Preprint of: Harri Haanpää and Patric Östergård, Steinerin kolmikkosysteemeistä sudokuun, Arkhimedes 2/2006 18–21

Steinerin kolmikkosysteemeistä sudokuun

Harri Haanpää
Tietojenkäsittelyteorian laboratorio
Teknillinen korkeakoulu
PL 5400
02015 TKK

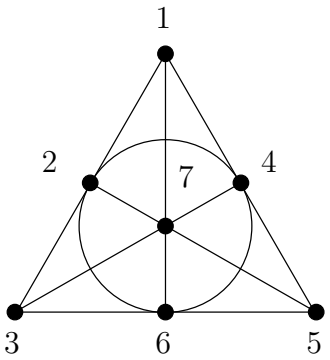
Patric R. J. Östergård
Tietoliikennelaboratorio
Teknillinen korkeakoulu
PL 3000
02015 TKK
E-mail: Harri.Haanpaa@tkk.fi, Patric.Ostergard@tkk.fi

Steinerin kolmikkosysteemit

1990-luvun puolessavälissä, muutama vuosi ennen ennenaikaista kuolemaansa, aina yhtä sympaattinen Ed Assmus oli kollegansa Aimo Tietäväisen vieraana Turun yliopistossa. Vierailun aikana Ed piti tavanmukaisen vierailuesitelmän, joka liittyi Steinerin kolmikkosysteemeihin. Esitelmän tarkka sisältö on jo jäänyt unholaan – se lienee liittynyt tällaisten systeemien insidenssimatriisien generoimiin koodeihin, joka oli hänen tutkimusalojaan [1] – mutta sen alustus on sen sijaan tarkassa muistissa.

Ennen kuin palataan Edin esitykseen palautettakoon mieleen muutama kombinatoriikan perusmääritelmä. Kertaluvun n Steinerin kolmikkosysteemi koostuu n-alkioisesta perusjoukosta V, jonka alkioita kutsutaan pisteiksi, ja sellaisesta kokoelmasta kolmikkoja joukosta V, että jokainen pari joukosta V esiintyy täsmälleen yhdessä kolmikossa. Yleisyyttä rajoittamatta valitsemme $V = \{1, 2, \ldots, n\}$. Helpolla jaollisuusargumentilla voidaan osoittaa, että Steinerin kolmikkosysteemi voi olla olemassa vain, kun $n \equiv 1$ tai $3 \pmod 6$. Voidaan todistaa konstruktiivisesti, että tämä ehto on myös riittävä. Esimerkiksi

$$\{\{1,2,3\},\{1,4,5\},\{1,6,7\},\{2,4,6\},\{2,5,7\},\{3,4,7\},\{3,5,6\}\}$$



Kuva 1: Fanon taso

on kertaluvun 7 Steinerin kolmikkosysteemi. Sen graafisen esityksen (Kuva 1) jokainen diskreetin matematiikan tai geometrian perusteisiin perehtynyt tunnistaa ns. Fanon tasoksi.

Kahta Steinerin kolmikkosysteemiä pidetään *isomorfisina,* jos toinen saadaan toisesta nimeämällä perusjoukon alkiot uudelleen. Olisiko sitten olemassa kertaluvun 7 Steinerin kolmikkosysteemi, joka ei olisi isomorfinen Fanon tason kanssa? Muutaman minuutin tapauskohtainen analyysi osoittaa, ettei sellaista löydy. Entä miten on asian laita suuremmilla kertaluvuilla? Siirrytään tässä vaiheessa takaisin Ed Assmuksen esitelmään.

Ed kertoi innostuneesti miten kertaluvuilla 9, 13 ja 15 oli todistettu, että Steinerin kolmikkosysteemien isomorfialuokkien lukumäärät ovat vastaavasti 1, 2 ja 80, jonka jälkeen yleisön tehtävä oli arvata lukumäärä seuraavalla käyvällä kertaluvulla, eli 19. Yleisö taisi aavistaa, että tämä ei ole mikä tahansa sarja kokonaislukuja, ja pysyi hiljaa. Hetken päästä Ed totesi, että miljoonan ylihän tämä lukumäärä menee ja ilmeisesti oikein reippaasti.

Itse asiassa kertaluvun 19 Steinerin kolmikkosysteemejä on *miljardeja.* Ed Assmus ei saanut eläessään tietää niiden lukumäärää, mutta hän olisi varmasti ilahtunut siitä, että hänen esitelmänsä antoi kipinän tämän ongelman ratkaisuun. Mutta miten ne on mahdollista laskea? Ne eivät mahdu minkään tavanomaisen tietokoneen muistiin, ja jos haluamme esittää yhden kappaleen jokaisesta isomorfialuokasta, algoritmin on oltava hyvin nopea – vuodessakin on vain hiukan yli 30 miljoonaa sekuntia. Kuitenkaan isomorfialuokkien laskemiseen ei tunneta tapauskohtaista konstruointia merkittävästi tehokkaampia menetelmiä.

