Trumpiausių kelių apskaičiavimas

Vieslav Lapin

Uždavinys I

- Duotas svorinis (*n,m*)-grafas *G*=(*V,U,C*)
  - V – viršunių aibė;
  - U – briaunų aibė;
  - *C* – briaunų svorių aibė;
- Rasti trumpiausius kelius tarp visų viršūnių porų.
- Šio uždavinio sprendimas:
  - matrica $D=[d_{ij}]$
  - d_{ij} = trumpiausio kelio tarp viršūnių *i* ir *j* poros

Uždavinys II

- Jei mus domina ir per kurias viršūnes kelias eina, tada dar papildomai reikalinga
  - Matrica $P=[p_{ij}]$
  - p_{ij} rodo, kuria kryptimi (į kurią viršūnę arba iš kurios viršūnės) iš viršūnės einama.

Uždavinio sprendimo metodai

- Deikstros metodas
- Floido metodas

Deikstros metodas

- Apskaičiujami trumpiausi keliai nuo viršūnės s iki visų likusių viršūnių
- Gauname d[1..n] ir prec[1..n]
- Į Deikstros algoritmą kreipiamasi, kai s kinta nuo 1 iki n
- d – matricos D s-toji eilutė
- prec – matricos P s-toji eilutė

Floido metodas

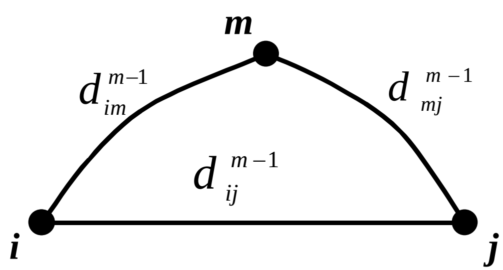
- (n,m)-grafo $G=(V,U,C)$ viršūnės sunumeruotos iš eilės einančiais natūraliaisiais skaičiais nuo 1 iki n.
- matrica D gaunama nuosekliai apskaičiuojant matricas D^0, D^1,…, D^m, … , D^n.
- $D^m = [d_{ij}^m] \quad i = \overline{1,n}, \quad j = \overline{1,n}$ elementas reiškia ilgį trumpiausio kelio tarp i ir j viršūnių, kai tarpinėmis šio kelio viršūnėmis gali būti tik viršūnės su numeriais nuo 1 iki m.

Floido metodas

- Jei tarp viršūnių *i* ir *j* nurodyto tipo kelio nėra, tai $d_{ij}^{m} = \infty$
- Apibrėžkime matricą D$^{0.}$
- $$d_{ij}^{0} = \begin{cases} c(i,j), & jei \quad (i,j) \in U, \\ \infty, & jei \quad (i,j) \notin U, \\ 0, & jei \quad i = j, \end{cases}$$ visiems $i = \overline{1,n}, \quad j = \overline{1,n}.$

Floido metodas

- Kaip iš matricos D^{m-1}apskaičiuoti matricą D^m , m=1, 2, … , n ?



- $d_{ij}^{m} = \min\left(d_{ij}^{m-1}, d_{im}^{m-1} + d_{mj}^{m-1}\right)$

$$d_{ij}^{m} = \begin{cases} d_{ij}^{m-1}, & jei \quad d_{ij}^{m-1} \le d_{im}^{m-1} + d_{mj}^{m-1}, \\ d_{im}^{m-1} + d_{mj}^{m-1} & prie\check{s}\text{ingu atveju,} \end{cases}$$

Trumpiausias ilgis tarp i ir j

- Mus domina ne tik trumpiausio kelio tarp viršūnių *i* ir *j* ilgis, bet ir per kokias viršūnes šis kelias eina.
- Tam tikslui apibrėšime matricas $P^0, P^1, \ldots, P^m, \ldots, P^n$.
- P^m elementas p_{ij}^m reiškia numerį viršūnės, į kurią tiesiogiai trumpiausias kelias veda iš viršūnės *i* į viršūnę *j*.

