

Ellinoora Vikman

VAIKUTUKSEN MAKSIMOINTI SOSIAALISESSA VERKOSTOSSA

Kandidaatintyö
Tekniikan ja luonnontieteiden tiedekunta
Tarkastaja: Henri Hansen
Toukokuu 2022

TIIVISTELMÄ

Ellinoora Vikman: Vaikutuksen maksimointi sosiaalisessa verkostossa
Kandidaatintyö
Tampereen yliopisto
Tekniikka ja luonnontieteet, TkK
Toukokuu 2022

Vaikutuksen maksimointi sosiaalisessa verkostossa on hyvin laajasti ja pitkään tutkittu ongelma, jonka merkitys on sosiaalisen median käytön yleistyessä korostunut. Ongelma on hyvin tärkeä erityisesti markkinatalouden, mutta myös poliittisen vaikuttamisen näkökulmasta. Ongelman tutkiminen ja ratkaisutapojen kehittäminen antavat keinoja kokonaisvaltaiselle ja tehokkaalle vaikuttamiselle. Tämän työn tavoitteena on selvittää, mitä vaikutuksen maksimointi sosiaalisessa verkostossa tarkoittaa ja miten ongelma voidaan määrittellä matemaattisesti. Työn lopussa perehdytään ongelman ratkaisuympäristöön ja sen vaatavuustasoon. Tutkielmaan perehtyvältä odotetaan matematiikan perusasioiden tuntemusta sekä matemaattisten yliopistoalojen ensimmäisten vuosien kerryttämää matemaattista osaamista.

Työn ensimmäisessä osiossa käydään lyhyesti läpi graafiteoriaa sekä esitellään yksi tiedon leviämisen malli *Independent Cascade*. Toisessa osiossa määritellään työssä tutkittava ongelma matemaattisesti sekä todistetaan vaikutusfunktion submodulaarisuus ja monotonisuus. Submodulaarisuuden todistamista varten osiossa esitellään *Independent Cascade* -malliin liittyvä *live-edge* -malli. Lopuksi tarkastellaan ongelman ratkaisuympäristöä muun muassa *Greedy*-algoritmin kautta.

Vaikutuksen maksimointi sosiaalisessa verkostossa tarkoittaa yksinkertaisuudessaan sitä, että verkostosta pyritään löytämään sellainen jollain tekijällä rajoitettu joukko ihmisiä tai sosiaalisen median sovellusten tapauksessa käyttäjiä, joiden levittämän tiedon avulla verkostossa saadaan enimmäismäärä ihmisiä muuttamaan mieltään tai käyttäytymistään haluttuun suuntaan. Ongelmaa tarkastellaan tässä työssä *Independent Cascade* -mallin avulla.

Tuloksina saadaan ongelman NP-täydellisyys sekä rajallinen algoritmien tehokkuus. Ratkaisun etsivä algoritmi pätee hyvin kohtuullisen kokoisille graafeille, mutta mitä suurempia graafeja käsitellään, sitä huonompi on algoritmien tehokkuus.

Avainsanat: sosiaalinen verkosto, vaikutuksen maksimointi, graafiteoria, *Independent Cascade*, vaikutusfunktio

Tämän julkaisun alkuperäisyys on tarkastettu Turnitin OriginalityCheck -ohjelmalla.

SISÄLLYSLUETTELO

1.	Johdanto	1
2.	Sosiaalinen verkosto -graafiteoria	3
2.1	Yleistä graafeista	3
2.2	Independent Cascade -malli	5
3.	Vaikutuksen maksimointi -ongelma	9
3.1	Vaikutuksen maksimointi IC-mallin avulla	9
3.2	Vaikutusfunktion submodulaarisuus	11
3.3	Algoritmit	13
4.	Yhteenveto	15
	Lähteet.	17

1. JOHDANTO

Tiedon leviäminen verkostoissa on laajasti tutkittu ilmiö ja sen kuvaamiseen on kehitetty useita erilaisia matemaattisia malleja. Oikeanlaisen mallin valinta sovelluskohteen tulkintaa varten riippuu erityisesti verkoston rakenteesta ja tiedon leviämisen todellisesta käyttäytymisestä. Erilaisia todellisen maailman verkostoja on huomattava määrä ja siksi onkin tärkeää tehdä ero tässä työssä tarkasteltavan sosiaalisen verkoston ja muiden verkostojen välillä.

Sosiaalisessa verkostossa *solmuina* ovat ihmiset ja *kaarina* ihmisten väliset yhteydet, kuten kaverisuhteet, sukulaissuhteet tai vaikkapa työsuhteet. Sosiaalisessa verkostossa tieto leviää ihmisten toiminnan seurauksena muun muassa suullisesti tai viestikanavien välityksellä. Sosiaalinen verkosto eroaa esimerkiksi kartastosta hyvin monella tapaa. Kartastoa voidaan ajatella vaikkapa siten, että solmuina ovat kaupungit ja kaarina tiet kaupunkien välillä. Kartastossa tiedon kulkeutuminen voisi olla vaikka autojen siirtyminen eri kaupunkien välillä. Toisaalta myös sosiaaliset verkostot voidaan luokitella esimerkiksi sosiaalisen median sisältämiin verkostoihin ja reaali maailman verkostoihin. Tiedon leviäminen sosiaalisessa mediassa on huomattavasti helpompaa ja nopeampaa. Keskitytään tässä työssä tarkastelemaan juuri sosiaalisen median verkostoja.

Ongelma, jota työssä tarkastellaan, on *vaikutuksen maksimointi sosiaalisessa verkostossa*. Vaikutus tarkoittaa yksinkertaisuudessaan sitä, että tiedon leviämisen seurauksena verkostossa tapahtuu ajatus- tai käyttäytymismallin muutoksia. Tätä ilmiötä havainnollistaa hyvin yritysmaailmasta tuttu käsite markkinointiviestintä. Ulkoisiin sidosryhmiin kohdistuvan markkinointiviestinnän tarkoitus on saada aikaan kysyntää tai kysyntään myönteisesti vaikuttavia ilmiöitä [7]. Vaikutuksen maksimoinnin avulla kohderyhmässä vallitsevaa myönteistä mielipidettä kyseessä olevaa palvelua tai tuotetta kohtaan voidaan lisätä.

Ongelman ratkaisut pyritään löytämään sellaisten algoritmien avulla, jotka etsivät kaikista graafin solmuista ne, jotka maksimoivat vaikutuksen graafissa. Kaksi keskeisintä ongelmaan liittyvää haastetta ovat siis: "Miten kehittää sellainen malli, jolla voidaan kuvata vaikutuksen leviämistä sosiaalisessa verkostossa?" ja "Miten kehittää tehokas algoritmi, joka löytää mahdollisimman optimaalisen ratkaisun ongelmalle?" Ongelmalle onkin saatu kehitettyjä algoritmeja, jotka kykenevät löytämään ratkaisun kohtuullisen kokoiselle verkostolle. Kuitenkaan vielä ei ole löydetty algoritmia, joka toimisi yhtä tehokkaasti myös paljon suuremmille graafeille. Kymmenien tuhansien solmujen graafeille 50 solmun syötejoukon löytäminen vie uusimmillakin tietokoneilla tunteja [13]. Lisäksi ongelman NP-täydellisyyden vuoksi jokainen ratkaisualgoritmi tuottaa vain likimääräisen arvion ongelman ratkaisulle.

