Gonda János

KÓDOLÁS ÉS REJTJELEZÉS

Budapest, 2010

Lektorálta **xxxx**

Utolsó módosítás: 2009. december 3.

1. Előszó

Ez a jegyzet az ELTE-n tartott Kódelmélet és kriptográfia című tárgy anyagát tartalmazza. Tekintettel arra, hogy a tárgy heti két (tan)órában, egy félév keretében kerül előadásra, az anyag ehhez a szűkre szabott időkerethez igazodik, így is az ennyi idő alatt elmondható ismeretek mennyiségének felső határát súrolva, esetleg ezt a korlátot kissé át is lépve. Éppen erre való tekintettel szükséges megjegyezni, hogy bizonyos részek a tényleges előadás és számonkérés során többé vagy kevésbé tömöríthetőek, belőlük egyes részek kihagyhatóak vagy csupán érintőlegesen kerülhetnek szóba. Ez függhet az előadó ízlésétől, a tárgyat hallgatók összetételétől és „előéletétől", a tárgyban tanultakra esetleg támaszkodó további tárgyaktól, az adott félév tényleges hosszától, és még más körülményektől is.

Miről szól a tárgy és ez a jegyzet? A címük szerint a kódolásról és a rejtjelezésről. A címek ilyen formán egy teljes, lezárt témát ígérnek a hallgatónak illetve olvasónak. A valóság ezzel szemben lényegesen szegényesebb. A már említett időkorlátot figyelembe véve nem vállalkozhattunk másra, és a valóság is az, hogy csupán az említett témakör egy kis, bár viszonylag jól körülhatárolható részével foglalkozzunk. Amiről szó lesz, az lényegében véve a véletlen hibát javító blokk-kódok alapjainak, korlátainak, néhány ilyen kódnak az ismertetése, illetve az információ algoritmikus védelmének, a titkosításnak, a személyazonosításnak, a hitelesítésnek a kódoláshoz hasonlóan igen vázlatos ismertetése. Nem foglalkozunk gyakorlati kérdésekkel, csupán érintjük a hibacsomó-javító kódokat, szóba sem kerülnek az egyébként fontos konvolúciós kódok, és csupán burkoltan, más nézőpontból, a kapcsolatot még csak meg sem említve tárgyalunk bizonyos algebrai geometriai kódokat. Igen szűkre szabottan, majdhogynem ismeretterjesztő szinten beszélünk a hibakorlátozás valószínűségi kérdéseiről, ami annyiban érthető, hogy a tárgy a kódolás algebrai vonatkozásaival foglalkozik. Nagyon kevés szó van nemlineáris kódokról, bár helyenként a szokásosnál általánosabban tárgyalunk egyes kérdéseket, kiterjesztve a fogalmakat a nemlineáris kódokra is. Végül a lineáris kódok jelenlegi ismeretanyaga is lényegesen bővebb annál, mint ami egy ilyen csaknem bevezető jellegű tárgy anyagába belefér. A kriptográfiával kapcsolatban is csupán a széles körben ismert és használt módszerekkel, technikákkal foglalkozunk, valamint egy keveset a kriptográfia matematikai vonatkozásaival. Természetesen az összeválogatott anyag sok egyéb mellett a válogatást végző személyiségétől, ízlésétől is függ, vagyis nem nélkülöz bizonyos szubjektivitást sem.

Mivel a kódolásnál algebrai kódokról van szó, az anyag megértéséhez szükség van algebrai ismeretekre. Ez részben lineáris algebrát, részben általános algebrai ismereteket (csoportokkal, gyűrűkkel, testekkel, polinomokkal kapcsolatos fogalmakat) jelent, jelenti azonban azt is, hogy lényegesen támaszkodik az általában sokkal kisebb részben oktatott véges testek bizonyos fokú ismeretére. Jóllehet a gyakorlatban alkalmazott kódok túlnyomó többsége bináris, ez nem jelenti azt, hogy csak ilyen kódok vannak és csak ilyeneket használnak a gyakorlatban, sőt, van olyan igen fontos kódosztály, amelyben csak triviális, és így lényegében véve használhatatlan formában léteznek a bináris kódok. Éppen ezért mindenütt általánosan, tetszőleges szimbólumhalmaz, illetve lineáris kód esetén tetszőleges véges test fölött tárgyaljuk a kódokat, és ezen belül utalunk a bináris kódok esetleges speciális tulajdonságaira.

Ami a titkosítást illeti, itt többek között bizonyos számelméleti alapokra kell támaszkodnunk, mint például a kongruenciák ismeretére.

A titkosság a társadalmak, egymástól elkülönült közösségek kialakulásához kapcsolódik. Egyrészt a rendelkezésre álló erőforrások különbözősége, másrészt az emberi léthez kapcsolódó bizonyos negatív tulajdonságok arra vezettek, hogy az egyes csoportok egymás rovására jutottak meghatározott javakhoz. A javak megszerzésében azok számíthattak nagyobb sikerre, akik képesek voltak meglepni a konkurens társaságot. A meglepetés alapja viszont az, hogy az egyik társaság tud valami olyat, ame-

lyet a másik csoport nem ismer, és amit az egymással való vetélkedés során eredményesen fel lehet használni.

A titkosítás története több könyvben is megtalálható. Közülük minden bizonnyal a leghíresebb *David Kahn* könyve, ugyanis szinte nincs olyan, a kriptológiával foglalkozó könyv, amely ne hivatkozna erre a több mint ezer oldalas könyvre. A történeti szemléletű magyar nyelvű könyvek közül említésre méltó *Simon Singh* műve, a *Kódkönyv*, amely már a legújabb rejtjelező eljárásokról is számot tud adni, hiszen ebben az évezredben jelent meg. Igen tanulságos elolvasni *Révay Zoltán Titkosírások* című könyvét. Ez a könyv egyrészt azért érdemel figyelmet, mert igen sok jeles magyar személyiségről derül ki, hogy intenzíven alkalmazta a titkosítás tudományát, és számos érdekes megoldást találtak ki a rejtjelezéshez, másrészt viszont a megjelenésének dátuma szempontjából is érdekes ez a könyv (bár ez elmondható *Kahn* könyvéről is). *Révay Zoltán* idézi *Aineiasz Taktikosz Taktika* című művének egyik könyvét, a *Poliorkétika*-t, közelebbről ennek XXXI. fejezetét, amelyben *Aineiasz* a titkos levelekről ír. Ez a fejezet azzal kezdődik, hogy „A titkos leveleknek mindenféle küldési módjuk van, de a küldőnek és a címzettnek egymás között előzőleg meg kell állapodnia.”. Ez az idézet azért érdekes, mert a könyv első megjelenése előtt két évvel jelent meg *Diffie* és *Hellman* cikke, amely alapjaiban rázkódtatta meg a rejtjelezés világát, és amely alaposan rácáfolt *Aineiasz*-ra, és közvetve *Révay*ra is, aki a fenti gondolatot lényegében véve a titkosítás alapjának tekintette. (Ez nem csökkenti a *Révay*-könyv értékét, csupán arra mutat rá, hogy a világ forgandó, és igen rövid idő alatt fenekestül tud egy-egy tudományág megváltozni. Hasonló változás történt például 1900-ban vagy 1905-ben a fizikában.)

Minden titkosító eljárás esetén lényeges, hogy az alkalmazott algoritmusról feltételezzük, azt mindenki ismeri, és a titkosságot az úgynevezett **kulcs** biztosítja. A kulcs az algoritmus egy olyan paramétere, amelytől függően ugyanaz az eljárás ugyanazon titkosítandó üzenetből a kulcs függvényében más és más rejtjelezett szöveget állít elő. A klasszikus rejtjelező eljárásoknál a visszafejtéshez szükséges kulcs vagy megegyezett a titkosításhoz használt kulccsal, vagy abból könnyen ki lehetett számolni, így szükséges volt a rejtjelezéshez használt kulcsot is titokban tartani, továbbá az üzenetváltásban résztvevő két fél között biztonságosan kicserélni. Más a helyzet akkor, ha az oda-, illetve visszatranszformáláshoz használt kulcsok olyanok, hogy az egyik ismeretében a másik csak olyan nagy költséggel határozható meg, hogy az meghaladja a megszerzett információ értékét. Ebben az esetben a titkosító kulcs akár nyilvános is lehet, mégsem képes senki illetéktelen elolvasni a rejtjelezett szöveget, mivel nem rendelkezik a visszafejtéshez szükséges kulccsal. Ez az a gondolat, amely *Diffie* és *Hellman* cikkében jelent meg, és amely alapján kialakult a nyilvános kulcsú rejtjelezés. E nélkül a mai világ egészen más lenne. A régi időkben lényegében véve csak az államnak voltak féltve őrzött titkai (persze a szépasszonyok sem akartak mindent az uruk orrára kötni, de ezek kevésbé lényeges titkok...), így elegendő volt csupán néhány tucat kulcsot előállítani és kicserélni. (Ez utóbbi egy kényes pontja a rejtjelezésnek, hiszen a kulcsot biztonságosan és titkosan kell eljuttatni a másik félnek, amikor persze felmerül a kérdés, hogy miért nem magát az üzenetet cserélik ez alkalommal ki. Erre azonban könnyű a válasz: a kulcsot bármely időben cserélhetjük, és a cserére ritkán van szükség, továbbá a kulcs általában rövid az üzenethez képest.) A mai világban viszont a titkosítandó információk túlnyomó többsége gazdasági jellegű, és magánszemélyekhez, vállalatokhoz kapcsolódik. Potenciálisan minden ember és minden vállalkozás rendelkezik titkolandó adattal, amelyet a legkülönbözőbb intézményekkel kell kicserélnie. A titkos kulcspár alkalmazása esetén különböző partnerhez más és más kulcsra lenne szükség, ami azt jelentené, hogy hihetetlenül sok kulcsot kellene igen gyakran rendkívül sok pár között kicserélni, és a titkos kulcsokat megfelelően adminisztrálva biztonságosan tárolni, ami megoldhatatlan feladat elé állítaná az egyszerű honpolgárokat. Még azt is figyelembe kell venni, hogy a kulcsot viszonylag gyakran kell cserélni, még az előtt, hogy illetéktelen személy megfejtené, és így a továbbiakban a titkos információnkat olvasni tudná.

Az előbbi gondolatok alapján joggal merül fel a kérdés, hogy kell-e egyáltalán foglalkozni a klasszikus, szimmetrikus rejtjelező rendszerekkel. A válasz meglepő módon igen. A helyzet ugyanis az, hogy a szimmetrikus rendszerek lényegesen gyorsabbak, mint a nyilvános kulcsú eljárások, ezért a legtöbb esetben egy-egy konkrét üzenetváltás előtt a nyilvános kulcsú rejtjel segítségével a két partner kicserél egy kulcsot, és a továbbiakban az aktuális információt az így megismert kulcs segítségével, egy klasszikus módszerrel küldik egy nyilvános csatornán keresztül.

A téma iránt mélyebben érdeklődő olvasó az irodalomjegyzékben említett könyvekből szerezhet további ismereteket, éppen ezért nem csak olyan könyveket soroltunk ott fel, amelyek szorosan kapcsolódnak az általunk kifejtett részletekhez.

Végül néhány jelölésről szólunk. Ebben a jegyzetben $\mathbb{N}^+$ a pozitív egész számokat jelöli, és $\mathbb{N}$ jelöli a nem negatív egész számokat. Egy polinomot például f-fel, és nem $f(x)$-szel jelölünk, megfelelően annak, hogy a polinom egy formális kifejezés, amelyet az együtthatói határoznak meg. Az f polinomhoz tartozó polinomfüggvény jele $\hat{f}$. A mátrixokat és vektorokat félkövér betű jelöli, a halmazokat dőlt betű, és egy struktúrát a hozzá tartozó halmaztól a betű típusa különbözteti meg, például az A halmazra épített struktúra jele $\mathcal{A}$. Végül a q-elemű test jele ebben a jegyzetben $\mathbb{F}_q$.

Tartalomjegyzék

2. A kódtér geometriája

2.1. Definíció

Legyen S nem üres véges halmaz, és $n \in \mathbb{N}^+$. Ekkor az S^n $\mathbf{u}$ és $\mathbf{v}$ elemének **Hamming-távolság**a $d(\mathbf{u},\mathbf{v}) = |\{n > i \in \mathbb{N} | u \neq v\}|$, és ha C az S^n legalább két elemből álló részhalmaza, akkor $d(C) = min\{d(\mathbf{u},\mathbf{v}) | \mathbf{u} \in C \wedge \mathbf{u} \neq \mathbf{v} \in C\}$ **a** C (**minimális**) **távolsága**.

Amennyiben $\mathcal{S}$ additív Abel-csoport a 0 neutrális elemmel, akkor $w(\mathbf{u}) = |\{n > i \in \mathbb{N} | u \neq 0\}|$ az **u Hamming-súlya**, és ha még a $C \subseteq S^n$ halmazra $C\backslash\{\mathbf{0}\} \neq \emptyset$, úgy $w(C) = min\{w(\mathbf{u}) | \mathbf{0} \neq \mathbf{u} \in C\}$ **a** C (**minimális**) **súlya**, ahol $\mathbf{0} = \underbrace{0 \cdots 0}_{n}$.

Δ

A definíció alapján $|C| \leq 1$ esetén C távolsága, $C \subseteq \{\mathbf{0}\}$-nál C súlya definiálatlan.

2.2. Tétel

Legyen $S \neq \emptyset$ véges halmaz, $n \in \mathbb{N}^+$, $C \subseteq S^n$. Ekkor

- a Hamming-távolság metrika az S^n halmazon;
- ha $|C| \geq 2$, akkor $d(C) \in \mathbb{N}$, és van olyan $(\mathbf{u},\mathbf{v}) \in C \times C\backslash\{\mathbf{u}\}$, hogy $d(\mathbf{u},\mathbf{v}) = d(C)$;

ha $\mathcal{S}$ Abel-csoport a 0 neutrális elemmel, $\mathbf{0}$ a csupa 0-ból álló vektor, és tetszőleges $\mathbf{u} \in S^n$, $\mathbf{v} \in S^n$ elemekkel $\mathbf{u} - \mathbf{v} = u_1 - v_1 \cdots u_n - v_n$, akkor

- $d(\mathbf{u},\mathbf{v}) = w(\mathbf{u} - \mathbf{v})$ és $w(\mathbf{u}) = d(\mathbf{u},\mathbf{0})$;
- ha a legalább kételemű $C \subseteq S^n$ olyan, hogy $C - C \subseteq C$ is teljesül, akkor $d(C) = w(C)$;
- ha $C\backslash\{\mathbf{0}\} \neq \emptyset$, akkor $w(C) \in \mathbb{N}$, és létezik C-nek olyan $\mathbf{u}$ eleme, amelyre $w(\mathbf{u}) = w(C)$.

Δ

Bizonyítás:

1. $d(\mathbf{u},\mathbf{v})$ minden S^n-beli rendezett párra értelmezett, értéke mint egy véges halmaz számossága nemnegatív egész, azaz egyben nemnegatív valós szám, és egy adott $\mathbf{u}$, $\mathbf{v}$ párhoz pontosan egy ilyen számérték tartozik, ezért d egy $S^n - S^n \to \mathbb{R}_0^+$ leképezés, továbbá $d(\mathbf{u},\mathbf{v})$ pontosan akkor 0, ha minden fellépő i indexre u_i és v_i azonos, vagyis ha $\mathbf{u} = \mathbf{v}$. A szimmetria nyilvánvaló, hiszen a nem-egyenlőség szimmetrikus tulajdonság, ezért még a háromszög-egyenlőtlenséget kell megvizsgálni. Ha $\mathbf{z}$ egy harmadik elem S^n-ből, akkor $u_i = z_i$ és $z_i = v_i$ esetén $u_i = v_i$, ami megfordítva azt jelenti, hogy ha egy adott i-re $u_i \neq v_i$, akkor vagy u_i nem egyezik z_i-vel, vagy z_i és v_i különbözik, így egy olyan i index, amely szerepel $d(\mathbf{u},\mathbf{v})$ meghatározásában, benne lesz a $d(\mathbf{u},\mathbf{z})$-t és $d(\mathbf{z},\mathbf{v})$-t meghatározó indexhalmaz legalább egyikében, tehát $d(\mathbf{u},\mathbf{v}) \leq d(\mathbf{u},\mathbf{z}) + d(\mathbf{u},\mathbf{v})$.

2. $d(C)$ meghatározásában nem egyenlő $\mathbf{u}$ és $\mathbf{v}$ elemek szerepelnek, ezért a definícióban szereplő halmaz minden eleme pozitív egész szám, és a halmaz nem üres, ezért ez a halmaz a természetes számok halmazának nem üres részhalmaza, így van benne egy és csak egy legkisebb pozitív egész, de akkor ez valamely $\mathbf{u}$, $\mathbf{v}$ pár távolsága.

3. Abel-csoportban $u_i = v_i$ és $u_i - v_i = 0$ egyszerre igaz, így az $u_i \neq v_i$-t és $u_i - v_i \neq 0$-t teljesítő i indexek azonosak, $d(\mathbf{u},\mathbf{v}) = w(\mathbf{u} - \mathbf{v})$, amiből $\mathbf{v} = \mathbf{0}$ helyettesítéssel $w(\mathbf{u}) = d(\mathbf{u},\mathbf{0})$, hiszen $\mathbf{0}$-ban minden i-re 0 áll, és tetszőleges u_i-re $u_i - 0 = u_i$.

4. Most legyen $C \subseteq S^n$ legalább kételemű, és $C - C \subseteq C$. Míg $\mathbf{u}$ és $\mathbf{v}$ végigfut minden C-beli különböző elempáron, azalatt a különbségük egy-egy C-beli, $\mathbf{0}$-tól különböző elem lesz, ezért az adott $d(\mathbf{u},\mathbf{v})$ érték szerepel $w(C)$ meghatározásában is. De fordítva is igaz a dolog: ha $\mathbf{u}$ a C egy nem nulla eleme, akkor $d(\mathbf{u},\mathbf{0})$ benne lesz a $d(C)$-t definiáló halmazban, ezért a két halmaz, de akkor a minimumuk is azonos.

5. Ennek az állításnak a bizonyítása lényegében véve azonos a $d(C)$-re vonatkozó kijelentés bizonyításához.

□

A súly definíciója alapján $w(-\mathbf{u}) = w(\mathbf{u})$, és $d(\mathbf{u},\mathbf{v}) \geq |d(\mathbf{u},\mathbf{w}) - d(\mathbf{w},\mathbf{u})|$ a háromszög-egyenlőtlenségből. Szintén a háromszög-egyenlőtlenség alapján

$$w(\mathbf{u}+\mathbf{v}) = w\big(\mathbf{u}-(-\mathbf{v})\big) = d(\mathbf{u},-\mathbf{v}) \leq d(\mathbf{u},\mathbf{0}) + d(\mathbf{0},-\mathbf{v})$$
$$= w(\mathbf{u}) + w(-\mathbf{v}) = w(\mathbf{u}) + w(\mathbf{v})$$

illetve

$$w(\mathbf{u}+\mathbf{v}) = w\big(\mathbf{u}-(-\mathbf{v})\big) = d(\mathbf{u},-\mathbf{v}) \geq |d(\mathbf{u},\mathbf{0}) - d(\mathbf{0},-\mathbf{v})|$$
$$= |w(\mathbf{u}) - w(-\mathbf{v})| = |w(\mathbf{u}) - w(\mathbf{v})|,$$

vagyis a súlyokra is teljesül a háromszög-egyenlőtlenség.

A továbbiakban $d(C)$ helyett d-t, $w(C)$ helyett w-t írunk.

2.3. Megjegyzés

Legyen S nem üres, véges halmaz és $n \in \mathbb{N}^+$. S-t **szimbólumhalmaz**nak, $C \subseteq S^n$-t **kód**nak nevezzük, $\mathbf{v} \in S^n$ egy S **fölötti** n **hosszúságú szó**, és $\mathbf{u} \in C$ egy S **fölötti kódszó**. Ha $q = |S|$, $M = |C|$ és $d(C) = d$, akkor C egy $(n, M, d)_q$**-paraméterű kód**. A jelölésben q-t és d-t, egymástól függetlenül, el lehet hagyni.

Δ

2.4. Definíció

Legyen $S \neq \emptyset$ véges halmaz és $n \in \mathbb{N}^+$. Ekkor $G_t(\mathbf{u}) = \{\mathbf{v} \in S^n | d(\mathbf{u},\mathbf{v}) \leq t\}$, ahol $t \in \mathbb{R}_0^+$ és $\mathbf{u} \in S^n$, az S^n**-beli u középpontú,** t **sugarú gömb**.

Δ

2.5. Tétel

Legyen $S \neq \emptyset$ véges halmaz, $n \in \mathbb{N}^+$**,** $C \subseteq S^n$, $|C| \geq 2$ és $t \in \mathbb{N}$. Ha $t < \frac{d}{2}$, akkor a C-beli középpontú, t-sugarú gömbök páronként diszjunktak, de $t \geq \frac{d}{2}$ esetén van olyan $\mathbf{u} \in C$, $\mathbf{u} \neq \mathbf{v} \in C$, hogy $G_t(\mathbf{u}) \cap G_t(\mathbf{v}) \neq \emptyset$.

Δ

Bizonyítás:

Ha $\mathbf{u} \in C$, $\mathbf{u} \neq \mathbf{v} \in C$, $t \in \mathbb{N}$, és az S^n-beli $\mathbf{x}$-re $\mathbf{x} \in G_t(\mathbf{u}) \cap G_t(\mathbf{v})$, akkor $d(\mathbf{u},\mathbf{x}) \leq t$ és $d(\mathbf{v},\mathbf{x}) \leq t$, azaz $2t = t + t \geq d(\mathbf{u},\mathbf{x}) + d(\mathbf{x},\mathbf{v}) \geq d(\mathbf{u},\mathbf{v}) \geq d$, tehát $t \geq \frac{d}{2}$, így $t < \frac{d}{2}$ esetén a C-beli középpontok köré írt gömbök páronként diszjunktak.

Nézzük a második állítást. Korábban beláttuk, hogy van olyan $\mathbf{u}$ és $\mathbf{v} \neq \mathbf{u}$ vektor C-ben, amelyekkel $d(\mathbf{u},\mathbf{v}) = d$. Ez azt jelenti, hogy $\mathbf{u}$ és $\mathbf{v}$ pontosan d $(\leq n)$ helyen tér el egymástól, mondjuk a $0 \leq i_1 < \cdots < i_d < n$ indexű komponensekben. Legyen $\mathbf{z} \in S^n$ $\mathbf{u}$-val azonos, kivéve az előbbi indexek közül $\left\lceil \frac{d}{2} \right\rceil$ helyet, ahol legyen $\mathbf{v}$-vel egyenlő. Ekkor $d(\mathbf{u},\mathbf{z}) = \left\lceil \frac{d}{2} \right\rceil$ és $d(\mathbf{z},\mathbf{v}) = d - \left\lceil \frac{d}{2} \right\rceil = \left\lfloor \frac{d}{2} \right\rfloor$. $\frac{d}{2} \leq t \in \mathbb{N}$, továbbá $\left\lfloor \frac{d}{2} \right\rfloor \leq \frac{d}{2} \leq \left\lceil \frac{d}{2} \right\rceil < \frac{d}{2} + 1$, így $t \geq \left\lceil \frac{d}{2} \right\rceil$, ezért $d(\mathbf{u},\mathbf{z}) = \left\lceil \frac{d}{2} \right\rceil \leq t$ és $d(\mathbf{z},\mathbf{v}) = \left\lfloor \frac{d}{2} \right\rfloor \leq t$,

azaz $\mathbf{z} \in G_t(\mathbf{u})$ és $\mathbf{z} \in G_t(\mathbf{v})$, vagyis $\mathbf{z} \in G_t(\mathbf{u}) \cap G_t(\mathbf{v})$. Ebből következik, hogy ez a C-beli középpontú két gömb nem idegen.

□

2.6. Következmény

Legyen $S \neq \emptyset$ véges halmaz, $n \in \mathbb{N}^+$, $C \subseteq S^n$, $|C| \geq 2$, $\mathbf{u} \in C$ és $\mathbf{x} \in S^n$. Ha $d(\mathbf{u}, \mathbf{x}) < \frac{d}{2}$, akkor minden $C\backslash\{\mathbf{u}\}$-beli $\mathbf{v}$ vektorra $d(\mathbf{u}, \mathbf{x}) < d(\mathbf{v}, \mathbf{x})$, míg $d(\mathbf{u}, \mathbf{x}) \geq \frac{d}{2}$ esetén ez nem feltétlenül igaz.

Δ

A tétel lényege, hogy $\frac{d}{2}$ egy vízválasztó. Amennyiben egy kódszó az átvitel során ennél kevesebb helyen hibásodik meg, akkor a megérkezett szó a kódszavak közül az eredeti, elküldött kódszóhoz van legközelebb, attól tér el a legkevesebb helyen, viszont ha a hibák száma legalább ennyi, akkor ez nem feltétlenül igaz, sőt, biztosan van olyan kódszó és olyan hiba, amikor ez nem igaz.

Bizonyítás:

Legyen $\mathbf{v} \in C\backslash\{\mathbf{u}\}$, és $d(\mathbf{u}, \mathbf{x}) < \frac{d}{2}$. Ekkor $\frac{d}{2} + d(\mathbf{x}, \mathbf{v}) > d(\mathbf{u}, \mathbf{x}) + d(\mathbf{x}, \mathbf{v}) \geq d(\mathbf{u}, \mathbf{v}) \geq d$-ből átrendezés után kapjuk, hogy $d(\mathbf{x}, \mathbf{v}) > d - \frac{d}{2} = \frac{d}{2} > d(\mathbf{u}, \mathbf{x})$. Ha viszont $\mathbf{u}$, $\mathbf{v}$ és $\mathbf{x} = \mathbf{z}$ az előző tétel bizonyításában szereplő három elem, úgy $d(\mathbf{u}, \mathbf{x}) = \left\lceil \frac{d}{2} \right\rceil \geq \frac{d}{2} \geq \left\lfloor \frac{d}{2} \right\rfloor = d(\mathbf{x}, \mathbf{v})$.

□

Hiba jelzésére alkalmas blokk-kódot könnyű szerkeszteni: ehhez elegendő, ha C valódi része S^n-nek, hiszen ha vételnél egy $S^n\backslash C$ elemet találunk, biztosak lehetünk benne, hogy hiba történt az átvitel során. Tovább visszük a gondolatot: partícionáljuk S^n-t, és legyen C olyan, hogy minden osztállyal legfeljebb egy közös pontja van. Ezt úgy használhatjuk hibajavításra, hogy amennyiben a vett jel egy olyan osztályban van, amelyben található kódszó, akkor úgy tekintjük, mintha ez a kódszó lett volna az üzenet, ellenkező esetben jelezzük, hogy hibás volt az átvitel. Természetesen a jelzés elmarad, ha a továbbítás során úgy változott meg a közlemény, hogy az eredmény is eleme C-nek, illetve javító kód esetén maga is egy kódszó, hiszen ekkor a hiba rejtve marad; hasonlóan hibajavítás esetén helytelenül korrigálunk, ha a vett szó nem abba az osztályba esik, amelyben az eredeti található, de olyanba, amelyben van reprezentáns. A probléma mindkét módszernél az S^n halmaz megfelelő felosztása, és javítás esetén a reprezentánsok megfelelő kiválasztása. Ez utóbbi a **dekódolás** problémája.

2.7. Definíció

Legyen A és S nem üres véges halmaz, $|S| = q$, $n \in \mathbb{N}^+$, és $\varphi: A \to S^n$ injektív. A φ A^+-ra való homomorf kiterjesztése által meghatározott betűnkénti kód **blokk-kód**.

Δ

Rögtön látható, hogy a fenti C kód egy $(n, M)_q$-paraméterű kód, ahol $M = |A|$. Egyébként a blokk-kódolás mellett létezik más kódolási eljárás is, ezzel azonban nem foglalkozunk.

Most egy speciális dekódolási sémát ismertetünk.

2.8. Definíció

Ha $S \neq \emptyset$ véges halmaz, $n \in \mathbb{N}^+$, $C \subseteq S^n$ egy (n, M)-paraméterű kód, és $f: S^n \to C$ olyan, hogy minden $\mathbf{v} \in S^n$-re $d\big(\mathbf{v}, f(\mathbf{v})\big) = \min\{d(\mathbf{u}, \mathbf{v}) | \mathbf{u} \in C\}$, akkor f **minimális távolságú dekódolás**.

Δ

A hibajavítás minimális távolságú dekódolás esetén tehát úgy történik, hogy amikor beérkezik n szimbólum, akkor megkeressük azt a (illetve egy olyan) kódszót, amely a legkevesebb helyen tér el a vett n-estől. Ezt vagy úgy tesszük, hogy a vétel helyén csak a kódszavakat tároljuk, és a beérkezett szót mindegyik kódszóval összehasonlítva kikeressük a(z egyik) legközelebb fekvő kódszót, vagy tároljuk az összes lehetséges szót, és mindegyikhez a hozzá legközelebb lévő (egyik) kódszót, vagyis a döntési függvényt, és ez esetben csupán a táblázatban kell kikeresni a beérkezett szót, és a hozzá tartozó kódszó megadja a dekódolást. Az előbbi esetben kisebb tárra, de hosszabb számolásra van szükség, míg a második esetben fordított a helyzet, vagyis most is a számítástechnikában szokásos tárméret - futási idő cserearány problémájával állunk szemben.

A blokk-kódolás esetén a minimális távolságú dekódolás szinte kizárólagos, jóllehet nem minden esetben eredményezi a legkisebb hibavalószínűséget.

2.9. Definíció

Legyen $t \in \mathbb{N}$. Egy kód **t-hiba jelző**, ha tetszőleges üzenetben előforduló minden legfeljebb t számú hibát képes jelezni, és **t-hiba javító**, ha tetszőleges üzenetben előforduló minden legfeljebb t számú hibát képes javítani. A kód **pontosan t-hiba jelző**, amennyiben t-hiba jelző, de van olyan $t+1$ hiba, amelyet nem jelez, és **pontosan t-hiba javító**, ha t-hiba javító, de van olyan $t+1$ hiba, amelyet nem javít, vagy hibásan javít.

Δ

A definícióban lényeges, hogy ha egy kód pontosan t-hiba jelző, az nem jelenti azt, hogy t-nél több hibát nem képes jelezni, csupán azt, hogy van legalább egy ilyen $t+1$ hibát tartalmazó hibaminta. Ha visszagondolunk a bevezetőben említett paritásbites kódra, tehát ahol egy n-bites bináris szót úgy toldottunk meg egy bittel, hogy a keletkezett $n+1$-bites szóban az 1-esek száma páros legyen, akkor tudjuk, hogy ez a kód minden olyan esetben jelez, ha a hibák száma páratlan, de soha nem jelez, ha páros számú helyen történt hiba, vagyis ez a kód 1-hiba jelző, jóllehet bármely olyan esetben jelzi, hogy hiba történt, ha például a hibák száma három. Olyan pontosan t-hiba jelző kódra is lehetne példát mutatni, amelyben van olyan $t+1$ hibát tartalmazó hibaminta, amelyet képes a rendszer jelezni.

Az előbbi megállapítások igazak a hibajavításra is.

2.10. Tétel

Egy (n, M, d)-kód akkor és csak akkor t-hiba jelző, ha $t < d$, és akkor és csak akkor pontosan t-hiba jelző, ha $t = d-1$. Minimális távolságú dekódolással a kód akkor és csak akkor t-hiba javító, ha $t < \frac{d}{2}$, és akkor és csak akkor pontosan t-hiba javító, ha $t = \left\lfloor \frac{d-1}{2} \right\rfloor$.

Δ

Bizonyítás:

Minden d-távolságú kódban van olyan $\mathbf{u}, \mathbf{v}$ pár, amelyre $d(\mathbf{u}, \mathbf{v}) = d$. Ha egy ilyen $\mathbf{u}$ üzenetben a d hiba úgy lép fel, hogy $\mathbf{u}$ átmegy $\mathbf{v}$-be, akkor a hibát nem tudjuk jelezni; viszont $\mathbf{u}$-ban bárhogy lép is fel d-nél kevesebb hiba, a keletkező elem nem lehet kódszó, mivel két kódszó között legalább d pozícióban különbség van.

Korábban láttuk, hogy $t < \frac{d}{2}$ esetén a vett szó az eredeti kódszóhoz van legközelebb, minden más kódszó távolabb esik a vett szótól, ezért minimális távolságú dekódolással helyesen döntünk, a dekódolás helyes. Mivel abban az esetben, ha t egész szám, $t < \frac{d}{2}$ ekvivalens a $t \leq \left\lfloor \frac{d-1}{2} \right\rfloor$ relációval, ezért ez egyben azt is jelenti, hogy minimális távolságú dekódolással minden olyan esetben helyesen javítunk, amikor a hibák száma nem nagyobb, mint $\left\lfloor \frac{d-1}{2} \right\rfloor$. Ha viszont $t \geq \frac{d}{2}$, akkor van két nem idegen, kódszó-középpontú gömb. Legyen $\mathbf{u}$ és $\mathbf{v}$ két olyan kódszó, amelyek távolsága éppen d, és $\mathbf{x}$ az a szó, amely $\mathbf{u}$-tól $\left\lfloor \frac{d}{2} \right\rfloor$, $\mathbf{v}$-től pedig $\left\lceil \frac{d}{2} \right\rceil$ távolságra van. Tudjuk, hogy ekkor minden más kódszótól is leg-

alább $\left\lceil \frac{d}{2} \right\rceil$ távolságra fekszik $\mathbf{x}$. Ha az előbbi két távolság nem azonos, akkor nyilván $\psi(\mathbf{x}) = \mathbf{u}$, ellenkező esetben $\psi(\mathbf{x})$ bármely olyan kódszó lehet, amely $\mathbf{x}$-től $\frac{d}{2}$ távolságra van, tehát lehet például ismét $\mathbf{u}$. Ekkor abban az esetben, ha $\mathbf{v}$-t küldjük, és a beérkezett szó $\mathbf{x}$, akkor a hibák száma $\left\lceil \frac{d}{2} \right\rceil = \left\lfloor \frac{d-1}{2} \right\rfloor + 1$, és ezt a vett szót hibásan javítjuk, hiszen nem $\mathbf{v}$-re, hanem $\mathbf{u}$-ra döntünk, ami azt jelenti, hogy egy d-távolságú kód esetén minimális távolságú dekódolással minden, legfeljebb $\left\lfloor \frac{d-1}{2} \right\rfloor$ hiba javítható, de van olyan $\left\lfloor \frac{d-1}{2} \right\rfloor + 1$ hiba, amelyet rosszul javítunk, így a minimális távolságú dekódolással a d-távolságú kód pontosan $\left\lfloor \frac{d-1}{2} \right\rfloor$-hiba javító.

□

A 2.8. definíció után felvázoltuk, hogy hogyan történhet általános esetben a minimális távolságú dekódolás. Ha például $n = 50$, és a kód bináris, akkor $|S^n| = 2^{50}$, vagyis a másodikként említett dekódolási eljárással ennyi szót és a hozzá tartozó kódszót kellene tárolnunk (egyenként 50 bites adatokként). Ha viszont csak a kódszavakat tároljuk, és mondjuk $M = 2^{40}$, akkor a vett szót 2^{40} különböző kódszóval kell összehasonlítani, hogy kiválasszuk a vett szóhoz legközelebbi kódszót. Látható, hogy egyik módszer sem túlságosan kedvező (az egyik a tárigény, a másik a futási idő szempontjából nem polinomiális algoritmus). A dolgon úgy tudunk segíteni, ha valamilyen egyéb módszerrel tudunk következtetni a vett szó alapján az elküldött kódszóra, vagyis ha valamilyen módon ki tudjuk számolni a kódszót a beérkezett hibás szóból. Ehhez a kódba valamilyen matematikai struktúrát építünk.

3. Az entrópiáról

Az **entrópia** információelméleti fogalmát Shannon határozta meg. Korábban **Heartley** vizsgálta matematikai szempontból az információt, és úgy találta, hogy ha n különböző üzenet lehetséges, akkor egy-egy **üzenet információtartalma**, az **egyedi információmennyiség** $I = \log n$. E szerint a kifejezés szerint azonban a különböző üzenetek azonos mennyiségű információt, új ismeretet közölnek a fogadóval. Ezzel szemben Shannon úgy gondolta, hogy egy üzenet annál több információt szolgáltat, minél váratlanabb, minél kevésbé lehet rá számítani, azaz minél kisebb a valószínűsége. Ha X egy véges eseménytér, az üzenetek halmaza, és az $x_i \in X$ üzenet p_i valószínűséggel fordul elő, akkor tehát Shannon szerint az x_i üzenet $I(p_i)$ információt szolgáltat, ahol I egyelőre ismeretlen függvény. A teljes **üzenethalmaz átlagos információtartalma** az egyes üzenetek egyedi információtartalmának várható értéke, $H(p_{n-1}, \cdots, p_0) = \sum_{i=0}^{n-1} p_i I(p_i)$, ahol n a különböző üzenetek száma, és H az **entrópiafüggvény**. Mivel egyelőre I ismeretlen, ezért H-t sem ismerjük. H meghatározásához bizonyos feltételeket kell megfogalmazni. Egy lehetséges axiomatikus bevezetés az alábbi kikötéseket tartalmazza:

1. $(p_{n-1}, \cdots, p_0) \in]0,1]^n \subseteq \mathbb{R}^n$ véges diszkrét valószínűségi eloszlás;
2. $H(p_{n-1}, \cdots, p_0)$ a változóinak szimmetrikus függvénye, azaz ha π az $\{i \in \mathbb{N} | i < n\}$ halmaz tetszőleges permutációja, akkor $H(p_{n-1}, \cdots, p_0) = H\left(p_{\pi(n-1)}, \cdots, p_{\pi(0)}\right)$;
3. $H(tp_{n-1}, (1-t)p_{n-1}, \cdots, p_0) = H(p_{n-1}, \cdots, p_0) + p_{n-1} H(t, 1-t)$, ha $t \in]0,1[\subseteq \mathbb{R}$;
4. $H(t, 1-t)$ t-nek folytonos függvénye, ha $t \in]0,1[\subseteq \mathbb{R}$;
5. $H\left(\frac{1}{2}, \frac{1}{2}\right) > 0$.

A fenti feltételeknek pontosan egy folytonos függvény, $H(p_{n-1}, \cdots, p_0) = -\sum_{i=0}^{n-1} p_i \log p_i$ felel meg, és ebből leolvasva $I(p_i) = -\log p_i$. Ha minden üzenet valószínűsége azonos, tehát bármelyik $\frac{1}{n}$ valószínűséggel fordul elő, akkor valóban igaz, hogy az egyedi üzenetek által közvetített információ mértéke $I = \log n$. Általános esetben viszont az egyes üzenetek bekövetkezése különböző valószínűséggel történik, tehát általában $I(p_i) \neq \log n$. A valós értékű logaritmusfüggvény csak a pozitív valós számokra értelmezett, és ha x a pozitív valós számokon keresztül tart a 0-hoz, akkor a logaritmusfüggvény értéke abszolút értékben a ∞-hez tart, így $|x \log x| \to 0 \cdot \infty$. De $\lim_{x \to 0+0}(x \log x)$ létezik, és 0-val egyenlő, ezért az entrópiafüggvényt kiterjeszthetjük arra az esetre is, amikor egy vagy több valószínűség értéke 0, azzal, hogy ekkor $p_i \log p_i = 0$.

Az előbbi felírásban nem adtuk meg konkrétan a logaritmus alapját, ám erre nincs is szükség. Ha ugyanis egy alapról áttérünk egy másikra, az csupán a mértékegység megváltozását jelenti (hasonlóan mondjuk a méterhez és lábhoz), hiszen $\log_a u = \log_a b \cdot \log_b u$. Magát a logaritmus r alapját $H\left(\frac{1}{2}, \frac{1}{2}\right) = c$ határozza meg, ugyanis $c = H\left(\frac{1}{2}, \frac{1}{2}\right) = -\left(\frac{1}{2}\log_r \frac{1}{2} + \frac{1}{2}\log_r \frac{1}{2}\right) = \log_r 2$-ből $r = 2^{\frac{1}{c}} > 1$. Az alap szokásos értéke az információelméletben 2, és ekkor az entrópia egysége a bit. Ezt az elnevezést **John W. Tukey** vezette be a **binary digit** rövidítéseként. Tekintettel arra, hogy ugyanez a neve egy kettes számrendszerben felírt szám egy-egy számjegyének, ezért megkülönböztetésül az információelméleti egységet szokás **binary unit**-ként, a **binary unit** rövidítéseként említeni.

Ha a p_i valószínűségek az X eseményhalmaz elemei előfordulásainak a valószínűségei, akkor $H(p_{n-1}, \cdots, p_0)$ tulajdonképpen az X tér entrópiája, ezért ezt az értéket $H(X)$-szel is jelölhetjük.

Csupán az érdekesség, és bizonyos patriotikus büszkeség miatt jegyezzük meg, hogy Shannonnak **Neumann János** javasolta az entrópia elnevezést, lévén, hogy a kifejezés matematikailag hasonló alakú, mint a korábbi fizikai entrópia. Az elnevezést a formális hasonlóságon kívül bizonyos tartalmi azonosságok is alátámasztják, bár igen komoly eltérések is kimutathatóak a két entrópiafogalom között, amiért többen károsnak tartják az azonos megnevezést. Shannonnak más „magyar kapcsolata" is volt: foglalkozott sakkautomatával, és ezzel kapcsolatban megemlítette **Kempelen Farkas** nevét, valamint a kommunikációról szólva **Gábor Dénes**t nevezi meg egyik úttörőként.

A fenti H-függvény a **Shannon-féle entrópiafüggvény**. Léteznek általánosabb kifejezések is az entrópiára. Egyik a $H_\alpha(p_{n-1},\cdots,p_0)=\frac{1}{1-\alpha}\log\sum_{i=0}^{n-1}p_i^\alpha$ **Rényi-féle entrópia**, ahol $1>\alpha\in\mathbb{R}_0^+$. Ez a kifejezés határértékként tartalmazza a Shannon-féle entrópiát, ha α balról tart 1-hez.

A továbbiakban részletesebben megvizsgáljuk az entrópiát. Mindenekelőtt bizonyítás nélkül ismertetjük a konvex függvények néhány jellemzőjét.

3.1. Definíció

Legyen $f:X\to\mathbb{R}$, ahol $X\subseteq\mathbb{R}$, és $I\subseteq X$ egy intervallum, továbbá a és $b\neq a$ az I két eleme. Ekkor $h_{a,b}=f(a)+\frac{f(b)-f(a)}{b-a}(x-a)$ az $f\left(a,f(a)\right),\left(b,f(b)\right)$ **pontjain átmenő szelő**je, és ennek a két pont közé eső része az $f\left(a,f(a)\right),\left(b,f(b)\right)$ **pontjait összekötő húr**.

Δ

3.2. Definíció

Legyen $f:X\to\mathbb{R}$, ahol $X\subseteq\mathbb{R}$, és $I\subseteq X$ egy intervallum. f **konvex az** I **intervallumon**, ha az I bármely $a<c<b$ elemeire $f(c)\leq h_{a,b}(c)$. f **szigorúan konvex az** I **intervallumon**, ha konvex I-n, és az előbbi egyenlőtlenségben mindig a szigorú egyenlőtlenség is teljesül. f **konkáv** illetve **szigorúan konkáv az** I **intervallumon**, ha $-f$ konvex illetve szigorúan konvex I-n.

Δ

3.3. Tétel

Legyen $f:X\to\mathbb{R}$, ahol $X\subseteq\mathbb{R}$, és $I\subseteq X$ egy intervallum. f akkor és csak akkor konvex az I intervallumon, ha az I bármely a, b elemére és tetszőleges $0<\lambda<1$ valós számra $f(\lambda a+(1-\lambda)b)\leq\lambda f(a)+(1-\lambda)f(b)$, és akkor és csak akkor szigorúan konvex, ha $a\neq b$ esetén az előbbi egyenlőtlenség bal oldala mindig kisebb a jobb oldali értéknél.

Δ

3.4. Tétel (Jensen-egyenlőtlenség)

Legyen $f:X\to\mathbb{R}$, ahol $X\subseteq\mathbb{R}$, és $I\subseteq X$ egy intervallum. f akkor és csak akkor konvex az I intervallumon, ha minden $n\in\mathbb{N}^+$, $\{a_i\in I|n\geq i\in\mathbb{N}^+\}$ és $\{0<\lambda_i\in\mathbb{R}|n\geq i\in\mathbb{N}^+\wedge\sum_{i=1}^n\lambda_i=1\}$ esetén $f(\sum_{i=1}^n\lambda_i a_i)\leq\sum_{i=1}^n\lambda_i f(a_i)$, és akkor és csak akkor szigorúan konvex, ha az előbbi egyenlőtlenség bal oldala mindig kisebb a jobb oldali értéknél, ha az a_i-k nem mindegyike azonos.

Δ

3.5. Megjegyzés

Ha $n\in\mathbb{N}^+$, minden $n\geq i\in\mathbb{N}^+$-ra $\mu_i\in\mathbb{R}^+$, és $\sum_{i=1}^n\mu_i=\mu$, akkor minden előbbi i indexre $0<\lambda_i=\frac{\mu_i}{\mu}\in\mathbb{R}$, és $\sum_{i=1}^n\lambda_i=1$.

Δ

3.6. Tétel

Legyen $n\in\mathbb{N}^+$, $n\geq i\in\mathbb{N}^+$-ra $a_i\in\mathbb{R}_0^+$ és $b_i\in\mathbb{R}^+$, $a=\sum_{i=1}^n a_i$ és $b=\sum_{i=1}^n b_i$, és $1<t\in\mathbb{R}$. Ekkor $\sum_{i=1}^n a_i\log_t\frac{b_i}{a_i}\leq a\log_t\frac{b}{a}$, és egyenlőség akkor és csak akkor teljesül, ha minden i-re $\frac{b_i}{a_i}=\frac{b}{a}$.

Δ

Bizonyítás:
1-nél nagyobb alap esetén a logaritmusfüggvény a teljes értelmezési tartományában szigorúan konkáv. Ekkor a Jensen-egyenlőtlenséggel

$$\sum_{i=1}^{n} a_i \log\frac{b_i}{a_i} = a\sum_{i=1}^{n}\frac{a_i}{a}\log\frac{b_i}{a_i} \le a\log\sum_{i=1}^{n}\frac{a_i}{a}\frac{b_i}{a_i} = a\log\frac{b}{a},$$

és egyenlőség akkor és csak akkor lesz, ha minden i-re $\frac{b_i}{a_i}$ azonos. Legyen $\frac{b_i}{a_i} = c$. Ekkor $b_i = ca_i$, tehát $b = ca$, és innen $c = \frac{b}{a}$.

□

3.7. Következmény

Ha az előbbi tételben

a) $a = b$ (speciális esetként $a = 1 = b$), akkor $\sum_{i=1}^{n} a_i \log\frac{b_i}{a_i} \le 0$, vagyis $\sum_{i=1}^{n} a_i \log b_i \le \sum_{i=1}^{n} a_i \log a_i$, és a két oldal akkor és csak akkor egyenlő, ha valamennyi i-re $a_i = b_i$;
b) $a = 1$ és $n \ge i \in \mathbb{N}^+$-ra $b_i = 1$, akkor $-\sum_{i=1}^{n} a_i \log a_i = \sum_{i=1}^{n} a_i \log\frac{b_i}{a_i} \le \log n$, és pontosan akkor lesz a két oldal egyenlő, ha valamennyi i-re $a_i = \frac{1}{n}$.

Δ

Bizonyítás:
Mindkét állítás közvetlenül kapható az előző tételből.

□

3.8. Definíció

Legyen $m \in \mathbb{N}^+$, $n \in \mathbb{N}^+$, $X = \left\{\underline{x}_k \in \mathbb{R}^m \middle| n \ge k \in \mathbb{N}^+\right\}$, $\underline{\xi} = (\xi_1, \cdots, \xi_m)$ valószínűségi változó X-en, $n \ge k \in \mathbb{N}^+$-ra $p_k = P\left(\underline{\xi} = \underline{x}_k\right)$ és $1 < r \in \mathbb{R}$. Ekkor $H_r = H_r\left(\underline{\xi}\right) = -\sum_{k=1}^{n} p_k \log_r p_k$ a $\underline{\xi}$ **entrópiája**, és $i_{\underline{\xi}} = -\log_r p\left(\underline{\xi}\right)$ az **entrópia-sűrűség** vagy **egyedi entrópia**.

Δ

$r = 2$-nél az entrópia egysége a **bit**. Ha nem szükséges, külön nem jelöljük r-et.

3.9. Tétel

Legyen $m \in \mathbb{N}^+$, $n \in \mathbb{N}^+$, $\underline{\xi} = (\xi_1, \cdots, \xi_m)$ valószínűségi változó, $X = \left\{\underline{x}_k \in \mathbb{R}^m \middle| n \ge k \in \mathbb{N}^+\right\}$ $\underline{\xi}$ lehetséges értékeinek halmaza, és legyen $n \ge k \in \mathbb{N}^+$-ra $p_k = P\left(\underline{\xi} = \underline{x}_k\right)$. Ekkor $0 \le H \le \log n$, továbbá $H = 0$ akkor és csak akkor, ha van olyan l, amellyel $p_l = P\left(\underline{\xi} = \underline{x}_l\right) = 1$ és minden más k-ra $p_k = P\left(\underline{\xi} = \underline{x}_k\right) = 0$, míg $H = \log n$ pontosan akkor igaz, ha minden k-ra $p_k = P\left(\underline{\xi} = \underline{x}_k\right) = \frac{1}{n}$.

Δ

Bizonyítás:
Mivel $0 \le p_k \le 1$, és a logaritmus alapszáma 1-nél nagyobb, ezért $\log p_k \le 0$ és $p_k \log p_k \le 0$, amiből következik, hogy $-\sum_{k=1}^{n} p_k \log p_k \ge 0$, és az összeg csak úgy lehet 0, ha minden tagja 0. Ha

$0 < p_k < 1$, akkor $\log p_k < 0$ és $p_k \log p_k \neq 0$, így $H = 0$ esetén minden k-ra p_k csak 0 vagy 1 lehet. Ugyanakkor $\sum_{k=1}^{n} p_k = 1$, ezért nem lehet minden k-ra $p_k = 0$ és nem lehet egynél több k-ra $p_k = 1$, vagyis $H = 0$ akkor és csak akkor, ha egy és csak egy k indexre $p_k = 1$, és minden más indexre $p_k = 0$.

Ami az entrópia maximumát illeti, az a 3.7. Következmény a) pontjából adódik.

□

A $H(p_{n-1}, \cdots, p_0)$ függvénynek a fentiek szerint az értelmezési tartományában pontosan egy maximuma van, a $(p_{n-1}, \cdots, p_0) = \left(\frac{1}{n}, \cdots, \frac{1}{n}\right)$ helyen, és ekkor az értéke $\log n$. Ez az entrópia intuitív értelmezése alapján világos, hiszen átlagosan akkor jutunk a legtöbb információhoz, akkor lehet a legkevésbé megjósolni a soron következő üzenetet, ha lényegében véve semmit sem tudunk az egyes üzenetekről, bármelyik esemény azonos valószínűséggel következhet be.

Most legyen $\underline{\eta} = (\eta_1, \cdots, \eta_u)$ is egy valószínűségi változó az $Y = \left\{\underline{y}_k \in \mathbb{R}^u \middle| v \geq k \in \mathbb{N}^+\right\}$ értékekkel és $q_k = P\left(\underline{\eta} = \underline{y}_k\right)$ valószínűségekkel. Ekkor valamennyi rögzített $v \geq l \in \mathbb{N}^+$ indexre létezik a $t_{k|l} = P\left(\underline{\xi} = \underline{x}_k \middle| \underline{\eta} = \underline{y}_l\right)$ feltételes eloszlás és a $H\left(\underline{\xi} \middle| \underline{\eta} = \underline{y}_l\right) = -\sum_{k=1}^{n} t_{k|l} \log t_{k|l}$ entrópia.

3.10. Definíció

Legyen $\underline{\xi} = (\xi_1, \cdots, \xi_m)$ és $\underline{\eta} = (\eta_1, \cdots, \eta_u)$ valószínűségi változó és $\underline{\eta}$ lehetséges értékeinek halmaza $Y = \left\{\underline{y}_k \in \mathbb{R}^u \middle| v \geq k \in \mathbb{N}^+\right\}$. Ekkor $H\left(\underline{\xi} \middle| \underline{\eta}\right) = E\left(H\left(\underline{\xi} \middle| \underline{\eta} = \underline{y}_l\right)\right)$ a **$\underline{\xi}$-nek $\underline{\eta}$-ra vonatkozó feltételes entrópiája**.

Δ

3.11. Tétel

Legyen $m \in \mathbb{N}^+$, $n \in \mathbb{N}^+$, $u \in \mathbb{N}^+$, $v \in \mathbb{N}^+$, $\underline{\xi} = (\xi_1, \cdots, \xi_m)$ és $\underline{\eta} = (\eta_1, \cdots, \eta_u)$ valószínűségi változó, $X = \left\{\underline{x}_k \in \mathbb{R}^m \middle| n \geq k \in \mathbb{N}^+\right\}$ a $\underline{\xi}$, $Y = \left\{\underline{y}_k \in \mathbb{R}^u \middle| v \geq k \in \mathbb{N}^+\right\}$ az $\underline{\eta}$ lehetséges értékeinek halmaza, $n \geq k \in \mathbb{N}^+$-ra és $v \geq l \in \mathbb{N}^+$-ra $r_{k,l} = P\left(\underline{\xi} = \underline{x}_k, \underline{\eta} = \underline{y}_l\right)$, és $t_{k|l} = P\left(\underline{\xi} = \underline{x}_k \middle| \underline{\eta} = \underline{y}_l\right)$. Ekkor $H\left(\underline{\xi} \middle| \underline{\eta}\right) = -\sum_{l=1}^{v} \sum_{k=1}^{n} r_{k,l} \log t_{k|l}$, $0 \leq H\left(\underline{\xi} \middle| \underline{\eta}\right) \leq H\left(\underline{\xi}\right)$, és $H\left(\underline{\xi} \middle| \underline{\eta}\right) = 0$ akkor és csak akkor, ha létezik olyan f függvény, hogy 1 valószínűséggel $\underline{\xi} = f\left(\underline{\eta}\right)$, míg $H\left(\underline{\xi} \middle| \underline{\eta}\right) = H\left(\underline{\xi}\right)$ pontosan akkor teljesül, ha $\underline{\xi}$ és $\underline{\eta}$ függetlenek.

