Tampereen yliopisto

Miikka Venäläinen

VIRHEIDEN HAVAITSEMINEN JA KORJAAMINEN KOODAUSTEORIAN AVULLA

Informaatioteknologian ja viestinnän tiedekunta
Kandidaattitutkielma
Marraskuu 2023

Tiivistelmä

Miikka Venäläinen: Virheiden havaitseminen ja korjaaminen koodausteorian avulla
Kandidaattitutkielma
Tampereen yliopisto
Matematiikan ja tilastotieteen tutkinto-ohjelma
Marraskuu 2023

Tässä tutkielmassa lähestytään koodausteoriaa matriisilaskennan kautta. Koodausteoria tutkii viestien koodaamista siten, että lähetyksen aikaiset kohinan aiheuttamat muutokset voitaisiin havaita ja korjata virheenkorjauskoodin avulla. Kohinasta aiheutuvaa koodisanan muutosta kutsutaan virheeksi. Virheenkorjauskoodin tärkeimmät käsitteet ja tulokset määritellään tutkielman toisessa luvussa. Kolmannessa ja neljännessa luvussa keskitytään ryhmäkoodin määrittelyyn ja Hamming-koodin muodostamiseen.

Luvussa 2 esitellään nimitys n-bittijono sellaisille bittijonoille, joiden pituus on n. Lukijalle tulee myös tutuksi koodaus- ja purkufunktioiden määritelmät. Tämän jälkeen määritellään Hamming-etäisyys, joka on funktio, ja myös metriikka, joka kertoo kahden n-bittijonon toisistaan eroavien bittien lukumäärän. Hamming-etäisyyden avulla määritellään koodin minimietäisyys, joka on minimi koodisanojen Hamming-etäisyyksien joukosta. Minimietäisyys on virheenkorjauskoodin tärkeä ominaisuus, sillä se määrittää voiko kohinasta aiheutuvaa virhettä havaita tai korjata. Aivan luvun lopussa osoitetaankin yhteys koodin minimietäisyyden ja purkumenetelmän virheentunnistus- ja virheenkorjauskyvyn välillä. Tulokseksi saadaan, että minimietäisyyden ollessa vähintään kolme, purkumenetelmä voi korjata yhden virheen tai havaita yhden ja kahden bitin virheet.

Luvun 3 pääpaino on ryhmäkoodeissa ja niiden ominaisuuksissa. Ryhmäkoodit ovat erityislaatuisia koodeja, joiden kahden koodisanan summa on edelleen koodisana. Luvun ensimmäisessä alaluvussa määritellään bittien yhteenlaskutoimitus ja paino, ja osoitetaan painon ja Hamming-etäisyyden välinen yhteys. Toinen alaluku pitää sisällään vektori- ja matriisilaskentaa, joiden tulosten avulla osoitetaan, että ryhmäkoodeja voidaan tuottaa virittäjämatriiseilla, joka on nopea tapa luoda virheenkorjauskoodi. Yksi tällainen virittäjämatriisi on luvun lopussa määritelty kanoninen

pariteettimatriisi H.

Luvussa 4 päästään vihdoin muodostamaan Hamming-koodi. Ennen Hamming-koodin muodostamista osoitetaan, että binäärimatriisin H ydin muodostaa virheen tunnistavan ryhmäkoodin jos, ja vain jos yksikään sen sarakkeista ei ole 0^T. Osoitetaan myös, että matriisin H ydin muodostaa virheen korjaavan ryhmäkoodin jos, ja vain jos edeltävä tulos pitää paikkansa ja ettei matriisi H sisällä kahta identtistä saraketta. Tästä päästään syndrooman määritelmään, ja todistukseen, ettei syndrooma ole riippuvainen koodisanasta. Lopuksi osoitetaan, että virheen sattuessa vastaanotetun viestin syndrooma vastaa jotakin matriisin H saraketta, joka kertoo virheen paikan.

Avainsanat: koodisana, purkumenetelmä, ryhmäkoodi, binäärimatriisi,
Hamming-koodi

Sisällys

1 Johdanto

Tässä tutkielmassa tarkastellaan virheenkorjauskoodin muodostamista.

Luvussa 2 määritellään virheenkorjauskoodin tärkeimmät käsitteet ja tutkitaan niiden ominaisuuksia. Luvussa määritellään sekä koodausteorian tärkeimmät funktiot, että purkumenetelmät sekä käydään läpi koodausteorian perusidea.

Luku 3 keskittyy ryhmäkoodin määrittelyyn ja ryhmäkoodien eri ominaisuuksiin. Ryhmäkoodit ovat erityislaatuinen joukko koodisanoja, joiden summatkin ovat koodisanoja. Ryhmäkoodien muodostaminen virittäjämatriisilla tuottaa nopean tavan luoda virheenkorjauskoodi.

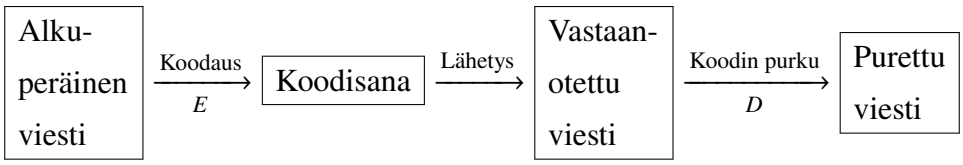
Luvussa 4 päästään muodostamaan täydellinen Hamming-koodi, joka pystyy havaitsemaan kaksi virhettä tai havaitsemaan ja korjaamaan yhden virheen. Hamming-koodi on erityislaatuinen ryhmäkoodi.

