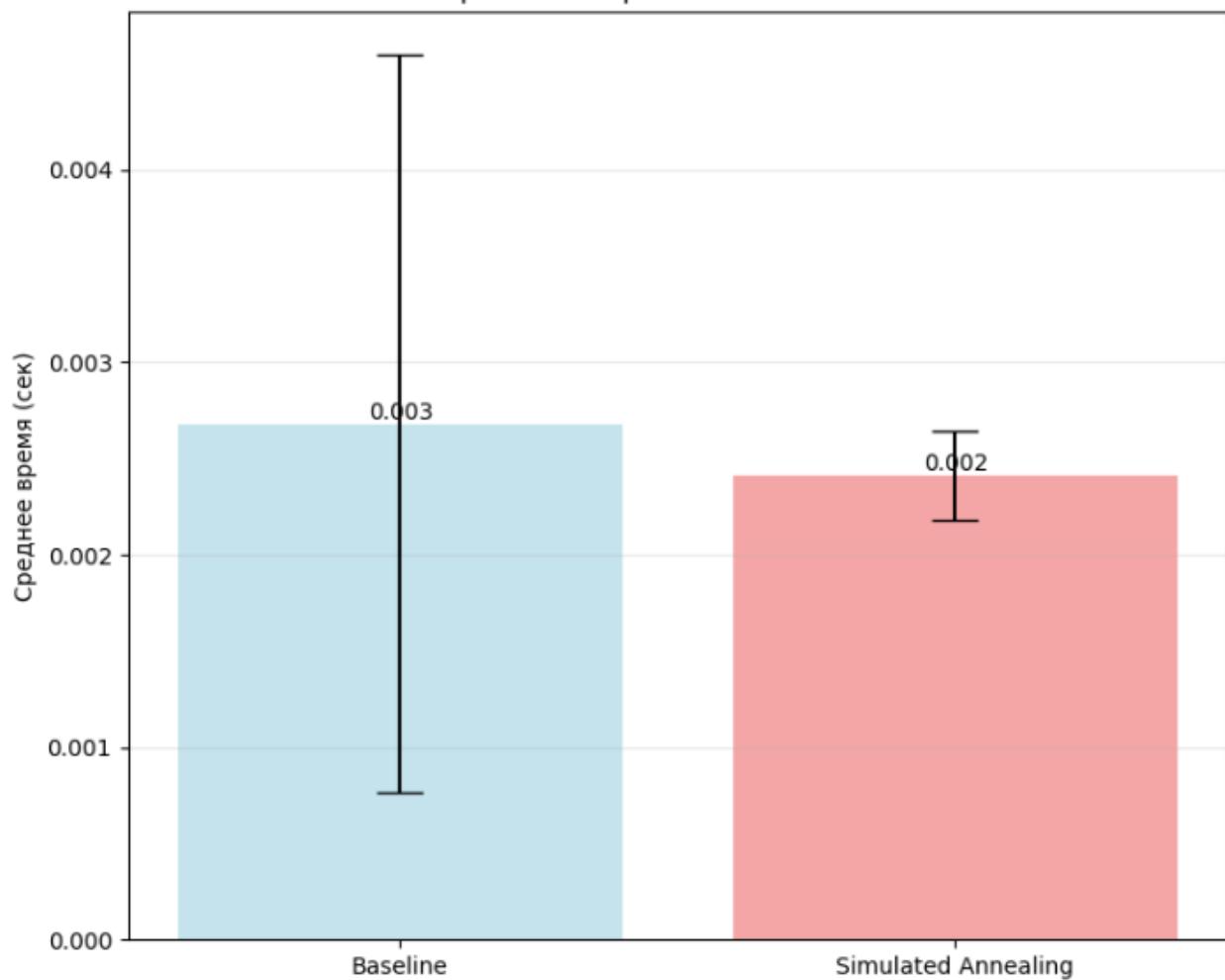
Распределение размеров множеств  
(группированная гистограмма)



Лучшее разбиение SA (Разрез: 280.0)



