Nem bináris kódok konstrukciója

GYÖRFI LÁSZLÓ
VAJDA ISTVÁN
BME Híradástechnikai Elektronika Intézet

ÖSSZEFOGLALÁS

Ebben a közleményben a nem-bináris Hamming kódok és a Reed-Solomon kódok konstrukcióját mutatjuk be alkalmazási példákkal.

1. Bevezetés

Mint minden igazán értékes és nagyszerű dolog a világon, így a hibajavító kódok elmélete is, előbb vagy utóbb elmondható egyszerűen is. A továbbiakban megmutatjuk, hogy a Reed-Solomon kódok, mint több hibát javító, nem-bináris és optimális kódok konstrukciója elsajátítható minden különösebb matematikai érdeklődés és előképzettség nélkül. Érdekes módon, ha közvetlenül ezeknek a kódoknak a tanulmányozását tűzzük ki célul, akkor elkerülhetjük azokat a területeket (BCH kód, primitív polinom stb.), melyek jellegzetesen elvesztik még egy újdonságokra fogékony mérnök kedvét is.

Az 1980-as évek végéig a kódelmélet alkalmazási területe igen szűk volt, mivel az átvitel megbízhatóságát vagy a csatorna javításával (adóteljesítmény-, sávszélességnövelés stb.) vagy visszacsatolás kiépítésével (ARQ) növelték. Ez utóbbi eredményezte a hibadetekció (CRC) tömeges alkalmazását. Az 1980-as években viszont egyrészt a technológiai lehetőségek jelentősen bővültek, másrészt a tömegszolgálatokban jelentkeztek olyan feladatok, ahol nincs visszacsatolás, tehát a hibadetekció önmagában semmit sem ér, hanem hibajavító kódolásra van szükség. A későbbiekben erre gyakorlati példákat is mutatunk.

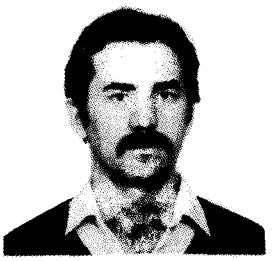
Ez a közlemény az [1] tanulmány egy részének a kibővítése. Nem tárgyaljuk a hibajavító kódok elméletének olyan fontos területét, mint a ciklikus kódok, bár a bemutatandó kódok ciklikusak. Nem beszélünk továbbá különböző dekódoló eljárásokról, mivel az igazán hatékonyakat nem tudjuk bonyolult algebrai apparátus bevezetése nélkül bemutatni. Ha valakit a téma ezek után mélyebben érdekel, akkor annak ajánljuk a [2]-[10] könyveket.

A [17] és [18] cikkek a hibajavító kódelmélet alkalmazásának és a várható technológiai fejlődést figyelembe vevő alkalmazhatóságának a trendjét elemzik a 80-as és a 90-es évekre vonatkozólag. Különös tekintettel mérlegelik a két legfontosabb irányzatot: a konvolúciós kódokat Viterbi dekódolással illetve a Reed-Solomon kódok hatékony felhasználási területeit.

Beérkezett: 1988. IX. 9. (H)

GYÖRFI LÁSZLÓ

1970-ben matematika-fizika szakos tanári oklevelet, 1974-ben egyetemi doktori címet, 1978-ban kandidátusi, 1988-ban tudomány doktora fokozatot szerzett. 1970-től 1975-ig a Távközlési Kutató Intézetben, azóta az MTA Informatikai Tanszéki Kutatócsoportban dolgozik. Fő érdeklődési területe a nemparaméteres becsléselmélet és a többszörös hozzáférésű csatornák kódolása.

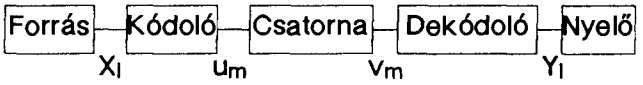


VAJDA ISTVÁN

1976-ban villamosmérnöki oklevelet, 1978-ban egyetemi doktori fokozatot 1984-ben kandidátusi oklevelet szerzett. 1978-tól az MTA Informatikai és Elektronikai Tanszéki Kutatócsoportban dolgozik. Fő érdeklődési területe a hibakorlátozó kódolás, a kódosztásos többszörös hozzáférésű csatornák kódolása valamint az algoritmikus adatvédelem.

2. Kódolási alapfogalmak

A hibajavító kódolás alapvető módszereit a következő egyszerű hírközlési struktúra kapcsán vizsgáljuk



X_l és Y_l sorozat elemei egy F halmazból veszik értékeiket, mely halmazt forrás-*ABC*-nek nevezzük. A kódoló az X_l sorozat egy szegmensét, azaz egy vektort, képez le az u_m sorozat egy szegmensébe, azaz egy vektorba. Az u_m értékeit egy véges G halmazból veszi, melyet kód-ABC-nek vagy csatorna bemeneti *ABC*-nek fogunk hívni. A csatorna kimenete v_m szintén *G*-ből veszi az értékeit. Azt mondjuk, hogy az m-edik időpontban a csatorna hibázott, ha $u_m \neq v_m$.

Egy

$$u=(u_1,\ldots u_n)$$

bemeneti és

$$v=(v_1,\ldots v_n)$$

kimeneti sorozat esetén jelölje d(u,v) azon i pozíciók számát, ahol $u_i \neq v_i$.

d(u, v) az u,v sorozatok Hamming távolsága, ós azt mondjuk, hogy az u sorozat küldésekor és a sorozat vételekor a hibák száma t=d(u, v). Ezt az esetet nevezzük egyszerű hibázásnak, amikor a hiba helye ós értéke egyaránt ismeretlen. d(u, v) valóban távolság, hiszen

$$d(u, v) \geq 0,$$
$$d(u, v) = d(v, u)$$

ós igaz a háromszög egyenlőtlenség:

$$d(u, v) \leq d(u, w) + d(w, v).$$

Kód (blokk-kód) alatt a G^n halmaz egy C részhalmazát értjük, azaz C minden eleme egy n hosszú vektor, melynek koordinátái G-beliek. C elemeit kódszavaknak nevezzük. A kódolás egy invertálható függvény, mely k hosszú F-beli sorozatot — üzenetet — képez le egy kódszóba, formalizálva

$$f: F^k \to C$$

ós minden különböző x,x'-re f(x), f(x') is különböző.