Δ

Bizonyítás:

$H\left(\underline{\xi} \middle| \underline{\eta} = \underline{y}_l\right)$ entrópia, így $0 \leq H\left(\underline{\xi} \middle| \underline{\eta} = \underline{y}_l\right)$, de akkor $H\left(\underline{\xi} \middle| \underline{\eta}\right) = E\left(H\left(\underline{\xi} \middle| \underline{\eta} = \underline{y}_l\right)\right) \geq 0$ is teljesül, és a várható érték akkor és csak akkor lesz 0, ha minden l-re $H\left(\underline{\xi} \middle| \underline{\eta} = \underline{y}_l\right) = 0$. Ez akkor és csak akkor teljesül, ha minden l indexhez pontosan egy i_l indexre $t_{i_l|l} = 1$, és minden más indexre $t_{k|l} = 0$. Ez azt jelenti, hogy minden $\underline{y}_l$-hez van egy és csak egy $\underline{x}_{i_l}$, hogy $P\left(\underline{\xi} = \underline{x}_{i_l} \middle| \underline{\eta} = \underline{y}_l\right) = 1$, és minden más k indexre $P\left(\underline{\xi} = \underline{x}_k \middle| \underline{\eta} = \underline{y}_l\right) = 0$, vagyis létezik egy f függvény, amellyel 1 valószínűséggel $\underline{\xi} = f\left(\underline{\eta}\right)$, $\underline{\xi}$ az $\underline{\eta}$ függvénye.

Nézzük a másik határt. Legyen $q_k = P\left(\underline{\eta} = \underline{y}_k\right)$. Ekkor

$$H\left(\underline{\xi}\right) - H\left(\underline{\xi}\middle|\underline{\eta}\right) = -\sum_{k=1}^{n} p_k \log p_k + \sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \log t_{k|l}$$
$$= -\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \log p_k + \sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \log t_{k|l}$$
$$= -\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \log \frac{p_k}{t_{k|l}} = -\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \log \frac{p_k q_l}{r_{k,l}} \geq 0,$$

mert $\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \log \frac{p_k q_l}{r_{k,l}} \leq \log \sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \frac{p_k q_l}{r_{k,l}} = \log \sum_{l=1}^{v}\sum_{k=1}^{n} p_k q_l$ (lásd a Jensen-egyenlőtlenséget), és $\sum_{l=1}^{v}\sum_{k=1}^{n} p_k q_l = (\sum_{l=1}^{v} p_k)(\sum_{k=1}^{n} q_l)$=1. $H\left(\underline{\xi}\right) - H\left(\underline{\xi}\middle|\underline{\eta}\right) \geq 0$-ból $H\left(\underline{\xi}\right) \geq H\left(\underline{\xi}\middle|\underline{\eta}\right)$, továbbá egyenlőség akkor és csak akkor lesz, ha minden k, l indexre $p_k q_l = r_{k,l}$, vagyis pontosan akkor, ha $\underline{\xi}$ és $\underline{\eta}$ függetlenek.

□

Az előbbihez hasonlóan definiáljuk a $H\left(\underline{\eta}\middle|\underline{\xi}\right)$ feltételes entrópiát, és erre hasonló tulajdonság igaz, mint az előző entrópiára.

3.12. Definíció

Legyen $m \in \mathbb{N}^+$, $n \in \mathbb{N}^+$, $u \in \mathbb{N}^+$, $v \in \mathbb{N}^+$, $\underline{\xi} = (\xi_1, \cdots, \xi_m)$ és $\underline{\eta} = (\eta_1, \cdots, \eta_u)$ valószínűségi változó, $X = \left\{\underline{x}_k \in \mathbb{R}^m \middle| n \geq k \in \mathbb{N}^+\right\}$ a $\underline{\xi}$, $Y = \left\{\underline{y}_k \in \mathbb{R}^u \middle| v \geq k \in \mathbb{N}^+\right\}$ az $\underline{\eta}$ lehetséges értékeinek halmaza, és $n \geq k \in \mathbb{N}^+$-ra és $v \geq l \in \mathbb{N}^+$-ra $r_{k,l} = P\left(\underline{\xi} = \underline{x}_k, \underline{\eta} = \underline{y}_l\right)$. Ekkor $\underline{\xi}$ **és** $\underline{\eta}$ **együttes entrópiája** $H\left(\underline{\xi}, \underline{\eta}\right) = -\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l} \log r_{k,l}$.

Δ

3.13. Tétel

Legyen $m \in \mathbb{N}^+$, $n \in \mathbb{N}^+$, $u \in \mathbb{N}^+$, $v \in \mathbb{N}^+$, $\underline{\xi} = (\xi_1, \cdots, \xi_m)$ és $\underline{\eta} = (\eta_1, \cdots, \eta_u)$ valószínűségi változó, $X = \left\{\underline{x}_k \in \mathbb{R}^m \middle| n \geq k \in \mathbb{N}^+\right\}$ a $\underline{\xi}$, $Y = \left\{\underline{y}_k \in \mathbb{R}^u \middle| v \geq k \in \mathbb{N}^+\right\}$ az $\underline{\eta}$ lehetséges értékeinek halmaza, és $n \geq k \in \mathbb{N}^+$-ra és $v \geq l \in \mathbb{N}^+$-ra $r_{k,l} = P\left(\underline{\xi} = \underline{x}_k, \underline{\eta} = \underline{y}_l\right)$. Ekkor $0 \leq H\left(\underline{\xi}, \underline{\eta}\right) = H\left(\underline{\xi}\right) + H\left(\underline{\eta}\middle|\underline{\xi}\right) = H\left(\underline{\eta}\right) + H\left(\underline{\xi}\middle|\underline{\eta}\right) \leq H\left(\underline{\xi}\right) + H\left(\underline{\eta}\right)$, és $H\left(\underline{\xi}, \underline{\eta}\right) = H\left(\underline{\xi}\right) + H\left(\underline{\eta}\right)$ akkor és csak akkor, ha $\underline{\xi}$és $\underline{\eta}$ függetlenek.

Δ

Bizonyítás:

$H\left(\underline{\xi}, \underline{\eta}\right) \geq 0$ a definíció közvetlen következménye, továbbá ha $t_{k|l} = P\left(\underline{\xi} = \underline{x}_k \middle| \underline{\eta} = \underline{y}_l\right)$ valamint $q_k = P\left(\underline{\eta} = \underline{y}_l\right)$, akkor

$$H\left(\underline{\xi},\underline{\eta}\right) = -\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l}\log r_{k,l} = -\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l}\log\left(t_{k|l}q_l\right)$$
$$= -\sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l}\log q_l - \sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l}\log t_{k|l}$$
$$= \sum_{l=1}^{v} q_l\log q_l - \sum_{l=1}^{v}\sum_{k=1}^{n} r_{k,l}\log t_{k|l} = H\left(\underline{\eta}\right) + H\left(\underline{\xi}\middle|\underline{\eta}\right),$$

és a $H\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\xi}\right) + H\left(\underline{\eta}\middle|\underline{\xi}\right)$ összefüggést hasonlóan kapjuk. $H\left(\underline{\xi}\middle|\underline{\eta}\right) \le H\left(\underline{\xi}\right)$, ezért $H\left(\underline{\xi},\underline{\eta}\right) \le H\left(\underline{\xi}\right) + H\left(\underline{\eta}\right)$, és $H\left(\underline{\xi}\middle|\underline{\eta}\right) = H\left(\underline{\xi}\right)$ akkor és csak akkor, ha $\underline{\xi}$ és $\underline{\eta}$ függetlenek, így igaz a tétel utolsó állítása is.

□

A $\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(w)}$ valószínűségi változók együttes entrópiája és az $\underline{\eta}^{(1)},\cdots,\underline{\eta}^{(z)}$ valószínűségi változókra vonatkozó feltételes entrópiája hasonlóan definiálható, és indukcióval könnyen belátható, hogy bármely $1 \le l \le w$-re

$$H\left(\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(w)}\right) = H\left(\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(l)}\right) + \sum_{i=l+1}^{w} H\left(\underline{\xi}^{(i)}\middle|\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(i-1)}\right)$$

és

$$H\left(\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(w)}\middle|\underline{\eta}^{(1)},\cdots,\underline{\eta}^{(z)}\right)$$
$$= H\left(\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(l)}\middle|\underline{\eta}^{(1)},\cdots,\underline{\eta}^{(z)}\right) + \sum_{i=l+1}^{w} H\left(\underline{\xi}^{(i)}\middle|\underline{\eta}^{(1)},\cdots,\underline{\eta}^{(z)},\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(i-1)}\right),$$

vagyis például $l = 1$ esetén

$$H\left(\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(w)}\right) = H\left(\underline{\xi}^{(1)}\right) + \sum_{i=2}^{w} H\left(\underline{\xi}^{(i)}\middle|\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(i-1)}\right)$$

valamint

$$H\left(\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(w)}\middle|\underline{\eta}^{(1)},\cdots,\underline{\eta}^{(z)}\right) = H\left(\underline{\xi}^{(1)}\middle|\underline{\eta}^{(1)},\cdots,\underline{\eta}^{(z)}\right) + \sum_{i=2}^{w} H\left(\underline{\xi}^{(i)}\middle|\underline{\eta}^{(1)},\cdots,\underline{\eta}^{(z)},\underline{\xi}^{(1)},\cdots,\underline{\xi}^{(i-1)}\right).$$

A $H\left(\underline{\xi}\right) \ge H\left(\underline{\xi}\middle|\underline{\eta}\right)$ egyenlőtlenség azt fejezi ki, hogy ha $\underline{\xi}$-ről már van valamilyen előzetes ismeretünk, akkor legfeljebb annyi új információhoz jutunk, mint az előbbi ismeretek nélkül. $H\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\xi}\right) + H\left(\underline{\eta}\middle|\underline{\xi}\right)$ viszont azt jelenti, hogy az együttes eloszlás átlagos információtartalmát például úgy kapjuk meg, hogy meghatározzuk önmagában a $\underline{\xi}$ információtartalmát, és ehhez még hozzávesszük azt az információmennyiséget, amelyet már a $\underline{\xi}$ ismeretében $\underline{\eta}$-ról nyerhetünk.

Most egy fontos fogalmat definiálunk.

3.14. Definíció

Legyen $\underline{\xi}$ és $\underline{\eta}$ valószínűségi változó. Ekkor $I\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\xi}\right) - H\left(\underline{\xi}\middle|\underline{\eta}\right)$ a $\underline{\xi}$**és** $\underline{\eta}$ **kölcsönös információja**.

Δ

3.15. Tétel

$0 \leq I\left(\underline{\xi},\underline{\eta}\right) \leq \min\left\{H\left(\underline{\xi}\right), H\left(\underline{\eta}\right)\right\}$. $I\left(\underline{\xi},\underline{\eta}\right) = 0$ akkor és csak akkor, ha $\underline{\xi}$ és $\underline{\eta}$ függetlenek, és $I\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\xi}\right)$, ha $\underline{\xi}$ 1 valószínűséggel az $\underline{\eta}$ függvénye.

Δ

Bizonyítás:

$H\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\xi}\right) + H\left(\underline{\eta}\middle|\underline{\xi}\right) = H\left(\underline{\eta}\right) + H\left(\underline{\xi}\middle|\underline{\eta}\right)$-ból $H\left(\underline{\xi}\right) - H\left(\underline{\xi}\middle|\underline{\eta}\right) = H\left(\underline{\eta}\right) - H\left(\underline{\eta}\middle|\underline{\xi}\right)$, így az is igaz, hogy így $I\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\eta}\right) - H\left(\underline{\eta}\middle|\underline{\xi}\right)$. A tétel állításai ezek után következnek a $H\left(\underline{\xi}\right)$ és $H\left(\underline{\xi}\middle|\underline{\eta}\right)$ közötti (illetve a $H\left(\underline{\eta}\right)$ és $H\left(\underline{\eta}\middle|\underline{\xi}\right)$ közötti hasonló) összefüggésekből.

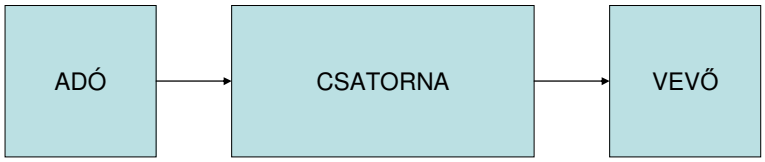
□

Szintén $H\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\xi}\right) + H\left(\underline{\eta}\middle|\underline{\xi}\right) = H\left(\underline{\eta}\right) + H\left(\underline{\xi}\middle|\underline{\eta}\right)$-ból látható, hogy a kölcsönös információ kifejezhető az $I\left(\underline{\xi},\underline{\eta}\right) = H\left(\underline{\xi}\right) + H\left(\underline{\eta}\right) - H\left(\underline{\xi},\underline{\eta}\right)$ alakban is, amiből látszik, hogy a kölcsönös információ szimmetrikus a két változójában.

A kölcsönös információ azt fejezi ki, hogy $\underline{\eta}$-t megfigyelve még mennyi bizonytalanság marad $\underline{\xi}$ vonatkozásában (illetve fordítva).

4. A kódolás valószínűségi alapjai

A **kommunikációs modell**t mutatja tömören az alábbi 1. ábra.



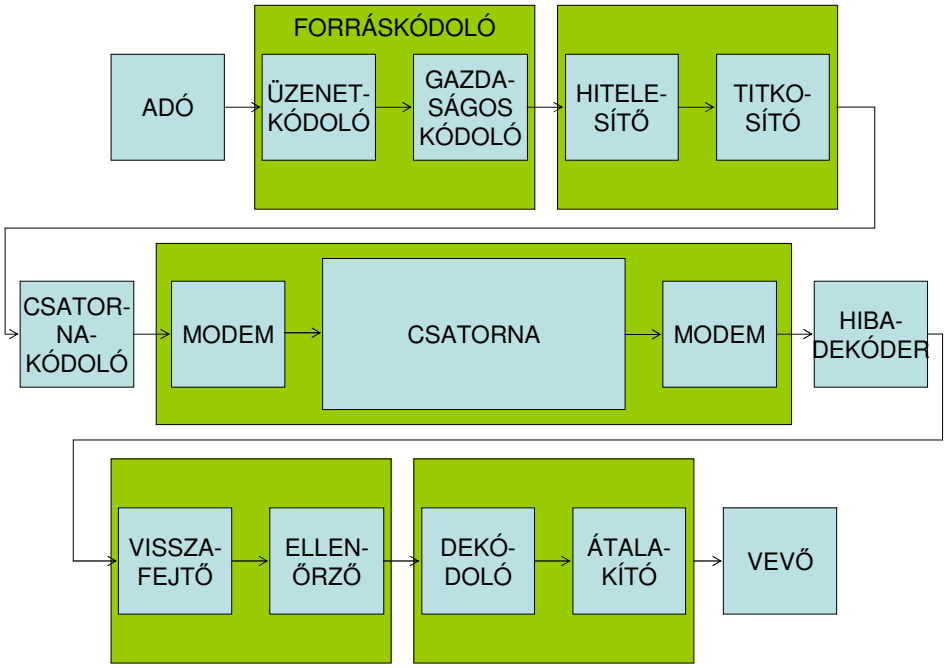
1. ábra

A kommunikáció során információt viszünk át az adótól a vevőhöz. Az információt adatok hordozzák, így valójában a csatornán az adatokat továbbítjuk a bemenettől a kimenet felé. Az információ átvitele térben és időben történik, bár közülük az egyik rendszerint domináns. Mivel a hibakorlátozás szempontjából elvileg közömbös, hogy adattárolásról vagy adatátvitelről van szó, ezért bármelyikről is szólunk, az a másikra is érvényes.

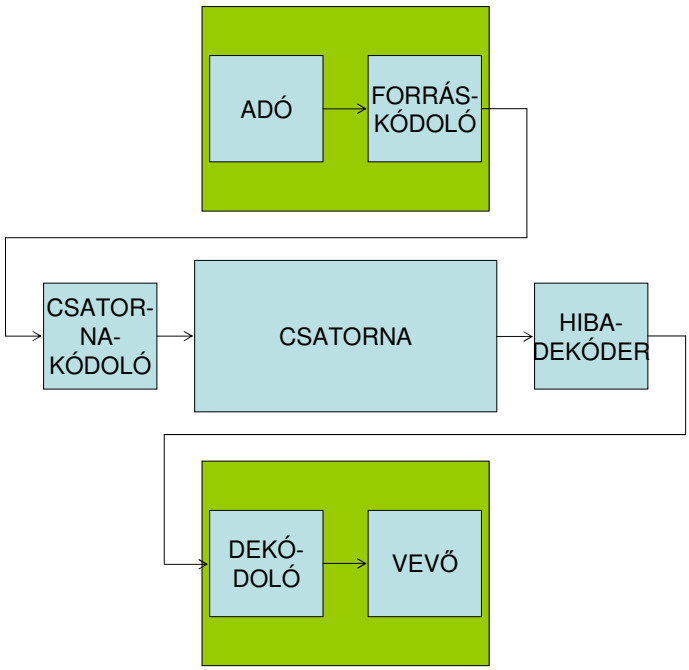
Az átvitel során négy probléma merül fel:

- a műszaki megvalósítás
- gazdaságossági kérdések
- az átvitel során fellépő hibák korlátozása
- titkosság - sértetlenség - hitelesség.

A fenti négy feladat megoldása külön - külön történik (bár nem biztos, hogy így a legjobb, de így lehet könnyen megvalósítani), amint a 2. ábra mutatja. Az előbbi modellt tömörebben, a hibakorlátozás feladatát kiemelve a 3. ábra mutatja.



2. ábra



3. ábra

A problémák megoldásához az adatokat kódolni kell. A **kódolás** során az üzeneteknek az átvitelre alkalmasabb jelsorozatot, adatot feleltetünk meg. A kódolástól elvárjuk, hogy injektív legyen, különben nem lenne lehetséges a **dekódolás**. Ha az átvitel során megengedünk egy adott nagyságú hibát, akkor az injektivitásnál gyengébb feltétel is elegendő.

A **hibakorlátozó kód**ok szinte mindig egyenletesek, azaz állandó hosszúságúak a kódszavak (ha nem így lenne, és hiba van az átvitel során, akkor esetleg már a kódszóhatárok sem ismerhetőek fel). Feltesszük, hogy nincs **szinkronhiba**, azaz a csatorna kimenetén ugyanannyi szimbólum detektálható, mint amennyit a bemenetén beadtunk (ez nem mindig igaz, és vannak olyan kódok, amelyek képesek detektálni, és esetleg – legalábbis részben – javítani a szinkronhibát).

Legyen I a csatorna **bemeneti ábécé**je (a továbbiakban mindig feltesszük, hogy a csatorna **diszkrét**), és legyen O a **kimeneti ábécé**. Feltesszük (mert feltehetjük), hogy $I \subseteq O$ (mert ha nem így lenne, akkor tekinthetnénk $O' = I \cup O$-t). Ha adott a kódszavak hossza, n, akkor a kódszavak halmaza, C, I^n egy részhalmaza. A csatorna kimenetén nem pontosan ugyanazt a jelsorozatot kapjuk, mint amit a bemenetére adtunk, és az eltérés általában nem determinisztikus, így a kimeneti és bementi jelsorozatok kapcsolatát alkalmas valószínűségi eloszlásokkal adhatjuk meg. A különböző $l \in \mathbb{N}^+$-okhoz tartozó $P(\eta_1 = v_1, \cdots, \eta_l = v_l | \xi_1 = u_1, \cdots, \xi_l = u_l)$ feltételes valószínűségek, ahol $u_i \in I$ és $v_i \in O$ meghatározzák a csatornát. Ha mindig teljesül a

$$P(\eta_1 = v_1, \cdots, \eta_l = v_l | \xi_1 = u_1, \cdots, \xi_l = u_l) = \prod_{i=1}^{l} P(\eta_i = v_i | \xi_i = u_i)$$

feltétel, ahol $u_j \in I$ és $v_i \in O$, akkor a csatorna **emlékezet nélküli**, és ilyenkor az $|I| = q \geq j \in \mathbb{N}^+$ és $|O| = q' \geq i \in \mathbb{N}^+$ indexekkel $C_{i,j} = P(v_i | u_j)$ a **csatornamátrix**.

Legyen $\mathbf{u} \in C$. Ha $\mathbf{u}$-t elküldjük, akkor a **csatornazaj**ok következtében egy $\mathbf{u}$-tól különböző $\mathbf{v} \in O^n$ érkezik a kimenetre. Ekkor dönteni kell, hogy mi lehetett az eredeti üzenet. Ez egy **döntési függvény**, vagy **döntési séma**, egy $f: O^n \to C \cup \{*\}$ leképezés, ahol $* \notin C$. Ha valamely $\mathbf{v} \in O^n$-re $f(\mathbf{v}) = \mathbf{u} \in C$, akkor ez azt jelenti, hogy úgy véljük, $\mathbf{u}$ volt az elküldött üzenet. Ha viszont $f(\mathbf{v}) = *$,

akkor csak annyit teszünk, hogy jelezzük, valami hiba történt az átvitel során, de nem tudjuk (vagy nem akarjuk) eldönteni, hogy mi volt az eredeti üzenet. Egy $\mathbf{v} \in O^n$ vételénél nyilván akkor és csak akkor helyes a döntésünk, ha valóban $f(\mathbf{v}) = \mathbf{u}$-t küldték, különben **döntési hiba** keletkezik, vagyis a helyes döntés valószínűsége $P(\boldsymbol{\xi} = f(\mathbf{v})|\boldsymbol{\eta} = \mathbf{v})$, és $P(hiba|\boldsymbol{\eta} = \mathbf{v}) = 1 - P(\boldsymbol{\xi} = f(\mathbf{v})|\boldsymbol{\eta} = \mathbf{v})$ a döntési hiba. Ennek átlaga $P(hiba) = 1 - \sum_{\mathbf{v}\in O^n} P(\boldsymbol{\xi} = f(\mathbf{v})|\boldsymbol{\eta} = \mathbf{v})P(\boldsymbol{\eta} = \mathbf{v})$, és ennek az átlagnak kellene a lehető legkisebbnek lennie a döntési séma függvényében. Az átlagos döntési hiba akkor minimális, ha $\sum_{\mathbf{v}\in O^n} P(\boldsymbol{\xi} = f(\mathbf{v})|\boldsymbol{\eta} = \mathbf{v})P(\boldsymbol{\eta} = \mathbf{v})$ maximális. Az összeg minden tagja nemnegatív, így akkor maximális, ha külön-külön minden tagja maximális. Mivel $P(\boldsymbol{\eta} = \mathbf{v}) \geq 0$, ezért ismét akkor kapjuk a maximumot, $P(\boldsymbol{\xi} = f(\mathbf{v})|\boldsymbol{\eta} = \mathbf{v})$ maximális (bár $P(\boldsymbol{\eta} = \mathbf{v}) = 0$ esetén közömbös az előbbi érték). Ez azt jelenti, hogy úgy kell az $f(\mathbf{v})$ értéket megválasztani, hogy a feltételes valószínűség maximális legyen, tehát legyen $P(\boldsymbol{\xi} = f(\mathbf{v})|\boldsymbol{\eta} = \mathbf{v}) = \max\{P(\boldsymbol{\xi} = \mathbf{u}|\boldsymbol{\eta} = \mathbf{v})|\mathbf{u} \in C\}$. Ehhez ismerni kellene $P(\boldsymbol{\xi} = \mathbf{u}|\boldsymbol{\eta} = \mathbf{v})$-t, ám nem ez, hanem a $P(\boldsymbol{\eta} = \mathbf{v}|\boldsymbol{\xi} = \mathbf{u})$ valószínűség adott. $P(\boldsymbol{\xi} = \mathbf{u}|\boldsymbol{\eta} = \mathbf{v}) = \frac{P(\boldsymbol{\eta}=\mathbf{v}|\boldsymbol{\xi}=\mathbf{u})P(\boldsymbol{\xi}=\mathbf{u})}{P(\boldsymbol{\eta}=\mathbf{v})}$, ahol $\mathbf{v}$, tehát a $P(\boldsymbol{\eta} = \mathbf{v})$ valószínűség rögzített, és így $P(\boldsymbol{\xi} = \mathbf{u}|\boldsymbol{\eta} = \mathbf{v})$ maximuma ismét függ a $P(\boldsymbol{\xi} = \mathbf{u})$ bemeneti eloszlástól.

Ha a döntési függvényt a $P(\boldsymbol{\xi} = f(\mathbf{v})|\boldsymbol{\eta} = \mathbf{v}) = \max\{P(\boldsymbol{\xi} = \mathbf{u}|\boldsymbol{\eta} = \mathbf{v})|\mathbf{u} \in C\}$ feltétellel határozzuk meg, akkor ezt a döntési függvényt **ideális megfigyelő**nek nevezzük. Az előbbiek alapján az ideális megfigyelő esetén a döntési hiba várható értéke minimális.

Most éljünk azzal a megkötéssel, hogy a bemeneti eloszlás egyenletes. Ha $|C| = M$, akkor az előbbi megkötéssel minden $\mathbf{u} \in C$ kódszó esetén $P(\boldsymbol{\xi} = \mathbf{u}) = \frac{1}{M}$, és

$$P(\boldsymbol{\xi} = \mathbf{u}|\boldsymbol{\eta} = \mathbf{v}) = \frac{P(\boldsymbol{\eta} = \mathbf{v}|\boldsymbol{\xi} = \mathbf{u})P(\boldsymbol{\xi} = \mathbf{u})}{P(\boldsymbol{\eta} = \mathbf{v})} = \frac{1}{MP(\boldsymbol{\eta} = \mathbf{v})}P(\boldsymbol{\eta} = \mathbf{v}|\boldsymbol{\xi} = \mathbf{u}).$$

Itt $\frac{1}{MP(\boldsymbol{\eta}=\mathbf{v})}$ független $\mathbf{u}$-tól, így

$$\max\{P(\boldsymbol{\xi} = \mathbf{u}|\boldsymbol{\eta} = \mathbf{v})|\mathbf{u} \in C\} = \frac{1}{MP(\boldsymbol{\eta} = \mathbf{v})}\max\{P(\boldsymbol{\eta} = \mathbf{v}|\boldsymbol{\xi} = \mathbf{u})|\mathbf{u} \in C\},$$

tehát $f(\mathbf{v}) = \hat{\mathbf{u}}$, ahol $P(\boldsymbol{\eta} = \mathbf{v}|\boldsymbol{\xi} = \hat{\mathbf{u}},)\max\{P(\boldsymbol{\eta} = \mathbf{v}|\boldsymbol{\xi} = \mathbf{u})|\mathbf{u} \in C\}$. Az így meghatározott döntési függvény a **maximum likelihood döntési séma**.

Nézzünk egy példát. Legyen $I = \{0,1\} = O$, a csatorna emlékezet nélküli, és a csatornamátrix legyen $\mathbf{C} = \begin{pmatrix} 1-p & p \\ p & 1-p \end{pmatrix}$, ahol p egy 1-nél nem nagyobb nemnegatív valós szám. Ez a **bináris szimmetrikus csatorna**, rövidítve a **BSC** (**B***inary* **Symmetric C***hannel*). Legyen továbbá a bemeneti eloszlás $P(\xi = 0) = q$ és $P(\xi = 1) = 1 - q$, ahol q is 1-nél nem nagyobb nemnegatív valós szám. Ekkor

$$P(\eta = 0) = P(\eta = 0|\xi = 0)P(\xi = 0) + P(\eta = 0|\xi = 1)P(\xi = 1) = (1-p)q + p(1-q)$$
$$P(\eta = 1) = P(\eta = 1|\xi = 0)P(\xi = 0) + P(\eta = 1|\xi = 1)P(\xi = 1) = pq + (1-p)(1-q)$$

$$P(\xi = 0|\eta = 0) = \frac{P(\eta = 0|\xi = 0)P(\xi = 0)}{P(\eta = 0)} = \frac{(1-p)q}{(1-p)q + p(1-q)}$$
$$P(\xi = 1|\eta = 0) = \frac{P(\eta = 0|\xi = 1)P(\xi = 1)}{P(\eta = 0)} = \frac{p(1-q)}{(1-p)q + p(1-q)}$$

és így $f(0) = \begin{cases} 0, \text{ha } q > p \\ 1, \text{ha } q < p, \end{cases}$ (és ha $p = q$, akkor mindegy),

$$P(\xi=0|\eta=1)=\frac{P(\eta=1|\xi=0)P(\xi=0)}{P(\eta=1)}=\frac{pq}{pq+(1-p)(1-q)}$$
$$P(\xi=1|\eta=1)=\frac{P(\eta=1|\xi=1)P(\xi=1)}{P(\eta=1)}=\frac{(1-p)(1-q)}{pq+(1-p)(1-q)},$$

vagyis most $f(1)=\begin{cases}0,\text{ha } p>1-q\\1,\text{ha } p<1-q,\end{cases}$ (és ismét mindegy, ha $p=1-q$). Az így meghatározott f függvény az ideális megfigyelő. Ha feltesszük, hogy $q=\frac{1}{2}$ és $p<\frac{1}{2}$, akkor $f(0)=0$ és $f(1)=1$. Ugyanezt kapjuk a $q=\frac{1}{2}$ feltétel esetén a $\max\{P(\eta=v|\xi=u)\big|u\in\{0,1\}\}$ feltételből is:

$$P(\eta=0|\xi=0)=1-p\wedge P(\eta=0|\xi=1)=p\wedge p<\frac{1}{2}<1-p\Rightarrow f(0)=0$$
$$P(\eta=1|\xi=0)=p\wedge P(\eta=1|\xi=1)=1-p\wedge p<\frac{1}{2}<1-p\Rightarrow f(1)=1.$$

Az utóbbi döntési függvény a maximum likelihood döntési függvény. Ekkor a döntési hiba

$$P^{(1)}(hiba)=P(\eta=1|\xi=0)P(\xi=0)+P(\eta=0|\xi=1)P(\xi=1)=pq+p(1-q)=p.$$

Most legyen a csatorna az előbbi, és $n=3$, továbbá $C=\{000,111\}$. Legyen a döntési függvény $f(\mathbf{v})=\left\lfloor\frac{v_1+v_2+v_3}{2}\right\rfloor$, vagyis legyen $f(000)=f(001)=f(010)=f(100)=0$, és hasonlóképpen legyen $f(111)=f(110)=f(101)=f(011)=1$. Ekkor a döntési hiba

$$P^{(3)}(hiba)=\binom{3}{2}p^2(1-p)+\binom{3}{3}p^3,$$

hiszen akkor döntünk rosszul, ha az átvitel során három összetartozó bitből legalább kettő meghibásodik. Összehasonlítva ezt az előbbi esettel, $\frac{P^{(3)}(hiba)}{P^{(1)}(hiba)}=3p\left(1-\frac{2}{3}p\right)$. A függvény maximuma $p=\frac{3}{4}$-nél van, így $p<\frac{1}{2}$ esetén az arány kisebb, mint $p=\frac{1}{2}$-nél, ahol az értéke 1, vagyis $p<\frac{1}{2}$ esetén a háromszorozásnál a döntési hiba kisebb lesz. Ha például $p=10^{-3}$ (ez egy elég zajos csatorna), akkor az előbbi arány $\frac{P^{(3)}(hiba)}{P^{(1)}(hiba)}=3\cdot10^{-3}\left(1-\frac{2}{3}\cdot10^{-3}\right)\approx3\cdot10^{-3}\approx\frac{1}{333}$, vagyis a javulás kb. 333-szoros.

Most legyen $|I|=q(\geq2)$, $O=I$, és a csatornamátrix $C_{i,i}=1-p$, és $i\neq j$-re $C_{i,j}=\frac{p}{q-1}$ (ez a csatorna az **emlékezet nélküli diszkrét szimmetrikus csatorna**, **az MDSC**, azaz a **M***emoryless* **D***iscrete* **S***ymmetric* **C***hannel*). Ekkor

$$\begin{aligned}P(\boldsymbol{\eta}=\mathbf{v}|\boldsymbol{\xi}=\mathbf{u})&=\prod_{i=1}^{s}P(\eta_i=v_i|\xi_i=u_i)\\&=\prod_{u_i\neq v_i}P(\eta_i=v_i|\xi_i=u_i)\cdot\prod_{u_i\neq=}P(\eta_i=v_i|\xi_i=u_i)\\&=\prod_{u_i\neq v_i}\frac{p}{q-1}\cdot\prod_{u_i\neq=}(1-p)\\&=\left(\frac{p}{q-1}\right)^d(1-p)^{n-d}=(1-p)^n\left(\frac{\frac{p}{q-1}}{1-p}\right)^d,\end{aligned}$$

ahol n a kódszavak hossza, és $d=d(\mathbf{u},\mathbf{v})$ az $\mathbf{u}$ és $\mathbf{v}$ azonos pozícióban lévő, eltérő komponenseinek száma, az $\mathbf{u}$ és $\mathbf{v}$ Hamming-távolsága. Ha $\frac{p}{q-1}<1-p$, azaz, ha $0<p<1-\frac{1}{q}$, akkor a fenti valószí-

nűség akkor maximális, ha $d(\mathbf{u},\mathbf{v})$ minimális, vagyis ha $d(f(\mathbf{v}),\mathbf{v}) = \min\{d(\mathbf{u},\mathbf{v})|\mathbf{u} \in C\}$. Az így meghatározott döntési függvény a **minimális távolságú dekódolás** (lásd 9. oldalon a 2.8. definíciót). MDSC esetén tehát a minimális távolságú dekódolás a maximum likelihood döntési függvény.

Visszatérve a háromszoros ismétléshez, nyilván három helyett az ismétlések száma bármely $n \in \mathbb{N}^+$-ra $2n+1$ is lehet. Annak a valószínűsége, hogy egy $2n+1 \geq k \in \mathbb{N}$-re az átvitel során k hiba lép fel, $P(\kappa = k) = \binom{2n+1}{k} p^k (1-p)^{(2n+1)-k}$. Ez binomiális eloszlás, és így a hibák számának várható értéke $E(\kappa) = (2n+1)p$. Ha $p < \frac{1}{2}$, akkor $E(\kappa) = (2n+1)p < n + \frac{1}{2} < n+1$, a hibás jegyek várható száma kisebb, mint a nem hibás jegyek várható száma, a döntés várhatóan helyes lesz. Mi ennek az ára? Ehhez bevezetjük az alábbi fogalmat.

4.1. Definíció

Legyen C egy $(n, M)_q$-paraméterű kód. Ekkor $\mathcal{R}_q = \frac{1}{n}\log_q M$ a **kódsebesség**.

Δ

$\log_q M$ azt adja meg, hogy mekkora minimális szóhosszal lehet q különböző szimbólum felhasználásával M különböző szót megadni, így a kódsebesség azt mutatja, hogy a minimálisan szükséges hosszhoz képest mekkora a kódszavak hossza. Nyilván igaz, hogy $0 \leq \mathcal{R}_q \leq 1$, feltéve, hogy a kód legalább egy szót tartalmaz. A fentebb ismertetett ismétléses kód esetén az volt a helyzet, hogy az ismétlések számának növelésével a döntési hiba nullához tartott, de ezzel együtt a kódsebesség is tart a nullához, vagyis végül hibátlanul visszük át a semmit. Általában, ha n tart a végtelenhez, miközben M nem változik, vagyis ugyanannyiféle üzenetet egyre hosszabb kódokkal akarunk átvinni a csatornán, akkor a kódsebesség, $\mathcal{R}_q$, tart a nullához, vagyis hiába tart a döntési hiba a nullához, ám az átvitt üzenetek száma is tart a nullához. **Shannon** szerint ez nem szükségszerű.

A **csatorna kapacitása** nem más, mint a csatorna bemenetén és kimenetén lévő valószínűségi mező kölcsönös információjának maximuma a bemenetére adott eloszlás függvényében, vagyis matematikailag felírva $\mathcal{C} = \max\{I(X,Y)|P(X)\}$. Itt arról van szó, hogy a csatorna kimenetén megjelenő jelsorozat a bemenetre adott jelsorozattól és a csatornától függ. Minél erősebb az eltérés, annál kevésbé lehet rekonstruálni a kimeneten megfigyelt jelsorozatból az elküldött kódszót. Az azonban, hogy milyen mértékű a torzulás, attól is függ, hogy hogyan választjuk meg a csatorna bemenetére adott jelsorozatokat. Ezt fejezi ki a csatornakapacitás.

Shannon azt állítja, hogy bizonyos csatornák esetén a csatorna kapacitásánál kisebb bármely sebességgel tetszőleges kis hibavalószínűséggel átvihető az információ, vagyis lehet olyan kódot konstruálni, amellyel az előbb említett módon lehet kommunikálni (a *kisebb* megkötés akkor igaz, ha a kölcsönös információ mérésénél is az alkalmazott szimbólumok száma a logaritmus alapja, különben egy megfelelő, az alkalmazott alaptól függő konstanssal való szorzás után igaz a *kisebb* feltétel). Ismeretes a tétel két megfordítása is. A *gyenge megfordítás* szerint a csatorna kapacitását meghaladó sebesség esetén a döntési hiba akármilyen kódolás esetén is nagyobb lesz egy pozitív korlátnál, míg az *erős megfordítás* szerint a kód hosszának növekedésével a döntési hiba valószínűsége 1-hez tart (a gyenge megfordítás nem következik az erős megfordításból, a két tétel önálló, továbbá most is igaz a logaritmus alapjára vonatkozó korábbi megjegyzés).

Bizonyítás nélkül a tételek az alábbiak, feltéve, hogy q szimbólummal kódolunk.

1. **A zajos csatorna kódolási tétele**. Ha $\mathcal{C} > \mathcal{R}_q \in \mathbb{R}^+$, akkor van olyan C_n: $\left(n, \left\lceil q^{\mathcal{R}_q n}\right\rceil\right)_q$ kódsorozat és döntési függvények olyan sorozata, hogy $P_n^{(max)}(hiba) \to 0$, ha $n \to \infty$.

Ezt a tételt néhány csatornatípusra, többek között az emlékezet nélküli csatornákra igazolták.

2. **Gyenge megfordítás**. $\mathcal{C} < \mathcal{R}_q \in \mathbb{R}^+$ esetén bármely $C_n: \left(n, \left\lceil q^{\mathcal{R}_q n} \right\rceil\right)_q$ kódsorozat és tetszőleges döntési függvény mellett van olyan $\varepsilon \in \mathbb{R}^+$, hogy minden $n \in \mathbb{N}^+$-re $P_n(hiba) > \varepsilon$.

3. **Erős megfordítás**. Ha $\mathcal{C} < \mathcal{R}_q \in \mathbb{R}^+$, akkor bármely $C_n: \left(n, \left\lceil q^{\mathcal{R}_q n} \right\rceil\right)_q$ kódsorozat, és a döntési függvények tetszőleges sorozata esetén $\lim_{n\to\infty} P_n(hiba) = 1$.

A két megfordítási tétel lényege, hogy a csatornakapacitásnál nagyobb sebességgel nem lehet hibátlanul dekódolni, sőt, minél hosszabbak a kódszavak, annál biztosabb, hogy hibázunk a dekódolásnál.

Shannon tétele elvi jelentőségű. A tétel szerint létező kódot még senkinek nem sikerült konstruálnia, ami nem okoz túl nagy problémát, ugyanis egy ilyen kód hossza és mérete használhatatlanul nagy lenne. A tétel jelentősége, hogy megmutatja, hogyan lehet egyszerre csökkenteni a dekódolási hibát a kódsebesség lényeges csökkenése nélkül: több üzenetet egyszerre kell, nagyobb hosszúságú kódszavakba kódolni, így nő a kód mérete is, ellensúlyozva a kódsebesség csökkenését.

Ha a csatorna MDSC, és a bemeneti eloszlás egyenletes, akkor a hibák várható száma n-hosszúságú kódszóban np, ahol p annak a valószínűsége, hogy a kimeneten megjelenő szimbólum különbözik a bemenetre adott jeltől, vagyis annak, hogy egy jel az átvitel során megváltozott. Minimális távolságú dekódolásnál a javítható hibák száma a 2.10. Tétel szerint arányos a kód távolságával, így ahhoz, hogy a várhatóan fellépő valamennyi hibát ki tudjuk javítani minimális távolságú dekódolással, szükséges, hogy d arányosan nőjön n-nel (mint láttuk, nagy kódsebesség és kis döntési hiba nagy kódszóhosszúsággal érhető el), vagyis biztosítani kell, hogy $np \sim d$ teljesüljön, azaz hogy $\delta = \frac{d}{n} \sim p$ legyen. Sajnos a legtöbb kódcsalád nem teljesíti ezt a feltételt, általában $\delta \to 0$, ha $n \to \infty$.

5. Lineáris kódok

Egy test elemeiből képezett rendezett n-esek a komponensenkénti összeadással és a komponenseknek a test egy adott elemével való szorzásával a test fölötti n-dimenziós teret alkotnak. Ebben a térben bázist alkotnak például azok a vektorok, az egységvektorok, amelyeknek egy és csak egy komponensük különbözik 0-tól, és ez a komponensük a test egységeleme.

5.1. Jelölés

Legyen $n \in \mathbb{N}^+$. $\mathbb{F}_q^n$-t mint $\mathbb{F}_q$ fölötti n-dimenziós lineáris teret $V_q^{(n)}$, ennek elemeit és a komponensekből álló oszlopvektort $\mathbf{u}$, a megfelelő sorvektort $\mathbf{u}^T$ jelöli. Ha $n \geq k \in \mathbb{N}$, akkor $W_q^{(n,k)}$ a $V_q^{(n)}$ (egy) k-dimenziós altere.

Δ

5.2. Definíció

Ha $\mathcal{S}$ Abel-csoport, és $\mathcal{C} \leq \mathcal{S}^n$ a komponensenkénti $\mathcal{S}$-beli művelettel, akkor $\mathcal{C}$ **csoportkód**. Ha $\mathcal{S}^n$ egyben egy test feletti vektortér, és $\mathcal{C}$ ennek k-dimenziós altere, akkor a kód **lineáris**. Az előbbi lineáris kód jele $[n, k, d]_q$, ahol d és q egymástól függetlenül elhagyható.

Δ

A $\mathcal{C}$ kód nyilván pontosan akkor csoportkód, ha $\mathcal{S}$ additív Abel-csoport, és $C - C \subseteq C$, továbbá a korábbi és a mostani definíciókból látható, hogy egy $[n, k]_q$ kódban $M = q^k$.

5.3. Tétel

Ha $n \in \mathbb{N}^+$, és $C \subseteq V_q^{(n)}$ legalább 1-dimenziós altér, akkor $d(\mathcal{C}) = w(\mathcal{C})$.

Δ

Bizonyítás:

$k \geq 1$ és $q \geq 2$, így $|C| = q^k \geq q \geq 2$, ahol k az altér dimenziója. Lineáris tér additív Abel-csoport az összeadással, és altér zárt a kivonásra, ezért $\mathcal{S} = \mathbb{F}_q$-val $\mathbb{F}_q^n$ és $\mathcal{C}$ megfelel a 2.1. Definíció és 2.2. Tétel előírásainak.

□

5.4. Definíció

Legyen $n \in \mathbb{N}^+$, $\mathbf{a} \in V_q^{(n)}$ és $\mathbf{b} \in V_q^{(n)}$. $(\mathbf{a}, \mathbf{b}) = \mathbf{a}^T\mathbf{b} = \sum_{i=0}^{n-1} a_i b_i$ az $\mathbf{a}$ és $\mathbf{b}$ **skalárszorzata**, és $\mathbf{a}$, $\mathbf{b}$ **ortogonális**, ha $(\mathbf{a}, \mathbf{b}) = 0$. $W_q^{(n,k)^\perp} = \left\{\mathbf{x} \in V_q^{(n)} \middle| \forall \left(\mathbf{v} \in V_q^{(n)}\right) : (\mathbf{v}, \mathbf{x}) = 0\right\}$ a $W_q^{(n,k)}$ **ortogonális altere**.

Δ

$W_q^{(n,k)^\perp} \subseteq V_q^{(n)}$ a definíció következménye, és hogy ez altér, az könnyen belátható. Azt is egyszerű meggondolni, hogy $W_q^{(n,k)} \subseteq \left(W_q^{(n,k)^\perp}\right)^\perp$. Igazolható, hogy $\dim\left(W_q^{(n,k)^\perp}\right) = n - k$, így viszont az előbbi tartalmazással következik, hogy $\left(W_q^{(n,k)^\perp}\right)^\perp = W_q^{(n,k)}$, vagyis egy lineáris tér egy altere ortogonális alterének ortogonális altere az eredeti altér.

Egyszerű példa illusztrálja, hogy $W_q^{(n,k)}$ és $W_q^{(n,k)^\perp}$ metszete általában nem csupán a nullvektort tartalmazza. Ha a test karakterisztikája p, és $n \geq p$, akkor legyen $\mathbf{u} \in V_q^{(n)}$ olyan, hogy $p > i \in \mathbb{N}$-re $u_i = e$, míg a többi indexre $u_i = 0$. Ekkor $(\mathbf{u}, \mathbf{u}) = \sum_{i=0}^{p-1} u_i^2 = pe = 0$, vagyis $\mathbf{u}$ egy nullától különböző, önortogonális vektor, így eleme például az egyedül általa generált altérnek (vagyis a skalárszorosaiból álló altérnek), és ugyanakkor az előbbi altér ortogonális alterének is. (Valójában, ha q páros, vagy 4-gyel osztva a maradék 3, akkor már a kétdimenziós térben, míg ha q néggyel való osztási maradéka 1, akkor a háromdimenziós térben van nullától különböző önortogonális vektor). Egy másik példaként tekintsük $A = \{0000,1100,0011,1111\} \subseteq \mathbb{F}_2^4$-et. Bináris esetben a konstanssal való szorzás 0-val vagy 1-gyel való szorzást jelet, és ennek eredménye vagy a csupa 0, vagy az eredeti vektor, így a halmaz pontosan akkor altér, ha zárt az összeadásra, ami láthatóan teljesül. Az is könnyen megállapítható, hogy a halmaz bármely elemének a halmaz tetszőleges elemével való skalárszorzata 0, viszont tetszőleges, halmazon kívüli elemmel szorozva A egy elemét, a szorzat nem 0, így A mint $\mathbb{F}_2^4$ egy alterének ortogonális altere önmaga, $A^\perp = A$. Ez viszont azt jelenti, hogy a két altér összege is az altér, ami valódi altere a teljes térnek.

Az előbbi bekezdés alapján véges test feletti lineáris térben nem minden igaz, ami 0-karakterisztikájú (tehát például valós vagy komplex) lineáris terekben igaz.

5.5. Definíció

Legyen $n \in \mathbb{N}^+$, k az n-nél kisebb természetes szám, továbbá $W_q^{(n,k)}$ egy $[n,k]_q$ kód. A $W_q^{(n,k)}$ lineáris tér egy bázisának elemeiből mint sorvektorokból álló $\mathbf{G}$ mátrix **a kód generátormátrixa**, a $W_q^{(n,k)^\perp}$ ortogonális tér bázisvektoraihoz tartozó sorvektorokból mint sorokból álló $\mathbf{H}$ mátrix **a kód (paritás)ellenőrző mátrixa**.

Δ

5.6. Tétel

Egy $[n,k]_q$ paraméterű kód generátormátrixa egy $\mathbb{F}_q$ fölötti, $k \times n$ méretű, k-rangú mátrix, míg a kód ellenőrző mátrixa $\mathbb{F}_q$ fölötti, $(n-k) \times n$ méretű, $n-k$-rangú mátrix.

Az $\mathbb{F}_q$ fölötti $k \times n$ méretű, k-rangú $\mathbf{G}$ és $(n-k) \times n$ méretű, $n-k$-rangú $\mathbf{H}$ mátrix akkor és csak akkor generátor- és ellenőrző mátrixa ugyanazon $[n,k]_q$ paraméterű kódnak, ha $\mathbf{H}\mathbf{G}^T = \mathbf{0}$, továbbá $\mathbf{v} \in V_q^{(n)}$ akkor és csak akkor eleme a kódnak, ha $\mathbf{H}\mathbf{v} = \mathbf{0}$.

Δ

Bizonyítás:

$W_q^{(n,k)}$-nak mint az $\mathbb{F}_q$ fölötti n-komponensű vektorokból álló tér k-dimenziós alterének minden eleme szintén $\mathbb{F}_q$ fölötti n-komponensű vektor, a bázis k vektort tartalmaz, és ezek lineárisan függetlenek; $\mathbf{H}$ sorai $V_q^{(n)}$-beli vektorokhoz tartozó sorvektorok, így $\mathbb{F}_q$ fölötti n-esek, továbbá az ortogonális altér dimenziója $n-k$, tehát $\mathbf{H}$ $\mathbb{F}_q$ elemeiből álló $(n-k) \times n$-es mátrix, és sorai mint bázisvektorok lineárisan függetlenek, a rang $n-k$. Az előbbiek alapján kiadódik a $\mathbf{G}$ méretére, rangjára és elemeire vonatkozó állítás.

$\mathbf{G}$ sorai az altér elemei, $\mathbf{H}$ sorai az ortogonális altérhez tartoznak, így $\mathbf{H}$ bármely sorát a $\mathbf{G}$ tetszőleges sorával szorozva nullát kapunk, $\mathbf{H}$-nak és $\mathbf{G}$ transzponáltjának a szorzata nulla.

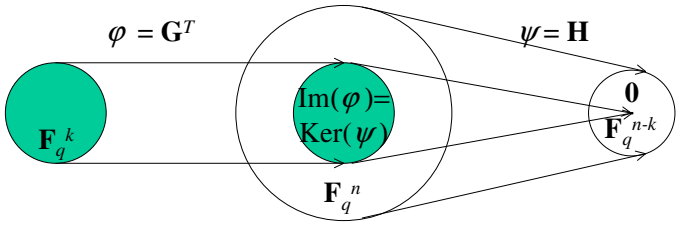
Fordítva, ha $\mathbf{G}$ sorai n-eleműek, akkor elemei a q-elemű test fölötti n-dimenziós térnek, és mivel a mátrix rangja azonos a sorok számával, a sorok lineárisan függetlenek, így az n-dimenziós tér egy k-dimenziós alterét feszítik ki. Hasonlóan adják $\mathbf{H}$ sorai a tér egy $n-k$-dimenziós alterének bázisát. Végül $\mathbf{H}\mathbf{G}^T = \mathbf{0}$-ból következik, hogy a két mátrix sorai, vagyis a két altér bázisának vektorai merőlegesek egymásra, amiből következik, hogy $\mathbf{G}$ sorainak bármely lineáris kombinációja merőleges $\mathbf{H}$ sorainak tetszőleges lineáris kombinációjára, tehát a $\mathbf{G}$ illetve $\mathbf{H}$ által meghatározott altér merőleges

egymásra, és a két dimenzió összege n, így a két altér egymás ortogonális komplementere, amiből következik, hogy $\mathbf{G}$ és $\mathbf{H}$ egy $[n,k]$ paraméterű kód generátor- és ellenőrző mátrixa.

Legyen $\mathbf{v} \in W_q^{(n,k)}$, akkor az ortogonális altér minden eleme, de így $\mathbf{H}$ minden sora merőleges $\mathbf{v}$-re, minden ilyen szorzat, és emiatt $\mathbf{Hv}$ is $\mathbf{0}$ lesz. Fordítva, ha $\mathbf{Hv}$ nulla, akkor $\mathbf{v}$ a $\mathbf{H}$ minden sorára mint $W_q^{(n,k)^\perp}$-beli vektorra ortogonális, de akkor $W_q^{(n,k)^\perp}$ minden vektorára is merőleges, hiszen ezek a mátrix sorainak lineáris kombinációi, $\mathbf{v}$ ortogonális a teljes $W_q^{(n,k)^\perp}$ altérre, azaz $\mathbf{v} \in W_q^{(n,k)}$.

□

A $\mathbf{G}$ illetve a $\mathbf{H}$ mátrix ismeretében egy $[n,k]_q$ kódot tekinthetünk a következő módon is. A kódolandó üzenetek az $\mathbb{F}_q$ fölötti k-dimenziós tér elemei, amelyeket a szintén $\mathbb{F}_q$ fölötti n-dimenziós tér elemeivel kódolunk, vagyis a kódolás egy $\varphi\colon \mathbb{F}_q^k \to \mathbb{F}_q^n$ leképezés, és a leképezést a $\varphi\colon \mathbf{u}^T \mapsto \mathbf{u}^T\mathbf{G}$ megfeleltetés adja, ahol $\mathbf{u}$ az $\mathbb{F}_q^k$ tetszőleges eleme. Ekkor $C = \mathrm{Im}(\varphi) = \mathrm{Im}(\mathbf{G})$ a kód. Ugyanakkor azt is mondhatjuk, hogy a kód az $\mathbb{F}_q$ fölötti n-dimenziós tér azon és csak azon elemeiből áll, amelyeket a $\psi\colon \mathbf{v} \mapsto \mathbf{Hv}$ megfeletetés az $n-k$-dimenziós tér nullelemére képez, vagyis amelyek benne vannak a leképezés magjában, tehát $C = \mathrm{Ker}(\psi) = \mathrm{Ker}(\mathbf{H})$. Az első leképezésnek injektívnek kell lennie, ugyanis csak ilyen leképezéssel kapunk kódolást, és ez teljesül, hiszen $\mathbf{G}$ sorai lineárisan függetlenek. Ugyanakkor a második leképezés szürjektíven képezi le $\mathbb{F}_q^n$-t $\mathbb{F}_q^{n-k}$-ra, mivel $\mathbf{H}$ sorai lineárisan függetlenek, így generálják az $n-k$-dimenziós teret. Az előbb mondottakat szemlélteti az alábbi 4. ábra.



4. ábra

Az ábrából leolvasható az az egyébként nyilvánvaló tény, hogy $\mathbb{F}_q^k \cong C \leq \mathbb{F}_q^n$, és hogy $\mathbf{HG}^T = \mathbf{0}$.

Felhívjuk a figyelmet rá, hogy egy kódnak több különböző generátormátrixa illetve ellenőrző mátrixa lehet, hiszen egy lineáris tér bázisa korántsem egyértelmű. Sőt, még az sem igaz, hogy egy adott generátormátrixhoz egy és csak egy ellenőrző mátrix tartozik, mivel a kód bármely generátormátrixára és tetszőleges ellenőrző mátrixára igaz a tétel. Végül a tételből az is kiolvasható, hogy egy kódot egyértelműen meghatározza valamely generátor- vagy ellenőrző mátrixa, és akkor ez a mátrix meghatározza a kód valamennyi generátor- és ellenőrző mátrixát. Kérdés, hogy hogyan lehet adott generátormátrixhoz meghatározni egy ellenőrző mátrixot, vagy fordítva.

5.7. Definíció

Az $[n,k]$-paraméterű kód generátormátrixa **standard alakú**, ha $\mathbf{G} = (\mathbf{I}_k \ \mathbf{P})$ vagy $\mathbf{G} = (\mathbf{P} \ \mathbf{I}_k)$ alakú. Hasonlóan definiáljuk a standard alakú ellenőrző mátrixot is.

Δ

5.8. Tétel

Ha $\mathbf{H} = (\mathbf{I}_{n-k} \ \mathbf{P})$ az $[n,k]$-paraméterű kód ellenőrző-mátrixa, akkor $\mathbf{G} = (-\mathbf{P}^T \ \mathbf{I}_k)$ a kód egy generátormátrixa. Fordítva, ha $\mathbf{G} = (\mathbf{I}_k \ \mathbf{P})$, akkor $\mathbf{H} = (-\mathbf{P}^T \ \mathbf{I}_{n-k})$ egy lehetséges ellenőrző mátrix.

Δ

Bizonyítás:
$(\mathbf{I}_{n-k}\ \mathbf{P})(-\mathbf{P}^T\ \mathbf{I}_k)^T = (\mathbf{I}_{n-k}\ \mathbf{P})\begin{pmatrix}-\mathbf{P}\\ \mathbf{I}_k\end{pmatrix} = -\mathbf{P}+\mathbf{P} = \mathbf{0}$. A másik állítás bizonyítása hasonló.