Ensimmäinen yritys kolmikkosysteemien konstruoimiseksi voisi olla yksinkertaisesti rakentaa kolmikkosysteemi kolmikko kolmikolta kaikilla mahdollisilla tavoilla. Toki voimme vaatia, että kolmikot lisätään rakenteeseen järjes-

tyksessä. Tätä varten määrittelemme järjestetyn joukon V osajoukoille ns. *sanakirjajärjestyksen*: kirjoitamme vertailtavien kahden osajoukon alkiot kasvavaan järjestykseen, ja ensimmäisten toisistaan poikkeavien alkioiden keskinäinen järjestys määrää osajoukkojen järjestyksen. Nyt voidaan konstruoida rekursiivisesti kaikki halutun kertaluvun Steinerin kolmikkosysteemit kokeilemalla aina jokaista mahdollista tapaa lisätä keskeneräiseen kolmikkosysteemiin kolmikko, joka tulee järjestyksessä kolmikossa jo olevien kolmikkojen jälkeen eikä sisällä pisteparia, jonka systeemissä jo oleva kolmikko sisältää.

Näin saadaan kyllä kaikki kertaluvun n Steinerin kolmikkosysteemit. Mutta mehän halusimme vain yhden kolmikkosysteemin kustakin isomorfialuokasta! Tämä menetelmä generoi samasta isomorfialuokasta pahimmillaan (ja usein) $n!$ kolmikkosysteemiä, yhden kutakin mahdollista perusjoukon nimeämistä kohti. Tämä on kenties vielä mahdollista, kun $n = 7$ ($7! = 5040$), mutta kun $n = 19$ ($19! \approx 1,2 \cdot 10^{17}$), menetelmä on täysin mahdoton – ja kaiken tämän generoinnin jälkeen pitäisi vielä pystyä seulomaan tasan yksi edustaja kustakin isomorfialuokasta.

Miten siis voidaan tehokkaasti luoda edustaja kustakin tietyt epätriviaalit ehdot täyttävien matemaattisten rakenteiden isomorfialuokasta? Viime vuosikymmeninä on esitetty useita menetelmiä tämän ongelman ratkaisemiseksi [5]. Edellisen perusteella on selvää, ettei voida ensin generoida kaikkia rakenteita ja sitten tehdä isomorfiakarsintaa, vaan karsintaa on tehtävä jo generoinnin aikana. Niinpä jaamme myös generoinnin aikana syntyvät osittaiset rakenteet isomorfialuokkiin ja käymme läpi vain yhden edustajan kustakin luokasta. Tässä tarkastelemme kahta menetelmätyyppiä: *kanonisen muodon* ja *kanonisen lisäyksen* menetelmää.

Kanonisen muodon ja kanonisen lisäyksen menetelmät

Kanonisen muodon menetelmä [2, 9] perustuu siihen, että generointia jatketaan osittaisesta rakenteesta vain, jos kyseinen osittainen rakenne on isomorfialuokkansa *kanoninen edustaja.* Edellä määrittelimme sanakirjajärjestyksen kolmikoille olettamalla joukon V järjestetyksi; nyt kun kolmikot on järjestetty, voimme vastaavasti määritellä sanakirjajärjestyksen kolmikkosysteemeille. Kolmikkosysteemi on *kanoninen,* jos se on järjestyksessä ensimmäinen isomorfialuokkansa edustaja.

Lisäämällä kanonisuustesti aiemmin hahmoteltuun naiiviin lähestymistapaan saadaan menetelmä, joka tuottaa jokaisen kolmikkosysteemien isomorfialuokan kanonisen edustajan. Tämä perustuu siihen, että jokainen kano-

ninen kolmikkosysteemi voidaan konstruoida siten, että osittainen systeemi, johon kolmikkoja lisätään, pysyy koko ajan kanonisena. Tämä perustuu viime kädessä sanakirjajärjestyksen ja algoritmin tekemien laajennusaskelien yhteensopivuuteen. Menetelmästä on hyvä huomata, että se ei missään vaiheessa vertaa kulloinkin tarkasteltavaa rakennetta esimerkiksi mahdollisesti aikaisemmin löytyneisiin rakenteisiin, vaan se tarkastelee ainoastaan, onko kulloinenkin rakenne kanoninen. Eri rakenteiden keskinäiseen vertailuun ei siis ole tarvetta.

Kanonisen muodon menetelmän haittapuolena on, että se käytännössä vaatii sanakirjajärjestyksen käyttöä. Vähemmän sallivilla isomorfismin käsitteillä menetelmä on hyvinkin käyttökelpoinen, mutta miten voisimme tässä päätellä tehokkaasti, mikä $n!$ nimeämisestä tuottaisi sanakirjajärjestyksessä ensimmäisen rakenteen?

