

AAAAAA

Реферат

Рассмотрим следующую динамическую систему, изучение которой мотивировано проблемой изучения поведения волновых пакетов, локализованных в малой окрестности единственной точки в начальный момент и развивающихся на метрических графах или гибридных пространствах. В начальный момент времени из истока s появляются точки вдоль всех ребер, инцидентных s , и движутся с единичной скоростью. Когда k точек, где k может принимать значения от 1 до степени v вершины v_j , находятся в вершине графа, появляется v новых точек, идущих по всем ребрам, инцидентным вершине v_j . Пусть $N(T)$ – количество точек, которые перемещаются по графику в момент времени T . Функция $N(T)$ кусочно-постоянная.

Целью работы является поиск общей формулы для $N(T)$ в случае ориентированного сильно связного графа. Мы находим полиномиальное приближение к $N(T)$ и явно выписываем первые два члена этого приближения.

Работа проводится путем выдвижения и доказательств различных гипотез и симуляции поведения точек на графе с помощью программы.

В ходе эксперимента был получен граф, формула старшего коэффициента для которого не соответствует формуле их гипотезы, а так же была написана программа-графопостроитель, которая дает возможность наглядно продемонстрировать движение точек на графе и график их количества, зависящего от времени.

Содержание

Реферат.....	2
Основные термины и определения.....	4
Введение.....	4
Задачи.....	5
Обзор и сравнительный анализ источников	5
Описание выбранных алгоритмов.....	5
Структура программы и разработанных алгоритмов.....	6
Проведение эксперимента.....	9
Заключение.....	10
Календарный план.....	11
Приложение 1	12
Литература	13

Основные термины и определения

Нам дан граф $G = (V, E)$, в котором V – множество вершин, а E – множество ребер. Предположим, что ребра e_i имеют длины t_i , и эти значения линейно независимы над \mathbb{Q} .

Пусть s – это зафиксированная вершина $s \in V(G)$, которую мы назовем **исток**ом.

Компонентой сильной связности называется такое максимальное по включению подмножество вершин C , что любые две вершины этого подмножества достижимы друг из друга, т.е. $\forall u, v \in C$:

$$u \mapsto v, v \mapsto u.$$

Отсюда логично определение **конденсации** G как графа, полученного из исходного графа сжатием каждой компоненты сильной связности в одну вершину. Каждой вершине графа конденсации соответствует компонента сильной связности графа G , а ориентированное ребро между двумя вершинами C_i и C_j графа конденсации проводится, если найдётся пара вершин $u \in C_i, v \in C_j$, между которыми существовало ребро в исходном графе, т.е. $(u, v) \in E$

Введение

Рассмотрим следующую динамическую систему, изучение которой мотивировано проблемой изучения поведения волновых пакетов, локализованных в малой окрестности единственной точки в начальный момент и развивающихся на метрических графах или гибридных пространствах. В начальный момент времени из истока s появляются точки вдоль всех ребер, инцидентных s , и движутся с единичной скоростью. Когда k точек, где k может принимать значения от 1 до степени v вершины v_j , находятся в вершине графа, появляется v новых точек, идущих по всем ребрам, инцидентным вершине v_j . Пусть $N(T)$ – количество точек, которые перемещаются по графику в момент времени T . Функция $N(T)$ кусочно-постоянная. Распространение точек соответствует динамике центров узких гауссовых волновых пакетов, если рассматривать эволюцию квазиклассического решения задачи Коши для нестационарного уравнения Шредингера на метрическом графе.

Задачи

В этом проекте мы планируем найти общую формулу для $N(T)$ и в случае ориентированного графа. Мы находим полиномиальное приближение к $N(T)$ и явно выписываем первые два члена этого приближения.

В нашем проекте целью будет доказательство гипотез о приближении значения формулы $N(T)$, написание программы, чей функционал будет позволять ввести небольшой граф, посмотреть графики зависимости количества точек от времени и сравнить с графиками значений предположительной формулы.