□

Az előző tétel alapján könnyen meghatározhatunk egy adott generátormátrixhoz egy ellenőrző mátrixot, vagy fordítva. Tegyük fel, hogy adott egy $[n,k]$ kód ellenőrző-mátrixa, $\mathbf{H}$. Mivel a mátrix sorainak száma $n-k$, és a rangja ugyanekkora, ezért van a mátrixban $n-k$ lineárisan független oszlop, és így egy $n-k$-méretű reguláris részmátrix. Most $\mathbf{H}$-t jobbról megszorozva egy alkalmas $\mathbf{\Pi}$ permutációs mátrixszal, a reguláris részmátrix a mátrix első $n-k$ oszlopába vihető, és ha ez a részmátrix $\mathbf{M}$, akkor $\mathbf{H'} = \mathbf{M}^{-1}\mathbf{H\Pi} = (\mathbf{I}_{n-k}\ \mathbf{P})$ lesz, és $\mathbf{G'} = (-\mathbf{P}^T\ \mathbf{I}_k)$ a $\mathbf{H'}$ által meghatározott kód egy generátormátrixa. Most legyen $\mathbf{G} = \mathbf{G'\Pi}^T$. Permutációs mátrix inverze egyenlő a mátrix transzponáltjával, így $\mathbf{G'} = \mathbf{G\Pi}$. Ekkor $\mathbf{0} = \mathbf{H'G'}^T = (\mathbf{M}^{-1}\mathbf{H\Pi})(\mathbf{G\Pi})^T = \mathbf{M}^{-1}\mathbf{H\Pi\Pi}^T\mathbf{G}^T = \mathbf{M}^{-1}(\mathbf{HG}^T)$, és mivel $\mathbf{M}^{-1}$ reguláris, ezért $\mathbf{HG}^T = \mathbf{0}$, vagyis $\mathbf{G}$ a kód generátormátrixa.

Az előzőek szerint egy ellenőrző mátrixból úgy tudunk meghatározni egy generátormátrixot, hogy a sorokon végzett reguláris műveletekkel (például Gauss-eliminációval), valamint az oszlopok sorrendjének változtatásával úgy alakítjuk át $\mathbf{H}$-t, hogy a bal szélén egységmátrix álljon. Ebből a mátrixból meghatározzuk $\mathbf{G'}$-t, és az így kapott mátrix oszlopait a $\mathbf{H}$ oszlopain végrehajtott cserékkel ellenkező sorrendben és irányban permutálva megkapjuk a keresett generátormátrixot.

Legyen például $\mathbb{F}_q = \mathbb{Z}_5$ és $\mathbf{H} = \begin{pmatrix}2&1&0&3&1&4&2\\1&3&0&1&0&2&0\\2&1&0&1&4&4&1\\3&3&1&4&2&1&3\end{pmatrix}$. Ebből a mátrixból a csak a sorokon végzett invertálható átalakításokkal kapjuk a

$$\mathbf{H''} = \mathbf{M}^{-1}\mathbf{H} = \begin{pmatrix}4&3&3&3\\3&3&2&4\\2&4&1&0\\2&2&2&0\end{pmatrix}\begin{pmatrix}2&1&0&3&1&4&2\\1&3&0&1&0&2&0\\2&1&0&1&4&4&1\\3&3&1&4&2&1&3\end{pmatrix} = \begin{pmatrix}1&0&3&0&2&2&0\\0&1&4&0&4&0&0\\0&0&0&1&1&0&0\\0&0&0&0&0&0&1\end{pmatrix}$$

mátrixot. Ez utóbbi mátrix oszlopait a $\pi = (3,5,6,7,4)$ ciklus szerint felcserélve a

$$\mathbf{H'} = \mathbf{H''\Pi} = \begin{pmatrix}1&0&3&0&2&2&0\\0&1&4&0&4&0&0\\0&0&0&1&1&0&0\\0&0&0&0&0&0&1\end{pmatrix}\begin{pmatrix}1&0&0&0&0&0&0\\0&1&0&0&0&0&0\\0&0&0&0&1&0&0\\0&0&1&0&0&0&0\\0&0&0&0&0&1&0\\0&0&0&0&0&0&1\\0&0&0&1&0&0&0\end{pmatrix}$$

$$= \begin{pmatrix}1&0&0&0&3&2&2\\0&1&0&0&4&4&0\\0&0&1&0&0&1&0\\0&0&0&1&0&0&0\end{pmatrix} = (\mathbf{I}_4\ \mathbf{P})$$

mátrixot kapjuk, ahol $\mathbf{P} = \begin{pmatrix}3&2&2\\4&4&0\\0&1&0\\0&0&0\end{pmatrix}$. Ebből

$$\mathbf{G'} = (-\mathbf{P}^T\ \mathbf{I}_3) = \begin{pmatrix}2&1&0&0&1&0&0\\3&1&4&0&0&1&0\\3&0&0&0&0&0&1\end{pmatrix},$$

és végül

$$\mathbf{G} = \mathbf{G}'\boldsymbol{\Pi}^T = \begin{pmatrix} 2 & 1 & 0 & 0 & 1 & 0 & 0 \\ 3 & 1 & 4 & 0 & 0 & 1 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$
$$= \begin{pmatrix} 2 & 1 & 1 & 0 & 0 & 0 & 0 \\ 3 & 1 & 0 & 4 & 1 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

5.9. Definíció

Az $(n, M)_q$-paraméterű kód **szisztematikus** vagy **szeparábilis a** $0 \leq i_1 < \cdots < i_l < n$ **indexekre**, ha $l = \lceil \log_q M \rceil$, és a kódban ezen az l pozíción álló l-esek páronként különbözőek. A kód **szisztematikus**, ha vagy az első, vagy az utolsó l pozícióra nézve szeparábilis.

Δ

Ha egy kód szisztematikus valamely pozíciókra, akkor az ezen pozíción álló l-esek tekinthetőek az üzenetnek, így hibajavítás után a dekódolás a többi pozíción lévő jegyek elhagyását jelenti.

Valamely indexekre szeparábilis kód ekvivalens egy szisztematikus kóddal, hiszen oszlopcserékkel a $0 \leq i_1 < \cdots < i_l < n$ indexű oszlopok átvihetőek a $0, 1, \cdots, l-1$ indexű oszlopokba.

5.10. Tétel

Minden lineáris kód ekvivalens egy szisztematikus kóddal.

Δ

Bizonyítás:

Tekintsük a kód $\mathbf{G}$ generátormátrixát. $\mathbf{G}$-nek k sora van, és a sorai lineárisan függetlenek, így van k lineárisan független oszlopa is. Ha ezek az oszlopok a $0 \leq i_0 < \cdots < i_{k-1} < n$ indexekhez tartoznak, akkor, miközben előállítjuk a kódot, azaz vesszük a generátormátrix sorainak valamennyi lineáris kombinációját, a kijelölt oszlopokhoz tartozó részmátrix előállítja a k-dimenziós tér valamennyi vektorát egyszer és csakis egyszer, vagyis a kód szeparábilis az előbb megadott indexekre, és akkor a korábbi megjegyzés alapján a kód szisztematikussá tehető az oszlopok egy alkalmasan választott permutációjával.

□

Az alábbi tétel szerint egy kód ellenőrző mátrixa szoros kapcsolatban áll a távolságával.

5.11. Tétel

Legyen $\mathbf{H}$ a d-távolságú $[n, k]$ kód ellenőrző mátrixa, ahol $k \geq 1$. Ekkor $\mathbf{H}$-nak van d lineárisan összefüggő oszlopa, de bármely ennél kevesebb oszlopa lineárisan független.

Δ

Bizonyítás:

A $V_q^{(n)}$-beli $\mathbf{c}$ akkor és csak akkor eleme $W_q^{(n,k)}$-nak, ha $\mathbf{Hc} = \mathbf{0}$. Legyen a $\mathbf{0} \neq \mathbf{c} \in V_q^{(n)}$ vektor súlya $(n \geq) t \in \mathbb{N}$, $0 \leq i_1 < \cdots < i_t < n$ azok az indexek, amelyekre $c_l \neq 0$, és $\mathbf{h}_{(i)}$ a $\mathbf{H}$ i-edik oszlopa. $\mathbf{0} = \mathbf{Hc} = \sum_{j=1}^{t} c_{i_j} \mathbf{h}_{(i_j)}$ pontosan akkor teljesül, ha a megadott c_{i_j}-k egy kódszó nem nulla kompo-

nensei. Lineáris kód súlya és távolsága azonos, vagyis a kód súlya is d, így a kódhoz tartozó bármely nem nulla vektor legalább d nem nulla komponenst tartalmaz, $\mathbf{Hc}$ csak úgy lehet $\mathbf{0}$, ha $t \geq d$. Ez azt jelenti, hogy d-nél kevesebb (de legalább egy) nem nulla komponensű vektor esetén $\mathbf{Hc} \neq \mathbf{0}$, d-nél kevesebb oszlop lineárisan független. Másrészt van d-súlyú $\mathbf{c}$ kódszó, és ekkor $\mathbf{H}$-ban a $\mathbf{c}$ nem nulla komponenseihez tartozó indexek által meghatározott oszlopok lineárisan összefüggenek, van $\mathbf{H}$-ban d lineárisan összefüggő oszlop.

□

5.12. Definíció

A C lineáris kód **duális**a $C^{\perp}$.

Δ

5.13. Tétel

Legyen $\mathbf{G}$ és $\mathbf{H}$ az $[n, k]$ paraméterű kód generátor- és ellenőrző mátrixa. Ekkor a duális kód generátor- és ellenőrző mátrixa $\mathbf{G}^D = \mathbf{H}$ és $\mathbf{H}^D = \mathbf{G}$.

Δ

Bizonyítás:
$\mathbf{G}$ sorai a kód mint altér egy bázisának elemei, és $\mathbf{H}$ sorai ezen altér ortogonális alterének egy bázisa. Ortogonális altér ortogonális altere az eredeti altér, így igaz az állítás.

□

Az $(n, M)_q$ paraméterű C kód kódsebessége $\mathcal{R}_q = \frac{1}{n}\log_q M$. $[n, k]_q$ kódban $M = q^k$, és ezt alkalmazva kapjuk egy lineáris kód kódsebességét.

5.14. Tétel

Az $[n, k]_q$ paraméterű C lineáris kód kódsebessége $\mathcal{R}_q = \frac{k}{n}$.

Δ

Korábban azt mondtuk, hogy a dekódolás egyszerűsítése érdekében alkalmazunk olyan kódokat, amelyeknek a szerkezete bizonyos strukturális öszefüggéssel rendelkezik. A lineáris kódok ilyenek, ezért azt reméljük, hogy valamilyen jól használható, viszonylag egyszerű és gyors algoritmussal végezhető a dekódolás. Az alábbiakban megmutatjuk, hogy ez valóban így van.

Tekinsünk egy $[n, k, d]_q$-paraméterű C kódot, és legyen $\mathbf{u}$ ennek egy eleme. Tegyük fel, hogy az $\mathbf{u}$-t elküldve, a vétel helyére egy $\mathbf{v} \in \mathbb{F}_q^n$ vektor érkezik. Mivel $\mathbb{F}_q^n$ lineáris tér, ezért $\mathbf{v}$-t felírhatjuk $\mathbf{v} = \mathbf{u} + \mathbf{e}$ alakban, ahol $\mathbf{e}$ szintén $\mathbb{F}_q^n$ egy eleme. $\mathbf{e}$ a **hibavektor**, ugyanis pontosan $\mathbf{e}$ nem nulla elemei mutatják, hogy mely pozícióban és az adott pozícióban milyen hiba történt az átvitel során.

A korábbiakban $\mathbf{H}$-t csupán arra használtuk, hogy ellenőrizzük, vajon $\mathbf{v}$ eleme-e a kódnak: mint tudjuk, $\mathbf{v} \in C$ akkor és csak akkor igaz, ha $\mathbf{Hv} = \mathbf{0}$. Valójában az utóbbi szorzat ennél több információt rejt. Legyen $\mathbf{s} = \mathbf{Hv}$. $\mathbf{s}$ a $\mathbf{v}$ szóhoz tartozó **szindróma**. A szindróma az $\mathbb{F}_q$ fölötti $n - k$-dimenziós tér egy eleme, hiszen $\mathbf{s}$ a $\mathbf{H}$ oszlopainak lineáris kombinációja, és $\mathbf{H}$ rangja $n - k$. Ha $\mathbf{v}$-t javítani akarjuk, akkor ismernünk kell $\mathbf{e}$-t. Mivel $\mathbf{e}$ az n-dimenziós tér bármely eleme lehet, azaz q^n különböző hibavektor van, és a szindrómák száma ennél kevesebb, csupán q^{n-k} (feltéve, hogy valódi kódról van szó, azaz k legalább 1), a szindróma ismeretében nem várhatjuk el, hogy minden esetben helyesen javítunk. Ezt egyébként egyetlen kódtól sem remélhetjük, hiszen ha a vétel helyére egy, a küldöttől eltérő kódszó érkezik, akkor arról elég kevéssé meggyőzően lehetne állítani, hogy hibás. A szindróma ismeretében azonban bizonyos következtetést le tudunk vonni a hibára vonatkozóan:

$$\mathbf{s} = \mathbf{Hv} = \mathbf{s} = \mathbf{H}(\mathbf{u} + \mathbf{e}) = \mathbf{Hu} + \mathbf{He} = \mathbf{0} + \mathbf{He} = \mathbf{He},$$

vagyis a szindróma értéke a hibától függ. Másrészről

$$\mathbf{He}_1 = \mathbf{s}_1 = \mathbf{s}_2 = \mathbf{He}_2 \Leftrightarrow \mathbf{0} = \mathbf{s}_1 - \mathbf{s}_2 = \mathbf{He}_1 - \mathbf{He}_2 = \mathbf{H}(\mathbf{e}_1 - \mathbf{e}_2) \Leftrightarrow \mathbf{e}_1 - \mathbf{e}_2 \in C,$$

tehát két hibavektor szindrómája akkor és csak akkor azonos, ha a különbségük kódszó. Mivel a kód egy altér, és az altér az összeadással részcsoport, ezért az előző megállapítás azt jelenti, hogy két hibavektor szindrómája pontosan akkor azonos, ha a két szó a C mint additív részcsoport szerinti azonos mellékosztályban van. Ezek szerint a szindróma alapján két hibavektort akkor és csak akkor tudunk megkülönböztetni, ha különböző mellékosztályban vannak, vagy másként, azonos mellékosztályban lévő hibavektorok között a szindróma alapján nem tudunk különbséget tenni. Mindez azt jelenti, hogy azonos mellékosztályban lévő hibavektorok közül egyet és csak egyet tekinthetünk reprezentánsnak, azaz minden olyan esetben, amikor a vett szó alapján számított szindróma **s**, akkor az **s** által meghatározott mellékosztály reprezentánsát tekintjük **javítható hibamintá**nak, vagy másként **mellékosztály-vezető**nek, vagyis feltesszük, hogy ez volt a hiba, és ezzel a vélt hibával korrigáljuk a vett szót, a mellékosztályba tartozó többi hibavektor viszont **nem javítható hibaminta**. Ennél többet nem is várhatunk. Ha a vett szó **v**, akkor ez a q^k különböző kódszó bármelyikéből származhatott, vagyis **v** q^k különböző **e** hibavektorral, hibamintával jöhetett létre, de a döntési függvény ezek közül egyre és csak egyre dönthet. Mivel a lehetséges hibaminták száma q^n, ezért átlagban $\frac{q^n}{q^k} = q^{n-k}$ lehet a javítható hibaminták száma, éppen annyi, ahány különböző szindróma van.

A következő kérdés, hogy milyen alapon válasszuk ki az egy mellékosztályba tartozó q^k különböző hibamintából az egyetlen javítható hibamintát. Legyen ez a mellékosztályhoz tartozó vektorok közül (egy) minimális súlyú, vagyis egy tetszőleges **e** hibavektor esetén az **e**-t tartalmazó, **s** szindrómájú C szerinti mellékosztály $\mathbf{e}^{(\mathbf{s})}$-sel jelölt reprezentánsa

$$\mathbf{e}^{(\mathbf{s})} \in \mathbf{e} + C = \{\mathbf{e}' \in \mathbb{F}_q^n | \mathbf{He} = \mathbf{s} = \mathbf{He}\} \wedge w(\mathbf{e}^{(\mathbf{s})}) = \min\{w(\mathbf{e}') | \mathbf{e}' \in \mathbf{e} + C\}$$

Ekkor az f döntési függvény olyan, hogy tetszőleges $\mathbf{v} \in \mathbb{F}_q^n$-re $f(\mathbf{v}) = \mathbf{v} - \mathbf{e}^{(\mathbf{s})}$, ahol $\mathbf{s} = \mathbf{Hv}$. Ha **u** az üzenet, akkor, amint azt már láttuk, $\mathbf{He}^{(\mathbf{s})} = \mathbf{s} = \mathbf{Hv} = \mathbf{H}(\mathbf{u} + \mathbf{e}) = \mathbf{Hu} + \mathbf{He} = \mathbf{He}$, vagyis a tényleges hibaminta a javításra felhasznált hibamintával azonos mellékosztályban van. Ekkor viszont

$$\begin{aligned} d(\mathbf{v}, f(\mathbf{v})) &= d(\mathbf{v}, \mathbf{v} - \mathbf{e}^{(\mathbf{s})}) = w\left(\mathbf{v} - \left(\mathbf{v} - \mathbf{e}^{(\mathbf{s})}\right)\right) = w(\mathbf{e}^{(\mathbf{s})}) = \min\{w(\mathbf{e}) | \mathbf{e} \in \mathbf{e}^{(\mathbf{s})} + C\} \\ &= \min\{w(\mathbf{e}^{(\mathbf{s})} + \mathbf{u}) | \mathbf{u} \in C\} = \min\{w(\mathbf{v} - f(\mathbf{v}) + \mathbf{u}) | \mathbf{u} \in C\} \\ &= \min\{w(\mathbf{v} - (f(\mathbf{v}) - \mathbf{u})) | \mathbf{u} \in C\} = \min\{w(\mathbf{v} - \mathbf{u}') | \mathbf{u}' \in C\} \\ &= \min\{d(\mathbf{v}, \mathbf{u}') | \mathbf{u}' \in C\} \end{aligned}$$

tehát a fenti módon választott mellékosztály-vezetőkkel *f* minimális távolságú dekódolást valósít meg.

Amint látjuk, lineáris kódok esetén a minimális távolságú dekódoláshoz elegendő minden szindrómához tárolni a hozzá tartozó hibamintát. Általános esetben a minimális távolságú dekódoláshoz a lehetséges q^n szó mindegyikéhez tárolni kell a döntési függvény értékét, vagyis egy q^n-méretű tömböt kell tárolni. Ezzel szemben az ismertetett **szindróma-dekódolás**nál a szükséges tárméret csupán q^{n-k}, ami lényegesen kisebb az előző értéknél, és ennek ára mindössze egy mátrixszorzás.

Most legyen **e** tetszőleges, $\frac{d}{2}$-nél kisebb súlyú hibaminta, és $\mathbf{e}'$ az **e**-vel azonos mellékosztályban lévő, **e**-től különböző hibaminta. Ekkor $\mathbf{0} \neq \mathbf{e} - \mathbf{e}' \in C$, így a súlyokra vonatkozó háromszög-egyenlőtlenséggel $d \leq w(\mathbf{e} - \mathbf{e}') \leq w(\mathbf{e}) + w(\mathbf{e}') < \frac{d}{2} + w(\mathbf{e}')$, és innen $w(\mathbf{e}') > d - \frac{d}{2} = \frac{d}{2} > w(\mathbf{e})$, így egy mellékosztályban legfeljebb egy $\frac{d}{2}$-nél kisebb súlyú hibaminta lehet, az ilyen hibaminták mindegyike mellékosztályvezető, tehát a szindróma-dekódolással minden $\frac{d}{2}$-nél kevesebb hiba javítható, a kód a szindrómadekódolással legalább $\left\lfloor \frac{d-1}{2} \right\rfloor$-hiba javító. Legyen ugyanakkor **u** egy d-súlyú kódszó. **u** felírható $\mathbf{u} = \mathbf{u}^{(1)} - \mathbf{u}^2$ alakban úgy, hogy $\mathbf{u}^{(1)}$ megegyezik **u**-val, kivéve valamely $\left\lfloor \frac{d}{2} \right\rfloor$ olyan pozíci-

ót, ahol $\mathbf{u}$ nem nulla, és ezeken a helyeken $\mathbf{u}^{(1)}$-ben 0 áll, míg $\mathbf{u}^2$ éppen ezeken a helyeken azonos $-\mathbf{u}$-val, és minden más helyen 0. Ekkor $w(\mathbf{u}^{(1)}) = \left\lceil \frac{d}{2} \right\rceil = \left\lfloor \frac{d-1}{2} \right\rfloor + 1$, $w(\mathbf{u}^{(2)}) = \left\lfloor \frac{d}{2} \right\rfloor$, és $\mathbf{u}^{(1)}$, $\mathbf{u}^2$ azonos mellékosztályban van (hiszen a különbségük kódszó), vagyis kettejük mint hibaminták közül legfeljebb az egyik javítható, így biztosan lesz olyan, legfeljebb $\left\lceil \frac{d}{2} \right\rceil = \left\lfloor \frac{d-1}{2} \right\rfloor + 1$ súlyú hibaminta, amely nem javítható, így a kód pontosan $\left\lfloor \frac{d-1}{2} \right\rfloor$-hiba javító (ez nyilván nem meglepő, hiszen a szindróma-dekódolás minimális távolságú dekódolás, és d-távolságú kód minimális távolságú dekódolással pontosan $\left\lfloor \frac{d-1}{2} \right\rfloor$-hiba javító).

6. Ciklikus kódok

Emlékeztetünk rá, hogy ha a és $c \neq 0$ valós számok, akkor $a \bmod c = a - c\left\lfloor\frac{a}{c}\right\rfloor$, ami abban az esetben, ha a egész szám és c pozitív egész szám, azt jelenti, hogy $a \bmod c$ az a osztási maradéka a c-vel való osztáskor, azaz $c > a \bmod c \in \mathbb{N}$ és $a \bmod c \equiv a\ (c)$. Most legyen $\mathcal{R}$ egységelemes kommutatív gyűrű, és f valamint h $\mathcal{R}$ fölötti polinomok, ahol h főegyütthatója egység $\mathcal{R}$-ben. Ekkor f maradékosan osztható h-val úgy, hogy a maradék vagy 0, vagy a fokszáma kisebb h fokszámánál, és ez a maradék egyértelműen meghatározott. Jelölje ezt az egyértelműen meghatározott osztási maradékot $f \bmod h$.

6.1. Definíció

A $C \subseteq S^n$ kód **ciklikus**, ha $\mathbf{u}^T = u_0 \dots u_{n-2}u_{n-1} \in C$ esetén $\mathbf{u}^T_{\rightarrow} = u_{n-1}u_0 \dots u_{n-2} \in C$. Az $l \in \mathbb{Z}$ hellyel való ciklikus jobbra léptetéssel kapott vektort $\mathbf{u}_{\underrightarrow{(l)}}$ jelöli.

Δ

Nyilván igaz, hogy $\mathbf{u}_{\underrightarrow{(l+1)}} = \left(\mathbf{u}_{\underrightarrow{(l)}}\right)_{\rightarrow}$, továbbá $\mathbf{u}_{\underrightarrow{(l)}} = \mathbf{u}_{\underrightarrow{(l \bmod n)}}$.

A definícióban nem kötöttük ki, ám a továbbiakban feltesszük, hogy a ciklikus kód lineáris is. Ekkor a ciklikus kódok tárgyalását segíti, ha a kódszavakat polinomként kezeljük.

6.2. Definíció

Legyen $n \in \mathbb{N}^+$, $\mathbf{u} \in V_q^{(n)}$, $u = \sum_{i=0}^{n-1} u_i x^i$, és $S^{(n)} = \left\{u \in \mathbb{F}_q[x] \middle| \mathbf{u} \in V_q^{(n)}\right\}$. Ha C egy $[n,k]_q$-paraméterű ciklikus kód, akkor $S^{[C]} = \left\{u \in \mathbb{F}_q[x] \middle| \mathbf{u} \in C\right\}$ **a kódpolinomok halmaza**, és $u \in S^{[C]}$ **az u kódszóhoz tartozó kódpolinom**.

Δ

Közvetlenül látható, hogy bármely két kódszóra és testbeli elemre $a\mathbf{g}$-hez illetve $\mathbf{g}^{(1)} + \mathbf{g}^{(2)}$-höz tartozó kódpolinom ag és $g^{(1)} + g^{(2)}$.

6.3. Tétel

Ha $n \in \mathbb{N}^+$ és $\mathbf{u} \in V_q^{(n)}$, akkor $u_{\rightarrow} = xu \bmod (x^n - e)$.

Δ

Bizonyítás:
$u = \sum_{i=0}^{n-1} u_i x^i$-ből $xu \bmod (x^n - e) = u_{n-1}x^0 + \sum_{i=1}^{n-1} u_{i-1}x^i = \sum_{i=0}^{n-1} u_{(i-1) \bmod n} x^i = u_{\rightarrow}$.
□

A fentiekből könnyen kapjuk, hogy $u_{\underrightarrow{(l)}} = x^{l \bmod n} u \bmod (x^n - e) = x^l u \bmod (x^n - e)$, tekintetbe véve, hogy $\mathbf{u}_{\underrightarrow{(l)}} = \mathbf{u}_{\underrightarrow{(l \bmod n)}}$ és $f(g \bmod h) = fg \bmod h$.

6.4. Tétel

Legyen $n \in \mathbb{N}^+$ és $n \geq k \in \mathbb{N}^+$. Az $[n,k]_q$-paraméterű C ciklikus kódhoz van $S^{[C]}$-ben olyan egyértelműen meghatározott $n-k$-adfokú g főpolinom, hogy $S^{[C]} = \left\{ag \middle| a \in \mathbb{F}_q[x] \wedge \delta(a) < k\right\}$, to-

vábbá $S^{[C]}$ elemeinek ilyen felírása egyértelmű, és $g|x^n - e$. Fordítva, ha az $n-k$-adfokú g főpolinom osztója az $x^n - e$ polinomnak, akkor az $S = \{ag | a \in \mathbb{F}_q[x] \wedge \delta(a) < k\}$ halmaz egy $[n,k]_q$-paraméterű ciklikus kódhoz tartozó kódpolinomhalmaz.

Δ

Bizonyítás:

1. $k \in \mathbb{N}^+$, ezért C legalább egydimenziós, van benne nem nulla vektor és $S^{[C]}$-ben nem nulla polinom, tehát $\emptyset \neq A = \{\deg(f) | 0 \neq f \in S^{[C]}\} \subseteq \mathbb{N}$, így létezik és egyértelmű az A legkisebb eleme. Ha ez t, akkor van $S^{[C]}$-ben olyan p polinom, amelynek a foka t. p főegyütthatója nem nulla, ezért van inverze, és ezzel szorozva p-t, egy t-edfokú g főpolinomot nyerünk. $n > t \in \mathbb{N}$, hiszen a kódszavak n-komponensűek. Legyen $S = \{ag | a \in \mathbb{F}_q[x] \wedge \delta(a) < n - t\}$, belátjuk, hogy $t = n - k$ és $S = S^{[C]}$.

$g \in S^{[C]}$, tehát $\mathbf{g}^{(0)} = \mathbf{g} \in C$. A kód ciklikus, ezért iterációval kapjuk, hogy bármely $i \in \mathbb{N}$-re $\mathbf{g}^{(i)} = \mathbf{g}_{\overrightarrow{(i)}}$ is eleme a kódnak. Fentebb láttuk, hogy $g^{(i)} = g_{\overrightarrow{(i)}} = x^i g \bmod (x^n - e)$. Ha $i < n - t$, akkor $x^i g$ fokszáma alacsonyabb n-nél. Mivel egy n-nél alacsonyabbfokú polinomot egy n-edfokú polinommal osztva a hányados 0, és így a maradék megegyezik az osztandóval, ezért az előbbiek alapján kapjuk, hogy a megadott intervallumba eső i-k esetén $g^{(i)} = x^i g$. Az $\mathbb{F}_q$-beli $a_0, \ldots, a_{n-t-1}$ elemekkel $\sum_{i=0}^{n-t-1} a_i \mathbf{g}^{(i)}$ is kódvektor, hiszen a kód lineáris, ezért

$$\sum_{i=0}^{n-t-1} a_i g^{(i)} = \sum_{i=0}^{n-t-1} a_i (x^i g) = \left(\sum_{i=0}^{n-t-1} a_i x^i \right) g = ag$$

is benne van $S^{[C]}$-ben, ahol $a = \sum_{i=0}^{n-t-1} a_i x^i \in \mathbb{F}_q[x]$ és $\delta(a) < n - t$, tehát $S \subseteq S^{[C]}$.

Most legyen $u \in S^{[C]}$. u egyértelműen írható $u = cg + r$ alakban, ahol $\delta(r) < \deg(g) = t$. Mivel u n-nél alacsonyabbfokú, míg r t-nél alacsonyabbfokú, és t kisebb, mint n, ezért $cg = u - r$ is n-nél alacsonyabbfokú polinom, vagyis $\delta(c) < n - t$. Ekkor $cg \in S^{[C]}$, és innen $r = u - cg \in S^{[C]}$, hiszen a kód lineáris, így két kódszó különbsége is kódszó. De az $S^{[C]}$-beli nem nulla polinomok legalább t-edfokúak, így r csak a nullpolinom lehet, tehát $u = cg$ és $u \in S^{[C]}$, azaz $S^{[C]} \subseteq S$. Figyelembe véve az előbb megállapított ellenkező irányú tartalmazást kapjuk, hogy $S = S^{[C]}$.

$g \neq 0$, így $a^{(1)} g = a^{(2)} g \Leftrightarrow a^{(1)} = a^{(2)}$, és ag pontosan akkor kódpolinom, ha $\delta(a) < n - t$, ezért kölcsönösen egyértelmű megfeleltetés létesíthető C és S elemei között. C-nek q^k, míg S-nek q^{n-t} eleme van, a két érték megegyezik, így $k = n - t$, azaz $t = n - k$, tehát g egy $n-k$-adfokú főpolinom.

2. $\mathbf{g}^{(k)} \in C$, tehát $x^k g \bmod (x^n - e) = g^{(k)} \in S^{[C]}$. Mivel g egy $n-k$-adfokú főpolinom, ezért $x^k g = (x^n - e) + r$, ahol $\delta(r) < n$. $r = g^{(k)} \in C$, így r, és akkor $x^n - e = x^k g - r$ is osztható g-vel.

3. Tekintsük az S-beli polinomokat. Ezek mindegyike n-nél alacsonyabbfokú, $\mathbb{F}_q$ feletti polinom, ezért mindegyikük egy-egy $\mathbb{F}_q$ feletti n-dimenzós vektort határoz meg. S-beli polinomok összege és $\mathbb{F}_q$-beli konstansszorosa is S-beli, így a megfelelő vektorok alteret alkotnak $V_q^{(n)}$-ben, és a számosság alapján az altér dimenziója k. Legyen $\sum_{i=0}^{n-1} f_i x^i = f \in S$. Ekkor $xf = f_{n-1} \cdot (x^n - e) + r$, és $\delta(r) < n$. Itt $g|f$, mivel f kódpolinom, továbbá $g|x^n - e$ a g választása folytán igaz, de ekkor $g|r$ is teljesül, és mivel a fokszám is megfelelő, ezért $f_{\rightarrow} = r \in S$, S egy ciklikus kód polinomhalmaza.

□

Ha $m \in \mathbb{N}^+$ kisebb, mint n, és az $[n,k]$-paraméterű ciklikus kódhoz tartozó g polinom osztója az $x^m - e$ polinomnak, akkor $x^m - e$ is kódpolinom. De az $x^m - e$-nek megfelelő kódszóban pontosan két nullától különböző komponens van, így a kód távolsága legfeljebb 2, a kód egyetlen hiba javítására sem alkalmas (pontosabban szólva van olyan egyetlen hiba, amelyet a kód nem képes javítani).

Mivel g osztója $x^n - e$-nek, és mindkét polinom főpolinom, ezért $h = \frac{x^n - e}{g}$ is főpolinom.

6.5. Definíció

Az $[n,k]$ ciklikus kódhoz tartozó, egyértelműen meghatározott g polinom a **kód generátorpolinomja**, és $h = \frac{x^n - e}{g}$ a **kód ellenőrző polinomja**.

Δ

Amint a generátorpolinom ismeretében valamennyi kódszó megkapható egy polinommal való szorzással, az ellenőrzéshez is elegendő az ellenőrző polinommal való szorzás.

6.6. Tétel

Legyen h egy $[n,k]_q$-praméterű C ciklikus kód ellenőrző polinomja. Ekkor $c \in \mathbb{F}_q[x]$ pontosan akkor kódszópolinom, ha $\delta(c) < n$ és $hc \bmod (x^n - e) = 0$.

Δ

Bizonyítás:

$[n,k]$-paraméterű ciklikus kód bármely c kódpolinomjára $\delta(c) < n$, legyen tehát $c \in \mathbb{F}_q[x]$-re $\delta(c) < n$. $hc \bmod (x^n - e) = 0$ akkor és csak akkor, ha $hc = a \cdot (x^n - e) = a(gh) = (ag)h$ egy a polinommal, vagyis pontosan akkor, ha $c = ag$, ahol $\delta(a) < k$, vagyis ha c kódpolinom.

□

Könnyű látni, hogy bármely n természetes számra $[n,0]_q$ és $[n,n]_q$ ciklikus. Az előbbi csak a nullvektort tartalmazza, míg az utóbbi a teljes tér, márpedig a nullvektor bármely lineáris kombinációja és ciklikus eltoltja önmaga, míg a teljes tér nyilván lineáris, és minden vektor eltoltja is eleme ugyanezen térnek. Az előbbi kódot akkor kapjuk, ha $g = x^n - e$, hiszen most ahhoz, hogy ag legfeljebb $n-1$-edfokú legyen, szükséges, hogy a maga a nullpolinom legyen. Ez a nullvektornak felel meg, ami minden lineáris kódnak eleme. Az utóbbi kódot a $g = e$ polinom generálja, és így minden legfeljebb $n-1$-edfokú polinom kódpolinom, ezek halmaza viszont izomorf az n-dimenziós térrel.

Egy ciklikus kód nem feltétlenül lineáris, ám a gyakorlatban szinte mindig az. Ekkor a kódhoz megadható a generátor- és ellenőrző mátrix. Legyen az $[n,k]$-paraméterű ciklikus kód generátor- és ellenőrző polinomja g és h. Tekintsük azt a $k \times n$-méretű $\mathbf{G}$ mátrixot, amelynek i-edik sora az $x^i g$ polinom által meghatározott $\mathbf{g}^{(i)^T} = \underbrace{0 \dots 0}_{i} g_0 \dots g_{n-k} \underbrace{0 \dots 0}_{k-1-i}$ sorvektor, míg $\mathbf{H}$ egy $(n-k) \times n$-méretű mátrix, amelynek i-edik sora $\mathbf{h}^{(i)^T} = \underbrace{0 \dots 0}_{i} (h_0^{-1}h_k) \dots (h_0^{-1}h_0) \underbrace{0 \dots 0}_{n-k-1-i}$, azaz az $x^i h_0^{-1} h^*$ polinomhoz tartozó kódszó (a h_0^{-1}-gyel való szorzás biztosítja, hogy $h_0^{-1}h^*$ főpolinom legyen). Kevés számolással belátható, hogy $\mathbf{G}$ a kód egy generátor- és $\mathbf{H}$ egy ellenőrző mátrixa.

Ha ismerjük egy lineáris kód $\mathbf{G}$ generátor- és $\mathbf{H}$ ellenőrző mátrixát, akkor meg tudjuk adni a duális kód valamely $\mathbf{G}^D$ generátor- és $\mathbf{H}^D$ ellenőrző mátrixát, hiszen például $\mathbf{G}^D = \mathbf{H}$ és $\mathbf{H}^D = \boldsymbol{G}$ egy alkalmas választás. Ebből látható, hogy a duális kód generátorpolinomja $g^{(D)} = h_0^{-1}h^*$, hiszen a ciklikus kód generátormátrixában a generátorpolinom szerepel, elől a konstans taggal, míg $\mathbf{H}$-ban az ellenőrző polinom, de balról a legmagasabb fokú taggal. A h_0^{-1}-gyel való szorzás azért kell, mert a generátorpolinom főpolinom. Mivel $(gh)^* = g^*h^*$, és $g_0 h_0 = (gh)_0 = -e$, ezért a duális kód ellenőrző polinomja $h^{(D)} = g_0^{-1} g^*$. Ha az eredeti kód n hosszúságú, akkor $h_0^{-1}h^*$ is osztója $x^n - e$-nek, így

6.7. Tétel

Ciklikus kód duálisa is ciklikus kód.

Δ

Gyakran a h által generált kódot tekintik a g-hez tartozó ciklikus kód duálisának. Ez nem azonos az előbbivel, de skalárekvivalens vele.

A ciklikus kódra adott generátormátrix általában nem standard alakú, azonban az $u \mapsto ug$ helyett más megfeletetést alkalmazva a kód szisztematikussá tehető. Legyen az $[n,k]_q$-paraméterű ciklikus kódban $u \mapsto v = x^{n-k}u - \left(x^{n-k}u \bmod g\right)$. Mivel $(a \bmod b) \bmod b = a \bmod b$, ezért az előbbi v-re $v \bmod g = 0$, vagyis g osztója v-nek, v tehát kódpolinom. Az így generált kód szisztematikus az utolsó k pozíciójára, mert $x^{n-k}u$-ban az $n-k$-nál alcsonyabbfokú tagok együtthatója 0, és a maradék legfeljebb $n-k-1$-edfokú, azaz v-ben az $n-k$-nál nem kisebb fokszámú tagok együtthatói egybeesnek u ugyanolyan sorrendben álló együtthatóival. Az előbbiek alapján, ha $r = x^{n-k}u \bmod g$, akkor az $\mathbf{u}$ üzenethez tartozó kódszó $\mathbf{v}^T = -\mathbf{r}^T|\mathbf{u}^T$. Ekkor generátormátrixot kapunk, ha az x^i-khez tartozó $-\mathbf{r}^{(i)^T}\left|\mathbf{e}^{(i)^T}\right.$ kódszavakat mint sorvektorokat tartalmazó mátrixot tekintjük, ahol $\mathbf{e}^{(i)}$ a k-dimenziós tér i-edik egységvektora, és $\mathbf{r}^{(i)}$ az x^{n-k+i} g-vel való osztásakor keletkező maradékának, $\mathbf{r}^{(i)}$-nek megfelelő vektor, vagyis $\mathbf{G}^{(sz)} = (-\mathbf{P}^T\ \mathbf{I}_k)$, ahol $\mathbf{P}$ i-edik oszlopa $\mathbf{r}^{(i)}$ (az sz jelölés a szisztematikus generálásra utal). Ebből az is következik, hogy az így generált kód ellenőrző mátrixa $\mathbf{H} = (\mathbf{I}_{n-k}\ \mathbf{P})$.

A lineáris kódok előnye az általános, strukturálatlan kódokkal szemben, hogy könnyebb a generálás, másrészt a szindróma segítségével könnyebb a hibajavítás is. A ciklikus kódok erősebb struktúrával rendelkeznek, mint általában egy lineáris kód, és ez megmutatkozott például abban is, hogy míg a lineáris kód generálásához egy egész mátrix kell, addig a ciklikus kódot teljes egészében meghatározza a generátorpolinomja (persze ha ismerjük a kód hosszát is). Most belátjuk, hogy a hibajavítás szempontjából is tömörebben tudjuk megadni a ciklikus kódot, mint egy általános lineáris kód esetén.

Tekintsük a g által generált $[n,k]_q$-paraméterű C ciklikus kódot, és legyen S az $\mathbb{F}_q$ fölötti, n-nél alacsonyabbfokú polinomok halmaza (beleértve a nullpolinomot is). $v \in S$ pontosan akkor eleme a kódnak, ha g osztója v-nek, vagyis ha $v \bmod g = 0$. Tetszőleges $v \in S$-re legyen $s = v \bmod g$. s a g-vel való osztás maradéka, ezért $\delta(s) < \deg(g) = n-k$, és $\deg(v) < n-k$ esetén $s = v \bmod g$, vagyis minden, legfeljebb $n-k-1$-edfokú polinom fellép maradékként, így a különböző maradékok száma q^{n-k}. Az S $v^{(1)}$ és $v^{(2)}$ elemére a $v^{(1)} \bmod g = s^{(1)} = s^{(2)} = v^{(2)} \bmod g$ egyenlőség pontosan akkor teljesül, amikor $\left(v^{(1)} - v^{(2)}\right) \bmod g = \left(v^{(2)} \bmod g\right) - \left(v^{(1)} \bmod g\right) = s^{(2)} - s^{(1)} = 0$, vagyis ha $v^{(1)} - v^{(2)} \in C$, azaz akkor és csak akkor, ha a két polinomhoz tartozó vektor szindrómája azonos. Ez azt jelenti, hogy kölcsönösen egyértelmű megfeleltetés adható az $\mathbb{F}_q$ fölötti n-dimenziós tér C szerinti szindrómái, valamint a megfelelő polinomok g-vel való osztási maradékai között, így ez a maradék ugyanúgy alkalmazható hibajavításra, mint a lineáris kódok esetén a szindróma. Másként szólva, ciklikus kódok esetén a generátorpolinommal való osztás maradéka tekinthető a vett szó szindrómájának.

Egészen szoros a kapcsolat egy vektor szindrómája és az előbbi osztási maradék között, ha a kódolást az $u \mapsto x^{n-k}u - \left(x^{n-k}u \bmod g\right)$ szabállyal végezzük. Ekkor a $v = \sum_{i=0}^{n-1} v_i x^i$ polinomhoz tartozó v vektor szindrómája

$$\mathbf{s} = \mathbf{H}\mathbf{v} = (\mathbf{I}_{n-k}\ \mathbf{P})\mathbf{v} = (\mathbf{I}_{n-k}\ \mathbf{P})\begin{pmatrix}\mathbf{v}^{(p)}\\ \mathbf{v}^{(a)}\end{pmatrix} = \mathbf{v}^{(p)} + \mathbf{P}\mathbf{v}^{(a)} = \sum_{i=0}^{n-k-1} v_i\mathbf{e}^{(i)} + \sum_{i=n-k}^{n-1} v_i\mathbf{r}^{(i-(n-k))},$$

és ennek a vektornak az

$$\begin{aligned} s &= \sum_{i=0}^{n-k-1} v_i x^i + \sum_{i=n-k}^{n-1} v_i\left(x^{n-k+(i-(n-k))} \bmod g\right) \\ &= \sum_{i=0}^{n-1} v_i\left(x^i \bmod g\right) = \sum_{i=0}^{n-1} v_i x^i \bmod g = v \bmod g \end{aligned}$$

polinom felel meg. Az átalakításnál kihasználtuk, hogy $\sum_{i=0}^{n-k-1} v_i x^i$ maradéka a g-vel való osztáskor önmaga. Azt látjuk tehát, hogy ebben az esetben a korábban említett bijektív megfeleltetés során egy szindrómát az általa reprezentált polinomnak feleltetünk meg.

Ciklikus kódban egy vektor eltoltjának szindrómáját az eredeti vektor szindrómájából is meghatározhatjuk. Legyen s a v és $s^{\rightarrow}$ a $v_{\rightarrow}$ szindrómája. Ekkor

$$\begin{aligned} s^{\rightarrow} &= v_{\rightarrow} \bmod g = (xv \bmod (x^n - e)) \bmod g \\ &= x(v \bmod g) \bmod g = xs \bmod g = xs - s_{n-k-1} g, \end{aligned}$$

ahol s_{n-k-1} az s polinom $n-k-1$-edfokú tagjának együtthatója.

Az $x^n - e$ polinom gyökei n-edik egységgyökök. A továbbiakban feltesszük hogy az $[n,k]_q$-paraméterű kódban n és q relatív prím. Ekkor $x^n - e$ gyökei egyszeresek, létezik primitív n-edik egységgyök, és a polinom gyökei egy primitív n-edik egységgyök páronként különböző, $n > i \in \mathbb{N}$ kitevős hatványai. Az $[n,k]_q$-paraméterű C ciklikus kód generátor- és ellenőrző polinomja osztója az $x^n - e$ polinomnak, ezért ezek gyökei is egyszeresek, és $gh = x^n - e$, ezért az egyszeres gyökökből az is következik, hogy g és h relatív prímek.

Most a ciklikus kódok más tulajdonságait vizsgáljuk. Az $[n,k]_q$-paraméterű C ciklikus kód a kód g generátorpolinomjának legfeljebb $n-1$-edfokú többszöröseiből áll, így egy legfeljebb $n-1$-edfokú $c \in \mathbb{F}_q[x]$ polinom pontosan akkor kódszó, ha osztható g-vel. Ebből kapjuk a következő tételt.

6.8. Tétel

Az $\mathbb{F}_q$ fölötti legfeljebb $n-1$-edfokú c polinom pontosan akkor eleme a g polinom által generált $[n,k]_q$-paraméterű C ciklikus kódnak, ha g minden gyöke gyöke c-nek.

Δ

Bizonyítás:
Legyen α a g egy gyöke. Ha $c \in C$, akkor $x - \alpha | g | c$, és így α gyöke c-nek. Fordítva, ha $\alpha_0, \dots, \alpha_{l-1}$ a g páronként különböző gyökei, és minden $l > i \in \mathbb{N}$-re α_i a c gyöke, akkor minden i-re $x - \alpha_i$ osztója c-nek, és így ezek legkisebb közös többszöröse is osztja c-t. De az α_i-k páronként különbözőek, így az $x - \alpha_i$ gyöktényezők páronként relatív prímek, és így a legkisebb közös osztójuk ezek szorzata, vagyis g, tehát g osztója c-nek.

□

c tehát akkor és csak akkor kódpolinom, ha g minden gyöke gyöke c-nek. Ennél kevesebb is elég. Legyen $g = \prod_{i=0}^{t-1} m_i$ a g $\mathbb{F}_q$ fölötti irreducibilis felbontása. Ha α_j gyöke m_i-nek, akkor m_i lényegében véve (egy esetleges nem nulla konstans szorzótól eltekintve) α_j $\mathbb{F}_q$ fölötti minimálpolinomja, így m_i akkor és csak akkor osztója c-nek, ha α_j gyöke c-nek, és nyilván c akkor és csak akkor kódpolinom, ha valamennyi m_i-vel osztható. Ebből azt kapjuk, hogy c akkor és csak akkor kódpolinom, ha valamennyi m_i legalább egy gyöke gyöke c-nek. Mindez azt jelenti, hogy az $\mathbb{F}_q$ fölötti, n hosszúságú kódszavakat tartalmazó ciklikus kód megadható mint a legbővebb halmaz, amelynek bizonyos elemek a gyökei. Ha az előírt gyökök $\alpha_0, \dots, \alpha_{l-1}$, ahol l nemnegatív egész, és az α_i-k páronként különböző, $\mathbb{F}_q$ fölötti n-edik egységgyökök, továbbá $m_{\alpha_i}^{(\mathbb{F}_q)}$ az α_i $\mathbb{F}_q$ fölötti minimálpolinomja, akkor ezen polinomok legkisebb közös többszöröse a legalacsonyabb fokú olyan polinom, amelynek a megadott egységgyökök mindegyike gyöke, és ha ez a polinom g, akkor tehát a g által generált kód a legbővebb, amelynek minden megadott α_i gyöke.

Legyen a g által generált $[n,k]_q$-paraméterű ciklikus kód C, $\alpha_0, \dots, \alpha_{l-1}$ a g gyökeinek olyan halmaza, amely a g valamennyi, $\mathbb{F}_q$ fölött irreducibilis tényezőjének legalább egy gyökét tartalmazza,

és S az $\mathbb{F}_q$ fölötti, n-nél alacsonyabbfokú polinomok halmaza. Ekkor az előbbiek szerint az S-beli c pontosan akkor eleme a kódnak, ha valamennyi megadott α_i gyöke a polinomnak. Ha $c = \sum_{i=0}^{n-1} c_i x^i$, akkor tehát c akkor és csak akkor eleme S-nek, ha minden $l > i \in \mathbb{N}$ -re $0 = \hat{c}(\alpha_i) = \sum_{j=0}^{n-1} c_j \alpha_i^j$. Most legyen $\widetilde{\mathbf{H}}$ egy $l \times n$-méretű mátrix, amelyben az $l > i \in \mathbb{N}$ és $n > j \in \mathbb{N}$ indexekre $\widetilde{H}_{i,j} = \alpha_i^j$. Ekkor $\left(\widetilde{\mathbf{H}}\mathbf{v}\right)_i = \sum_{j=0}^{n-1} \widetilde{H}_{i,j} v_j = \sum_{j=0}^{n-1} \alpha_i^j v_j = \hat{v}(\alpha_i)$, vagyis $\mathbf{v}$ akkor és csak akkor kódszó, ha $\widetilde{\mathbf{H}}\boldsymbol{v} = \mathbf{0}$. $\widetilde{\mathbf{H}}$ rangja l. Ehhez elég megmutatni, hogy a sorai lineárisan függetlenek. Mivel az α_i-k páronként különböző n-edik egységgyökök, ezért $l \leq n$. Tekintsük az első l oszlopból álló részmátrix determinánsát. Ez a páronként különböző α_i-k által generált Vandermonde-determináns, így az értéke nem 0, ami mutatja, hogy ez a részmátrix reguláris, vagyis a sorai lineárisan függetlenek. Ebből viszont következik, hogy $\widetilde{\mathbf{H}}$ sorai is lineárisan függetlenek.

A megadott $\widetilde{\mathbf{H}}$ azonban általában nem a kód ellenőrző mátrixa, ugyanis α_i általában nem eleme $\mathbb{F}_q$-nak, és így $\widetilde{\mathbf{H}}$ nem egy $\mathbb{F}_q$ test fölötti mátrix. Legyen $\mathbb{F}_{q^r}$ az $\mathbb{F}_q$ legszűkebb olyan bővítése, amely tartalmazza a kód megadott gyökeit, és legyen $\left\{\beta^{(i)} \middle| r > i \in \mathbb{N}\right\}$ az $\mathbb{F}_{q^r}$ egy $\mathbb{F}_q$ fölötti bázisa. $\mathbb{F}_{q^r}$ valamennyi eleme egyértelműen felírható a $\beta^{(i)}$-k $\mathbb{F}_q$-beli együtthatós lineáris kombinációjaként, így kölcsönösen egyértelmű megfeleltetés adható $\mathbb{F}_{q^r}$ elemei, valamint az $\mathbb{F}_q^r$ elemei között, és az is igaz, hogy ez a megfeleltetés művelettartóan képezi le $\mathbb{F}_{q^r}$-t mint $\mathbb{F}_q$ fölötti lineáris teret az $\mathbb{F}_q$ fölötti $\mathbb{F}_q^r$ lineáris térre. Helyettesítsük most $\widetilde{\mathbf{H}}$ elemeit a megfelelő $\mathbb{F}_q^r$-beli elemmel, tehát egy r-komponensű oszlopvektorral. Ekkor egy $\mathbb{F}_q$ fölötti $lr \times n$-méretű $\mathbf{H}'$ mátrixot kapunk. Ennek a mátrixnak a sorai azonban nem feltétlenül lineárisan függetlenek. Legyen $\mathbf{H}$ az előbbi mátrix sorainak egy maximális lineárisan független rendszeréből álló mátrix. Az nyilván igaz, hogy egy $\mathbf{v} \in \mathbb{F}_q^n$-re $\widetilde{\mathbf{H}}\mathbf{v} = \mathbf{0}$ akkor és csak akkor, ha $\mathbf{H}'\mathbf{v} = \mathbf{0}$, és ha $\mathbf{H}'\mathbf{v} = \mathbf{0}$, akkor $\mathbf{H}\mathbf{v} = \mathbf{0}$. De $\mathbf{H}'$ minden sora a $\mathbf{H}$ sorainak lineáris kombinációja, így ha $\mathbf{H}\mathbf{v} = \mathbf{0}$, akkor $\mathbf{H}'\mathbf{v} = \mathbf{0}$ is teljesül, és végeredményben $\widetilde{\mathbf{H}}\mathbf{v} = \mathbf{0}$ pontosan akkor igaz, ha $\mathbf{H}\mathbf{v} = \mathbf{0}$, így $\mathbf{H}$ a kód ellenőrző mátrixa.

Mivel $\widetilde{\mathbf{H}}$ sorai lineárisan függetlenek $\mathbb{F}_{q^r}$, de ekkor $\mathbb{F}_q$ fölött is, ezért van $\widetilde{\mathbf{H}}$-ban l $\mathbb{F}_q$ fölött lineárisan független oszlop. Az ezeknek megfelelő $\mathbf{H}'$-beli oszlopok is lineárisan függetlenek $\mathbb{F}_q$ fölött, és így $\mathbf{H}'$-ben van l $\mathbb{F}_q$ fölött lineárisan független sor. Ekkor $\mathbf{H}$ sorainak száma legalább l, ugyanakkor legfeljebb lr, hiszen ennyi sora volt $\mathbf{H}'$-nek, azaz ha a kód $[n, k]$-paraméterű, vagyis $\mathbf{H}$ sorainak száma, azaz $\mathbf{H}$ rangja $n - k$, akkor $l \leq n - k \leq lr$, és innen $n - lr \leq k \leq n - l$.

Az előbbi eredmények alapján tegyük fel, hogy α a q-elemű test fölötti n-edik primitív egységgyök, és az $[n, k]_q$-paraméterű kódnak – esetleg többek között – $\alpha^{\tau+i}$-k páronként különböző gyökei, ahol $\tau \in \mathbb{Z}$, $2 \leq \delta \in \mathbb{N}$, és $\delta - 1 > i \in \mathbb{N}$. Azt nyilván feltehetjük, hogy $k > 0$, mert különben a kód csupán a nullvektorból állna, és így $\delta \leq n$ (hiszen csak n különböző n-edik egységgyök van, és ha $\delta > n$, akkor valamennyi n-edik egységgyök gyöke a kódnak, tehát minden kódszónak legalább n gyöke van, ami csak úgy lehet, ha egyedül a nullpolinom eleme a kódnak, mert minden más kódpolinom legfeljebb $n - 1$-edfokú, azaz legfeljebb $n - 1$ gyöke van), továbbá $0 \leq \tau < n$. Legyen a kód előbbi gyökeire $\alpha_i = \alpha^{\tau+i}$. Ekkor $\widetilde{H}_{i,j} = \left(\alpha^{\tau+i}\right)^j = \left(\alpha^{j\tau}\right)\left(\alpha^j\right)^i$ a $\delta - 1 > i \in \mathbb{N}$ és $n > j \in \mathbb{N}$ indexekre. Tekintsük a $\widetilde{\mathbf{H}}$ első $\delta - 1$ sorából és a $0 \leq j_0 < \cdots < j_{\delta-2} < n$ indexű oszlopokból álló $\delta - 1$-edrendű $\mathbf{M}^{(j_0,\ldots,j_{\delta-2})}$ részmátrix determinánsát. Ebben a determinánsban a t-edik oszlopban mindegyik elem tartalmazza szorzóként a 0-tól különböző $\alpha^{j_t\tau}$-t. Ha ezt az elemet kiemeljük az oszlopból, akkor a t-edik oszlop i-edik sorában $\left(\alpha^{j_t}\right)^i$ áll, vagyis ha minden oszlopból kiemeltük az adott oszlophoz tartozó közös tényezőt, akkor a visszamaradt determináns egy csupa különböző elemmel generált Vandermonde-determináns, tehát különbözik nullától. Ez azt jelenti, hogy $\widetilde{\mathbf{H}}$ bármely legfeljebb $\delta - 1$ oszlopa lenárisan független, de akkor ez igaz $\mathbf{H}$-ra is, és így a kód távolsága legalább δ. Bebizonyítottuk tehát a következőt.