Kanonisen lisäyksen menetelmässä [7] ei tarkastella *rakenteiden* vaan *lisäysten* kanonisuutta. Menetelmän pääpiirteinen idea on seuraava. Konstruoidessamme rakennetta X rakenteesta $p(X)$ kuvaamme tehtyä lisäystä järjestetyllä parilla $(X, p(X))$. Liitämme jokaiseen konstruoituun rakenteeseen X sen *kanonisen isän* $m(X)$, missä funktion $m(\cdot)$ on oltava siten nimeämisestä riippumaton, että jos X ja Y ovat isomorfiset, niin $m(X)$ ja $m(Y)$ ovat myös isomorfiset. Kun sitten konstruoimme rakenteen X jostakin rakenteesta $p(X)$, tarkastamme, onko tehty lisäys kanoninen – ovatko lisäykset $(X, p(X))$ ja $(X, m(x))$ keskenään isomorfiset – ja jatkamme generointia rakenteesta X vain, jos testin tulos on positiivinen. Näin saamme varmistettua, että jokaisen isomorfialuokan edustaja tulee generoiduksi kanonisesta isästään, ja vieläpä oikealla tavalla. Tietystä rakenteesta generoituja ja testin läpäisseitä rakenteita on edelleen vertailtava toisiinsa, mutta tämä vertailu saadaan tehtyä tehokkaasti hyödyntämällä rakenteen X automorfismiryhmää. Yksityiskohtia laadittaessa on tietenkin huolehdittava siitä, että jokainen generoinnin kannalta tarpeellinen osittainen rakenne saadaan konstruoitua kanonisesta isästään.

Kanonisen lisäyksen menetelmä on käsitteellisestikin hankalampi kuin kanonisen muodon menetelmä, mutta sen suurena etuna on, että kanonisuustarkasteluissa ei tarvitse rajoittua sanakirjajärjestykseen. Itse asiassa isäfunktioksi kelpaa mikä tahansa funktio, jolla kustakin osittaisesta rakenteesta saadaan sen nimeämisestä riippumattomalla tavalla jokin sen mahdollinen isä.

Kertaluvun 19 Steinerin kolmikkosysteemit

Kertaluvun 19 Steinerin kolmikkosysteemien isomorfialuokkien edustajat saatiin generoitua kanonisen lisäyksen menetelmällä [4]. Jokainen kertaluvun n Steinerin kolmikkosysteemin piste esiintyy $n-1$ parissa, ja jokainen kolmikko, johon piste kuuluu, sisältää näistä kaksi – jokainen piste on siis mukana $(n-1)/2$ kolmikossa. Kertaluvun 19 Steinerin kolmikkosysteemien generoimiseksi etsittiin aluksi siemenrakenteet, joissa oli kolmikko $\{1, 2, 3\}$ ja 24 muuta kolmikkoa, joista jokaisessa esiintyy 1, 2 tai 3. Näitä siemenrakenteita oli kaikkiaan 14 648.

Tämän jälkeen jokainen siemenrakenne täydennettiin kaikilla mahdollisilla tavoilla kertaluvun 19 Steinerin kolmikkosysteemiksi. Yksittäinen kolmikkosysteemi on toki mahdollista saada monestakin eri siemenestä, joten jokaiselle saadulle kolmikkosysteemille tehtiin kanonisen lisäyksen menetelmän mukainen isyystesti. Isyystestin ja kanonisen lisäyksen menetelmän toisen testin läpäisset rakenteet ovat sitten kertaluvun 19 Steinerin kolmikkosysteemien isomorfialuokkien edustajat. Montako niitä sitten on? 11 084 874 729. Eräs kaksoislaskenta-argumentti, jonka yksityiskohtiin ei ole mahdollista mennä tämän artikkelin puitteissa, tukee tämän laskennallisen tuloksen oikeellisuutta.

Ympyrä sulkeutuu

Onko 11 miljardin kolmikkosysteemin luonti pelkkää numeronmurskausta? Ei suinkaan. Tällainen kattava kokoelma rakenteita tarjoaa ihanteellisen mahdollisuuden esimerkiksi tarkastella niihin liittyviä otaksumia, varsinkin vastaesimerkkien löytämiseksi. Esitämme tässä yhden sellaisen esimerkin.

