Mincost-flow-algoritmit

Tuukka Korhonen

October 24, 2016

Mincost-flow

Annetaan virtausverkko $G = (V, E, U, C, s, t)$, jossa $U(e) \in \mathbb{Z}_+$ on kaaren e kapasiteetti, $C(e) \in \mathbb{R}$ on kaaren e hinta ja s ja t ovat lähtö- ja kohdesolmut. Annetaan lisäksi kokonaisluku k joka tarkoittaa kuinka paljon virtausta lähetetään. Merkinnällä uv tarkoitetaan kaarta solmusta u solmuun v. Oletetaan tämä merkintä yksikäsitteiseksi eli kahden solmun välillä ei saa olla useaa kaarta. Lisäksi oletetaan että jos verkossa on kaari uv, siinä ei ole kaarta vu. Nämä oletukset ovat vain teoriaa varten ja niitä ei tehdä implementaatiossa. Mincost-flow-ongelmassa tavoitteena on löytää virtaus jota nyt merkitään funktiona $f : E \to \mathbb{Z}$ joka:

Minimoi

$$\sum_{uv \in E} C(uv) f(uv)$$

Toteuttaa ehdot:
Säilyvyysehto

$$\sum_{v \in V} f(uv) = \sum_{v \in V} f(vu) \text{ kaikilla } u \in V \setminus \{s, t\}$$

Kapasiteettiehto

$$0 \leq f(uv) \leq U(uv) \text{ kaikilla } uv \in E$$

Virtauksen määrä

$$\sum_{v \in V} f(sv) = k \text{ ja } \sum_{v \in V} f(vt) = k$$

Lisäksi määritellään että Mincost-flow-ongelmassa verkossa ei saa olla suunnattuja syklejä, joiden hintojen summa on negatiivinen.

Toisin sanoen lähetetään k yksikköä virtausta lähtösolmusta s kohdesolmuun t niin että virtauksen käyttämien kaarien hinta on mahdollisimman pieni. Tämän voi ajatella esimerkiksi ongelmana jossa verkon solmut ovat kaupunkeja. Kaupungissa s on tehdas josta pitää lähettää joka päivä k tonnia tavaraa varastoon joka on kaupungissa t. Lisäksi tiedetään että kaupunkien välillä on junayhteyksiä (eli kaaria). Kaaren uv kapasiteetti kertoo paljonko sillä junayhteydellä voi kuljettaa tavaraa päivässä kaupungista u kaupunkiin v. Kaarien hinnat kertovat paljonko maksaa yhden tonnin tavaraa kuljettaminen sillä junayhteydellä. Nyt halutaan kuljettaa k tonnia tavaraa tehtaasta varastoon päivittäin ja minimoida kuljetuksen hinta.

Syy negatiivisten syklien kieltämiseen on, että saadaan yksinkertaisempia toimivia algoritmeja ja tämä määritelmä sopii ongelman luonteeseen jossa lähetetään tavaraa lähtösolmusta kohdesolmuun. Jos negatiivisia syklejä saisi olla, optimaalinen ratkaisu voisi pyörittää virtausta jossain muualla kuin reiteillä lähtösolmusta kohdesolmuun. Myöhemmin esitetään MINCOST CIRCULATION ongelma, jossa negatiiviset syklit ovat sallittuja.

Shortest-Augmenting-Path-algoritmi

Shortest-Augmenting-Path-algoritmi (lyhyemmin SAP-algoritmi) etsii minimihintaisen virtauksen lähettämällä virtausta lyhintä reittiä kaarien hintojen suhteen lähtösolmusta kohdesolmuun. Tällaista lyhintä reittiä kutsutaan *augmentoivaksi poluksi.* Algoritmi on muunnelma Ford-Fulkerson-algoritmista maksimivirtauksen etsimiseen. Intuitiota saa miettimällä miten minimihintainen virtaus etsittäisiin jos virtausta lähetettäisiin vain yksi yksikkö: se olisi lyhin reitti lähtösolmusta kohdesolmuun.

Jäännösverkko

Jotta virtausta voidaan lähettää useampia yksiköitä, täytyy olla tapa korjata aiemmin lähetettyä virtausta. Lähetetään virtausta siis jäännösverkossa. Jäännösverkko toimii kuten Ford-Fulkersonin jäännösverkko: virtausta voidaan peruuttaa kulkemalla jäännösverkossa vastakkaiseen suuntaan menevää kaarta pitkin. Virtausta peruuttaessa kaaren hinta on alkuperäisen kaaren hinnan vastaluku. Tästä seuraa että negatiivisia kaaria syntyy jäännösverkkoon vaikka niitä ei olisi verkossa alunperin.