Dekódolás alatt két függvény egymásutánját értjük. Az egyik a csatorna kimenetének n hosszú szegmensét képezi le C-be, azaz igyekszik eltalálni a küldött kódszót, a másik pedig az f függvény inverze, tehát

$$g: G^n \to C, \qquad f^{-1}: C \to F^k$$

Mivel f egyértelműen meghatározza f^{-1}-t, ezért dekódolás alatt a későbbiekben csak a g függvényt értjük. A g dekódoló függvényként az algebrai hibajavító kódok elméletében speciális függvényt választunk, nevezetesen a v vektorhoz megkeressük azt a $c' \in C$ kódszót, mely Hamming távolság szerint hozzá a legközelebb van, vagy ha több ilyen van, akkor az egyiket, tehát teljesül hogy ha $c' = g(v)$, akkor $d(c', v) = \min_{c \in C} d(c, v)$.

A dekódolás feladata ezek után arra a messze nem triviális feladatra szűkül, hogy egy v vett szóhoz hogyan keressük meg a hozzá legközelebbi c' kódszót anélkül, hogy minden d(c, v)-t kiszámítanánk.

A későbbiekben kiderül, hogy a kódoló f függvény leglényegesebb tulajdonsága a C kód paramétere, amit kódtávolságnak nevezünk, ós d_{min}-nel jelölünk:

$$d_{min} = \min_{c \neq c' \; c, c' \in C} d(c, c').$$

Hibajelzés a hibakorlátozó kódolás azon feladata, amikor a vevőben csupán detektálni akarjuk a hibázás tényét. Nyilván egy v vett szó esetén akkor tudjuk a hibázást észrevenni, ha v nem kódszó, amire garancia, hogy ha c küldött kódszó esetén

$$d_{min} > d(v, c)$$

azaz a hibák számára

$$d_{min} > t,$$

tehát egy d_{min} kódtávolságú kód minden legfeljebb $d_{min} - 1$ számú hibát jelezni tud.

Hibajavítás esetén azt kérdezzük, hogy ha t a hibák száma, akkor mi biztosítja, hogy a v vett szóból a c kódszó egyértelműen visszaállítható legyen, azaz minden más c' kódszóra.

$$d(v, c') > d(v, c) \qquad (2.1)$$

legyen. Mivel a Hamming távolság valóban távolság, ezért teljesíti a háromszögegyenlőtlenséget, azaz

$$d(v, c') \geq d(c, c') - d(v, c) \qquad (2.2)$$

tehát (2.1) úgy biztosítható, hogy

$$d(c, c') - d(v, c) > d(v, c),$$

azaz minden $c' \neq c$-re

$$d(c, c') > 2\, d(v, c)$$

vagyis

$$d_{min}/2 > d(v, c).$$

Összefoglalva: egyszerű hibázás esetén minden legfeljebb $\mathrm{int}((d_{min} - 1)/2)$ számú hiba javítható.

Gyakran fordul elő olyan hibázás, amikor tudjuk, hogy egy pozícióban hiba lehet, vagyis tudjuk, hogy más pozícióban nincs hiba, tehát a hiba helyét ismerjük, csak a hiba értékét nem. Az ilyen hibát törléses hibának nevezzük. Egyszerűen belátható, hogy minden $d_{min} - 1$ törléses hiba javítható, ugyanis a legrosszabb esetben sem fordulhat elő, hogy két c, c' kódszó ugyanazon, de legfeljebb $d_{min} - 1$ pozíciójának törlésével ugyanazt a szót kapnánk.

Singleton korlát: Egy M kódszóból álló, n hosszú és d_{min} kódtávolságú kódra

$$M \leq q^{n - d_{min} + 1},$$

ahol q a kód-ABC elemszáma. Ezt bizonyítandó legyen k egy természetes szám, melyre

$$q^{k-1} < M \leq q^k.$$

Mivel a k – 1 hosszú különböző sorozatok száma q^{k-1}, ezért $q^{k-1} < M$ miatt létezik két kódszó c és c', melyek az első k – 1 koordinátában megegyeznek. Ezekre

$$d(c, c') \leq n - k + 1,$$

következésképp $d_{min} \leq n - k + 1$, azaz

$$M \leq q^k \leq q^{n - d_{min} + 1}$$

Azon kódot, melyre a Singleton korlátban = áll MDS (maximum distance separable) kódnak nevezzük. Ha a kódolandó üzenetek k hosszú vektorok, ahol a koordináták q lehetséges értéket vesz-

nek fel, akkor $M = q^k$, tehát a Singleton korlát $d_{min} \leq n - k + 1$.

3. Bináris lineáris kódok, bináris Hamming kód

A továbbiakban a kódjainkban szereplő kódszavakat alkotó szimbólumok legyenek 0 vagy 1 értékűek, az összeadás és a szorzás pedig a bináris összeadás és a bináris szorzás, vagyis a mod 2 összeadás és szorzás.

Egy bináris C kódot lineárisnak nevezünk, ha $0 \in C$ és a C halmaz lineáris tér, azaz ha minden $c, c' \in C$-re $c + c' \in C$.