6.9. Tétel

Legyen C egy $[n,k,d]_q$-paraméterű ciklikus kód, ahol $k>0$, α egy $\mathbb{F}_q$ fölötti primitív n-edik egységgyök, $\tau\in\mathbb{Z}$, $2\le\delta\in\mathbb{N}$, és $\delta-1>i\in\mathbb{N}$-re $\alpha^{\tau+i}$ a C gyöke. Ekkor $d\ge\delta$, ha $\delta\le n$.

Δ

A tétel alapján annak az $\mathbb{F}_q$ fölötti legbővebb, n-hosszúságú kódszavakból álló, legalább egydimenziós C ciklikus kódnak a távolsága, amelynek az n-edik primitív egységgyök $\delta-1$ egymás utáni hatványa a gyöke, legalább δ. Az így konstruált kód az $\mathbb{F}_q$ **fölötti** $(n,\tau,\delta)_q$**-paraméterű BCH**-kód, ahol τ az első gyök kitevője. (A BCH-kód elnevezés a három megalkotójának nevéből ered: **B***ose* és *Ray*-**C***haudhuri* 1960-ban, **H***ocquenghem* 1959-ben foglalkozott ezzel a kódkonstrukcióval).

A BCH-kód jelentőségét az adja, hogy olyan kód tervezésére ad lehetőséget, amelynek minimális távolsága egy előre adott értéknél nem kisebb, márpedig a javítható hibák száma a távolsággal van összefüggésben.

A kód gyökeivel kapcsolatban még egy dolgot említünk. Legyen $g|x^n-e$, de $\hat{g}(e)\neq 0$. e gyöke az x^n-e polinomnak, ezért még $g^{(1)}=(x-e)g$ is osztója x^n-e -nek. Ha C a g és $C^{(1)}$ a $g^{(1)}$ által generált ciklikus kód, akkor a nyilvánvaló $g\big|g^{(1)}$ következtében $C^{(1)}\subseteq C$, és egy C-beli $\mathbf{c}$ akkor és csak akkor eleme $C^{(1)}$-nek, ha c-nek gyöke e, azaz ha $0=\hat{c}(e)=\sum_{i=0}^{n-1}c_ie^i=\sum_{i=0}^{n-1}c_i$. Ez azt jelenti, hogy $C^{(1)}$ paritáselemes maximális részkódja C-nek.

BCH-kódnak például e pontosan akkor gyöke, ha egy $\delta-1>i\in\mathbb{N}$-re $(\tau+i)\bmod n=0$.

Most néhány példát adunk blokk-kódra.

6.10. Példák

1. Legyen $0<k=n-1$, a kód az $\mathbb{F}_q$ fölötti k hosszúságú szavakat képezi le a szintén $\mathbb{F}_q$ fölötti n-betűs szavakba: $\mathbf{G}=(\mathbf{I}_{n-1}\ -\bar{\mathrm{e}}^{(n-1)})$, vagyis olyan $(n-1)\times n$-méretű mátrix, amelynek első $n-1$ oszlopa $n-1$-edrendű egységmátrix, az utolsó oszlopában pedig mindenütt az $\mathbb{F}_q$-beli egységelem mínusz egyszerese áll. $\mathbf{G}$ rangja $n-1$, hiszen ennél nagyobb a sorok száma miatt nem lehet, viszont első $n-1$ oszlopa lineárisan független. Ha az eredeti üzenet $a_1,\dots,a_{n-1}$, akkor a megfelelő kódszó $c_1,\dots,c_{n-1},c_n=a_1,\dots,a_{n-1},c_n$, és itt $c_n=-\sum_{i=1}^{n-1}a_i$. $\mathbf{H}=\bar{\mathrm{e}}^{(n)^T}$ az ellenőrző mátrix: a mérete $1\times n$, és $\mathbf{G}$ bármely sorával szorozva 0-t kapunk, mivel $\mathbf{G}$ minden sorában egyszer szerepel e, egyszer $-e$, a többi elem 0, és $\mathbf{H}$ a $\mathbf{G}$ bármely sorában összeadja a sor valamennyi elemét. Eszerint egy $\mathbf{c}$ vektor pontosan akkor kódszó, ha komponenseinek összege 0-t ad. A kód távolsága 2: $\mathbf{H}$ minden oszlopa egyetlen, nullától különböző elemet tartalmaz, így bármely oszlopa önmagában lineárisan független rendszert alkot, viszont minden eleme azonos, ezért bármely két oszlopa összefügg ($\mathbf{H}$-nak van legalább két oszlopa, mert $0<k=n-1<n$). Egy kód az általunk választott dekódolási sémával $\frac{d}{2}$-nél kevesebb hibát tud javítani, jelen esetben tehát 1-nél kevesebbet, a kód nem javít hibát, viszont jelez egy hibát, mivel ha egyetlen pozícióban hiba keletkezik, akkor ott c_i helyett $c_i'=c_i+d_i$ lesz $d_i\neq 0$-val, és összegzés után d_i-t kapunk, ami nem nulla, észrevesszük, hogy a kapott vektor nem kódszó. A hibajelző képesség tulajdonképpen ennél jobb: minden olyan hibát jelez a kód, ahol az egyes pozíciókban fellépő hibák összege 0-tól különbözik. Ez $q=2$ esetén azt jelenti, hogy ha páratlan számú pozícióban lép fel hiba, akkor jelez a kód, míg páros számú hiba esetén nem (az összeadást $\mathbb{F}_q$-ban végezzük, így $q=2$-nél modulo 2). A bináris esetben c_n a **paritásbit**. Gyakran ennek ellentettjével egészítik ki az eredeti szót, azaz az 1-ek számát páratlanra egészítik ki (ekkor a kód már nemlineáris: két kódszó összege páros számú 1-est tartalmaz, tehát nem kódszó). Ez utóbbi hibakorlátozást alkalmazzák általában számítógépek operatív memóriájában védelemre. Tárolás előtt minden bájtot kiegészítenek egy paritásbittel, és kiolvasáskor ellenőrzik a 9-bites adat paritását. Ha az 1-ek száma páratlan, akkor feltesszük, hogy nincs hiba, ellenkező esetben hibajelzésre kerül sor.

2. Legyen $q = 2, n = 7, k = 4$, és $\mathbf{G} = \begin{pmatrix} 1\,1\,1\,0\,0\,0\,0 \\ 1\,0\,0\,1\,1\,0\,0 \\ 0\,1\,0\,1\,0\,1\,0 \\ 1\,1\,0\,1\,0\,0\,1 \end{pmatrix}$, $\mathbf{H} = \begin{pmatrix} 1\,0\,1\,0\,1\,0\,1 \\ 0\,1\,1\,0\,0\,1\,1 \\ 0\,0\,0\,1\,1\,1\,1 \end{pmatrix}$.

$\mathbf{G}$ 3., 5. 6. és 7. oszlopa egységmátrix, ezért ezek az oszlopok lineárisan függetlenek, $\mathbf{G}$ rangja legalább 4, a négy sora lineárisan független, $\mathbf{G}$ a négydimenziós vektortérnek a 7-dimenziós vektortér egy négydimenziós alterére való leképezés mátrixa. Hasonlóan kapjuk, hogy $\mathbf{H}$ sorai lineárisan függetlenek: a számuk három, és $\mathbf{H}$ 1., 2. és 4. oszlopa lineárisan független. Kis számolással azt is megkapjuk, hogy $\mathbf{H}$ és $\mathbf{G}$ sorai páronként ortogonálisak: ha $\mathbf{H}$ első sorával szorozzuk balról $\mathbf{G}$ transzponáltját, akkor ez a szorzás a $\mathbf{G}$ 1., 3., 5. és 7. oszlopának összege lesz (modulo 2), ami a nullvektor, hasonlóan $\mathbf{H}$ második sora összeadja $\mathbf{G}$ 2., 3., 6. és 7. oszlopát, ez ismét zérus, végül a harmadik szorzásnál a 4., 5., 6. és 7. oszlopot összegezzük, ami megint 0-ra vezet. A kód távolsága 3: két vektor akkor és csak akkor lineárisan összefüggő, ha egyik a másiknak konstansszorosa, ami $\mathbb{F}_2$ felett azt jelenti, hogy vagy megegyeznek, vagy egyikük a $\mathbf{0}$-vektor, márpedig $\mathbf{H}$-ra egyik feltétel sem teljesül, bármely két oszlop $\mathbf{H}$-ban lineárisan független, a távolság legalább 3. Viszont $\mathbf{H}$ első három oszlopának összege a nullvektor, van három lineárisan összefüggő oszlop, a kód távolsága pontosan 3, a kód egy hiba javítására alkalmas. Tegyük fel, hogy $\mathbf{c}^T = c_1 \dots c_7$ kódszó, ekkor $\mathbf{Hc} = \mathbf{0}$. Ha egyetlen hiba lép fel, mondjuk az m-edik pozícióban, akkor $\mathbf{c}$ helyett egy $\mathbf{c}'$ vektort kapunk, ami $\mathbf{c}$ és $\mathbf{e}^T = \underbrace{0 \dots 0}_{m-1}\,1\,\underbrace{0 \dots 0}_{n-m}$ összegének tekinthető. Most $\mathbf{Hc}' = \mathbf{Hc} + \mathbf{He} = \mathbf{He}$, hiszen az első tag $\mathbf{0}$. $\mathbf{e}$ minden pozíciójában 0 áll az m-ediktől eltekintve, ahol viszont 1 található, ezért $\mathbf{He}$ a $\mathbf{H}$-nak m-edik oszlopát adja. Figyelmesen megnézve $\mathbf{H}$ oszlopait látható, hogy azokat mint 2-es számrendszerbeli számot olvasva (felül áll a legalacsonyabb helyiértékhez tarozó jegy) éppen az oszlop indexét kapjuk (1-től 7-ig számozva), azaz $\mathbf{Hc}'$ ilyen olvasata pontosan a hiba helyének indexét adja. A javítás tehát abból áll, hogy a $\mathbf{Hc}'$ által meghatározott indexhez tartozó pozícióban a bitet az ellentettjére módosítjuk ($\mathbb{F}_2$ esetén a hiba azt jelenti, hogy 0 helyett 1, 1 helyett 0 áll). Kevés munkával ellenőrizhető, hogy pontosan 2 hiba esetén $\mathbf{Hc}'$ biztosan nem $\mathbf{0}$, és olyan indexet ad, amely különbözik mindkét hiba helyétől, ezért most "javítás" után három hibánk lesz. Ha viszont $\mathbf{c}'$-ben legalább 3 hiba van, akkor lehet, hogy $\mathbf{Hc}' = \mathbf{0}$, így azt hisszük, hogy nem volt hiba, és lehet, hogy $\mathbf{Hc}' \neq \mathbf{0}$, és ekkor egy addig hibátlan bitet javítunk, amikor a hibák száma nő. Könnyen beláthatóan – mert a kód bináris – most nem fordulhat elő, hogy javításkor valamelyik hibás bitet javítjuk, amikor a hibák száma eggyel csökkenne, de még mindig hibás lenne az adat (viszont mi azt hinnénk, hogy már hibátlan). Ez a (**bináris**) [7,4]-paraméterű **Hamming-kód** akkor hatásos, ha kicsi a hibák valószínűsége, és egymástól függetlenek, amikoris egy kódszóban egynél több hiba felléptének valószínűsége igen csekély. A kód az $n = 2^m - 1$, $k = n - m$ választással bármely $m \in \mathbb{N}^+$-ra kiterjeszthető.

3. Nézzünk egy $q = 2$, $n = 7$, $\tau = 1$, $\delta = 3$-paraméterű BCH-kódot. $x^7 + 1$ irreducibilis faktorokra való felbontása $x^7 + 1 = (x + 1)(x^3 + x + 1)(x^3 + x^2 + 1)$ ($\mathbb{F}_2$ fölött az összeadás egybeesik a kivonással, ezért $x^7 - 1 = x^7 + 1$, továbbá 1 az $\mathbb{F}_2$ egységeleme). Egy $\mathbb{F}_2$ fölötti 7-edik primitív egységgyök vagy $x^3 + x + 1$, vagy $x^3 + x^2 + 1$ gyöke, válasszuk az előbbit, ekkor ez lesz α minimálpolinomja. Mivel $\tau = 1$ és $\delta = 3$, ezért most α-t és α^2-et kell tekintenünk. q-elemű test fölött α és α^q minimálpolinomja azonos, és most $q = 2$, ezért $g = x^3 + x + 1$, ami egy [7,4]-kódot generál. h $x^7 + 1$ másik két faktorának szorzata, azaz $h = x^4 + x^2 + x + 1$, és $h^* = x^4 + x^3 + x^2 + 1$ (most $h_0 = 1$), így

$$\mathbf{G} = \begin{pmatrix} 1\,1\,0\,1\,0\,0\,0 \\ 0\,1\,1\,0\,1\,0\,0 \\ 0\,0\,1\,1\,0\,1\,0 \\ 0\,1\,0\,1\,1\,0\,1 \end{pmatrix}, \mathbf{H} = \begin{pmatrix} 1\,0\,1\,1\,1\,0\,0 \\ 0\,1\,0\,1\,1\,1\,0 \\ 0\,0\,1\,0\,1\,1\,1 \end{pmatrix}.$$

Most az elméletből tudjuk, hogy $\mathbf{G}$ generátormátrix és $\mathbf{H}$ a kód paritásellenőrző mátrixa, továbbá $d \geq 3$. De d pontosan 3, ugyanis $\mathbf{H}$ 1., 2. és 4. oszlopa lineárisan összefügg. Ha megnézzük $\mathbf{H}$ oszlopait mint 2-es számrendszerbeli számokat (felül áll a legalacsonyabb helyiérték), akkor látjuk, hogy azok kiadják az 1, 2, 3, 4, 5, 6 és 7 számokat, hasonlóan a [7,4] bináris Hamming-kódhoz. Azt talál-

tuk, hogy ez a kód megkapható az említett Hamming-kódból a bitek pozícióinak egy permutációjával. Ez a permutáció nyilván nem változtatja meg a kód hibajavító tulajdonságait, tehát a két kód ekvivalensnek tekinthető. Ami különbség, hogy a mostani kód ciklikus, így egyetlen polinom ismeretében generálható és az ellenőrzés is elvégezhető, míg a korábban ismertetett Hamming-kód nem ciklikus. Megnézzük a kódhalmazt, mindkét oszlopban a bal oldalon a mostanit, a jobb oldalon a korábbit (vagyis vesszük a **G**-mátrixok sorainak összes lehetséges lineáris kombinációját). Az eredményt az 1. Táblázat mutatja.

A bal oldalon a 0. és a 13. vektor bármely eltoltja önmaga. Balra tolással az 1. vektorból az $1 \to 11 \to 14 \to 7 \to 8 \to 4 \to 2 \to 1$, a 3. vektorból a $3 \to 10 \to 5 \to 9 \to 15 \to 12 \to 6 \to 3$ láncot kapjuk, és ezzel kimerítettük a kódhalmazt, a kód valóban ciklikus. Ugyanakkor a jobb oldal 1. vektora egy hellyel ciklikusan balra tolva 1100001 lesz, ami nem eleme a kódnak, a korábbi Hamming-kód nem ciklikus.

0.	0000000	0000000	8.	0001101	1101001
1.	1101000	1110000	9.	1100101	0011001
2.	0110100	1001100	10.	0111001	0100101
3.	1011100	0111100	11.	1010001	1010101
4.	0011010	0101010	12.	0010111	1000011
5.	1110010	1011010	13.	1111111	0110011
6.	0101110	1100110	14.	0100011	0001111
7.	1000110	0010110	15.	1001011	1111111

1. Táblázat

Δ

7. Kódkonstrukció I.

Ebben a részben azt vizsgáljuk, hogy adott kódból vagy kódokból hogyan lehet új kódot konstruálni, és hogyan függnek az új kód paraméterei az eredeti kód(ok) paramétereitől. A konstrukciók egy része inkább csak elméleti szempontból, elméleti megfontolásoknál érdekes, más módszerek azonban a gyakorlatban is jelentősek, segítségükkel ugyanis valamilyen jó kódból kiindulva, a konkrét felhasználáshoz jobban igazodó, jó tulajdonságú kódot lehet létrehozni.

Kiterjesztés (extending). A q-elemű S szimbólumhalmaz fölötti $(n, M, d)_q$-paraméterű C kód minden kódszavát jobbról kiegészítjük egy új komponenssel. Ha az új C' kód $(n', M', d')_q$-paraméterű, akkor nyilván $n' = n + 1$, a kód mérete nem változik, tehát $M' = M$. Most még megnézzük az új kód távolságát.

Feltesszük, hogy a kód legalább kételemű. A kód távolsága nyilván nem csökken, ha az eredeti komponenseket kiegészítjük egy újjal, hiszen ha két kódszó valamelyik pozíción eltért, akkor a kiegészítés után is különbözik ezen a pozíción. Az is magától értetődő, hogy a távolság legfeljebb eggyel nőhet, ugyanis $d(\mathbf{u}u_{n+1}, \mathbf{v}v_{n+1}) = d(\mathbf{u}, \mathbf{v}) + d(u_{n+1}, v_{n+1})$, és a jobb oldali második tag 0 vagy 1, attól függően, hogy $u_{n+1} = v_{n+1}$ vagy $u_{n+1} \neq v_{n+1}$, tehát $d \leq d' \leq d + 1$. d akkor és csak akkor nem változik, ha C-ben van olyan d távolságú kódszópár, amelyeket azonos elemmel egészítettünk ki.

Az új komponenst természetesen nem „hasraütés-szerűen" választjuk, hanem egy $f: S^n \to S''$ függvénnyel határozzuk meg, vagyis $u_{n+1} = f(u_1, \dots, u_n)$. Ha most az **u**-t elküldve, a vétel helyére **v** érkezik, akkor ellenőrizzük, hogy teljesül-e a $v_{n+1} = f(v_1, \dots, v_n)$ egyenlőség. Ha nem, akkor biztosan hibás a vett szó, de azt ebből az eredményből nem lehet megállapítani, hogy melyik komponense(i) hibás(ak), és mi a hiba. Természetesen az a szerencsétlen helyzet is előfordulhat, hogy a $v_1, \dots, v_n$ jegyek mindegyike azonos az eredeti szó megfelelő komponensével, vagyis maga az eredeti üzenet hibátlanul érkezett meg, és csupán a kiegészítő ellenőrző jegy sérült.

Amennyiben a kód valamilyen algebrai struktúrára épül, akkor az f függvény speciális alakot ölt. Ha a szimbólumhalmaz az S additív Abel-csoport, akkor legyen $u_{n+1} = -\sum_{i=1}^{n} u_i + c$, ahol c az S egy rögzített eleme. A kiterjesztett kód egy kódszavát elküldve, a beérkezett **v** akkor és csak akkor kódszó, és így akkor és csak akkor tekintjük az átvitelt hibátlannak, ha $\sum_{i=1}^{n+1} v_i = c$. Amennyiben az átvitel során pontosan egy hiba lép fel, akkor az előbbi egyenlőség biztosan nem teljesül, így a kód egy hibát biztosan jelez. Két hiba esetén előfordulhat, hogy a két hiba összege éppen 0, és ekkor az ellenőrzés nem jelez hibát, ezért a kód pontosan egy-hiba jelző. Ugyanakkor a kóddal javítani nem lehet a hibát, hiszen bármely pozíción keletkezik ugyanolyan értékű hiba, az összeg azonos lesz, így a hiba helyét az ellenőrzés nem tudja meghatározni.

Lineáris kód és $c = 0$ esetén C' is lineáris, és C' generátor- és ellenőrző mátrixa

$$\mathbf{G}' = (\mathbf{G} \; -\mathbf{Ge}) \quad \mathbf{H}' = \begin{pmatrix} \mathbf{e}^T & e \\ \mathbf{H} & \mathbf{0}^{(n-k)} \end{pmatrix},$$

ahol **e** minden komponense e, és $\mathbf{0}^{(n-k)}$ az $n-k$-méretű nullvektor.

A kódkiterjesztés egyik leggyakoribb alkalmazása a bináris kódok **paritásbit**tel való kiegészítése. Ez azt jelenti, hogy ha az eredeti kódszóban az 1-ek száma páros, akkor a kódszót egy 1-gyel, ellenkező esetben egy 0-val egészítik ki, tehát a kiterjesztett kód minden kódszavában az 1-esek száma páratlan. Ez a **páratlanra való kiegészítés**, és ekkor valamennyi kódszó, és így a kód súlya is, páratlan. Hasonlóan működik a **párosra való kiegészítés**, csupán most akkor fűzünk 1-et a kódszóhoz, ha az eredeti kódszó páratlan sok 1-est tartalmazott, és ebben az esetben a nem nulla kódszavak, és ha van nem nulla kódszó, akkor a kód súlya is, páros. Az előbbit használják a számítógépek operatív

memóriájánál: mielőtt a gép kiírna a memóriába egy bájtot, páratlanra egészíti ki, és az így kapott kilencbites kódszó kerül a memóriába, kiolvasáskor pedig a gép ellenőrzi, hogy a kiolvasott bájtban valóban páratlan-e az 1-esek száma. Ha igen, akkor rendben van, ellenkező esetben hibajelzés jön létre, amely például egy megszakítást generálhat. Párosra való kiegészítést használnak viszont általában az aszinkron adatátvitelnél.

A paritásbit kiszámítása $u_{n+1} = \oplus_{i=1}^{n} u_i \oplus c$ szerint történik, ahol $c = 0$ vagy $c = 1$, és az előbbiek alapján a kód akkor és csak akkor lineáris, ha az eredeti kód lineáris és $c = 0$.

Ha egy bináris kód d távolsága páratlan, akkor a kiterjesztett kód távolsága eggyel nagyobb. Legyen ugyanis **u** és **v** két eredeti kódszó, akkor a fentebbi képlet alapján

$$u_{n+1} \oplus v_{n+1} = (\oplus_{i=1}^{n} u_i \oplus c) \oplus (\oplus_{i=1}^{n} v_i \oplus c) = \oplus_{i=1}^{n} (u_i \oplus v_i)$$
$$= \sum_{i=1}^{n} (u_i \oplus v_i) \bmod 2 = w(\mathbf{u} - \mathbf{v}) \bmod 2 = d(\mathbf{u}, \mathbf{v}) \bmod 2,$$

és ha $d(\mathbf{u}, \mathbf{v}) = d$, akkor $d(\mathbf{u}, \mathbf{v}) \bmod 2 = d \bmod 2 = 1$, a két paritásbit különböző, így a kiterjesztett kódban a két kódszó $d + 1$ helyen tér el egymástól. Ebből következően, ha létezik $(n, M, 2t + 1)_2$-paraméterű kód, akkor van $(n + 1, M, 2t + 2)_2$-paraméterű kód is.

Azt már láttuk, hogy a kiterjesztett kód az $u_{n+1} = -\sum_{i=1}^{n} u_i + c$ kiterjesztéssel egy hibát mindig képes jelezni. Bináris esetben ennél többet tud a kód, ugyanis minden olyan esetben jelez, amikor a hibák száma páratlan. Ekkor ugyanis

$$w(\mathbf{u} + \boldsymbol{\varepsilon}) = \sum_{i=1}^{n+1} (u_i \oplus \varepsilon_i) = \sum_{i=1}^{n+1} (u_i + \varepsilon_i - 2u_i\varepsilon_i) = \sum_{i=1}^{n+1} u_i + \sum_{i=1}^{n+1} \varepsilon_i - 2\sum_{i=1}^{n+1} u_i\varepsilon_i$$
$$= w(\mathbf{u}) + w(\boldsymbol{\varepsilon}) - 2\sum_{i=1}^{n+1} u_i\varepsilon_i \equiv w(\mathbf{u}) + 1\ (2),$$

és mivel korábban láttuk, hogy egy paritásbittel kiterjesztett bináris kódban minden kódszó súlya azonos paritású, és a hibás vektor súlyának paritása ezzel ellentétes, ezért észrevesszük a hibát. Ugyanakkor az előbbi levezetésből látható, hogy ha a hibák száma páros, akkor $w(\mathbf{u} + \boldsymbol{\varepsilon}) \equiv w(\mathbf{u})\ (2)$, tehát a hibát nem vesszük észre, a hiba nem jelezhető. Összefoglalva tehát, egy bináris kódot egy paritásbittel kiterjesztve, a kiterjesztett kód minden páratlan hibát jelez, de egyetlen páros hibát sem jelez.

Átszúrás (puncturing). Ez a kiterjesztés megfordítása: minden kódszóból elhagyjuk egy előre megadott, rögzített pozíción álló komponensét. Legyen az eredeti kód távolsága nagyobb, mint 1. Ekkor átszúrás után még legalább egy helyen különbözik bármely két szó, így a kódszavak száma nem csökken, és akkor nem is változik. Az átszúrt kód távolsága vagy megegyezik az eredeti kód távolságával, vagy eggyel kisebb nála, hiszen ha két kódszó valahány helyen eltért egymástól, akkor elhagyva egy pozíciót, az eltérő helyek száma biztosan nem nő, és legfeljebb eggyel kevesebb, mint volt.

Könnyen belátható, hogy ha az eredeti kód lineáris kód, akkor az új kód is ilyen. Az átszúrt kód generátormátrixát úgy kapjuk az eredeti kód generátormátrixából, hogy a mátrixnak az átszúráshoz tartozó oszlopát töröljük. Szintén törölni kell az ellenőrző mátrixból is ezt az oszlopot. Mivel a kódszavak száma nem változott, ezért nem változik a lineáris kód dimenziója, k sem. Ugyanakkor n eggyel csökkent, így ugyanennyivel csökken az ortogonális altér dimenziója, és vele együtt az ellenőrző mátrix sorainak száma. Ez azt jelenti, hogy a megfelelő oszlop törlése után a kapott mátrix sorai lineárisan összefüggőek, vagyis valamelyik sor a többi sor lineáris kombinációja (ilyen sor több is lehet). Ezt a sort (illetve egy ilyen sort) törölve kapjuk az új kód ellenőrző mátrixát.

Amennyiben egy bináris kód távolsága legalább 2, és páros, akkor egy olyan helyen átszúrva a kódot, ahol egy minimális távolságú kódszópár eltér, a kód távolsága csökken, tehát azt kaptuk, hogy

ha létezik $(n+1, M, 2t+2)_2$ kód, akkor van $(n, M, 2t+1)_2$ kód is. A kiterjesztésnél látott ellenkező irányú megállapításból tehát azt kapjuk, hogy akkor és csak akkor van $(n, M, 2t+1)_2$-paraméterű kód, ha van $(n+1, M, 2t+2)_2$-paraméterű kód. Ezt majd a kódolási korlátoknál felhasználjuk.

Növelés (augmenting). Ennél az eljárásnál a kódhoz új kódszavakat veszünk, amelynek a komponensei esetleg más szimbólumhalmazból lehetnek. Az új szavak hozzávételével a meglévő szavak távolsága nem változik, így a kód minimális távolsága – ha volt – biztosan nem nő, de hogy mennyi lesz, azt általánosságban nem tudjuk megmondani. Azt sem lehet elvileg megmondani, hogy az új kód lineáris kód lesz-e, ez ugyanis semmi korrelációt nem mutat az eredeti kóddal.

Törlés (expunging, expurgating, throwing away codewords). Ez az előbbi növelés párja: kódszavakat hagyunk el. Szűkebb halmaz minimuma nem kisebb, mint az eredeti halmazé, tehát ha marad a kódban legalább két elem, akkor távolsága nagyobb, vagy egyenlő, mint a kiinduló kód távolsága volt. A linearitásról ismét semmit nem lehet általánosságban mondani

Egy alkalmazása a konstrukciónak, amikor egy bináris kódban csak a páros súlyú kódszavakat hagyjuk meg. Ha egy bináris kód lineáris, akkor vagy minden kódszó páros súlyú, vagy pontosan a kódszavak fele ilyen tulajdonságú. Legyen ugyanis **u** és **v** két kódszó. Mivel

$$w(\mathbf{u}+\mathbf{v}) = \sum_{i=1}^{n}(u_i \oplus v_i) = \sum_{i=1}^{n}(u_i + v_i - 2u_i v_i) = \sum_{i=1}^{n} u_i + \sum_{i=1}^{n} v_i - 2\sum_{i=1}^{n} u_i v_i$$
$$= w(\mathbf{u}) + w(\mathbf{v}) - 2\sum_{i=1}^{n} u_i v_i \equiv w(\mathbf{u}) + w(\mathbf{v}) \ (2),$$

ezért a páros súlyú kódszavak részcsoportot képeznek (ezek halmaza nem üres, mert a nullvektor súlya páros). Az összefüggésből az is látszik, hogy két kódszó különbsége akkor és csak akkor eleme a részcsoportnak, ha a súlyuk paritása megegyezik, így valamennyi páratlan súlyú kódszó, ha van, azonos mellékosztályban van az előző részcsoport szerint. De egy részcsoport szerinti mellékosztályok számossága azonos, így ha van páratlan súlyú kódszó, akkor pontosan a kódszavak fele ilyen. Ez tehát azt jelenti, hogy ebben az esetben a csökkentéssel a kód mérete a felére, és a dimenziója eggyel csökkent.

Nem feltétlenül bináris, de legalább két vektort tartalmazó lineáris kód esetén egy lehetséges eljárás, hogy a kód egy valódi alterét tartjuk meg.

Rövidítés (shortening). Ez a konstrukció a csökkentés és átszúrás kombinációja. Ha adott a C kód, akkor ebből elhagyunk bizonyos kódszavakat, majd a megmaradt kódszavakat átszúrjuk egy adott pozíción. A kódszavak száma nyilván csökken, a kódhosszúság szintén (ez utóbbi eggyel). Mi a helyzet a távolsággal? A csökkentésnél nem csökken, legfeljebb nő a távolság, míg az átszúrásnál (a speciális esetektől eltekintve) legfeljebb csökken, így két ellentétes hatás érvényesül. Ha a két módosítás között semmi kapcsolat nincs, akkor nem lehet általánosságban megmondani, hogy hogyan változik a rövidített kód távolsága. De ilyen általánosan nincs is értelme a konstrukciónak, hiszen így ez nem több, mint két, egymástól független eljárás egymás utáni alkalmazása. A gyakorlatban rögzítjük a kódszavak valamely pozícióját, valamint a szimbólumhalmaz egy elemét, majd azokat a kódszavakat tartjuk meg, amelyeknek a megadott pozíción lévő komponense a megadott elemmel azonos, és végül ezen a pozíción átszúrjuk a kódot, vagyis

$$C' = \{(u_1, \dots, u_{l-1}, u_{l+1}, \dots, u_n) | (u_1, \dots, u_{l-1}, c, u_{l+1}, \dots, u_n) \in C\},$$

ahol l a kijelölt pozíció indexe, és c az S adott eleme. Most egyáltalán nem biztos, hogy csökken a kódszavak száma, de az is előfordulhat, hogy az új kód üres lesz (vagy mert az adott pozíción nem szerepel a megadott karakter, vagy mert az eredeti szóhosszúság 1 volt). Ha azonban az új kód legalább két szót tartalmaz, akkor a távolsága legalább akkora, mint a kiinduló kódé volt. Azt már mondtuk, hogy a csökkentés következtében a kód távolsága nem csökkenhet. De az átszúrás sem csökkenti most a kód távolságát, ugyanis a csökkentés után már csak olyan kódszavak maradtak, amelyek az át-

szúrás helyén megegyeztek, és így két megmaradt kódszó távolsága nem változik, ha ezt a közös szimbólumot elhagyjuk.

A rövidítés egy igen gyakran alkalmazott eljárás, a későbbiek során többször találkozunk vele. Ha egy lineáris kód súlya legalább 2, akkor a rövidített kód akkor és csak akkor lesz lineáris, ha 0-ra rövidítünk. Mivel két olyan vektor összege, amelyben ugyanazon pozíción 0 áll, valamint egy ilyen vektor konstansszorosa is hasonló tulajdonságú, ezért lineáris kódot 0-ra rövidítve ismét lineáris kódot kapunk. Most tegyük fel, hogy a rövidítést egy nullától különböző c-re végezzük, és az egyszerűség kedvéért az utolsó pozícióra történik a rövidítés. Ha $\mathbf{u}$ és $\mathbf{v}$ eleme a rövidített kódnak, akkor benne kell, hogy legyen $\mathbf{u}+\mathbf{v}$ is, vagyis az eredeti kódban benne volt $\mathbf{u}c$, $\mathbf{v}c$ és $(\mathbf{u}+\mathbf{v})c$, továbbá a linearitás következtében $\mathbf{u}c+\mathbf{v}c=(\mathbf{u}+\mathbf{v})(2c)$. Ám ekkor, ismét a linearitás miatt, az eredeti kódnak eleme $(\mathbf{u}+\mathbf{v})(2c)-(\mathbf{u}+\mathbf{v})c=\mathbf{0}c$ is, vagyis ebben az esetben az eredeti kód távolsága 1 (mert $c\neq 0$).

Lineáris kód tartalmazza a nullvektort, így bármely pozíción 0-ra rövidítve, a kapott kód nem üres. Amennyiben a rövidítés helyén minden kódszóban 0 áll, akkor a rövidítés azonos az átszúrással. Ellenekező esetben azok a kódszavak, amelyekben a rövidítés helyén 0 áll, az összeadásra nézve egy valódi részcsoportot képeznek az összes kódszó additív csoportjában (mert két ilyen kódszó különbsége is ilyen, és az ilyen kódszavak halmaza nem üres. A részcsoport szerinti egy-egy mellékosztályban pontosan azok a kódszavak vannak, amelyeknek a kijelölt pozíción azonos a komponensük, és az alaptest különböző eleméhez különböző mellékosztály tartozik, továbbá a linearitás következtében a test minden eleméhez nem üres mellékosztály tartozik. Mivel minden mellékosztályban ugyanannyi elem van, és a mellékosztályok száma q, ezért rövidítés után a kód mérete az eredeti kós méretének q-adrésze, vagyis a kód dimenziója éppen eggyel csökken. Ebből az átszúrásnál követett gondolatmenettel kapjuk, hogy a kód generátor- és ellneőrző mátrixát az eredeti mátrixokból úgy kapjuk, hogy elhagyjuk a megfelelő oszlopot, és töröljük a generátormátrix egy olyan sorát, amely lineárisan függ a többi sortól.

Hosszabítás (lengthening). Ez az eljárás a rövidítés megfordítása, vagyis egy kiterjesztés és egy növelés egymás utáni végrehajtása, így a tulajdonságai könnyen megadhatóak.

8. Kódolási korlátok

A kódolási korlátok a kód három paramétere, nevezetesen a kódszavak n hossza, a kód M mérete és d távolsága közötti kapcsolatra mutatnak rá, arra, hogy ha közülük kettőt megadunk, a harmadik már nem vehet fel tetszőleges értéket. A fő probléma az, hogy nagy kódsebességhez és kis hibavalószínűséghez a Shannon-tétel szerint hosszú kódszavakra, azaz nagy n-re, és minimális távolságú dekódolás esetén nagy kódtávolságra, tehát nagy d-re van szükség, ám ez a két feltétel ellentmondó: ha nagy a kód távolsága, akkor az összes lehetséges szónak csak kis része használható kódolásra, vagyis kicsi lesz $\frac{M}{q^n}$ és az $\mathcal{R} = n^{-1}\log_q M$ kódsebesség, ahol q a kódoló szimbólumok száma.

A továbbiakban többször lesz szükségünk egy adott szótól legfeljebb t távolságra lévő szavak számára, ahol t tetszőleges nemnegatív valós szám. Ezt az értéket $V_q(n,t)$-vel jelöljük. A V jelölés azt fejezi ki, hogy ez az érték mintegy az adott szó mint középpont körüli t sugarú gömb „térfogata". Ha adott egy $\mathbf{u} \in S^n$ szó, ahol S a szimbólumok halmaza, és $n \geq i \in \mathbb{N}$, akkor $(q-1)^i$ olyan szó van S-ben, amely rögzített i számú pozícióban különbözik $\mathbf{u}$-tól, hiszen ezen pozíciók mindegyikén S bármely eleme előfordulhat, kivéve azt az egyet, amely $\mathbf{u}$-ban az adott helyen áll. Ebből következik, hogy

$$V_q(n,t) = \sum_{i=0}^{\lfloor t \rfloor} \binom{n}{i} (q-1)^i,$$

feltéve, hogy $n \geq t \in \mathbb{R}_0^+$, míg $V_q(n,t) = q^n$, ha $n < t \in \mathbb{R}$.

Egy másik, többször hivatkozott kifejezés a kódsebesség lesz, amelyet korábban már definiáltunk: egy $(n,M)_q$-paraméterű kód sebessége $\mathcal{R} = n^{-1}\log_q M$. Innen közvetlenül kapjuk, hogy egy n-hosszúságú kódszavakból álló, $\mathcal{R}$ kódsebességű kódban a kódszavak száma $M = q^{n\mathcal{R}}$.

A továbbiakban S jelöli a kódoló szimbólumok halmazát, q az S elemeinek számát, amelyről feltesszük, hogy legalább 2, n a kódszavak hosszát, amely minimum 2, és d a kód távolságát, amely szintén legalább 1, továbbá $M > 1$, ahol M a kódszavak száma.

8.1. Definíció

Az $(n,M,d)_q$-paraméterű C kód

- **maximális**, ha nincs olyan $(n,M',d')_q$-paraméterű C' kód, amely valódi részként tartalmazza C-t, és amelyre $d' \geq d$;
- **optimális**, ha $M = \max\{M' | \exists C' : (n,M',d)_q \text{ kód}\}$, vagyis C a lehető legnagyobb méretű n-hosszúságú, d-távolságú kód. A q szimbólummal felírható, n-hosszúságú és d-távolságú optimális kód méretét $A_q(n,d)$-vel jelöljük.

Δ

Ha $\mathbf{u}$ és $\mathbf{v}$ a C két olyan eleme, amelyre $d(\mathbf{u},\mathbf{v}) = d(C)$, és $C \subseteq C'$, akkor $\mathbf{u}$ és $\mathbf{v}$ C'-ben is benne van, így $d(C') \leq d(\mathbf{u},\mathbf{v}) = d(C)$, tehát a maximális kódnál $d' \geq d$ helyett írható $d' = d$ is.

Egy optimális kód nyilván maximális is, de ez fordítva nem igaz. Legyen például

$$C = \{0000,0101,0110,1011,1100\} \subseteq \{0,1\}^4.$$

Ez egy $(4,5,2)_2$-paraméterű maximális kód. A maximalitást beláthatjuk úgy, hogy a kódban nem szereplő 4-hosszúságú bináris sorozatok mindegyikéhez van a kódban olyan szó, amely legfeljebb csak

egy helyen különbözik a kiválasztott szótól, de a következő módon is. 0000-tól legalább 2-távolságra lévő szavak legalább két 1-est tartalmaznak. Azok a pontosan két 1-est tartalmazó szavak, amelyekben a második pozíción 1 áll, benne vannak a kódban. Ha viszont egy 2-súlyú szó ezen a pozíción 0, akkor csak úgy különbözhet legalább két helyen az 1011 kódszótól, ha a három 1-esből legalább kettő 0, de ekkor az így kapott szó súlya legfeljebb 1, tehát nem lehet kódszó. Az 1111 csak 1 távolságra van 1011-től, tehát szintén nem szerepelhet a megadott kód bővítésében. Végül ha a nem kódszó **v** súlya 3, akkor a második pozícióján 1 áll, és a további három pozíció egyikén és csak egyikén 0, ám ekkor valamelyik 2-súlyú kódszótól vett távolsága 1, így ilyen **v**-vel sem bővíthető a kód. Ugyanakkor C nem optimális, hiszen például $C = \{aaaa, aabb, abab, abba, baab, baba, bbaa, bbbb\}$ is 4-hosszúságú és 2-távolságú, 2 szimbólummal felírt kód, és a kódszavak száma 8. Az ennek a fejezetnek egy későbbi részén ismertetett *Singleton*-korlát alkalmazásával belátható, hogy ez a kód optimális, vagyis két szimbólummal legfeljebb nyolc szóból állhat egy kód, ha a távolsága 2.

Ezek után rátérünk a korlátok ismertetésére. Nézzük először a „triviális" korlátokat.

Az első korlát a legkisebb távolságú kódra vonatkozik. Legyen $C = S^n$, ekkor C egy $(n, q^n)_q$-kód, hiszen $|S^n| = |S|^n = q^n$. $q > 1$ és $n \geq 1$ következtében $M = q^n \geq q > 1$, és két különböző szó távolsága legalább 1, így bármely legalább két elemből álló kód távolsága minimum 1. Ugyanakkor a teljes halmazban van két olyan szó, amely pontosan egy helyen, mondjuk az első pozíción különbözik, tehát a kód távolsága legfeljebb 1, vagyis d pontosan 1. Mivel q szimbólummal legfeljebb q^n szó írható fel, ezért C optimális n-hosszúságú, 1-távolságú kód, és

$$A_q(n, 1) = q^n.$$

Most a legnagyobb távolságú kódokat nézzük. Tegyük először fel, hogy C egy $(n, M, n)_q$-paraméterű kód. Ekkor a kódhoz tartozó bármely két különböző szó mindegyik pozícióban, tehát az elsőben is különbözik egymástól. De ilyen szó legfeljebb q darab lehet, hiszen a különböző szimbólumok száma q, tehát q-nál több szó esetén legalább kettő megegyezik ezen a pozíción. Ebből következik, hogy $A_q(n, n) \leq q$. A másik irányhoz vegyük az S halmaz tetszőleges n (nem feltétlenül különböző) permutációját, és legyen $q \geq i \in \mathbb{N}^+$-ra és $n \geq j \in \mathbb{N}^+$-ra $\pi^{(j)}(i)$ az S i-edik eleme a j-edik permutációban. Ekkor az $\mathbf{u}^{(i)^T} = \pi^{(1)}(i) \dots \pi^{(n)}(i)$ vektorok száma q, és ha a k és $l \neq k$ pozitív egészek egyike sem nagyobb q-nál, akkor bármely $1 \leq i \leq n$ egészre $\pi^{(i)}(k) \neq \pi^{(i)}(l)$, így $d\left(\mathbf{u}^{(k)}, \mathbf{u}^{(l)}\right) = n$, és a q darab $\mathbf{u}^{(j)}$-ből álló kód távolsága n, vagyis $A_q(n, n) \geq q$, így a korábbi ellenkező irányú relációval együtt

$$A_q(n, n) = q.$$

Bináris kódokra érvényes a következő korlát:

$$A_2(n, 2k + 1) = A_2(n + 1, 2k + 2).$$

Ez azért igaz, mert a kódkonstrukcióknál láttuk, hogy akkor és csak akkor létezik n-hosszúságú, $2k + 1$-távolságú bináris kód, ha van $n + 1$-hosszúságú, $2k + 2$ távolságú bináris kód, így ez igaz a legtöbb kódszóból álló megfelelő paraméterű kódokra is.

Most különböző távolságú optimális kódokat hasonlítunk össze. Ekkor

$$d_1 < d_2 \Rightarrow A_q(n, d_1) \geq A_q(n, d_2).$$

Legyen ugyanis C_2 egy $(n, M, d_2)_q$-kód. Ekkor van a kódban olyan **u** és **v** kódszó, amely pontosan d_2 helyen különbözik. Válasszunk ki ebből a d_2 pozícióból tetszőleges, de rögzített $d_2 - d_1$ helyet, és szúrjuk át a kódot ezeken a pozíciókon. Az új kódban, C-ben, az **u**-ból és **v**-ből kapott kódszó $d_2 - (d_2 - d_1) = d_1 > 0$ helyen tér el, és bármely más kódszópárban is legalább ennyi az eltérések száma, amiből következik, hogy egyrészt az új kód mérete azonos az eredeti kód méretével, másrészt az új kód távolsága d_1, így C egy $(n - (d_2 - d_1), M, d_1)_q$-kód. Most legyen u az S szimbólumhalmaz

tetszőleges eleme. Terjesszük ki C-t oly módon, hogy minden szó végére írjunk $d_2 - d_1$ darab u-t, és legyen az így kapott kód C_1. C_1-ben a kódszavak hossza n, a kódszavak száma M, és mivel valamennyi kódszót ugyanazon toldalékkal egészítettük ki, a kódszavak távolsága, tehát magának a kódnak a távolsága sem változott, így C_1 egy $(n, M, d_1)_q$-paraméterű kód, ami mutatja, hogy ha létezik $(n, M, d_2)_q$-paraméterű kód, akkor biztosan létezik $(n, M, d_1)_q$-paraméterű kód is. De $A_q(n, d_1) \geq M$, és mivel ez bármely $(n, M, d_2)_q$-kód esetén igaz, igaz akkor is, amikor C_2 optimális, vagyis amikor $M = A_q(n, d_2)$, így $A_q(n, d_1) \geq M = A_q(n, d_2)$.

A másik összehasonlításban a kódszavak hossza tér el. Ebben az esetben

$$A_q(n+1, d) \leq qA_q(n, d).$$

Válasszunk ugyanis egy $(n+1, M, d)_q$-paraméterű C kódot. A kódszavak száma M, az alkalmazott szimbólumok száma q, így van olyan $u \in S$, amely a kódszavak első betűjeként legalább $\frac{M}{q}$-szor fordul elő. Rövidítsünk az első pozíción erre az u-ra. Az új kód $(n, M', d')_q$-paraméterű, ahol $M' \geq \frac{M}{q}$ és $d' \geq d$. Ez bármely $(n+1, M, d)_q$-kód esetén igaz, így akkor is, amikor C-ben $M = A_q(n+1, d)$, tehát $A_q(n, d) \geq A_q(n, d') \geq M' \geq \frac{1}{q} A_q(n+1, d)$, és átszorzással kapjuk az állítást.

Most nézzünk további korlátokat. Ezek egyrészt alsó, másrészt felső határt adnak adott hosszúságú és távolságú kódok méretére. A felső határ azt jelenti, hogy a megadott hosszúsággal és távolsággal maximum hány kódszót tudunk kiválasztani, az azonban egyáltalán nem biztos, hogy létezik is ilyen kód, vagyis a korlátok nem adnak garanciát arra, hogy ez a maximum ténylegesen elérhető. Ugyanakkor az alsó korlát azt garantálja, hogy mindig lehet találni ennyi kódszóból álló, a megadott hosszúsággal és távolsággal felépített kódot. Elsőként egy alsó korlátot adunk.

***Varshamov-Gilbert* korlát**. Ennek két változatát szokás megadni.

1.

$$A_q(n, d) \geq \frac{q^n}{V_q(n, d-1)}.$$

Ez az első alak tetszőleges szimbólumhalmaz esetén érvényes. A korlát azt mutatja, hogy mindig kiválasztható legalább $\frac{q^n}{V_q(n,d-1)}$ szó S^n-ből úgy, hogy bármely két különböző szó távolsága minimum d. Legyen ugyanis C egy $(n, M, d)_q$-paraméterű maximális kód. A kódszavak körüli $d-1$-sugarú gömbök uniója lefedi S^n-t. Ha nem így lenne, akkor lenne olyan $\mathbf{u} \in S^n$, amelynek bármely kódszótól való távolsága legalább d. Ekkor viszont $\mathbf{u}$ hozzávehető C-hez úgy, hogy az új halmaz bármely két elemének távolsága minimum d, vagyis egy $(n, M+1, d)_q$-paraméterű, a C-t tartalmazó kódunk lenne, ami lehetetlen, hiszen C maximális kód. Ebből következik, hogy a gömbök térfogatainak összege legalább akkora – és nyilván ekkor pontosan akkora – , mint S^n térfogata, vagyis q^n. De valamennyi gömb térfogata $V_q(n, d-1)$, és összesen M gömb van, tehát a gömbök térfogatának együttes összege $MV_q(n, d-1) \geq q^n$, ahonnan átosztással kapjuk, hogy $A_q(n, d) \geq M \geq \frac{q^n}{V_q(n,d-1)}$.

2. A másik változat lineáris kódok méretére ad alsó határt, így q prímhatvány. A korlát szerint ha egy $l \in \mathbb{N}$-re

$$q^l < \frac{q^n}{V_q(n-1, d-2)},$$

ahol $2 \leq d \leq n$, akkor van $[n, l, d]_q$-paraméterű kód.

Ha q prímhatvány, akkor két alsó korlátot is kaptunk $A_q(n,d)$-re: egyrészt az 1. pontban megadott $A_q(n,d) \geq \frac{q^n}{V_q(n,d-1)}$ korlátot, másrészt ha 2.-ben $k = \max\left\{l \in \mathbb{N} \middle| q^l < \frac{q^n}{V_q(n-1,d-2)}\right\}$, akkor van $[n,k,d]_q$-kód, tehát $\left(n,q^k,d\right)_q$-paraméterű kód, hiszen az $\mathbb{F}_q$ fölötti k-dimenziós tér elemeinek száma q^k, és így $A_q(n,d) \geq q^k$. Mindenesetre $\frac{q^n}{V_q(n,d-1)} < \frac{q^n}{V_q(n-1,d-2)}$, ugyanis

$$V_q(n-1,d-2) = \sum_{i=0}^{d-2}\binom{n-1}{i}(q-1)^i < \sum_{i=0}^{d-1}\binom{n-1}{i}(q-1)^i$$
$$< \sum_{i=0}^{d-1}\binom{n}{i}(q-1)^i = V_q(n,d-1),$$

ezért ha van olyan l nemnegatív egész, amellyel $\frac{q^n}{V_q(n,d-1)} < q^l < \frac{q^n}{V_q(n-1,d-2)}$, akkor a másodikként megadott korlát szigorúbb, azaz nagyobb alsó korlátot biztosít az adott paraméterű kódok méretére. Mivel mindig van ilyen pozitív egész l (ezt nem bizonyítjuk), ezért a második korlát minden esetben jobb becslést ad (feltéve, hogy q prímhatvány!). E szerint tehát a lineáris kódra adott második feltétel nagyobb alsó korlátot ad $A_q(n,d)$-re, mint az első, és ez a korlát nem csupán azt biztosítja, hogy van ennyi kódszóból álló n- hosszúságú, d-távolságú kód a q-elemű ábécé fölött, de még lineáris kód is létezik ezekkel a paraméterekkel. Ne feledjük azonban, hogy ez csak olyan q-ra igaz, amely egy prímszám pozitív egész kitevős hatványa.

Ha egy $(n,M,d)_q$-paraméterű C kódban $M \geq \frac{q^n}{V_q(n,d-1)}$, vagyis legalább annyi kódszóból áll, amennyit a Varshamov-Gilbert korlát garantál, akkor C **kielégíti a Varshamov-Gilbert korlátot**. Amennyiben C egy kódcsalád, vagyis minden $n \in \mathbb{N}^+$-ra C_n egy $(n,M_n,d_n)_q$-paraméterű kód, és mindegyik C_n kielégíti a Varshamov-Gilbet korlátot, akkor C egy **jó kód**.

A Varshamov-Gilbert korlátot más alakban is megadhatjuk, felhasználva a kódsebességet. Ha $M \geq \frac{q^n}{V_q(n,d-1)}$, akkor $V_q(n,d-1) \geq \frac{q^n}{M} = \frac{q^n}{q^{n\mathcal{R}}} = q^{n(1-\mathcal{R})}$, vagyis bármely adott q-hoz, n-hez és d-hez létezik olyan kód, amelynek a sebességére teljesül az

$$\mathcal{R} \geq 1 - n^{-1}\log_q V_q(n,d-1)$$

egyenlőtlenség.

Áttérünk a felső korlátokra.

***Hamming*-korlát**, **gömbkitöltési korlát**:

$$A_q(n,d) \leq \frac{q^n}{V_q\left(n,\left\lfloor\frac{d-1}{2}\right\rfloor\right)}.$$

Valóban, egy d-távolságú kódban a kódszavak köré írt, $\frac{d}{2}$-nél kisebb t-sugarú gömbök páronként diszjunktak, azaz a térfogatuk összege nem haladhatja meg a teljes tér térfogatát, q^n-t. A legnagyobb ilyen sugár $\left\lfloor\frac{d-1}{2}\right\rfloor$, és mivel a kódszavak körüli azonos sugarú gömbök térfogata azonos, ezért $MV_q\left(n,\left\lfloor\frac{d-1}{2}\right\rfloor\right) \leq q^n$, ahol M a kódban lévő kódszavak száma. Ez minden $(n,M,d)_q$ kódra igaz, ezért igaz az optimális kódra is. Lineáris kód esetén $M = q^k$ egy nemnegatív egész k-val, és ekkor a Hamming-korlát alakja $V_q\left(n,\left\lfloor\frac{d-1}{2}\right\rfloor\right) \leq q^{n-k}$.

Egy $(n, M, d)_q$-paraméterű C kód **tökéletes**, **teljes** vagy **perfekt**, ha $M = \frac{q^n}{V_q\left(n, \left\lfloor \frac{d-1}{2} \right\rfloor\right)}$, illetve lineáris kód esetén ha $V_q\left(n, \left\lfloor \frac{d-1}{2} \right\rfloor\right) = q^{n-k}$. A kód tehát akkor tökéletes, ha a kódszavak száma a Hamming-korlát által megadott maximális érték. Korábban láttuk, hogy egy $[n, k, d]$-kód esetén minden olyan szó, amelynek a súlya kisebb, mint $\frac{d}{2}$, mellékosztályvezető. De a $\frac{d}{2}$-nél kisebb súlyú szavak száma megegyezik a nullvektor körüli $\left\lfloor \frac{d-1}{2} \right\rfloor$ -sugarú gömb térfogatával, $V_q\left(n, \left\lfloor \frac{d-1}{2} \right\rfloor\right)$-vel, és ha a kód tökéletes, akkor ez q^{n-k}-val, amely viszont a kód szerinti mellékosztályok száma, vagyis tökéletes kód esetén pontosan a $\frac{d}{2}$-nél kisebb súlyú vektorok mellékosztályvezetők. Ez más szavakkal azt jelenti, hogy ha a kód tökéletes, akkor a $\frac{d}{2}$-nél kisebb súlyú hibaminták és csak ezek a hibaminták javíthatóak, más szóval a kód kijavít, tehát helyesen javít minden olyan hibát, ahol a hibahelyek száma kisebb, mint $\frac{d}{2}$, és egyetlen olyan hibát sem javít helyesen, amelyben a hibák száma legalább $\frac{d}{2}$.