Обзор и сравнительный анализ источников

Главная часть асимптотики $N(T)$ при $T \rightarrow \infty$ для произвольного конечного неориентированного метрического графа с несоизмеримыми длинами ребер были найдены ранее. (см. [1])

В случае линейно независимых над \mathbb{Q} длин ребер задача связана с проблемой подсчета количества узлов решетки в расширяющейся симплексах с действительными вершинами. Асимптотическое разложение для $N(T)$ было найдено с использованием множественных полиномов Бернулли. Явные формулы для первых двух членов разложения для считающей функции количества движущихся точек были представлены (см. [2]). Главный член был найден ранее (см. [5]) и зависит только от числа вершин V , числа ребер E , суммы и произведения длин ребер t_j .

$$N(T) = \frac{1}{2^{V-2}(E-1)!} \frac{\sum_{j=1}^E t_j}{\prod_{j=1}^E t_j} T^{E-1} (1 + o(1)).$$

Описание выбранных алгоритмов

Алгоритм конденсации графа состоит в следующем:

- 1) Строим граф T с обратными (инвертированными) рёбрами.
- 2) Выполняем в T поиск в глубину и находим $\mathbf{tout}[u]$ — время окончания обработки вершины u .
- 3) Выполняем поиск в глубину в G , перебирая вершины во внешнем цикле в порядке убывания $\mathbf{tout}[u]$.

Полученные на 3-ем этапе деревья поиска в глубину будут являться компонентами сильной связности графа G .

Структура программы и разработанных алгоритмов

Была написана программа на языке C++ с использованием библиотеки SFML. Данная программа может строить ориентированный граф по заданным в текстовом поле данным, либо графически посредством нажатия левой кнопкой мыши добавлять новые вершины и соединять их ребрами. Программа располагает вершины определенным образом, описанным ниже, но также имеется возможность двигать вершины, если хочется расположить их иным образом.

Для построения графа по введенным данным программа использует алгоритм конденсации для разделения графа на компоненты сильной связности, а дальше строит эти компоненты по уровням. Для этого она считает для каждой компоненты ее глубину.

После вычисления глубины **depth** итогового дерева компонент сильной связности и максимального количество вершин **maxcnt** на одной глубине, находим высоту h и ширину w пространства для одной компоненты по формулам:

$$w = \frac{W}{\text{maxcnt}},$$
$$h = \frac{H}{\text{depth}}.$$

Где W , H – ширина и высота квадрата, предназначенного для рисования всего графа.

Чтобы построить компоненту сильной связности, мы вписываем в область размера $w \times h$ правильный n -угольник, где n – количество вершин в данной компоненте сильной связности. Для построения правильного многоугольника находим центр квадрата, откладываем от него вектор до стороны квадрата, отмечая первую точку, а затем поворачиваем этот вектор на $\frac{360}{n}$ градусов $n - 1$ раз.

Также если в компоненте сильной связности присутствует Гамильтонов цикл, компонента сильной связности строится так, чтобы они в многоугольнике шли по порядку. Чтобы проверить существование Гамильтонова цикла(и найти его), используется динамическое программирование **dp[mask][v]** (в котором лежит 1, если существует такой путь, и 0, если не существует), где **mask** – множество посещенных по одному разу вершин, **v** – последняя посещенная вершина. Пересчет ведется представленным ниже способом(см. Рис. 1):

```

vector<vector<int>> > dp((1 << n), vector<int>(n));
vector<vector<int>> > prev((1 << n), vector<int>(n, -1));

dp[1 << 0][0] = 1;

for (int mask = 0; mask < (1 << n); ++mask) {
    for (int v = 0; v < n; ++v) {
        if (!dp[mask][v]) continue;
        for (auto u : graph[comp[v]]) {
            if (num.find(u) != num.end() &&
                (mask | (1 << num[u])) != mask) {
                dp[mask | (1 << num[u])][num[u]] = 1;
                prev[mask | (1 << num[u])][num[u]] = v;
            }
        }
    }
}

```

Рис. 1: Пересчет динамики

После того, как нашли координаты всех вершин, рисуем сначала ребра, а потом сверху вершины.

Если ребро задается графически, после соединения двух вершин нужно ввести квадрат веса ребра и нажать **Enter**.

Нажав на **GO!**, мы запустим процесс моделирования движения точек на графе. При нажатии на **view plots** можно посмотреть на графики формулы $N(t)$.