Sanotaan että G_f on virtauksen f jäännösverkko. Määritellään kapasiteetit jäännösverkossa:

$$U_f(uv) = \begin{cases} U(uv) - f(uv) \text{ jos } uv \in E \\ f(vu) \text{ jos } vu \in E \\ 0 \text{ muuten} \end{cases}$$

Määritellään hinnat jäännösverkossa:

$$C_f(uv) = \begin{cases} C(uv) \text{ jos } uv \in E \\ -C(vu) \text{ jos } vu \in E \\ 0 \text{ muuten} \end{cases}$$

Sanotaan että kaari uv on jäännösverkossa jos $U_f(uv) > 0$.

Jos jäännösverkossa on polku $s \rightsquigarrow t$ jonka pienin kapasiteetti on u, niin sitä polkua pitkin voidaan lähettää maksimissaan u yksikköä virtausta ja yhden lähetetyn virtausyksikön hinnaksi tulee polun kaarien hintojen summa. Tämä kasvattaa virtauksen määrää. Toinen sallittu tapa muuttaa virtausta on lähettää virtausta jotain jäännösverkon sykliä pitkin, maksimissaam syklin pienimmän kapasiteetin verran. Laskemalla voidaan tarkistaa että nämä muutokset säilyttävät virtauksen säilyvyys- ja kapasiteettiehdot.

SAP-algoritmi

SAP-algoritmi lähettää virtausta jäännösverkon lyhintä $s \rightsquigarrow t$-polkua pitkin kunnes virtausta on lähetetty k yksikköä tai maksimivirtaus on löydytty. Tässä lyhin polku tarkoittaa lyhintä polkua jäännösverkon hintojen suhteen. Lähettämällä virtausta aina lyhintä polkua pitkin jäännösverkkoon ei koskaan synny negatiivisia syklejä. (Negatiivinen sykli on sykli jonka kaarien hintojen summa on negatiivinen.) Tästä seuraa myös tapa nähdä että virtaus on optimaalinen: ainoa tapa muuttaa virtauksen hintaa muuttamatta virtaukseen määrää on lähettää virtausta jotain sykliä pitkin. Jos negatiivisia syklejä ei ole, virtauksen hintaa ei voi muuttaa mitenkään pienemmäksi. Seuraavaksi todistetaan nämä väitteet.

Virtauksen optimaalisuus

Lemma: k-virtaus on optimaalinen jos jäännösverkossa ei ole sykliä jonka hinta on negatiivinen.

Todistus: Olkoon f joku k-virtaus niin että G_f:ssä ei ole negatiivista sykliä. Olkoon f' joku k-virtaus jonka hinta on pienempi kuin f:n hinta. Määritellään $(f' - f)(uv) = f'(uv) - f(uv)$. Nyt $f' - f$ on virtaus joka toteuttaa säilyvyysehdon kaikissa solmuissa, mukaan lukien lähtö- ja kohdesolmut, koska f' ja f ovat molemmat k-virtauksia. Virtauksen $f' - f$ hinta on negatiivinen. Lisäksi $f' - f$ on sallittu virtaus G_f:ssä. Säilyvyysehdon toteuttavan virtauksen voi hajottaa joukoksi syklejä jotka toteuttavat säilyvyysehdon, koska jos otetaan mikä tahansa sykli virtauksesta pois, jäljelle jää virtaus joka toteuttaa säilyvyysehdon. Siispä kun $f' - f$ hajotetaan joukoksi syklejä, niin ainakin yhden syklin hinta on negatiivinen. Sitä sykliä pitkin voidaan lähettää virtausta G_f:ssä, joten G_f:ssä on negatiivinen sykli. Seuraa ristiriita, joten f on optimaalinen.

Potentiaalifunktio

Sanotaan että potentiaalifunktio jäännösverkossa G_f on funktio $p : V \to \mathbb{R}$, joka toteuttaa $p(u) + C_f(uv) \geq p(v)$ kaikilla jäännösverkon kaarilla uv. Jos G_f:ssä ei ole negatiivista sykliä, potentiaalifunktio on olemassa, sillä potentiaalifunktioksi voidaan ottaa $p(u)$ on lyhimmän reitin pituus solmusta s solmuun u. Määritellään että kaaren uv *vähennetty hinta* on $C_{fr}(uv) = p(u) + C_f(uv) - p(v)$. Nähdään että syklin hinta vähennetyissä hinnoissa on sama kuin syklin hinta koska potentiaalit mitätöivät toisensa. Nähdään myös että kaikki vähennetyt hinnat ovat epänegatiivisia, josta on helppo nähdä että jos on olemassa potentiaalifunktio, niin negatiivisia syklejä ei ole.