A lineáris kódok jelentőségét az adja, hogy az egyes üzenetekhez tartozó kódszavak viszonylag egyszerűen generálhatók, és ugyancsak egyszerű módszer található a vett kódszavak hibamentességének vizsgálatára, vagyis a hibadetektálásra, és a hibák javítása sem bonyolult. A következőkben e módszereket fogjuk bemutatni.

Jelentsen C továbbra is egy lineáris kódot, a kódszóhossz legyen n. (Ekkor C az n hosszúságú bináris elemeket tartalmazó sorozatok terének egy altere; "kódszó" helyett gyakran "vektor"-t fogunk mondani.)

A valós vektortérben megszokott lineáris függetlenség ós bázis fogalmak itt is teljesen hasonlóan értelmezhetők, vagyis a $g_1, g_2, \ldots, g_j \in C$ vektorok lineárisan függetlenek, ha $\alpha_l \in \{0, 1\}$ mellett

$$\sum_{l=1}^{j} \alpha_l g_l = 0$$

csak úgy állhat elő, ha $\alpha_l = 0$ minden $l = 1, 2, \ldots, j$-re. A $g_1, g_2, \ldots, g_k \in C$ vektorok a C lineáris tér egy bázisát alkotják, ha lineárisan függetlenek, továbbá igaz az, hogy minden $c \in C$ vektor előállítható

$$c = \sum_{l=1}^{k} u_l g_l \qquad (3.1)$$

alakban, ahol $u_l \{0,1\}$ minden $l = 1, 2, \ldots, k$-ra.

A (3.1) egyenlőség fölírható mátrixalakban:

$$c = uG, \qquad (3.2)$$

ahol $u = (u_1, u_2, \ldots, u_k)$, G pedig a bázisvektorokból mint sorvektorokból álló mátrix. A (3.2.) egyenlettel tehát egy k dimenziós ós egy n-dimenziós vektort rendelünk össze lineáris transzformációval mégpedig kölcsönösen egyértelmű módon. Azt fogjuk mondani, hogy az u üzenethez a c kódszó tartozik.

A k-dimenziós u vektorokkal 2^k-féle üzenetet fejezhetünk ki, s ezeket kódolhatjuk a C kóddal. C elemei azonban n-dimenziós vektorok, és n nem kisebb k-nál, hiszen k az n-dimenziós vektorok C alterének dimenzlószáma. A $k = n$ esetnek nincs most jelentősége, ha k kisebb, mint n, akkor viszont világos, hogy nem minden vektort kell felhasználni kódszónak, vagyis kódunk redundáns lesz, s ezt a redundanciát tudjuk hibajavításra felhasználni.

Az üzenetekhez a kódszavakat a G mátrix segítségével rendeljük hozzá, vagyis a G mátrix jelöli ki az n-dimenziós vektortérnek a kódot jelentő C alterót, a kódot G "generálja". A fenti tulajdonságú G mátrixot a C kód *generátormátrixának* nevezzük.

Vegyük észre, hogy ha nem törődünk azzal, hogy melyik kódszó melyik üzenethez tartozik, csak a kódszavak halmazát tekintjük, akkor G nem egyértelmű, vagyis több mátrix is generálhatja ugyanazt a C kódszóhalmazt. Egy (n, k) paraméterű lineáris kód szisztematikus, ha minden kódszavára igaz, hogy annak utolsó n-k szimbólumát elhagyva éppen a neki megfelelő k hosszúságú üzenetet kapjuk.

A 2. pontban már leszögeztük, hogy dekódolás alatt csak az esetleges hibák javítását értjük, aminek eredményeképp egy kódszót kapunk. Az üzenetvektor visszanyeréséhez még el kell ugyan végezni a kódolás inverz műveletét, ez azonban rendszerint triviális lépés, szisztematikus kód esetén például csak el kell hagyni a kódszó egy részét.

Ilyenkor (tehát szisztematikus kód esetén) a generátormátrix is egyértelmű, mégpedig

$$G = (I_k, B) \qquad (3.3)$$

alakú, ahol I_k a k-szor k-as egységmátrix, B pedig k-szor (n – k) méretű mátrix. Az u üzenethez tartozó c kódszó szerkezete tehát:

$$c = (u_1, u_2, \ldots, u_k, c_{k+1}, c_{k+2}, \ldots, c_n).$$

A c első k koordinátájából álló szegmensét üzenetszegmensnek, az utolsó n – k koordinátájából állót paritásszegmensnek nevezzük.

A következőkben olyan észrevételeket fogunk tenni, amelyek elvezetnek az ígért egyszerű hibadetektáláshoz illetve hibajavításhoz.

Ha egy n – k sorból és n oszlopból álló H mátrixra

$$Hc^T = 0$$

akkor ós csak akkor, ha $c \in C$, akkor H-t a C kód paritás ellenőrző mátrixának nevezzük. (Röviden paritásmátrixot fogunk mondani.)

H segítségével, tehát meg tudjuk állapítani, hogy egy vett szó valóban a kódszó-e.

3.1. Tétel: Ha G és H ugyanazon C lineáris kód generátormátrixa illetve paritásmátrixa, akkor

$$HG^T = 0.$$

Bizonyítás: Jelölje Q^k a k hosszú bináris sorozatok halmazát. Ekkor minden $u \in Q^k$-hoz létezik $c \in C$, amire $c = uG$, ós $c \in C$ miatt

$$Hc^T = 0,$$

azaz

$$Hc^T = H(uG)^T = HG^T u^T = 0.$$

Az utolsó egyenlőség pedig csak úgy állhat fönn minden $u \in G^K$-ra, ha $HG^T = 0$, amint állítottuk.

