с/к “Эффективные алгоритмы”
Лекция 11: Вероятностные алгоритмы

А. Куликов

Computer Science клуб при ПОМИ
http://logic.pdmi.ras.ru/∼infclub/

План лекции

План лекции

1. Нахождение матрицы расстояний

2. Поиск виновников в умножении булевых матриц

План лекции

1. Нахождение матрицы расстояний
2. Поиск виновников в умножении булевых матриц
3. Поиск кратчайших путей

Кратчайшие пути в графе

Определение

Кратчайшие пути в графе

Определение

- Дан связный неориентированный граф $G = (V, E)$, вес каждого ребра равен 1.

Кратчайшие пути в графе

Определение

- Дан связный неориентированный граф $G = (V, E)$, вес каждого ребра равен 1.
- Матрицей расстояний (distance matrix) называется матрица D, в которой $D[i, j]$ равно длине кратчайшего пути из i в j.

Кратчайшие пути в графе

Определение

- Дан связный неориентированный граф $G = (V, E)$, вес каждого ребра равен 1.
- Матрицей расстояний (distance matrix) называется матрица D, в которой $D[i, j]$ равно длине кратчайшего пути из i в j.
- Матрицей кратчайших путей (shortest path matrix) называется матрица P, в которой $P[i, j]$ — первая вершина на кратчайшем пути из i в j.

Кратчайшие пути в графе

Определение

- Дан связный неориентированный граф $G = (V, E)$, вес каждого ребра равен 1.
- Матрицей расстояний (distance matrix) называется матрица D, в которой $D[i, j]$ равно длине кратчайшего пути из i в j.
- Матрицей кратчайших путей (shortest path matrix) называется матрица P, в которой $P[i, j]$ — первая вершина на кратчайшем пути из i в j.
- Задача нахождения кратчайших расстояний (all-pairs distances problem, ADP) и задача нахождения кратчайших путей (all-pairs shortest paths problem, APSP) заключаются в построении соответствующих матриц.

План лекции

1. Нахождение матрицы расстояний
2. Поиск виновников в умножении булевых матриц
3. Поиск кратчайших путей

Матрица смежности и длины путей

Матрица смежности и длины путей

Матрица смежности и длины путей

Матрица смежности и длины путей

- Вспомним, что если A — матрица смежности графа G, то $A^k[i,j]$ — количество путей длины k из i в j.

Матрица смежности и длины путей

Матрица смежности и длины путей

- Вспомним, что если A — матрица смежности графа G, то $A^k[i,j]$ — количество путей длины k из i в j.
- Таким образом, посчитав все степени A от 1 до n, мы решим задачу ADP, но это займет $O(n^{3.376})$ времени.

Квадрат графа

Определение

Квадратом графа $G = (V, E)$ называется граф $G = (V, E')$, где различные i и j соединены ребром, если расстояние между ними в G не более 2.

Матрица расстояний квадрата графа

Квадрат графа

Определение

Квадратом графа $G = (V, E)$ называется граф $G = (V, E')$, где различные i и j соединены ребром, если расстояние между ними в G не более 2.

Матрица расстояний квадрата графа

- Ясно, что по квадрату матрицы смежности A^2 графа G легко построить матрицу смежности графа G'.

Квадрат графа

Определение

Квадратом графа $G = (V, E)$ называется граф $G = (V, E')$, где различные i и j соединены ребром, если расстояние между ними в G не более 2.

Матрица расстояний квадрата графа

- Ясно, что по квадрату матрицы смежности A^2 графа G легко построить матрицу смежности графа G'.
- Если диаметр (то есть длина самого длинного кратчайшего пути) графа G не более 2, то граф G' — полный.

Квадрат графа

Определение

Квадратом графа $G = (V, E)$ называется граф $G = (V, E')$, где различные i и j соединены ребром, если расстояние между ними в G не более 2.

Матрица расстояний квадрата графа

- Ясно, что по квадрату матрицы смежности A^2 графа G легко построить матрицу смежности графа G'.
- Если диаметр (то есть длина самого длинного кратчайшего пути) графа G не более 2, то граф G' — полный.
- В таком случае $D = 2A' - A$.

Квадрат графа

Определение

Квадратом графа $G = (V, E)$ называется граф $G = (V, E')$, где различные i и j соединены ребром, если расстояние между ними в G не более 2.

