Поиск наименьшего разрезающего циклы набора рёбер

Шевцова Маргарита

22 ноября 2023 г.

1 Формулировка задачи

Определение 1. Разрезающим циклы набором рёбер (feedback arc set) в ориентированном графе $G = (V, E)$ называется множество $F \subset E$, такое что любой ориентированный цикл в G содержит хотя бы одно ребро из F.

Задание (а) Докажите, что задача поиска разрезающего циклы набора рёбер NP-трудная;

(б) Имплементируйте какой-нибудь полиномиальный алгоритм приближённого решения этой задачи со степенью приближения, равной $O(n^{\varepsilon})$ для любого $\varepsilon > 0$, где n — число вершин в графе.

2 Доказательсво NP-трудности

Покажем, что задача NP-трудная. Определим множество $FEEDBACKARCSET = \{(G, k) \mid$ в ориентированном графе G есть разрезающий набор ребер с не более чем k листьями $\}$. Построим полиномиальную машину Тьюринга M, такую что $x \in FEEDBACKARCSET \leftrightarrow \exists s : M(x, s) = 1$.

А именно M воспринимает s как подмножетво ребер $F \subset E$. Она проверяет, что $|E| \leq k$ и в графе $(V, E \setminus F)$ нет ориентированных циклов. Проверить последнее можно с помощью dfs, а он работает за $O(|V| + |E|)$. То есть проверка за полиномиальное время. Если все проверки пройдены, M возвращает 1, иначе 0. Если $x \in FEEDBACKARCSET$, то такой сертификат найдется, иначе не найдется. Таким образом, $FEEDBACKARCSET \in NP$.

3 Небольшой обзор литературы о задаче апроксимации

Википедия и cs.stackexchange утверждают, что задача feedback arc set принадлежит классу APX, с коэффициентом аппроксимации $O(\log n \log \log n)$, где n – количество вершин в графе.

Важно отметить, что задача feedback arc set является родственной задаче feedback vertex set, где вместо ребер удаляются вершины (она тоже является NP-трудной и принадлежит классу APX).

В статье 3 авторы предлагают аппроксимационным алгоритмом с постоянным коэффициентом для задачи directed feedback vertex set problem со сложностью $4k \cdot k! \cdot n^{O(1)}$.

В литературе есть не мало алгоритмов аппроксимации задачи feedback arc set для турниров (полных ориентированных графов) и других типов графов. Например, в статье 4 авторы предлагают алгоритм $(1+\varepsilon)$-аппроксимации задачи feedback arc set для турнира за время $O(poly(n)2^{poly(1/\varepsilon)})$.

Скорее всего в этой работе будет тоже реализован алгоритм именно для турнира.

4 Описание алгоритма 2

Определение 2. Ориентированный граф (орграф) называется сильно связным, если любые две его вершины s и t сильно связны, то есть если существует ориентированный путь из s в t и одновременно ориентированный путь из t в s.

Определение 3. Компонентами сильной связности ориентированного графа называются его максимальные по включению сильно связные подграфы.

Из максимальности по включению следует, что компонентами сильной связности не пересекаются.

Из определения следует, что для любого цикла в ориентированном графе все вершины этого цикла лежат в одной компоненте сильной связности.

Сформулируем хорошо известное утверждение о структуре ориентированных графов.

Лемма 1. *Любой ориентированный граф разбивается на компоненты сильной связности, при чем эти компоненты можно пронумеровать так, что ребра между компонентами идут только от меньших по номеру к большим по номеру компонентам сильной связности.*

Значит, удаление ребер между компонентами сильной связности не изменит количество ориентированных циклов в графе. То есть все ребра минимального разрезающего набора ребер содержатся внутри компонент сильной связности. Теперь задачу можно решать для компонент сильной связности.

Определение 4. Компонентой 2-сильносвязности в ориентированном графе называются такой максимальный по включению сильно связный подграф, что при удалении любой вершины он остается сильно связным.

5 Отчет о тестовых запусках

Отчет о тестовых запусках должен включать в себя:

5.1 Описание наборов тестов

Делать выводы об алгоритме по его работе на одном тесте невозможно. Поэтому тестирование ваших алгоритмов должно производиться на наборах тестов. Тест / сет должен покрывать некоторый класс ситуаций. Например, для алгоритмов на графах можно взять подобные наборы тестов: – Все простые графы на не более, чем 8 вершинами. – Случайные графы G(n, p). – Дистанционные, планарные, кнезеровские графы. – Реальные графы дорог, web-графы, любые другие реальные данные.

5.2 Анализ запусков

Агрегированные по тест-сетам статистики данных более значимы, чем точные значения по конкретным запускам. В первую очередь нужно вести их анализ (считать минимум / максимум / среднее значение / медиану / различные квантили / эмпирическое распределение таких величин, как время / память / точность / вероятность). Работу на конкретных тестах имеет смысл рассматривать, если результаты сильно выбиваются или приближаются к теоретическим оценкам, или же в случае проведения визуализации алгоритма или подробного анализа его шагов. Если есть возможность реализовать наивный/оптимальный/детерминированный алгоритм для решения той же задачи, то это нужно сделать и сравнить результаты разных алгоритмов. Отображение данных на графиках оценивается лучше, чем просто набор табличек с данными.

5.3 Выводы, сделанные по полученным данным

На каких данных алгоритм работает хорошо или плохо. Сравнение с работой базового алгоритма. К каким параметрам или ситуациям наиболее чуствительны интересующие вас величины. Достигаются ли теоретические оценки на тест-сетах или дело обстоит лучше/хуже, чем в теории. Попытки объяснения замеченных аномалий.

6 Литература

1. "Reducibility Among Combinatorial Problems Proc. Symposium on Complexity of Computer Computations, IBM Thomas J. Watson Res. Center, Yorktown Heights, N.Y., New York: Plenum, pp. 85–103

2. "FPT-approximation for FPT Problems Daniel Lokshtanov, Pranabendu Misra, M. S. Ramanujan, Saket Saurabh , and Meirav Zehavi

3. "A fixed-parameter algorithm for the directed feedback vertex set problem Jianer Chen, Yang Liu, Songjian Lu, Barry O'sullivan, Igor Razgon

4. "An Efficient Semi-Streaming PTAS for Tournament Feedback Arc Set with Few Passes Anubhav Baweja, Justin Jia, David P. Woodruff