Nézzük milyen alakú lehet H, ha kódunk szisztematikus. Ekkor tudjuk, hogy

$$G = (I_k, B)$$

alakú, keressük H-t

$$H = (A, I_{n-k})$$

alakban. A feltétel tehát:

$$HG^T = (A, I_{n-k})(I_k, B)^T = A + B^T = 0$$

Azaz

$$A = -B^T$$

kell teljesüljön (Bináris esetben $-B^T = B^T$.)

Egy c vektor súlya a koordinátái között levő nem nulla elemek száma, jelölése w(c). Egy C kód minimális súlyán a

$$w_{min} = \min_{c \in C, c \neq 0} w(c)$$

számot értjük.

3.2. Tétel: Egy C lineáris kódra

$$d_{min} = w_{min}$$

Bizonyítás:

$$d_{min} = \min_{c \neq c'} d(c, c') = \min_{c \neq c'} w(c - c') = \min_{c \neq 0} w(c) = w_{min}$$

ahol az utolsó előtti egyenlőség felírásakor a C kód linearitását használtuk ki, ebből következik ugyanis, hogy c–c' is kódszó, továbbá, az is, hogy minden kódszó előáll ilyen különbség alakjában. (Utóbbi ahhoz szükséges, hogy a minimum képzésekor valóban minden $o \in C$ -t figyelembe vehessük.)

A 3.2. Tétel jelentősége abban áll, hogy segítségével a d_{min} definíció alapján történő kiszámításához szükséges $|C|(|C|-1)/2$ műveletet a w_{min} kiszámításához szükséges $|C| - 1$ műveletre redukálhatjuk. ($|C|$ -vel a C elemszámát jelöljük.)

A következőkben azt mutatjuk meg, hogyan használható a H mátrix a dekódoláskor.

Legyen az adott kódszó c, a vett szó v. Az e = v–c vektort hibavektornak nevezzük. Vegyük észre, hogy

$$Hv^T = H(c + e)^T = Hc^T + He^T = He^T$$

vagyis a Hc^T értéke csak a hibavektortól függ, az adott kódszótól nem. Az $s = Hv^T$ mennyiséget szindrómának nevezzük.

A fentiek alapján a dekódolás a következőképpen mehet végbe: a vett v szóból kiszámítjuk az $s = Hv^T = He^T$ szindrómát, ennek alapján megbecsüljük a hibavektort, s ezt v-ből levonva megkapjuk a kódszóra vonatkozó becslésünket.

A dekódolás azon lépését, amikor s-ből megbecsüljük e-t legegyszerűbben az ún. táblázatos dekódolás esetén láthatjuk át. Ez azt jelenti, hogy a dekódernek rendelkezésére áll egy olyan táblázat, melynek elemei hibavektorok, és az azonos szindrómát előidézők (tehát, amikre He^T ugyanaz) rendre ugyanazon sorban helyezkednek el. A sorok első elemei az adott sorban minimális súlyú hibavektorok. A dekóder a szindróma ismeretében kiválasztja az annak megfelelő sort, és úgy dönt, hogy az adott sor első eleme, vagyis a legkisebb súlyú hibavektor fordult elő.

Illusztrációként egy klasszikusnak számító kódot mutatunk be, mely bináris Hamming kód néven ismeretes.

Olyan kódot keresünk, mely egy hibát tud javítani, vagyis ha c-t adjuk, ós v-t vesszük, akkor $1 \geq d(c, v)$ esetén biztosan meg tudjuk mondani v ismeretében c-t. Megkívánjuk még kódunktól, hogy lineáris és bináris legyen.

A hibajavítás céljára r bitet kívánunk felhasználni, vagyis az n kódszóhossz ós a k üzenethossz különbségét r-nek rögzítjük. Ezen adott r mellett szeretnénk a lehető legnagyobb k-t elérni, hogy minél több üzenetünk lehessen. Legyen a majdani kód paritásmátrixa:

$$H = (a_1^T, a_2^T, \ldots, a_n^T)$$

Legfeljebb egy hiba esetén az e hibavektor vagy 0, vagy egységvektor, tehát pontosan egy koordinátája 1-es. Ekkor az s szindróma vagy 0, vagy valamelyik a_i-vel egyenlő. Akkor és csak akkor tudjuk tehát e-t (ós így c-t is) egyértelműen megállapítani, ha a_i-k mind különbözők, és egyik sem 0. Mivel H soralnak a száma $n - k = r$, az a_i-k legfeljebb $2^r - 1$-félék lehetnek, ha egyikük sem 0. Mivel minél nagyobb k-ra, ós így a rögzített r révén minél nagyobb $k + r = n$-re törekszünk, mind a $2^r - 1$ lehetőséget használni fogjuk. Ezzel lényegében megadtuk H-t, hiszen megmondtuk, hogy H az összes lehetséges r hosszúságú nem nulla oszlopvektorból álló mátrix. Ez pedig már definiálja a kódszóhalmazt a

$$Hc^T = 0$$

egyenletrendszer összes megoldásvektorának halmazaként. Az így kapott kódot nevezzük Hamming kódnak, mely tehát k hosszú üzenethez n hosszú kódszót rendel, ahol n és k között fennáll az

$$n = 2^{n-k} - 1 \qquad (3.4)$$

összegfüggés. Ilyen tulajdonságú számpárok a következők:

n =	k =
3	1
7	4
15	11
31	26
63	57
127	120

Az egyik legismertebb Hamming-kód a (7,4) paraméterű, ezt mutatja be a következő példa.

Példa: A (7,4) paraméterű Hamming-kód paritásmátrixa

$$H = \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

A generátormátrixa ebből könnyen kiszámítható a már szerepelt $A = -B^T$ összefüggés alapján

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

A (7,4)-es Hamming kódot egy páratlan paritásúra kiegészítő paritásbittel kapunk egy (8,4) paraméterű, továbbra is egy hibát javító kódot, amelyet a Teletexben használnak ([11]).