Матрица расстояний квадрата графа

- Ясно, что по квадрату матрицы смежности A^2 графа G легко построить матрицу смежности графа G'.
- Если диаметр (то есть длина самого длинного кратчайшего пути) графа G не более 2, то граф G' — полный.
- В таком случае $D = 2A' - A$.
- В общем же случае матрицу D можно строить рекурсивно.

Рекурсивное вычисление матрицы расстояний

Лемма

Пусть D — матрица кратчайших расстояний графа G, G' — квадрат графа G, D' — матрица кратчайших расстояний графа G'. Тогда для любых двух вершин i, j

$$D[i,j] = \begin{cases} 2D'[i,j], & \text{если } D[i,j] \text{ четно,} \\ 2D'[i,j] - 1, & \text{если } D[i,j] \text{ нечетно.} \end{cases}$$

Рекурсивное вычисление матрицы расстояний

Лемма

Пусть D — матрица кратчайших расстояний графа G, G' — квадрат графа G, D' — матрица кратчайших расстояний графа G'. Тогда для любых двух вершин i, j

$$D[i,j] = \begin{cases} 2D'[i,j], & \text{если } D[i,j] \text{ четно,} \\ 2D'[i,j] - 1, & \text{если } D[i,j] \text{ нечетно.} \end{cases}$$

Упражнение

Доказать эту лемму.

Рекурсивное вычисление матрицы расстояний

Лемма

Пусть D — матрица кратчайших расстояний графа G, G' — квадрат графа G, D' — матрица кратчайших расстояний графа G'. Тогда для любых двух вершин i, j

$$D[i,j] = \begin{cases} 2D'[i,j], & \text{если } D[i,j] \text{ четно,} \\ 2D'[i,j] - 1, & \text{если } D[i,j] \text{ нечетно.} \end{cases}$$

Упражнение

Доказать эту лемму.

Итак, мы могли бы рекурсивно вычислять матрицу D, если бы заранее знали четность ее элементов.

Нахождение четности крачтайшего пути

Лемма

Для любых двух вершин i и j графа G

Нахождение четности крачтайшего пути

Лемма

Для любых двух вершин i и j графа G

1. $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$ для любого соседа k вершины i;

Нахождение четности крачтайшего пути

Лемма

Для любых двух вершин i и j графа G

1. $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$ для любого соседа k вершины i;
2. $D[k,j] = D[i,j] - 1$ для некоторого соседа k вершины i.

Нахождение четности крачтайшего пути

Лемма

Для любых двух вершин i и j графа G

1. $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$ для любого соседа k вершины i;
2. $D[k,j] = D[i,j] - 1$ для некоторого соседа k вершины i.

Лемма

Для любых двух вершин i и j графа G

Нахождение четности кратчайшего пути

Лемма

Для любых двух вершин i и j графа G

1. $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$ для любого соседа k вершины i;
2. $D[k,j] = D[i,j] - 1$ для некоторого соседа k вершины i.

Лемма

Для любых двух вершин i и j графа G

1. если $D[i,j]$ четно, то $D'[k,j] \geq D'[i,j]$ для любого соседа k вершины i;

Нахождение четности крачтайшего пути

Лемма

Для любых двух вершин i и j графа G

1. $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$ для любого соседа k вершины i;
2. $D[k,j] = D[i,j] - 1$ для некоторого соседа k вершины i.

Лемма

Для любых двух вершин i и j графа G

1. если $D[i,j]$ четно, то $D'[k,j] \geq D'[i,j]$ для любого соседа k вершины i;
2. если $D[i,j]$ нечетно, то $D'[k,j] \leq D'[i,j]$ для любого соседа k вершины i; более того, для какого-то соседа k вершины i $D'[k,j] < D'[i,j]$.

Доказательство

Доказательство

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$
   - $D[k,j] \geq 2l-1$ для любого соседа k вершины i

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$
   - $D[k,j] \geq 2l - 1$ для любого соседа k вершины i
   - $D'[k,j] \geq D[k,j]/2 \geq l - 1/2$, а следовательно, $D'[k,j] \geq l = D'[i,j]$

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$
   - $D[k,j] \geq 2l - 1$ для любого соседа k вершины i
   - $D'[k,j] \geq D[k,j]/2 \geq l - 1/2$, а следовательно, $D'[k,j] \geq l = D'[i,j]$
2. Пусть $D[i,j] = 2l - 1$.