Työssä käytetään useita eri lähteitä liittyen vaikutuksen maksimointiin, sosiaalisiin verkostoihin

sekä graafiteoriaan. Tärkeimmät työssä käytetyt lähteet ovat Kempe *et al.* kirjoittama artikkeli *Maximizing the spread of influence through a social network (2003)*, Lu *et al.* kirjoittama artikkeli *Scalable influence maximization under independent cascade model (2003)* sekä Shakarian *et al.* kirjoittama kirja *Diffusion in Social Networks (2015)*. Lisäksi työssä on käytetty muita aiheeseen liittyviä artikkeleita ja kirjoja, joitain enemmän ja joitain vain yhden määritelmän verran.

2. SOSIAALINEN VERKOSTO -GRAAFITEORIA

Verkko -eli graafiteoria on matematiikan osa-alue, joka pyrkii mallintamaan asioiden välisiä suhteita erilaisissa verkoissa. Verkko tai verkosto määritellään tietorakennetyyppinä, joka koostuu yhteen liitettyistä järjestelmistä. Esitellään tässä luvussa perustietoa graafeista sekä yksi tavallisesti käytettävä tiedon leviämisen malli *Independent Cascade*.

2.1 Yleistä graafeista

Graafi on pari $G = (V, E)$, missä V on joukko *solmuja* ja E joukko *kaaria*. Jokainen kaari on solmuista muodostuva pari, jonka alkiot ovat kaaren päätepisteitä. Jos parit ovat järjestettyjä, puhutaan suunnatusta ja muussa tapauksessa suuntaamattomasta graafista. Määritellään tässä työssä graafi suunnattuna graafina siten, että kunkin kaaren $e = (v, u) \in E$ suunta on solmusta v solmuun u . Tällöin $(v, u) \neq (u, v)$. Solmujen lisäksi myös kaarissa voi olla dataa, jota kutsutaan kaaren painoksi. [12, s. 1]

Havainnollisesti graafi voidaan esittää diagrammina, jossa solmut ovat pisteitä tai ympyröitä ja kaaret pisteitä tai ympyröitä yhdistäviä viivoja. Suunnatun graafin tapauksessa viivat voidaan esittää nuolina. [12, s.2]

Määritellään seuraavaksi asian hahmottamisen kannalta muutamia tärkeitä seikkoja graafeihin liittyen.

Määritelmä 2.1. [12] *Graafi* on pari (V, E) , missä V on joukko *solmuja* ja $E \subseteq V \times V$ joukko *kaaria*. Jos on lisäksi annettu kuvaus $w : E \rightarrow \mathbb{R}$, kutsutaan kolmikkoa (V, E, w) *painotetuksi graafiksi*. Suuretta $w(e)$ kutsutaan kaaren e *painoksi*.

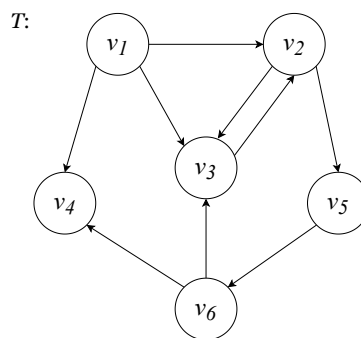
Määritelmä 2.2. [12, s. 1] *Silmukka* on kaari, joka yhdistää solmun siihen itseensä. Esimerkiksi kaari $e = (v, v)$ on silmukka solmussa v .

Määritelmä 2.3. [12, s. 1–2] Solmun *aste* on solmuun liittyvien kaarien määrä. Jos aste on 0, solmua sanotaan *erilliseksi*.

Määritelmä 2.4. [1, s. 29] Graafissa G *polku* on sellainen jono solmuja $\{v_0, v_1, \dots, v_n\}$, että jokaiselle luvulle $i = 0, 1, 2, \dots, n$ pätee $v_i \in V$ ja lisäksi jokaiselle luvulle $i = 1, 2, 3, \dots, n$ pätee $(v_{i-1}, v_i) \in E$.

Sosiaalinen verkosto on yksi käytännön esimerkki siitä, mihin graafiteoriaa voidaan soveltaa. Sosiaalinen verkosto voidaan ajatella graafina, jonka solmuina ovat ihmiset ja kaarina ihmisten väliset sosiaaliset yhteydet. Sosiaalisia verkostoja voivat olla esimerkiksi sosiaalisen median palveluissa luodut kaveri- tai seuraajaverkostot, maantieteellisten valtioiden väliset sosiaaliset verkostot tai työpaikkojen työntekijäverkostot. Tarkastellaan tässä työssä sosiaalista verkostoa, joka pohjautuu Twitter-palvelun seuraussuhteisiin. Twitter-palvelun seuraajaverkosto on suunnattu graafi, sillä seuraajasuhteet ovat yksisuuntaisia. Käyttäjä saattaa seurata toista ilman, että häntä seurataan takaisin. On olemassa myös verkostoja, kuten Facebook, joissa käyttäjien väliset suhteet ovat aina molempiin suuntiin. Tällöin kyseessä on suuntaamaton graafi. [9]

Otetaan esimerkiksi satunnaisesti valittu graafi T , jonka solmut ovat Twitter-palvelun käyttäjiä ja kaaret seuraajasuhteita. Graafi on esitetty kuvassa 2.1.



Kuva 2.1. Piirros esimerkkgraafista T

Graafi T muodostuu siis kuudesta solmusta v_1, v_2, v_3, v_4, v_5 ja v_6 . Kuvasta 2.1 nähdään solmujen väliset suunnatut kaaret, jotka on esitetty nuolilla. Nuolen suunta kertoo kuka vaikuttaa kehenkin. Twitterin tapauksessa seurattava käyttäjä vaikuttaa seuraajaan. Graafi T kertoo esimerkiksi sen, että käyttäjät v_2 ja v_3 seuraavat toisiaan ja käyttäjä v_2 seuraa käyttäjää v_1 , mutta käyttäjä v_1 ei seuraa käyttäjää v_2 .

Määritelmä 2.5. [14, s. 22] Solmun $v \in V$ naapureiden joukko $N(v)$ saadaan määrittelemällä kuvaus $N : V \rightarrow 2^V$ siten, että $N(v) = \{u \mid (v, u) \in E \text{ tai } (u, v) \in E\}$.

Tarkastellaan esimerkin vuoksi edellä määriteltyä graafia T . Määritelmän 2.5 mukaan solmun v_3 naapureiden joukko on $N(v_3) = \{v_1, v_2, v_6\}$ ja vastaavasti solmun v_1 naapureiden joukko on $N(v_1) = \{v_2, v_3, v_4\}$.