Lukijan odotetaan osaavan lukiotason matematiikkaa sekä ymmärtävän yksinkertaista matriisilaskentaa. Tutkielman päälähteenä käytetään Lauferin teosta Discrete Mathematics and Applied Modern Algebra [1, s. 1–61].

2 Virheenkorjauskoodit

Määritellään ensiksi koodausteorian funktioita seuraten Rosenin kirjaa [2, s. 817]. Koodausteorian perusideana on tutkia viestien koodaamista siten, että lähetyksen aikaisen kohinan aiheuttamat muutokset voidaan havaita ja korjata. Tutkielmassa *viestit* tulkitaan bittijonoiksi, sillä esimerkiksi ASCII on yleisesti käytössä oleva koodausstandardi, jossa kutakin merkkiä vastaa bittijono. Sellaisten bittijonojen joukosta joiden pituus on n käytetään merkintää B_n ja nimitystä *n-bittijono*.

Lähettämisen ja vastaanottamisen välillä koodisana saattaa muuttua lähetyksen aikana tapahtuneen kohinan johdosta. Purettu koodisana ei näin ollen vastaa alkuperäistä viestiä. Kohinasta aiheutuvaa koodisanan muutosta kutsutaan *virheeksi*.



Määritelmä 2.1. *Koodausfunktio* on injektio $E\colon B_m \to B_n$. Viestin $\mathbf{a} \in B_m$ koodaus on tällöin $E(\mathbf{a})$, joka tuottaa koodisanan. Kaikkien koodisanojen joukkoa $C \subseteq B_n$ kutsutaan *koodiksi*.

Määritelmä 2.2. Jos E on koodausfunktio, niin sen *purkufunktio* on sellainen funktio $D\colon B_n \to B_m \cup e$ $(m \leq n)$, että yhdistetty kuvaus $D \circ E$ on identtinen kuvaus, ja $D(\mathbf{b}) = e$ kaikilla ei-koodisanoilla $\mathbf{b}$.

2.1 Virheen huomaaminen ja korjaaminen

Luku 2.1 pohjautuu teokseen [1, s. 1–18]. Käydään ensiksi läpi tilanteita, joissa lähetyksen aikana tapahtuu yhden merkin virhe. Tällöin kohina muuttaa viestin yksittäisen bitin nollasta ykköseksi tai toisin päin. Esimerkiksi koodisana $\mathbf{a} = (0000)$ vastaanotetaan koodisanana $\mathbf{b} = (0001)$. Lähetyksen aikainen kohina on muuttanut koodisanan $\mathbf{a}$ neljännen bitin $0 \to 1$. Koska koodisanat $\mathbf{a}$ ja $\mathbf{b}$ eroavat yhden bitin kohdalta, puhutaan yhden merkin virheestä.

Lause 2.1. *Olkoon c viesti. Yhden merkin virhe voidaan havaita koodauksella $E\colon B^m \to B^{m+1}$, joka lisää viestin c loppuun pariteettibitin. Tällöin jokainen yhden bitin virhe johtaa virheilmoitukseen.*

Todistus (vrt. [1, s. 7]). Olkoon c koodisana. Jos koodisanan c ykkösbittien lukumäärä on parillinen, parititeettibitti on 0, ja jos ykkösbittien lukumäärä on pariton, pariteettibitti on 1. Yhden bitin muuttuminen lähetyksen aikana lisää tai vähentää koodisanan ykkösten lukumäärää yhdellä. Koska lähetetty koodisana on koodattu siten, että sen ykkösten lukumäärä on parillinen, muuttunut sana c ei ole koodisana ja täten johtaa virheilmoitukseen. □

Esimerkki 2.1 (vrt. [1, s. 10, Ex. 3])**.** Tarkastellaan muutamia vastaanotettuja 7-bittijonoja, jotka on koodattu käyttäen pariteettibittiä, ja katsotaan onko lähetyksen aikana tapahtunut virhettä.

Vastaanotettu koodisana	Dekoodattu viesti
110 011 0	110 011
001 011 1	001 011
000 110 1	e
101 011 1	e

Määritelmä 2.3. Olkoot $\mathbf{a}$ ja $\mathbf{b}$ n-bittijonoja. *Hamming-etäisyys* on funktio $H\colon B_n \times B_n \to \mathbb{Z}_+$, missä $H(\mathbf{a}, \mathbf{b})$ on bittijonojen $\mathbf{a}$ ja $\mathbf{b}$ toisistaan eroavien bittien lukumäärä.

Lause 2.2. *Hamming-etäisyys on metriikka, sillä se täyttää seuraavat ehdot kaikilla* $\mathbf{a}, \mathbf{b}, \mathbf{c} \in B_n$

1. $H(\mathbf{a}, \mathbf{b}) \geq 0$.
2. $H(\mathbf{a}, \mathbf{b}) = 0$*, jos ja vain jos* $\mathbf{a} = \mathbf{b}$.
3. $H(\mathbf{a}, \mathbf{b}) = H(\mathbf{b}, \mathbf{a})$.
4. $H(\mathbf{a}, \mathbf{b}) \leq H(\mathbf{a}, \mathbf{c}) + H(\mathbf{c}, \mathbf{b})$.

Todistus. Todistetaan kohdat 1 ja 4. Kohta 1 saadaan suoraan Hamming-etäisyyden määritelmästä. Bittijonojen $\mathbf{a}$ ja $\mathbf{b}$ eroavien bittien lukumäärä on aina positiivinen. Jokainen eroavaisuus kasvattaa etäisyyttä yhdellä.