A közvetlen műholdas műsorszórás (DBS) digitalizált hangját is hibajavító kóddal védik. Így a D2-MAC/PACKET szabványa ([12]) szerint az egyik változatban a 14 bites hanghiba felső 11 bitjót egy (16,11) paraméterű kóddal kódolják, ami a (15,11) paraméterű Hamming kód kiegészítése egy páratlan paritásbittel. A másik változatban a 10 bites hangminta felső 6 bitjót kódolják egy (11,6)-es kóddal. Megjegyezzük még, hogy a csomagolt kódolt beszédmintákat egy olyan csomagfejjel látják el, melyet 2 hibát javító (71,57) ill. (94,80) paraméterű BCH kóddal védik, míg a legfontosabb adatokat, az úgynevezett szolgáltatásazonosítást egy három hibát javító (23,12) paraméterű Golay kóddal kódolják. Mind a Golay, mind a BCH kódok tárgyalása újabb algebrai apparátus bevezetését igényelné.

4. Véges test

Hatékony hibajavító kódok konstrukciójához szükséges, hogy a G halmaz struktúrált legyen, mely például úgy lehetséges, hogy műveleteket vezetünk be G-n.

Egy G halmazt testnek nevezünk, ha értelmezve van tetszőleges két eleme között két művelet, melyeket összeadásnak illetve szorzásnak nevezünk, "+ " illetve "·" szimbólumokkal jelöljük, és rendelkezik a következő tulajdonságokkal:

1. G az összeadásra nézve kommutatív csoport, azaz
1.1. Minden $\alpha, \beta \in G$ esetén $\alpha + \beta \in G$, tehát G az összeadásra nézve zárt.
1.2. Minden $\alpha, \beta, \gamma \in G$ esetén $\alpha + (\beta + \gamma) = (\alpha + \beta) + \gamma$ (asszociativitás)
1.3. Létezik egy 0-val jelölt eleme G-nek úgy, hogy minden $\alpha \in G$-re $0 + \alpha = \alpha + 0 = \alpha$. 0-t nullelemnek nevezzük.
1.4. Minden $\alpha \in G$-hez létezik $\beta \in G$ úgy, hogy $\alpha + \beta = 0$. β-t az α additív inverzének nevezzük és $-\alpha$-val jelöljük.
1.5. Minden $\alpha, \beta \in G$-re $\alpha + \beta = \beta + \alpha$ (kommutativitás).
2. G–{0} a szorzásra nézve kommutatív csoport, azaz
2.1. Minden $\alpha, \beta \in G-\{0\}$ esetén $\alpha . \beta \in G-\{0\}$
2.2. Minden $\alpha, \beta, \gamma \in G-\{0\}$ esetén $(\alpha . \beta) . \gamma = \alpha . (\beta . \gamma)$.
2.3. Létezik egy 1-gyel jelölt eleme G-{0}-nek úgy, hogy $1 . \alpha = \alpha . 1$. 1-t egységelemnek nevezzük.
2.4. Minden $\alpha \in G-\{0\}$ esetén létezik $\beta \in G-\{0\}$ úgy, hogy $\alpha . \beta = \beta . \alpha = 1$. β-t az α multiplikatív inverzének nevezzük, és α^{-1}-gyel jelöljük.
2.5. Minden $\alpha, \beta \in G-\{0\}$-ra $\alpha . \beta = \beta . \alpha$.
3. Minden $\alpha, \beta, \gamma \in G$-re $\alpha . 0 = 0 . \alpha$ és $\alpha . (\beta + \gamma) = (\alpha . \beta) + (\alpha . \gamma)$ (disztributivitás).

Egyszerű konvenciókkal egy G testben definiálható a kivonás és az osztás a következő módon:

$\alpha - \beta$ alatt az α-nak és a β additiv inverzének összegét értjük azaz $\alpha + (-\beta)$-t.

α / β alatt az α-nak és a β multiplikativ inverzének a szorzatát értjük, azaz $\alpha . \beta^{-1}$-t, amennyiben β nem 0.

Példák testre

1. Valós számok halmaza a valós összeadással és szorzással.
2. Racionális számok halmaza a valós összeadással és szorzással.
3. Komplex számok halmaza a komplex összeadással és szorzással.
4. {0,1} a bináris összeadással és szorzással.

Egy q véges elemszámú G testet véges testnek nevezünk és GF(q)- val jelöljük a *Galois field* rövidítéseként.

Sajnos egy GF(q) esetén q nem lehet bármilyen. Ez azért fontos, mert a kód-ABC a későbbiekben GF(q) lesz. Bizonyítás nélkül közöljük, hogy egy GF(q) esetén $q=p^m$ alakú, ahol p prímszám, tehát q vagy prímszám, vagy prímhatvány.

Lényeges különbség van a prím és a prímhatvány méretű véges testek aritmetikája között. A testaxiómák ellenőrzésével egyszerűen belátható, hogy a G={0,1,...p-1} halmaz egy p prímszám esetén véges test a modulo p aritmetikával, azaz

$$a+b = a+b \bmod p \qquad a.b = a.b \bmod p,$$

ahol + illetve . jelöli a valós összeadást illetve szorzást.

Sajnos $q=p^m$ esetben a modulo q aritmetika nem felel meg. Például $q=2^8$ esetén 2-nek és 128-nak a modulo 256-tal vett szorzata 0, ami sérti a 2.1. számú testaxiómát. Az érdeklődő olvasó a nem prim méretű test aritmetikájával a [3]-[10] könyvekben ismerkedhet meg.

Egy $\alpha \in GF(q)$ primitiv elemének nevezzük, ha a $1, \alpha, \alpha^2, \alpha^3, \ldots \alpha^{q-1}$ mind különbözik. Bizonyítás nélkül közöljük, hogy minden GF(q)- ban létezik primitív elem.