Määritelmä 2.6. [14, s. 23] Määritellään erikseen solmuun tulevat ja solmusta lähtevät naapurit. Tulevien naapureiden joukko $N_{in}(v)$ saadaan määrittelemällä kuvaus $N_{in} : V \rightarrow 2^V$ siten, että $N_{in}(v) = \{u \mid (u, v) \in E\}$. Vastaavasti lähtevien naapureiden joukko $N_{out}(v)$ saadaan määrittelemällä kuvaus $N_{out} : V \rightarrow 2^V$ siten, että $N_{out}(v) = \{u \mid (v, u) \in E\}$.

Määritelmän 2.6 perusteella graafin T solmuun v_3 tulevien naapureiden joukko on $N_{in}(v_3) = \{v_1, v_2, v_6\}$ ja vastaavasti solmusta v_3 lähtevien naapureiden joukko on $N_{out}(v_3) = \{v_2\}$.

Määritelmä 2.7. [2, Määritelmä D.1.4, s. 121]. Olkoon $G = (V, E)$ graafi ja $G' = (V', E')$ toinen graafi. G' on graafin G *aligraafi*, jos ja vain jos $V' \subseteq V$ ja $E' \subseteq E$.

2.2 Independent Cascade -malli

Independent Cascade (IC) on diffuusiomalli, jota käytetään paljon tiedon leviämiseen liittyvissä ongelmissa. Sen lisäksi on muitakin yleisesti käytettyjä malleja, kuten esimerkiksi *Linear Threshold*, mutta perehdytään tässä työssä tarkemmin vain IC -malliin. [14]

IC-mallissa sosiaalinen verkosto esitetään suunnattuna *vaikutusgraagina* (*influence graph*). Vaikutusgraafissa kaaren suunta kertoo kuka vaikuttaa kehenkin. Kaaren lähtösolmulla on vaikutus kohdesolmuun. Toisin sanoen solmulla on vaikutus sen lähteisiin naapureihin ja toisaalta solmuun kohdistuu vaikuttamista sen tulevilta naapureilta. Twitter-palvelun tapauksessa vaikutusgraafina voidaan käyttää suoraan verkoston rakennetta, sillä seuraajasuhteet ovat yksisuuntaisia. Suuntaamattomien graafien tapauksessa vaikutusgraafi eroaa verkoston rakenteesta siten, että solmujen välillä oleva suuntaamaton kaari erotellaan kahdeksi vastakkaisiin suuntiin kulkeväksi kaareksi. [14, s. 21]

Lähteen [14, s. 22] mukaan vaikutusgraafin solmuilla on kolme mahdollista tilaa: *epäaktiivinen*, *positiivinen* ja *negatiivinen*, joista *aktiivisia* tiloja ovat positiivinen ja negatiivinen. Yksinkertaistetaan ongelmaa kuitenkin tässä työssä siten, että määritetään solmulle vain kaksi mahdollista tilaa: epäaktiivinen (I) ja aktiivinen (A).

Määritelmä 2.8. [14, s. 22] *Solmun tila* $M(v)$ saadaan kuvauksesta $M : V \rightarrow \{A, I\}$ siten, että $M(v) \in \{A, I\}$.

Jaetaan aktiiviset solmut vielä kahdeksi erilliseksi joukoksi SIR -mallin [3] mukaan: juuri aktivoituneiden solmujen joukoksi, jonka solmut ovat kykeneviä tartuttamaan muita solmuja, ja aikaisemmin aktivoituneiden solmujen joukoksi, jonka solmut eivät enää tartuta muita solmuja. Tämä malli noudattaa siis muun muassa lähteessä [3] mainittua *Susceptible-Infected-Recovered* (SIR) -mallia eli suomennettuna Altis - Tartutettu - Toipunut -mallia. Malli jakaa graafin solmut kolmeen erilliseen joukkoon, missä ensimmäinen joukko S (*susceptible*) on tartunnalle alttiiden eli toisin sanoen solmujen joukko, toinen joukko I (*infected*) on juuri tartunnan saaneiden solmujen joukko, jonka solmut kykenevät vielä vaikuttamaan muihin solmuihin, ja kolmas joukko R (*recovered*) on aikaisemmin aktivoituneiden solmujen joukko, jonka solmut eivät enää vaikuta muihin solmuihin.

Merkitään tässä työssä graafin kaikkien juuri aktivoituneiden solmujen joukkoa H , muiden aktiivisten eli toipunneiden solmujen joukkoa R ja kaikkien epäaktiivisten solmujen joukkoa S . Merkitään näitä joukkoja ajanhetkellä t : H_t , R_t ja S_t .

Tiedon leviäminen IC-mallissa tapahtuu solmujen *aktivoitumisten* kautta. Kun juuri aktivoituneella solmulla $v \in H$ on vähintään yksi naapurisolmu $u \in N_{out}(v)$, se voi laittaa tiedon leviämään naapurilleen, joka näin altistuu vaikutukselle. Jos altistuminen johtaa naapurisolmun aktivoitumiseen, puhutaan *tartunnasta*. Se, miten tieto leviää ja mitä aktivoituminen tarkoittaa riippuu sovelluskohteesta. Sosiaalisessa mediassa tieto voi levitä esimerkiksi julkaisujen kautta, jolloin julkaisijan

seuraajat altistuvat julkaisun sisältämälle tiedolle. Jos seuraajat sitten muuttavat käyttäytymistään tai ajatuksiaan tiedon perusteella, voidaan sanoa, että seuraajat saavat tartunnan ja näin aktivoituvat graafissa. Esimerkiksi jonkin taudin tartuntaverkostoja tutkittaessa aktivoituminen voisi olla taudin tarttuminen henkilöstä toiseen. Toisaalta sosiaalisen median kanavissa aktivoituminen voitaisiin nähdä vaikkapa vaikuttavan käyttäjän julkaisun jakamisena. [4]

IC-mallissa diffuusioprosessi etenee diskreetteinä aika-askelina $t = 0, 1, 2, 3, \dots$. Diffuusioprosessin edetessä aktiivisilla solmuilla on mahdollisuus aktivoida niiden epäaktiiviset lähtevät naapurinsa. [10, s. 35] Toisaalta epäaktiivisilla solmuilla on mahdollisuus aktivoitua niiden tulevien naapureiden vaikuttamina.

Seuraavat määritelmät ja tulokset on pienin muutoksin otettu lähteestä [10].

Määritelmä 2.9. *Syötejoukko* H_0 on kaikkien aktiivisten solmujen joukko graafissa $G = (V, E)$ ajanhetkellä $t = 0$.

Oletetaan, että solmu voi prosessin aikana muuttua epäaktiivisesta aktiiviseksi, mutta ei toiseen suuntaan. Tällöin puhutaan *progressiivisesta* tai *monotonisesta* prosessista. Ei-progressiivisessä prosessissa aktiiviset solmut voivat muuttua takaisin epäaktiivisiksi.