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$
   - $D[k,j] \geq 2l - 1$ для любого соседа k вершины i
   - $D'[k,j] \geq D[k,j]/2 \geq l - 1/2$, а следовательно, $D'[k,j] \geq l = D'[i,j]$
2. Пусть $D[i,j] = 2l - 1$.
   - $D'[i,j] = l$

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$
   - $D[k,j] \geq 2l - 1$ для любого соседа k вершины i
   - $D'[k,j] \geq D[k,j]/2 \geq l - 1/2$, а следовательно, $D'[k,j] \geq l = D'[i,j]$
2. Пусть $D[i,j] = 2l - 1$.
   - $D'[i,j] = l$
   - $D[k,j] \leq 2l$ для любого соседа k вершины i

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$
   - $D[k,j] \geq 2l - 1$ для любого соседа k вершины i
   - $D'[k,j] \geq D[k,j]/2 \geq l - 1/2$, а следовательно, $D'[k,j] \geq l = D'[i,j]$
2. Пусть $D[i,j] = 2l - 1$.
   - $D'[i,j] = l$
   - $D[k,j] \leq 2l$ для любого соседа k вершины i
   - $D'[k,j] \leq (D[k,j]+1)/2 \leq l + 1/2$, а следовательно, $D'[k,j] \leq l = D'[i,j]$

Доказательство

Доказательство

1. Пусть $D[i,j] = 2l$.
   - $D'[i,j] = l$
   - $D[k,j] \geq 2l - 1$ для любого соседа k вершины i
   - $D'[k,j] \geq D[k,j]/2 \geq l - 1/2$, а следовательно, $D'[k,j] \geq l = D'[i,j]$
2. Пусть $D[i,j] = 2l - 1$.
   - $D'[i,j] = l$
   - $D[k,j] \leq 2l$ для любого соседа k вершины i
   - $D'[k,j] \leq (D[k,j] + 1)/2 \leq l + 1/2$, а следовательно, $D'[k,j] \leq l = D'[i,j]$
   - для какого-то соседа k вершины i $D[k,j] = D[i,j] - 1 = 2l - 2$ и $D'[k,j] = l - 1 < l = D'[i,j]$

$\square$

Полезное следствие

Следствие

Пусть $\Gamma(i)$ — множество соседей вершины i. Пусть также

$$\Delta = \sum_{k \in \Gamma(i)} D'[k,j] \geq |\Gamma(i)| D'[i,j].$$

Тогда

Полезное следствие

Следствие

Пусть $\Gamma(i)$ — множество соседей вершины i. Пусть также

$$\Delta = \sum_{k \in \Gamma(i)} D'[k,j] \geq |\Gamma(i)| D'[i,j].$$

Тогда

1. $D[i,j]$ четно тогда и только тогда, когда $\Delta \geq 0$.

Полезное следствие

Следствие

Пусть $\Gamma(i)$ — множество соседей вершины i. Пусть также

$$\Delta = \sum_{k \in \Gamma(i)} D'[k,j] \geq |\Gamma(i)| D'[i,j].$$

Тогда

1. $D[i,j]$ четно тогда и только тогда, когда $\Delta \geq 0$.
2. $D[i,j]$ нечетно тогда и только тогда, когда $\Delta < 0$.

Алгоритм

Алгоритм

APD-ALG(матрица смежности A графа $G(V, E)$)

Алгоритм

Алгоритм

APD-Alg(матрица смежности A графа $G(V, E)$)

- $Z = A^2$

Алгоритм

Алгоритм

APD-ALG(матрица смежности A графа $G(V, E)$)

- $Z = A^2$
- построим матрицу A': $A'[i, j] = 1$, если и только если $i \neq j$, а также либо $A[i, j] = 1$, либо $Z[i, j] > 0$

Алгоритм

Алгоритм

APD-ALG(матрица смежности A графа $G(V, E)$)

- $Z = A^2$
- построим матрицу A': $A'[i, j] = 1$, если и только если $i \neq j$, а также либо $A[i, j] = 1$, либо $Z[i, j] > 0$
- если в A' единицы стоят везде, кроме диагонали, вернуть $D = 2A' - A$

Алгоритм

Алгоритм

$\textsc{APD-Alg}$(матрица смежности A графа $G(V,E)$)

- $Z = A^2$
- построим матрицу A': $A'[i,j] = 1$, если и только если $i \neq j$, а также либо $A[i,j] = 1$, либо $Z[i,j] > 0$
- если в A' единицы стоят везде, кроме диагонали, вернуть $D = 2A' - A$
- $D' = \textsc{APD-Alg}(A')$

Алгоритм

Алгоритм

$\textsc{APD-Alg}$(матрица смежности A графа $G(V, E)$)