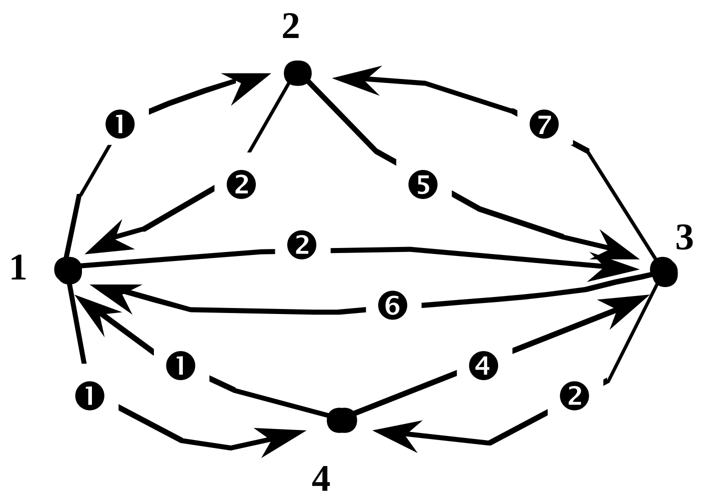
$$p_{ij}^0 = \begin{cases} j, & jei \quad (i, j) \in U, \\ 0, & jei \quad (i, j) \notin U, \end{cases}$$

$$p_{ij}^m = \begin{cases} p_{ij}^{m-1}, & jei \quad d_{ij}^{m-1} \le d_{im}^{m-1} + d_{mj}^{m-1}, \\ p_{im}^{m-1} & priešingu \ atveju, \end{cases}$$

Pseudokodas

- **For** i := 1 to n **do**
  - **For** j= 1 to n **do**
    - d[i][j] = inf
    - prec[i][j] = 0
  - d[i][i] = 0
- **For** (i,j) in U   *//Iteruojame per visas briaunas*
  - d[i][j] = w(i,j)
  - prec[i][j] = j
- **For** k := 1 to n **do**
  - **For** i := 1 to n **do**
    - **For** j := 1 to n **do**
      - **If** d[i][j] > d[i][k] + d[k][j] **then**
        - » d[i][v2] = d[i][k] + d[k][j]
        - » prec[i][j] = prec[i][k]

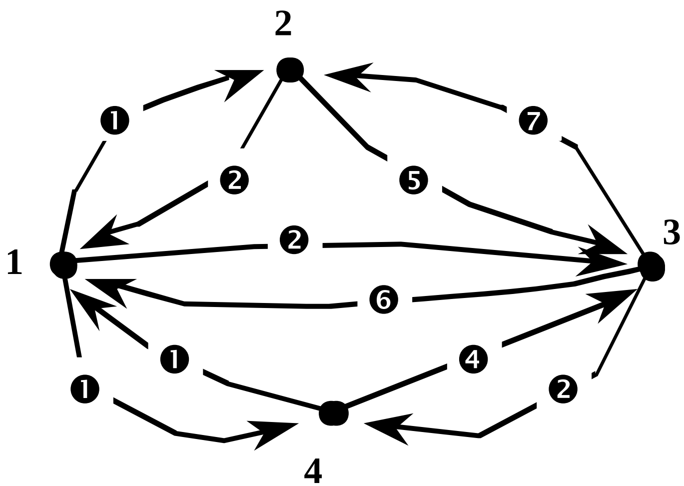
Pavyzdys – Pasiruošimas



D^0	1	2	3	4
1	0	1	2	1
2	2	0	5	inf
3	6	7	0	2
4	1	inf	4	4

P^0	1	2	3	4
1	0	2	3	4
2	1	0	3	0
3	1	2	0	4
4	1	0	3	0

Pavyzdys - k=1



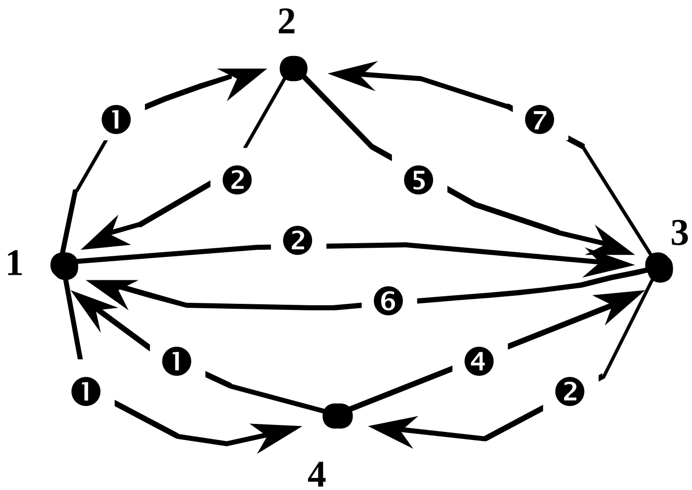
D^0	1	2	3	4
1	0	1	2	1
2	2	0	5	inf
3	6	7	0	2
4	1	inf	4	4