Lemma: Jos jäännösverkossa ei ole negatiivista sykliä, niin lähettämällä virtausta lyhintä $s \rightsquigarrow t$ polkua pitkin sinne ei synny negatiivista sykliä.

Todistus: Kun lähetetään virtausta lyhintä $s \rightsquigarrow t$ polkua pitkin, voidaan olemassaolevasta potentiaalifunktiosta p päivittää potentiaalifunktio p' joka on pätevä uudessa jäännösverkossa. Olkoon $sp(u)$ lyhimmän reitin pituus solmusta s solmuun u kun tarkastellaan reitin hintaa vähennetyissä hinnoissa. Jos ei ole polkua solmusta s solmuun u, niin $sp(u)$ on joku riittävän suuri vakio. Uusi potentiaalifunktio on $p'(u) = p(u) + sp(u)$. Tämä toteuttaa potentiaalifunktion ehdot kaikilla kaarilla jotka olivat jo jäännösverkossa koska $sp(u) + C_{fr}(uv) \geq sp(v)$ koska sp on lyhin-reitti-funktio joten $sp(u) + p(u) + C_f(uv) - p(v) \geq sp(v)$ joten $sp(u) + p(u) + C_f(uv) \geq sp(v) + p(v)$ joten $p'(u) + C_f(uv) \geq p'(v)$. Lisäksi verkkoon saattoi ilmestyä uusia kaaria kun lyhintä $s \rightsquigarrow t$ polkua päinvastaiseen suuntaan menevien kaarien kapasiteetti kasvoi. Kun u ja v ovat lyhimmällä polulla peräkkäin olevat solmut, $sp(u) + C_{fr}(uv) = sp(v)$ joten

$p'(u) + C_f(uv) = p'(v)$ joten $p'(v) - C_f(uv) = p'(u)$ eli potentiaalifunktio toimii kaikilla uusilla kaarilla jotka ilmestyivät jäännösverkkoon.

Aikavaativuus

Tässä määritelty SAP-algoritmi ei anna mitään tietoa tarvittavien augmentoivien polkujen määrästä. Nähdään kuitenkin että virtauksen määrä kasvaa ainakin yhdellä, joten augmentoivia polkuja on maksimissaan k. Periaatteessa virtausta voi lähettää korkeintaan Um yksikköä, jossa U on verkon suurin kapasiteetti. Lyhyimmän reitin etsiminen ei ole suoraviivaista koska verkossa on negatiivisia kaaria. Lyhyimmän reitin verkossa jossa on negatiivisia kaaria voi etsiä Bellman-Ford-algoritmilla, jonka aikavaativuus on $O(nm)$. Kokonaisaikavaativuudeksi tulee $O(Um^2n)$. Käytännössä käytetään SPFA-algoritmia, joka on käytännössä hyvin nopeasti toimiva versio Bellman-Fordista.

Dijkstran algoritmin käyttö

Dijkstran algoritmia ei voi käyttää ongelmaan suoraan koska verkossa on kaaria joiden hinta on negatiivinen. Kuitenkin aiemmin esitellyn potentiaalifunktion avulla voidaan muuttaa kaikkien kaarien hinnat epänegatiivisiksi. Vaikka jäännösverkko muuttuu, potentiaalifunktion voi päivittää helposti päteväksi käyttämällä lyhimpiä polkuja jotka kuitenkin lasketaan kun lähetetään virtausta. Jos jäännösverkossa olevan $s \rightsquigarrow t$-polun pituus on L, niin sen pituus vähennetyissä hinnoissa on $L + p(s) - p(t)$, koska kaikki muut potentiaalit mitätöivät toisensa. Seuraa että lyhimmän $s \rightsquigarrow t$ polun voi yhtä hyvin hakea vähennettyjen hintojen verkossa.
Saadaan nopeampi algoritmi: Etsitään ensin joku potentiaalifunktio Bellman-Ford-algoritmilla. Sen jälkeen käytetään Dijkstran algoritmia vähennettyjen hintojen verkossa augmentoivien polkujen löytämiseen ja potentiaalifunktion päivittämiseen. Algoritmin aikavaativuus on
$O(nm + Um^2 \log n)$.