Кратчайшие пути в графе

Лекция 5

Кратчайший путь

Пусть задан связный ориентированный граф $G = (V, E)$ с весовой функцией $w: E \to R^{+}$.

Вес пути $p \;= \langle v_0, v_1, \dots, v_k \rangle$ равен суммарному весу входящих в него ребер:

$$\delta(p) = \sum_{i=1}^{k} w(v_{i-1}, v_i)\,, \text{где} (v_{i-1}, v_i) \in E$$

Вес кратчайшего пути из вершины v в вершину u определяется соотношением

$$\delta(v,u) = \begin{cases} \min\{\delta(p) \mid p - \text{путь из } v \text{ в } u\}, & \text{если существуют пути } p \text{ из } v \text{ в } u \\ \infty, & \text{в противном случае.} \end{cases}$$

Кратчайший путь ≡ путь минимального веса (вес=длина).

Кратчайшее расстояние между вершинами ≡ длина кратчайшего пути между ними

Разновидности задач о кратчайших путях

- Найти кратчайшее расстояние от заданной вершины s до остальных вершин (Алгоритм Дейкстры).
- Найти кратчайшее расстояние от заданной вершины s до вершины t (Алгоритм Дейкстры).
- Найти кратчайшее расстояние от любой вершины до заданной вершины s (Алг Дейкстры наоборот).
- Найти кратчайшее расстояние между всеми парами вершин (n раз Алг Дейкстры или Алг Флойда-Уоршалла).

Алгоритм Дейкстры $(w: E \to R^+)$

Найти кратчайшее расстояние от заданной вершины s до остальных вершин

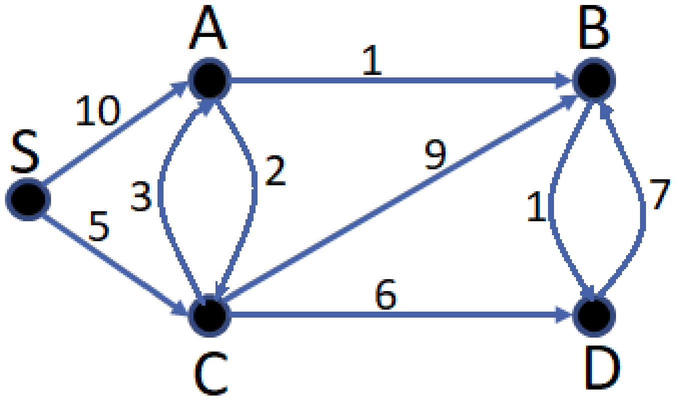
- W – множество вершин, для которых уже вычислены кратчайшие расстояния из s.
- $\delta(x)$ – вес минимального пути из s в x.
- $\pi(x)$ – предпоследняя вершина в минимальном пути из s в x.

1. Положить $W = \emptyset,\ \delta(s) = 0,\ \delta(x) = \infty,\ \forall\, x \in V \backslash \{s\}$
2. Пока $W \neq V$ выполнять:
   1. Найти $x \in V \backslash W$, такую, что $\delta(x) = \min\{\delta(y) \mid y \in V \backslash W\}$
   2. $W := W \cup \{x\};$
   3. Для всех $y \in V \backslash W$

      Если $\delta(y) > \delta(x) + w(x,y)$, то $\delta(y) = \delta(x) + w(x,y),\ \pi(y) = x;$

Массив $\pi(x)$ нужен для восстановления кратчайшего пути в вершину x (узнаем предыдущую для x (это $\pi(x)$), для нее ее предыдущую и так до s)

Алгоритм Дейкстры



	S	A	B	C	D
$\delta(x)$	0	∞	∞	∞	∞
S		10		5	

$W = \emptyset$

Находим вершину с мин δ, это S

Обновляем $\delta(A)$, $\delta(C)$,

		A	B	C	D
$\delta(x)$		10	∞	5	∞
C		8	14		11

$W = \{S\}$

Находим вершину с мин δ, это C

Обновляем $\delta(A)$, $\delta(B)$, $\delta(D)$

		A	B		D
$\delta(x)$		8	14		11
A			9		

$W = \{S, C\}$

Находим вершину с мин δ, это A

Обновляем $\delta(B)$

			B		D
$\delta(x)$			9		11
D					10

$W = \{S, C, A\}$

Находим вершину с мин δ, это B

Обновляем $\delta(D)$

$W = \{S, C, A, B\}$, остается только вершина D, добавляем ее и все кончается

Алгоритм Дейкстры. Теорема

Теорема 1

Алгоритм Дейкстры находит кратчайшие пути из вершины s до каждой из остальных вершин за время $O(|V|^2)$.

Доказательство. Покажем, что на каждой итерации:

а) $\forall\, x \in W$ величина $\delta(x)$ равна длине кратчайшего из путей от s до x.

б) $\forall y \in V$ величина $\delta(y)$ равна длине кратчайшего из путей от s до y, все промежуточные вершины которых принадлежат W.

