

Reitinhaku verkossa

Johdanto

Tämän harjoitustyön aiheena on verkossa tapahtuvaan reitinhakuun liittyvät tietorakenteet ja algoritmit. Lähteenä on Helsingin yliopiston Tietojenkäsittelytieteen laitoksen kurssilla Tietorakenteet ja algoritmit käytetty luentomateriaali (Floréen P. 2013: [Tietorakenteet ja algoritmit](#)).

Ongelman kuvaus

Ratkaistavana ongelmana on löytää nopein reitti kahden pisteen välillä. Reitti valitaan kartalta, jota kuvaa kaksiulotteinen kokonaislukutaulukko. Taulukon alkiot vastaavat kartan alueita, joita kutsumme ruuduiksi. Kussakin ruudussa on vain yksi mahdollinen sijainti, joka on ruudun keskipiste.

Liikkuminen kartalla on mahdollista ruutu kerrallaan ylös, alas, vasemmalle ja oikealle. Naapuriruutuun liikkuminen kestää niin monta aikayksikköä kuin on kohteena olevaa ruutua vastaavan alkion arvo.

Taulukon alkiot ovat kokonaislukuja väliltä $[0, 9]$. Alkion arvo kuvaa ruudun kulkukelpoisuutta: mitä pienempi arvo, sitä nopeammin ruutuun pääsee. Poikkeuksena on arvo 0, joka tarkoittaa kulkukelvotonta estettä.

Olkoon ruudukossa m riviä ja n saraketta. Käytämme rivin i sarakkeessa j olevan ruudun sijainnista merkintää (i, j) , missä $i = 0, \dots, m - 1$ ja $j = 0, \dots, n - 1$. Merkitään lisäksi lähtöruutua s :llä ja maaliruutua t :llä.

Kuviossa 1 on esimerkki karttaa kuvaavasta taulukosta, jossa $m = 8$ ja $n = 13$. Lähtöruutu $s = (0, 7)$ ja maaliruutu $t = (7, 12)$. Lyhin reitti ruudusta s ruutuun t kulkee tässä tapauksessa pitkin ykkösen arvoisia ruutuja, ja on korostettu vihreällä värillä. Reitin kulkemiseen tarvittava kokonaisaika on 40 yksikköä. Huomaa, että lähtöruudun arvo voi olla 0, sillä lähtöruutuun ei reitinhaun alettua tarvitse enää siirtyä.

9	1	1	1	6	6	5	0	4	3	3	2	2
1	1	7	1	7	1	1	1	0	0	0	0	0
1	9	6	1	1	1	3	3	0	1	1	1	1
1	1	0	0	0	0	0	0	3	1	0	9	1
7	1	1	1	1	3	3	3	3	1	0	1	1
5	5	4	5	1	1	1	3	3	1	0	1	9
2	3	4	4	3	3	1	1	1	1	0	1	1
3	3	3	3	3	3	2	2	3	3	0	3	1

Kuvio 1: esimerkki karttaa kuvaavasta taulukosta

Ratkaisumenetelmät

Ongelman ratkaisemiseksi mallinnetaan tilanne reitinhauksi painotetussa verkossa, jonka solmuja kartan ruudut ovat. Yksittäisen solmun tunnuksena on solmua vastaavan ruudun sijainti: $v = (v_1, v_2)$, missä $v_1 = i$ ja $v_2 = j$. Solmun avaimena v_{key} käytetään taulukon alkia.

Solmusta $a = (a_1, a_2)$ on kaari solmuun $b = (b_1, b_2)$ jos ja vain jos seuraavat ehdot täyttyvät:

- (E1) $0 \leq a_1, b_1 \leq m - 1$ ja $0 \leq a_2, b_2 \leq n - 1$,
- (E2) joko ($|a_1 - b_1| = 0$ ja $|a_2 - b_2| = 1$) tai ($|a_1 - b_1| = 1$ ja $|a_2 - b_2| = 0$), ja
- (E3) $0 < b_{key}$.

Reitinhakuun verkossa käytetään Dijkstran algoritmia, jonka aputietorakenteena on minimikeko. Vertailun vuoksi ongelma ratkaistaan myös A*-algoritmillä minimikekoa käyttäen. A*:n heuristiikkafunktiona on $h(v) = \|v - t\|_1 = |v_1 - t_1| + |v_2 - t_2|$, missä v on nykyinen solmu ja t maalisolmu.

Olkoon verkkomme $G = (V, E)$ solmujen lukumäärä $|V|$ ja kaarien lukumäärä $|E|$. Tällöin Dijkstran algoritmin aikavaativuus on $O((|V| + |E|) \log |V|)$ ja tilavaativuus $O(|V|)$. A*:n pahimman tapauksen aikavaativuus on sama kuin Dijkstralla, $O((|V| + |E|) \log |V|)$.