Páros távolságú kód nem lehet tökéletes. Legyen ugyanis $\mathbf{u}$ és $\mathbf{v}$ két olyan kódszó, amelyek távolsága pontosan d, ahol d a kód távolsága. Ha d páros, akkor van olyan $\mathbf{w}$ szó, amely mindkét kódszótól pontosan $\frac{d}{2}$ távolságra van. $\left\lfloor \frac{d-1}{2} \right\rfloor \leq \frac{d-1}{2} < \frac{d}{2}$, így $\mathbf{w}$ nincs benne sem az $\mathbf{u}$, sem a $\mathbf{v}$ kódszó körüli $\left\lfloor \frac{d-1}{2} \right\rfloor$-sugarú gömbben, és nem lehet benne egyetlen $\mathbf{c}$ kódszó körüli $\left\lfloor \frac{d-1}{2} \right\rfloor$ -sugarú gömbben sem, mert ha $\mathbf{c} \neq \mathbf{u}$, akkor $d(\mathbf{c}, \mathbf{u}) \geq d$, és $d(\mathbf{u}, \mathbf{w}) = \frac{d}{2}$ felhasználásával a háromszög-egyenlőtlenségnek a különbségre vonatkozó alakjából azt kapjuk, hogy

$$d(\mathbf{c}, \mathbf{w}) \geq |d(\mathbf{c}, \mathbf{u}) - d(\mathbf{u}, \mathbf{w})| \geq d(\mathbf{c}, \mathbf{u}) - d(\mathbf{u}, \mathbf{w}) \geq d - \frac{d}{2} = \frac{d}{2}.$$

Ez viszont azt jelenti, hogy $\mathbf{w}$ nincs benne a kódszavak körüli $\left\lfloor \frac{d-1}{2} \right\rfloor$ -sugarú gömbök uniójában, vagyis M határozottan kisebb, mint a Hamming-korlátban megadott lehetséges maximális érték, így a kód nem tökéletes.

Tökéletes kód nem sok létezik, ami érthető, hiszen a tökéletesség azt jelenti, hogy a teljes tér egyrétűen lefedhető azonos sugarú gömbökkel. Mivel kevés tökéletes kód van, ezért érdekesek és fontosak az úgynevezett **kváziperfekt** kódok, amelyekre $\frac{q^n}{V_q\left(n, \left\lfloor \frac{d+1}{2} \right\rfloor\right)} \leq M < \frac{q^n}{V_q\left(n, \left\lfloor \frac{d-1}{2} \right\rfloor\right)}$, vagyis amelyeknél a kódszavak körüli $\left\lfloor \frac{d-1}{2} \right\rfloor$-sugarú gömbök nem tartalmaznak valamennyi szót, de az 1-gyel nagyobb sugarú gömbök már igen. Lineáris kódnál ez azt jelenti, hogy valamennyi mellékosztályvezető legfeljebb $\left\lfloor \frac{d+1}{2} \right\rfloor$ súlyú, azaz javítható minden $\frac{d}{2}$-nél kisebb súlyú hiba, de nem javítható egyetlen olyan hiba sem, amelyben a hibás helyek száma nagyobb $\frac{d+1}{2}$-nél.

Mint már mondtuk, nem sok tökéletes kód van. Tökéletes minden Hamming-kód, továbbá a **Golay-kódok** fele. A Hamming-kódok olyan $[n, k, d]_q$-paraméterű kódok, ahol $1 < r \in \mathbb{N}$-re $n = \frac{q^r - 1}{q - 1}$, $k = n - r$ és $d = 3$. Ekkor

$$M \sum_{i=0}^{\left\lfloor \frac{d-1}{2} \right\rfloor} \binom{n}{i} (q-1)^i = q^{n-r} \sum_{i=0}^{1} \binom{n}{i} (q-1)^i = q^{n-r} \left(1 + \frac{q^r - 1}{q - 1}(q-1)\right) = q^n,$$

a kód valóban tökéletes. Négy Golay-kód van, a $[24,12,8]_2$-paraméterű G_{24} és a $[12,6,6]_3$-paraméterű G_{12}, valamint az előzőekből átszúrással kapott $[23,12,7]_2$-paraméterű G_{23} és $[11,6,5]_3$-paraméterű G_{11}. $V_2\left(23, \left\lfloor \frac{7-1}{2} \right\rfloor\right) = \sum_{i=0}^{3} \binom{23}{i} = 2048 = 2^{23-12}$ és $V_3\left(11, \left\lfloor \frac{5-1}{2} \right\rfloor\right) = \sum_{i=0}^{2} \binom{11}{i} 2^i = 243 = 3^{11-6}$, így a két átszúrt kód tökéletes.

A Hamming-korlátot is átírhatjuk olyan alakba, amelyben a méret helyett a kódsebesség áll. A megfelelő átalakítás után kapjuk, hogy bármely kódban

$$\mathcal{R} \leq 1 - n^{-1} \log_q V_q \left(n, \left\lfloor \frac{d-1}{2} \right\rfloor\right).$$

***Singleton*-korlát**. Legyen C egy $(n, M, d)_q$-paraméterű kód. Ha ezt átszúrjuk $d-1$ helyen, akkor még mindig legalább egy helyen különbözik az eredeti kód bármely két kódszava, vagyis átszúrás után is M különböző kódszavunk lesz, és a kódszavak hossza $n-d+1$. De a q szimbólummal felírható $n-d+1$ hosszúságú szavak száma q^{n-d+1}, vagyis legfeljebb ennyi különböző kódszó lehet, ahonnan $M \leq q^{n-d+1}$, és mivel ez bármely $(n, M, d)_q$-kódra igaz, ezért

$$A_q(n, d) \leq q^{n-d+1}.$$

A 8.1. Definíció után a 49. oldalon megadott példában a második kód $(4,8,2)_2$-paraméterű volt, és $2^{4-2+1} = 8$, vagyis a kód valóban optimális.

Lineáris kód esetén a Singleton-korlátot az $M = q^k$ összefüggéssel kapjuk: $q^k \leq q^{n-d+1}$, vagyis bármely $[n, k, d]_q$-kódban

$$k \leq n - d + 1,$$

vagy átrendezés után

$$d \leq n - k + 1.$$

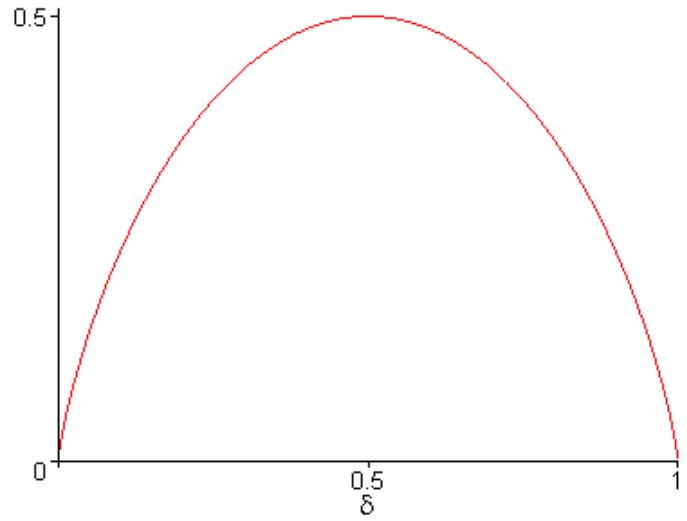
Ezt a korlátot másként is megkaphatjuk: ha a kód $[n, k, d]_q$-paraméterű, akkor az ellenőrző mátrixában bármely $d-1$ oszlop lineárisan független, vagyis a mátrixban van $d-1$ lineárisan független oszlop, a mátrix rangja legalább $d-1$. De a rang nem lehet nagyobb a sorok számánál, jelen esetben $n-k$-nál, így $d-1 \leq n-k$, ahonnan $d \leq n-k+1$. Az olyan lineáris kódot, amelyben a Singleton-korlát egyenlőséggel teljesül, vagyis amelyben $d = n-k+1$, **maximális távolságú kód**nak, vagy **MDS**-kódnak neveznek.

Most vezessük be a $\delta = \frac{d}{n}$ jelölést. Mivel innen $d = \delta n$, ezért $A_q(n, d) = A_q(n, \delta n)$, vagyis ez az n-től és δ-tól függő kifejezés, jelöljük $A_q^*(n, \delta)$-val. A Shannon-tételből tudjuk, hogy nagy kódsebességet és kis dekódolási hibát hosszú kódokkal lehet megvalósítani, ezért érdekes és fontos $A_q^*(n, \delta)$ aszimptotikus viselkedése. Ez indokolja, hogy bevezessük az $a_q(\delta) = \overline{\lim}_{n\to\infty}\left(n^{-1} \log_q A_q^*(n, \delta)\right)$ függvényt. Mivel $A_q^*(n, \delta)$ egy kód mérete, ezért $n^{-1} \log_q A_q^*(n, \delta)$ kódsebesség, azaz $a_q(\delta)$ a δ függvényében megadja az elérhető kódsebességek felső határát. Kérdéses még δ szerepe. Tudjuk, hogy minimális távolságú dekódolás esetén a javítható hibák száma d-vel arányos, így $\delta = \frac{d}{n}$ lényegében véve az n-hosszúságú kódszóban fellépő javítható hibák arányát fejezi ki. Ez azt jelenti, hogy $a_q(\delta)$ a javítható hibaarány függvényében elérhető maximális kódsebességet adja meg. Az $a_q(\delta)$-ra az alábbiakban megadott kifejezéseket **aszimptotikus korlát**oknak nevezzük.

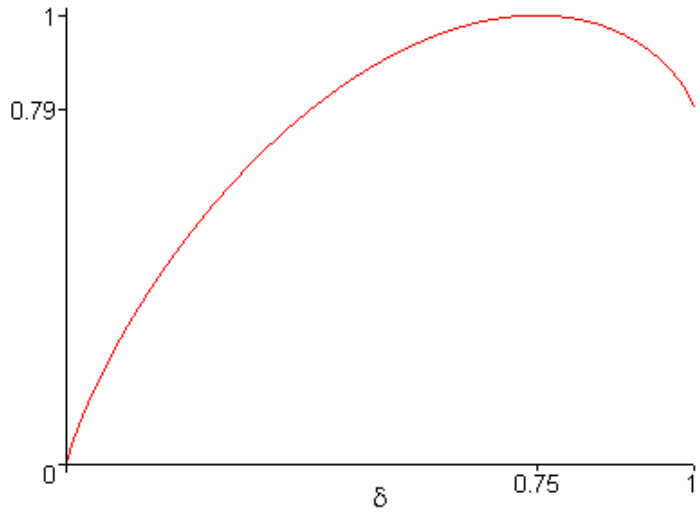
Mielőtt rátérnénk az aszimptotikus korlátok ismertetésére, ismertetünk néhány ezzel kapcsolatos dolgot. Korábban már volt szó az entrópiáról: ha Ω egy n elemi eseményből álló eseménytér, és az i-edik esemény bekövetkezésének valószínűsége p_i, akkor $I_i = -\log_r p_i$ ezen esemény egyedi információja, ahol r tetszőleges, 1-nél nagyobb valós szám, és ha $r = 2$, akkor az információ egysége a bit. $H_r(p_1, \dots, p_n) = -\sum_{i=1}^n p_i \log_r p_i$ az eseménytér átlagos információtartalma, az entrópia. Láthatóan ez a függvény a pozitív p_i-ken értelmezett, ám kiterjeszthető az értelmezés arra az esetre is, amikor egyes i indexre p_i értéke 0, ugyanis $x \log x$-nek a 0-ban van jobb oldali határértéke, nevezetesen a 0. Legyen tehát definíciószerűen $I = 0$, ha $p = 0$, így H_r már a nemnegatív valós argumentumokon értelmezett, és természetesen $\sum_{i=1}^n p_i = 1$. Amennyiben $n = 2$, akkor az előbbi feltétel azt jelenti, hogy

$p_2 = 1 - p_1$, vagyis p_1 helyett egyszerűen p-t írva az entrópia ebben az esetben egyetlen változótól, p-től függ, és ekkor azt írjuk, hogy $H_r(p)$, ahol p 1-nél nem nagyobb nemnegatív valós szám. Nézzük meg ezt a függvényt. A definíció alapján $H_r(p) = -p\log_r p - (1-p)\log_r(1-p)$, és szintén a definíció miatt $H_r(0) = 1 = H_r(1)$, továbbá az is könnyen látható, hogy $H_r(p) = H_r(1-p)$. Ez másként írva $H_r\left(\frac{1}{2} + x\right) = H_r\left(\frac{1}{2} - x\right)$, vagyis a függvény szimmetrikus az abszcisszatengely $\frac{1}{2}$ pontján átmenő, és az ordinátatengellyel párhuzamos egyenesre. A függvény a $(0,1)$ intervallumban deriválható, és a deriváltja $H_r'(p) = -\log_r p + \log_r(1-p) = \log_r \frac{1-p}{p}$. A derivált akkor és csak akkor 0, amikor a logaritmusfüggvény argumentuma 1, vagyis amikor $\frac{1-p}{p} = 1$, vagyis, ha $p = \frac{1}{2}$. Mivel 0-ban és 1-ben a függvény értéke 0, és az intervallum többi pontjában pozitív, ezért a $p = \frac{1}{2}$ pontban maximuma van, továbbá a maximum értéke $H_r\left(\frac{1}{2}\right) = \log_r 2 = \frac{1}{\log_2 r}$, és ez 1, ha $r = 2$, míg $r > 2$ esetén 1-nél kisebb. A 0-ban a deriváltfüggvény jobb oldali határértéke $+\infty$, és a szimmetria miatt 1-ben a bal oldali határérték $-\infty$. A második derivált $H_r''(p) = -\frac{1}{\ln r}\frac{1}{p(1-p)}$, és ez $(0,1)$-ben negatív, így a függvény alulról konkáv.

Természetesen a H_r függvény argumentuma tetszőleges $1 \geq \delta \in \mathcal{R}_0^+$ érték lehet, nem kell, hogy valószínűség legyen. Ezek után megrajzolható a függvény, amelyet $r = 4$-re az 5. ábra mutat.



5. ábra



6. ábra

Szükségünk lesz az előbbi entrópiafüggvényből származtatott H_r^* függvényre is: ez szintén az 1-nél nem nagyobb nemnegatív valós számokon értelmezett, és $H_r^*(\delta) = H_r(\delta) + \delta \log_r(r-1)$, ahol $r > 1$. Rögtön látni, hogy $H_2^* = H_2$, továbbá $H_r^*(0) = 0$, $H_r^*(1) = \log_r(r-1)$, és $H_r^*(\delta) > 0$, ha $1 > \delta \in \mathbb{R}^+$. A derivált $H_r^{*\prime}(\delta) = H_r'(\delta) + \log_r(r-1) = \log_r\left(\frac{1-\delta}{\delta}(r-1)\right)$, így a $\delta = 1 - \frac{1}{r} = \vartheta$ pontban maximum van, és a maximum értéke 1. A 0-ban a derivált jobb oldali határértéke ismét $+\infty$, és 1-ben a bal oldali határérték $-\infty$. $H_r^{*\prime\prime}(\delta) = -\frac{1}{\ln r}\frac{1}{\delta(1-\delta)} = H_r''(\delta)$ a $0 < \delta < 1$ intervallumban negatív, a függvény alulról konkáv. Az $r = 4$ esetre a függvényt a 6. ábra mutatja.

Használni fogjuk még az alábbi összefüggést, amelyet nem bizonyítunk: ha $0 \le \delta \le \vartheta$, akkor

$$\lim_{n\to\infty}(n^{-1}\log_r V_r(n, \lfloor \delta n \rfloor)) = H_r^*(\delta).$$

A továbbiakban δ tetszőleges nemnegatív valós szám lehet, így általában δn sem egész szám, és esetleg 0, ezért legyen $A_q(n,0) = 1$, továbbá $A_q(n,t) = A_q(n, \lceil t \rceil)$, ahol $t \in \mathbb{R}_0^+$. Ezek után lássuk az aszimptotikus korlátokat.

A Varshamov-Gilbert korlátból egy aszimptotikus alsó korlátot kapunk: ha $0 \le \delta \le \vartheta$, akkor

$$a_q(\delta) \ge 1 - H_q^*(\delta).$$

A definíció és a Varshamov-Gilbert korlát első alakja alapján ugyanis

$$A_q^*(n,\delta) = A_q(n,\delta n) = A_q(n,d) \ge \frac{q^n}{V_q(n,d-1)} = \frac{q^n}{V_q(n,\delta n - 1)} \ge \frac{q^n}{V_q(n,\lfloor \delta n \rfloor)},$$

és ha $0 \le \delta \le \vartheta$, akkor

$$\begin{aligned} a_q(\delta) &= \overline{\lim_{n\to\infty}}\left(n^{-1}\log_q A_q^*(n,\delta)\right) \ge \overline{\lim_{n\to\infty}}\left(n^{-1}\log_q \frac{q^n}{V_q(n,\lfloor \delta n \rfloor)}\right) \\ &\ge \lim_{n\to\infty}\left(n^{-1}\log_q \frac{q^n}{V_q(n,\lfloor \delta n \rfloor)}\right) = 1 - \lim_{n\to\infty}\left(n^{-1}\log_q V_q(n,\lfloor \delta n \rfloor)\right) = 1 - H_q^*(\delta). \end{aligned}$$

A Hamming-korlát szerint $A_q(n,d) \le \frac{q^n}{V_q\left(n,\left\lfloor \frac{d-1}{2} \right\rfloor\right)}$, és ebből némi ügyeskedéssel kijön, hogy ha $0 \le \frac{\delta}{2} \le \vartheta$, akkor

$$a_q(\delta) \le 1 - H_q^*\left(\frac{\delta}{2}\right).$$

De $\vartheta \ge \frac{1}{2}$, ezért ez a korlát a teljes $0 \le \delta \le 1$ intervallumban érvényes.

Az aszimptotikus Singleton-korlát könnyen megkapható. A korlát szerint $A_q(n,d) \le q^{n-d+1}$, így $A_q^*(n,\delta) \le q^{n-\delta n+1}$, és innen $n^{-1}\log_q A_q^*(n,\delta) \le 1 - \delta + \frac{1}{n}$. A határértékekre áttérve

$$a_q(\delta) \le 1 - \delta, \text{ feltéve, hogy } 0 \le \delta \le 1.$$

Nem foglalkoztunk a Plotkin-korláttal. Ez a korlát csak akkor alkalmazható, ha $\vartheta < \delta \le 1$, és ekkor $A_q(n,d) \le \frac{d}{d-\vartheta n}$, vagyis $A_q^*(n,\delta) \le \frac{\delta}{\delta-\vartheta}$. A jobb oldali kifejezés értéke egy n-től független pozitív valós szám, így a logaritmusa sem függ n-től. Ezt n-nel osztva a kapott érték a 0-hoz tart, miközben n minden határon túl nő, így azt kaptuk, hogy a $\vartheta < \delta \le 1$ tartományban $a_q(\delta) = 0$. Nézzük

ezek után a $0 \le \delta < \vartheta$ értékeket. Ekkor a Plotkin-korlát közvetlenül nem alkalmazható, ám némi ügyeskedéssel, amit nem részletezünk, eredményre jutunk, amely szerint $a_q(\delta) \le 1 - \frac{\delta}{\vartheta}$. Ennek a kifejezésnek létezik a határértéke, amikor δ tart ϑ -hoz, és ez a határérték 0, amely megegyezik a korábban a $\vartheta < \delta \le 1$ esetre kapott érték jobb oldali határértékével, vagyis

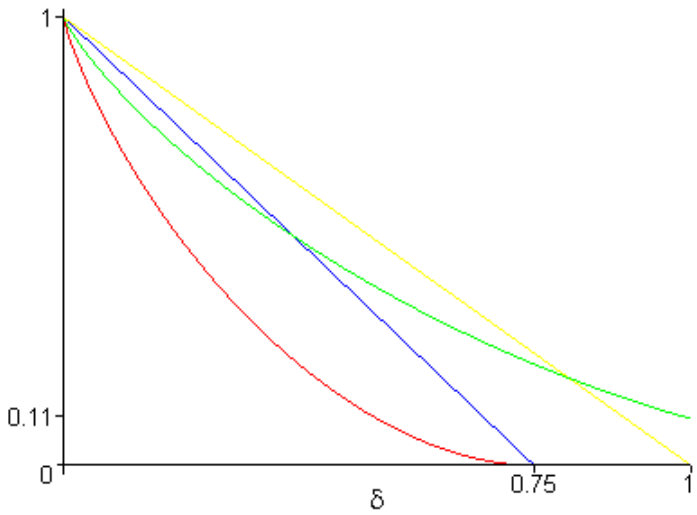
$$a_q(\delta)\begin{cases} \le 1 - \dfrac{\delta}{\vartheta}, \text{ha } 0 \le \delta < \vartheta \\ = 0, \qquad \text{ha } \vartheta \le \delta \le 1. \end{cases}$$

A fentiekben megadott aszimptotikus korlátokat $q = 4$ esetén a 7. ábra mutatja. Az alsó görbe a Varshamov-Gilbert korlát, a felső egyenes a Singleton-korlát, a másik egyenes adja a Plotkin-korlátot, és a negyedik görbe a Hamming-korlát. Az ábra is mutatja, hogy miért van szükség több különböző korlátra: különböző tartományban más és más korlát ad szigorúbb feltételt.

A megismert korlátok alkalmazására lássunk egy példát. Legyen $q = 2$, $n = 13$ és $d = 5$.

Mivel a kód bináris, és d páratlan, ezért $A_2(13,5) = A_2(14,6)$, így mindkettőt meghatározzuk, és közülük az erősebbet választjuk.

Az $A_q(n+1,d) \le qA_q(n,d)$ szabály alkalmazásával $A_2(13,5) \le 2^8 A_2(5,5) = 2^8 \cdot 2 = 512$, és hasonlóan kapjuk, hogy $A_2(14,6) \le 2^8 A_2(6,6) = 512$.



7. ábra

Most nézzük a Varshamov-Gilbert korlátokat. 2 prímhatvány, alkalmazható a lineáris kódokra adott erősebb alak. $2^4 = 16 < \frac{q^n}{V_q(n-1,d-2)} = \frac{2^{13}}{\sum_{i=0}^{3}\binom{12}{i}} = \frac{8192}{299} \le 32 = 2^5$, és a 14-hosszúságú kódokra hasonló számítással $8 < \frac{2^{14}}{\sum_{i=0}^{4}\binom{13}{i}} \le 16$, tehát $A_2(13,5) \ge 16$, és az is igaz, hogy van $[13,4,5]_2$ kód.

A Singleton-korlátból $A_2(13,5) \le 2^{13-5+1} = 512$, és ugyanezt kapjuk $A_2(14,6)$-ra, továbbá az 5-távolságú, 13-hosszúságú lineáris kódok legfeljebb 9-dimenziósak.

A Hamming-korláttal $A_q(n,d) \le \frac{q^n}{V_q\left(n,\left\lfloor\frac{d-1}{2}\right\rfloor\right)}$, ami most $A_2(13,5) \le \frac{2^{13}}{\sum_{i=0}^{2}\binom{13}{i}} = \frac{8192}{92} \approx 89{,}04$ és $A_2(14,6) \le \frac{2^{14}}{\sum_{i=0}^{2}\binom{14}{i}} = \frac{16384}{106} \approx 154{,}57$, vagyis innen $A_2(13,5) \le 89$.

A Plotkin-korlát nem alkalmazható közvetlenül, hiszen $\vartheta = 1 - \frac{1}{2} = \frac{1}{2}$, és 5 nem nagyobb, mint $\frac{1}{2} \cdot 13$, illetve 6 is kisebb 7-nél. Tudjuk azonban, hogy $2^4 A_2(9,5) \ge A_2(13,5)$, és 5 nagyobb, mint 9

fele, így alkalmazható a Plotkin-korlát. E szerint $A_2(9,5) \leq 2\left\lfloor \frac{5}{2\cdot 5-9} \right\rfloor = 10$, és innen $A_2(13,5) \leq 160$. A másik kódnál elég három pozícióval csökkenteni a kód hosszúságát. Most $2^3 A_2(11,6) \geq A_2(14,6)$ és $A_2(11,6) \leq 2\left\lfloor \frac{6}{2\cdot 6-11} \right\rfloor = 12$, tehát $A_2(14,6) \leq 96$, és ez adja a szigorúbb korlátot.

Végeredményként $16 \leq A_2(13,5) \leq 89$, továbbá lineáris kódokra $4 \leq k \leq \log_2 89 = 6{,}48$, vagyis $4 \leq k \leq 6$.

9. Reed-Solomon kódok

Felidézzük a BCH-kódokat. Adott a q-hoz relatív prím n, a τ egész és a 2-nél nem kisebb, de n-nél nem nagyobb δ egész. Ha α egy $\mathbb{F}_q$ fölötti primitív n-edik gyök, és $m_{\alpha^i}^{(\mathbb{F}_q)}$ az α^i $\mathbb{F}_q$ fölötti minimálpolinomja, akkor a $g = \text{lkkt}\left\{m_{\alpha^{\tau+i}}^{(\mathbb{F}_q)} \middle| \delta - 1 > i \in \mathbb{N}\right\}$ polinom által generált ciklikus kód $(n, \tau, \delta)_q$-paraméterű BCH-kód. Ez a kód $[n, k, d]_q$-paraméterű kód, ahol $k = n - \deg(g)$, és $d \geq \delta$, feltéve, hogy $k > 0$.

9.1. Definíció

Az $(n, \tau, \delta)_q$-paraméterű BCH-kód **Reed-Solomon kód**, ha $n | q - 1$.

Δ

Mivel $n | q - 1$, ezért n és q relatív prím, létezik primitív n-edik egységgyök $\mathbb{F}_q$ fölött. Az $\mathbb{F}_q$ fölötti primitív n-edik egységgyök eleme $\mathbb{F}_q$-nak, így α^i $\mathbb{F}_q$ fölötti minimálpolinomja $x - \alpha^i$, tehát a kód generátorpolinomja $g = \prod_{i=0}^{\delta-2}(x - \alpha^{\tau+i})$. Ekkor $k = n - \deg(g) = n - (\delta - 1) = n - \delta + 1$, és a Singleton-korlát valamint a BCH-kódok távolsága alapján $d \geq \delta = n - k + 1 \geq d$-ből kapjuk, hogy $d = n - k + 1$, vagyis igaz az alábbi tétel.

9.2. Tétel

A Reed-Solomon kódok MDS-kódok.

Δ

A Reed-Solomon kód ellenőrző polinomja $h = \frac{x^n - e}{g} = \frac{\prod_{j=0}^{n-1}(x-\alpha^j)}{\prod_{i=0}^{\delta-2}(x-\alpha^{\tau+i})} = \prod_{i=\delta-1}^{n-1}(x - \alpha^{\tau+i})$, és a duális kód generátorpolinomja $g^{(D)} = h_0^{-1}h^* = \prod_{i=\delta-1}^{n-1}(x - \alpha^{-(\tau+i)})$ (mert $h_0^{-1}h^*$ főpolinom, és reciprok polinom gyökei a polinom nem nulla gyökeinek inverzei). Ez szintén egy Reed-Solomon kód generátorpolinomja, ami igazolja a következő tételt.

9.3. Tétel

Reed-Solomon kód duálisa is Reed-Solomon kód.

Δ

Azt tudjuk, hogy ciklikus kód duálisa ciklikus, most pedig azt láttuk, hogy Reed-Solomon kód duálisa is Reed-Solomon kód. Mivel minden BCH-kód ciklikus, ezért egy BCH-kód duálisa is ciklikus, az azonban általában nem igaz, hogy BCH-kód duálisa BCH-kód. Ha például $n = 8$, $q = 3$, $\tau = 1$ és $\delta = 3$, akkor a kód gyökeinek kitevői egyrészt 1 és 2, másrészt az α és α^2 minimálpolinomjai gyökeinek kitevői. α^j minimálpolinomjának akkor és csak akkor gyöke α^k, ha $k \equiv jq^l\ (n)$ egy alkalmas l kitevővel, vagyis α-hoz tartozik még az 1 kitevőn kívül $3 \cdot 1 = 3$, és más már nem, mert $3 \cdot 3 = 9 \equiv 1\ (8)$, és hasonlóan 2-höz tartozik 2 és 6, mert 18 kongruens 2-vel modulo 8. Ekkor a duális kód gyökeinek kitevői 0, 4, 5 és 7. A 0-hoz nem tartozik más gyök, és hasonlóan a 4 is egyedül áll, míg az 5. és 7. hatvány minimálpolinomja azonos. Ha a duális kód BCH-kód, akkor kell, hogy legyenek egymás utáni kitevőhöz tartozó gyökei úgy, hogy az ezen gyökökhöz tartozó minimálpolinomok gyökei kiadják az előbb felsorolt valamennyi gyököt. Ezek szerint minden minimálpolinomból legalább egy gyöknek szerepelnie kell a sorozatban, tehát a 0. és 4., továbbá az 5. és a 7. legalább egyike (a 0. és a 7. gyök szomszédosak, mert a 8. hatvány azonos a 0. hatvánnyal!), de bárhogy is vá-

lasztunk ki a megadott módon hármat, vagy mind a négyet, ezek a kiválasztott gyökök nem alkotnak hézagmentes sorozatot, tehát a duális kód nem lehet BCH-kód.

Mint azt a BCH-kódoknál láttuk, ha egy $\delta - 1 > i \in \mathbb{N}$ -re $(\tau + i) \bmod n = 0$, akkor e gyöke a kódnak, vagyis ekkor a kód egy paritáselemes kód.

Most ismét emlékeztetünk a ciklikus kódokra. Ha a kód gyökei az α_i-k, ahol $l > i \in \mathbb{N}$, és $\widetilde{\mathbf{H}}$ az a mátrix, amelyben az $l > i \in \mathbb{N}$, $n > j \in \mathbb{N}$ indexekre $\widetilde{H}_{i,j} = \alpha_i^j$, akkor az $\mathbf{u} \in \mathbb{F}_q^n$ vektor akkor és csak akkor eleme a kódnak, ha $\widetilde{\mathbf{H}}\mathbf{u} = \mathbf{0}$, jóllehet $\widetilde{\mathbf{H}}$ általában nem a kód ellenőrző mátrixa. Ha azonban a kód egy Reed-Solomon kód, akkor az $\alpha_i = \alpha^{\tau+i}$ gyök eleme az alaptestnek, így $\widetilde{\mathbf{H}}$ egy $\mathbb{F}_q$ fölötti mátrix, továbbá $l = n - k$, vagyis most $\widetilde{\mathbf{H}}$ a kód ellenőrző mátrixa, $\widetilde{\mathbf{H}} = \mathbf{H}$, és az $n - k > i \in \mathbb{N}$, $n > j \in \mathbb{N}$ indexpárokra $H_{i,j} = \left(\alpha^{\tau+i}\right)^j$.

Legyen $\mathbf{G}$ $k \times n$-méretű mátrix, ahol $G_{i,j} = (ne)^{-1}\left(\alpha^{-(\tau+\delta-1+i)}\right)^j = (ne)^{-1}\left(\alpha^{-(\tau-k+i)}\right)^j$. Ekkor könnyen ellenőrizhető, hogy $\mathbf{H}\mathbf{G}^T = \mathbf{0}$, továbbá $(ne)\mathbf{G}$ első k oszlopa a páronként különböző $\alpha^{-(\tau-k+i)}$-vel generált Vandermonde-típusú mátrix, így $\mathbf{G}$ rangja k, tehát $\mathbf{G}$ a kód generátormátrixa. Ez azt jelenti, hogy a kódszó j-edik komponense

$$c_j = (ne)^{-1}\sum_{i=0}^{k-1} C_i\left(\alpha^{-j}\right)^{\tau-k+i} = (ne)^{-1}\left(\alpha^{-(\tau-k)}\right)^j \hat{f}\left(\alpha^{-j}\right),$$

ahol $f = \sum_{i=0}^{k-1} C_i x^i$, és $C_0 \dots C_{k-1}$ a kódolandó üzenet. Amennyiben $(\tau - k) \bmod n = 0$ és $n = q - 1$, akkor $\alpha^{-(\tau-k)} = e$, $(ne)^{-1} = (-e)^{-1} = -e$, és ekkor a C üzenethez tartozó kódszó j-edik komponense $c_j = -\hat{f}\left(\alpha^{-j}\right)$.

A Reed-Solomon kódokat kiterjedten használják olyan helyeken, ahol a véletlen hibákon kívül úgynevezett csomós hibák előfordulása is gyakori. A csomós hibára az jellemző, hogy ha valahol fellép egy hiba, akkor a környezetében lévő további jegyek meghibásodása is valószínű. Gondoljunk például egy megkarcolt lemezre, egy kávéval leöntött CD-re, vagy például arra, hogy ha 100 Mbit/s-os sebességgel viszünk át adatokat (nem üvegszál-kábelen), vagyis egy-egy jel időtartama 10 ns, akkor egy légköri elektromos kisülés, amelynek az időtartama ennél lényegesen nagyobb, sok egymás utáni jel értékét változtatja meg. Az ilyen helyzetekre definiáljuk a hibacsomót.

9.4. Definíció

Egy l-hosszúságú hibacsomó egy olyan jelsorozat, amelynek az első és utolsó eleme nem 0 (tehát lehetséges, hogy a közbülső helyeken bizonyos jelek hibátlanok).

Δ

A hibacsomókat is javító egyes kódtípusokkal foglalkozunk a következő részben.

Mivel a k-dimenziós Reed-Solomon kód generátormátrixának bármely k oszlopa lineárisan független, ezért a kódot tetszőleges l pozícióján 0-ra rövidítve egy $k - l$-dimenziós, $n - l$ szóhosszúságú MDS-kódot kapunk, feltéve, hogy $l < k$. Az így kapott kódot **rövidített Reed-Solomon kód**nak nevezzük, és a Reed-Solomon kódokra kidolgozott bármely dekódolási eljárással dekódolhatjuk, ha a vett szót a kihagyott pozíciókon 0-val egészítjük ki. Ugyanezen okból a Reed-Solomon kód jól alkalmazható olyan esetekben is, amikor várhatóan bizonyos hibák helye ismert. Igazolható, hogy MDS-kód rövidítésével ismét MDS-kódot kapunk, így a rövidített Reed-Solomon kód is MDS-kód.

A Reed-Solomon kódok felhasználhatóak a kriptográfiában titokmegosztásra. Titokmegosztásnál n résztvevő mindegyike rendelkezik egy olyan adattal, amelyből akkor lehet megismerni a titkos

adatot, ha legalább k résztvevő adata rendelkezésre áll. A kód tulajdonságainak köszönhetően a segítségével megosztott titkot akkor is helyesen kapjuk vissza, ha néhányan hamis adatot szolgáltatnak.

A CD-ken (mind a zenei, mind az adatokat tartalmazó lemezeken) rövidített Reed-Solomon kódokat alkalmaznak, de egyrészt a következő fejezetben tárgyalt direkt szorzat kód formájában, másrészt a blokk-kódnál bonyolultabb konvolúciós kód részeként.

10. Kódkonstrukció II.

Az alábbiakban három konstrukciós eljárást ismertetünk.

Átfűzéses kód.

Adott egy $(n, M, d)_q$-paraméterű C kód, valamint a λ pozitív egész szám. A C tetszőleges λ kódszavát egymás alá írva, majd oszlopfolytonosan kiolvasva kapjuk az új kód egy kódszavát:

$$\begin{pmatrix} u_0^{(0)} & \cdots & u_j^{(0)} & \cdots & u_{n-1}^{(0)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ u_0^{(i)} & \cdots & u_j^{(i)} & \cdots & u_{n-1}^{(i)} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ u_0^{(\lambda-1)} & \cdots & u_j^{(\lambda-1)} & \cdots & u_{n-1}^{(\lambda-1)} \end{pmatrix} \rightarrow$$

$$\rightarrow \mathbf{u}^T = u_0^{(0)} \ldots u_0^{(i)} \ldots u_0^{(\lambda-1)} \ldots u_j^{(0)} \ldots u_j^{(i)} \ldots u_j^{(\lambda-1)} \ldots u_{n-1}^{(0)} \ldots u_{n-1}^{(i)} \ldots u_{n-1}^{(\lambda-1)},$$

vagyis $u_{\lambda j+i} = u_j^{(i)}$, ahol $\lambda > i \in \mathbb{N}$ és $n > j \in \mathbb{N}$. Ha a kapott C' kód paraméterei $(n', M', d')_{q'}$, akkor az rögtön látszik, hogy $n' = \lambda n$ és $q' = q$, továbbá az is világos, hogy $M' = M^\lambda$, hiszen a λ sorba egymástól függetlenül a C bármely eleme választható. Ebből az is következik, hogy

$$\mathcal{R}' = \frac{1}{n'} \log_{q'} M' = \frac{1}{\lambda n} \log_q M^\lambda = \frac{1}{n} \log_q M = \mathcal{R},$$

vagyis az új kód sebessége megegyezik az alapkód sebességével. Nézzük még a kód távolságát. Mivel λ pozitív egész szám, ezért M' akkor és csak akkor nagyobb, mint 1, ha M is legalább 2, vagyis az új kódnak pontosan akkor van távolsága, ha az eredeti kódnak is volt, tegyük tehát fel, hogy $M \geq 2$. Ha $\mathbf{u}$ és $\mathbf{v} \neq \mathbf{u}$ C' két különböző eleme, akkor legalább egy i-re és j-re $u_j^{(i)} \neq v_j^{(i)}$. De ha $u_j^{(i)} \neq v_j^{(i)}$, akkor $\mathbf{u}^{(i)} \neq \mathbf{v}^{(i)}$, és mivel C távolsága d, ezért $\mathbf{u}^{(i)}$ és $\mathbf{v}^{(i)}$ legalább d helyen különbözik, amiből következik, hogy $\mathbf{u}$ és $\mathbf{v}$ távolsága is legalább d, tehát $d' \geq d$. Most legyen $\mathbf{u}^{(0)}$ és $\mathbf{v}^{(0)}$ a C két olyan eleme, amelyek távolsága d, és legyen $\mathbf{u}$ és $\mathbf{v}$ többi eleme azonos. Ekkor $\mathbf{u}$ és $\mathbf{v}$ távolsága d, vagyis van a kódban két olyan szó, amelyek távolsága d, amiből következik, hogy $d' \leq d$, azaz C' távolsága megegyezik C távolságával, $d' = d$, és C' egy $(\lambda n, M^\lambda, d)_q$-paraméterű kód.

Felmerül a kérdés, hogy mire jó egy olyan kód, amelyben nagyobb kódhosszúsághoz azonos távolság, vagyis azonos számú javítható hiba tartozik, azaz amelynek a relatív hibajavítóképessége kisebb. Nyilván semmire. Vegyük azonban tekintetbe, hogy bár t-hiba javító kód esetén λ kódszóban összesen λt hiba kijavítható, de ez nem jelenti azt, hogy egymás után küldött λ kódszóban közvetlenül egymás után következő λt számú hibát tudunk javítani. Az új kód azonban képes bármely $l \leq \lambda t$-hosszúságú hibacsomó javítására, ugyanis amennyiben a vett szóban legfeljebb λt hosszú hibacsomó van, akkor ezt a szót oszlopfolytonosan írva a $\lambda \times n$ méretű mátrixba, a hiba minden sorban legfeljebb t egymás melletti elemet érint, vagyis egy-egy eredeti szóban legfeljebb t hiba van, amit az eredeti kód képes kijavítani, és így a teljes vett szót ki tudjuk javítani. Ezzel beláttuk, hogy igaz a következő tétel.

10.1. Tétel

t-hiba javító kódból λ-szoros átfűzéssel kapott kód λt hosszúságú hibacsomót javító kód.

Δ

Most tegyük fel, hogy C egy $[n, k, d]_q$-paraméterű lineáris kód. Ekkor $M = q^k$, míg a λ-szoros átfűzés után $M' = M^\lambda = \left(q^k\right)^\lambda = q^{\lambda k}$. C' is lineáris, ugyanis az átfűzéssel kapott kód elemei is az

eredeti test elemei, vagyis összeadhatóak és megszorozhatóak a test egy elemével, tehát maguk a kódszavak is összeadhatóak és szorozhatóak az alaptest elemeivel, azaz skalárokkal, és ha a és b a test két eleme, akkor nyilván teljesül, hogy $(a\mathbf{u}+b\mathbf{v})_{\lambda j+i}=au_j^{(i)}+bv_j^{(i)}=au_{\lambda j+i}+bv_{\lambda j+i}$. De ha C' lineáris, és $M'=q^{\lambda k}$, ahol q a test elemszáma, akkor C' egy λk-dimenziós kód, azaz C' egy $[\lambda n,\lambda k,d]_q$-paraméterű kód.

Végül tegyük fel, hogy C a g generátorpolinommal generált ciklikus kód, és nézzük meg, hogy milyen szót kapunk, ha C' egy elemét ciklikusan egy hellyel jobbra léptetjük.

$$u_0^{(0)}\dots u_0^{(i)}\dots u_0^{(\lambda-1)}\dots u_j^{(0)}\dots u_j^{(i)}\dots u_j^{(\lambda-1)}\dots u_{n-1}^{(0)}\dots u_{n-1}^{(i)}\dots u_{n-1}^{(\lambda-2)}u_{n-1}^{(\lambda-1)}$$
$$\downarrow$$
$$u_{n-1}^{(\lambda-1)}u_0^{(0)}\dots u_0^{(i)}\dots u_0^{(\lambda-1)}\dots u_j^{(0)}\dots u_j^{(i)}\dots u_j^{(\lambda-1)}\dots u_{n-1}^{(0)}\dots u_{n-1}^{(i)}\dots u_{n-1}^{(\lambda-2)},$$

vagy a táblázatos alakban

$$\begin{pmatrix} u_0^{(0)} & \cdots & u_j^{(0)} & \cdots & u_{n-2}^{(0)} & u_{n-1}^{(0)} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ u_0^{(i)} & \cdots & u_j^{(i)} & \cdots & u_{n-2}^{(i)} & u_{n-1}^{(i)} \\ u_0^{(i+1)} & & u_j^{(i+1)} & & u_{n-2}^{(i+1)} & u_{n-1}^{(i+1)} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ u_0^{(\lambda-2)} & \cdots & u_j^{(\lambda-2)} & \cdots & u_{n-2}^{(\lambda-2)} & u_{n-1}^{(\lambda-2)} \\ u_0^{(\lambda-1)} & \cdots & u_j^{(\lambda-1)} & \cdots & u_{n-2}^{(\lambda-1)} & u_{n-1}^{(\lambda-1)} \end{pmatrix}$$

$$\downarrow$$

$$\begin{pmatrix} u_{n-1}^{(\lambda-1)} & u_0^{(\lambda-1)} & \cdots & u_j^{(\lambda-1)} & \cdots & u_{n-2}^{(\lambda-1)} \\ u_0^{(0)} & \cdots & u_j^{(0)} & \cdots & u_{n-2}^{(0)} & u_{n-1}^{(0)} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ u_0^{(i-1)} & & u_j^{(i-1)} & & u_{n-2}^{(i-1)} & u_{n-1}^{(i-1)} \\ u_0^{(i)} & \cdots & u_j^{(i)} & \cdots & u_{n-2}^{(i)} & u_{n-1}^{(i)} \\ \vdots & \ddots & \vdots & \ddots & \vdots & \vdots \\ u_0^{(\lambda-2)} & \cdots & u_j^{(\lambda-2)} & \cdots & u_{n-2}^{(\lambda-2)} & u_{n-1}^{(\lambda-2)} \end{pmatrix}$$

Látjuk, hogy a 0. sor az 1. sor helyére került változatlan formában, az i-edik sor az $i+1$-edik helyére, és a $\lambda-2$-edik sor a $\lambda-1$-edik helyére szintén minden változás nélkül, ugyanakkor az utolsó, $\lambda-1$-edik sor felkerült a 0. sor helyére, de ciklikusan egy hellyel jobbra tolva. Mivel az eredeti táblázat minden sora egy-egy C-beli kódszó, és C ciklikus, vagyis bármely kódszavának ciklikus eltoltja is kódszó, ezért a második táblázat valamennyi sorában is C egy-egy kódszava áll, és így ez a táblázat is, oszlopfolytonosan kiolvasva, C' egy kódszava, ami éppen azt jelenti, hogy C' is ciklikus.

Ez algebrai úton is kijön. Az $\mathbf{u}$ kódszóhoz tartozó polinom

$$u=\sum_{i=0}^{\lambda n-1}u_ix^i=\sum_{i=0}^{\lambda-1}\sum_{j=0}^{n-1}u_{\lambda j+i}x^{\lambda j+i}=\sum_{i=0}^{\lambda-1}\sum_{j=0}^{n-1}u_j^{(i)}x^{\lambda j+i}$$
$$=\sum_{i=0}^{\lambda-1}x^i\sum_{j=0}^{n-1}u_j^{(i)}\left(x^\lambda\right)^j=\sum_{i=0}^{\lambda-1}x^i\left(u^{(i)}\circ x^\lambda\right).$$

Ekkor a polinom eltoltja

$$u_{\rightarrow} = xu \bmod \left(x^{\lambda n} - e\right) = x \sum_{i=0}^{\lambda-1} x^i \left(u^{(i)} \circ x^{\lambda}\right) \bmod \left(x^{\lambda n} - e\right)$$
$$= \sum_{i=0}^{\lambda-1} \left(x^{i+1} \left(u^{(i)} \circ x^{\lambda}\right) \bmod \left(x^{\lambda n} - e\right)\right)$$
$$= \sum_{i=1}^{\lambda-1} \left(x^i \left(u^{(i-1)} \circ x^{\lambda}\right) \bmod \left(x^{\lambda n} - e\right)\right) + \left(x^{\lambda} \left(u^{(\lambda-1)} \circ x^{\lambda}\right) \bmod \left(\left(x^{\lambda}\right)^n - e\right)\right)$$
$$= \sum_{i=1}^{\lambda-1} x^i \left(u^{(i-1)} \circ x^{\lambda}\right) + \left(xu^{(\lambda-1)} \bmod \left(x^n - e\right)\right) \circ x^{\lambda}$$
$$= \sum_{i=1}^{\lambda-1} x^i \left(u^{(i-1)} \circ x^{\lambda}\right) + x^0 \left(u_{\rightarrow}^{(\lambda-1)} \circ x^{\lambda}\right) = \sum_{i=0}^{\lambda-1} x^i \left(v^{(i)} \circ x^{\lambda}\right),$$

ahol $\mathbf{v}^{(0)} = \mathbf{u}_{\rightarrow}^{(\lambda-1)}$, és $\lambda > i \in \mathbb{N}^+$-ra $\mathbf{v}^{(i)} = \mathbf{u}^{(i-1)}$. De ha C ciklikus, akkor $\mathbf{v}^{(0)} = \mathbf{u}_{\rightarrow}^{(\lambda-1)} \in C$, így $\mathbf{u}_{\rightarrow} \in C'$, a C' kód ciklikus.

Határozzuk meg C' generátorpolinomját, g'-t (a vessző nem a deriválást jelenti). C' egy $[\lambda n, \lambda k, d]_q$-paraméterű ciklikus kód, amelyben a generátorpolinom az egyetlen $\lambda n - \lambda k = \lambda(n-k)$-adfokú főpolinom, tehát ha megadunk egy ilyen polinomot, akkor az lesz a kód generátorpolinomja. Legyen $u^{(0)} = g$, ahol g a C generátor-polinomja, és legyen $\lambda > i \in \mathbb{N}^+$-ra $u^{(i)} = 0$. Ekkor

$$u = \sum_{i=0}^{\lambda-1} x^i \left(u^{(i)} \circ x^{\lambda}\right) = u^{(0)} \circ x^{\lambda} = g \circ x^{\lambda},$$

tehát u főpolinom, és a fokszáma $\lambda(n-k)$, vagyis $g' = g \circ x^{\lambda}$. Beláttuk tehát a következő tételt.

10.2. Tétel

$[n, k, d]_q$-paraméterű C kód λ-szoros átfűzésével kapott C' kód $[\lambda n, \lambda k, d]_q$-paraméterű kód, és ha C ciklikus a g generátorpolinommal, akkor C' is ciklikus a $g' = g \circ x^{\lambda}$ generátorpolinommal.

Δ

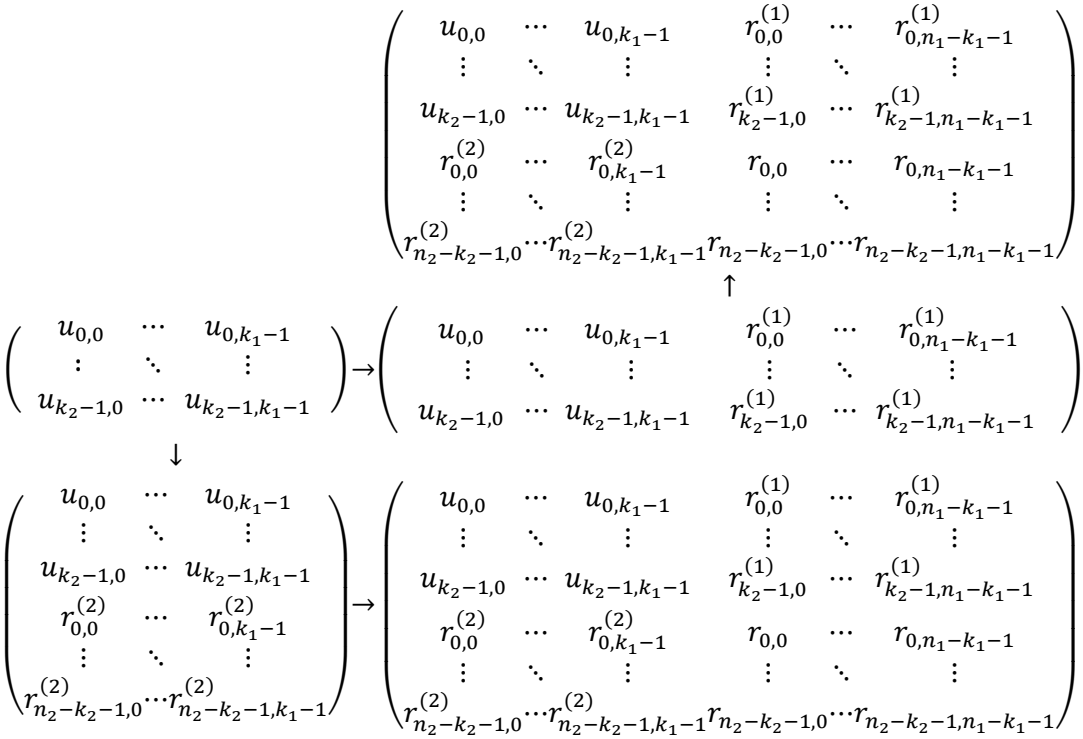
Direkt szorzat kód

Legyen adott a q-elemű test fölötti $[n_1, k_1, d_1]_q$-paraméterű C_1 és $[n_2, k_2, d_2]_q$-paraméterű C_2 kód, $\mathbf{G}_1$ és $\mathbf{H}_1$ a C_1, $\mathbf{G}_2$ és $\mathbf{H}_2$ a C_2 kód generátor- és ellenőrző mátrixa, továbbá legyen az üzenethalmaz $q^{k_1 k_2}$-elemű. Ekkor az üzenetek tekinthetőek az $\mathbb{F}_q$ fölötti $k_2 \times k_1$-méretű mátrixoknak (ezt persze úgy is felfoghatjuk, hogy az üzenethalmaz a q-elemű test fölötti k_1-dimenziós tér, és ennek a térnek k_2 üzenetét kódoljuk, illetve a q-elemű test fölötti k_2-dimenziós tér mint üzenettér k_1 elemét kódoljuk). Egy ilyen $\mathbf{U}$ mátrix egy-egy sora kódolható a C_1 kóddal, és $\mathbf{UG}_1$ egy $k_2 \times n_1$-méretű mátrix, amelynek minden sora egy-egy C_1-beli kódszó. Ugyanakkor $\mathbf{UG}_1$ minden oszlopa $\mathbb{F}_q$ fölötti k_2-dimenziós vektor, amely kódolható C_2-ben. Ez a kódolás a $\mathbf{G}_2$-vel való szorzással végezhető el, és a szorzás után a $\mathbf{V} = (\mathbf{UG}_1)^T \mathbf{G}_2 = \mathbf{G}_1^T \mathbf{U}^T \mathbf{G}_2$ mátrixot kapjuk mint az $\mathbf{U}$ üzenet kódját. De ugyanezt a kódot kapjuk akkor is, ha először az $\mathbf{U}$ oszlopait kódoljuk a $\mathbf{G}_2$ mátrixszal, és utána az így kapott mátrix sorait $\mathbf{G}_1$-gyel: ebben az esetben az első lépésben az $\mathbf{U}^T \mathbf{G}_2$, majd a második lépés után az $(\mathbf{U}^T \mathbf{G}_2)^T \mathbf{G}_1 = \mathbf{G}_2^T \mathbf{U} \mathbf{G}_1$ mátrixot kapjuk, és ez utóbbi mátrix transzponáltja $\mathbf{G}_1^T \mathbf{U}^T \mathbf{G}_2$, ami éppen az előbbi $\mathbf{V}$ mátrix. Ebből következik, hogy a végső mátrix oszlopai C_1-beli, míg sorai C_2-beli kódszavak.

Ez onnan is látszik, hogy $\mathbf{H}_1\mathbf{V} = \mathbf{H}_1(\mathbf{G}_1^T\mathbf{U}^T\mathbf{G}_2) = (\mathbf{H}_1\mathbf{G}_1^T)(\mathbf{U}^T\mathbf{G}_2) = \mathbf{0}$ és $\mathbf{H}_2\mathbf{V}^T = \mathbf{H}_2(\mathbf{G}_1^T\mathbf{U}^T\mathbf{G}_2)^T = (\mathbf{H}_2\mathbf{G}_2^T)(\mathbf{U}\mathbf{G}_1) = \mathbf{0}$.

Standard alakú generátormátrixszal az előbbieket a

8. ábra szemlélteti.