Todistetaan seuraavaksi kohta 4. Jos bittijonot $\mathbf{a}$ ja $\mathbf{b}$ ovat identtiset, tällöin $H(\mathbf{a}, \mathbf{b}) = 0$. Kohdan 1 nojalla $H(\mathbf{a}, \mathbf{c}) \geq 0$ ja $H(\mathbf{c}, \mathbf{b}) \geq 0$, jolloin selvästi $H(\mathbf{a}, \mathbf{b}) \leq H(\mathbf{a}, \mathbf{c}) + H(\mathbf{c}, \mathbf{b})$.

Jos taas bittijonot $\mathbf{a}$ ja $\mathbf{b}$ eivät ole identtisisiä, niiden välinen Hamming-etäisyys $H(\mathbf{a}, \mathbf{b}) \geq 0$. Tällöin, jos bittijono $\mathbf{a}$ muutetaan ensin bittijonoksi $\mathbf{c}$ ja edelleen bittijonoksi $\mathbf{b}$, on muutettavien bittien määrä $H(\mathbf{a}, \mathbf{c}) + H(\mathbf{c}, \mathbf{b})$. Koska bittijono $\mathbf{a}$

muutettiin tässä bittijonoksi $\mathbf{b}$, niin bittimuunnoksia tehtiin yhteensä ainakin $H(\mathbf{a}, \mathbf{b})$.

□

Koodin kyky havaita ja korjata lähetyksen aikana sattuneita virheitä on riippuvainen sen minimietäisyydestä, joka määritellään seuraavaksi.

Määritelmä 2.4. Koodin *minimietäisyys* d on minimi joukosta $\{ H(\mathbf{a}, \mathbf{b}) \mid \mathbf{a}, \mathbf{b} \in B_n, \mathbf{a} \neq \mathbf{b} \}$.

Määritelmä 2.5. *Suurimman uskottavuuden purkumenetelmä* tulkitsee jokaisen ei-koodisanan sitä lähimpänä olevaksi koodisanaksi.

Määritelmän 2.5 menetelmä ei välttämättä toimi yksikäsitteisesti, sillä koodin sisällä kahden tai useamman koodisanan Hamming-etäisyys ei-koodisanasta voi olla yhtä suuri, jolloin suurimman uskottavuuden purkumenetelmä ei kykene valitsemaan niiden väliltä.

Esimerkki 2.2 (vrt. [1, s. 17, Ex. 4])**.** Tarkastellaan koodisanojen kokoelmaa {000000, 001110, 010101, 011011, 100011, 101101, 110110, 111000}, jossa minimietäisyys $d = 3$. Oletetaan, että lähetyksen aikana on tapahtunut enintään yksi virhe. Lasketaan vastaanotettujen viestien {110011, 110110, 111011, 001111, 011111, 100011} Hamming-etäisyydet annettuihin koodisanoihin seuraavassa taulukossa. Lauseessa 2.3 osoitetaan, että koodi pystyy havaitsemaan ja korjaamaan kaikki yhden bitin virheet, kun minimietäisyys $d \geq 3$. Vasemmassa reunassa ovat edellä määritetyt koodisanat ja kullakin pystyrivillä vastaanotetun viestin Hamming-etäisyydet eri koodisanoihin. Lyhyimmät etäisyydet ovat ympyröity.

	110011	110110	111011	001111	011111	100011
000000	4	4	5	4	5	3
001110	5	3	5	(1)	2	4
010101	3	3	4	3	2	4
011011	2	4	(1)	3	(1)	3
100011	(1)	3	2	3	4	(0)
101101	4	4	3	2	3	3
110110	2	(0)	3	4	3	3
111000	3	3	2	5	4	4

Taulukosta huomataan, että osassa viesteistä on tapahtunut virhe. Koska minimietäisyys on kolme ja oletettiin, että lähetyksen aikana voi tapahtua korkeintaan yksi

virhe, saadaan määritelmän 2.5 menetelmällä tulkinnaksi yksikäsitteinen koodisana, jonka Hamming-etäisyys vastaanotetusta viestistä on pienin. Huomataan myös, että osa koodisanoista lähetettiin useamman kerran.

Lause 2.3. *Olkoon d koodin minimietäisyys. Kun $d = 1$, koodi ei kykene havaitsemaan kaikkia yksittäisiä virheitä. Kun $d = 2$, koodi kykenee havaitsemaan, mutta ei korjaamaan, yksittäisen virheen.*

Todistus (vrt. [1, s. 14]). Tarkastellaan ensin tapausta $d = 1$. Olkoot $\mathbf{a}$ ja $\mathbf{b}$ sellaisia koodisanoja, että $H(\mathbf{a}, \mathbf{b}) = 1$. Koodisanat $\mathbf{a}$ ja $\mathbf{b}$ siis eroavat toisistaan yhden bitin kohdalla. Mikäli kyseinen bitti muuttuu lähetyksen aikana, niin koodisana $\mathbf{a}$ vastaanotetaan virheellisesti koodisanana $\mathbf{b}$ eikä virhettä huomata.

Tarkastellaan seuraavaksi tapausta $d = 2$. Olkoon $\mathbf{a}$ koodisana. Oletetaan, että koodisanan lähetyksen aikana tapahtuu yksi virhe. Tällöin vastaanotettu sana on ei-koodisana, koska koodisana $\mathbf{a}$ eroaa muista koodisanoista ainakin kahden bitin verran. Ei-koodisanan vastaaanottaminen aiheuttaa virheilmoituksen.