Diffuusioprosessi aloitetaan valitsemalla jokin syötejoukko H_0 . Sen jälkeen jokainen juuri aktivoitunut solmu $v \in H$ yrittää vaikuttaa epäaktiivisiin lähteviin naapureihinsa. Jokaisella solmulla v on vain yksi mahdollisuus vaikuttaa kuhunkin lähtevään naapuriinsa. Vaikutusyrityksen seurauksena solmu $u \in N_{out}(v)$ joko aktivoituu tai ei aktivoitu. Prosessi jatkuu, kunnes uusia aktivoitumisia ei enää tapahdu.

IC-mallissa vaikutusgraafi on painotettu graafi $G = (V, E, w)$, jossa jokaisen kaaren $e = (u, v) \in E$ paino $w(e)$ on todennäköisyys, jolla solmu u aktivoi solmun v .

Määritelmä 2.10. Olkoon H_t ajanhetkellä t tartuttavien solmujen joukko. Ajanhetkellä t jokainen solmu $v \in H_t$ tartuttaa epäaktiiviset naapurinsa $u \in N_{out}(v)$ *todennäköisyydellä* $w(u, v)$.

Määritelmä 2.11. Olkoon $G = (E, V, w)$ painotettu graafi. Graafi G on vaikutusgraafi jos ja vain jos jokaisen kaaren (u, v) painolle $w(u, v)$ pätee $0 \leq w(u, v) \leq 1$.

Todennäköisyysfunktio w voi määräytyä tilanteesta ja aktivoitumistavasta riippuen esimerkiksi solmujen välisen vuorovaikutuksen tai maantieteellisen etäisyyden perusteella [10, s. 34]. Jos aktivoituminen tarkoittaa esimerkiksi taudin tarttumista henkilöstä toiseen, on maantieteellisellä etäisyydellä suuri vaikutus tartunnan todennäköisyyteen. Jos henkilöt eivät ikinä kohtaa fyysisesti, on todennäköisyys sille, että toinen toinen henkilö tartuttaa toisen 0. Toisaalta jos henkilöt esimerkiksi työskentelevät samassa tilassa päivittäin, on tartunnan mahdollisuus olemassa. Tällöin todennäköisyysfunktio w määriteltäessä, otetaan huomioon henkilöiden välinen fyysinen etäisyys.

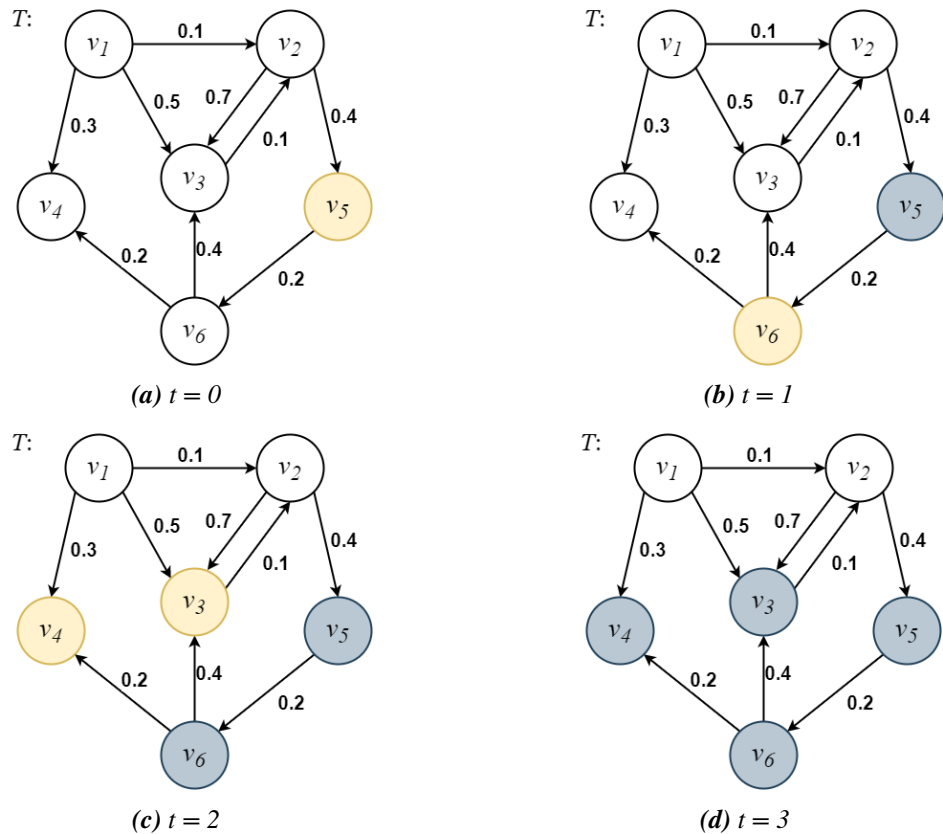
Määritetään pseudokoodina algoritmi, joka kuvaa IC-prosessin etenemistä graafissa $G = (V, E, w)$. Algoritmissa R_i on kaikkien toipuneiden solmujen joukko, H_i kaikkien tartuttavien solmujen ja S_i kaikkien epäaktiivisten solmujen joukko graafissa G ajanhetkellä i .

Algorithm 1 Independent Cascade

```

1:  $i = 0$ 
2: annettu  $H_0$ 
3: while  $H_i \neq \emptyset$  do
4:   for  $v \in H_i$  do
5:     for  $u \in N_{out}(v)$  do
6:       if  $u \in S_i$  then
7:         todennäköisyydellä  $w(v, u)$ 
8:         lisää  $u \in H_{i+1}$  ja aktivoi  $u$ 
9:       end if
10:    end for
11:    lisää  $v \in R_{i+1}$ 
12:  end for
13:   $i = i + 1$ 
14: end while
  
```

Esimerkki 2.1. Kuvassa 2.2 on esimerkki diffuusioprosessista IC -mallissa. Juuri aktivoituneet solmut on esitetty esimerkissä keltaisella, muut aktiiviset solmut harmaalla ja vielä epäaktiiviset solmut valkoisella. Kuvasta nähdään myös tartuntojen todennäköisyydet, eli kaarien painot.



Kuva 2.2. Kolmen aika-askleen pituinen diffuusioprosessi graafissa T

Kuvasta 2.2a nähdään diffuusioprosessin syötejoukko $H_0 = \{v_5\}$. Prosessin alussa ajanhetkellä $t = 0$, solmulla v_5 on mahdollisuus aktivoida sen ainoa lähtävä naapuri v_6 . Kuvasta 2.2b nähdään, että tartunta on onnistunut ja solmu v_6 on muuttunut aktiiviseksi. Samalla aktiivinen solmu v_5 muuttuu harmaaksi. Seuraavalla ajan hetkellä $t = 1$ solmulla v_6 on mahdollisuus aktivoida sen naapurit v_4 ja v_3 . Kuvasta 2.2c nähdään, että solmut v_3 ja v_4 aktivoituvat ja muuttuvat keltaisiksi. Samalla tartuttanut solmu v_6 muuttuu harmaaksi. Ajanhetkellä $t = 2$, solmulla v_3 on mahdollisuus aktivoida solmu v_2 , mutta kuten kuvasta 2.2d nähdään, aktivointi ei onnistu ja solmu v_2 jää epäaktiiviseksi. Kuvasta 2.2d nähdään myös, että juuri aktivoituneita solmuja ei ole enää jäljellä, joten diffuusioprosessi pysähtyy.