8. ábra

Ha $\mathbf{U}_1$ és $\mathbf{U}_2$ két üzenet, és a_1 valamint a_2 az $\mathbb{F}_q$ két eleme, akkor $a_1\mathbf{U}_1 + a_2\mathbf{U}_2$ is $\mathbb{F}_q$ fölötti $k_2 \times k_1$-méretű mátrix, és $\mathbf{G}_1^T(a_1\mathbf{U}_1 + a_2\mathbf{U}_2)^T\mathbf{G}_2 = a_1\mathbf{G}_1^T\mathbf{U}_1^T\mathbf{G}_2 + a_2\mathbf{G}_1^T\mathbf{U}_2^T\mathbf{G}_2$, tehát a kapott kód is lineáris. Meg kell határozni ezen C kód paramétereit. A kódszavak $n_1 \times n_2$-méretű $\mathbb{F}_q$ fölötti mátrixok, így a kódhossz $n = n_1n_2$. Mivel az üzenetek az $\mathbb{F}_q$ fölötti $k_2 \times k_1$-méretű mátrixokkal reprezentálhatóak, ezért a kód az $\mathbb{F}_q$ fölötti n_1n_2-dimenziós tér k_1k_2-dimenziós altere, $k = k_1k_2$. A kód távolságához elegendő a kód súlyának meghatározása, hiszen a kód lineáris (feltesszük, hogy az eredeti mindkét kód legalább egydimenziós, és így az eredő kód is tartalmaz nem nulla kódszót). Ha $\mathbf{V}$ nem nulla, akkor a mátrix legalább egy eleme nem nulla. Ha az i-edik oszlopban van nem nulla elem, akkor az i-edik oszlop a C_1 kód nem nulla eleme, és mivel a C_1 távolsága d_1, ezért az i-edik oszlopban legalább d_1 nullától különböző elem áll, vagyis a mátrixban legalább d_1 nullától különböző sor van. Ezek a sorok C_2-beli nem nulla kódszavak, és így a súlyuk legalább akkora, mint a C_2 kód távolsága, azaz legalább d_2. Ez azt jelenti, hogy legalább d_1 olyan sor van, amelyek mindegyikében legalább d_2 nem nulla elem található, tehát egy nem nulla kódszóban minimum d_1d_2 nullától eltérő elem áll, vagyis a két kódból keletkezett C kód d távolsága legalább d_1d_2. Most legyen $\mathbf{u}^{(1)}$ a C_1 kód egy d_1-súlyú eleme, és $\mathbf{u}^{(2)}$ egy C_2-beli, d_2-súlyú kódszó, továbbá legyen $\mathbf{V}$ egy olyan $n_1 \times n_2$-es mátrix, amelyben $V_{i,j} = u_i^{(1)}u_j^{(2)}$. A $\mathbf{V}$ j-edik oszlopában álló elemek esetén $u_j^{(2)}$ állandó, tehát ez az oszlop az $\mathbf{u}^{(1)}$ C_1-beli kódszó $u_j^{(2)}$-szerese, és így maga is C_1-beli kódszó. Hasonlóan láthatjuk, hogy a mátrix i-edik sora a C_2-beli $\mathbf{u}^{(2)}$ kódszó $u_i^{(1)}$-szerese, tehát C_2-beli kódszó. Mindebből következik, hogy ez a $\mathbf{V}$ eleme C-nek. $V_{i,j} = u_i^{(1)}u_j^{(2)}$ akkor és csak akkor nem nulla, ha mind $u_i^{(1)}$, mind $u_j^{(2)}$ nullától különböző. Összesen d_1 olyan i index van, amelyre $u_i^{(1)} \neq 0$, és azon j indexek száma, amelyekre $u_j^{(2)} \neq 0$, d_2, így $\mathbf{V}$ súlya d_1d_2, amiből következik, hogy a kód súlya legfeljebb ekkora. Mivel korábban azt találtuk,

hogy a kód súlya legalább $d_1 d_2$, ezért $d = d_1 d_2$, tehát C egy $[n_1 n_2, k_1 k_2, d_1 d_2]_q$-paraméterű lineáris kód. Beláttuk tehát a következő tételt.

10.3. Tétel

Legyen C_1 egy $[n_1, k_1, d_1]_q$-paraméterű kód a $\mathbf{G}_1$ és C_2 egy $[n_2, k_2, d_2]_q$-paraméterű kód a $\mathbf{G}_2$ generátormátrixszal. Ekkor az $\mathbb{F}_q^{k_2 \times k_1}$ elemein az $\mathbf{U} \mapsto \mathbf{G}_1^T \mathbf{U}^T \mathbf{G}_2$ szabállyal értelmezett C kód egy $[n_1 n_2, k_1 k_2, d_1 d_2]_q$-paraméterű lineáris kód.

Δ

10.4. Definíció

Legyen C_1 a $\mathbf{G}_1$ és C_2 a $\mathbf{G}_2$ generátormátrixszal generált kód, és legyen $\mathbf{U} \in \mathbb{F}_q^{k_2 \times k_1}$. Az $\mathbf{U} \mapsto \mathbf{G}_1^T \mathbf{U}^T \mathbf{G}_2$ szabállyal értelmezett C kód a C_1 **és** C_2 **direkt szorzata**, egy **direkt szorzat kód**.

Δ

A kód sebessége $\mathcal{R} = \frac{1}{n_1 n_2} \log_q M = \frac{1}{n_1 n_2} \log_q q^{k_1 k_2} = \frac{k_1 k_2}{n_1 n_2} = \frac{k_1}{n_1} \frac{k_2}{n_2} = \mathcal{R}_1 \mathcal{R}_2$.

Legyen $t_1 = \left\lfloor \frac{d_1 - 1}{2} \right\rfloor$ és $t_2 = \left\lfloor \frac{d_2 - 1}{2} \right\rfloor$. Tegyük fel, hogy a kódolás eredményeképpen kapott táblázatot sorfolytonosan olvassuk ki. Amennyiben az átvitel során egy legfeljebb $n_1 t_2$ hosszúságú hibacsomó lép fel, és ezen kívül még olyan véletlen hibák vannak, amelyeket soronként C_1 képes javítani, akkor sorfolytonosan visszaírva a vett szót a táblázatba, és C_1-gyel kijavítva a véletlen hibákat, a táblázat minden oszlopában legfeljebb t_2 hiba lesz, amit a C_2 kód képes javítani. Ugyanígy, ha B-t kódoljuk, és oszlopfolytonos a kiolvasás illetve beírás, akkor a kód képes egy maximum $n_2 t_1$ hosszú hibacsomó, valamint a C_2-vel javítható további véletlen hibák javítására.

Mivel a direkt szorzat kódban a kódszavak olyan mátrixoknak tekinthetőek, amelyeknek mind a sorai, mind az oszlopai egy-egy kód elemei, ezért megpróbálhatjuk a hibajavítást oly módon, hogy először például soronként javítunk a C_2 kód alapján, majd az ily módon javított mátrix oszlopait javítjuk a C_1-beli döntési függvény alapján. Ilyen módszerrel azonban nem használjuk ki az összetett kód hibajavító képességét. A C_2 kód távolsága d_2, így soronként $t_2 = \left\lfloor \frac{d_2 - 1}{2} \right\rfloor$ hiba javítható, míg az oszloponként javítható hibák száma $t_1 = \left\lfloor \frac{d_1 - 1}{2} \right\rfloor$, hiszen a C_1 kód távolsága d_1, vagyis összességében a soronként és oszloponként elkülönítetten történő hibajavítás esetén $t_1 t_2$ hiba javítható. Ennél valamivel jobb a helyzet. Ha ugyanis a hibák száma kisebb, mint $(t_1 + 1)(t_2 + 1)$, akkor ez soronkénti és oszloponkénti javítással kijavítható. Ekkor ugyanis az olyan sorok száma, amelyben t_2-nél több hiba van, legfeljebb t_1, így a soronkénti javítás után legfeljebb t_1 sorban marad hiba (nem feltétlenül az eredeti pozíciókban). Ez viszont azt jelenti, hogy minden oszlopban legfeljebb t_1 hiba van, amelyeket az oszloponkénti javítással meg lehet szüntetni.

Most legyen $(t_1 + 1)(t_2 + 1)$ olyan hibánk, amely $t_1 + 1$ sorban és $t_2 + 1$ oszlopban helyezkedik el, egy $(t_1 + 1)(t_2 + 1)$-pontú rács rácspontjaiban. Mind C_1-ben van $t_1 + 1$ hibát tartalmazó, nem javítható hibaminta, mind C_2-ben $t_2 + 1$ hibából álló, nem javítható hibaminta. Ha a kód súlya páratlan, akkor egy ilyen hibaminta nem nulla konstansszorosa is nem javítható hibaminta, míg ha a kód súlya páros, akkor ez nem feltétlenül igaz, de a döntési függvény mindig megváltoztatható úgy, hogy az előbbi feltétel teljesüljön (és ettől a kód hibajavító képessége nem változik). Ha $\mathbf{u}^{(1)}$ a C_1 egy kódszavából $t_1 + 1$ hibával keletkező szó, és $\mathbf{u}^{(2)}$ a C_2 egy kódszavából $t_2 + 1$ hibával keletkező szó, akkor az $U_{i,j} = u_i^{(1)} u_j^{(2)}$ szabállyal alkotott $n_1 \times n_2$ méretű $\mathbf{U}$ mátrix a direkt szorzat kód egy kódszavából kapott, az előbbiekben leírt, $(t_1 + 1)(t_2 + 1)$ hibát tartalmazó szó. Ha ezt a szót javítjuk előbb soronként, majd oszloponként, akkor minden sorban azonos, de az eredetitől különböző kódszót kapunk. A soronkénti javítás után tehát olyan mátrix lesz, amelyben az előbbi hibás sorok az eredetitől különböző, nem nulla konstans szorzótól eltekintve azonos kódszavak, így a mátrixban továbbra is

lesznek hibás oszlopok, amelyek továbbra is $t_1 + 1$ hibát, az eredeti mátrix oszlopaiban volt hiba nem nulla konstansszorosait tartalmazzák. Mivel ezek a hibák nem javíthatóak, így az oszloponkénti javítás után sem kapjuk vissza az eredeti, elküldött kódszót, de a mátrix minden sora és minden oszlopa a megfelelő kód kódszava, és ezért már nem javítható. Ez mutatja, hogy van olyan $(t_1 + 1)(t_2 + 1)$ hibát tartalmazó hibaminta, amely a külön-külön soronként és oszloponként végrehajtott javítással nem javítható.

Ugyanakkor a direkt szorzat kód távolsága $d = d_1 d_2$, így ez a kód $t = \left\lfloor \frac{d-1}{2} \right\rfloor = \left\lfloor \frac{d_1 d_2 - 1}{2} \right\rfloor$ hibát képes javítani minimális távolságú dekódolással, így, ha $t \geq (t_1 + 1)(t_2 + 1)$, akkor a soronkénti-oszloponkénti javítással nem használjuk ki teljes mértékben a kód hibajavító képességét. Az előbbi feltétel viszont pontosan akkor igaz, ha mindkét kód távolsága legalább 2.

Ha például egy n darab m-bites bináris szóból álló adatblokk minden szavát kiegészítjük egy-egy paritásbittel, és a blokk valamennyi, az egyes szavakban azonos pozíción álló bitekből álló oszlopát is megtoldjuk egy paritásbittel, akkor ha mindkét irányban párosra egészítünk ki, a keletkezett kód lineáris, és az eredeti szavak paritásbitjeiből álló oszlop ellenőrzőbitje megegyezik az oszlopok ellenőrző bitjeiből álló szó paritásbitjével. Most mind a soronként, mind az oszloponként alkalmazott kód távolsága 2, vagyis külön-külön egyik sem alkalmas hibajavításra, ugyanakkor a direkt szorzat kód távolsága 4, tehát a kód képes egy hiba javítására, és két hiba jelzésére. Az egy hiba úgy javítható, hogy egyetlen hiba esetén pontosan egy sorban és egy oszlopban kapunk hibajelzést, és a jelzett sor és jelzett oszlop metszéspontjában áll a hibás bit, amelyet az ellentettjére változtatva kijavítottuk a hibát. Ha viszont két hiba lép fel, akkor vagy legalább két sorban, vagy (nem kizáró módon) minimum két oszlopban kapunk hibajelzést, ami mutatja, hogy biztosan legalább két hiba lépett fel, amit a kód már nem tud (és nem is kell, hogy tudjon) javítani.

Végül vizsgáljuk a kód ciklikusságát. Sor- illetve oszlopfolytonos működés esetén hiába ciklikus mindkét kód, a direkt szorzat kód nem ciklikus. Ha mondjuk oszlopfolytonos a kiolvasás, akkor az egy hellyel való ciklikus jobbraléptetés során például a második oszlopban álló kódszó úgy módosul, hogy az utolsó jegye elvész, míg az elejére az első oszlop utolsó jegye kerül, és ez a jelsorozat általában nem kódszó (a sorokkal a helyzet ugyanaz, mint az átfűzéses kódnál, vagyis az egyes sorok egy sorral lejjebb kerülnek, míg az utolsó sor a 0. sor helyére megy, egy hellyel ciklikusan jobbra léptetve, tehár soriránybanban nincs probléma, ám a direkt szorzat kódban oszloponként is kódszavaknak kell állnia, és ez nem teljesül a teljes kód egy hellyel való elléptetése esetén, ha a kódot oszlopfolytonosan olvassuk – és nyilván a sorirány sérülne sorfolytonos kiolvasásnál). Megmutatható, hogy bizonyos feltételek teljesülése esetén más sorrendben olvasva a mátrix elemeit, elérhető, hogy amennyiben mindkét kód ciklikus, úgy a direkt szorzat kód is ciklikus legyen. Ha például $n_1 = 7$ és $n_2 = 5$, akkor az alábbi kiolvasással ciklikus kódot kapunk, feltéve, hogy az eredeti két kód ciklikus volt:

$$\begin{pmatrix} 0 & 15 & 30 & 10 & 25 & 5 & 20 \\ 21 & 1 & 16 & 31 & 11 & 26 & 6 \\ 7 & 22 & 2 & 17 & 32 & 12 & 27 \\ 28 & 8 & 23 & 3 & 18 & 33 & 13 \\ 14 & 29 & 9 & 24 & 4 & 19 & 34 \end{pmatrix}$$

Kaszkád kód

Legyen C_2 egy $[n_2, k_2, d_2]_{q_2}$-, míg C_1 egy $[n_1, k_1, d_1]_{q_1}$-paraméterű kód, ahol $q_2 = q_1^{k_1}$. Ekkor $\mathbb{F}_{q_2}$ k_1-dimenziós tér $\mathbb{F}_{q_1}$ felett, és ha választunk $\mathbb{F}_{q_2}$-ben egy bázist, akkor $\mathbb{F}_{q_2}$ minden eleme egy és csak egyféleképpen adható meg egy $\mathbb{F}_{q_1}$-beli k_1-komponensű vektorral, vagyis egy k_1-hosszúságú szóval. Ha most $\mathbb{F}_{q_1}^{k_1 k_2}$ az üzenettér, akkor egy-egy üzenet az $\mathbb{F}_{q_1}$ elemeiből álló $k_1 k_2$ hosszúságú sorozat. Ennek minden k_1 hosszúságú szakasza kölcsönösen egyértelműen leképezhető $\mathbb{F}_{q_2}$-be, és a leképezés után egy $\mathbb{F}_{q_2}$ feletti k_2 hosszúságú szót kapunk, amely C_2-vel egy szintén egy $\mathbb{F}_{q_2}$ feletti n_2 hosszúságú szóba kódolható. Ennek a szónak minden betűje visszaírható egy $\mathbb{F}_{q_1}$ fölötti k_1 hosszúságú szóba, és ez C_1-gyel kódolva $\mathbb{F}_{q_1}$ fölötti n_1 hosszúságú szót ad, vagyis végeredményként az $\mathbb{F}_{q_1}$ fölötti

k_1k_2 hosszúságú üzenetet egy $\mathbb{F}_{q_1}$ fölötti n_1n_2 hosszúságú kódszóba kódoltunk, tehát így egy $[n_1n_2, k_1k_2, d]_{q_1}$-paraméterű C kódot kapunk.

Mivel a kód lineáris, ezért egy üzenet kódja akkor és csak akkor 0, ha az üzenet minden eleme 0. Egy nullától különböző üzenetet tekintve, ennek valamely k_1 hosszúságú szakasza, tehát a megfelelő $\mathbb{F}_{q_2}$-beli elem sem 0, és így a C_2-beli kódszó is nullától különböző. Mivel ennek a kódnak a távolsága d_2, ezért a kódszóban legalább ennyi komponens nem 0. Ezeket a komponenseket a C_1 kóddal kódolva, mindegyik legalább d_1 súlyú kódszót ad, és így a teljes kódszó súlya, és akkor vele együtt a kód távolsága legalább d_1d_2, tehát a C kód távolsága $d \geq d_1d_2$. Ennél többet azonban nem mondhatunk, ugyanis ha az üzenetet első lépésben a C_2 egy minimális súlyú kódszavába kódoltuk, akkor utána az egyes betűk C_1-beli kódja általában nem minimális súlyú, vagyis általában nincs C-ben d_1d_2 súlyú kódszó.

A kódolás most az alábbi módon néz ki:

9. ábra

Az ábra alapján érthető, hogy C_1-et belső, míg C_2-t külső kódnak nevezik.

10.5. Definíció

Ha egy üzenetet egy C_2 kóddal kódolva, a kódszavak betűit egy C_1 kóddal kódoljuk, akkor az így kapott C kód egy **kaszkád kód**, ahol C_2 a **külső kód**, és C_1 a **belső kód**.

Δ

Állapítsuk meg a lineáris kaszkád kód kódsebességét.

$$\mathcal{R} = \frac{1}{n_1n_2}\log_q q^{k_1k_2} = \frac{k_1k_2}{n_1n_2} = \frac{k_1}{n_1}\frac{k_2}{n_2} = \mathcal{R}_1\mathcal{R}_2,$$

tehát az $\mathcal{R}_1$ sebességű C_1 és $\mathcal{R}_2$ sebességű C_2 lineáris kódból álló kaszkád kód sebessége $\mathcal{R} = \mathcal{R}_1\mathcal{R}_2$.

Összefoglalva azt kaptuk, hogy

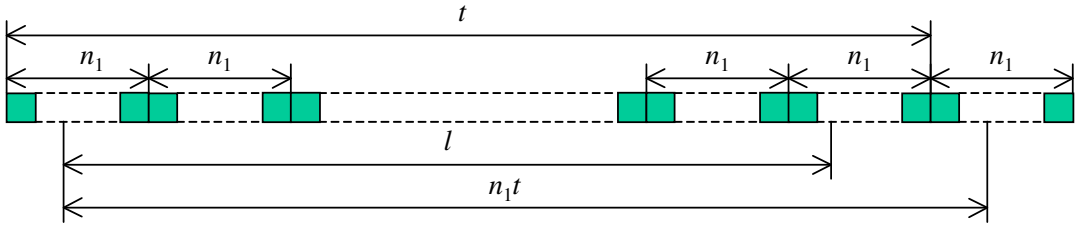
10.6. Tétel

Az $\mathbb{F}_{q_1}$ fölötti $[n_1, k_1, d_1]_{q_1}$-paraméterű, $\mathcal{R}_1$ sebességű C_1 kódból mint belső kódból és az $\mathbb{F}_{q_1^{k_1}}$ fölötti $[n_2, k_2, d_2]_{q_1^{k_1}}$-paraméterű $\mathcal{R}_2$ sebességű C_2 kódból mint külső kódból konstruált kaszkád kód egy $\mathbb{F}_{q_1}$ fölötti $[n_1n_2, k_1k_2, d]_{q_1}$-paraméterű, $\mathcal{R} = \mathcal{R}_1\mathcal{R}_2$ sebességű C kód, ahol $d \geq d_1d_2$.

Δ

A kaszkád kód általában nem ciklikus, hiszen ha egy kódszót egy pozícióval ciklikusan jobbra tolunk, akkor az egyes n_1-hosszúságú szakaszok mint C_1-beli kódszavak úgy változnak, hogy a jobb szélső betű kicsúszik, viszont a bal oldalon az előtte lévő kódszó utolsó betűje jelenik meg első betűként, és általában egy ilyen szó nem eleme a C_1 kódnak (ha például C_1 egy paritásbites bináris kód, akkor jobbról lehagyva egy bitet és balról kiegészítve egy másik kódszó utolsó bitjével, a kapott szóban az egyesek száma akár páros, akár páratlan is lehet).

A távolságok alapján C_2 ki tud javítani bármely $t = \left\lfloor \frac{d_2-1}{2} \right\rfloor$-nél nem több hibát minimális távolságú dekódolással. Amennyiben egy kódszó az átvitel során úgy sérül meg, hogy az egyes n_1-hosszúságú szakaszokat mint C_1-beli kódszavakat javítva, egy legfeljebb $n_1(t-1)+1$-hosszúságú hibacsomó marad, akkor az n_1 hosszú szakaszokat visszaírva az eredeti k_1 hosszúságú S_1 feletti szavakká, majd ezeket a megfelelő S_2-beli betűvel helyettesítve, egy olyan S_2 feletti n_2-hosszúságú szót kapunk, amely legfeljebb t hibát tartalmaz, és ezt C_2 képes javítani, vagyis C ki tud javítani egy legfeljebb $n_1(t-1)+1$ hosszúságú hibacsomót:



10. ábra

Gondoljuk meg, hogy milyen kódot kapunk akkor, ha előbb a C_1 kóddal kódolunk, majd az így kapott n_1-hosszúságú kódszavakat $\mathbb{F}_{q_1^{n_1}}$ elemeinek tekintve, k_2 egymás utáni kódszóra alkalmazzuk a C_2 kódot, végül egy ilyen kódszó minden betűjét visszaírjuk $\mathbb{F}_{q_1}$-be. Az eredmény egy $\mathbb{F}_{q_1}$ fölötti n_1n_2-hosszúságú kódszó lesz, ugyanúgy, mint az előbb, és a kód mérete, valamint a sebessége sem változik. Más a helyzet azonban a kód távolságával. Ha egy nullától különböző üzenetet kódolunk, akkor első lépésben legalább egy nem nulla kódszót kapunk. Az így kapott kódszavak az átírás után egy nem nulla üzenetet adnak a C_2 bemenetére, amelyet kódolva egy legalább d_2-súlyú kódszót kapunk. Ha azonban most ezt a kódszót visszaírjuk $\mathbb{F}_{q_1}$-be, akkor csak annyit állíthatunk biztosan, hogy a nem nulla elemek nem nulla $\mathbb{F}_{q_1}$ fölötti n_1-hosszúságú sorozatot adnak, azonban hogy egy-egy ilyen sorozatban hány nem nulla elem van, azt nem tudhatjuk (szélső esetben az ilyen elemek száma akár egy is lehet). Mindössze tehát annyit tudunk, hogy a konstruált kódban egy nem nulla kódszó legalább d_2 nem nulla elemet tartalmaz, azaz a kód súlya $d' \geq d_2$.

A kaszkád kód egy elterjedt használata, amikor a bináris üzenetet külső kódként egy 2^r-elemű test fölötti Reed-Solomon kóddal (vagy rövidített Reed-Solomon kóddal) kódolunk. Az ilyen kódot **binárisba fejtett Reed-Solomon kód**nak nevezik. Az elterjedt használatban $r = 8$, vagyis a külső kód betűi bájtok, míg a belső kód többnyire egy paritásbites kód, vagyis a Reed-Solomon kódból kapott bájtokat egy paritásbittel egészítjük ki.

11. Euklideszi algoritmus

Emlékeztetünk rá, hogy az $\mathcal{R}$ integritási tartomány akkor euklideszi gyűrű, ha létezik olyan $\varphi: R^* \to \mathbb{N}$ leképezés (az euklideszi norma), hogy tetszőleges $u \in R$ és $v \in R^*$ esetén $u = qv + r$, ahol $q \in R$, $r \in R$, és $r = 0$, vagy $r \neq 0$ és $\varphi(r) < \varphi(v)$. Ekkor a gyűrű egységelemes, bármely két elemének létezik asszociálttól eltekintve egyértelműen meghatározott legnagyobb közös osztója, és ez a legnagyobb közös osztó meghatározható az euklideszi algoritmussal. Az algoritmus a következő. Vezessük be az $r_{-1} = u$, $r_0 = v$ jelölést. Ekkor van olyan $n \in \mathbb{N}$, hogy minden $n \geq i \in \mathbb{N}$ indexre

$$r_{i-1} = q_i r_1 + r_{i+1}$$

úgy, hogy $r_i \neq 0$ és $r_{n+1} = 0$, továbbá ha $i < n$, akkor $\varphi(r_{i+1}) < \varphi(r_i)$. Ekkor $d = r_n$ az u és v legnagyobb közös osztója, és minden $-1 \leq i \leq n$ indexre van R-nek olyan a_i és b_i eleme, amellyel $r_i = a_i u + b_i v$. Ezeket az együtthatókat is meghatározhatjuk az euklideszi algoritmussal. $a_{-1} = e$ és $b_{-1} = 0$-ra nyilván igaz, hogy $r_{-1} = u = a_{-1}u + b_{-1}v$ és az $a_0 = 0$, $b_0 = e$ együtthatókkal a hasonló $r_0 = v = a_0 u + b_0 v$ egyenlőség. Ha most $r_{i-1} = a_{i-1}u + b_{i-1}v$ és $r_i = a_i u + b_i v$ az $n > i \in \mathbb{N}$ indexre, akkor $a_{i+1} = a_{i-1} - q_i a_i$, $b_{i+1} = b_{i-1} - q_i b_i$ jelöléssel

$$\begin{aligned} r_{i+1} &= r_{i-1} - q_i r_i = (a_{i-1}u + b_{i-1}v) - q_i(a_i u + b_i v) \\ &= (a_{i-1} - q_i a_i)u + (b_{i-1} - q_i b_i)v = a_{i+1}u + b_{i+1}v, \end{aligned}$$

vagyis $i + 1$-re is teljesül az összefüggés.

Igazolható, hogy euklideszi gyűrűben mindig van olyan euklideszi norma, amelynél $uv \neq 0$ esetén $\varphi(uv) \geq \varphi(u)$. A továbbiakban feltételezzük, hogy az euklideszi norma teljesíti ezt a feltételt.

Legyen $u \neq 0$ és $\varphi(u) < \varphi(v)$. Ekkor

$$r_{-1} = u = 0 \cdot v + u = q_0 r_0 + r_1,$$

ahol $r_1 = u \neq 0$ és $\varphi(r_1) = \varphi(u) < \varphi(v) = \varphi(r_0)$, majd

$$v = r_0 = q_1 r_1 + r_2.$$

Ugyanide jutunk, ha $r_{-1} = v$ és $r_0 = u$, de eggyel kevesebb lépésből áll az algoritmus, elmarad az első, felesleges lépés. Emiatt a továbbiakban azt is feltesszük, hogy ha $u \neq 0$, akkor $\varphi(u) \geq \varphi(v)$.

Mint tudjuk, test fölötti polinomgyűrű euklideszi, ahol a nem nulla f polinomra $\varphi(f) = \deg(f)$ euklideszi norma, amely kielégíti a fenti megkötést.

Az alábbiakban az euklideszi algoritmus néhány további tulajdonságát vizsgáljuk.

11.1. Tétel

Legyen $\mathcal{R}$ euklideszi gyűrű az e egységelemmel és φ normával, amelynél $\varphi(ab) \geq \varphi(a)$, valahányszor $ab \neq 0$, és legyen $u \in R$ és $v \in R^*$ úgy, hogy $u = 0$, vagy $u \neq 0$ és $\varphi(u) \geq \varphi(v)$. Ekkor

1. ha $u|v$, akkor $v|u$;

ha az euklideszi algoritmusban $(u, v) = d = r_n$, akkor

2. $n = 0$ akkor és csak akkor, ha $v|u$;
3. ha $u = 0$, akkor $q_0 = 0$, egyébként $n \geq i \in \mathbb{N}$ esetén $q_i \neq 0$;

4. $n \geq i \in \mathbb{N}^+$ esetén $a_i \neq 0$, $b_i \neq 0$;
5. ha $n \geq i \in \mathbb{N}$, akkor $a_{i-1}b_i - a_i b_{i-1} = (-1)^i e$;
6. ha $-1 \leq i \leq n$, akkor $(a_i, b_i) = e$, míg $n \geq i \in \mathbb{N}$-re $(a_{i-1}, a_i) = e$ és $(b_{i-1}, b_i) = e$;
7. $n \geq i \in \mathbb{N}$-re $r_i a_{i-1} - r_{i-1}a_i = (-1)^i v$ és $r_{i-1}b_i - r_i b_{i-1} = (-1)^i u$.

Ha $\mathcal{R}$ egy $\mathcal{K}$ test feletti polinomgyűrű az $f \neq 0$ polinomokra a $\varphi(f) = \deg(f)$ normával, akkor az f és g nem nulla polinomokra

8. $n \geq i \in \mathbb{N}$ esetén $\deg(q_i) = \deg(r_{i-1}) - \deg(r_i)$, és ha $i > 0$, akkor $\deg(q_i) > 0$;
9. $-1 \leq i \leq n$-re $\deg(q_i) = \deg(f) - \sum_{j=0}^{i} \deg(q_j)$;
10.
   a) $\deg(b_0) \leq \deg(b_1)$, és $n > i \in \mathbb{N}^+$-re $\deg(a_i) < \deg(a_{i+1})$, $\deg(b_i) < \deg(b_{i+1})$;
   b) az $n \geq i \in \mathbb{N}$ indexekre $\deg(b_i) = \deg(f) - \deg(r_{i-1}) < \deg(f)$, és ha az index pozitív, akkor $\deg(a_i) = \deg(g) - \deg(r_{i-1}) < \deg(g)$.

Δ

Bizonyítás:

Az euklideszi algoritmus szerint $n \geq i \in \mathbb{N}$-re $r_{i-1} = q_i r_i + r_{i+1}$. Innen $q_i r_i = r_{i-1} - r_{i+1}$, másrészt $r_{i+1} = r_{i-1} - q_i r_i$. Ezt a két összefüggést többször alkalmazzuk.

1. Ha $u|v$, akkor $u \neq 0$ (mert $v \neq 0$) és $v = q'u$, így

$$u = qv + r = qq'u + r,$$

ahol $r = 0$, vagy $r \neq 0$ és $\varphi(r) < \varphi(u)$. A fenti egyenlőségből $r = (e - qq')u = q''u$, és ha $r \neq 0$, akkor ebből következik, hogy $\varphi(u) \leq \varphi(r) < \varphi(u)$, ami nyilvánvalóan lehetetlen, így $r = 0$, és akkor $u = qv$, tehát $v|u$.

2. Tegyük fel, hogy $v|u$. Ekkor

$$r_{-1} = u = qv = qv + 0 = q_0 r_0 + r_1,$$

vagyis ekkor $n = 0$ és $d = r_0 = v$. Fordítva, ha $n = 0$, akkor

$$u = r_{-1} = q_0 r_0 + r_1 = q_0 r_0 + 0 = q_0 r_0 = qv,$$

ami azt jelenti, hogy $v|u$.

3. Ha $u = 0$, akkor $u = 0 \cdot v + 0$, tehát $q_0 = 0$, és mivel ekkor $v|u$, ezért $n = 0$. Most tegyük fel, hogy $u \neq 0$. Ekkor $q_i = 0$ esetén $r_{i-1} = r_{i+1}$, ami lehetetlen, mert $r_{i-1} \neq 0$, és vagy $r_{i+1} = 0$, vagy ha nem, akkor $\varphi(r_{i-1}) \geq \varphi(r_i) > \varphi(r_{i+1})$.

4. Ha egy $n > i \in \mathbb{N}^+$-ra $a_i = 0$, akkor $0 \neq r_i = b_i v$, és így $\varphi(v) = \varphi(r_0) > \varphi(r_i) \geq \varphi(v)$, míg $b_i = 0$-val $0 \neq r_i = a_i u$, így $u \neq 0$, és ekkor $\varphi(u) \geq \varphi(v) = \varphi(r_0) > \varphi(r_i) \geq \varphi(u)$, ami lehetetlen.

5. $a_{-1}b_0 - a_0 b_{-1} = e \cdot e - 0 \cdot 0 = e$, és ha $n > i \in \mathbb{N}$ -re $a_{i-1}b_i - a_i b_{i-1} = (-1)^{i+1} e$, akkor

$$\begin{aligned} a_i b_{i+1} - a_{i+1} b_i &= a_i(b_{i-1} - q_i b_i) - (a_{i-1} - q_i a_i) b_i \\ &= -(a_{i-1}b_i - a_i b_{i-1}) = -(-1)^{i+1} e = (-1)^{(i+1)+1} e. \end{aligned}$$

6. 5. szerint $a_{i-1}b_i - a_i b_{i-1} = \pm e$, és ez csak a tétel legnagyobb közös osztóival lehetséges, hiszen a felsorolt legnagyobb közös osztók mindegyike szükségszerűen osztója a jobb oldalnak, e-nek.

7. Ha $i = -1$ vagy $i = 0$, akkor

$$\begin{aligned} r_0 a_{-1} - r_{-1} a_0 &= v \cdot e - u \cdot 0 = v = (-1)^0 v \\ r_1 a_0 - r_0 a_1 &= r_1 \cdot 0 - v \cdot e = -v = (-1)^1 v, \end{aligned}$$

valamint

$$r_{-1}b_0 - r_0b_{-1} = u \cdot e - v \cdot 0 = u = (-1)^0u$$
$$r_0b_1 - r_1b_0 = r_0(-q_0) - r_1e = -(q_0r_0 + r_1) = -r_{-1} = -u = (-1)^1u,$$

ami mutatja, hogy a fentebb megadott két indexre érvényesek az egyenlőségek. Most tegyük fel, hogy $0 \leq j \leq i < n$ esetén $r_ja_{j-1} - r_{j-1}a_j = (-1)^jv$ és $r_{j-1}b_j - r_jb_{j-1} = (-1)^ju$. Ekkor

$$r_{i+1}a_i - r_ia_{i+1} = r_{i+1}a_i - r_i(a_{i-1} - q_ia_i) = (r_{i+1} + q_ir_i)a_i - r_ia_{i-1}$$
$$= -(r_ia_{i-1} - r_{i-1}a_i) = -(-1)^iv = (-1)^{i+1}v,$$

$$r_ib_{i+1} - r_{i+1}b_i = r_i(b_{i-1} - q_ib_i) - r_{i+1}b_i = -(r_{i+1} + q_ir_i)b_i + r_ib_{i-1}$$
$$= -(r_{i-1}b_i - r_ib_{i+1}) = -(-1)^iu = (-1)^{i+1}u,$$

tehát valamennyi tekintetbe vett indexre érvényes az állított egyenlőség.

A polinomokra vonatkozó állítások bizonyításánál felhasználjuk, hogy test fölötti nem nulla polinomok szorzatának foka a tényezők fokainak összege, és különböző fokú polinomok összegének és különbségének foka a tagok fokszámainak maximuma.

8. $n \geq i \in \mathbb{N}$-re $q_i \neq 0$ és $r_{i-1} \neq 0 \neq r_i$, ezért $q_ir_i = r_{i-1} - r_{i+1} \neq 0$, $\deg(r_{i-1}) \geq \deg(r_i)$, $r_{i+1} = 0$ (az $i = n$ esetben) vagy $r_{i+1} \neq 0$ és $\deg(r_i) > \deg(r_{i+1})$, így

$$\deg(q_i) = \deg(r_{i-1} - r_{i+1}) - \deg(r_i) = \deg(r_{i-1}) - \deg(r_i),$$

és ez nagyobb nullánál, ha $i > 0$.

9. $\deg(f) - \sum_{j=0}^{-1} \deg(q_j) = \deg(f) = \deg(r_{-1})$. Ha $n \geq i \in \mathbb{N}$, akkor $\deg(q_j)$-t az előző pontból helyettesítve

$$\sum_{j=0}^{i} \deg(q_j) = \sum_{j=0}^{i} \left(\deg(r_{j-1}) - \deg(r_j)\right) = \deg(r_{-1}) - \deg(r_i) = \deg(f) - \deg(r_i),$$

és ebből átrendezéssel kapjuk r_i fokát.

10.Azt tudjuk, hogy aaz a_i és b_i együtthatók és a q_i hányadosok egyike sem 0, és a pozitív indexekre $\deg(q_i) > 0$. Legyen n pozitív egész.

a) $b_{-1} = 0 = a_0$, $b_0 = e = a_1$, így $\deg(b_0) = 0 = \deg(a_1)$, és $b_1 = b_{-1} - q_0b_0 = -q_0$, tehát $\deg(b_1) = \deg(q_0) \geq 0 = \deg(b_0)$. Legyen $n > 1$. $\deg(a_2) = \deg(q_1) > 0 = \deg(a_1)$, hiszen $a_2 = a_0 - q_1a_1 = -q_1$, és ha $\deg(b_i) \geq \deg(b_{i-1})$ és $\deg(a_i) > deg(a_{i-1})$, akkor a rekurziós összefüggésből és $\deg(q_i) > 0$-ból $\deg(b_{i+1}) > deg(b_i)$ és $\deg(a_{i+1}) > deg(a_i)$.

b) Az előbbi pontot felhasználva

$$\deg(b_0) = 0 = \deg(f) - \deg(f) = \deg(f) - \deg(r_{-1}) = \deg(f) - \deg(r_{0-1})$$

és

$$\deg(a_1) = 0 = \deg(g) - \deg(g) = \deg(g) - \deg(r_0) = \deg(g) - \deg(r_{1-1}).$$

A továbbiakban tegyük fel, hogy $i < n$ esetén egy nemnegatív i-re $\deg(b_i) = \deg(f) - \deg(r_{i-1})$ és $\deg(a_i) = \deg(g) - \deg(r_{i-1})$, ha az i pozitív. Ekkor

$$\deg(b_{i+1}) = \deg(q_i) + \deg(b_i) = \deg(r_{i-1}) - \deg(r_i) + \deg(f) - \deg(r_{i-1})$$
$$= \deg(f) - \deg(r_i) = \deg(f) - \deg\left(r_{(i+1)-1}\right)$$

valamint

$$\begin{aligned}\deg(a_{i+1}) &= \deg(q_i) + \deg(a_i) = \deg(r_{i-1}) - \deg(r_i) + \deg(g) - \deg(r_{i-1}) \\ &= \deg(g) - \deg(r_i) = \deg(g) - \deg\left(r_{(i+1)-1}\right),\end{aligned}$$

mert ha $i < n$, akkor $\deg(r_i) > \deg(r_{i+1}) \geq 0$. Még azt kell belátnunk, hogy $\deg(b_i) < \deg(f)$ valamint $\deg(a_i) < \deg(g)$. Ellenkező esetben $\deg(r_{i-1}) = 0$. Ekkor $\deg(r_i) = 0$ is igaz, mert a maradékok fokszáma nem növekedhet, és a fokszámok egyenlősége is csak akkor lehetséges, ha $i = 0$, ami most nem igaz, hiszen i pozitív egész.

□

A továbbiakban elsősorban az $n > i \in \mathbb{N}$ indexekkel talált

$$\deg(d) = \deg(r_n) < \deg(r_i) < \deg(g) \leq \deg(f)$$

$$\deg(b_i) = \deg(f) - \deg(r_{i-1})$$

egyenlőtlenségekre valamint arra lesz szükségünk, hogy az a_i és b_i polinomok relatív prímek valamennyi, a $-1 \leq i \leq n$ feltételnek megfelelő indexre.

12. Alternáns kódok

Először megismétlünk néhány dolgot a ciklikus kódoknál tanultakból.

Legyen C egy $[n,k,d]_q$ ciklikus kód az $\mathbb{F}_q$ test fölött, ahol k pozitív egész, és g a kód generátorpolinomja. Ekkor $0 \neq g \in \mathbb{F}_q[x]$, $\deg(g) = n-k$, g osztója az $x^n - e$ polinomnak, ahol e az $\mathbb{F}_q$ egységeleme, továbbá az $\mathbb{F}_q$ fölötti legfeljebb $n-1$-edfokú c polinom akkor és csak akkor kódszópolinom a C kóddal, ha g osztója c-nek. Ha g gyökei egyszeresek, és biztosan ez a helyzet, ha n és q relatív prímek, akkor az előbbi feltétel pontosan akkor teljesül, ha g minden gyöke gyöke c-nek. Legyen g $\mathbb{F}_q$ fölötti irreducibilis felbontása $g = \prod_{i=1}^{s} m^{(i)}$. Ha $\alpha^{(i)}$ gyöke $m^{(i)}$-nek, és $\deg\left(m^{(i)}\right) = n_i$, akkor ${\alpha^{(i)}}^{q^j}$ is gyöke $m^{(i)}$-nek, ahol $0 \leq j < n_i$, ezek a gyökök páronként különbözőek, és nincs más gyöke $m^{(i)}$-nek. De $\alpha^{(i)}$ akkor és csak akkor gyöke c-nek, ha a $[0..n_i - 1]$ intervallumba eső legalább egy j kitevőre ${\alpha^{(i)}}^{q^j}$ gyöke c-nek, így az előbbi feltétel úgy is igaz, hogy c akkor és csak akkor kódpolinom a C kóddal, ha minden $m^{(i)}$ legalább egy-egy gyöke gyöke c-nek. Legyen $A^{(i)} = \left\{ {\alpha^{(i)}}^{q^j} \middle| n_i > j \in \mathbb{N} \right\}$, és $\{\alpha_l | n-k \geq t > l \in \mathbb{N}\} = B \subseteq A = \bigcup_{i=1}^{s} A^{(i)}$ g gyökeinek olyan halmaza, amely valamennyi $m^{(i)}$ legalább egy gyökét tartalmazza, vagyis $|B| = t \leq n-k$ és $\forall (s \geq i \in \mathbb{N}): B \cap A^{(i)} \neq \emptyset$, továbbá $\mathbf{H}$ egy olyan $t \times n$-méretű mátrix, amelyben a $0 \leq i < t$ és $0 \leq j < n$ indexekre $H_{i,j} = \alpha_i^j$, ahol $\alpha_i \in B$. Mivel c pontosan akkor kódszó, ha $0 = \hat{c}(\alpha_i) = \sum_{j=0}^{n-1} c_j \alpha_i^j = \sum_{j=0}^{n-1} H_{i,j} c_j = (\mathbf{Hc})_i$ minden $t > i \in \mathbb{N}$ indexre, ezért c akkor és csak akkor kódszó, ha $\mathbf{Hc} = \mathbf{0}$.

Mivel g osztója $x^n - e$-nek, ezért g valamennyi gyöke az $x^n - e$-nek is gyöke. Az utóbbi polinom gyökei $\mathbb{F}_q$ fölötti n-edik egységgyökök, és ha $(n,q) = 1$, akkor ezek a gyökök egy $\mathbb{F}_q$ fölötti α primitív n-edik egységgyök n-nél kisebb nemnegatív egész kitevős hatványai. Tegyük fel, hogy n és q relatív prímek. Legyen az A halmaz valamely B_1 részhalmaza olyan, hogy a B_1-ben lévő gyökök az α egymás után következő hatványai, vagyis $B_1 = \{\alpha^{\tau+l} | \delta - 1 > l \in \mathbb{N}\}$, ahol $\tau \in \mathbb{Z}$ tetszőleges egész, és $n - k + 1 \geq \delta \in \mathbb{N}$ (nem kötjük ki, hogy B_1 a g minden faktorának tartalmazza legalább egy gyökét), és legyen $\mathbf{H}$ olyan mátrix, amelyben az i-edik sor j-edik eleme $\left(\alpha^{\tau+i}\right)^j$, ahol $\delta - 1 > i \in \mathbb{N}$ és $n - 1 > j \in \mathbb{N}$. Ekkor $H_{i,j} = \left(\alpha^{\tau+i}\right)^j = \alpha^{\tau j}\left(\alpha^j\right)^i = h_j \beta_j^i$, ahol $h_j = \alpha^{\tau j}$ és $\beta_j = \alpha^j$. Ha most $\mathbf{H}'$ a $\mathbf{H}$ tetszőlegesen kiválasztott, páronként különböző $\delta - 1$ oszlopából álló részmátrix, akkor $\mathbf{H}'$ egy $\delta - 1$ -edrendű kvadratikus mátrix. Legyen D a $\mathbf{H}'$ determinánsa. A determináns j_l-indexű oszlopából, ahol $\delta - 1 > l \in \mathbb{N}$ és $n > j_l \in \mathbb{N}$, kiemelhető a 0-tól különböző h_{j_l}, és ha minden oszlopból kiemeltük ezt a szorzót, és a kiemelés utáni determinánst D'-vel jelöljük, akkor $D = D' \prod_{l=0}^{\delta-2} h_{j_l}$. Mivel a D' mögött álló szorzat nem nulla, ezért D pontosan akkor nulla, ha D' nulla. De D' egy olyan Vandermonde-determináns, amelyben a generátorelemek páronként különbözőek, hiszen $D'_{1,l} = \alpha^{j_l}$, az α primitív n-edik egységgyök és a j_l indexek csupa különböző, n-nél kisebb nemnegatív egészek. Ebből következik, hogy $\mathbf{H}'\mathbf{c} = \mathbf{0}$ csak úgy lehet, ha $\mathbf{c}$ a nullvektor, vagy $w(\mathbf{c}) \geq \delta$, és így a kód súlya legalább δ.

A BCH-kódot az előbbieknek megfelelően konstruáljuk. Legyen n a q prímhatványhoz relatív prím pozitív egész, τ tetszőleges egész, és δ 1-nél nagyobb egész, továbbá α egy $\mathbb{F}_q$ fölötti primitív n-edik egységgyök, végül g az $\alpha^{\tau+i}$-k $\mathbb{F}_q$ fölötti minimálpolinomjainak legkisebb közös többszöröse, ahol $\delta - 1 > i \in \mathbb{N}$. Ekkor a g által generált ciklikus kód dimenziója $k = n - \deg(g)$, és távolsága legalább δ, feltéve, hogy $k > 0$ (ellenkező esetben a kód csupán a nullvektort tartalmazza, így a kód távolsága nincs értelmezve). A konstrukcióból látszik, hogy ha τ-t tetszőleges, vele $\mathrm{modulo}\, n$ kongruens egésszel helyettesítjük, akkor ugyanazt a kódot kapjuk, továbbá ha δ nem kisebb n-nél, akkor g δ -tól függetlenül $x^n - e$, így kiköthetjük, hogy $n > \tau \in \mathbb{N}$ és $n \geq \delta \in \mathbb{N}$.

A BCH-kódnak $\delta - 1 > i \in \mathbb{N}$-re $\alpha^{\tau+i}$ gyöke, és a kód valamennyi gyöke az előbbi gyökök minimálpolinomjainak gyöke, tehát az $\mathbb{F}_q$ fölötti legfeljebb $n-1$-edfokú c polinom akkor és csak akkor eleme ennek a BCH-kódnak, ha az előbbi $\alpha^{\tau+i}$-k mindegyike gyöke c-nek, vagyis ha $\mathbf{Hc} = \mathbf{0}$, ahol $\mathbf{H}$ $(\delta - 1) \times n$-méretű, és $H_{i,j} = \left(\alpha^{\tau+i}\right)^j$ ($\delta - 1 > i \in \mathbb{N}$ és $n > j \in \mathbb{N}$). Hogy ennek a kódnak a

távolsága legalább δ, az azon múlott, hogy a $\mathbf{H}$ bármely $\delta - 1$-rendű kvadratikus részmátrixa, az oszlopokból egy-egy alkalmas nem nulla értéket kiemelve, reguláris Vandermonde-determinánst határoz meg. Ebből viszont következik, hogy ha $\mathbf{H}$ tetszőleges olyan, $r \times n$-méretű mátrix, amelynek bármely r-edrendű kvadratikus részmátrixához tartozó determinánsa az oszlopokból való alkalmas, nullától különböző elem kiemelése után reguláris Vandermonde-determináns, akkor mindazok az $\mathbb{F}_q$ fölötti n-dimenziós vektorok, amelyeknek $\mathbf{H}$-val vett szorzata $\mathbf{0}$, egy olyan kódot adnak, amelynek a távolsága legalább $r + 1$.

12.1. Definíció

Legyen m, r és az r-nél nagyobb n pozitív egész, $n > j \in \mathbb{N}$-re h_j az $\mathbb{F}_{q^m}$ nem nulla elemei, ugyanezen indexekre az α_j-k az $\mathbb{F}_{q^m}$ páronként különböző elemei, és $C = \{\mathbf{c} \in \mathbb{F}_q^n | \mathbf{Hc} = \mathbf{0}\}$, ahol $\mathbf{H}$ az $\mathbb{F}_{q^m}$ fölötti olyan $r \times n$-méretű mátrix, amelyben $H_{i,j} = h_j \alpha_j^i$ ($r > i \in \mathbb{N}, n > j \in \mathbb{N}$). Ekkor C egy $\mathbb{F}_q$ fölötti **alternáns kód**. Ezt az alternáns kódot $A(q, m, r, \mathbf{h}, \boldsymbol{\alpha})$, vagy ha q, m és r ismert, akkor $A(\mathbf{h}, \boldsymbol{\alpha})$ jelöli.

Δ

Egy BCH-kód nyilván alternáns: ha a kód hossza n, és a kezdő értéket megadó paraméter τ, akkor $h_j = \alpha^{\tau j}$ és $\alpha_j = \alpha^j$.

Az rögtön látszik, hogy az alternáns kód lineáris, és a kódszavak hossza n, továbbá a kód konstrukciója következtében a kód d távolsága legalább $r + 1$. Korábban azt is láttuk, hogy a fenti $\mathbf{H}$ mátrix, amennyiben $m > 1$, általában nem ellenőrző mátrixa a kódnak, hiszen $\mathbf{H}$ nem minden eleme van benne feltétlenül $\mathbb{F}_q$-ban. Ha megadjuk $\mathbb{F}_{q^m}$-nek egy $\mathbb{F}_q$ fölötti bázisát, akkor $\mathbb{F}_{q^m}$ minden eleme bijektíven megfeleltethető egy $\mathbb{F}_q$ fölötti m-dimenziós vektornak, ahol a vektor komponensei az $\mathbb{F}_{q^m}$ adott elemének az előbbi bázisban való felírásánál kapott együtthatói. Ha a $\mathbf{H}$ minden elemét helyettesítjük a neki megfelelő vektor oszlopmátrixával, akkor már egy $\mathbf{H}$ fölötti $mr \times n$-méretű mátrixunk lesz. Ennek a mátrixnak azonban lehetnek lineárisan összefüggő sorai. Kiválasztva maximális számú lineárisan független sort, az így kapott mátrix lesz a kód ellenőrző mátrixa. A lineárisan független sorok száma legalább r, hiszen az eredeti mátrix sorai lineárisan függetlenek, és így van benne r lineárisan független oszlop, de akkor az új mátrixnak is legalább r oszlopa lineárisan független, és maximum rm, így a kód k dimenziójára azt kapjuk, hogy $n - mr \leq k \leq n - r$, és C egy $\mathbb{F}_q$ fölötti, $[n, k, d]_q$-paraméterű kód. Meg kell jegyezni, hogy ez a kód általában nem ciklikus (ugyanakkor minden BCH-kód alternáns, vagyis léteznek ciklikus alternáns kódok).

Most egy másik kódot definiálunk, amelyről kiderül, hogy szintén alternáns. A definíció előtt szükségünk lesz egy egyszerű tényre. Ha g egy $\mathcal{K}$ test fölötti r-edfokú polinom, ahol r nagyobb nullánál, és $u \in K$ olyan, hogy $\hat{g}(u) \neq 0$, vagyis u nem gyöke g-nek, akkor g és $x - u$ relatív prímek. Test fölötti polinomgyűrű euklideszi, így létezik, és asszociálttól eltekintve egyértelmű a legnagyobb közös osztó. $x - u$ mint elsőfokú polinom, irreducibilis a $\mathcal{K}[x]$ gyűrűben, így az előbbi legnagyobb közös osztó, asszociálttól eltekintve, csupán e vagy $x - u$ lehet, ahol e a test egységeleme. De ha a legnagyobb közös osztó $x - u$, akkor $x - u$ osztója g-nek, ami ekvivalens azzal, hogy u gyöke g-nek. Mivel $\hat{g}(u) \neq 0$, ezért a legnagyobb közös osztó csak e lehet. Euklideszi gyűrűben a legnagyobb közös osztó felírható a megadott elemek olyan lineáris kombinációjaként, ahol az együtthatók a gyűrűből vannak, így létezik olyan $g^{(u)}$ és $t^{(u)}$ $\mathcal{K}$ fölötti polinom, hogy $e = g^{(u)} \cdot (x - u) + t^{(u)} g$. $g^{(u)}$ nem nulla, mert $r > 0$, és mindig megválasztható úgy, hogy a foka kisebb legyen, mint g foka. Ha ugyanis f_1 és f_2 nem konstans polinomok, amelyek relatív prímek, és $e = h_1 f_1 + h_2 f_2$, akkor $f_2 \neq 0$, így $h_1 = q f_2 + r$, ahol $r = 0$, vagy $r \neq 0$ és $\deg(r) < \deg(f_2)$. Innen behelyettesítés és átrendezés után $e = r f_1 + (q f_1 + h_2) f_2 = t_1 f_1 + t_2 f_2$, és $t_1 = r \neq 0$ ismét azért, mert f_2 nem konstans.

Az $e = g^{(u)} \cdot (x - u) + t^{(u)} g$ felírásból látjuk, hogy $g | e - g^{(u)} \cdot (x - u)$, amit úgy is írhatunk, hogy $e \equiv g^{(u)} \cdot (x - u)\ (g)$, vagy hogy $g^{(u)} \equiv (x - u)^{-1}\ (g)$, és azt mondjuk, hogy $g^{(u)} \cdot (x - u)$ kongruens e-vel modulo g, illetve hogy $g^{(u)}$ modulo g inverze $x - u$-nak. Általánosan, ha egy test fölötti polinomgyűrűben az f_1 és f_2 polinom különbsége osztható a g polinommal, akkor f_1 és

f_2 kongruens modulo g, és azt írjuk, hogy $f_1 \equiv f_2\ (g)$, míg ha $h_1 h_2 \equiv e\ (g)$, ahol a bal oldali szorzat két tényezője ugyanezen polinomgyűrű két eleme, és e a test egységeleme, akkor h_2 a h_1 modulo g inverze, jelölésben $h_2 \equiv h_1^{-1}\ (g)$.

Ezek után lássuk a kódot.

12.2. Definíció

Legyen m és n pozitív egész, $0 \neq g \in \mathbb{F}_{q^m}[x]$, $n > j \in \mathbb{N}$-re az α_j-k az $\mathbb{F}_{q^m}$ páronként különböző olyan elemei, amelyek nem gyökei g-nek, végül $C = \left\{\mathbf{c} \in \mathbb{F}_q^n \middle| \sum_{i=0}^{n-1} c_i (x - \alpha_i)^{-1} \equiv 0\ (g)\right\}$. Ekkor C egy $\mathbb{F}_q$ fölötti **Goppa-kód**, amelyet $\Gamma(q, m, g, \boldsymbol{\alpha})$, vagy röviden $\Gamma(g, \boldsymbol{\alpha})$ jelöl.

Δ

12.3. Tétel

$\mathbb{F}_q$ fölötti Goppa-kód $\mathbb{F}_q$ fölötti alternáns kód.

Δ

A tételt nem bizonyítjuk.

Az előbbiekből következik, hogy a fenti Goppa-kód távolsága legalább $r + 1$, a kód lineáris, és $n - mr \leq k \leq n - r$, ahol k a kód dimenziója.

Egy BCH-kódot **szűkebb értelemben vett** BCH-**kód**nak nevezünk, ha $\tau = 1$. Megmutatható, hogy a szűkebb értelemben vett BCH-kód Goppa-kód. Ugyanakkor egy BCH-kód általában nem Goppa-kód.

Szintén bizonyítás nélkül említjük, hogy bináris Goppa-kód távolsága akár $d = d(C) \geq 2r + 1$ is lehet.

Alternáns kódok dekódolására több módszer is létezik, mi az egyik leghatékonyabbat ismertetjük. Ehhez felhasználjuk az euklideszi algoritmus tulajdonságait.