Példa:

GF(q), ha q=7.

elem ($\neq 0$)	hatványai
1	1

2	2,4,1
3	3,2,6,4,5,1 (primitiv elem)
4	4,2,1
5	5,4,6,2,3,1 (primitiv elem)
6	6,1

5. Lineáris kódok, nembináris Hamming kód

Ebben a pontban kódok egy fontos csoportjával ismerkedünk meg, melyek a 3. pontban megismert bináris lineáris kódok kiterjesztései nembináris esetre.

A továbbiakban a kódjainkban szereplő kódszavakat alkotó szimbólumokat vegyük GF(q)-ból, a lehetséges szimbólumok tehát a 0,1,2,...,q-1 számoknak feleltethetők meg.

Egy C kódot lineárisnak nevezünk, ha a C halmaz lineáris tér GF(q) fölött, azaz ha minden $c,c' \in C$-re

$$c+c' \in C$$

illetve $\alpha \in GF(q)$ esetén

$$\alpha c \in C.$$

A bináris esethez hasonló módon belátható, hogy tetszőleges C lineáris kódhoz létezik egy k lineárisan független sorból és n oszlopból álló G mátrix melyre

$$c = uG \qquad (5.1)$$

ahol a k hosszú u üzenethez a c kódszó tartozik, és G mátrixot a C kód generátormátrixának nevezzük.

A bináris esethez hasonlóan a C lineáris kódhoz létezik egy n-k sorból és n oszlopból álló H mátrix, melyre

$$Hc^T = 0$$

akkor és csak akkor, ha $c \in C$, és H-t a C kód paritás ellenőrző mátrixának nevezzük.

Példaként bemutatjuk a nembináris Hamming kódot $n \leq q+1$ és $k=n-2$ esetén. A bináris Hamming kódhoz hasonlóan egy 1 hibát javító kód konstrukciója a cél, melyet a paritásmátrixával definiálunk a következő módon:

$$H = \begin{pmatrix} -1 & -1 & -1 & \dots & -1 & 1 & 0 \\ -1 & -\alpha & -\alpha^2 & \dots & -\alpha^{n-3} & 0 & 1 \end{pmatrix},$$

ahol α a GF(q) egy primitív eleme. Ez egy (n,n-2) paraméterű kód paritásmátrixa. Meg kell mutatnunk, hogy ez minden egyedi hibát ki tud javítani. Ennél többet bizonyítunk, amikor megadunk egy egyszerű dekódolási eljárást. Legyen az e hibavektor olyan, hogy az l-edik pozíción e, egyébként 0, akkor a szindróma $s = e \cdot a_l$, ahol a_l a paritásmátrix l-edik oszlopa. Ha s második koordinátája 0, akkor a hiba az utolsó előtti pozíción történt, ami a paritásszegmensben van, javítása felesleges. Ugyanígy nem szükséges javítani, ha s első koordinátája 0, mert ekkor a hiba az utolsó helyen történt. Minden egyéb esetben a hiba értékét, e-1 az s első koordinátájának -1-szerese adja, míg a hiba helyét, l-t az $a_l = s/e$-ből visszakereshetjük.

A 3.1. Tétel alkalmazásával nyerjük a kód generátormátrixát:

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & \dots & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & \dots & 0 & 1 & \alpha \\ 0 & 0 & 1 & 0 & \dots & 0 & 1 & \alpha^2 \\ \dots & & & & & & & \\ 0 & 0 & 0 & 0 & \dots & 1 & 1 & \alpha^{n-3} \end{pmatrix}$$

6. Véges test feletti polinomok

Az $a(x) = a_0 + a_1 x + \dots + a_m x^m$ GF(q) feletti m-ed fokú polinom, ha

$$a_i \in GF(q), i=0\dots m, a_m \neq 0$$

$$x \in GF(q)$$

A polinom fokszámát deg a(x) = m alakban fogjuk jelölni.

$a(x) = b(x)$ ha $a_i = b_i$ minden i-re.

$b \in GF(q)$ gyöke az a(x) polinomnak, ha a(b) = 0.

Műveletek polinomok között:

1. *Polinomok összeadása*: $c(x) = a(x) + b(x)$; tagonként történik GF(q) feletti műveletekkel: $c_i = a_i + b_i$

2. *Polinomok szorzása*: $c(x) = a(x)b(x)$; minden tagot minden taggal szorzunk, majd az azonos fokú tagokat csoportosítjuk (az összeadások és szorzások GF(q) felettiek):

$$c_i = \sum_{j=0}^{\min[i, d\in g\, a(x)]} a_j . b_{i-j},$$

Példa: Ha GF(2) felett $a(x) = 1+x$ és $b(x) = 1+x+x^3$, akkor $a(x) + b(x) = x^3$ és $a(x)\,b(x) = 1+x^2+x^3+x^4$.

A valós számtest feletti polinomokhoz hasonlóan igaz az Euklideszi osztás GF(q) feletti polinomokra, következésképp adott a(x) és $d(x) \neq 0$ esetén egyértelműen létezik olyan q(x), r(x), hogy

$$a(x) = q(x)d(x) + r(x) \text{ és } \deg r(x) < \deg d(x).$$

Jelölés: r(x) = a(x) mod d(x) és r(x)-t az a(x) d(x)-re vonatkozó maradékának nevezzük.

Azt mondjuk, hogy d(x) osztja a(x)-t, ha a(x) mod d(x) = 0. Ezt a továbbiakban d(x) | a(x) formában fogjuk jelölni.

6.1. Tétel: Ha α az a(x) polinom gyöke, akkor az előáll

$$a(x) = b(x)(x-\alpha)$$

alakban.