Kuvasta 2.2 huomataan myös, että solmulla v_1 ei ole yhtään naapuria joukossa $N_{in}(v_1)$, joten sillä ei ole kyseisessä prosessissa mahdollisuutta aktivoitua. Lisäksi tapauksessa, jossa jonkin solmun lähtävä naapuri on aktivoitunut jo jonkun muun solmun tartuttamana ei kyseisellä solmulla ole enää mahdollisuutta vaikuttaa naapuriinsa.

3. VAIKUTUKSEN MAKSIMOINTI -ONGELMA

3.1 Vaikutuksen maksimointi IC-mallin avulla

Vaikutuksen maksimoinnissa (IM) (*influence maximization*) yritetään löytää sellainen syötejoukko IC-prosessille, että prosessissa aktivoituvien solmujen lukumäärä olisi mahdollisimman suuri. Syötejoukko on se joukko solmuja, jotka asetetaan aktiivisiksi ennen kuin prosessissa on tapahtunut yhtään vaikuttamisyritystä. Syötejoukko on siis kaikkien aktiivisten solmujen joukko graafissa ennen, kuin tieto siinä alkaa levitä. [4] Syötejoukko valitaan usein joidenkin merkittyjen solmujen joukosta, sillä sen valitsemiseen liittyy poikkeuksetta jokin lukumäärään liittyvä rajoite. Jos rajoitetta ei olisi, vaikutuksen maksimoinnin ongelma olisi melko yksinkertainen: asetettaisiin vain kaikki solmut alussa aktiivisiksi ja maksimaalinen vaikutus olisi saatu. Toinen ääripää olisi valita syötejoukoksi tyhjä joukko, jolloin graafissa ei intuitiivisesti ajateltuna voi tapahtua uusia aktivoitumisia.

Ajatellaan esimerkin vuoksi tilannetta, jossa yritys haluaa markkinoida uutta tuotettaan tietylle kohderyhmälle jonkin budjetin rajoissa. Markkinoidakseen tuotetta yritys päättää valita tunnettuja somevaikuttajia tuotekokeiluun. Suostuakseen mainontaan somevaikuttajat haluavat vastineeksi rahaa. Yrityksen budjetti ei riitä siihen, että jokaiselle kohderyhmän jäsenelle maksettaisiin tuotteen kokeilusta. Sen sijaan yrityksen on löydettävä somevaikuttajien joukosta sellainen budjetin kokoinen joukko, jonka vaikutus kohderyhmään olisi mahdollisimman suuri.

Määritellään syötejoukon H_0 vaikutus (*influence*) IC-prosessissa aktivoituvien solmujen lukumäärän odotusarvona seuraavasti.

Määritelmä 3.1. Olkoon H_0 syötejoukko graafissa $G = (V, E)$. $\text{Prob}[v|H_0]$ on todennäköisyys, jolla solmu v aktivoituu IC-prosessissa, kun prosessin syötejoukko on H_0 . *Vaikutusfunktio* $\sigma(H_0)$ on aktiivisten solmujen lukumäärän odotusarvo IC-prosessin jälkeen. Toisin sanoen vaikutusfunktio on

$$\sigma(H_0) = \sum_{v \in V} \text{Prob}[v|H_0]. \quad (3.1)$$

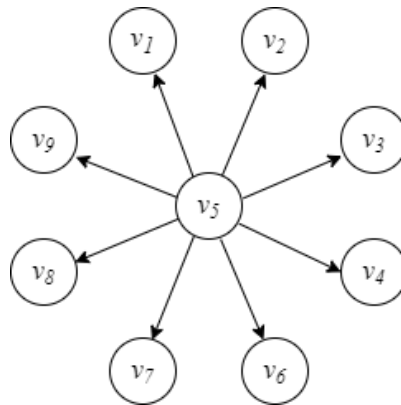
Vaikutuksen maksimoinnissa tavoitteena on löytää sellainen syötejoukko, jolle vaikutus on kaikista suurin. Seuraavassa määritelmässä 3.2 kuvataan matemaattisesti tässä työssä tarkasteltava ongelma.

Määritelmä 3.2. Olkoon H_0 syötejoukko graafissa $G = (V, E)$. Lisäksi on annettu luku k siten, että $|H_0| \leq k$. *Vaikutuksen maksimoinnin* ongelma voidaan ratkaista löytämällä sellainen syötejoukko H_0 , jolle

$$\sigma(H_0) = \sum_{v \in V} \text{Prob}[v|H_0] \quad (3.2)$$

on mahdollisimman suuri.

Joissain tapauksissa IM-ongelma on hyvin yksinkertainen ja oikea syötejoukko voidaan nähdä melko suoraan. Otetaan esimerkiksi tilanne, jossa valittavassa syötejoukossa voi olla korkeintaan yksi solmu ja graafi on kuvan 3.1 mukainen.



Kuva 3.1. Yksinkertainen tapaus vaikutuksen maksimoinnille

Kuvasta 3.1 nähdään, että solmulla v_5 on vaikutus kaikkiin muihin solmuihin, joten valitsemalla syötejoukoksi $H_0 = \{v_5\}$ saadaan suurin arvo $\sigma(H_0)$. Tällöin huonoimmassa tapauksessa aktiivisten solmujen lukumäärä prosessin lopuksi on $H = \{v_5\}$ ja parhaassa tapauksessa $H = V$. Valitsemalla syötejoukoksi mikä tahansa muu solmu saadaan parhaassakin tapauksessa aktiivisten solmujen lukumääräksi vain $|H| = 1$.

Toisaalta usein sosiaalisissa verkostoissa on huomattava määrä solmuja ja yhteyksiä, jolloin esimerkiksi kuvan 3.2 kaltaisesta graafista on lähes mahdotonta valita parasta syötejoukkoa intuitiivisesti.



Kuva 3.2. Kuva graafista, jossa on suuri määrä solmuja [11].

3.2 Vaikutusfunktion submodulaarisuus

Vaikutuksen maksimoinnin ongelmalle voidaan kehittää algoritmeja, jotka etsivät graafista sellaisen syötejoukon, jolle vaikutusfunktion arvo on mahdollisimman suuri. Tällainen algoritmi voidaan kehittää muun muassa vaikutusfunktion submodulaarisuuden kautta [4]. Tarkastellaan siis seuraavaksi IC-mallin vaikutusfunktion submodulaarisuutta.

Määritellään ensiksi submodulaarinen funktio. Submodulaarisuuden määritelmä 3.3 löytyy esimerkiksi lähteestä [8].