Legyen C egy $A(q, m, r, \mathbf{h}, \boldsymbol{\alpha})$-paraméterű alternáns kód, ekkor a kód d távolsága legalább $r + 1$, és kössük ki, hogy $\boldsymbol{\alpha}$ egyetlen komponense sem 0. Legyen $t_0 = \left\lfloor \frac{r}{2} \right\rfloor$, ekkor C t_0 hibát biztosan javít. Legyen egy üzenet továbbításánál fellépő hibavektor $\boldsymbol{\varepsilon}$, és $1 \leq w(\boldsymbol{\varepsilon}) = t \leq t_0$, ahol $w(\boldsymbol{\varepsilon})$ a hibavektor súlya, vagyis a hibahelyek száma, és legyen a hibás pozíciók indexeinek halmaza $J = \{j_i \in \mathbb{N} | t > i \in \mathbb{N} \wedge n > j_i\}$. Jelölje $t > i \in \mathbb{N}$-re X_i α_{j_i}-t, és Y_i ε_{j_i}-t. Ha ismerjük az X_i-ket és Y_i-ket, akkor ismerjük a hibás pozíciókat, és a hibás helyeken a hiba értékét, így a javítás már elvégezhető. Kérdés, hogy hogyan tudjuk ezeket az értékeket meghatározni.

Legyen $\mathbf{H}$ az az $r \times n$-méretű mátrix, amelyben $0 \leq i < r$-re és $0 \leq j < n$-re $H_{i,j} = h_j \alpha_j^i$, $\mathbf{v}$ a vett szó az $\boldsymbol{\varepsilon}$ hibával, és $\mathbf{s} = \mathbf{Hv} = \mathbf{H\varepsilon}$ a szindróma, ekkor ismét $0 \leq i < r$-re

$$s_i = (\mathbf{H\varepsilon})_i = \sum_{j=0}^{n-1} H_{i,j} \varepsilon_j = \sum_{j \in J} H_{i,j} \varepsilon_j = \sum_{l=0}^{t-1} H_{i,j_l} \varepsilon_{j_l} = \sum_{l=0}^{t-1} h_{j_l} X_l^i Y_l.$$

A korábbi feltétel szerint a hibák t száma legalább 1 és legfeljebb t_0, így $\mathbf{s} \neq \mathbf{0}$. Definiálunk három polinomot. Legyen $\sigma = \prod_{i=0}^{t-1} (e - X_i x)$, ekkor látható, hogy $\hat{\sigma}(0) = e$, $\deg(\sigma) = t \leq \frac{r}{2}$, és $\hat{\sigma}(u) = 0$ akkor és csak akkor, ha $u = X_l^{-1}$ egy $t > l \in \mathbb{N}$ indexszel. A második polinomhoz legyen $t > i \in \mathbb{N}$-re $\sigma^{(i)} = \prod_{\substack{l=0 \\ l \neq i}}^{t-1} (e - X_l x)$, és $\omega = \sum_{i=0}^{t-1} h_{j_i} Y_i \sigma^{(i)}$. Az α_j-k, tehát az X_i-k is, páronként különbözőek, és egyikük sem nulla, így $\sigma^{(i)}$ gyökeinek halmaza $\left\{X_j^{-1} \middle| t > j \in \mathbb{N} \wedge j \neq i\right\}$, és

$$\hat{\omega}\left(X_i^{-1}\right) = \sum_{l=0}^{t-1} h_{j_l} Y_l \hat{\sigma}^{(l)}\left(X_i^{-1}\right) = h_{j_i} Y_i \hat{\sigma}^{(i)}\left(X_i^{-1}\right).$$

$\hat{\sigma}^{(i)}\left(X_i^{-1}\right) \neq 0$, továbbá a kód definíciója alapján $h_{j_i} \neq 0$, ezért $Y_i = \frac{\hat{\omega}\left(X_i^{-1}\right)}{h_{j_i}\hat{\sigma}^{(i)}\left(X_i^{-1}\right)}$. $Y_i \neq 0$ is igaz, tehát $\hat{\omega}\left(X_i^{-1}\right) \neq 0$, amiből következik, hogy σ és ω relatív prímek. Mivel feltettük, hogy van hiba, és így legalább egy indexre $Y_i \neq 0$, ezért ω nem lehet a nullpolinom. ω egy olyan összeg, amelynek minden tagja egy $t-1$-edfokú polinom, így $\deg(\omega) \leq t-1$. σ a **hibahelypolinom**, míg ω a **hibaérték-polinom**. Az elnevezések érthetőek: ha ismerjük σ-t, akkor a polinom gyökeinek inverzei megadják a hibahelyeket, míg ω ismeretében meghatározhatjuk a hiba értékét.

A harmadik polinom, $S = \sum_{i=0}^{r-1} s_i x^i$, a legfeljebb $r-1$-edfokú szindrómapolinom.

Most belátjuk az egész eljárás lényegét jelentő $\omega \equiv \sigma S\ (x^r)$ kongruenciát.

$$\begin{aligned}
\omega - \sigma S &= \sum_{l=0}^{t-1} h_{j_l} Y_l \sigma^{(l)} - \sum_{i=0}^{r-1} \sigma s_i x^i = \sum_{l=0}^{t-1} h_{j_l} Y_l \sigma^{(l)} - \sum_{i=0}^{r-1} \sigma \sum_{l=0}^{t-1} h_{j_l} Y_l X_l^i x^i \\
&= \sum_{l=0}^{t-1} h_{j_l} Y_l \sigma^{(l)} - \sum_{l=0}^{t-1} \sigma h_{j_l} Y_l \sum_{i=0}^{r-1} (X_l x)^i = \sum_{l=0}^{t-1} h_{j_l} Y_l \sigma^{(l)} - \sum_{l=0}^{t-1} \sigma h_{j_l} Y_l \frac{e-(X_l x)^r}{e - X_l x} \\
&= \sum_{l=0}^{t-1} h_{j_l} Y_l \sigma^{(l)} - \sum_{l=0}^{t-1} h_{j_l} Y_l \sigma^{(l)} (e - X_l x) \frac{e-(X_l x)^r}{e - X_l x} = \left(\sum_{l=0}^{t-1} h_{j_l} Y_l \sigma^{(l)} X_l^r \right) x^r,
\end{aligned}$$

vagyis $\omega - \sigma S$ osztható x^r-rel, és ez éppen az említett kongruencia. A fenti kongruencia ekvivalens az $\omega = \vartheta x^r + \sigma S$ polinomegyenlőséggel, ahol ϑ egy alkalmas polinom. Ebben az egyenlőségben ismert x^r és S. Ha ismerjük ω-t, akkor σ az euklideszi algoritmus segítségével meghatározható, azonban ω-t nem ismerjük. Az alábbi tételből azonban kiderül, hogy S ismeretében mind σ, mind ω előállítható.

12.4. Tétel

Legyen $r_{-1} = x^r$, $r_0 = S \neq 0$, és az euklideszi algoritmus szerint $b_{-1} = 0$, $b_0 = e$, valamint $r_{i-1} = q_i r_i + r_{i+1}$ és $b_{i+1} = b_{i-1} - q_i b_i$, amíg $\deg(r_i) \geq \frac{r}{2}$. Ha r_k az első maradék, amelynek a fokszáma kisebb, mint $\frac{r}{2}$, akkor $\hat{b}_k(0) \neq 0$, és $\sigma = \left(\hat{b}_k(0)\right)^{-1} b_k$ és $\omega = \left(\hat{b}_k(0)\right)^{-1} r_k$.

Δ

Bizonyítás:

Láttuk, hogy ω felírható az x^r és S polinom-együtthatós lineáris kombinációjaként, vagyis megoldható az $\omega = x^r y + Sz$ egyenlet. A megoldhatóságból következik, hogy x^r és S legnagyobb közös osztója osztója ω-nak, és mivel ω nem nulla, ezért a legnagyobb közös osztó fokszáma nem nagyobb ω fokszámánál, vagyis kisebb, mint $\frac{r}{2}$. Ekkor van olyan $l \in \mathbb{N}$, hogy r_l foka kisebb, mint $\frac{r}{2}$. Mivel a maradékok fokszáma szigorúan monoton csökken, ezért az ilyen l indexek halmazában van egyértelműen meghatározott legkisebb k index, tehát a tételben megfogalmazott feltételnek megfelelő r_k polinom létezik és egyértelmű. Most nézzük az alábbi két egyenletet:

$$\begin{aligned}
\omega &= \vartheta x^r + \sigma S \\
r_k &= a_k r^k + b_k S
\end{aligned}$$

(a_i-t az $a_{-1} = e$, $a_0 = 0$, $a_{i+1} = a_{i-1} - q_i a_i$ rekurzió határozza meg). A fenti két egyenletből

$$b_k\omega - r_k\sigma = (b_k\vartheta - a_k\sigma)x^r.$$

Ha a jobb oldalon zárójelben álló polinom nem nulla, akkor a jobb oldali polinom legalább r-edfokú. $\deg(b_k) = \deg(x^r) - \deg(r_{k-1}) \leq r - \frac{r}{2} = \frac{r}{2}$, hiszen k a legkisebb l index, amelyre r_l foka kisebb $\frac{r}{2}$-nél, $\deg(\omega) < t \leq \frac{r}{2}$, $\deg(r_k) < \frac{r}{2}$ és $\deg(\sigma) = t \leq \frac{r}{2}$. Ebből $\deg(b_k\omega) < \frac{r}{2} + \frac{r}{2} = r$ és hasonlóan $\deg(r_k\sigma) < \frac{r}{2} + \frac{r}{2} = r$, tehát ha a bal oldali polinom nem nulla, akkor a fokszáma r-nél kisebb, ami lehetetlen, hiszen az előbb láttuk, hogy a jobb oldalon legalább r-edfokú polinom áll. Ebből következik, hogy az egyenlet mindkét oldalán nulla áll, és így a bal oldali illetve a jobb oldalon a zárójelben álló különbség nulla, és így

$$b_k\omega = r_k\sigma$$
$$b_k\vartheta = a_k\sigma.$$

Figyelembe véve, hogy egyrészt a_k és b_k, másrészt σ és ω relatív prím, a fenti egyenlőségekből következik, hogy σ osztója b_k-nak és b_k osztója σ -nak, vagyis b_k és σ asszociáltak, és ha $\sigma = cb_k$, akkor c egy nem nulla konstans polinom és $\omega = cr_k$. A $\sigma = cb_k$ egyenlőségből $e = \hat{\sigma}(0) = c\hat{b}_k(0)$, tehát $\hat{b}_k(0) \neq 0$, majd innen $c = \left(\hat{b}_k(0)\right)^{-1}$, és végül $\sigma = \left(\hat{b}_k(0)\right)^{-1} b_k$ és $\omega = \left(\hat{b}_k(0)\right)^{-1} r_k$.

□

12.5. Megjegyzés

Bár az algoritmus szempontjából nem lényeges, azért megjegyezzük, hogy S nem osztója x^r-nek, és így $S \neq 0$ esetén $k > 0$. Ha $S = 0$, akkor nyilván igaz, hogy S nem osztja az x^r polinomot, ezért tegyük fel, hogy $S \neq 0$. Ekkor sem σ, sem ω nem 0, sőt, σ legalább elsőfokú. σ foka nagyobb, mint ω foka, és így σS foka még inkább meghaladja a hibaérték-polinom fokszámát, ezért $\omega - \sigma S$ nem a nullpolinom, és a foka σS fokával azonos. $\omega - \sigma S$ osztható x^r-rel, amiből következik, hogy σS foka legalább r, és akkor S foka legalább $\frac{r}{2}$, hiszen σ foka azonos a hibák számával, amiről feltettük, hogy legfeljebb $\frac{r}{2}$. Ha S osztója x^r-nek, akkor osztója $\omega - \sigma S$ -nek, tehát ω -nak is, ami lehetetlen, hiszen ekkor S foka nem haladhatná meg ω fokát, ami viszont kisebb, mint σ foka, tehát kisebb, mint $\frac{r}{2}$. Mellékeredményként azt kaptuk, hogy ha van hiba, de a hibák száma nem haladja meg $\frac{r}{2}$-t, akkor az egyébként legfeljebb $r - 1$-edfokú S polinom foka legalább $\frac{r}{2}$.

□

13. A DES

A klasszikus rendszerek két nagy módszert alkalmaztak. Az egyik a **helyettesítés**, amikor egy-egy betűt, vagy a betűk egy csoportját helyettesítik valamilyen jellel, vagy jelcsoporttal, míg a másiknál az üzenet egy-egy meghatározott hosszúságú szakaszán megváltoztatják a betűk sorrendjét, vagyis **transzpozíció**t alkalmazunk. Ha nagy redundanciájú üzeneteket sifrírozunk, akkor elegendően hosszú üzenet esetén a kulcs könnyen megfejthető. Igencsak meglepő, hogy ha az úgynevezett **egyszerű**, vagy más néven **monoalfabetikus helyettesítés**t alkalmazzuk, vagyis ha mindenegyes betűt ugyanazon szabállyal helyettesítünk, akkor a lehetséges helyettesítések, tehát a különböző kulcsok száma a 26-betűs angol ábécé alkalmazása esetén $403\,291\,461\,126\,605\,635\,584\,000\,000$, így az ember azt gondolná, hogy szinte lehetetlen megállapítani az aktuális kulcsot. A valóság viszont az, hogy ha a rejtjelezett szöveg egy tipikus, hétköznapi szöveg, akkor egy körülbelül 20 betűs szöveg egyértelműen visszafejthető a kulcs előzetes ismerete nélkül. A fejtés alapja a betűfrekvencia. Minden nyelvre jellemző, hogy az egyes betűk milyen gyakorisággal fordulnak elő. Ha monoalfabetikus helyettesítést alkalmazunk, akkor a szöveg betűi a sifrírozás során grafikusan megváltoznak, de nem változik meg az eredeti szövegben lévő előfordulásuk gyakorisága, és ezt lehet kihasználni a fejtéshez. Ha csak az egyes betűk gyakoriságát nézzük, akkor körülbelül 80 betűs szöveg kell az egyértelmű fejtéshez, ám nézhetünk egyéb jellemző tulajdonságokat is. Ilyen például a betűk egymásra következésének gyakorisága, a kettős betűk gyakorisága, vizsgálhatjuk a szókezdő és a szó végén álló betűk gyakoriságát (feltéve, hogy a rejtjelezés során nem gyomlálták ki az árulkodó szóközöket), stb. Ha mindezt figyelembe vesszük, akkor alakul ki a már említett körülbelül 20 betűs szöveg fejthetősége. Nehezíti a fejtést, ha tömörítünk. Ha például számsorozatokat rejtjelezünk, amikor is bármely sorozat értelmes üzenet lehet, vagyis ha a redundancia 0, akkor ilyen kapaszkodónk nincs a fejtéshez.

A módszert úgy lehet bonyolítani, ha vagy más és más szabállyal végezzük az egyes pozíciókon a helyettesítést – ez a **többábécés**, vagyis másként a **polialfabetikus helyettesítés** – , vagy sok betűből álló blokkokat helyettesítünk, ami a **blokk-kód**ok alapja. Az előbbi szélső esete a *Vernam* által javasolt **véletlen átkulcsolás**. Ez olyan eljárás, ahol minden pozícióhoz abszolút véletlenül választjuk a helyettesítési szabályt (ez a további üzenetekre is érvényes, vagyis az egyszer elkezdett véletlen sorozatot kell folytatni, nem lehet újra kezdeni a generálást). Szemléletesen is belátható, hogy ez a titkosítás elvileg is fejthetetlen, hiszen bármely eredeti betűhöz, és tetszőleges rejtjelbetűhöz van olyan transzformáció, amely az előbbit az utóbbiba alakítja, és minden helyettesítés azonos valószínűséggel kerülhetett alkalmazásra, vagyis a rejtjelezett szövegből egyenlő valószínűséggel bármilyen eredeti szöveg előállítható. A pontos matematikai bizonyítást – ami lényegében véve semmivel nem nehezebb az előbbi gondolatmenetnél – *Shannon* végezte el. Ennél a módszernél természetesen fokozottan igaz, hogy igen nehéz a kulcs biztonságos kicserélése a két fél között, ám mégis alkalmazták a gyakorlatban, nevezetesen a Moszkva és Washington közötti *forró drót*on.

A gyakorlatban inkább a blokk-kódok terjedtek el, amelyeknek egyik leghíresebb képviselője a **DES** (***D**ata **E**ncryption **S**tandard*), amelyet 1977-ben fogadtak el szabványként, és egészen 2002-ig volt szabvány, ekkor váltotta fel az **AES** (***A**dvanced **E**ncryption **S**tandard*), ám amelyet főleg a háromszor egymás után alkalmazott formájában még ma is igen széles körben alkalmaznak. A *DES* egy igen fontos jellemzője, hogy jóllehet elvileg bármely rejtjelrendszer esetén felteszik, hogy maga az algoritmus ismert, ez volt a világ első olyan rejtjelező algoritmusa, amelyet hivatalosan nyilvánosságra hoztak (bár vannak, akik ezt nem akarják elhinni, és feltételezik, hogy a rendszert kifejlesztő *IBM* bizonyos információt megtartott magának, amelynek a segítségével képes fejteni a titkosított üzeneteket). A *DES* jelentőségét még az is alátámasztja, hogy az azóta kifejlesztett blokkos rejtjelező rendszerek majd mindegyike többé-kevésbé a *DES*-nél alkalmazott elvekre, de legalábbis az elvek egy részére épül.

A *DES* három pilléren nyugszik.

1. Tegyük fel, hogy egy r-betűs ábécével írt n-betűs blokkot p-hosszúságú kulccsal rejtjelezünk (maga a kulcs a szöveggel azonos ábécéből épül fel). Ekkor egy blokk kiszámítása lényegében véve egy $n + p$ változótól függő, n egyenletből álló egyenletrendszer:

$$n \geq i \in \mathbb{N}^+ : c_i = h_i(m_1, \cdots, m_n; k_1, \cdots, k_p),$$

ahol m_j a nyílt szöveg egy blokkjának j-edik, k_l a kulcs l-edik, és c_i a rejtjelezett szöveg blokkjának i-edik betűje. Ha r egy prímhatvány, akkor a szimbólumhalmaz egy véges testnek tekinthető. Ilyen esetben bármely leképezés, amely a véges testet önmagába képezi, megadható egy polinommal, így feltehetjük, hogy a h_i leképezés az f_i polinomhoz tartozó $\widehat{f_i}$ polinomfüggvény. Hasonló a helyzet a desifrírozás esetén:

$$n \geq i \in \mathbb{N}^+ : m_i = \hat{g}_i(c_1, \cdots, c_n; k_1, \cdots, k_p).$$

Ha az algoritmus nyilvános, akkor ismertek a polinomok, és ekkor a fejtés egyszerű behelyettesítés, ám a kulcs ismerete nélkül a feladat az eredeti polinomok által meghatározott egyenletrendszer megoldását jelenti. Ha a polinomok nem lineárisak, akkor viszont az ilyen egyenletrendszer megoldása **NP**-nehéz, és így elegendően nagy blokkok esetén a fejtés – bár elméletileg lehetséges – gyakorlatilag a használható időn belül reménytelen feladat.

2. *Shannon* szerint önmagában sem a helyettesítés, sem a transzpozíció nem nyújt kellő védelmet – leszámítva a véletlen átkulcsolást, amely viszont egészen extrém alkalmazásoktól eltekintve a gyakorlatban alkalmazhatatlan. *Shannon* az úgynevezett keverő transzformáció többfordulós alkalmazását javasolta. Itt egy-egy forduló egy kulcstól függő helyettesítésből és egy transzpozícióból áll. A nehézséget az okozza, hogy ha nagy a blokkméret – márpedig a megfelelő titkossághoz elegendően nagy blokkméretre van szükség – , akkor nehéz a helyettesítő táblázatok tárolása (egyszerűen számítható helyettesítés nem jöhet szóba, hiszen azt bárki könnyen tudná fejteni). *Shannon* ezt úgy javasolta megoldani, hogy a blokkot több kisebb részblokkra bontotta, és ezekre a kisebb részekre alkalmazta a helyettesítést. A dologban lényeges, hogy az utána következő transzpozíció az előbbi fordulóban egy részblokkban elhelyezkedő betűket különböző blokkba helyezi. A leképezést úgy alakítják ki, hogy érvényesüljön az úgynevezett **lavinahatás**, vagyis ha a bemeneten egyetlen betűt megváltoztatunk, akkor a kimeneten, azaz a rejtjelezett szövegben a teljes blokk betűinek a fele változzon, de úgy, hogy a kimenet minden betűje ½ valószínűséggel változzon, továbbá a kimenet bármely betűje a bemenet valamennyi betűjétől függjön, és egy kimeneti betű megváltozásából ne lehessen következtetni a bemeneti változásra.

3. A *DES* az úgynevezett **Feistel-struktúra** alapján működik. Legyen a rejtjelezendő szöveg m, amelynek hossza $2n$, és válasszuk két részre úgy, hogy az egyik rész a szöveg első n betűjéből, míg a másik a hátsó n betűből áll, azaz az előbbit $m^{(0)}$-lal, a másodikat $m^{(1)}$-gyel jelölve $m = m^{(0)} \parallel m^{(1)}$ ($\parallel$ most és a későbbiekben a konkatenációt jelöli). Tegyük továbbá fel, hogy az algoritmus t **fordulóból**, és mindenegyes fordulóban az eredeti kulcsból származtatott alkulcsot alkalmazunk. A *Feistel*-struktúrában alkalmazott transzformáció ezek után a következő:

$$t \geq i \in \mathbb{N}^+ : m^{(i+1)} = m^{(i-1)} + f(m^{(i)}, k^{(i)}).$$

Az m-hez tartozó rejtjelezett szöveg $c = m^{(t+1)} \parallel m^{(t)}$. A kulcs ismeretében a visszafejtés rendkívül egyszerű, hiszen c-ből ismert $m^{(t+1)}$ és $m^{(t)}$, és ha ismerjük valamilyen $t \geq j \in \mathbb{N}^+$-ra $m^{(j+1)}$-et és $m^{(j)}$-t, akkor $m^{(j+1)} - f(m^{(j)}, k^{(j)}) = m^{(j-1)}$, vagyis c-ből meg tudjuk határozni $m^{(1)}$-et és $m^{(0)}$-t, tehát m-et. Látható, hogy a rejtjelező és a visszafejtő algoritmus csak annyiban tér el, hogy az egyikben összeadás, a másikban kivonás áll (ami bináris esetben egyébként megegyezik), és a kulcsokat fordított sorrendben kell alkalmazni. Igen lényeges tulajdonsága a Feistel-struktúrának, hogy f nem feltétlenül invertálható, amely tulajdonság nagyon megkönnyíti a rejtjelezés szempontjából jó tulajdonságú függvény keresését.

Konkrétan a *DES* esetén az ábécé két betűből, a 0-ból és az 1-ből áll, és a blokkméret 64 bit. A kulcs is 64 bites, de ebből 8 bit ellenőrző funkciót lát el, így valójában a kulcs 56 bites (ezt vetették

leginkább a DES szemére, és sokan úgy vélték, hogy azért választották ilyen méretűre a kulcsot, mert a titkosszolgálatok a maguk számítástechnikai apparátusaikkal abban az időben ekkora méretekkel boldogultak). Az eljárás 16-fordulós, és a kulcsból úgy állítják elő az egyes fordulókhoz az aktuális alkulcsot, hogy a 16. forduló után éppen visszanyerik az eredeti kulcsot (ez inkább a hardveres megoldás szempontjából jelent némi előnyt). A bináris ábécé esetén, mint fentebb már említettük, a kivonás megegyezik az összeadással, így a desifrírozás teljes egészében megegyezik a sifrírozással, csupán a kulcsokat kell fordított sorrendben alkalmazni.

A mai számítástechnikai eszközökkel a *DES* fejtése könnyű feladat, ezért különböző kulccsal többször egymás után alkalmazzák. Egy rejtjelező rendszer több kulccsal való iterációja csak akkor eredményezhet az egy kulccsal való titkosításnál erősebb védelmet, ha a leképezések kompozíciója nem alkot csoportot, vagyis ha két egymás utáni titkosítás nem állítható elő valamely kulccsal történő egylépéses leképezésként. Ez a *DES* esetén teljesül. Kevés számolással kimutatható, hogy a kétszeres *DES* sem nyújt ma már kellő védelmet, ám a háromszoros *DES* biztonságosnak mondható, és igen sok helyen alkalmazzák ma is.

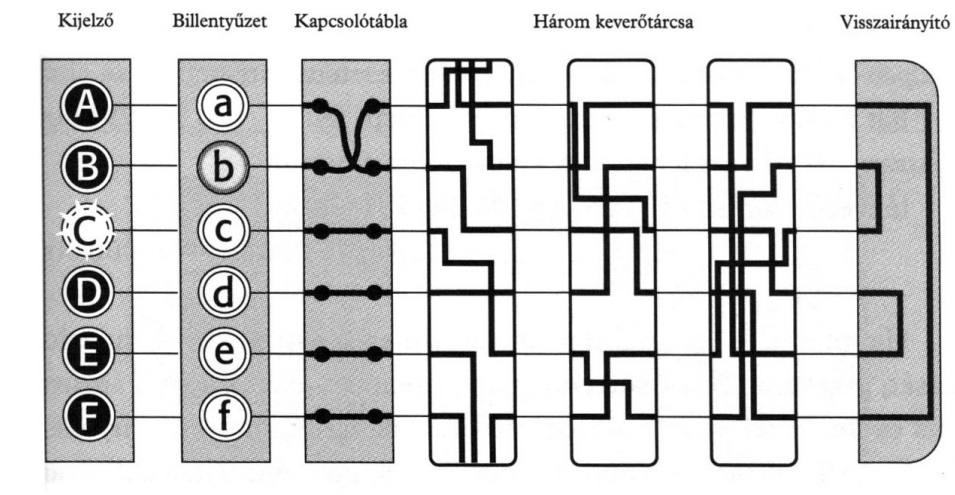
A blokkos rejtjelezéseket, és így a *DES*-t is, különböző üzemmódban alkalmazzák, amelyek még növelik a rendszer hatékonyságát, biztonságát. Egy ilyen az úgynevezett **CBC**-mód (***C**ipher **B**lock **C**haining*), a **blokkláncolás**. Ennél az üzemmódnál választunk egy c_0 kezdőblokkot, és ebből kiindulva egymás után képezzük a $c_i = E_k(m_i, c_{i-1})$ blokkokat, ahol E_k a k kulcstól függő rejtjelező leképezés. A szabályból látszik, hogy a rejtjelszöveg i-edik blokkja nem csupán a nyílt szöveg i-edik blokkjától, hanem valamennyi korábbi blokktól is függ.

14. Az ENIGMA

A rejtjelező – visszafejtő tevékenység gépesíthető is, így bonyolultabb eljárások alkalmazhatóak. Ennek egyik példája az **Enigma**.

Az *Enigma*[1] egy olyan elektromos írógép, melynek három főbb egysége van: egy billentyűzet a nyílt szöveg betűinek bevitelére, egy átalakító egység, amely a nyílt szöveg betűit a rejtjeles szöveg megfelelő betűivé alakítja, és egy kijelző panel, amelyen kis lámpácskák felvillanása jelzi a rejtjeles szöveg betűit. A felvillanó betűket leírva kapjuk meg a rejtjeles szöveget, amelyet azután rádión továbbítottak a címzettnek. A vevő oldalon egy azonos beállítású enigmával írták le a szöveget, és a lámpák felvillanása adta vissza a nyílt üzenetet.

A gép legfontosabb része az átalakító egység, amely három kivehető keverőtárcsából áll, így ezek cserélhetőek. A keverőtárcsa egy vezetékekkel sűrűn teleszőtt, vastag gumitárcsa. A nyílt szöveg betűinek sifrírozását a keverőtárcsák belső huzalozása határozza meg. Ha a tárcsák fix helyzetűek lennének, akkor a tárcsák huzalozása egy egyszerű egyábécés helyettesítéses eljárást valósítana meg. *Sherbius* gépének viszont a legfontosabb jellemzője, hogy a keverőtárcsák forognak. Minden egyes betű sifrírozása után az első tárcsa $1/26$-nyival elfordul (26 betűs ábécé esetén). A második tárcsa csak akkor fordul $1/26$-nyit, ha az első tárcsa megtett egy teljes fordulatot, a harmadik tárcsa akkor fordul $1/26$-nyit, ha a második tárcsa megtett egy teljes fordulatot, miközben az első tárcsa már 26×26-szor fordult $1/26$-nyit. Ez a mechanizmus hasonlít az autók kilométerórájához. A rotáció révén a gép többábécés helyettesítéses eljárás megvalósítására használható. A három keverőtárcsa kezdeti beállítása $26 \times 26 \times 26 = 17\,576$ különböző kulcsnak felel meg. Az ábrán az *Enigma* kétdimenziós ábrázolása látható, az áttekinthetőség kedvéért hatbetűs ábécé esetén. A keverőtárcsa egybetűnyi elfordulása során a tárcsákat összekötő vezetékek egy hellyel lentebb kerülnek.



Enigma kapcsolási rajza

A kapcsolási rajzon még két szerkezeti elem is látható. A visszairányító szintén egy belső huzalozású gumitárcsa, de nem forog. Mikor a kezelő begépel egy betűt, azzal egy elektromos jelet küld át a három keverőtárcsán. A visszairányító ezt a beérkező jelet küldi vissza, de más útvonalon. Az ábrán látható tárcsaállások esetén a leütött leütött b betű a C-t villantja fel, ha azonban a c-t ütöttük volna le, akkor a kijelzőn a B villant volna fel. Ebből látható a visszairányító szerepe: a gép a nyílt szöveg egyik betűjét a rejtjeles szöveg egyik betűjévé alakítja, és ha egy másik gép ugyanígy van beál-

[1] Az *Enigma*-ra vonatkozó részt **Tóthné Mészáros Ágnes Rejtjelezés a középiskolában** című szakdolgozatából vettem át.

lítva, akkor az előbb megkapott rejtjeles szöveg betűjét leütve megkapjuk az eredeti nyílt szöveg betűjét, vagyis a sifrírozáshoz és a desifrírozáshoz ugyanaz a gép szükséges megegyező kezdő beállítással!

A másik új elem az ábrán a kapcsolótábla, mely a billentyűzet és az első keverőtárcsa közé van iktatva. E kapcsolótábla lehetővé teszi, hogy beiktassunk néhány vezetéket, amelyek még az első keverőtárcsába való belépés előtt felcserélnek bizonyos betűket. Az *Enigma* kezelőjének hat ilyen vezetéke van, miáltal hat betűpárt tud felcserélni a huszonhatból. Egy 26-betűs ábécé esetenkénti hat betűpárjának felcserélési lehetőségeinek száma

$$\frac{\prod_{k=0}^{5}\binom{26-2k}{2}}{6!} = 100\,391\,791\,500.$$

A gép alapbeállításához tartozik még a keverőtárcsák sorrendje is. Scherbius úgy szerkesztette meg gépét, hogy a keverőtárcsák sorrendjét meg lehessen változtatni, a keverőtárcsák kivehetőségével. A három tárcsa hatféleképpen helyezhető a gépbe, így a lehetséges kezdőbeállítások, vagyis kulcsok száma:

keverőtárcsák beállítása (minden tárcsa 26-féle pozícióba állítható):	17 576
keverőtárcsák sorrendje:	6
kapcsolótábla beállításai:	100 391 791 500
összesen (előző három tényező szorzata):	10 586 916 764 424 000

A rejtjelezés kulcsát (a gép kezdőbeállítását) naponta változtatták, amit a négyhetente szétosztott 28 kulcsot tartalmazó kódkönyv határozott meg. A kapcsolótábla eredményezi a kulcsok számának legnagyobb növekedését, de a sifrírozás megkezdése után már nem változik beállítása, így egyedüli alkalmazása olyan rejtjeles szöveget generálna, amely gyakorisági elemzéssel megfejthető. A keverőtárcsák kevesebb számú kulccsal járulnak hozzá a végeredményhez, de beállításuk folyamatosan változik, aminek eredményeképpen a rejtjeles szöveg gyakorisági elemzéssel nem fejthető meg. Mivel a rejtjelezés során a gép 17 576 különböző sifre-ábécét használ, Babbage módszerével sem fejthető meg a rejtjeles szöveg.

15. A rejtjelezés információelméleti alapjai

Most nézzük meg, hogy hogyan alkalmazhatóak az előbbi eredmények a kriptográfiában.

A továbbiakban $S = (\mathcal{M}, \mathcal{C}, \mathcal{K}, E, D)$ egy kriptorendszer, A_M a nyílt és A_C a titkosított szövegekhez alkalmazott ábécé, és $\mathcal{M}^{(n)}$ illetve $\mathcal{C}^{(n)}$ az n-betűs nyílt és titkosított szövegek halmaza, Pr_M egy eloszlás az $\mathcal{M}$ halmazon egy adott nyelv mellett, Pr_K egy eloszlás a $\mathcal{K}$ halmazon, míg $Pr_{M\times K}$ az indukált eloszlás $\mathcal{M} \times \mathcal{K}$-n: $Pr(m,k) = Pr_{M\times K}(m,k) = Pr_M(m)Pr_K(k)$ (mert m és k függetlenek). Minden $m \in \mathcal{M}$-re m ismerete előtt új $k \in \mathcal{K}$ kulcsot választunk, így k választása független m-től.

Adott $m \in \mathcal{M}$ és $k \in \mathcal{K}$ egyértelműen meghatároz egy és csak egy $E_k(m) = c \in \mathcal{C}$-t, és (az injektivitás miatt) hasonlóan, adott $c \in \mathcal{C}$-hez és $k \in \mathcal{K}$-hoz van egy és csak egy olyan $m \in \mathcal{M}$, amellyel $D_k(c) = m \in \mathcal{M}$. Ebből következik, hogy $H(\mathcal{M}|\mathcal{K},\mathcal{C}) = 0 = H(\mathcal{C}|\mathcal{K},\mathcal{M})$.

15.1. Definíció

$H(\mathcal{M}|\mathcal{C})$ és $H(\mathcal{K}|\mathcal{C})$ az S **üzenet-ekvivokáció**ja illetve **kulcs-ekvivokáció**ja.

Δ

A kulcs-ekvivokáció azt méri, mennyi információ nyerhető a kulcsról a rejtjel ismeretében.

15.2. Tétel

$H(\mathcal{K}|\mathcal{C}) = H(\mathcal{M}|\mathcal{C}) + H(\mathcal{K}|\mathcal{M},\mathcal{C})$.

Δ

Bizonyítás:
$H(\mathcal{K}|\mathcal{C}) = H(\mathcal{M},\mathcal{K}|\mathcal{C}) - H(\mathcal{M}|\mathcal{K},\mathcal{C}) = H(\mathcal{M},\mathcal{K}|\mathcal{C}) = H(\mathcal{M}|\mathcal{C}) + H(\mathcal{K}|\mathcal{M},\mathcal{C})$.

□

Mivel diszkrét rendszerekben minden entrópia nem negatív, ezért igaz az alábbi.

15.3. Következmény

$H(\mathcal{K}|\mathcal{C}) \geq H(\mathcal{M}|\mathcal{C})$.

Δ

A fenti következmény szerint a kulcs-ekvivokáció legalább akkora, mint az üzenet-ekvivokáció.

15.4. Tétel

$H(\mathcal{K}|\mathcal{C}) = H(\mathcal{K}) + H(\mathcal{M}) - H(\mathcal{C})$.

Δ

Bizonyítás:
$H(\mathcal{C},\mathcal{K},\mathcal{M}) = H(\mathcal{K},\mathcal{M}) + H(\mathcal{C}|\mathcal{K},\mathcal{M}) = H(\mathcal{K},\mathcal{M})$. $\mathcal{K}$ és $\mathcal{M}$ független, így $H(\mathcal{K},\mathcal{M}) = H(\mathcal{K}) + H(\mathcal{M})$, és innen $H(\mathcal{C},\mathcal{K},\mathcal{M}) = H(\mathcal{K}) + H(\mathcal{M})$. Hasonlóan $H(\mathcal{C},\mathcal{K},\mathcal{M}) = H(\mathcal{K},\mathcal{C})$. Mindent egybevetve kapjuk, hogy

$$H(\mathcal{K}|\mathcal{C}) = H(\mathcal{K},\mathcal{C}) - H(\mathcal{C}) = H(\mathcal{C},\mathcal{K},\mathcal{M}) - H(\mathcal{C})$$
$$= H(\mathcal{K}) + H(\mathcal{M}) - H(\mathcal{C}).$$

□

Amennyiben $H(\mathcal{M}) = H(\mathcal{C})$, akkor a $H(\mathcal{K}|\mathcal{C})$ kulcs-ekvivokáció megegyezik a kulcs *a priori* bizonytalanságával, $H(\mathcal{K})$-val. Mint később majd látjuk, ez a helyzet a tökéletes rendszereknél.

Az üzenet-ekvivokációra vonatkozó kifejezés hasonló:

$$H(\mathcal{C}) + H(\mathcal{M}|\mathcal{C}) = H(\mathcal{M},\mathcal{C}) = H(\mathcal{M}) + H(\mathcal{C}|\mathcal{M}),$$

és ebből

$$H(\mathcal{M}|\mathcal{C}) = H(\mathcal{M}) + H(\mathcal{C}|\mathcal{M}) - H(\mathcal{C}).$$

15.5. Tétel

$I(\mathcal{M},\mathcal{C}) \geq H(\mathcal{M}) - H(\mathcal{K})$.

Δ

Bizonyítás:
$H(\mathcal{M}|\mathcal{C}) \leq H(\mathcal{K}|\mathcal{C}) \leq H(\mathcal{K})$, és így $I(\mathcal{M},\mathcal{C}) = H(\mathcal{M}) - H(\mathcal{M}|\mathcal{C}) \geq H(\mathcal{M}) - H(\mathcal{K})$.

□

15.6. Definíció (Tökéletes titkosság)

S akkor és csak akkor garantál **tökéletes titkosság**ot, ha $\forall(m \in \mathcal{M})\forall(c \in \mathcal{C})$: $Pr(m|c) = Pr(m)$.

Δ

A definíció ekvivalens azzal, hogy $I(\mathcal{M},\mathcal{C}) = 0$, vagyis $H(\mathcal{M}|\mathcal{C}) = H(\mathcal{M})$. Ez pontosan azt jelenti, hogy $\mathcal{M}$ és $\mathcal{C}$ egymástól teljesen függetlenek. Valóban:

$$\begin{aligned} I(\mathcal{M},\mathcal{C}) &= H(\mathcal{M}) - H(\mathcal{M}|\mathcal{C}) \\ &= -\sum_m Pr(m)\log Pr(m) + \sum_m\sum_c Pr(m,c)\log Pr(m|c) \\ &= -\sum_m\sum_c Pr(m,c)\log\frac{Pr(m)}{Pr(m|c)} = -\sum_m\sum_c Pr(m,c)\log\frac{Pr(m)Pr(c)}{Pr(m,c)} \\ &\geq -\sum_m\sum_c \log Pr(m)Pr(c) \geq 0 \end{aligned}$$

és egyenlőség pontosan akkor lesz, ha $\frac{Pr(m)}{Pr(m|c)}$ állandó, vagyis valamilyen t-vel $Pr(m) = tPr(m|c)$. De $1 = \sum_m Pr(m) = t\sum_m Pr(m|c) = t$, így $t = 1$, tehát $Pr(m) = Pr(m|c)$, vagyis $I(\mathcal{M},\mathcal{C}) = 0$ akkor és csak akkor, ha minden $m \in \mathcal{M}$-re és $c \in \mathcal{C}$-re $Pr(m) = Pr(m|c)$, azaz pontosan akkor, ha $\mathcal{M}$ és $\mathcal{C}$ egymástól teljesen függetlenek.

A fenti definíció szerint a rendszer akkor és csak akkor tökéletes titkosságú, ha egy támadó semmit nem tud meg m-ről c ismeretében, vagyis c elfogása után pontosan annyi ismerete van m-ről, mint korábban volt.

15.7. Tétel

Legyen $|\mathcal{C}| = |\mathcal{K}|$ és $\forall(m \in \mathcal{M}): Pr(m) > 0$. Ekkor S pontosan akkor garantál tökéletes titkosságot, ha

1. $\forall(m \in \mathcal{M})\forall(c \in \mathcal{C})\exists!\,(k \in \mathcal{K}): E_k(m) = c$;
2. Pr_K egyenletes.

Δ

Bizonyítás:

a) Először tegyük fel, hogy S garantálja a tökéletes titkosságot. Legyen $m \in \mathcal{M}$ rögzített, és tegyük fel, hogy egy $c \in \mathcal{C}$-hez nincs olyan $k \in \mathcal{K}$, hogy $E_k(m) = c$. Ekkor $Pr(m) \neq 0 = Pr(m|c)$, vagyis S nem garantálja a tökéletes titkosságot, ami a feltevésünkkel ellentétes, így, mivel $|\mathcal{C}| = |\mathcal{K}|$, pontosan egy olyan $k \in \mathcal{K}$ van, amellyel $E_k(m) = c$, vagyis teljesül 1.

Most rögzítsünk egy $c \in \mathcal{C}$-t, és legyen $m \in \mathcal{M}$-re $k(m)$ az az egyetlen $k \in \mathcal{K}$, amellyel $E_k(m) = c$. Bayes tétele szerint $Pr(m|c) = \frac{Pr(c|m)Pr(m)}{Pr(c)} = \frac{Pr(k(m))Pr(m)}{Pr(c)}$ minden $m \in \mathcal{M}$-re. Mivel S garantálja a tökéletes titkosságot, ezért $Pr(m|c) = Pr(m)$, és így, az előbbi egyenlőség alapján, $Pr\big(k(m)\big) = Pr(c)$, és a jobb oldal független m-től, tehát $Pr(k)$ minden $k \in \mathcal{K}$-ra azonos, ezért $Pr(k) = \frac{1}{|\mathcal{K}|}$, Pr_K egyenletes.

b) Visszafelé, legyen $k = k(m, c)$ az egyetlen kulcs, amellyel $E_k(m) = c$. Ekkor

$$Pr(m|c) = \frac{Pr(m)Pr(c|m)}{Pr(c)} = \frac{Pr(m)Pr\big(k(m,c)\big)}{\sum_{q \in M} Pr(q)Pr\big(k(q,c)\big)}.$$

Mivel Pr_K egyenletes, ezért $Pr\big(k(m,c)\big) = \frac{1}{|\mathcal{K}|}$, továbbá

$$\sum_{q \in M} Pr(q)Pr\big(k(q,c)\big) = \frac{\sum_{q \in M} Pr(q)}{|\mathcal{K}|}.$$

Ezeket figyelembe véve $Pr(m|c) = Pr(m)$, és így S garantálja a tökéletes titkosságot.

□

$Pr(m|c) = Pr(m)$-ből következik, hogy *m* és *c* független, és ekkor $Pr(c|m) = Pr(c)$ is igaz.

15.8. Definíció

Legyen $r \in \mathbb{N}^+$, $A = \{k \in \mathbb{N} | k < r\}$ ábécé, $n \in \mathbb{N}^+$, $\mathcal{M} = \mathcal{C} = \mathcal{K} = A^n$, és $m \in \mathcal{M}$, $k \in \mathcal{K}$-ra $E_k(m)_i = c_i = (m_i + k_i) \bmod r$, ahol $n > i \in \mathbb{N}$, és k egyenletes eloszlású, teljesen véletlen, az m-től független sorozat egy eleme. Ekkor E a **véletlen átkulcsolás**, angolul a **one-time pad**, az **OTP**.

Δ

15.9. Tétel

A véletlen átkulcsolás tökéletes titkosító algoritmus.

Δ

Bizonyítás:

$E_k(m)_i = c_i = (m_i + k_i) \bmod r$-ből $k_i = (c_i - m_i) \bmod r$, vagyis minden $m \in \mathcal{M}$ és $c \in \mathcal{C}$ meghatároz egy és csak egy kulcsot, amellyel $E_k(m) = c$. A másik feltétel a definícióból adódik.

□

15.10. Tétel

Tökéletes titkosító algoritmus esetén $H(\mathcal{K}) \geq H(\mathcal{M})$.

Δ

Bizonyítás:

Tökéletes titkosító algoritmus esetén $I(\mathcal{M}, \mathcal{C}) = 0$. De $I(\mathcal{M}, \mathcal{C}) \geq H(\mathcal{K}) - H(\mathcal{M})$, és így azonnal kapjuk az állítást.

□

Mivel tökéletes titkosítás esetén $\mathcal{K}$ egyenletes eloszlású, ezért ekkor $H(\mathcal{M}) \leq \log|\mathcal{K}| = l(\mathcal{K})$, ahol $l(\mathcal{K})$ a kulcs effektív hossza, vagyis azt adja meg, hogy milyen hosszúak, hány jegyből állnak a kulcsok.

A tökéletes titkosítás más szavakkal az alábbi. Azon kulcsok teljes valószínűsége, amelyek m_i-t egy adott c-be transzformálnak azonos azon kulcsok teljes valószínűségével, amelyek m_j-t ugyanebbe a c-be transzformálják. Most a c-k száma azonos kell, hogy legyen az m-ek számával, mivel egy rögzített k-ra E_k egy-egyértelmű megfeleltetést hoz létre $\mathcal{M}$ elemei és $\mathcal{C}$ bizonyos elemei között. Tökéletes titkosítás esetén $Pr(c|m) = Pr(c) \neq 0$ minden ilyen c-re és minden m-re. Ennél fogva van legalább egy olyan kulcs, amely egy tetszőleges m-et bármely ilyen c-be transzformál. De az egy adott m-et különböző c-be transzformáló kulcsoknak különbözőeknek kell lenniük, és ezért a különböző kulcsok száma legalább akkora, mint az m-ek száma. Tökéletes titkosság csak a kulcsok ilyen számával nyerhető.

Tökéletes rendszerek, amelyekben a kriptogrammok, az üzenetek és a kulcsok száma azonos, jellemezhetőek azzal, hogy

- az összes kulcsot tekintve, minden m-nek minden c pontosan egy kulcsnál a képe;
- a kulcsok azonos valószínűségűek.

Legyen egy tökéletes titkosító rendszer, és legyen a lehetséges üzenetek száma n. Tekintsük bármely rögzített k kulcsot. Ekkor k az n különböző $m_0, \cdots, m_{n-1}$ nyílt szöveget a páronként különböző $c, \cdots, c_{v-1}$ kriptogrammba transzformálja, így $Pr(c_j) = Pr(c_j|m_j) > 0$ minden $0 \leq j < n$-re. De a rendszer tökéletes, és így bármely $u \neq j$-re $Pr(c_j|m_u) = Pr(c_j) > 0$. Ekkor kell lennie egy másik k' kulcsnak, amelyikre $E_{k'}(m_u) = c_j$. Ennek minden $0 \leq u < n$, $u \neq j$-re teljesülnie kell, és minden ilyen kulcsnak különbözőnek kell lennie, következésképpen legalább n kulcsnak kell lennie.

16. RSA

Az RSA a leggyakrabban alkalmazott és a legjobban bevált nyilvános kulcsú rejtjelezési algoritmus, amelyet sokan és igen alaposan vizsgáltak, és amely a publikus információk alapján gyakorlatilag fejthetetlen, ha a paramétereket a megfelelő gondossággal választják. Az algoritmus neve az őt kifejlesztő három matematikus: **R**ivest, **S**hamir és **A**dleman nevének kezdőbetűje.

A továbbiakban az a és $b \neq 0$ valós számra $a \bmod b = a - b\left\lfloor \frac{a}{b} \right\rfloor$. Ha a és b egész, akkor $a \bmod b \equiv a\ (b)$ és $0 \leq a \bmod b < b$, vagyis ekkor $a \bmod b$ az a b-vel való osztási maradéka.

Használni fogjuk $n \in \mathbb{N}^+$-ra az $M^{(n)} = \{m \in \mathbb{N} | m < n\}$ jelölést. Ekkor láthatóan $\left|M^{(n)}\right| = n$.

16.1. Definíció

Legyen $2 < p_1^{(A)} < p_2^{(A)}$ prímszám, $n^{(A)} = p_1^{(A)} p_2^{(A)}$, $e^{(A)}$ a $\varphi\left(n^{(A)}\right)$-hoz relatív prím pozitív egész, és $d^{(A)}$ az $e^{(A)}x \equiv 1\ \left(\varphi\left(n^{(A)}\right)\right)$ kongruencia tetszőleges pozitív megoldása. Ekkor $\left(n^{(A)}, e^{(A)}\right)$ az A nyilvános kulcsa, $d^{(A)}$ a titkos kulcsa, $M^{(A)} = M^{\left(n^{(A)}\right)} = C^{(A)}$ az A nyílt illetve rejtjeles szövegeinek halmaza, és az $m \in M^{(A)}$ nyílt szöveg rejtjeles párja $c = E_{e^{(A)}}(m) = m^{e^{(A)}} \bmod n^{(A)}$.

Δ

Maga a rejtjelezés könnyű feladat, polinomiális időben végrehajtható. Valóban: mivel a hatványozást csupán modulo n végezzük, ezért minden lépésben két, legfeljebb $b = \lfloor \log_r n \rfloor + 1$ hosszúságú számot szorzunk (r a számrendszer alapszáma), aminek az időigénye b^2 nagyságrendű, és ilyen szorzásból legfeljebb $2t$-re van szükség, ahol $t = \lfloor \log_2 e \rfloor$, amint az alábbi tétel mutatja.

16.2. Tétel

Legyen $1 < n \in \mathbb{N}$, $m \in \mathbb{Z}$ és $\sum_{i=0}^{t-1} u_i 2^i = u \in \mathbb{N}^+$, ahol $t = \lfloor \log_2 u \rfloor + 1$ és $t > i \in \mathbb{N}$-re $u_i \in \{0,1\}$. Ekkor az $m^{(t-1)} = m$, $t - 1 > i \in \mathbb{N}$: $m^{(i)} = m^{u_i}\left(m^{(i+1)}\right)^2 \bmod n$ algoritmus végén $m^{(0)} = m^u \bmod n$, és a szorzások s számára $t - 1 \leq s \leq 2(t-1)$.

Δ

Bizonyítás:

Könnyű igazolni, hogy $a(b \bmod n) = ab \bmod n$, ha a, b és n egész számok, így elég belátni, hogy ha $P^{(t-1)} = m$ és a $t - 1 > i \in \mathbb{N}$ indexekre $P^{(i)} = m^{u_i}\left(P^{(i+1)}\right)^2$, akkor $P^{(0)} = m^u$.

Ha $t = \lfloor \log_2 u \rfloor + 1$, akkor $2^{t-1} \leq u < 2^t$, vagyis u felíráshoz pontosan t bit kell, és $u_{t-1} = 1$. Most $P^{(t-1)} = m = m^1 = m^{u_{t-1}} = m^{\sum_{j=t-1}^{t-1} u_j 2^{j-(t-1)}}$, és ha $P^{(i+1)} = m^{\sum_{j=i+1}^{t-1} u_j 2^{j-(i+1)}}$ valamilyen $t - 1 > i \in \mathbb{N}$ indexre, akkor

$$P^{(i)} = m^{u_i}\left(P^{(i+1)}\right)^2 = m^{u_i}\left(m^{\sum_{j=i+1}^{t-1} u_j 2^{j-(i+1)}}\right)^2 = m^{u_i} m^{\sum_{j=i+1}^{t-1} u_j 2^{j-i}} = m^{\sum_{j=i}^{t-1} u_j 2^{j-i}},$$

innen pedig $i = 0$ esetén kapjuk, hogy $P^{(0)} = m^{\sum_{j=0}^{t-1} u_j 2^j} = m^u$.

Az algoritmus – t bit esetén – $t - 1$ lépésből áll (leszámítva az első értékadást). Minden lépés tartalmaz egy négyzetre emelést, amely egy szorzás, ez összesen $t - 1$ szorzás. u_i értéke 0 vagy 1; az első esetben $m^{u_i} = 1$, tehát a négyzetre emelés már $m^{(i)}$-t adja, míg $u_i = 1$ esetén $m^{u_i} = m$, vagyis $\left(m^{(i+1)}\right)^2$-et még meg kell szorozni m-mel, így az ilyen szorzások száma legfeljebb $t - 1$.

□

16.3. Megjegyzés

A bizonyításból látszik, hogy az algoritmus nem csak a moduláris, de a közönséges hatványozásra is hasonlóan működik, vagyis a tétel és a bizonyítás jelöléseivel $m^u = P^{(0)}$, és az eredmény most is legalább $t-1$ és legfeljebb $2(t-1)$ szorzással megkapható. Van azonban egy lényeges különbség a két hatványozás között. Legyen n valamilyen számrendszerben r-jegyű. Míg a moduláris hatványozás esetén minden lépésben n-nél kisebb, vagyis legfeljebb r-jegyű számokat kell szorozni, addig a közönséges hatványozásnál (nem nulla alap esetén) minden lépésben a négyzetre emelésnél megduplázódik a jegyek száma (vagy legfeljebb ennél eggyel kisebb lesz). Mivel a szorzáshoz nagyjából a tényezők jegyei számának szorzatával megegyező számú lépésre (egy-egy számjegy szorzására) van szükség, ezért most minden fordulóban hozzávetőleg négyszer több elemi számításra van szükség, mint az előző fordulóban. Ha m jegyeinek száma b, akkor tehát a moduláris hatványozásnál nagyságrendileg tb^2 elemi szorzás (tehát jegyenkénti szorzás) szükséges, míg a közönséges hatványozás esetén az ilyen lépések száma hozzávetőleg $\sum_{l=0}^{t-2}\left(2^l b\right)^2 = b^2\frac{4^{t-1}-1}{4-1} \sim 4^t b^2$, vagyis az előbbi esetben az elvégzendő műveletek száma t-nek polinomiális, a második esetben viszont exponenciális függvénye. Másként szólva, míg a moduláris hatványozás polinomiális időben elvégezhető, addig a közönséges hatványozás nem polinomiális bonyolultságú algoritmus.

Δ

Ahhoz, hogy a 16.1. definícióban valóban rejtjelezést adtunk meg, meg kell mutatni, hogy az $m \mapsto m^e \bmod n$ leképezés injektív $M^{(n)}$-en, a fejtés a titkos információ hiányában gyakorlatilag lehetetlen, de d birtokában könnyű végrehajtani. Ami a támadót illeti, neki egy $x^e \equiv c\ (n)$ kongruenciát kell megoldania. Jelenleg az egyetlen járható út (általában) az, ha c-nek vesszük a d-edik hatványát, mert amint a következő tételből kiderül, $m = c^d \bmod n$. Ám ehhez ismerni kellene d-t, és ehhez általában $\varphi(n)$-et, amit viszont csak akkor tudunk könnyen számítani, ha adott az n faktorizációja. Az utóbbi problémára – mármint adott szám felbontása prímtényezőinek szorzatára – nem ismeretes polinomiális idejű algoritmus, sőt, az tűnik valószínűnek, hogy ilyet nem is lehet megadni. Felhívjuk a figyelmet rá, hogy nem állítottuk, hogy a visszafejtés csupán így történhet, ezért nem mondtuk, hogy a fejtés nehézsége azonos a faktorizálás nehézségével; ezt sem nem bizonyították, sem nem cáfolták eddig (nyilvánosan!), továbbá azt sem mondtuk, hogy minden esetben a hatványozás a legkézenfekvőbb megoldás, bizonyos szerencsétlen (m, c) pár esetén egyszerűbben is megoldható a feladat (a szerencsétlen jelző a legális partnerek szempontjából értendő, a hívatlan támadó számára ez inkább szerencsés véletlen). Amit állíthatunk, az csupán annyi, hogy az RSA fejtése legfeljebb annyira nehéz, mint az összetett egész számok tényezőkre bontása, hiszen ha fel tudjuk n-et bontani, akkor már fejteni is tudunk, de elvileg elképzelhető, hogy van ennél egyszerűbb módja is a fejtésnek. Egyébként n felbontásának vagy $\varphi(n)$-nek az ismerete algoritmikus szempontból egyenértékű, mert egyikből a másik polinomiálisan számítható. Ez a felbontás ismeretében nyilvánvaló, hiszen $\varphi(n) = (p_1-1)(p_2-1)$, ami lényegében véve egyetlen szorzás. $p_1p_2 - p_1 - p_2 + 1 = (p_1-1)(p_2-1) = \varphi(n)$ és $p_1p_2 = n$, az előbbiből $p_1 + p_2 = n - \varphi(n) + 1$, így $\varphi(n)$ ismeretében p_1 és p_2 az $x^2 - (n - \varphi(n) + 1) + n$ polinom két gyöke, és a két gyök polinomiális időben meghatározható. d ismeretében viszont m könnyen nyerhető, mert a moduláris hatványozásról már megmutattuk, hogy könnyű feladat, így a legális címzett könnyen hozzájut a nyílt szöveghez. Most megmutatjuk, hogy tetszőleges $m \in M^{(n)}$ nyílt üzenetre $(m^e)^d \bmod n = m$, ebből majd az is következik, hogy a rejtjelszabályunk injektív.