P^0	1	2	3	4
1	0	2	3	4
2	1	0	3	0
3	1	2	0	4
4	1	0	3	0

D^1	1	2	3	4
1	0	1	2	1
2	2	0	4	3
3	6	7	0	2
4	1	2	3	0

P^1	1	2	3	4
1	0	2	3	4
2	1	0	1	1
3	1	2	0	4
4	1	1	1	0

Pavyzdys - k=2



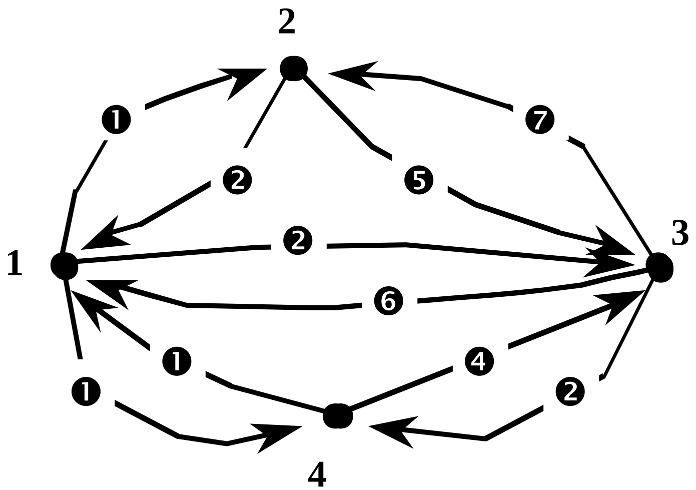
D^1	1	2	3	4
1	0	1	2	1
2	2	0	4	3
3	6	7	0	2
4	1	2	3	0

P^1	1	2	3	4
1	0	2	3	4
2	1	0	1	1
3	1	2	0	4
4	1	1	1	0

D^2	1	2	3	4
1	0	1	2	1
2	2	0	4	3
3	6	7	0	2
4	1	2	3	0

P^2	1	2	3	4
1	0	2	3	4
2	1	0	1	1
3	1	2	0	4
4	1	1	1	0

Pavyzdys - k=3



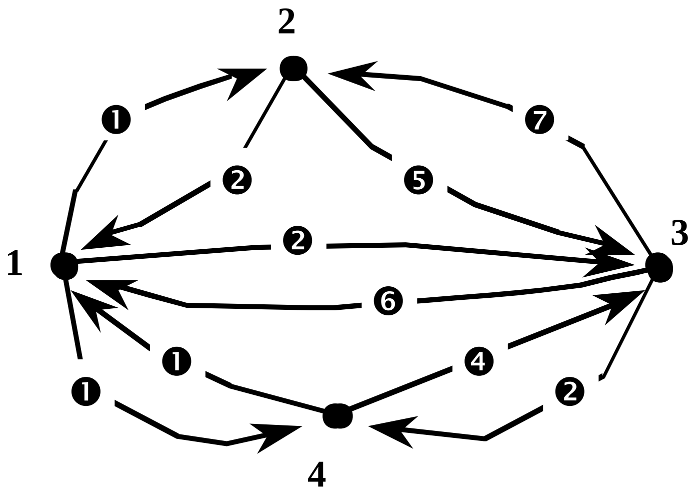
D^2	1	2	3	4
1	0	1	2	1
2	2	0	4	3
3	6	7	0	2
4	1	2	3	0

P^2	1	2	3	4
1	0	2	3	4
2	1	0	1	1
3	1	2	0	4
4	1	1	1	0

D^3	1	2	3	4
1	0	1	2	1
2	2	0	4	3
3	6	7	0	2
4	1	2	3	0

P^3	1	2	3	4
1	0	2	3	4
2	1	0	1	1
3	1	2	0	4
4	1	1	1	0

Pavyzdys - k=4



D^3	1	2	3	4
1	0	1	2	1
2	2	0	4	3
3	6	7	0	2
4	1	2	3	0

P^3	1	2	3	4
1	0	2	3	4
2	1	0	1	1
3	1	2	0	4
4	1	1	1	0

D^4	1	2	3	4
1	0	1	2	1
2	2	0	4	3
3	3	4	0	2
4	1	2	3	0

P^4	1	2	3	4
1	0	2	3	4
2	1	0	1	1
3	4	4	0	4
4	1	1	1	0

Uždavinys