Tarkastellaan vielä tapausta, jossa tapahtuu yksittäinen virhe, mutta virhettä ei voida korjata, kun $d = 2$. Olkoot $\mathbf{b}$ ja $\mathbf{c}$ sellaisia koodisanoja, että $H(\mathbf{b}, \mathbf{c}) = 2$. Koodisanat $\mathbf{b}$ ja $\mathbf{c}$ siis eroavat toisistaan kahden bitin kohdalla. Oletetaan, että koodisanan $\mathbf{b}$ lähetyksen aikana tapahtuu yksittäinen virhe joka muuttaa toisen koodisanoja $\mathbf{b}$ ja $\mathbf{c}$ erottavista biteistä. Tällöin vastaanotettu bittijono $\mathbf{r}$ eroaa koodisanoista $\mathbf{b}$ ja $\mathbf{c}$ yhden bitin verran. Tästä johtuen koodi ei kykene valitsemaan kumpi koodisanoista, $\mathbf{b}$ vai $\mathbf{c}$, on alkuperäinen lähetetty koodisana. Virhe voidaan siis huomata, mutta ei aina korjata. □

Lause 2.4. *Olkoon $d \geq 3$ koodin minimietäisyys. Tällöin suurimman uskottavuuden purkumenetelmä korjaa kaikki yhden bitin virheet.*

Todistus (vrt. [1, s. 15]). Olkoon $\mathbf{c}$ lähetettävä koodisana. Lähetyksen aikana tapahtuu yksi virhe ja vastaanotetaan koodisana $\mathbf{r}$. Koska $\mathbf{c}$ muuttui yhden bitin kohdalta, Hamming-etäisyys $H(\mathbf{c}, \mathbf{r}) = 1$. Olkoon $\mathbf{c}'$ toinen koodisana siten, että $H(\mathbf{c}, \mathbf{c}') \geq 3$. Yhden bitin muuttuminen muuttaa koodisanan $\mathbf{c}$ koodisanaksi $\mathbf{r}$. Koodisanan $\mathbf{r}$ muuttaminen koodisanaksi $\mathbf{c}'$ vaatii vähintään kahden bitin muuttumista. Koodisana $\mathbf{c}$ voidaan muuttaa koodisanaksi $\mathbf{c}'$ koodisanan $\mathbf{r}$ kautta. Kolmioepäyhtälön nojalla

$$H(\mathbf{c}, \mathbf{c}') \leq H(\mathbf{c}, \mathbf{r}) + H(\mathbf{r}, \mathbf{c}'),$$

joten

$$H(\mathbf{r}, \mathbf{c}') \geq H(\mathbf{c}, \mathbf{c}') - H(\mathbf{c}, \mathbf{r}) = 3 - 1 = 2.$$

Koska $H(\mathbf{r}, \mathbf{c}') \geq 2$ ja $H(\mathbf{c}, \mathbf{r}) = 1$, suurimman uskottavuuden purkumenetelmä valitsee koodisanaksi koodisanan $\mathbf{c}$. □

Lause 2.5. *Olkoon $d \geq 3$ koodin minimietäisyys. Tällöin purkumenetelmä, joka kuvaa ei-koodisanat virheeksi, havaitsee yhden ja kahden bitin virheet.*

Todistus. Olkoot $\mathbf{c}$ ja $\mathbf{c}'$ sellaisia koodisanoja, joille $H(\mathbf{c}, \mathbf{c}') \geq 3$. Todistetaan, että koodi kykenee havaitsemaan yhden ja kahden bitin virheet. Lähetetään koodisana $\mathbf{c}$. Oletetaan, että lähetyksen aikana tapahtuu yksi tai kaksi virhettä. Vastaanotetaan viesti $\mathbf{r}$, jolle $H(\mathbf{c}, \mathbf{r}) = 1$ tai $H(\mathbf{c}, \mathbf{r}) = 2$. Koodin purkaja pystyy havaitsemaan virheen, sillä vastaanotettu koodisana $\mathbf{r}$ ei minimietäisyyden perusteella vastaa muita koodisanoja. Koska $H(\mathbf{c}', \mathbf{r}) \leq 2$ ja $H(\mathbf{c}, \mathbf{c}') \geq 3$, kolmioepäyhtälön nojalla

$$H(\mathbf{c}', \mathbf{r}) + H(\mathbf{r}, \mathbf{c}) \geq H(\mathbf{c}', \mathbf{c}),$$

joten

$$H(\mathbf{r}, \mathbf{c}) \geq H(\mathbf{c}', \mathbf{c}) - H(\mathbf{c}', \mathbf{r}) \geq H(\mathbf{c}, \mathbf{c}') - 2 = 1.$$

Täten nähdään, että koodin purkaja pystyy havaitsemaan kaikki yhden ja kahden bitin virheet. □

3 Ryhmäkoodit ja niiden muodostaminen

Luku 3 pohjautuu teokseen [1, s. 18–44].

3.1 Ryhmäkoodin ominaisuuksia

Käydään seuraavaksi läpi ryhmäkoodin määritelmä ja ryhmäkoodin ominaisuuksia. Kahden n-bittijonon summa määritellään laskemalla niiden toisiaan vastaavat bitit yhteen, eli $(a_1 \dots a_n) + (b_1 \dots b_n) = (a_1 + b_1 \cdots a_n + b_n)$. Määritellään bittien joukolle $\{0, 1\}$ laskutoimitus + asettamalla

+	0	1
0	0	1
1	1	0

Kutsutaan näin saatua ryhmää sen tyypillisellä nimellä, $\mathbb{Z}_2$.