- $Z = A^2$
- построим матрицу A': $A'[i, j] = 1$, если и только если $i \neq j$, а также либо $A[i, j] = 1$, либо $Z[i, j] > 0$
- если в A' единицы стоят везде, кроме диагонали, вернуть $D = 2A' - A$
- $D' = \textsc{APD-Alg}(A')$
- $S = AD'$

Алгоритм

Алгоритм

APD-ALG(матрица смежности A графа $G(V, E)$)

- $Z = A^2$
- построим матрицу A': $A'[i, j] = 1$, если и только если $i \neq j$, а также либо $A[i, j] = 1$, либо $Z[i, j] > 0$
- если в A' единицы стоят везде, кроме диагонали, вернуть $D = 2A' - A$
- $D' = \text{APD-ALG}(A')$
- $S = AD'$
- вернуть матрицу D, где

$$D[i, j] = \begin{cases} 2D'[i, j], & \text{если } S[i, j] \geq D'[i, j]Z[i, i], \\ 2D'[i, j] - 1, & \text{если } S[i, j] < D'[i, j]Z[i, i]. \end{cases}$$

Анализ

Теорема

Алгоритм APD-Alg корректно находит матрицу кратчайших расстояний за время $O(n^{2.376} \log n)$.

План лекции

1 Нахождение матрицы расстояний

2 Поиск виновников в умножении булевых матриц

3 Поиск кратчайших путей

Виновники в умножении булевых матриц

Виновники в умножении булевых матриц

Виновники в умножении булевых матриц

Виновники в умножении булевых матриц

- Пусть P — булево произведение матриц A и B, то есть

$$P[i,j] = \bigvee_{1 \le k \le n} (A[i,k] \wedge B[k,j]).$$

Виновники в умножении булевых матриц

Виновники в умножении булевых матриц

- Пусть P — булево произведение матриц A и B, то есть
$$P[i,j] = \bigvee_{1 \le k \le n} (A[i,k] \wedge B[k,j]).$$
- Виновником (witness) элемента $P[i,j]$ называется такой индекс k, что $A[i,k] = B[k,j] = 1$.

Виновники в умножении булевых матриц

Виновники в умножении булевых матриц

- Пусть P — булево произведение матриц A и B, то есть

$$P[i,j] = \bigvee_{1 \le k \le n} (A[i,k] \wedge B[k,j]).$$

- Виновником (witness) элемента $P[i,j]$ называется такой индекс k, что $A[i,k] = B[k,j] = 1$.
- Ясно, что $P[i,j] = 1$ тогда и только тогда, когда для элемента $P[i,j]$ есть виновник.

Виновники в умножении булевых матриц

Виновники в умножении булевых матриц

- Пусть P — булево произведение матриц A и B, то есть

$$P[i,j] = \bigvee_{1 \le k \le n} (A[i,k] \wedge B[k,j]).$$

- Виновником (witness) элемента $P[i,j]$ называется такой индекс k, что $A[i,k] = B[k,j] = 1$.
- Ясно, что $P[i,j] = 1$ тогда и только тогда, когда для элемента $P[i,j]$ есть виновник.
- Матрицей виновников произведения матриц (Boolean product witness matrix, BPWM) называется матрица W, такая что $W[i,j]$ —виновников элемента $P[i,j]$, если хотя бы один виновник есть, и $W[i,j] = 0$ в противном случае.

Виновники в умножении булевых матриц

Виновники в умножении булевых матриц

- Пусть P — булево произведение матриц A и B, то есть

$$P[i,j] = \bigvee_{1 \le k \le n} (A[i,k] \wedge B[k,j]).$$

- Виновником (witness) элемента $P[i,j]$ называется такой индекс k, что $A[i,k] = B[k,j] = 1$.
- Ясно, что $P[i,j] = 1$ тогда и только тогда, когда для элемента $P[i,j]$ есть виновник.
- Матрицей виновников произведения матриц (Boolean product witness matrix, BPWM) называется матрица W, такая что $W[i,j]$ —виновников элемента $P[i,j]$, если хотя бы один виновник есть, и $W[i,j] = 0$ в противном случае.
- Заметим, что целочисленное произведение булевых матриц A и B есть в точности матрица виновников.

Виновники в графе

Виновники в графе

Виновники в графе

Виновники в графе

- Если $A = B$ — матрица смежности графа G, то $P[i, j] = 1$, если и только если существует путь из i в j длины 2.