Määritelmä 3.3. Jos Ω on äärellinen joukko niin *submodulaarinen funktio* on joukkofunktio $f : 2^\Omega$, missä 2^Ω on kaikkien joukon Ω alijoukkojen joukko (sisältäen myös joukon Ω), joka täyttää jonkun seuraavista ehdoista:

1. Jokaiselle $X, Y \subseteq \Omega$, kun $X \subseteq Y$, ja jokaiselle $x \in \Omega \setminus Y$ pätee $f(X \cup \{x\}) - f(X) \geq f(Y \cup \{x\}) - f(Y)$.
2. Jokaiselle $S, T \subseteq \Omega$ pätee $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$
3. Jokaiselle $X \subseteq \Omega$ ja $x_1, x_2 \in \Omega \setminus X$, kun $x_1 \neq x_2$ pätee $f(X \cup \{x_1\}) + f(X \cup \{x_2\}) \geq f(X \cup \{x_1, x_2\}) + f(X)$.

Seuraavat määritelmät, lauseet ja tulokset luvussa 3.2 on kirjoitettu lähes yksinomaan lähteen [4] pohjalta.

Lause 3.1. *Mielivaltaiselle IC-mallille vaikutusfunktio $\sigma(\cdot)$ on submodulaarinen.*

Submodulaarisuus voidaan siis todistaa tarkastelemalla erotusta $\sigma(H_0 \cup v) - \sigma(H_0)$, missä H_0 on mielivaltainen syötejoukko ja v jokin solmu graafissa. Toisin sanoen halutaan tarkastella sitä muutosta vaikutusfunktiossa, joka saadaan, kun solmu v lisätään syötejoukkoon. Tätä muutosta on vaikea analysoida suoraan, sillä vaikutusfunktion suuruutta voi olla vaikea arvioida. Vaikeaksi arvioinnin tekee esimerkiksi se, että ei ole määritetty sitä järjestystä, millä tietyllä ajanhetkellä olevat tartuttavat solmut yrittävät aktivoida naapureitaan.

Lause 3.1 on todistettu lähteessä [4] ja käydään todistus myös tässä työssä läpi. Määritellään kuitenkin ensin todistusta varten tarvittava live-edge-malli. Live-edge-mallissa näkökulma IC-prosessin kulkuun poikkeaa hieman aiemmin esitellystä algoritmista 1, mutta ei muuta prosessin tulosta. Live-edge -mallin avulla on kuitenkin helpompi nähdä, että tietyllä ajanhetkellä tartuttavien solmujen vaikuttamisjärjestyksellä ei ole väliä. Tästä tuleekin IC-malliin nimeen liitetty sana *independent* eli riippumaton.

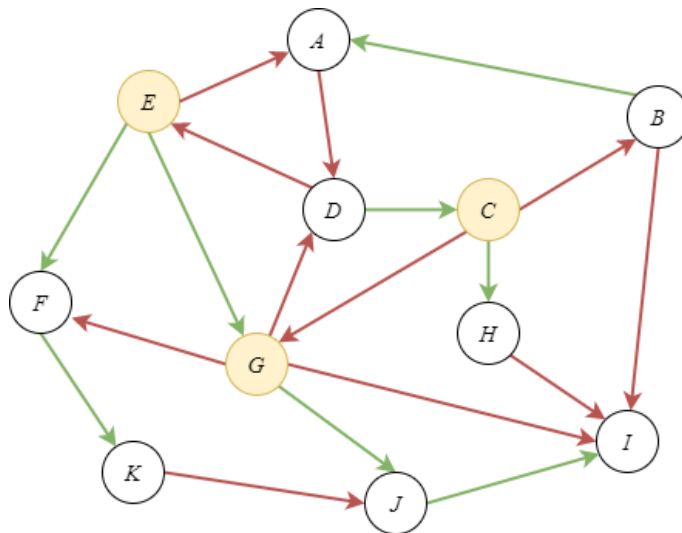
Tarkastellaan sellaista kohtaa prosessissa, jossa tartuttava solmu v yrittää aktivoida sen naapurisolmun $u \in N_{out}(v)$. Ajatellaan aktivoimisyritys tapahtumana, jossa heitetään painotettua kolikkoa niin, että solmun u aktivoivan puolen todennäköisyys on $w(v, u)$. Jos kolikonheiton tulos katsotaan vasta vaikuttamishetkellä, heiton ajankohdalla ei ole merkitystä. Voidaan siis päätellä, että jokaiselle graafin kaarelle (u, v) kolikko painotuksella $w(u, v)$ on heitetty jo aivan prosessin alussa ja

heiton tulos on tallennettu muistiin. Kunkin heiton tulos katsotaan vasta tilanteessa, jossa solmu v on tartuttava solmu ja u epäaktiivinen solmu. Jos jokaisen kaaren kohdalla kolikko heitetään jo aivan prosessin alussa, voidaan prosessia kuvata seuraavasti: kaikki ne kaaret, joille kolikon heittäminen tuottaa solmun aktivoitumisen graafissa G ovat *live*-kaaria ja kaikki loput kaaret ovat *blocked*-kaaria. Jos siis heitetään kaikki kolikot ja aktivoidaan syötejoukko prosessin alussa, voidaan aktiivisten solmujen joukko prosessin jälkeen määritellä melko helposti.

Määritelmä 3.4. Olkoon H_0 syötejoukko graafissa $G = (V, E)$. Solmu $v \in V$ aktivoituu IC-prosessissa jos ja vain jos solmuun v on olemassa sellainen polku jostain solmusta $u \in H_0$, joka koostuu vain live-kaarista. Määritellään tällainen polku *live-edge-polkuna*.

Esimerkki 3.2. Otetaan esimerkiksi kuvan 3.3 mukainen graafi. Kuva on graafin tilanne aivan prosessin alussa, kun kaikkien kaarien kohdalla on heitetty kaaren painolla painotettu kolikko. Painoja ei ole merkitty kuvaan, mutta kaaren väri kertoo kolikon heiton tuloksen. Vihreä kaari on live-kaari ja punainen kaari puolestaan blocked-kaari. Graafiin on merkitty keltaisilla solmuilla prosessin syötejoukko.

Kuvan 3.3 graafista nähdään kaikki live-edge-polut, jolloin voidaan helposti määrittää prosessin aikana aktivoituvat solmut. Kaikki live-edge polut ovat (E, F) , (E, F, K) , (E, G) , (G, J) , (G, J, I) ja (C, H) , joten lopussa aktiivisten solmujen joukko on $\{E, G, C, F, K, J, I, H\}$. Lisäksi kuvasta voidaan nähdä, että vaikka kaari (B, A) onkin live-kaari, ei solmu A voi aktivoitua, sillä siihen ei ole olemassa live-edge-polkua yhdestäkään syötejoukon solmusta.



Kuva 3.3. Esimerkkipiirros graafista, jonka kaarille on suoritettu live-edge-mallin mukainen kolikonheitto. Kuvasta nähdään kaikki live- ja blocked-kaaret.