Täydelliselle yhden virheen korjaavalle koodille $C \subseteq Z_2^n$ (missä Z_2^n siis on Hamming-avaruus, joka koostuu n bitin binäärisanoista) pätee, että jokainen avaruuden sana on korkeintaan Hamming-etäisyydellä 1 täsmälleen yhdestä koodisanasta. Yleisyyttä rajoittamatta voimme olettaa, että nollasana kuuluu koodiin, jolloin koodisanat, joissa on kolme 1-bittiä, muodostavat Steinerin kolmikkosysteemin (kolmikon alkiot vastaavat 1-bittien positioita). Tästä herää sitten kysymys, esiintyykö jokainen Steinerin kolmikkosysteemi jossakin tällaisessa täydellisessä koodissa?

Täydellinen yhden virheen korjaava koodi on olemassa täsmälleen silloin, kun $n = 2^m - 1$, joten tarkastelu pitää kohdistaa tätä muotoa oleviin parametrien arvoihin. Kuten mainitsimme, kertalukua 7 oleva Steinerin kolmikkosysteemi on yksikäsitteinen, joten ensimmäinen epätriviaali tapaus on $n = 15$. Jo entuudestaan tiedettiin, että suuri osa 80:stä kertaluvun 15 Steinerin kol-

mikkosysteemi esiintyy tällä tavoin täydellisen koodin osana, mutta nyt tämä pitkään avoinna ollut ongelma on ratkennut, kun erään Steinerin nelikkosysteemeihin liittyvän tutkimuksen pohjalta on löydetty vastaesimerkkejä [6, 8].

Niin, viimeisen askelen täydellisten koodien olemassaolokysymyksen ratkaisussa (kun aakkoston koko on alkulukupotenssi) otti Aimo Tietäväinen [10].

Ja miten sudoku liittyy tähän?

Olemme jo aikaisemmin tässä lehdessä [3] törmänneet muoti-ilmiöön sudokuun ja kysymykseen sudokun matemaattisuudesta. Osoittautuu, että Steinerin kolmikkosysteemien konstruointi voidaan esittää samassa kehysrakenteessa kuin sudokutehtävien ratkaiseminen.

Olkoon annettu äärellinen perusjoukko U ja joukko $\mathcal{S} = \{S_1, S_2, \ldots, S_k\}$, missä $S_i \subseteq U$. Kysymys kuuluu, voidaanko joukoista S_i muodostaa joukon U ositus, eli löytyykö $a_1, a_2, \ldots, a_s$ siten, että $U = \cup_{i=1}^{s} S_{a_i}$ ja $S_{a_i} \cap S_{a_j} = \emptyset$ kun $1 \leq i < j \leq s$? Tietojenkäsittelytieteessä tämä laskennallinen ongelma tunnetaan täsmällinen peitto -ongelmana (exact cover).

Kertaluvun n Steinerin kolmikkosysteemin muodostaminen voidaan nyt esittää täsmällinen peitto -ongelmana seuraavasti:

$$U = \{\{i, j\} : 1 \leq i < j \leq n\},$$

$$\mathcal{S} = \{\{\{i, j\}, \{i, k\}, \{j, k\}\} : 1 \leq i < j < k \leq n\}.$$

Algoritmit tämän ongelman ratkaisemiseksi muodostivat itse asiassa tärkeän osan kertaluvun 19 Steinerin kolmikkosysteemien siemenrakenteiden täydentämisessä.

Ja miten sudokun ratkaiseminen sitten voidaan esittää täsmällisen peitteen ongelmana? Jätämme tämän helpon tehtävän lukijan pohdittavaksi.

Viitteet

[1] E. F. Assmus, Jr., J. D. Key, Designs and Their Codes, Cambridge University Press, Cambridge, 1992.

[2] I. A. Faradžev, Constructive enumeration of combinatorial objects, *Problèmes Combinatoires et Théorie des Graphes*, (Université d'Orsay, July 9–13, 1977), CNRS, Paris, 1978, 131–135.

[3] M. Gyllenberg, Sudoku, *Arkhimedes* 6/2005, 31.

[4] P. Kaski, P. R. J. Östergård, The Steiner triple systems of order 19, *Math. Comp.* 73 (2004), 2075–2092.

[5] P. Kaski, P. R. J. Östergård, Classification Algorithms for Codes and Designs, Springer, Berlin, 2006.

[6] P. Kaski, P. R. J. Östergård, O. Pottonen, The Steiner quadruple systems of order 16, lähetetty julkaistavaksi.

[7] B. D. McKay, Isomorph-free exhaustive generation, *J. Algorithms* 26 (1998), 306–324.

[8] P. R. J. Östergård, O. Pottonen, There exist Steiner triple systems of order 15 that do not occur in a perfect binary one-error-correcting code, lähetetty julkaistavaksi.

[9] R. C. Read, Every one a winner; or, How to avoid isomorphism search when cataloguing combinatorial configurations, *Ann. Discrete Math.* 2 (1978), 107–120.

[10] A. Tietäväinen, On the nonexistence of perfect codes over finite fields, *SIAM J. Appl. Math.* 24 (1973), 88–96.