Виновники в графе

Виновники в графе

- Если $A = B$ — матрица смежности графа G, то $P[i,j] = 1$, если и только если существует путь из i в j длины 2.
- Тогда $C[i,j]$ — количество таких путей, а виновник k элмента $P[i,j]$ — это просто вершина на пути длины 2 из i в j.

Виновники в графе

Виновники в графе

- Если $A = B$ — матрица смежности графа G, то $P[i,j] = 1$, если и только если существует путь из i в j длины 2.
- Тогда $C[i,j]$ — количество таких путей, а виновник k элмента $P[i,j]$ — это просто вершина на пути длины 2 из i в j.
- Таким образом, нахождение виновников может помочь превратить алгоритм нахождения расстояний в алгоритм нахождения кратчайших путей.

Виновники в графе

Виновники в графе

- Если $A = B$ — матрица смежности графа G, то $P[i,j] = 1$, если и только если существует путь из i в j длины 2.
- Тогда $C[i,j]$ — количество таких путей, а виновник k элмента $P[i,j]$ — это просто вершина на пути длины 2 из i в j.
- Таким образом, нахождение виновников может помочь превратить алгоритм нахождения расстояний в алгоритм нахождения кратчайших путей.

Упражнение

Пусть матрица $\hat{A}$ получена из A умножением каждого элемента на номер его столбца: $\hat{A}[i,k] = kA[i,k]$. Покажите, что при умножении матриц $\hat{A}$ и B получается матрица, содержащая количсетво виновников для всех элементов, для которых это количество равно 1.

Угадывание виновников

Угадывание виновников

Угадывание виновников

Угадывание виновников

- Конечно, бывают и элементы, для которых есть больше одного виновника.

Угадывание виновников

Угадывание виновников

- Конечно, бывают и элементы, для которых есть больше одного виновника.
- Пусть число виновников элемента $P[i,j]$ равно $w \geq 2$.

Угадывание виновников

Угадывание виновников

- Конечно, бывают и элементы, для которых есть больше одного виновника.
- Пусть число виновников элемента $P[i, j]$ равно $w \geq 2$.
- Рассмотрим целое число r, для которого $n/2 \leq wr \leq n$.

Угадывание виновников

Угадывание виновников

- Конечно, бывают и элементы, для которых есть больше одного виновника.
- Пусть число виновников элемента $P[i, j]$ равно $w \geq 2$.
- Рассмотрим целое число r, для которого $n/2 \leq wr \leq n$.
- С большой вероятностью случайное подмножество размера r множества $\{1, \ldots, n\}$ содержит ровно одного виновника.

Оценка вероятности

Лемма

В корзине имеется n шаров, w из которых белого цвета, $n - w$ — черного. Для r, удовлетворяющего $n/2 \leq wr \leq n$, вероятность того, что среди случайно выбранных r шаров будет ровно один белый, хотя бы $\frac{1}{2e}$.

Оценка вероятности

Лемма

В корзине имеется n шаров, w из которых белого цвета, $n-w$ — черного. Для r, удовлетворяющего $n/2 \le wr \le n$, вероятность того, что среди случайно выбранных r шаров будет ровно один белый, хотя бы $\frac{1}{2e}$.

Доказательство

$$\frac{\binom{w}{1}\binom{n-w}{r-1}}{\binom{n}{r}} = w\frac{r!}{(r-1)!}\frac{(n-w)!}{n!}\frac{(n-r)!}{(n-w-r+1)!}$$

$$= wr\left(\prod_{i=0}^{w-1}\frac{1}{n-i}\right)\left(\prod_{j=0}^{w-2}(n-r-j)\right)$$

Угадывание уникальных виновников

Угадывание уникальных виновников

Угадывание уникальных виновников

Угадывание уникальных виновников

- Зная множество R, содержащее ровно одного виновника для $P[i, j]$, самого это виновника найти легко.

Угадывание уникальных виновников

Угадывание уникальных виновников

- Зная множество R, содержащее ровно одного виновника для $P[i,j]$, самого это виновника найти легко.
- Пусть R задано характеристическим вектором ($R[k]=1$, если $k \in R$, и $R[k]=0$ в противном случае).