Esimerkki 3.1. Muutama esimerkki bittijonojen yhteenlaskusta. Bittijonoille pätee

$$(0\ 0\ 0\ 0) + (1\ 1\ 1\ 1) = (0+1 \quad 0+1 \quad 0+1 \quad 0+1) = (1\ 1\ 1\ 1)$$
$$(1\ 0\ 0\ 1) + (1\ 0\ 0\ 1) = (1+1 \quad 0+0 \quad 0+0 \quad 1+1) = (0\ 0\ 0\ 0)$$
$$(0\ 1\ 0\ 1) + (1\ 0\ 1\ 0) = (0+1 \quad 1+0 \quad 0+1 \quad 1+0) = (1\ 1\ 1\ 1)$$
$$(1\ 1\ 0\ 0) + (1\ 0\ 1\ 0) = (1+1 \quad 1+0 \quad 0+1 \quad 0+0) = (0\ 1\ 1\ 0)$$

Määritelmä 3.1. *Ryhmäkoodi* on sellainen koodi, jossa kahden koodisanan summa on koodisana.

Määritelmä 3.2. Bittijonon **a** *paino* $W(\mathbf{a})$ on sen ykkösten lukumäärä.

Lause 3.1. *Olkoon* **c** *koodisana ryhmäkoodissa. Tällöin pätee*

$$\mathbf{c} + \mathbf{c} = 0$$

Erityisesti 0 on koodisana jokaisessa ryhmäkoodissa.

Todistus (vrt. [1, s. 22]). Merkitään pelkistä nollista koostuvaa jonoa $\mathbf{0} = (0 \dots 0)$. Yhteenlaskusäännön mukaan kahden saman bitin summa on 0, joten $\mathbf{c}+\mathbf{c} = \mathbf{0}$. Koska kyseessä on ryhmäkoodi, myös $\mathbf{c} + \mathbf{c}$ on koodisana, ja edelleen, $\mathbf{0}$ on koodisana. □

Apulause 3.1. Olkoon $\mathbf{a}$ n-bittijono. Silloin

$$W(\mathbf{a}) = H(\mathbf{a}, \mathbf{0}).$$

Todistus (vrt. [1, s. 22]). Koska bitti b eroaa bitistä 0 täsmälleen silloin kun $b = 1$, bittijonon $\mathbf{a}$ paino $W(\mathbf{a})$ on sama kuin sen Hamming-etäisyys bittijonosta $\mathbf{0}$. □

Apulause 3.2. Olkoot $\mathbf{a}$ ja $\mathbf{b}$ n-bittijonoja. Silloin

$$H(\mathbf{a}, \mathbf{b}) = W(\mathbf{a} + \mathbf{b}).$$

Todistus (vrt. [1, s. 23]). Yhteenlaskusäännön mukaan bittijono $\mathbf{a} + \mathbf{b}$ saa arvon 1 niissä kohdissa, joissa bittijonojen $\mathbf{a}$ ja $\mathbf{b}$ bitit eroavat, ja arvon 0 muulloin. Täten $H(\mathbf{a}, \mathbf{b}) = W(\mathbf{a} + \mathbf{b})$. □

Käydään seuraavaksi läpi esimerkki painoista. Samalla voidaan havainnollistaa apulausetta 3.2 esimerkin avulla.

Esimerkki 3.2. Olkoon $\mathbf{a} = (0\ 1\ 1\ 1\ 0)$ ja $\mathbf{b} = (1\ 1\ 0\ 0\ 1)$. Näiden Hamming-etäisyys $H(\mathbf{a}, \mathbf{b}) = 4$. Lasketaan summa $\mathbf{a} + \mathbf{b}$

$$\begin{array}{rccccc} & 0 & 1 & 1 & 1 & 0 \\ + & 1 & 1 & 0 & 0 & 1 \\ \hline & 1 & 0 & 1 & 1 & 1 \end{array}$$

Summan painoksi $W(\mathbf{a} + \mathbf{b})$ saadaan $W(1\ 0\ 1\ 1\ 1) = 4$, joka vastaan Hamming-etäisyyttä $H(\mathbf{a}, \mathbf{b}) = 4$.

Lause 3.2. *Olkoon d ryhmäkoodin A minimietäisyys. Silloin d on myös ryhmäkoodin nollajonosta poikkeavien koodisanojen painojen minimi, eli* $d = \min\{\, w(a) \mid a \in A, a \neq 0 \,\}$

Todistus (vrt. [1, s. 24]). Olkoon d' kaikkien nollasta poikkeavien koodisanojen painojen minimi. Todistetaan ensin, että $d \leq d'$.

Apulauseen 3.1 perusteella tiedetään, että jokaisen nollasta poikkeavan koodisanan paino on sama kuin sen Hamming-etäisyys nollavektorista. Tällöin painojen minimi d' on sama kuin joidenkin koodisanojen $\mathbf{a}, \mathbf{b} \in A, (\mathbf{a} \neq \mathbf{b})$ välisten Hamming-etäisyyksien minimi. Koska d on kaikkien koodisanojen välisten Hamming-etäisyyksien minimi, niin on oltava $d \leq d'$.