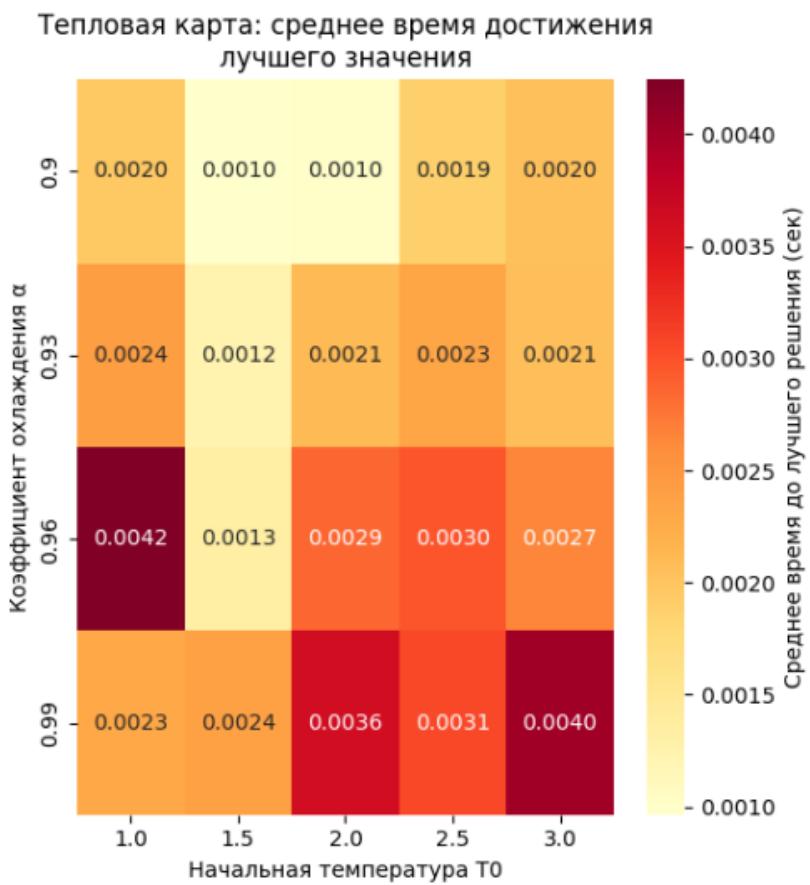
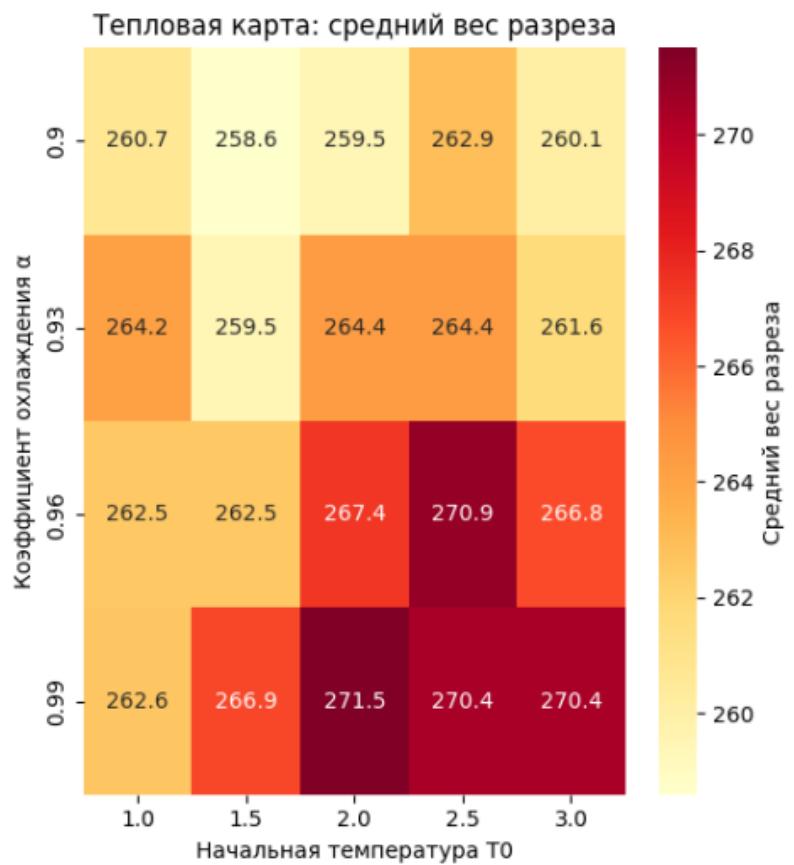
### Сравнение времени выполнения



