Дискретная оптимизация. Осень 2015.

Введение

На лекции мы вспомнили/узнали о следующих классических задачах дискретной оптимизации:

(1) покрытие множествами,
(2) вершинное покрытие,
(3) кратчайший путь,
(4) минимальное остовное дерево (MST),
(5) паросочетание наибольшей мощности,
(6) паросочетание наибольшего веса,
(7) паросочетание наибольшей мощности и наименьшей стоимости,
(8) задача о назначениях,
(9) (одномерная) задача об упаковке в контейнеры (bin packing),
(10) задача о 0,1-рюкзаке,
(11) нахождение потока наибольшей величины,
(12) нахождение потока заданной величины и наименьшей стоимости,
(13) мультипродуктовый поток максимальной суммарной величины,
(14) задача коммивояжёра (TSP).

Далее будем считать, что все веса и стоимости в задачах (1)–(14) суть целые числа.

Задача 1. Задачу дискретной оптимизации в общем виде можно представить в форме: «найти точку максимума/минимума функционала f на конечном множестве $\mathcal{S}$». Для каждой из задач (1)–(14) проделайте следующее.

(a) Укажите соответствующие «разумные» множество $\mathcal{S}$ и функционал f (ответ может быть неоднозначным). Обратите внимание, что выбор пары $\mathcal{S}, f$ неоднозначен, его можно осуществлять по-разному: ценой более сложного устройства $\mathcal{S}$ можно уменьшить сложность f и наоборот.

(b) Определите, сколько асимптотически времени требуется, чтобы вычислить значение функционала f в фиксированной точке множества $\mathcal{S}$.

(c) Определите, как мощность множества $\mathcal{S}$ зависит от «размера» содержательной задачи (например, в задаче TSP это количества вершин и рёбер графа, а также диапазон весов рёбер (ведь на запись чисел тоже требуется место)), — полиномиально/линейно/экспоненциально/сверхэкспоненциально.

(d) Оцените максимальный размер задачи, которую можно решить за сутки полным перебором, если на рассмотрение каждого элемента множества $\mathcal{S}$ требуется 10^{-9} с.

Задача 2. Рассмотрим следующие задачи:

1. Для заданного полного n-вершинного графа G с целочисленными положительными весами на рёбрах *найти гамильтонов цикл* наименьшей стоимости в G. (Стандартная формулировка оптимизационной задачи TSP.)
2. Для заданного полного n-вершинного графа G с целочисленными положительными весами на рёбрах *найти минимальную стоимость* гамильтонова цикла в G (при этом сам цикл, имеющий такую стоимость, указывать необязательно). (Вместо поиска точки минимума функционала ищем значение.)
3. Для заданного полного n-вершинного графа G с целочисленными положительными весами на рёбрах и *заданного* числа $C_0 \in \mathbb{Z}$ определить, *существует ли* в G гамильтонов цикл, имеющий стоимость не более C_0. (Задача с ответом «да/нет», так называемая *задача распознавания*.)

Докажите следующие импликации.

(a) Пусть существует алгоритм, решающий третью задачу за время $T(n)$. Тогда существует алгоритм, решающий вторую задачу за время $O\left(T(n) \cdot \log C_{\min}\right)$, где $C_{\min}$ — минимальная стоимость гамильтонова цикла в G.

(b) Пусть существует алгоритм, решающий вторую задачу за время $T(n)$. Тогда существует алгоритм, решающий первую задачу за время $O(T(n) \cdot n^2)$.

Предложите аналогичные постановки для задач о 0,1-рюкзаке, упаковке в контейнеры и вершинном покрытии; докажите аналогичные импликации.

Хотя множество $\mathcal{S}$ в оптимизационной задаче может быть очень большим, описать его обычно можно компактно (например, в задаче TSP множество $\mathcal{S}$ — все гамильтоновы циклы графа, но чтобы полностью определить $\mathcal{S}$, достаточно задать только сам граф). Под входными данными [алгоритма решения] оптимизационной задачи будем подразумевать именно их компактное описание.

Задача 3. Будем говорить, что задача $\mathcal{A}$ *полиномиально сводится* к задаче $\mathcal{B}$, если существует алгоритм, который преобразует за полиномиальное время входные данные задачи $\mathcal{A}$ во входные данные задачи $\mathcal{B}$, так, что по ответу в задаче $\mathcal{B}$ можно за полиномиальное время вычислить ответ задачи $\mathcal{A}$. Определите, какие из задач (1)–(14) полиномиально сводятся друг к другу. Этот вид сводимости является промежуточным между *сводимостью по Карпу*, при которой ответ в задаче $\mathcal{B}$ должен *совпадать* с ответом в задаче $\mathcal{A}$, и *сводимостью по Тьюрингу*, она же *сводимость по Куку*. Последняя определяется так: говорят, что задача $\mathcal{A}$ *полиномиально сводится (по Тьюрингу)* к задаче $\mathcal{B}$, если существует полиномиальный алгоритм, решающий задачу $\mathcal{A}$, который может обращаться к волшебной процедуре-оракулу, разрешающей задачу $\mathcal{B}$ за один такт времени. Поясните, почему из сводимости по Карпу следует сводимость по Тьюрингу.