Угадывание уникальных виновников

Угадывание уникальных виновников

- Зная множество R, содержащее ровно одного виновника для $P[i,j]$, самого это виновника найти легко.
- Пусть R задано характеристическим вектором ($R[k] = 1$, если $k \in R$, и $R[k] = 0$ в противном случае).
- Пусть матрицы A^R, B^R задаются следующим образом:
$$A^R[i,k] = kR[k]A[i,k], \; B^R[k,j] = R[k]B[k,j].$$

Угадывание уникальных виновников

Угадывание уникальных виновников

- Зная множество R, содержащее ровно одного виновника для $P[i,j]$, самого это виновника найти легко.
- Пусть R задано характеристическим вектором ($R[k] = 1$, если $k \in R$, и $R[k] = 0$ в противном случае).
- Пусть матрицы A^R, B^R задаются следующим образом:
$$A^R[i,k] = kR[k]A[i,k],\ B^R[k,j] = R[k]B[k,j].$$
- В частности, все столбцы A и строчки B, для которых нет соответствующего элемента в R, обнуляются.

Угадывание уникальных виновников

Угадывание уникальных виновников

- Зная множество R, содержащее ровно одного виновника для $P[i,j]$, самого это виновника найти легко.
- Пусть R задано характеристическим вектором ($R[k]=1$, если $k \in R$, и $R[k]=0$ в противном случае).
- Пусть матрицы A^R, B^R задаются следующим образом:
$$A^R[i,k]=kR[k]A[i,k],\ B^R[k,j]=R[k]B[k,j].$$
- В частности, все столбцы A и строчки B, для которых нет соответствующего элемента в R, обнуляются.

Упражнение

Пусть множество R содержит ровно одного виновника элемента $P[i,j]$. Показать, что этот виновник равен $(A^R B^R)[i,j]$.

Алгоритм

Алгоритм

BMPW-Alg($A, B \in \{0, 1\}^{n \times n}$)

Алгоритм

Алгоритм

BMPW-Alg($A, B \in \{0,1\}^{n \times n}$)

- $W = -AB$

Алгоритм

Алгоритм

$\text{BMPW-}\textsc{Alg}(A, B \in \{0,1\}^{n \times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять

Алгоритм

Алгоритм

$\text{BMPW-}\textsc{Alg}(A, B \in \{0,1\}^{n \times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять
  - $r = 2^t$

Алгоритм

Алгоритм

$\text{BMPW-}\textsc{Alg}(A, B \in \{0,1\}^{n\times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять
  - $r = 2^t$
  - повторять $4\log n$ раз

Алгоритм

Алгоритм

$\text{BMPW-}\textsc{Alg}(A, B \in \{0,1\}^{n \times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять
  - $r = 2^t$
  - повторять $4 \log n$ раз
    - выбрать случайное множество $R \subseteq \{1, \ldots, n\}$ размера r

Алгоритм

Алгоритм

$\text{BMPW-Alg}(A, B \in \{0,1\}^{n \times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять
  - $r = 2^t$
  - повторять $4 \log n$ раз
    - выбрать случайное множество $R \subseteq \{1, \ldots, n\}$ размера r
    - $Z = A^R B^R$

Алгоритм

Алгоритм

$\text{BMPW-Alg}(A, B \in \{0,1\}^{n\times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять
  - $r = 2^t$
  - повторять $4\log n$ раз
    - выбрать случайное множество $R \subseteq \{1,\ldots,n\}$ размера r
    - $Z = A^R B^R$
    - для всех (i,j): если $W[i,j] < 0$ и $Z[i,j]$ является виновником, то $W[i,j] = Z[i,j]$

Алгоритм

Алгоритм

$\textsc{BMPW-Alg}(A, B \in \{0,1\}^{n \times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять
  - $r = 2^t$
  - повторять $4 \log n$ раз
    - выбрать случайное множество $R \subseteq \{1, \dots, n\}$ размера r
    - $Z = A^R B^R$
    - для всех (i,j): если $W[i,j] < 0$ и $Z[i,j]$ является виновником, то $W[i,j] = Z[i,j]$
- для всех (i,j): если $W[i,j] < 0$, то найти виновника перебором

Алгоритм

Алгоритм

$\text{BMPW-ALG}(A, B \in \{0,1\}^{n \times n})$

- $W = -AB$
- для t от 1 до $\log n$ повторять
  - $r = 2^t$
  - повторять $4 \log n$ раз
    - выбрать случайное множество $R \subseteq \{1, \ldots, n\}$ размера r
    - $Z = A^R B^R$
    - для всех (i, j): если $W[i, j] < 0$ и $Z[i, j]$ является виновником, то $W[i, j] = Z[i, j]$
- для всех (i, j): если $W[i, j] < 0$, то найти виновника перебором
- вернуть W

Анализ алгоритма

Теорема

Мат. ожидание времени работы алгоритма $\mathrm{BMPW\text{-}A\textsc{lg}}$ равно $O(n^{2.376} \log^2 n)$.