Määritellään todennäköisyysavaruus, jossa jokainen piste on mahdollinen tulosjoukko tapahtumalle, jossa jokaisen kaaren kolikko heitetään. Olkoon X yksi piste tässä otosavaruudessa. Määritellään $\sigma_X(H_0)$ kaikkien prosessissa aktivoituneiden solmujen joukkona, kun H_0 on prosessin syötejoukko ja X kaikkien kolikonheittojen tulosjoukko. Koska joukko X on jo muodostettu, $\sigma_X(H_0)$ on itse asiassa deterministinen suure, jonka arvo voidaan ilmaista seuraavasti: Olkoon $R(v, X)$ kaikkien

niiden solmujen joukko, joihin on olemassa live-edge-polku solmusta v . Määritelmän 3.4 perusteella $\sigma_X(H_0)$ on niiden solmujen joukko, joihin on olemassa live-edge-polku jostain solmusta $v \in H_0$, ja sen kardinaalisuus on ekvivalentti yhdisteen $\cup_{v \in H_0} R(v, X)$ kardinaalisuuden kanssa.

Todistus lauseelle 3.1 Olkoon X yksi mahdollinen kolikonheittojen tuottama tulosjoukko. Väitetään ensin, että jokaiselle tulosjoukolle X , vaikutusfunktio $\sigma_X(A)$ on submodulaarinen ja yritetään todistaa tämä. Olkoot S ja T solmujoukkoja siten, että $S \subseteq T$. Tarkastellaan erotusta $\sigma_X(S \cup \{v\}) - \sigma_X(S)$. Tämä erotus on siis kaikkien niiden solmujen $u \in R(v, X)$ lukumäärä, jotka eivät ole vielä yhdisteessä $\cup_{u \in S} R(u, X)$. Toisin sanoen erotus $\sigma_X(S \cup \{v\}) - \sigma_X(S)$ on vähintään yhtä suuri kuin niiden solmujen $u \in R(v, X)$ lukumäärä, jotka eivät ole vielä yhdisteessä $\cup_{u \in T} R(u, X)$. Tästä seuraa $\sigma_X(S \cup \{v\}) - \sigma_X(S) \geq \sigma_X(T \cup \{v\}) - \sigma_X(T)$, mikä todistaa vaikutusfunktion $\sigma_X(\cdot)$ submodulaarisuuden. Vaikutusfunktio $\sigma(H_0)$ saadaan kaikkien mahdollisten vaikutusfunktioiden $\sigma_X(H_0)$ painotettuna keskiarvona seuraavasti:

$$\sigma(H_0) = \sum_{\text{kaikki tulosjoukot } X} \text{Prob}[X] \cdot \sigma_X(H_0) \quad (3.3)$$

Koska submodulaaristen funktioiden epänegatiivinen lineaarikombinaatio on myös submodulaarinen, niin vaikutusfunktio $\sigma(\cdot)$ on submodulaarinen. \square

3.3 Algoritmit

Vaikutuksen maksimoinnin ongelmalle on kehitetty useita algoritmeja, jotka yrittävät löytää mahdollisimman hyvän ratkaisun. Lähteessä [4] näytetään, että optimaalisen ratkaisun löytäminen on NP-täydellinen ongelma ja siksi algoritmit perustuvatkin odotusarvojen laskentaan ja Monte-Carlo-simulointiin. Esitellään tässä luvussa algoritmi, joka pohjautuu David Kempen kehittämään alkuperäiseen Greedy-algoritmiin [4]. Kempe kehitti Greedy-algoritmin tuottamaan approksimaation syötejoukosta, joka maksimoi vaikutuksen IC-prosessissa. Sen jälkeen algoritmin tehokkuutta on yritetty parantaa ja useita algoritmeja onkin kehitetty, esimerkkeinä lähteessä [6] esitetyt algoritmit *CascadeDiscount* ja *ScoreCumulate*.

Aiemmin osoitettiin, että vaikutusfunktio $\sigma(\cdot)$ on submodulaarinen. Näytetään nyt, että funktio on myös *monotoninen*.

Määritelmä 3.5. Olkoon S ja T joukkoja siten, että $S \subseteq T$. Joukkofunktio f on *monotoninen*, jos sille pätee $f(S) \leq f(T)$.

Lause 3.3. Mielivaltaiselle IC-mallille vaikutusfunktio $\sigma(\cdot)$ on *monotoninen*. [13]

Todistus lauseelle 3.3. Käytetään todistuksessa apuna osiossa 3.2 määriteltyä todennäköisyysavaaruutta ja live-edge-mallia. Olkoon $\sigma_X(H_0)$ kaikkien prosessissa aktivoituneiden solmujen joukko, kun H_0 on prosessin syötejoukko ja X kaikkien kolikonheittojen tulosjoukko. Väitetään ensin, että jokaiselle tulosjoukolle X ja syötejoukolle H_0 , vaikutusfunktio $\sigma_X(H_0)$ on monotoninen ja todistetaan tämä. Oletetaan, että solmu v kuuluu jo syötejoukkoon H_0 . Jos lisätään syötejoukkoon solmu

v , vaikutusfunktion $\sigma_X(H_0)$ arvo pysyy samana. Oletetaan sitten, että solmu v ei kuulu vielä syötejoukkoon. Selvästi nähdään, että lisäämällä v syötejoukkoon saadaan $\sigma_X(H_0) < \sigma_X(H_0 \cup \{v\})$. Vaikutusfunktio σ_X on siis monotoninen. Kuten lauseen 3.1 todistuksessa näytettiin, vaikutusfunktio $\sigma(H_0)$ saadaan kaikkien tulosjoukkojen X funktioiden $\sigma_X(H_0)$ lineaarikombinaationa. Koska monotonisten funktioiden epänegatiivinen lineaarikombinaatio on myös monotoninen, niin vaikutusfunktio $\sigma(\cdot)$ on monotoninen. \square

Lähteen [13] mukaan jokaiselle sekä submodulaariselle että monotoniselle funktiolle f , jolle $f(\emptyset) = 0$, saadaan k solmun kokoinen syötejoukko approksimoitua yksinkertaisella Greedy-algoritilla. Algoritmi on esitetty algoritmissa 2.

Algoritmi 2 on niin kutsuttu ahne algoritmi, mikä tarkoittaa sitä, että algoritmi pyrkii ratkaisemaan ongelman tekemällä jokaisessa vaiheessa mahdollisimman optimaalisen, mutta lyhytnäköisen päätöksen [5]. Algoritmissa on k kierrosta. Aluksi syötejoukko H_0 alustetaan tyhjäksi ja jokaisella kierroksella algoritmi pyrkii lisäämään syötejoukkoon yhden solmun $v \in V$, joka maksimoi vaikutusfunktion kohdistuvan lisäyksen $\sigma(H_0 \cup \{v\}) - \sigma(H_0)$. [13] Algoritmi 2 on otettu lähes sellaisenaan lähteestä [13].

