Защита

1 слайд

Добрый день, члены комиссии, меня зовут Параничева Алена, хочу представить вашему вниманию дипломную работу на тему

**Сравнение нескольких алгоритмов решения задач о независимом множестве и вершинной раскраске для связных и хордальных графов**

(Введение и актуальность)

В настоящее время графы применяются почти во всех научных отраслях, так как с помощью них можно представить и описать различные объекты и структуры. Для решения задач на графах используются различные алгоритмы, в том числе и те, у которых нет общего оптимального решения.

2 слайд

(Цели и задачи)

Цель работы – это реализовать алгоритмы поиска наибольшего независимого множества и вершинной раскраски для связных графов и адаптированные алгоритмы для хордальных графов, а также сравнить результаты испытаний.

Задачи работы:

* Изучить теоретический материал, необходимый для исследований
* Разработать класс для работы с графами
* Разработать алгоритмы генерирования и распознавания хордальных графов
* Реализовать вышеупомянутые алгоритмы поиска
* Провести испытания и сравнить результаты

3 слайд

Какие графы считаются хордальными?

Хордальный граф - это связный граф без порождённых циклов длины более чем три

(то есть любой цикл длины 4 и больше содержит хорду).

В хордальном графе всегда найдется симплициальная вершина. Это такая вершина окрестность которой порождает клику.

4 слайд

Данная теория позволяет распознавать этот вид графов среди прочих.

Также для проведения испытаний была разработана функция генерирования хордальных графов, которая базируется на алгоритме поиска в глубину.

(Чтобы провести испытания, кроме функции распознавания нам нужна функция генерирования хордальных графов.

Для этого генерируем случайный связный граф и при помощи алгоритма поиска в глубину достраиваем недостающие ребра для обеспечения хордальности графа.

Испытания по данным алгоритмам показывают, что время их реализации в большей степени зависит от количества вершин в графе, так как мы обходим его целиком.)

5 слайд

Перейдем к задачам по поиску

Наибольшее независимое множество графа – это наибольшее множество вершин графа, в котором любые две вершины не смежны.

Пример выполнения общего алгоритма представлен на слайде.

Чтобы найти такое множество дублируем исходный граф и в одной копии удаляем выбранную вершину, а в другой - удаляем окрестность этой вершины. Такие преобразования происходят до пустоты графов, где наибольший из них и есть искомое независимое множество.

6 слайд

Для того чтобы найти наибольшее независимое множество в хордальном графе воспользуемся наличием в нем смежно поглощающей вершины.

Вершина a смежно поглощает вершину b, если вершины a и b смежные и окрестность b полностью включена в окрестность а (N(b)\{a} ⊆ N(a)\{b}).

7 слайд

При удалении такой вершины число независимости в графе не меняется.

Также для нахождения наибольшего независимого множества можно использовать те же симплициальные вершины. Так как их окрестности не могут содержаться в независимом множестве.

8 слайд

Проведенные испытания показывают, что время работы алгоритма со смежным поглощением не зависит от заполненности графа, тогда как при уменьшении количества ребер алгоритм с симплициальными вершинами замедляется.

9 слайд

Среднее время работы общего алгоритма посчитать сложно, так как при равных условиях разница во времени могла быть колоссальной. Здесь представлены его наименьшие результаты для наглядности преимущества хордальных алгоритмов.

Следующий график показывает, что при уменьшении количества рёбер и при увеличении числа вершин в графе его число независимости растет. В среднем число независимости у хордальных графов немного выше из-за меньшего разнообразия.

10 слайд

Рассмотрим следующий алгоритм поиска

Правильная вершинная раскраска – это раскраска вершин графа, при которой любые смежные вершины окрашены в разные цвета.

Пример выполнения общего алгоритма показан на слайде.

Чтобы найти раскраску по общему алгоритму дублируется исходный граф и в одной из копий проводится ребро между двумя несмежными вершинами, а в другой две несмежные вершины объединяются. Графы изменяются до их полноты, где число вершин в наименьшем из всех – это хроматическое число графа.

11 слайд

Чтобы найти вершинную раскраску в хордальном графе, последовательно удаляются все симплициальные вершины. Далее в обратном удалению порядке вершинам раздаются цвета.

Хроматическое число в хордальном графе совпадает с размером наибольшей клики.

12 слайд

Перейдем к испытаниям

На основе проведенных тестов, можно увидеть, насколько время выполнения общей раскраски зависит от заполненности графа. Лишь подняв заполнение графа ребрами до 80% удалось провести испытания на графах с большим числом вершин без остановки программы из-за нехватки мощности вычислительной машины. Второй график демонстрирует преимущество по времени хордального алгоритма.

13 слайд

На следующих графиках показаны средние значения хроматического числа в хордальных и произвольных графах. Хроматическое число везде растет равномерно увеличению количества вершин и заполненности графа. Хроматическое число в хордальном графе обычно выше, чем у произвольного, так как при его создании образуются большие клики.

14 слайд

(Итоги и выводы)

В ходе работы были выполнены все поставленные задачи.

А по результатам исследований можно сделать главный вывод о том, что при решении задач правильная типизация графа - это одно из лучших решений, для усовершенствования неоптимальных алгоритмов поиска.