Анализ алгоритма

Теорема

Мат. ожидание времени работы алгоритма BMPW-Alg равно $O(n^{2.376} \log^2 n)$.

Доказательство

Анализ алгоритма

Теорема

Мат. ожидание времени работы алгоритма $\mathrm{BMPW\text{-}A{\scriptsize LG}}$ равно $O(n^{2.376}\log^2 n)$.

Доказательство

- Вероятность того, что для элемента $P[i,j]$ не найдется виновник в цикле хотя бы $(1-1/2e)^{4\log n} \le 1/n$.

Анализ алгоритма

Теорема

Мат. ожидание времени работы алгоритма $\mathrm{BMPW\text{-}A{\scriptsize LG}}$ равно $O(n^{2.376}\log^2 n)$.

Доказательство

- Вероятность того, что для элемента $P[i,j]$ не найдется виновник в цикле хотя бы $(1-1/2e)^{4\log n} \le 1/n$.
- Значит, мат. ожидание кол-ва элементов, для которых виновников придется искать перебором, не более n.

$\square$

План лекции

Поиск кратчайших путей

Поиск кратчайших путей

Поиск кратчайших путей

Поиск кратчайших путей

- Пусть S — матрица кратчайших путей графа G, то сеть $S[i, j]$ — первая вершина на кратчайшем пути из i в j.

Поиск кратчайших путей

Поиск кратчайших путей

- Пусть S — матрица кратчайших путей графа G, то сеть $S[i,j]$ — первая вершина на кратчайшем пути из i в j.
- $S[i,j] = k$, если и только если $D[i,j] = d$, $D[k,j] = d - 1$ и $D[i,k] = 1$.

Поиск кратчайших путей

Поиск кратчайших путей

- Пусть S — матрица кратчайших путей графа G, то сеть $S[i,j]$ — первая вершина на кратчайшем пути из i в j.
- $S[i,j] = k$, если и только если $D[i,j] = d$, $D[k,j] = d - 1$ и $D[i,k] = 1$.
- Пусть $0/1$-матрица B^d задается следующим образом: $B^d[k,j] = 1$, если и только если $D[k,j] = d - 1$.

Поиск кратчайших путей

Поиск кратчайших путей

- Пусть S — матрица кратчайших путей графа G, то сеть $S[i,j]$ — первая вершина на кратчайшем пути из i в j.
- $S[i,j] = k$, если и только если $D[i,j] = d$, $D[k,j] = d-1$ и $D[i,k] = 1$.
- Пусть $0/1$-матрица B^d задается следующим образом: $B^d[k,j] = 1$, если и только если $D[k,j] = d-1$.
- Ясно, что B^d легко вычислить, зная D.

Поиск кратчайших путей

Поиск кратчайших путей

- Пусть S — матрица кратчайших путей графа G, то сеть $S[i, j]$ — первая вершина на кратчайшем пути из i в j.
- $S[i, j] = k$, если и только если $D[i, j] = d$, $D[k, j] = d - 1$ и $D[i, k] = 1$.
- Пусть $0/1$-матрица B^d задается следующим образом: $B^d[k, j] = 1$, если и только если $D[k, j] = d - 1$.
- Ясно, что B^d легко вычислить, зная D.
- Более того, зная B^d легко вычислить первую вершину на пути между вершинами i и j, расстояние между которыми равно d.

Поиск кратчайших путей

Поиск кратчайших путей

- Пусть S — матрица кратчайших путей графа G, то сеть $S[i, j]$ — первая вершина на кратчайшем пути из i в j.
- $S[i, j] = k$, если и только если $D[i, j] = d$, $D[k, j] = d - 1$ и $D[i, k] = 1$.
- Пусть $0/1$-матрица B^d задается следующим образом: $B^d[k, j] = 1$, если и только если $D[k, j] = d - 1$.
- Ясно, что B^d легко вычислить, зная D.
- Более того, зная B^d легко вычислить первую вершину на пути между вершинами i и j, расстояние между которыми равно d.

Упражнение

Показать, что, используя алгоритм $\mathrm{BMPW\text{-}A{\scriptsize LG}}$ для матриц A и B^d, можно найти все первые вершины на кратчайших путях длины d.

Улучшение идеи

Улучшение идеи

Улучшение идеи

Улучшение идеи

- Итак, мы умеем находить первые вершины на путях длины d.

Улучшение идеи

Улучшение идеи

- Итак, мы умеем находить первые вершины на путях длины d.
- Но перебирать все значения d мы не можем.