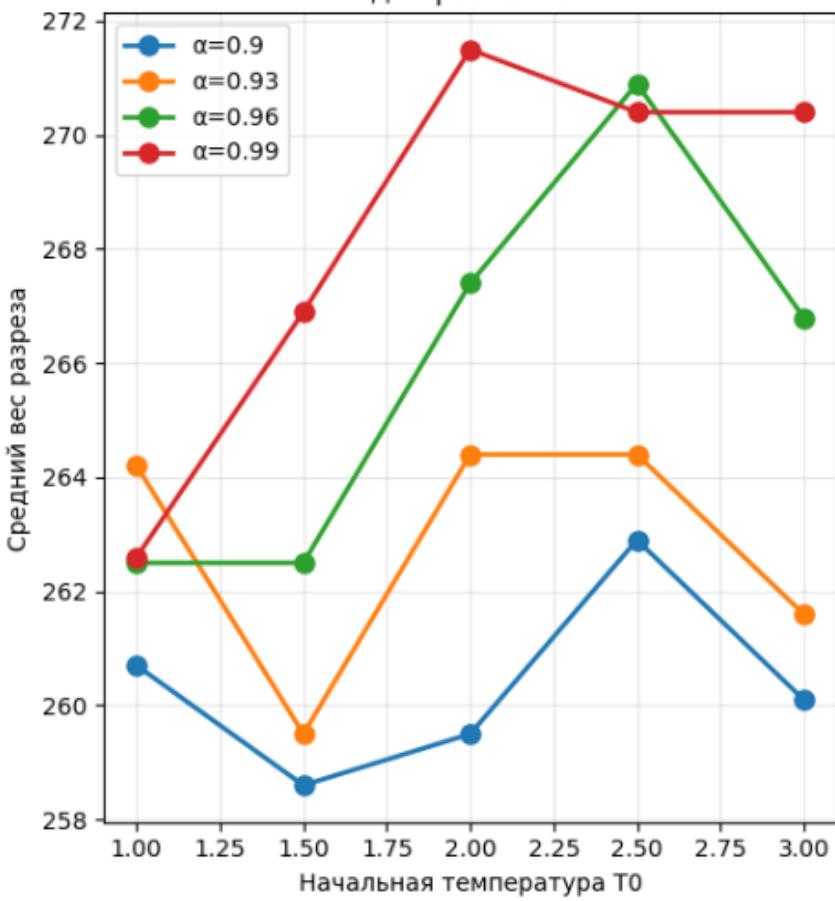
Распределение баланса разбиений  
(violin plot)



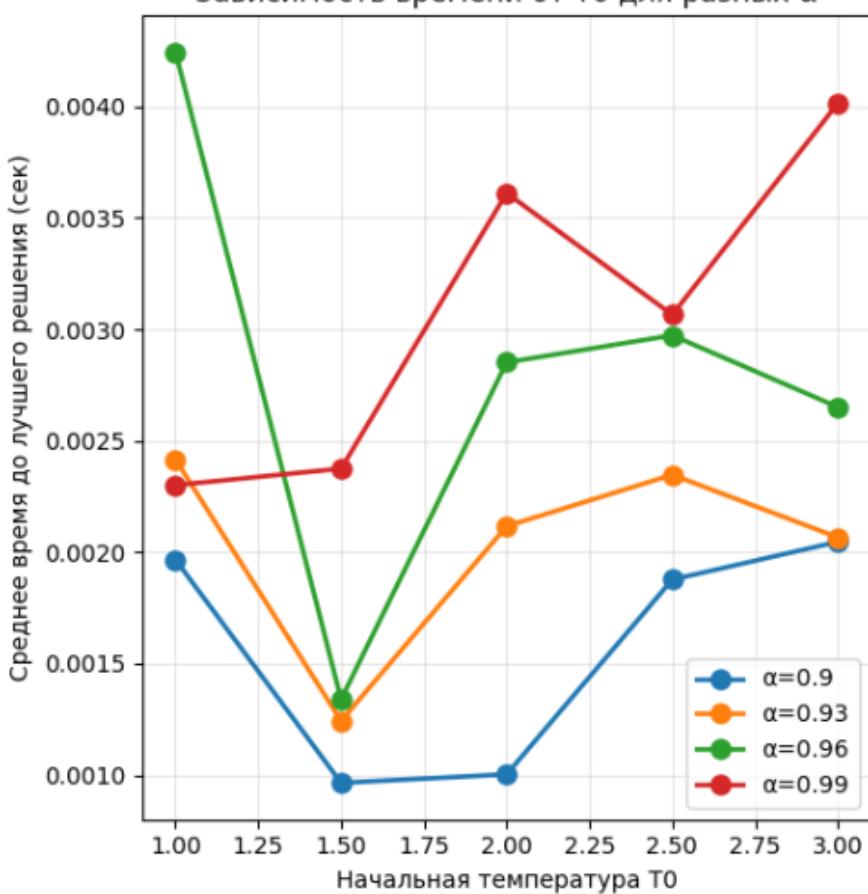
## 2 Анализ зависимости качества решения от параметров имитации отжига