Todistetaan seuraavaksi, että $d' \leq d$. Olkoot $\mathbf{a}$ ja $\mathbf{b}$ eri koodisanoja. Tällöin $H(\mathbf{a}, \mathbf{b}) \neq 0$. Apulauseen 3.2 mukaan $H(\mathbf{a}, \mathbf{b}) = W(\mathbf{a} + \mathbf{b})$. Koska $H(\mathbf{a}, \mathbf{b}) \neq 0$,

myös $W(\mathbf{a}+\mathbf{b}) \neq 0$ ja $\mathbf{a}+\mathbf{b} \neq \mathbf{0}$. Koska tarkastellaan ryhmäkoodia, myös $\mathbf{a}+\mathbf{b}$ on koodisana. Minimietäisyys d voidaan siis laskea myös muotoa $\mathbf{a}+\mathbf{b}$ olevien koodisanojen painojen miniminä. Koska d' on kaikkien koodisanojen painojen minimi, niin $d' \leq d$. □

3.2 Ryhmäkoodin rakentaminen

Kahden n-bittijonon kertolasku määritellään kertomalla niiden toisiaan vastaavat bitit yhteen, eli $(a_1 \ldots a_n) \cdot (b_1 \ldots b_n) = (a_1 \cdot b_1 \cdots a_n \cdot b_n)$. Määritellään kertolasku $\cdot$ joukossa $\mathbb{Z}_2 = \{0, 1\}$:

$\cdot$	0	1
0	0	0
1	0	1

Määritelmä 3.3. Olkoot $\mathbf{a} = (a_1\ a_2\ \ldots\ a_n)$ ja $\mathbf{b} = (b_1\ b_2\ \ldots\ b_n)$ n-bittijonoja. Niiden *sisätulo* on joukon $\mathbb{Z}_2$ yhteen- ja kertolaskusääntöä noudattaen

$$\mathbf{a} \cdot \mathbf{b} = a_1 \cdot b_1 + a_2 \cdot b_2 + \cdots a_n \cdot b_n.$$

Tarkastellaan seuraavaksi matriiseja, joiden alkiot ovat bittejä. Sanotaan niitä *bittimatriiseiksi* tai *binäärimatriiseiksi*. Oletetaan tunnetuksi, että bittimatriisit muodostavat $\mathbb{Z}_2$-vektoriavaruuden.

Määritelmä 3.4. Olkoon $\mathbf{a} = (a_1\ a_2\ \ldots\ a_n)$ rivimatriisi. Sen *transpoosi* $\mathbf{a}^T$ on

$$\mathbf{a}^T = \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix}.$$

Yleisemmin ottaen $m \times n$ -matriisin A_{ij} *transpoosi* on $m \times n$ -matriisi A^T, missä

$$A^T_{ij} = A_{ji}.$$

kaikilla $i \in \{1, 2, \ldots, m\}$, $j \in \{1, 2, \ldots, n\}$.

Määritelmä 3.5. Olkoon $\mathbf{a}$ $n \times 1$ -rivimatriisi ja $\mathbf{b}$ $1 \times n$ -sarakematriisi. Tällöin *matriisitulo* $\mathbf{a} \cdot \mathbf{b}$ on 1×1 -matriisi, jonka ainoa alkio on sisätulo $\mathbf{a} \cdot \mathbf{b}^T$.

Apulause 3.3. Olkoot $\mathbf{a}$ ja $\mathbf{b}$ n-bittijonoja. Silloin

$$(\mathbf{a}+\mathbf{b})^T = \mathbf{a}^T + \mathbf{b}^T$$

Todistus. Olkoot $\mathbf{a} = (a_1\ a_2\ \dots\ a_n)$ ja $\mathbf{b} = (b_1\ b_2\ \dots\ b_n)$ n-bittijonoja. Bittijonojen $\mathbf{a}$ ja $\mathbf{b}$ summan transpoosiksi saadaan

$$\begin{aligned}(\mathbf{a}+\mathbf{b})^T &= ((a_1\ a_2\ \dots\ a_n) + (b_1\ b_2\ \dots\ b_n))^T \\ &= (a_1 + b_1\ a_2 + b_2\ \cdots\ a_n + b_n)^T \\ &= \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_n + b_n \end{pmatrix},\end{aligned}$$

ja transpoosien summaksi saadaan

$$\begin{aligned}\mathbf{a}^T + \mathbf{b}^T &= (a_1\ a_2\ \dots\ a_n)^T + (b_1\ b_2\ \dots\ b_n)^T \\ &= \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{pmatrix} + \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix} \\ &= \begin{pmatrix} a_1 + b_1 \\ a_2 + b_2 \\ \vdots \\ a_n + b_n \end{pmatrix}.\end{aligned}$$

Nähdään, että $(\mathbf{a}+\mathbf{b})^T = \mathbf{a}^T + \mathbf{b}^T$. □

Lause 3.3. *Olkoon H $r \times n$ -binäärimatriisi. Oletetaan, että koodi koostuu n-bittijonoista $\mathbf{c}$, joille pätee $H \cdot \mathbf{c}^T = \mathbf{0}^T$. Tällöin koodi on ryhmäkoodi.*

Todistus (vrt. [1, s. 36]). Määritelmän 3.1 mukaan koodissa, joka on ryhmäkoodi, kahden koodisanan summa on koodisana. Olkoot $\mathbf{b}$ ja $\mathbf{c}$ koodisanoja. Tällöin

$$H \cdot \mathbf{b}^T = \mathbf{0}^T \quad \text{ja} \quad H \cdot \mathbf{c}^T = \mathbf{0}^T.$$

Koodi on ryhmäkoodi, jos $H \cdot (\mathbf{b}+\mathbf{c})^T = \mathbf{0}^T$. Näin on, sillä apulauseen 3.3 ja

osittelulain nojalla

$$\begin{aligned} H \cdot (\mathbf{b} + \mathbf{c})^T &= H \cdot (\mathbf{b}^T + \mathbf{c}^T) \\ &= H \cdot \mathbf{b}^T + H \cdot \mathbf{c}^T \\ &= \mathbf{0}^T + \mathbf{0}^T \\ &= \mathbf{0}^T. \end{aligned}$$

□

Määritelmä 3.6. Olkoon H $m \times n$ -binäärimatriisi. Matriisin H *ydin* on kokoelma n-jonoja $\mathbf{c}^T$, jotka toteuttavat ehdon $H \cdot \mathbf{c}^T = \mathbf{0}^T$.

