

# Разработка итерационных алгоритмов поиска автоморфизмов и изоморфизмов комбинаторных объектов.

---

Автор:

Ефремов Степан Сергеевич (419 группа)

Научный руководитель:

доцент, к.ф.- м.н.

Егоров Владимир Николаевич

Московский государственный университет имени М.В.Ломоносова

Факультет вычислительной математики и кибернетики

Кафедра информационной безопасности

Постановка задачи

Существующие решения

Разработанные решения

Результаты

Положения, выносимые на защиту

# Постановка задачи

---

# Постановка задачи

1. Исследование свойств алгоритма:

# Постановка задачи

## 1. Исследование свойств алгоритма:

- Определение класса решаемых задач
- Вероятностная сложность
- Теоретическая возможность распараллеливания
- Модернизация алгоритма на основе исследований

# Постановка задачи

## 1. Исследование свойств алгоритма:

- Определение класса решаемых задач
- Вероятностная сложность
- Теоретическая возможность распараллеливания
- Модернизация алгоритма на основе исследований

## 2. Задачи, связанные с реализацией для ПК:

# Постановка задачи

## 1. Исследование свойств алгоритма:

- Определение класса решаемых задач
- Вероятностная сложность
- Теоретическая возможность распараллеливания
- Модернизация алгоритма на основе исследований

## 2. Задачи, связанные с реализацией для ПК:

- Реализация в виде программы с графическим интерфейсом
- Эксперименты поиска автоморфизмов на известных графах
- Поддержка функционала нахождения изоморфного вложения графов

# Постановка задачи

## 1. Исследование свойств алгоритма:

- Определение класса решаемых задач
- Вероятностная сложность
- Теоретическая возможность распараллеливания
- Модернизация алгоритма на основе исследований

## 2. Задачи, связанные с реализацией для ПК:

- Реализация в виде программы с графическим интерфейсом
- Эксперименты поиска автоморфизмов на известных графах
- Поддержка функционала нахождения изоморфного вложения графов

## 3. Задачи, связанные с реализацией для суперкомпьютера:



# Постановка задачи

## 1. Исследование свойств алгоритма:

- Определение класса решаемых задач
- Вероятностная сложность
- Теоретическая возможность распараллеливания
- Модернизация алгоритма на основе исследований

## 2. Задачи, связанные с реализацией для ПК:

- Реализация в виде программы с графическим интерфейсом
- Эксперименты поиска автоморфизмов на известных графах
- Поддержка функционала нахождения изоморфного вложения графов

## 3. Задачи, связанные с реализацией для суперкомпьютера:

- Исследование ресурса параллелизма
- Реализация в виде программы для запуска на суперкомпьютере
- Эксперименты поиска автоморфизмов графов на суперкомпьютере

# Постановка задачи

## 1. Исследование свойств алгоритма:

- Определение класса решаемых задач
- Вероятностная сложность
- Теоретическая возможность распараллеливания
- Модернизация алгоритма на основе исследований

## 2. Задачи, связанные с реализацией для ПК:

- Реализация в виде программы с графическим интерфейсом
- Эксперименты поиска автоморфизмов на известных графах
- Поддержка функционала нахождения изоморфного вложения графов

## 3. Задачи, связанные с реализацией для суперкомпьютера:

- Исследование ресурса параллелизма
- Реализация в виде программы для запуска на суперкомпьютере
- Эксперименты поиска автоморфизмов графов на суперкомпьютере

## 4. Исследование практического применения алгоритма для задачи Коши

## Существующие решения

---

На данный момент сформированы два направления изучения и решения проблемы поиска изоморфизмов графов:

- Теоретическое, в котором проблема изоморфизма рассматривается с позиций современной теории сложности алгоритмов и вычислений (подход использует понятие инвариантов графа).
- Практическое, предполагающее разработку алгоритмов, решающих задачу изоморфизма графов за «практически приемлемое» время (направленный перебор).