Улучшение идеи

Улучшение идеи

- Итак, мы умеем находить первые вершины на путях длины d.
- Но перебирать все значения d мы не можем.
- Вспомним, что для любых двух вершин i и j и для любого соседа $k \in \Gamma(i)$ выполнено $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$.

Улучшение идеи

Улучшение идеи

- Итак, мы умеем находить первые вершины на путях длины d.
- Но перебирать все значения d мы не можем.
- Вспомним, что для любых двух вершин i и j и для любого соседа $k \in \Gamma(i)$ выполнено $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$.
- Более того, если $D[i,j] - 1 \leq D[k,j]$, то k — первая вершина на (каком-то) кратчайшем пути из i в j.

Улучшение идеи

Улучшение идеи

- Итак, мы умеем находить первые вершины на путях длины d.
- Но перебирать все значения d мы не можем.
- Вспомним, что для любых двух вершин i и j и для любого соседа $k \in \Gamma(i)$ выполнено $D[i,j] - 1 \leq D[k,j] \leq D[i,j] + 1$.
- Более того, если $D[i,j] - 1 \leq D[k,j]$, то k — первая вершина на (каком-то) кратчайшем пути из i в j.
- Значит, нам достаточно найти такое k, что $A[i,k] = 1$ и $D[k,j] \equiv D[i,j] - 1 \mod 3$.

Алгоритм

Алгоритм

APSP-Alg(матрица смежности A графа G)

Алгоритм

Алгоритм

$\textsc{APSP-Alg}$(матрица смежности A графа G)

- $D = \textsc{APD-Alg}(A)$

Алгоритм

Алгоритм

APSP-Alg(матрица смежности A графа G)

- $D = \text{APD-Alg}(A)$
- для $s \in \{0, 1, 2\}$ повторять

Алгоритм

Алгоритм

$\mathrm{APSP\text{-}A{\scriptsize LG}}$(матрица смежности A графа G)

- $D = \mathrm{APD\text{-}A{\scriptsize LG}}(A)$
- для $s \in \{0, 1, 2\}$ повторять
  - вычислить $0/1$-матрицу D^s: $D^s[k, j] = 1$, если и только если $D[k, j] + 1 \equiv s \mod 3$

Алгоритм

Алгоритм

$\text{APSP-}\textsc{Alg}$(матрица смежности A графа G)

- $D = \text{APD-}\textsc{Alg}(A)$
- для $s \in \{0, 1, 2\}$ повторять
  - вычислить $0/1$-матрицу D^s: $D^s[k, j] = 1$, если и только если $D[k, j] + 1 \equiv s \mod 3$
  - $W^s = \text{BPWM-}\textsc{Alg}(A, D^s)$

Алгоритм

Алгоритм

$\text{APSP-}\textsc{Alg}$(матрица смежности A графа G)

- $D = \text{APD-}\textsc{Alg}(A)$
- для $s \in \{0, 1, 2\}$ повторять
  - вычислить $0/1$-матрицу D^s: $D^s[k, j] = 1$, если и только если $D[k, j] + 1 \equiv s \mod 3$
  - $W^s = \text{BPWM-}\textsc{Alg}(A, D^s)$
- вернуть S, такую что $S[i, j] = W^{D[i,j] \mod 3}[i, j]$

Что мы узнали за сегодня?

Что мы узнали за сегодня?

Что мы узнали за сегодня?

Что мы узнали за сегодня?

- Детерминированный алгоритм поиска расстояний между всеми парами вершин графа.

Что мы узнали за сегодня?

Что мы узнали за сегодня?

- Детерминированный алгоритм поиска расстояний между всеми парами вершин графа.
- Вероятностный алгоритм поиска виновников в умножении булевых матриц.

Что мы узнали за сегодня?

Что мы узнали за сегодня?

- Детерминированный алгоритм поиска расстояний между всеми парами вершин графа.
- Вероятностный алгоритм поиска виновников в умножении булевых матриц.
- Вероятностный алгоритм поиска кратчайших путей между всеми парами вершин графа.

Что мы узнали за сегодня?

Что мы узнали за сегодня?

- Детерминированный алгоритм поиска расстояний между всеми парами вершин графа.
- Вероятностный алгоритм поиска виновников в умножении булевых матриц.
- Вероятностный алгоритм поиска кратчайших путей между всеми парами вершин графа.
- Все алгоритмы работают примерно столько же, сколько алгоритм перемножния матриц.

Спасибо за внимание!