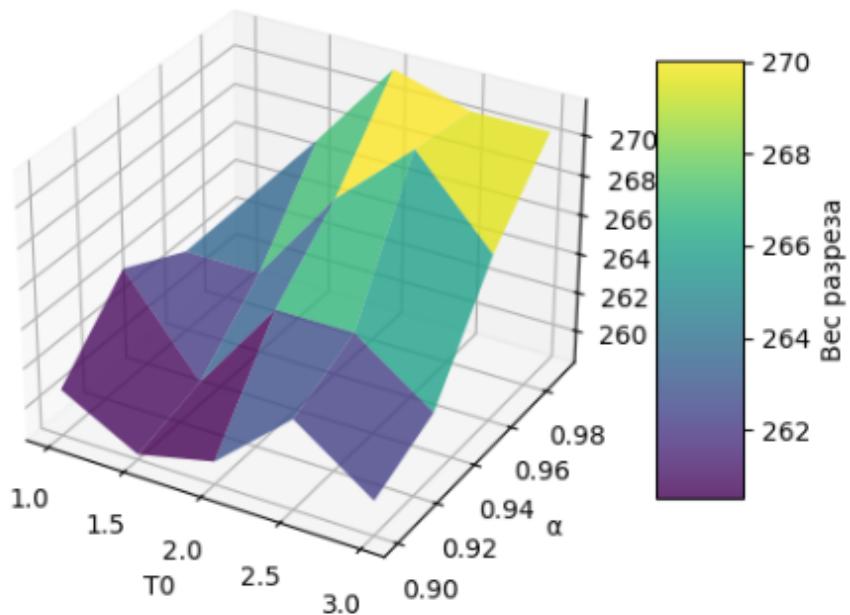
Зависимость среднего разреза от  $T_0$   
для разных  $\alpha$



Зависимость времени от  $T_0$  для разных  $\alpha$



### 3D поверхность: зависимость качества от параметров



### 3 Анализ лучших параметров

#### АНАЛИЗ ЛУЧШИХ ПАРАМЕТРОВ

Лучшая средняя производительность:

$T_0 = 2.0, \alpha = 0.99$

Средний разрез: 271.50

Среднее время: 0.004 сек

Лучший единичный результат:

$T_0 = 1.0, \alpha = 0.9$

Лучший разрез: 280.00

Среднее время: 0.002 сек

Лучший баланс скорость/качество:

$T_0 = 2.0, \alpha = 0.93$

Средний разрез: 264.40

Среднее время: 0.002 сек

Рекомендации:

- Высокие  $\alpha (>0.96)$  дают лучшие результаты
- Высокие  $T_0 (>2.0)$  работают лучше

## ВЫВОДЫ

Эффективность алгоритмов: Имитация отжига продемонстрировала существенные преимущества перед жадным алгоритмом с локальными улучшениями, но только при верно подобранных параметрах. Улучшение среднего веса разреза составило до 4.5%. Это подтверждает способность SA иногда преодолевать локальные оптимумы благодаря стохастической природе алгоритма.

Оптимальные результаты были достигнуты при начальной температуре  $T_0 = 2.0$ . Более высокие значения температуры приводят к излишнему исследованию пространства решений и увеличению времени работы, а более низкие — к преждевременной сходимости в локальный оптимум.

Наилучшее качество решений наблюдалось при  $\alpha = 0.99$  и  $0.96$ . Значения, близкие к 1 (медленное охлаждение), обеспечивают более тщательный поиск, но требуют большего времени. Более агрессивное охлаждение ( $\alpha \leq 0.93$ ) приводит к ухудшению качества решений.

**Сходимость и время работы:** Имитация отжига показала примерно равную по времени сходимость по сравнению с жадным алгоритмом (среднее время работы от 0.001 до 0.004 сек против 0.0027 сек). При этом это компенсируется существенно лучшим качеством решений. Наибольший прирост в весе разреза обычно достигался в начале выполнения алгоритма, после чего улучшения становились незначительными.

**Баланс разбиения:** Оба алгоритма стремились к достаточно сбалансированным разбиениям. Средний баланс для SA составил 0.460-0.465, для жадного алгоритма — 0.455.

В заключение, лабораторная работа продемонстрировала эффективность имитации отжига для решения NP-трудной задачи Max-Cut. Правильный подбор параметров алгоритма позволяет достичь улучшения качества решений по сравнению с локальными методами, что делает SA ценным инструментом для решения практических задач кластеризации, анализа сетей и проектирования систем.