Bizonyítás Alkalmazzuk az Euklideszi osztást $d(x) = x - \alpha$ esetén, akkor

$$a(x) = b(x)(x-\alpha) + \beta.$$

Mivel α gyök, ezért

$$0 = a(\alpha) = b(\alpha)(\alpha - \alpha) + \beta = \beta.$$

6.2. Tétel: Egy k-adfokú polinomnak legfeljebb k gyöke lehet.

Bizonyítás A 6.1. Tétel miatt a b(x) polinom fokszáma eggyel kisebb, mint az a(x) polinom fokszáma, tehát ezt a faktorizációt legfeljebb k-szor lehet megismételni.

7. Reed-Solomon kód

Ebben a pontban a lineáris kódok egyik leggyakrabban használt osztályával a Reed-Solomon kódokkal ismerkedünk meg. Legyen $u=(u_0, u_1, ..., u_{k-1})$ a k hosszúságú üzenetszegmens, és

$$u(x)=u_0+u_1x+u_2x^2+...+u_{k-1}x^{k-1},$$

akkor az n hosszú c Reed-Solomon kódszót a következő módon állítjuk elő ($n \leq q-1$):

$$c_0=u(1)$$
$$c_1=u(\alpha)$$
$$c_2=u(\alpha^2)$$
$$\vdots$$
$$c_{n-1}=u(\alpha^{n-1}),$$

ahol α a GF(q) primitiv eleme.

Egyszerűen belátható, hogy a Reed-Solomon kód lineáris, és a generátormátrixa

$$G=\begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \alpha & \alpha^2 & \dots & \alpha^{n-1} \\ 1 & \alpha^2 & \alpha^4 & \dots & \alpha^{2(n-1)} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & \alpha^{k-1} & \alpha^{2(k-1)} & \dots & \alpha^{(k-1)(n-1)} \end{pmatrix}$$

7.1. Tétel: Az (n,k) paraméterű RS kód kódtávolsága

$$d_{min}=n-k+1,$$

vagyis a RS kód maximális távolságú.

Bizonyítás: $w(c)=|\{c \text{ nem 0 koordinátái}\}|=$
$$=n-|\{c \text{ 0 koordinátái}\}| \geq$$
$$\geq n-|\{u(x) \text{ gyökei}\}| \geq$$
$$\geq n-(k-1),$$

tehát

$$w_{min} \geq n-k+1.$$

Ugyanakkor a Singleton korlát és a 3.2. Tétel miatt

$$n-k+1 \geq d_{min}=w_{min},$$

következésképp az állítást bebizonyítottuk.

Az (n,k) paraméterű Reed-Solomon kód tehát n-k hibát tud jelezni, int ((n-k)/2) egyszerű hibát javítani és n-k törléses hibát javítani. Ez utóbbi azt is jelenti, hogy az u ismeretlenre vonatkozó

$$uG=c$$

n darab egyenletből bármelyik n-k egyenlet elhagyásával egy egyértelműen megoldható egyenletrendszer marad, tehát a G mátrix minden kxk-s négyzetes részmátrixa invertálható.

Példaként a digitális hangrögzítésben (CD és DAT) alkalmazott Reed-Solomon kódot említenénk ([13]-[17]). A kódolási eljárás lényegét közelítőleg a következő módon lehet összefoglalni: a 44.1 kHz-cel mintavételezett és 16 bitbe kvantált mintákat két byte-ban ábrázoljuk, és egy mátrixba írjuk be oszlopfolytonosan, nevezetesen egy 24x24-es mátrix oszlopai egymásután következő 6 mintavételi időpontban vett két minta (bal és jobb csatorna) 4 byte-ját tartalmazza. Ha $x_{i,1}, x_{i,2}$ jelöli a jobb csatorna mintáját az i-edik időpillanatban, és $y_{i,1}, y_{i,2}$ a bal csatornáét, akkor az 1. ábra mutatja a minták beírását a táblázatba. A kapott 24x24-es mátrix minden oszlopát kódoljuk egy

RÖGZÍTÉS IRÁNYA

MINTAVÉTEL IRÁNYA								
$x_{1,1}$	$x_{7,1}$	$x_{13,1}$	.	$x_{139,1}$	$r_{1,1}$	$r_{1,2}$	$r_{1,3}$	$r_{1,4}$
$x_{1,2}$	$x_{7,2}$	$x_{13,2}$	.	$x_{139,2}$	$r_{2,1}$	$r_{2,2}$	$r_{2,3}$	$r_{2,4}$
$y_{1,1}$	$y_{7,1}$	$y_{13,1}$	.	$y_{139,1}$	$r_{3,1}$	$r_{3,2}$	$r_{3,3}$	$r_{3,4}$
$y_{1,2}$	$y_{7,2}$	$y_{13,2}$	.	$y_{139,2}$	$r_{4,1}$	$r_{4,2}$	$r_{4,3}$	$r_{4,4}$
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
.	.	.	.	.	.	.	.	.
$x_{6,1}$	$x_{12,1}$	$x_{18,1}$	.	$x_{144,1}$	$r_{21,1}$	$r_{21,2}$	$r_{21,3}$	$r_{21,4}$
$x_{6,2}$	$x_{12,2}$	$x_{18,2}$	.	$x_{144,2}$	$r_{22,1}$	$r_{22,2}$	$r_{22,3}$	$r_{22,4}$
$y_{6,1}$	$y_{12,1}$	$y_{18,1}$	.	$y_{144,1}$	$r_{23,1}$	$r_{23,2}$	$r_{23,3}$	$r_{23,4}$
$y_{6,2}$	$y_{12,2}$	$y_{18,2}$	.	$y_{144,2}$	$r_{24,1}$	$r_{24,2}$	$r_{24,3}$	$r_{24,4}$
$q_{1,1}$	$q_{1,2}$	$q_{1,3}$	.	$q_{1,24}$	$q_{1,25}$	$q_{1,25}$	$q_{1,27}$	$q_{1,28}$
$q_{2,1}$	$q_{2,2}$	$q_{2,3}$	.	$q_{2,24}$	$q_{2,25}$	$q_{2,26}$	$q_{2,27}$	$q_{2,28}$
$q_{3,1}$	$q_{3,2}$	$q_{3,3}$	.	$q_{3,24}$	$q_{3,25}$	$q_{3,26}$	$q_{3,27}$	$q_{3,28}$
$q_{4,1}$	$q_{4,2}$	$q_{4,3}$	.	$q_{4,24}$	$q_{4,25}$	$q_{4,26}$	$q_{4,27}$	$q_{4,28}$