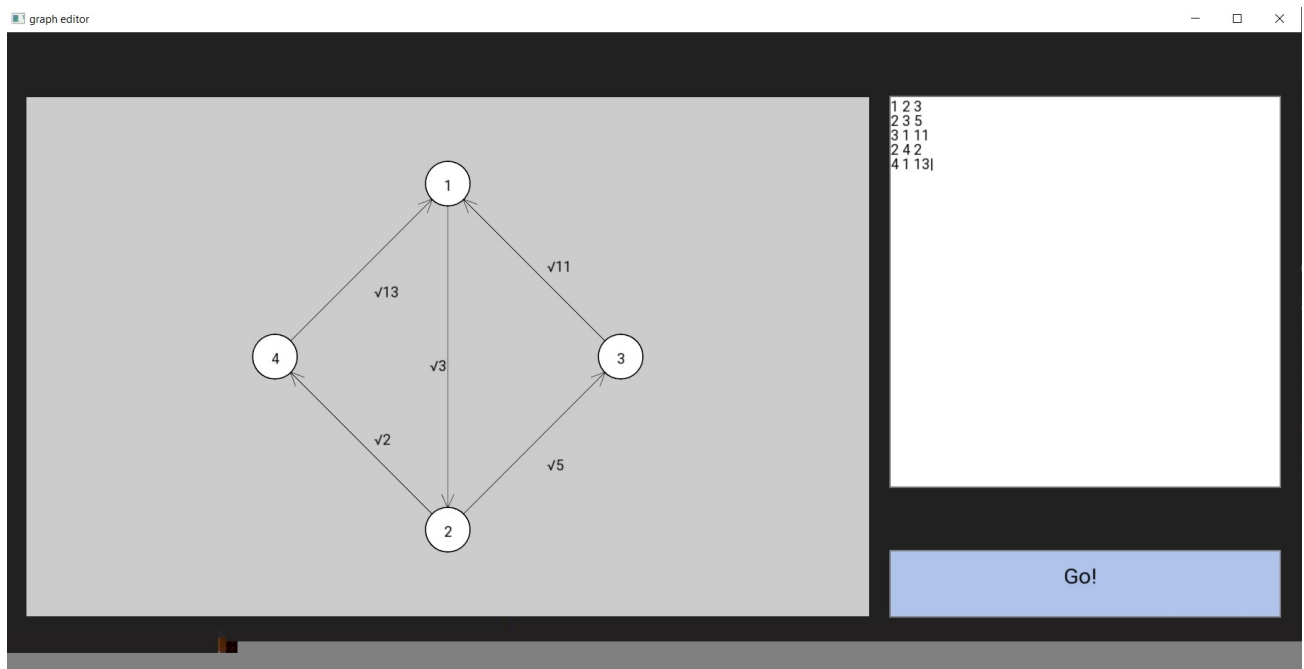


Рис. 2: Визуализатор графа

Описание классов, функций и заголовочных файлов

- **condense.h** – заголовочный файл, в котором описана функция **condense()**, принимающую на вход граф и возвращающую вектор компонент, состоящих из векторов входящих в них вершин.
- **form.h** – заголовочный файл, в котором описаны класс **Label** и его подклассы **Bottom** и **input**.
 - **Label** содержит вектор строк **text**, ширину и высоту соответственно **width** и **height**, координаты **x** и **y**, размер текста **sz** и отображаемый текст **TXT**.
 - **Button** дополнительно содержит счетчик **cnt**, прямоугольник **BOX**, и цвет в обычном, наведенном и нажатом состоянии **clr**, **clr_selected** и **clr_pressed** соответственно.
 - **Input** содержит те же поля, что и **Label**.
- **graph.h** – заголовочный файл, в котором описана функция **build_graph()**, которая по введенному графу рассчитывает позиции его вершин, **draw_graph()**, которая уже отрисовывает вершины и ребра, и **draw_all_points()**, которая рисует движущиеся точки на графе в режиме симуляции.
- **hamilthon.h** – заголовочный файл, в котором описана функция **sort_by_hamilthonian_cycle()**, которая сортирует переданный в функцию граф в порядке гамильтонова цикла, если тот в графе присутствует. В ином случае ничего не делает.
- **plot.h** – заголовочный файл, в котором описана функция **draw_plot**, которая позволяет строить нужные графики.
- **points_on_edges.h** – заголовочный файл, который ответственен за нахождение положения точек на ребрах в зависимости от времени.
- **topsort.h** – заголовочный файл, в котором описана функция **topsort**, которая сортирует компоненты сильной связности в нужном порядке.