Задача 4. Напомним, что в задаче о 0,1-рюкзаке для каждого из предметов нужно определить, включать его в рюкзак или нет. В более общей задаче о целочисленном рюкзаке имеется несколько различных наименований предметов и известно, сколько предметов каждого наименования у нас есть (максимально возможные «кратности»). Требуется, как и в задаче о 0,1-рюкзаке, максимизировать суммарную ценность попавших в рюкзак предметов. *(a)* Поставьте эту задачу математически формально. *(b)* Докажите, что задача о целочисленном рюкзаке полиномиально сводится к задаче о 0,1-рюкзаке.

Локальный поиск и эвристика Кернигана—Лина

Задача 5. *(a)* При рассмотрении 3-окрестности гамильтонова цикла при локальном поиске в задаче TSP мы удаляем 3 ребра из текущего цикла и пытаемся провести три новых ребра. Перечислите все возможные варианты добавления новых рёбер, так, что снова получается гамильтонов цикл.

(b) Пусть теперь мы используем в локальном поиске k-окрестности. Сколькими (в точности) способами можно провести k новых рёбер для получения нового цикла (при условии, что мы уже удалили из текущего цикла k фиксированных рёбер)?

Задача 6. Рассмотрим задачу о вершинном покрытии. Пусть $\mathcal{S}$ — семейство всех подмножеств вершин графа G, образующих в G вершинное покрытие, и мы ищем множество из $\mathcal{S}$, имеющее минимальную мощность. Для $x \in \mathcal{S}$ положим $\mathcal{N}(x) := \{x' \in \mathcal{S} \mid x' \subset x$ и $|x'| = |x| - 1\}$. Докажите, что относительно такой системы окрестностей может экспоненциально много локальных оптимумов, например, когда G является простой цепью. Сколько в этом случае глобальных оптимумов?

Задача 7. Приведите примеры сильно связных полиномиально обозримых систем окрестностей для следующих задач: *(a)* вершинное покрытие, *(b)* сбалансированное разбиение графа, *(c)* покрытие множествами, *(d)* 0,1-рюкзак.

Задача 8. Докажите сильную связность системы 2-окрестностей в задаче TSP: покажите, что для любых двух гамильтоновых циклов C', C'' в полном графе можно найти последовательность гамильтоновых циклов $C_1, C_2, \ldots, C_k$, такую, что $C_1 = C'$, $C_k = C''$, и для каждого i цикл C_{i+1} принадлежит 2-окрестности цикла C_i.

Задача 9. Определим k-окрестность гамильтонова цикла C в графе как множество всех циклов, которые можно получить удалением k рёбер из C и добавлением k рёбер (необязательно отличных от бывших). На лекции было показано, что рассмотрение 2-окрестностей не гарантирует достижение глобального оптимума в задаче коммивояжёра при локальном поиске.

(a) Докажите, что система 3-окрестностей не является точной (существуют локальные оптимумы, не являющиеся глобальными). (Внимание! В примерах-картинках вес ребра вовсе не обязан быть пропорциональным длине ребра.)

(b) Докажите, что система $\lceil n/2 \rceil$-окрестностей не является точной, где n — количество вершин в графе.

(c) Докажите, что система $(n-1)$-окрестностей является точной.

Задача 10. Пусть t — фиксированное натуральное число. Будем рассматривать в локальном поиске в задаче TSP уменьшенные 2-окрестности, в которых из цикла удаляется-добавляется только пара рёбер, удалённых друг от друга по циклу на расстояние не больше t. Будет ли такая система окрестностей слабее, чем система обычных 2-окрестностей (то есть, может ли цикл быть локально-оптимальным относительно системы уменьшенных 2-окрестностей, но не являться при этом локально-оптимальным относительно системы обычных 2-окрестностей)?

Задача 11. Оцените время работы итерации обычного локального поиска в задаче о сбалансированном разбиении графа при использовании k-окрестностей (обмен не более k пар вершин между частями разбиения) и сравните это со временем работы локального поиска переменной глубины (Керниган—Лин) с использованием на каждом элементарном шаге замен одной вершины на одну вершину и глубиной не более k.

Задача 12. Рассмотрим задачу о сбалансированном разбиении графа с весами на рёбрах, в которой части разбиения должны быть равными ($\alpha = \frac{1}{2}$). Напомним, что в этой задаче k-окрестностью разбиения мы называем семейство всех таких разбиений, которые можно получить перебросом в общей сложности не более чем $2k$ вершин между частями.

(a) Верно ли, что для любого $4n$-вершинного графа обычный локальный поиск при использовании n-окрестностей всегда найдёт оптимальное разбиение?

(b) Придумайте пример такого $4n$-вершинного графа и такого начального разбиения этого графа, на котором поиск Кернигана—Лина глубины не более n находит оптимальное решение, а обычный локальный поиск при использовании $(n-1)$-окрестностей не найдёт оптимума.

(c) Придумайте пример $4n$-вершинного графа, на котором поиск Кернигана—Лина глубины не более n может не найти оптимум, а обычный локальный поиск при использовании n-окрестностей всегда находит оптимум.

(d) Придумайте пример $4n$-вершинного графа, на котором поиск Кернигана—Лина глубины не более $2n$ (который при построении цепочек переходов использует 1-окрестности) всегда находит оптимальное решение, а обычный локальный поиск при использовании $(n-1)$-окрестностей может не найти оптимума. Сравните вычислительную сложность соответствующих алгоритмов.