1. ábra

(28,24) paraméterű $GF(2^8)$ feletti szisztematikus Reed-Solomon kóddal. A j-dik oszlop paritásbyte-jait jelöltük $q_{1,j}, q_{2,j}, q_{3,j}, q_{4,j}$-vel. Ennek a kódnak a kódtávolsága 5, tehát 4 hibát tud jelezni, 2 egyszerű hibát tud javítani és 4 törléses hibát tud javítani. A digitális lemezen előforduló hibák jól modellezhetők egy kétállapotú csatornával. Az egyik állapotot nevezzük Jó állapotnak, melyben átlagosan 10000-20000 bit ideig tartózkodik, és ekkor a hibák előfordulása független egymástól és valószínűsége kb. 10^{-4}. A másik állapotot nevezzük ROSSZ állapotnak, amiben 30-40 bit ideig tartózkodik, és ekkor gyakorlatilag használhatatlan a vétel. Ekkor azt mondjuk, hogy hibázás burst-ös. Az ilyen csatornák kódolására találták ki az interleaving technikát, amikor az előbbi mátrixot sorfolytonosan olvassák ki, de előtte mindegyik sort is kódolják ugyanazzal a (28,24) paraméterű Reed-Solomon kóddal. A j-edik sor paritásbyte- jait jelöli $r_{j,1}, r_{j,2}, r_{j,3}, r_{j,4}$.

A Sony és a Philips megegyezett a fentihez hasonló (kicsit bonyolultabb) kódolásban azért, hogy a tömeges digitális hanglemezgyártás elindulhasson. A verseny nyitott viszont a lejátszó készülékben, vagyis a dekódolás terén. A különböző dekódolások igazából a következő egyszerű eljárás finomításai: számítsuk ki soronként a szindrómát! Ha a szindróma 0, akkor azzal a sorral készen vagyunk. Ha egy hiba volt, akkor azt kijavítjuk, és az oszloponkénti javításhoz ezeket a hibahelyeket megjegyezzük, azaz mesterségesen törléses hibákat generálunk. Minden egyéb esetben az egész sort törléses hibaként regisztráljuk. Ezek után oszloponként javítunk, ha ott legfeljebb két

törlóses hiba volt (emlékeztetünk, hogy 4 törlóses hibát képes a rendszer javítani). Ha a hibák száma nagyobb, mint 2, akkor a környező hibátlan mintákból interpolálunk. Látható, hogy a hibajavítás nem használja ki a Reed-Solomon kód hibajavítási lehetőségeit, aminek elsősorban technológiai okai vannak, mivel a dekódolás bonyolultsága a javítandó hibák számának négyzetével arányos, ós itt igen gyorsan kell dekódolni (a forrás sebessége 2x44x100x16 = 1.4112 Mbit/sec)

Irodalom

[1] Győrfi László, Simonyi Gábor, Vajda István, Zelsel Tamás "A hibajavító kódolás elemei" Intézeti tanulmány, BME HEI 1986.

[2] Fritz József, Csiszár Imre "Információelmélet" Tankönyvkiadó 1983.

[3] Vajda István "Hibajavító kódolás és műszaki alkalmazásai" BME Mérnöki Továbbképző Intézet, 1982.

[4] R.B.Ash "Information Theory" Interscience Publishers, 1965.

[5] R.G.Gallager "Information Theory and Reliable Communication" Wiley 1968.

[6] R.J. McElice "The theory of Information and Coding" Addison-Wesley Publishing Company, 1977.

[7] W.W.Peterson "Error Correcting Codes" MIT Press Cambridge, Mass., and Wiley, 1961

[8] E.R.Berlekamp "Algebraic Coding Theory" McGraw Hill, 1968

[9] F.J.MacWilliams, N.J.A.Sloane "The Theory of Error-Correcting Codes" North-Holland, 1977

[10] R.E.Blahut "Theory and Practice of Error Control Codes" Addison-Wesley Publishing Company, 1983

[11] Ferenczy Pál "Video- és hangszerek" Műszaki Könyvkiadó, 1986.

[12] Specification of the D2-MAC/PACKET System EBU/SPB 352/B/1985 febr.

[13] All about the Compact Disc System Compact Disc Digital Audio - Sony 1981-es kiadvány

[14] K.Odaka, T. Furuya, A. Taki "LSIos for Digital Signal Processing to be used in CD Players" No 1860 AES 71. Convention, 1982 March, Motreux

[15] T.Doi "Error Correction for Digital Audio Recordings" No 1991 AES 73. Convention, 1983 March, Eindhoven

[16] Y. Ishida, M. Ishkia, K. Nakagawa, Y. Osuga, J. Yanabe "On the development of a car use rotary-head digital audio tape recorder" No 2318 AES 80. Convention, 1986 March, Montreux

[17] E.R. Berlekamp "The Technology of Error-Correcting Codes" Proc. IEEE, 68, May 1980.

[18] E.R Berlekamp, R.E. Pelle, S.P. Pope "The Application of Erreo Control to Communications" IEEE Communications Magazine, 25, April 1987.