Так как в конце работы алгоритма $W = V$, то из а) следует, что $\delta(x)$ — вектор кратчайших расстояний.

Алгоритм Дейкстры. Теорема продолжение

… Покажем, что на каждой итерации:

а) $\forall\, x \in W$ величина $\delta(x)$ равна длине кратчайшего из путей от s до x.

б) $\forall y \in V$ величина $\delta(y)$ равна длине кратчайшего из путей от s до y, все промежуточные вершины которых принадлежат W.

Докажем по индукции. При $W = \{s\}$ утверждения а) и б) верны.

Пусть а) и б) верны до некоторого шага, покажем, что они остаются верны.

На шаге 2.1 выбирается вершина x, такая что $\delta(x) = \min\{\, \delta\,(y) |\, y \in V \setminus \mathrm{W}\,\}$. Вершина x добавляется в W, нужно проверить для нее условие а). Пусть а) не выполняется для вершины x. Значит существует путь $(s, v_1, \dots, v_t, x)$, длина которого меньше $\delta\,(x)$. Тогда из б) следует, что в этом пути есть вершина $v_i \notin W$. Если таких несколько, выберем вершину с наименьшим номером.
Тогда $\delta(v_i) \leq$ длина $(s, v_1, \dots, v_i) \leq$ длина $(s, v_1, \dots, v_t, x) < \delta(x)$, что противоречит выбору x. Значит такого пути нет и $\delta(x)$ – длина кратчайшего пути от s до x и а) будет выполняться после добавления x к W.

На шаге 2.3 выполняется пересчет для вершин y через новую добавленную к W вершину x. Значит б) будет выполняться, т.к. путь будет либо без x (и верно по предположению индукции) либо через x он короче.

Трудоемкость: Цикл 2 требует $O(|V|)$ итераций. На каждой итерации 2.1 или 2.3 требуется $O(|V|)$ действий. Итого $O(|V|^2)$

Алгоритм Флойда-Уоршалла

Найти кратчайшее расстояние $\delta(i,j)$ между всеми вершинами

Инициализация $\forall(i,j) \in E, \delta(i,j) = w(i,j)$, остальные $\delta(i,j) = \infty$

For $j = 1..n$ do

For $i = 1..n$ do

For $k = 1..n$ do

If $(i \neq j)\&\ (k \neq j)\&\ (i \neq k)$

$z := \delta(i,k);$

$\delta(i,k) := \min\{\delta(i,k);\ \delta(i,j) + \delta(j,k)\};$ т. е. $\delta(i,k) \leq \delta(i,j) + \delta(j,k)$ - операция треугольник

если $\delta(i,k) < z$, то $\pi(i,k) = j$.

Время $O(|V|^3)$.

Алгоритм работает корректно, даже если есть дуги отрицательной длины, но нет контуров отрицательной длины.

Алгоритм Флойда-Уоршалла. Теорема

Теорема 2

Алгоритм Флойда-Уоршалла находит длины кратчайших путей из каждой вершины до остальных вершин за время $O(|V|^3)$.

Доказательство. Трудоемкость очевидна.

- Покажем, что для каждого j после выполнения операций треугольника для $t = 1, 2, \dots, j$ элемент $\delta(i, k)$ для любых i и k равен длине кратчайшего пути из i в k среди всех путей, промежуточные вершины которых имеют номера не больше j.
- Для $j = 1$ это утверждение очевидно.
- Пусть утверждение верно для $j = t - 1$, и проводится операция для t:
  $\delta(i, k) \coloneqq \min\{\delta(i, k);\ \delta(i, j) + \delta(j, k)\}$.
  Рассмотрим подграф G' орграфа G на вершинах $\{1, 2, \dots, t, i, k\}$. Если кратчайший путь из i в k в G' не проходит через t, то минимум достигается на первом аргументе и утверждение верно. Если же он проходит через t, то $\delta(i, t) + \delta(t, k) \le \delta(i, k)$, а по предыдущему предположению $\delta(i, t)$ и $\delta(t, k)$– длины кратчайших путей из i в t и из t в k по вершинам с номерами не более t.

Задача о кратчайшей связывающей сети

Кратчайшая связывающая сеть = остовное дерево минимального веса

Остовное дерево — ациклический связный подграф данного связного неориентированного графа, в который входят все его вершины.

Неформально говоря, остовное дерево состоит из некоторого подмножества рёбер графа, таких, что из любой вершины графа можно попасть в любую другую вершину, двигаясь по этим рёбрам, и в нём нет циклов, то есть из любой вершины нельзя попасть в саму себя, не пройдя какое-то ребро дважды.

Алгоритм Краскала $(w: E \to R^{+})$

1. *Отсортировать ребра в порядке неуменьшения веса* $w(e_1) \leq w(e_2) \leq \cdots \leq w(e_m)$.
2. $T = \emptyset$
3. $For\ i = 1\ to\ m$
4. *Если* $T \cup e_i$ *не содержит цикл, то добавить* e_i *к* T

Полученное дерево T является искомым