Määritelmä 3.7. Olkoon H $r \times n$ -binäärimatriisi. Oletetaan, että $n \geq r$, ja että matriisin H viimeiset r pystysaraketta muodostavat $r \times r$ -yksikkömatriisin. Tällöin H on *kanoninen pariteettimatriisi*.

Lause 3.4. *Olkoon H kanoninen $r \times n$ -pariteettimatriisi. Olkoon $\mathbf{c}$ jokin matriisin H rivi. Tällöin matriisin H ydin koostuu informaatiosta ja tarkistusbiteistä, joista ensimmäiset $(n - r)$ bittiä ovat mielivaltaisia ja loput r bittiä on määritetty ehdon $H \cdot \mathbf{c}^T = \mathbf{0}^T$ mukaan. Tällöin viimeiset r bittiä toimii tarkistusbitteinä biteille $n - r$. Rivimatriisin $\mathbf{c}$ ensimmäiset $n - r$ bittiä ovat informaatiota ja loput r bittiä sen tarkistusbittejä.*

Lauseen 3.4 todistus sivuutetaan, mutta havainnollistetaan sitä seuraavalla esimerkillä.

Esimerkki 3.3. Olkoon $H = \left(\begin{smallmatrix} 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 \end{smallmatrix}\right)$ ja bittijono $\mathbf{c} = (c_1\ c_2\ c_3\ c_4\ c_5)$. Tällöin

$$H \cdot \mathbf{c}^T = \begin{pmatrix} c_2 + c_3 + c_4 \\ c_1 + c_3 + c_5 \end{pmatrix}.$$

Lauseen 3.4 mukaan c_4 on tarkistusbitti biteille c_2 ja c_3, ja c_5 on tarkistusbitti biteille c_1 ja c_3. Toisin sanoen bitit c_1, c_2 ja c_3 ovat bittijonon $\mathbf{c}$ informaatiobittejä ja bitit c_4 ja c_5 sen tarkistusbittejä. Ehdon $H \cdot \mathbf{c}^T = \mathbf{0}^T$ toteuttavat seuraavat koodisanat:

$$\begin{array}{cccc} (0\,0\,0\,0\,0) & (0\,0\,1\,1\,1) & (0\,1\,0\,1\,0) & (0\,1\,1\,0\,1) \\ (1\,0\,0\,0\,1) & (1\,0\,1\,1\,0) & (1\,1\,0\,1\,1) & (1\,1\,1\,0\,0) \end{array}$$

Koska saadun koodin minimietäisyys $d = 2$, kyseessä on yksittäisen virheen tunnistava ryhmäkoodi.

4 Virheenkorjauskoodin muodostaminen

Luku 4 pohjautuu teokseen [1, s. 44–61].

4.1 Yksinkertainen virheenkorjauskoodi

Apulause 4.1. Olkoon $\mathbf{e}_i$ n-bittijono, jonka paino $W(\mathbf{e}_i) = 1$ ja indeksin i bitti on 1. Olkoon H $r \times n$ -binäärimatriisi. Tällöin $H \cdot \mathbf{e}_i^T$ vastaa matriisin H saraketta kohdassa i.

Lause 4.1. *Olkoon H binäärimatriisi. Silloin matriisin H ydin muodostaa ryhmäkoodin, joka tunnistaa yksittäiset virheet, jos ja vain jos yksikään matriisin H sarakkeista ei ole* $\mathbf{0^T}$.

Todistus (vrt. [1, s. 45]). Oletetaan, että matriisin H ydin tunnistaa yksittäiset virheet. Lauseen 2.3 nojalla minimietäisyys on tällöin $d \geq 2$. Toisaalta, minimietäisyys on nollasta eroavien koodisanojen painojen minimi. Jokaisesta nollasta eroavan koodisanan $\mathbf{c}$ paino on siis vähintään kaksi. Koodisanat $\mathbf{c}$ ovat niitä sanoja, jotka toteuttavat ehdon $H \cdot \mathbf{c}^T = \mathbf{0}^T$. Koska bittijonot $\mathbf{e}_i$ eivät ole koodisanoja, niin on oltava $H \cdot \mathbf{e}_i^T \neq \mathbf{0}^T$, eli apulauseen 4.1 perusteella sarakkeet eroavat nollasta.