Проведение эксперимента

Задачей эксперимента являлось проверка формулы для сильно связных ориентированных графов, которая, как было известно на момент эксперимента, выполнялась для графов определенного типа:

$$N(T) = \frac{T^{\beta-1}}{(\beta-1)!} \cdot \frac{\sum_{e \in E} t(e)}{\prod_{i=1}^{\beta} t(c_i)} (1 + o(T^{\beta-1})).$$

Программа эксперимента генерировала ориентированные сильно связные графы и проверяла, выполняется ли на них формула.

По результатам эксперимента выяснилось, что формула точна не для всех видов графов, и нашлись примеры, на которых она не работает (см. Рис. 3):

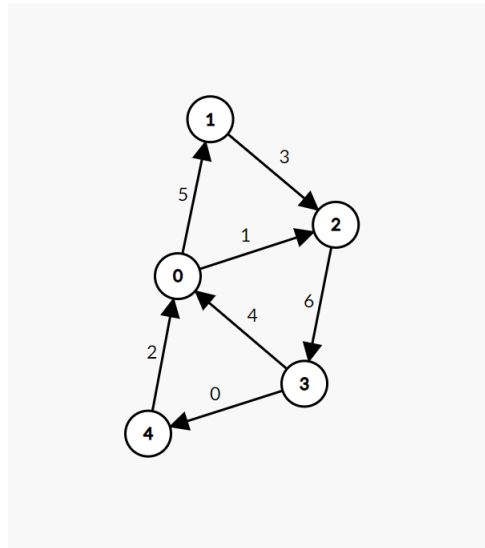


Рис. 3: Граф-контрпример

$\frac{N(t)}{t^2}$ для этого графа сходится к 0.0134, в то время, как предсказанный по формуле $\frac{1}{2} \frac{\sum_{e \in E} w(e)(c_0 + c_3)}{c_0 c_1 c_2 c_3}$ коэффициент равен 0.0077, где

Заключение

В ходе эксперимента был получен граф, формула старшего коэффициента для которого не соответствует формуле из гипотезы.

Была написана программа-графопостроитель, которая дает возможность наглядно продемонстрировать движение точек на графе и график их количества, зависящего от времени.

Всё ещё остаётся большое количество постановок задач, которые практически не исследованы, поэтому в перспективе с нашим объемом знаний и набором инструментов реально дальше исследовать эту тему и получить выдающиеся результаты.

Календарный план

1 февраля: написали программу, которая строит график зависимости кол-ва точек на графе от времени.

8 февраля: написали отчет для КТ-1.

22 февраля: правили отчет для КТ-1.

1 апреля: эксперимент по проверке точности формулы.

16 апреля: написание отчета для КТ-2.

29 апреля: правка отчета для КТ-2.

20 мая: Обобщение графа-контрпримера на большее семейство графов, заканчивание написания программы.

23 мая: Оформление отчета к КТ-3 и тестирование программы.

29 мая: Правка отчета для КТ-3.

3 июня: Защита проекта.

Приложение 1

Ссылка на репозиторий с кодом программы:

<https://github.com/E-Maxon/graphs/tree/main/graphs>

Литература

- [1] V. Chernyshev, A. Tolchennikov, Polynomial approximation for the number of all possible endpoints of a random walk on a metric graph, *Electronic Notes in Discrete Mathematics*, Elsevier, Volume 70, Pages 31–35. 2018.
- [2] Chernyshev V. L., Tolchennikov A. A., The Second Term in the Asymptotics for the Number of Points Moving Along a Metric Graph, *Regular and Chaotic Dynamics*, 2017, vol. 22, no. 8, pp. 937–948.
- [3] Chernyshev, V.L., Tolchennikov, A.A. Correction to the leading term of asymptotics in the problem of counting the number of points moving on a metric tree. *Russian Journal of Mathematical Physics*. Volume 24, Issue 3. 2017. pp 290–298.
- [4] Chernyshev V. L., Tolchennikov A. A., Shafarevich A. I., Behavior of Quasi-particles on Hybrid Spaces. Relations to the Geometry of Geodesics and to the Problems of Analytic Number Theory, *Regular and Chaotic Dynamics*, 2016, vol. 21, no. 5, pp. 531-537.
- [5] Chernyshev V. L., Shafarevich A. I., Statistics of gaussian packets on metric and decorated graphs, *Philosophical transactions of the Royal Society A.*, Volume: 372, Issue: 2007, (2014). Article number: 20130145.