# Сравнение сложностей алгоритмов

Алгоритм	Ограничения	Сложность
Полный перебор	без ограничений	$n!$
Егоров В.Н., Егоров А.В.	без ограничений	$\approx O(n^2(\frac{\epsilon}{2})^{\ln(n)^2} \ln(n))$
László Babai, Eugene M.	без ограничений	$O(e^{\sqrt{n \times \log(n)}})$
Daniel A. Spielman	сильно регулярные	$n^{O(n^{1/3} \log^2 n)}$
Vaibhav Amit Patel	специальный вид	$O(n^4)$

**Таблица 1:** Алгоритмы поиска автоморфизмов

# Разработанные решения

---

# Модернизированный алгоритм

Разработан алгоритм с вероятностной сложностью:

$$O(n^2(\frac{e}{2})^{\ln(n)^2} \ln(n))$$

Алгоритм является универсальным и может применяться для решения большого количества задач (как минимум всех указанных в пункте 4.1 дипломной работы).

Написанная программа с интерфейсом имеет масштабируемые функционал. На данный момент поддерживает функционал решения следующих задач:

- Поиск автоморфизмов
- Поиск изоморфизмов
- Поиск гомоморфизмов
- Решение задачи Коша в подстановках

Разработано 2 программы:

1. Приложение с графическим интерфейсом, написанное на *Qt*
  - не поддерживает распараллеливание
  - справляется с графами размера до 500 вершин
  - удобный анализ графов
  - помимо нахождения автоморфизмов, решает задачи изоморфизма и гомоморфизма графов
2. Программа на языке *C++* с поддержкой *OpenMPI*
  - поддерживает распараллеливание средствами *OpenMPI*
  - помимо нахождения автоморфизмов, сохраняет много информации для анализа графов с большим количеством вершин



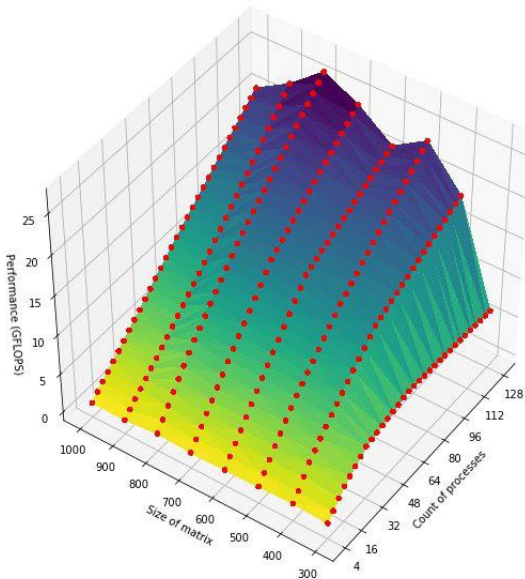
Набор и границы значений изменяемых параметров запуска реализации алгоритма:

1. Число процессоров  $[4 : 128]$  с шагом  $2^n$  (точки отображены с шагом 4, усреднив результаты);
2. Размер матрицы  $[300 : 1000]$ .

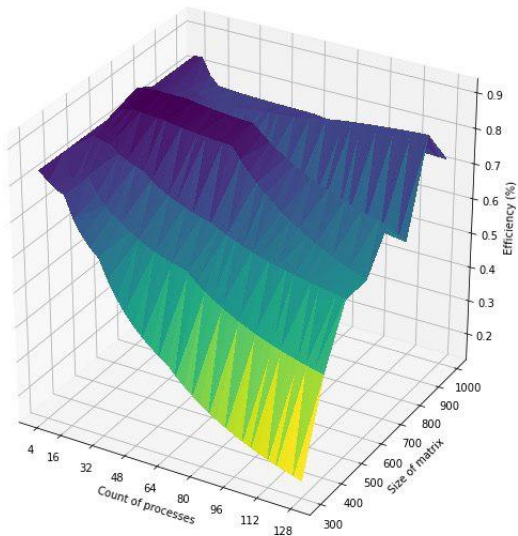
# Результаты

---

# Изменение производительности



# Изменение эффективности



## Положения, выносимые на защиту

---

Разработка итерационного алгоритма поиска автоморфизмов графов:

- Выполнена модернизация алгоритма
- Сформулирована гипотеза вероятностной сложности алгоритма
- Реализовано 2 программы:
  - с графическим интерфейсом для удобного использования
  - с консольным интерфейсом для запуска на суперкомпьютере
- Проведены опыты на суперкомпьютере «Ломоносов»