Oletetaan sitten, ettei yksikään sarakkeista ei ole $\mathbf{0}^T$. Tällöin $H \cdot \mathbf{e}_i \neq \mathbf{0}^T$, joten yksikään $\mathbf{e}_i$ ei ole koodisana. Siispä pienin mahdollinen nollasta eroavan koodisanan paino on 2. □

Lause 4.2. *Olkoon H binäärimatriisi. Silloin matriisin H ydin muodostaa ryhmäkoodin, joka korjaa yksittäiset virheet jos, ja vain jos yksikään matriisin H sarakkeista ei ole* $\mathbf{0}^T$ *eikä matriisi H sisällä kahta identtistä saraketta.*

Todistus (vrt. [1, s. 47]). Tarkastellaan matriisin H ytimen muodostamaa koodia ja sen minimietäisyyttä d. Osoitetaan tapauksissa $d = 1$ ja $d = 2$, että molemmat ehdot ovat epätosia.

Kun $d = 1$, niin lauseen 2.3 nojalla matriisi H ei havaitse eikä korjaa yksittäisiä virheitä. Lauseen 4.1 nojalla ainakin yksi matriisin H sarakkeista on $\mathbf{0}^T$.

Oletetaan, että $d = 2$. Silloin lauseen 2.3 mukaan koodi ei kykene korjaamaan yksittäisiä virheitä. Koska $d = 2$, niin on olemassa koodisana $\mathbf{e}_{ij}$, missä indeksien i ja j bitit ovat ykkösiä ja loput nollia, $i \neq j$. Huomataan, että

$$\begin{aligned} H \cdot \mathbf{e}_{ij}^T &= H \cdot (\mathbf{e}_i + \mathbf{e}_j)^T \\ &= H \cdot \mathbf{e}_i^T + H \cdot \mathbf{e}_j^T \\ &= \mathbf{0}^T. \end{aligned}$$

Täten $H \cdot \mathbf{e}_i = H \cdot \mathbf{e}_j$ ja matriisi H sisältää kaksi identtistä saraketta.

Oletetaan, että $d \geq 3$. Lauseen 2.4 nojalla koodi kykenee korjaamaan kaikki yksittäiset virheet. Matriisilla H ei siten voi olla nollasaraketta eikä kahta identtistä saraketta sillä tällöin $d \leq 2$. □

4.2 Hamming-koodin muodostaminen

Määritelmä 4.1. Olkoon H $r \times n$ -binäärimatriisi. Olkoon $\mathbf{r}$ n-jono. Silloin n-jonon $\mathbf{r}$ *syndrooma* on $r \times 1$ -sarakematriisi $H \cdot \mathbf{r}^T$.

Apulause 4.2. Olkoon koodina $r \times n$ -matriisin H ydin ja olkoon $\mathbf{r}$ vastaanotettu n-jono. Lähetetyn ja vastaanotetun koodisanan välillä on yhteys $\mathbf{r} = \mathbf{c} + \mathbf{e}$, missä $\mathbf{c}$ on lähetetty koodisana ja $\mathbf{e}$ vastaa lähetyksen aikana tapahtunutta virhettä. Tällöin vastaanotetun viestin $\mathbf{r}$ syndrooma $H \cdot \mathbf{r}^T$ on virheen $\mathbf{e}$:n syndrooma $H \cdot \mathbf{e}^T$. Erityisesti vastaanotetun viestin r syndrooma ei riipu koodisanasta c.

Todistus (vrt. [1, s. 54]). Osittelulain perusteella

$$\begin{aligned} H \cdot \mathbf{r}^T &= H \cdot (\mathbf{c} + \mathbf{e})^T \\ &= H \cdot \mathbf{c}^T + H \cdot \mathbf{e}^T \\ &= \mathbf{0}^T + H \cdot \mathbf{e}^T \\ &= H \cdot \mathbf{e}^T \end{aligned}$$

Näin on osoitettu, että vastaanotetun viestin syndrooma riippuu ainoastaan lähetyksen aikaisen virheen syndroomasta. □

Lause 4.3. *Olkoon H $r \times n$ binäärimatriisi, jonka ydin vastaa yhden virheen korjaavan koodin koodisanoja. Olkoon $\mathbf{r}$ vastaanotettu viesti, jonka lähetyksen aikana on tapahtunut korkeintaan yksi virhe. Viesti vastaanotettiin virheettömästi, jos n-jonon $\mathbf{r}$ syndrooma on $\mathbf{0}^T$. Muutoin vastaanotetun viestin $\mathbf{r}$ syndrooma vastaa jotakin matriisin H saraketta, jonka indeksi i kertoo virheen paikan vastaanotetussa n-jonossa $\mathbf{r}$.*

Todistus (vrt. [1, s. 55]). Selvästi $H \cdot \mathbf{r}^T = \mathbf{0}^T$ silloin, kun lähetyksen aikana ei tapahdu yhtään virhettä. Muissa tapauksissa $\mathbf{r} = \mathbf{c} + \mathbf{e}$, missä $\mathbf{c}$ vastaa koodisanaa ja $\mathbf{e}$ virhettä. Yhden virheen tapauksessa $\mathbf{e} = \mathbf{e}_i$ jollakin i. Apulauseen 4.2 mukaan n-jonojen $\mathbf{e}$ ja $\mathbf{r}$ syndroomat ovat samat. Lauseen 4.2 mukaan matriisissa H ei ole kahta identtistä saraketta ja apulauseen 4.1 mukaan syndrooma on sarake i matriisissa H. Koska matriisissa H ei ole kahta samanlaista saraketta ja syndrooma on sen sarake kohdassa i, virhe on tapahtunut indeksin i bitissä. □

Lähteet

[1] Laufer, H. *Discrete Mathematics and Applied Modern Algebra*. BWS Publishers, 1984.

[2] Rosen, K. *Discrete Mathematics and Its Applications, Seventh Edition*. McGraw-Hill, 2013.