Algorithm 2 Greedy Algorithm

- 1: Alusta $H_0 = \emptyset$
 - 2: **for** $i = 1$ to k **do**
 - 3: valitse $v = \operatorname{argmax}_{u \in V \setminus H_0} (\sigma(H_0 \cup \{u\}) - \sigma(H_0))$
 - 4: $H_0 = H_0 \cup \{v\}$
 - 5: **end for**
-

Todellisuudessa algoritmin 2 siirtäminen käytäntöön on tietenkin hiukan monimutkaisempaa kuin miltä näyttää. Ennen kuin parhaat mahdolliset solmut kyetään valitsemaan, on tehtävä suuri määrä Monte-Carlo -simulaatioita ja suoritettava muun muassa todennäköisyyksiin perustuvia laskutoimituksia. [13] Käytännössä algoritmi joutuu jokaisella kierroksella käymään läpi Monte-Carlo -simulaatioilla tuotettuja aligraafeja, kun se laskee odotusarvoja solmujen vaikutuksille. Tämä selittää myös algoritmin tehokkuutta suurikokoisten graafien tapauksissa. Lähteen [6] mukaan Greedy-algoritmin laskennallinen tehokkuus O -notaatiolla ilmaistuna on $O(kR|V||E|)$, missä R on Monte-Carlo -simulaatioiden lukumäärä ja $|V|$ ja $|E|$ solmujen ja kaarien lukumäärät graafissa. Tämä tehokkuus tekee algoritmista tosielämän verkostoihin sovellettuna hyvin epäkäytännöllisen.

Vaikutuksen maksimoinnille kehitettyjen algoritmien laskennallinen teho suurikokoisille graafeille on edelleen kehityksestä huolimatta huono. Lähteen [13] mukaan esimerkiksi vuoden 2009 tutkimuksessa kehitetyllä *lazy-forward* -algoritilla menee useita tunteja valita 50 solmun syötejoukko muutaman tuhannen solmun graafista.

4. YHTEENVETO

Graafiteoriaa voidaan soveltaa sosiaalisiin verkostoihin niin sosiaalisen median kuin todellisenkin elämän tapauksissa. Sosiaalisiin verkostoihin liittyy aina tiedon kulkeutumista ihmisten välillä ja tätä tutkimalla voidaan johonkin verkostoon yrittää levittää haluttua vaikutusta. Vaikutuksen maksimointi tähtää siihen, että mahdollisimman suuri osa ihmisiä jossain tietyssä verkostossa saadaan muuttamaan mieltään tai käyttäytymistään haluttuun suuntaan. Tämän tutkiminen matemaattisesti ja ratkaisutapojen esittäminen ovatkin olleet tämän työn tarkoituksena.

Taulukossa 4.1 on tiivistetty yhteenveto tästä tutkielmasta. Taulukossa on lueteltuna työn sisältö lyhyiden avainsanojen avulla.

Taulukko 4.1. Tiivistetty yhteenveto tutkielmasta

Ongelma	Vaikutuksen maksimointi graafissa
Sovellusalue	Sosiaalinen verkosto
Käytetyt mallit	Independent Cascade, Live-Edge
Olellaiset lauseet	Vaikutusfunktion submodulaarisuus ja monotonisuus
Ratkaisumallit	Greedy-algoritmi
Haastavuus	Ongelman NP-täydellisyys, algoritmien huono tehokkuus suurikokoisille graafeille

Työn alussa esiteltiin yleistä teoriaa graafeista ja pyrittiin kertomaan, miten graafiteoriaa voidaan soveltaa sosiaaliseen verkostoon. Todettiin esimerkiksi se, että sosiaaliset verkostot voivat erota toisistaan monella tapaa. Kuitenkin moniin verkostoihin voidaan hyödyntää jo olemassa olevia tiedon leviämisen malleja, joista yksi on Independent Cascade. IC-malliin perehdyttiin luvussa 2.2 ja olennaista on, että kyseessä on diskreetteinä aika-askelina etenevä prosessi, jonka jokaisessa

vaiheessa aktiiviset solmut pyrkivät vaikuttamaan lähteviin naapureihinsa. Prosessi jatkuu, kunnes uusia aktivoitumisia ei enää tapahdu.

Luvussa 3 määriteltiin varsinainen ongelma IC-mallin kautta. Kappaleessa todistettiin vaikutusfunktion submodulaarisuus ja monotonisuus, jotka ovat olennaisia seikkoja ratkaisualgoritmeja muodostettaessa. Submodulaarisuus todistettiin *live – edge* -mallin avulla, mikä osoitti myös sen, että solmujen aktivoitumiset voidaan ajatella toisistaan riippumattomina tapahtumina.

Lopulta aliluvussa 3.3 päädyttiin tarkastelemaan ongelman ratkaisemista. Saatiin tulos, että ongelma on NP-täydellinen ja, että suurikokoisille graafeille algoritmien suorittaminen saattaa viedä edelleen useita tunteja. Ongelmalle on kuitenkin kehitetty algoritmeja, jotka pyrkivät löytämään sellaisen jollain annetulla luvulla k rajoitetun syötejoukon, jolle vaikutusfunktio saadaan maksimoitua. Luvussa 3.3 esitetty yksinkertainen Greedy-algoritmi laskee jokaisella kierroksella solmun, joka tuottaa suurimman lisäyksen vaikutusfunktioon ja lisää sen syötejoukkoon.

LÄHTEET

- [1] J. L. Gross ja J. Yellen. *Graph Theory and its Applications - Second Edition*. Chapman & Hall/CRC, 2006.
- [2] H. Hansen. *Graafialgoritmit -opintojakson opintomoniste*. Tampereen yliopisto. Tampere, 2021.
- [3] T. Harko et al. Exact analytical solutions of the Susceptible-Infected-Recovered (SIR) epidemic model and of the SIR model with equal death and birth rates. *Applied Mathematics and Computation* 236 (2014), 184–194.
- [4] D. Kempe et al. Maximizing the spread of influence through a social network (2003).
- [5] S. Khreher. *Combinatorial algorithms*. CRC Press, 1999.
- [6] F. Lu et al. Scalable influence maximization under independent cascade model (2003).
- [7] A. Partner. Tehokkuutta asiakashankintaan (). URL: <https://www.slideshare.net/JMoisio/tehokkuutta-asiakashankintaan-tiivistelm>.
- [8] A. Schrijver. *Combinatorial Optimization*. Springer, 2003.
- [9] J. Scott. *Social Network Analysis*. 2017.
- [10] P. Shakarian et al. *Diffusion in Social Networks*. Springer, 2015.
- [11] sosiaalinen verkosto-kuva (). URL: https://commons.wikimedia.org/wiki/File:Social_Network_Analysis_Visualization.png.
- [12] K. Thulasiraman ja M. Swamy. *Graphs: Theory and Algorithms*. 1992.
- [13] C. Wang et al. Scalable influence maximization for independent cascade model in large-scale social networks. *Data Mining and Knowledge Discovery* 22 (2012), 545–576.
- [14] Q. Wang et al. Influence maximization in social networks under an independent cascade-based model. *Physica A: Statistical Mechanics and its Applications* 444 (2016), 20–34.