16.4. Tétel

Legyen p és $p \neq q$ páratlan prím, és $n = pq$. Ekkor bármely a egész számra $a^{1+k\varphi(n)} \equiv a\ (n)$, ahol k nem negatív egész szám és φ az Euler-féle φ-függvény.

Δ

Bizonyítás:

Ha $p \nmid a$, akkor $a^{p-1} \equiv 1\ (p)$-ből $a^{1+k\varphi(n)} = a \cdot a^{k(p-1)(q-1)} = a \cdot (a^{p-1})^{k(q-1)} \equiv a\ (p)$, míg $p|a$ esetén $a \equiv 0\ (p)$, de akkor $a^{1+k\varphi(n)} \equiv 0 \equiv a\ (p)$. A másik prím esetén hasonló eredményt

kapunk, így $p \mid a^{1+k\varphi(n)} - a$ és $q \mid a^{1+k\varphi(n)} - a$, de akkor a két prím legkisebb közös többszöröse, azaz a szorzatuk is osztója $a^{1+k\varphi(n)} - a$-nak.

□

A tétel szerint az $M^{(n)} = \{m \in \mathbb{N} | m < n\}$ halmazon az $m \mapsto m^{1+k\varphi(n)} \bmod n$ leképezés az identikus leképezés. Ha e a $\varphi(n)$-hez relatív prím pozitív egész, akkor van olyan d pozitív egész, amellyel $ed \equiv 1 \left(\varphi(n)\right)$, vagyis amellyel $ed = 1 + k\varphi(n)$, ahol $k \in \mathbb{N}$. Ekkor $(m^e)^d = m^{1+k\varphi(n)}$, amiből következik, hogy az $m \mapsto m^e \bmod n$ megfeleltetés injektíven képezi le $M^{(n)}$-et önmagába, és mivel $M^{(n)}$ véges, ezért ez a szabály egyben bijektív is, és az inverze a $c \mapsto c^d \bmod n$ leképezés, amely szintén bijektíven képezi le az $M^{(n)}$ halmazt önmagára. Igaz tehát a következő tétel.

16.5. Tétel

Legyen p és $p \neq q$ páratlan prím, $n = pq$, e a $\varphi(n)$-hez relatív prím pozitív egész, ahol φ az Euler-féle φ-függvény és d olyan pozitív egész, amellyel $ed \equiv 1 \left(\varphi(n)\right)$. Ekkor az $m \mapsto m^e \bmod n$ megfeleltetés bijektíven képezi le az $M^{(n)} = \{m \in \mathbb{N} | m < n\}$ halmazt önmagára, és $c \mapsto c^d \bmod n$ az előbbi leképezés inverze.

Δ

Most olyan fejtési módszert vizsgálunk, amelyhez nem kell ismerni a d titkos paramétert, és megnézzük, hogyan lehet ez ellen a támadás ellen védekezni. Az eljárás csak nyilvános adatokat alkalmaz, és ismételt hatványozással állítja elő a nyílt üzenetet. Szükségünk lesz az alábbi tételre.

16.6. Tétel

Ha u és v pozitív egész, és $u|v$, akkor $\varphi(u)|\varphi(v)$.

Δ

Bizonyítás:

Legyen $s \in \mathbb{N}^+$, $s \geq i \in \mathbb{N}^+$-ra és $i > j \in \mathbb{N}^+$-ra $r_i \in \mathbb{N}^+$ és $p_i \neq p_j$ prímek, és $u = \prod_{i=1}^{s} p_i^{r_i}$. Mivel $u|v$, ezért $v = v_1 v_2 = v_2 \prod_{i=1}^{s} p_i^{t_i}$ úgy, hogy $(u, v_2) = (v_1, v_2) = 1$, és valamennyi t_i az r_i-nél nem kisebb egész. Most

$$\varphi(v) = \varphi(v_1)\varphi(v_2) = \varphi(v_2) \prod_{i=1}^{s} p_i^{t_i - 1}(p_i - 1)$$
$$= \varphi(v_2) \prod_{i=1}^{s} p_i^{r_i - 1}(p_i - 1) \prod_{i=1}^{s} p_i^{t_i - r_i} = \varphi(u)\varphi(v_2) \prod_{i=1}^{s} p_i^{t_i - r_i},$$

így valóban igaz, hogy $\varphi(u)|\varphi(v)$.

□

Nézzük meg, hogy adott $1 < n \in \mathbb{N}$, $1 < e \in \mathbb{N}$, $c \in M^{(n)}$ esetén mikor lesz olyan $k \in \mathbb{N}^+$, amellyel $c^{e^{k-1}} \bmod n = m$, ha $m \in M^{(n)}$-re $c = m^e \bmod n$. Rögtön látjuk, hogy ha az előbb megadott feltételek teljesülnek, akkor $c = m^e \bmod n = \left(c^{e^{k-1}} \bmod n\right)^e \bmod n = c^{e^k} \bmod n$, vagyis ekkor $n \left| c^{e^k} - c = c\left(c^{e^k - 1} - 1\right)\right.$, ami viszont pontosan akkor igaz, ha $\frac{n}{(c,n)} \left| c^{e^k - 1} - 1\right.$. Ez ekvivalens a $c^l \equiv 1 \left(\frac{n}{(c,n)}\right)$ kongruenciával, ahol $l = e^k - 1$. A kongruenciának akkor és csak akkor van megoldása, ha $1 = \left(c, o_n^+(c)\right)$, ahol a rövidség kedvéért bevezettük az $o_n^+(c) = \frac{n}{(c,n)}$ jelölést. Ez viszont akkor és csak akkor teljesül, ha a c bármely p prímosztója c-ben legalább akkora hatványon fordul elő, mint

n-ben. Ez biztosan így van, ha n négyzetmentes. Ekkor $c^l \equiv 1 \left(o_n^+(c)\right)$-hez szükséges és elégséges, hogy $o_{o_n^+(c)}(c) \big| l = e^k - 1$, vagy ismét átírva kongruenciába, ha $e^k \equiv 1 \left(o_{o_n^+(c)}(c)\right)$. Ilyen k pontosan akkor van, ha e relatív prím $o_{o_n^+(c)}(c)$-hez. De $o_{o_n^+(c)}(c) \left| \varphi\left(o_n^+(c)\right) = \varphi\left(\frac{n}{(c,n)}\right) \right| \varphi(n)$, így, ha $\left(e, \varphi(n)\right) = 1$, akkor van ilyen k, és a legkisebb ilyen k pozitív egész éppen $o_{o_{o_n^+(c)}(c)}(e)$. Ha tehát n négyzetmentes és e relatív prím $\varphi(n)$-hez, akkor $M^{(n)}$ egy c elemére a $k_c = o_{o_{o_n^+(c)}(c)}(e)$ pozitív egész számmal $c^{e^{k_c - 1}} \bmod n = m$, és ha k a k_c-k legkisebb közös többszöröse, akkor valamennyi $c \in M^{(n)}$-re $c^{e^{k-1}} \bmod n = m$, és k a legkisebb ilyen tulajdonságú pozitív egész szám.

$o_u(v)$ osztója $\varphi(u)$- nak, $o_u^+(v)$ pedig u-nak, így felhasználva az előző eredményeket

$$o_{o_{o_n^+(c)}(c)}(e) \Big| \varphi\left(o_{o_n^+(c)}(c)\right) \Big| \varphi\left(\varphi\left(o_n^+(c)\right)\right) \Big| \varphi\left(\varphi(n)\right)$$

minden c-re, így $k_c | k \big| \varphi\left(\varphi(n)\right)$, tehát, ha azt akarjuk, hogy k_c a lehető legtöbb c-re nagy legyen, akkor n-et úgy kell választani, hogy $\varphi\left(\varphi(n)\right)$-nek kevés kis osztója legyen, és a kis osztókkal csak kevés c-t lehessen fejteni. Természetesen mindig lesz olyan c, amely kis kitevővel fejthető, hiszen az RSA-nak vannak fixpontjai, és ezek már $k = 1$-gyel fejthetőek. Az lenne a jó, ha a fixpontok száma minél kisebb lenne, és minden más rejtjelezett szövegből csak nagy k' kitevővel lehetne visszanyerni az eredeti üzenetet.

Az előbbi eredményből levezethető, hogy az RSA biztonságos működéséhez úgy kell megválasztani a két prímet, hogy egyrészt teljesüljön a $p \approx q$ feltétel, továbbá $p = 2p_1 + 1$, $q = 2q_1 + 1$, $p_1 = 2p_2 + 1$, $q_1 = 2q_2 + 1$ legyen, ahol p_1, p_2, q_1 és q_2 egyaránt prímek. Valóban, $n = pq$, ahol $2 < p < q$ prímszám, így $\varphi(n) = (p-1)(q-1)$. Mivel mindkét prím páratlan, ezért $\varphi(n)$ mindkét tényezője páros, és így $\varphi(n) = 4\frac{p-1}{2}\frac{q-1}{2}$. Ennek akkor lesz a lehető legkevesebb kis osztója, ha mindkét hányados prímszámot ad, vagyis ha $p = 2p_1 + 1$ és $q = 2q_1 + 1$ a p_1, q_1. prímekkel, és ekkor $\varphi(n) = 4p_1q_1$. p és q nagy prímek, ezért p_1 és q_1 egyaránt páratlan, és így $\varphi\left(\varphi(n)\right) = 8\frac{p_1-1}{2}\frac{q_1-1}{2}$. Végül ennek ismét akkor lesz a lehető legkevesebb kis osztója, ha $\frac{p_1-1}{2}$ és $\frac{q_1-1}{2}$ egyaránt prím, vagyis ha $p_1 = 2p_2 + 1$, $q_1 = 2q_2 + 1$.

Az algoritmust alkalmazva, ha $\left(\left(c^{e^k} \bmod n\right) - c, n\right) > 1$ egy pozitív k egész kitevővel, de $c^{e^k} \bmod n \neq c$, akkor $\left(\left(c^{e^k} \bmod n\right) - c, n\right) = p_1$ vagy $\left(\left(c^{e^k} \bmod n\right) - c, n\right) = p_2$, és ekkor ismert n felbontása, tehát d meghatározható, a rendszert sikerült feltörni. Ám a faktorok fentiekben ismertetett választásával a legkisebb ilyen k kitevő $2p_1^{(2)} \approx \frac{\sqrt{n}}{2}$, feltéve, hogy $p_1 < p_2$.

A p pímszám **Sophie Germain-prím**, ha $p = 2p' + 1$ alakú a p' prímmel. Láttuk, hogy RSA-hoz a kétszeresen Sophie Germain prímek a jók (vagyis ahol p' is Sophie Germain-prím). Kérdés, hogy létezik-e ilyen prím. A válasz igenlő: például $2 \cdot 2 + 1 = 5$ és $2 \cdot 5 + 1 = 11$, $2 \cdot 5 + 1 = 11$ és $2 \cdot 11 + 1 = 23$, $2 \cdot 11 + 1 = 23$ és $2 \cdot 23 + 1 = 47$, $2 \cdot 41 + 1 = 83$ és $2 \cdot 83 + 1 = 167$ stb.

Az $n = pq$ választásánál az eddigieken túl egy további szempont, hogy $|q - p|$ sem lehet kicsi, ugyanis $(q+p)^2 - (q-p)^2 = 4pq = 4n$, innen $(q+p)^2 = 4n + (q-p)^2$, vagyis egy kis pozitív egész négyzetét $4n$-hez adva ismét négyzetszámot kapunk, amit könnyen lehet ellenőrizni. Ha tehát u és v olyan egészek, hogy $4n + u^2 = v^2$, akkor $a = \frac{v-u}{2}$, $b = \frac{v+u}{2}$ és $n = ab$, de n egyetlen felbontása pq, tehát $a = p$ és $b = q$, n-et könnyű faktorizálni, és így már könnyű $\varphi(n)$-et és az $ex \equiv 1 \left(\varphi(n)\right)$ megoldását megtalálni. Ez mutatja, hogy p és q választásánál a $|q - p| \approx p (\approx q)$ feltételnek is teljesülnie kell, ha azt akarjuk, hogy ne lehessen könnyen megfejteni a rejtjelünket.

Ha $p = 2p' + 1$ és $q = 2q' + 1$, ahol p' és q' páratlan prímszám, akkor $(p-1, q-1) = 2$, ami azért is fontos, mert ha $t = [p-1, q-1]$, és $ed \equiv 1\ (t)$, akkor ezzel a d kitevővel is fejthető az RSA. Most $= [p-1, q-1] = \frac{(p-1)(q-1)}{\delta} < \frac{pq}{\delta} = \frac{n}{\delta}$, ahol $\delta = (p-1, q-1)$, és ha δ nagy, akkor t kicsi, és kevés próbálkozással található olyan d, amellyel $c^d \bmod n = m$.

Most azt vizsgáljuk, hogy mi a kapcsolat a rejtjel biztonsága és a rejtjelből nyerhető részleges információ között. Megmutatjuk, hogy ha meg tudjuk állapítani a rejtjeles szövegből a nyílt szöveg utolsó bitjét, akkor már az egész szöveget könnyen fejthetjük. Először egy segéderedményt látunk be.

16.7. Tétel

Legyen $2 \nmid n \in \mathbb{N}$ négyzetmentes, e a $\varphi(n)$-hez relatív prím pozitív egész, $f: x \mapsto x^e \bmod n$ az $M^{(n)}$ halmaz önmagába való leképezése, $K \in \mathbb{Z}$ olyan, hogy $2^e K \equiv 1\ (n)$, $u \in M^{(n)}$, $v = f(u)$ és $z = u \bmod 2$. Ekkor $u' = 2^{-1}((-1)^z v \bmod n) \in M^{(n)}$ és $v' = K(-1)^z v \bmod n = f(u')$.

Δ

Bizonyítás:

Elsőként megjegyezzük, hogy létezik a tételben igényelt K, hiszen n páratlan.

$u \in M^{(n)}$ esetén $-n < (-1)^z u < n$-ből $(-1)^z u \bmod n = zn + (-1)^z u$. Ez egy n-nél kisebb, páros, nem negatív egész, tehát osztható kettővel, és u' ismét n-nél kisebb nem negatív egész. e páratlan, hiszen n páratlan, egynél nagyobb páratlan számra $\varphi(n)$ páros, és páros számhoz relatív prím páratlan, ezért $(-1)^{ez} = (-1)^z$. Ezt felhasználva $2u' = (-1)^z u \bmod n \equiv (-1)^z u\ (n)$-ből

$$\begin{aligned} u'^e \equiv (2^e K)u'^e = K(2u')^e \equiv K(-1)^{ez} u^e \\ \equiv K(-1)^z v \equiv K(-1)^z v \bmod n = v'(n), \end{aligned}$$

és így v' az u' képe az f leképezésnél, $v' = f(u')$, ahogy a tételben állítottuk.

□

Ezt az eredményt felhasználjuk a következő tétel bizonyításában.

16.8. Tétel

Legyen $r \in \mathbb{N}^+$, $2 \nmid n \in [2^{r-1}, 2^r[$ négyzetmentes és e a $\varphi(n)$-hez relatív prím egész, $K \in \mathbb{Z}$-re $2^e K \equiv 1\ (n)$, $M^{(n)}$-en $f: x \mapsto x^e \bmod n$, $m \in M^{(n)}$ és $c = f(m)$. Ha $g(v) = f^{-1}(v) \bmod 2$, úgy az alábbi algoritmus c-ből előállítja m-et:

$$\begin{array}{rlcllcl} & y_0 & = & c & & z_0 & = & g(y_0) \\ r-1 > i \in \mathbb{N}: & y_{i+1} & = & K(-1)^{z_i} y_i \bmod n & & z_{i+1} & = & g(y_{i+1}) \end{array}$$

majd

$$\begin{array}{rlcl} & t_{r-1} & = & z_{r-1} \\ r-1 > i \in \mathbb{N}: & t_i & = & ((-1)^{z_i}(2t_{i+1}) \bmod n) \bmod 2^{r-i}. \end{array}$$

Δ

Bizonyítás:

Azt már tudjuk, hogy ha $x_0 = m$ és $x_{i+1} = 2^{-1}((-1)^{z_i} x_i \bmod n)$, akkor $y_i = f(x_i)$, így az algoritmusban meghatározott z_i éppen $x_i \bmod 2$. Azt fogjuk belátni, hogy minden $r - 1 > i \in \mathbb{N}$ egészre $t_i = x_i \bmod 2^{r-i}$. Ebből már következik az állítás, hiszen $n < 2^r$ következtében mind x_0, mind t_0 n-nél kisebb nem negatív egész a definíciók alapján, és így $m = x_0 = t_0$.

z_i az x_i paritását mutatja, így z_i éppen x_i jobb szélső bitje az x_i kettes számrendszerbeli felírásánál, és ha $i = r - 1$, akkor tehát $x_{r-1} \bmod 2 = z_{r-1} = t_{r-1}$. Tegyük fel, hogy egy $r - 1 > i \in \mathbb{N}$, esetén $t_{i+1} = x_{i+1} \bmod 2^{r-(i+1)}$, azaz $t_{i+1} \equiv x_{i+1} \left(2^{r-i-1}\right)$. Ekkor $2t_{i+1} \equiv 2x_{i+1} \left(2^{r-i}\right)$, továbbá az előző tétel alapján y_{i+1} az $x_{i+1} = 2^{-1}((-1)^{z_i} x_i \bmod n)$ üzenethez tartozó rejtjel. Innen

$$t_i \equiv (-1)^{z_i}(2t_{i+1}) \bmod n = z_i n + (-1)^{z_i}(2t_{i+1}) \equiv z_i n + (-1)^{z_i}(2x_{i+1})$$
$$= z_i n + (-1)^{z_i}(z_i n + (-1)^{z_i} x_i) = z_i n + (-1)^{z_i} z_i n + x_i = x_i \left(2^{r-i}\right),$$

mert $z_i n + (-1)^{z_i} z_i n = 0$, tekintettel arra, hogy z_i csupán nulla vagy egy lehet.

□

16.9. Megjegyzés

Az előző tétel alapján belátható, hogy amennyiben azt tudjuk y-ból megállapítani, hogy x kisebb-e, mint n fele, vagy nagyobb, akkor hasonlóan egyszerű már a fejtés (harmadik lehetőség, azaz egyenlőség most kizárt, mert n páratlan egész és x egész).

Δ

Most három kérdésre térünk ki. Mi történik, ha

1. $(m, n) > 1$ (m a nyílt szöveg és n a modulus);
2. n több kulcsra azonos;
3. illetve több résztvevőre megegyezik az e kitevő (de a modulusok különbözőek).

Az első kérdés könnyen elintézhető. Ha $n = pq$ és $m \neq 0$, akkor $1 < (m, n)$ csak p vagy q lehet. Ha a közös osztó p, akkor $(c, n) = p$ is igaz, és ebből a támadó meg tudja határozni q-t, $\varphi(n)$-et és d-t, vagyis feltöri a rendszert. Ennek valószínűsége azonban csekély, hiszen az n-hez nem relatív prím, n-nél kisebb nem negatív egészek száma $n - \varphi(n) = pq - (p-1)(q-1) = p + q - 1$, és ezek aránya az n-nél kisebb nem negatív egészekhez $\frac{p+q-1}{pq} \approx \frac{1}{p} + \frac{1}{q} \approx \frac{2}{\sqrt{n}}$, viszont n nagy.

Térjünk rá a második esetre. Legyen k résztvevő esetén azonos az n modulus, és közülük az i-edik nyilvános kulcsa e_i. Ha közülük akár csak kettő, mondjuk az 1 és 2 indexű, azonos nyílt szöveg rejtjeles változatát kapja (például egy körlevelet), és e_1, e_2 relatív prím, akkor egy támadó is vissza tudja fejteni c_1-ből és c_2-ből az m üzenetet. Most ugyanis $c_1 \equiv m^{e_1}\ (n)$ és $c_2 \equiv m^{e_2}\ (n)$. Mivel e_1 és e_2 relatív prím, ezért van olyan u_1 és u_2 egész, hogy $1 = u_1 e_1 + u_2 e_2$. e_1 és e_2 1-nél nagyobb pozitív egész, ezért az előbbi egyenlőség csak úgy állhat fenn, ha u_1 és u_2 egyike pozitív, a másik negatív. Szimmetriaokokból bármelyiküket tekinthetjük negatívnak, legyen például $u_1 < 0$ és $u_2 > 0$. Ekkor

$$m = m^1 = m^{u_1 e_1 + u_2 e_2} = (m^{e_1})^{u_1} (m^{e_1})^{u_1} \equiv c_1^{u_1} c_2^{u_2}\ (n),$$

innen $(c_1)^{(-u_1)} m \equiv c_2^{u_2}\ (n)$ ($-u_1$ már pozitív), vagyis m a $(c_1)^{(-u_1)} x \equiv c_2^{u_2}\ (n)$ kongruencia megoldása, és megoldás biztosan létezik, például az eredeti m üzenet, vagyis a rejtjeles szövegekből ismert kitevős hatványokkal egy egyismeretlenes lineáris kongruencia megoldásaként, tehát polinomiális időben megkapjuk a nyílt szöveget (egy ilyen kongruencia például euklideszi algoritmussal megoldható, és ez polinomiális algoritmus).

Most tegyük fel, hogy k-számú résztvevőnek azonos e rejtjelkitevője van, és az i-edikhez az n_i modulus tartozik, továbbá a modulusok páronként relatív prímek (ennél enyhébb feltétel is elegendő lenne). Ha egy körlevél következtében mindegyikük azonos rejtjeles szöveget kap, akkor a közös m könnyen meghatározható egy kívülálló részéről is. Legyen c_i az i-edik rejtjeles szöveg, akkor tehát $c_i \equiv m^e\ (n_i)$ valamennyi i-re, azaz m^e a megoldása a $c_i \equiv x\ (n_i)$ szimultán kongruenciarendszernek. De ennek egy és csak egy megoldása van modulo n, ahol n az n_i-k szorzata, így egy és csak egy

olyan megoldás van, ahol $n > x \in \mathbb{N}$. Nyilván érvényesnek kell lennie az $n_i > m \in \mathbb{N}$ feltételnek minden i-re, így ha $k \geq e$, akkor $n > m^e \in \mathbb{N}$, és ezért most $m^e = x$, ahonnan m gyökvonással megkapható.

Az, hogy több felhasználó nyilvános rejtjelkitevője azonos, nem rendkívüli. *e* nagysága nem befolyásolja különösebben a rejtjel biztonságát, ezért a számítás egyszerűsége érdekében célszerű kicsire választani. Ha a rendszerben sok szereplő vesz részt, akkor előfordul, hogy bár egymástól függetlenül választják a paramétereket, de a kevés számú kis érték közül többen is azonosat választanak.

Még nézzük meg a paraméterek választását. Véletlen prímet például véletlenszám generátorral nyerhetünk: generálunk egy számot, prímteszttel megvizsgáljuk, és ha nem prím (illetve nem minősítjük prímnek), akkor vehetjük a természetes számsorban következő páratlan egészt. Tegyük fel, hogy m-nagyságrendű prímet keresünk. Csebisev tétele szerint bármely szám és a kétszerese között van prím, és a nagy prímszámtétel szerint az x számnál nem kisebb prímek száma, $\pi(x)$, nagy x-ekre körülbelül $\frac{x}{\ln x}$, $\pi(x) \sim \frac{x}{\ln x}$. Ekkor az m és $2m$ közötti prímek várható aránya

$$\frac{\pi(2m) - \pi(m)}{m} \sim \frac{\frac{2m}{\ln(2m)} - \frac{m}{\ln m}}{m} = \frac{2}{\ln 2 + \ln m} - \frac{1}{\ln m} \approx \frac{1}{\ln m},$$

vagyis várhatóan $\ln m$ kísérlet után prímet kapunk, sőt, ha figyelembe vesszük, hogy a prímek páratlanok (kivéve a 2-t), és csak minden második szám páratlan (és csak ezekkel kísérletezünk), akkor átlagosan $\frac{\ln m}{2}$ kísérlettel prímhez jutunk. Konkrétan $m \sim 10^{100}$ esetén ez körülbelül 115 próbálkozást jelent (megjegyezzük, hogy az előbbi kétszeres tartománynál lényegesen kisebb intervallumra is igaz, hogy van benne prím, ha x elegendően nagy, másrészről láttuk, hogy nem akármilyen prím alkalmas).

e-nek relatív prímnek kell lennie $\varphi(n)$-hez. Ez például úgy biztosítható, ha $q < e < n$ és e prím, ahol q az n-ben lévő nagyobbik prím. Ez az e valóban relatív prím $\varphi(n)$-hez, hiszen ez utóbbi minden prímosztója kisebb e-nél.

Δ

17. Diszkrét logaritmus

Legyen G egy n-edrendű ciklikus csoport a g generátorelemmel. Ekkor a G bármely u eleméhez van egy, a g és u által egyértelműen meghatározott, $n > k \in \mathbb{N}$ egész szám, amellyel $g^k = u$, és ennek megfelelően az az $u \mapsto k$ szabály, amely u-hoz az előbbi k-t rendeli, G-nek $M^{(n)}$-be való bijektív leképezése ($M^{(n)}$ a korábban már más összefüggésben definiált halmaz, amely az n-nél kisebb nem negatív egész számokat tartalmazza.) Az előbbi leképezést $\mathrm{ind}_g\, u$-val vagy $\log_g u$-val jelöljük, és g-**alapú diszkrét logaritmus**nak vagy g**-alapú index**nek nevezzük. Könnyű ellenőrizni, hogy

- $\mathrm{ind}_g(uv) = \left(\mathrm{ind}_g\, u + \mathrm{ind}_g\, v\right) \bmod n$;
- $\mathrm{ind}_g\, u = 0 \Leftrightarrow u = e$;
- $u \neq e \Rightarrow \mathrm{ind}_g\, u^{-1} = n - \mathrm{ind}_g\, u$;
- $\mathrm{ind}_g\, u^r = \left(r \cdot \mathrm{ind}_g\, u\right) \bmod n$.

ahol e a csoport egységeleme, és r tetszőleges egész szám.

k ismeretében, adott g esetén, u meghatározása könnyű feladat, ám az inverz művelet a mai ismereteink szerint algoritmikusan nehéz feladat, ezért alkalmazhatjuk a rejtjelezésben.

Most a diszkrét logaritmus három kriptográfiai alkalmazását mutatjuk meg.

1. **Kulcscsere** Diffie és Hellman cikkében alapvetően nem a nyilvános kulcsú rejtjelezésről volt szó, ez csupán mint egy lehetőség merült fel, ám megoldást erre a kérdésre a cikk nem tartalmazott (viszont, meglepő módon, igen hamar megjelentek megoldások, például az azóta már eredeti formájában nem biztonságos, a hátizsák-algoritmuson alapuló módszer, és az azóta is talán leggyakrabban alkalmazott nyilvános kulcsú rejtjelező algoritmus, az RSA). A két szerző alapvetően a kulcscsere problémájával foglalkozott, azt a kérdést vizsgálták, hogy lehetséges-e, és ha igen, akkor például hogyan lehet nyilvános csatornán kicserélni két kommunikáló fél között a titkos kulcsukat.

A feladat tehát az, hogy két fél szeretne nyilvános csatornán keresztül titkos kulcsot cserélni. Legyen G egy nyilvánosan ismert n-edrendű ciklikus csoport a (szintén nyilvánosan ismert) g generátorelemmel. A választ egy $n > k_A \in \mathbb{N}^+$ és B egy $n > k_B \in \mathbb{N}^+$ értéket. A elküldi B-nek g^{k_A}-t, míg B A-nak g^{k_B}-t. Most A kiszámolja g^{k_B}-ből $g^{k_A k_B}$-t, és hasonlóan, B kiszámítja g^{k_A}-ból $g^{k_A k_B}$-t, és így, láthatóan, lesz egy közös kulcsuk. Ha valaki megfigyeli a csatornán g^{k_A}-t és g^{k_B}-t, és valamelyikből meg tudja határozni a kitevőt, vagyis meg tudja oldani a diszkrét logaritmus problémáját, akkor rendelkezik a közös kulccsal. Ebből látszik, hogy a közös kulcs támadó általi meghatározása legfeljebb olyan bonyolultságú, mint a diszkrét logaritmus problémája. Ám az nem biztos, hogy csak így juthat hozzá a közös kulcshoz. A feldat, amelyet meg kell oldania, az, hogy ha ismeri g két hatványát, ebből meg tudja-e állapítani a két kitevő szorzatához tartozó hatványt, azaz g^{k_A} és g^{k_B} ismeretében ki tudja-e számítani $g^{k_A k_B}$-t. Ez a feladat a **Diffie-Hellman probléma**. Az előbbiek szerint ez tehát legfeljebb olyan bonyolultságú, mint a diszkrét logaritmus probléma, és a rejtjelezés szempontjából reméljük, amint ma hisszük, hogy azzal azonos nehézségű.

Egy aktív támadó eredményesen tud beavatkozni a rendszerbe egyéb intézkedések hiányában. A módszer az úgynevezett **középre állás**. Legyen C a támadó, aki képes a csatornából üzeneteket kivonni illetve beszúrni. Amikor A elküldi B-nek g^{k_A}-t, akkor ezt C kivonja a csatornából, és helyette, választva egy k_{CA} kitevőt, visszaküldi A-nak $g^{k_{CA}}$-t. Hasonlóan, amikor B küldi A-nak g^{k_B}-t, ezt kiemeli a csatornából, és helyette egy általa választott k_{CB} kitevővel visszaküldi B-nek $g^{k_{CB}}$-t. Ha most A egy üzenetet akar váltani B-vel, akkor ezt titkosítja a B-vel közösnek gondolt $g^{k_A k_{CA}}$-val, és elküldi. Ezt elfogja C, megfejti az A-val közösen ismert $g^{k_A k_{CA}}$ kulccsal, majd akár ezt az üzenetet, akár helyette egy másikat a $g^{k_B k_{CB}}$ kulccsal titkosítva, elküldi B-nek. C ugyanígy tud eljárni, ha B küld rejtjelezett üzenetet A-nak

2. **Közös kulcs nélküli üzenetváltás** Felmerül a kérdés, vajjon lehet-e közös kulcs nélkül titkosított üzenetet cserélni. Ha igen, akkor elkerülhető a kulcs kicserélésének problémája. Nézzük a következő protokollt. A beteszi az üzenetet egy ládikába, és lelakatolja a ládikát (amelyhez csak neki van kulcsa), majd elküldi B-nek. B nem tudja kinyitni a ládikát, hiszen nincs kulcsa hozzá, ezért mérgében rátesz még egy lakatot, amelyhez viszont csak neki van kulcsa, és visszaküldi a feladónak. A mosolyogva leveszi a saját lakatját, és ismét elküldi a ládikát B-nek, aki most már hozzáfér az üzenethez.

Az előbbi eljárás ismét támadható a középre állással. Például a postás megteheti, hogy ahelyett, hogy kikézbesítené a ládikát B-nek, inkább ő lakatolja le és küldi vissza, majd amikor ismét visszaérkezik a ládika, akkor kiveszi a levelet. Ezek után vagy ugyanezt a levelet, vagy egy másikat, beteszi a ládikába, és végrehajtja a protokollt B-vel, mintha ő lenne A.

Az előbb leírt módszer a kriptográfia nyelvén is megadható. Legyen A titkosító algoritmusa $E^{(A)}$ és fejtő algoritmusa $D^{(A)}$, valamint egy összetartozó kulcspárja $k_A^{(E)}$ és $k_A^{(D)}$, és hasonlóan, legyen $E^{(B)}$ és $D^{(B)}$ a B titkosító és desifrírozó algoritmusa a $k_B^{(E)}$ és $k_B^{(D)}$ kulcspárral. Legyen m az üzenet, amelyet A titkosítva akar B-nek elküldeni. Ekkor kiszámítja $u = E^{(A)}_{k_A^{(E)}}(m)$-et, és ezt elküldi B-nek. B, mivel nem rendelkezik a visszafejtéshez szükséges kulccsal, nem tudja visszafejteni u-t, ezért visszaküldi A-nak $v = E^{(B)}_{k_B^{(E)}}(u)$-t. Most A erre az üzenetre alkalmazza a saját fejtő algoritmusát, vagyis meghatározza $w = D^{(A)}_{k_A^{(D)}}(v)$. Amennyiben $D^{(A)}_{k_A^{(D)}}\left(E^{(B)}_{k_B^{(E)}}\left(E^{(A)}_{k_A^{(E)}}(m)\right)\right) = E^{(B)}_{k_B^{(E)}}(m)$, akkor

$$w = D^{(A)}_{k_A^{(D)}}(v) = D^{(A)}_{k_A^{(D)}}\left(E^{(B)}_{k_B^{(E)}}\left(E^{(A)}_{k_A^{(E)}}(m)\right)\right) = E^{(B)}_{k_B^{(E)}}(m),$$

és ha ezt A visszaküldi B-nek, akkor ebből B a saját visszafejtő algoritmusával visszanyeri m-et, hiszen $D^{(B)}_{k_B^{(D)}}(w) = D^{(B)}_{k_B^{(D)}}\left(E^{(B)}_{k_B^{(E)}}(m)\right) = m$. Láthatóan az eljárás akkor működik általánosan helyesen, ha $D^{(A)}$ és $E^{(B)}$ felcserélhetőek.

Legyen például G egy nyilvánosan ismert n-edrendű csoport. A választ egy, csak általa ismert $n > e_A \in \mathbb{N}^+$ és B egy $n > e_B \in \mathbb{N}^+$ értéket úgy, hogy mind e_A, mind e_B relatív prím n-hez. Ekkor mindketten meg tudnak határozni egy d_A és d_B egészt úgy, hogy teljesüljön az $e_A d_A \equiv 1\ (n)$ illetve $e_B d_B \equiv 1\ (n)$ kongruencia. e_A és e_B a két fél titkos rejtjelező kulcsa, míg d_A és d_B a szintén titkos fejtő kulcsuk. Tekintettel arra, hogy n-edrendű csoport minden g elemére $g^n = e$, ahol e a csoport egységeleme, $(g^e)^d = g$. Ha m az üzenet, amelyet A küld B-nek (és m eleme G-nek, ami kódolással megoldható), akkor, az előbbi jelölésekkel,

$$\begin{array}{rcccccccl} u & = & E^{(A)}_{k_A^{(E)}}(m) & = & m^{e_A} & & & & \\ v & = & E^{(B)}_{k_B^{(E)}}(u) & = & (m^{e_A})^{e_B} & & & & \\ w & = & D^{(A)}_{k_A^{(E)}}(v) & = & ((m^{e_A})^{e_B})^{d_A} & = & (m^{e_B})^{e_A d_A} & = & m^{e_B} \\ z & = & D^{(B)}_{k_B^{(E)}}(w) & = & (m^{e_B})^{d_B} & = & m. & & \end{array}$$

Ha egy támadó fel akarja törni a rendszert, akkor egy diszkrét logaritmus problémát kell megoldania, hiszen a nyilvánosan látható mondjuk $u = m^{e_A}$-ból a kitevőt kell meghatároznia.

Speciális esetként legyen p egy közösen ismert prímszám, e_A és e_B relatív prím $p-1$-hez. Ekkor $E_e = m^e \bmod p$ megfelel az előbbieknek.

3. **AlGamal titkosítás** Megint legyen G egy nyilvánosan ismert n-edrendű ciklikus csoport a (szintén nyilvánosan ismert) g generátorelemmel. A választ egy, csak általa ismert $n > k_A \in \mathbb{N}^+$ értéket, és meghatározza $u_A = g^{k_A}$-t. A továbbiakban k_A az A titkos, és u_A a nyilvános kulcsa. Ha B titkosított üzenetet akar küldeni A-nak, és m az üzenet (amely most G valamely eleme), akkor választ egy

véletlen t egész számot, kiszámítja $r = g^t$-t valamint $s = u_A^t m$-et, és elküldi A-nak a $c = (r, s)$ párt. A ebből meg tudja határozni m-et, ugyanis $r^{n-k_A} s = (g^t)^{n-k_A} \left(g^{k_A}\right)^t m = g^{tn} m = m$. Ha viszont valaki nem ismeri k_A-t, de c-ből meg tudja határozni m-et, akkor m, $u_A = g^{k_A}$ és $r = g^t$ ismeretében meg tudja határozni $sm^{-1} = u_A^t = g^{k_A t}$-t, vagyis meg tudja oldani a Diffie-Hellman problémát.

Az AlGamal módszer sávszélesség-növekedéssel jár, hiszen m helyett egy hasonló méretű elemekből álló párt kell átvinni a csatornán. Ugyanakkor ez az eljárás randomizált: ugyanazon üzenetet különböző alkalmakkor ugyanannak a személynek elküldve mindenegyes alkalommal más és más lesz a rejtjelezett üzenet, feltéve, hogy minden alkalommal egymástól függetlenül választott véletlen számot alkalmaz a küldő fél.

Ügyelni kell rá, hogy bármilyen is legyen az A-nak küldött üzenet, minden alkalommal más és más véletlenül választott számmal történjen a rejtés. Legyen ugyanis m_1 és $m_2 \neq m_1$ két üzenet, és mindkettőt titkosítsuk ugyanazon t kitevővel. Ekkor $\frac{m_1}{m_2} = \frac{s_1}{s_2} = s$ nem függ A titkos kulcsától, és ha az üzenet elegendően redundáns, akkor fejthető.

Az AlGamal rendszert széles körben alkalmazzák, az RSA mellett egy gyakran alkalmazott nyilvános kulcsú titkosító módszer. Ennek az elvnek fontos alkalmazása van a digitális aláírásnál is.

18. Integritás, személyazonosítás, hitelesítés

Az aktív támadással szembeni védekezés során a következőkről van szó

- a küldött üzenet integritásának ellenőrzése;
- a rendszerhez való hozzáférés jogosultságának ellenőrzése;
- a küldött üzenet hitelességének ellenőrzése.

Az üzenet integritása annak sértetlenségét jelenti. Azt jelenti, amit úgy szoktak mondani, hogy „semmit el nem vettem belőle, és semmit hozzá nem tettem". Erre a célra úgynevezett ujjlenyomatot használnak, amelyet egy hasítófüggvény, másként egy **hash-függvény** állít elő. Az ilyen függvények tetszőleges hosszúságú karaktersorozatból egy fix hosszúságú karaktersorozatot állítanak elő. Két fajtája van:

- az **MDC** (***M**odification **D**etection **C**ode*);
- a **MAC** (***M**essage **A**uthentication **C**ode*).

Az előbbi csupán az eredeti üzeneten végrehajtott módosítást jelzi, míg a második titkos kulcsú rendszerekben működik, és egyben hitelesítést is végez, amelyet úgy biztosít, hogy ehhez az eljáráshoz a feladó kulcsára van szükség. Az *MDC* egy m üzenethez egy $h(m)$ értéket, míg a *MAC* egy $h_k(m)$ értéket számít ki, ahol h a hasítófüggvény, és k a kulcs. *MDC* esetén $h(m)$-et valamilyen módon védeni kell a támadótól, hiszen h normális körülmények között nyilvános. Ha az eredeti adat tárolása során felmerülő változások ellenőrzésére használjuk a kivonatot, akkor elegendő, ha ezt a kivonatot az eredeti adattól elkülönítve, biztonságos helyen tároljuk. Adatátvitel esetén az egyik lehetőség, hogy miközben m-et egy nyilvános csatornán küldjük, $h(m)$ egy biztonságos csatornán kerül átvitelre. Ha viszont $h(m)$-et m-mel együtt egy nyilvános csatornán küldjük, akkor gondoskodni kell arról, hogy $h(m)$-et ne lehessen az üzenet manipulálásával együtt, annak megfelelően változtatni. Az egyik lehetőség, hogy $h(m)$-et titkosítjuk a szimmetrikus kulcsunkkal, vagy aláírjuk a nyilvános kulcsú rendszerben, és ezt a titkosított vagy aláírt kivonatot mellékeljük m-hez (ez egyben már hitelesítés is), vagy $m \parallel h(m)$-et titkosítjuk, ahol a kettős vonal a *konkatenáció*t, az egymás mellé írást jelöli, vagyis a kivonatot egyszerűen az üzenet végéhez illesztjük, és az így toldalékolt szöveget sifrírozzuk. A *MAC* esetén elegendő a kivonatot összefűzni az eredeti üzenettel.

Ismert *MDC*-algoritmusok az *MD*4, *MD*5 (***M**essage **D**igest algorithm*; az *MD*4 egyértelműen nem biztonságos, és a másikban is találtak ütközést, ezért nem javasolják a használatát), továbbá az *SHA*-1 (***S**ecure **H**ash **A**lgorithm*) és a *RIPEMD*-160 (***R**ACE* [***R**esearch and Development in **A**dvanced **C**ommunications Technology in **E**urope*] ***I**ntegrity **P**rimitives **E**valuation **M**essage **D**igest algorithm*). *MAC*-et például bármely blokkos rejtjellel elő lehet állítani *CBC*-üzemmódban, mint az utolsó blokk.

Az *MDC*-nél használt *hash-függvény*től elvárt tulajdonságok:

- legyen *ősrezisztens*, vagy másként *egyirányú*, ami azt jelenti, hogy tetszőleges x üzenethez könnyen lehessen kiszámítani a megfelelő $h(x)$ kivonatot, de szinte minden olyan y-ra, amely egy lehetséges ujjlenyomat, nehéz, tehát gyakorlatilag kivitelezhetetlen legyen olyan x-et találni, amelyre $y = h(x)$;
- legyen *második ősrezisztens*, *gyengén ütközésrezisztens*, vagyis adott x-hez legyen gyakorlatilag lehetetlen olyan, az x-től különböző x'-t találni, amellyel $h(x') = h(x)$;
- legyen (*erősen*) *ütközésmentes*, azaz legyen gyakorlatilag lehetetlen olyan x, $x' \neq x$ párt találni, hogy $h(x') = h(x)$.

A harmadik tulajdonságból következik a második. Ha ugyanis a függvény nem második ősrezisztens, akkor van olyan x, hogy $h(x)$-hez könnyen lehet egy $x' \neq x$-t találni, amelyre $h(x') = h(x)$.

Ekkor erre az x-re és $x' \neq x$-re $h(x') = h(x)$, vagyis találtunk olyan két különböző bemenetet, amelyekhez azonos függvényérték tartozik, és így a függvény nem ütközésmentes.

A *MAC*-nél alkalmazott kivonatoló-függvénnyel szembeni elvárás, hogy legyen **kiszámítás-rezisztens**, azaz adott $(x_i, h_k(x_i))$-k mellett számítástechikailag ne lehessen egy, az x_i-ktől különböző x-szel $h_k(x)$-et kiszámítani a kulcs ismerete nélkül.

A következő kérdés az *identifikáció*. Az információs biztonság megköveteli, hogy adott tevékenységet csak arra feljogosított személy végezhessen, vagyis a tevékenység megkezdése előtt igazolja a személyazonosságát. Ennek különböző megoldási módszerei vannak, amelyek három fő csoportba sorolhatóak:

- az illető birtokol valamit;
- az illető tud valamit;
- az illető inherensen rendelkezik valamivel.

Az elsőre példa egy kulcs, a harmadikra példa az ujjlenyomat. Most a másodikkal foglalkozunk.

Az identifikációs protokollal szembeni elvárások a következőek:

- ha A és B becsületes, A sikeresen tudja magát igazolni B-vel szemben;
- B ne legyen képes A egy korábbi azonosítási eljárását felhasználva A-ként azonosítani magát C-vel szemben;
- elhanyagolható legyen annak a valószínűsége, hogy egy A-tól különböző C magát A-ként igazolja B-vel szemben;
- az előbbiek akkor is teljesüljenek, ha C (polinomiálisan) sok korábbi, A és B közötti identifikációt figyelt meg, vagy korábban akár A-val, akár B-vel résztvett a protokollban, illetve, ha szimultán több folyamat résztvevője lehet C.

A leggyakoribb a jelszavas identifikáció. Ennél erősebb módszer a **kihívás – válasz** (*challenge and response*), amelyet például katonai repülőgépeken használnak a barát – ellenség felismerésére (*IFF - **I**dentification **F**riend or **F**oe*). Röviden ismertetünk néhány módszert.

Az alkalmazott módszer szerint az eljárás alapja

- szimmetrikus kulcsú rejtjelező rendszer;
- kulcsolt egyirányú hash-függvény;
- nyilvános kulcsú rejtjelező rendszer, ezen belül
  - titkosítás;
  - digitális aláírás.

Más szempontból az eljárás lehet

- egyirányú;
- kétirányú (kölcsönös).

A kihívás lehet

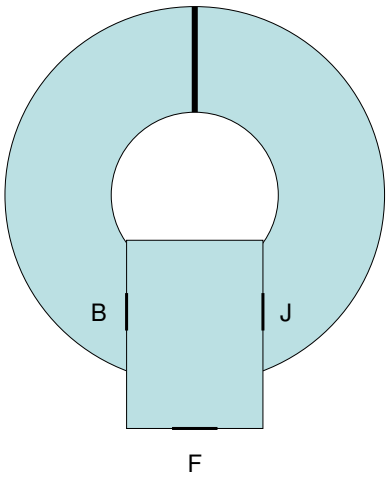
- időkeret (időpecséttel);
- véletlenszám;
- sorszám.

Az alábbi táblázat tartalmaz a fentieknek megfelelő eljárásokat.

Szimmetrikus kulcsú	Kulcsolt egyirányú hash-függvény	Nyilvános kulcsú
egyirányú		
időpecséttel (t)		
$A \to B: E_k(t_A \| B^*)$	$A \to B: t_A \| h_k(t_A \| B^*)$	$A \to B: cert_A \| t_A \| B \| S_A(t_A \| B)$
véletlen számmal (r)		
(1) $A \leftarrow B: r_B$ (2) $A \to B: E_k(r_B \| B^*)$ vagy (2) $A \to B: E_k(r_A \| r_B \| B^*)$	(1) $A \leftarrow B: r_B$ (2) $A \to B: h_k(r_B \| B^*)$ vagy (2) $A \to B: r_A \| h_k(r_A \| r_B \| B^*)$ ($SKID2$;ekkor B kötelező)	(1) $A \leftarrow B: r_B$ (2) $A \to B: cert_A \| r_A \| B \| S_A(r_A \| r_B \| B)$ vagy (1) $A \leftarrow B: h(r) \| B \| P_A(r \| B)$ (2) $A \to B: r$
kétirányú		
véletlen számmal (r)		
(1) $A \leftarrow B: r_B$ (2) $A \to B: E_k(r_A \| r_B \| B^*)$ (3) $A \leftarrow B: E_k(r_B \| r_A)$	(1) $A \leftarrow B: r_B$ (2) $A \to B: r_A \| h_k(r_A \| r_B \| B^*)$ (3) $A \leftarrow B: h_k(r_B \| r_A)$ vagy (2) $A \to B: r_A \| h_k(r_A \| r_B \| B)$ (3) $A \leftarrow B: h_k(r_B \| r_A \| A)$ ($SKID3$)	(1) $A \leftarrow B: r_B$ (2) $A \to B: cert_A \| r_A \| B \| S_A(r_A \| r_B \| B)$ (3) $A \leftarrow B: cert_B \| A \| S_B(r_B \| r_A \| A)$ vagy (1) $A \to B: P_B(r_A \| A)$ (2) $A \leftarrow B: P_B(r_A \| r_B)$ (3) $A \to B: r_B$
Megjegyzés: a *-gal jelölt adat opcionális		

Most a legerősebb módszerrel, a *ZKP*-vel (***Z**ero **K**nowledge **P**rotocol*) foglalkozunk röviden.

A *ZKP* lényege, hogy az ellenőrző személy csupán egyetlen bitnyi információ birtokába jut az azonosítás végén, nevezetesen, hogy A az-e, akinek mondja magát. A megoldást az alábbi ábra segítségével lehet megérteni.



Az ábrán **B**, **J** és **F** ajtók, míg fölül a vastag vonal egy falat reprezentál. A azt állítja, hogy keresztül tud menni ezen a falon, és erről meg akarja győzni B-t, de úgy, hogy nem akarja megmutatni neki a trükköt. Az eljárás a következő. A az **F** ajtón keresztül belép az épületbe, majd becsukja az ajtót, és vagy **B**-n, vagy **J**-n megy tovább, becsukva maga mögött ezt az ajtót is. Ezek után B belép **F**-en keresztül az előtérbe, és szól A-nak, hogy jöjjön ki mondjuk a **J** ajtón keresztül. Ha A valóban keresztül tud menni a falon, akkor bármelyik oldalon is ment be az épület belsejébe, ki tud jönni **J**-n keresztül. Persze akkor is ki tud itt jönni, ha nem igaz, amit állított, de éppen ezen az ajtón ment be, vagyis ebben az esetben is van 50%-nyi sansza a sikerre. Ha azonban pechére a másik oldalon ment be, akkor lebukik. Ez azt jelenti, hogy elég nagy esélye van arra, hogy nem bukik le (pontosan akkora, mint annak, hogy lebukik). Meggyőző ez az eredmény? Ha elbukott, akkor igen, ám, ha sikerrel vette az aka-

dályt, akkor nem túlságosan. Igen ám, de ha mondjuk tíz egymás utáni kísérlet mindegyikében a jó oldalon jelenik meg, akkor már csak 1 az 1000-hez (pontosabban az 1024-hez) az esélye, hogy mindegyik alkalommal jól teljesít, és ha még ez sem elég, akkor ennél is több próbát kérhet B. Ha összesen n fordulót játszanak le, akkor 2^{-n} annak a valószínűsége, hogy egy csalónak mindig szerencséje van, vagyis, hogy mindig előre megérzi, honnan kell majd kijönnie. Egy szemernyi kétség mindig maradhat B-ben, ha nagyon nem akar hinni A-nak, de azért a józan ész mégiscsak hajlik arra, hogy elegendően sok kísérlet után elhiggye, A valóban keresztül tud menni a falon.

Könnyű ellenőrizni, hogy teljesülnek-e az identifikációval kapcsolatban megfogalmazott elvárások.

Egy ilyen identifikációs algoritmus a *Fiat-Shamir protokoll*. Itt van mondjuk egy közös $n = pq$ modulus, ahol p és q különböző páratlan prímszám, minden résztvevőnek van egy titkos i, és nyilvános $s = i^2 \bmod n$ azonosítója. A úgy akarja igazolni magát B felé, hogy nem árulja el i-t. Ezt, mint a fentebbi példában, több fordulóban hajtja végre (annyiban, amennyit B óhajt – de azért az észszerűség határain belül). A minden fordulóban elküld B-nek egy u számot, amely, ha A becsületes, akkor egy általa ebben a fordulóban választott és titokban tartott r véletlen szám négyzetének a maradéka, vagyis $u = r^2 \bmod n$. Ekkor B visszaküld A-nak egy általa tetszés szerint választott b bitet, mire A-nak az a feladata, hogy elküldje B-nek $ri_A^b \bmod n$-et. Tegyük fel, hogy A egy v-t küldött most. B-kiszámítja $v^2 \bmod n$-et, és ezt az értéket egybeveti $us_A^b \bmod n$-nel. Ha A tényleg az, akinek mondja magát, akkor ismeri i_A-t, és becsületesen játszik, vagyis ekkor

$$\begin{aligned} v^2 \bmod n &= \left(ri_A^b \bmod n\right)^2 \bmod n = r^2 (i_A^2)^b \bmod n \\ &= (r^2 \bmod n)(i_A^2 \bmod n)^b \bmod n = us_A^b \bmod n. \end{aligned}$$

Amennyiben viszont A' nem A, csak annak mondja magát, akkor csak 50%-nyi esélye van minden fordulóban, hogy átmegy a teszten. Ha arra tippel, hogy B $b = 0$-t mond, akkor az első körben szabályosan elküldi a választott véletlen szám négyzetének maradékát, és ha B valóban a 0-ás bitet küldi, akkor A' vissza tudja küldeni r-et. Ha viszont B 1-est küld, akkor bajban lesz a hamis A', hiszen nem ismeri i_A-t, és jelenlegi tudásunk szerint összetett modulus esetén, a faktorok ismerete nélkül gyakorlatilag lehetetlen a négyzet maradékából az eredeti számot meghatározni, vagyis bukik. Ha viszont 1-re számít, akkor ravaszul u-ként nem $r^2 \bmod n$-et, hanem $v = \frac{r^2}{s_A} \bmod n$-et küldi B-nek (s_A relatív prím n-hez, mert ha nem az, akkor n-nel való legnagyobb közös osztója vagy p vagy q, és ezzel bárki ki tudja számolni bárkinek a titkos azonosítóját a megfelelő nyilvános adatból, ugyanis prímszám modulus esetén a moduláris gyökvonás könnyű feladat). Ha b valóban 1, akkor A' a második körben r-et küldi vissza, és B az ellenőrzésnél egyezőséget talál. Ám, ha a b most A' számítása ellenére 0, akkor bajban lesz az ál-A, mert most olyan t számot kellene küldenie, amellyel $t^2 \bmod n = v$, vagyis egy moduláris gyökvonást kellene végrehajtania egy összetett modulusra nézve, amelynek nem ismeri a felbontását.

A *Feige-Fiat-Shamir protokoll* az előbbi módszerhez hasonló. Most mind p, mind q 3-mal kongruens modulo 4, azaz $n = pq$ egy Blum-egész. Ekkor $\left(\frac{-1}{n}\right) = 1$, de -1 kvadratikus nemmaradék, és az n-hez relatív prím bármely u egész esetén u és $-u$ közül pontosan az egyik kvadratikus maradék. Most A választ $s_1, \cdots, s_l$, az n-hez relatív prím egészeket valamint $b_1, \cdots, b_l$ biteket, ezek lesznek a titkos azonosítói, és kiszámolja $1 \le i \le l$-re a $v_i = (-1)^{b_i}\left(s_i^2\right)^{-1} \bmod n$ nyilvános azonosítóit. Amikor igazolni akarja magát B-nél, akkor választ egy r véletlen számor és egy b bitet, és elküldi B-nek $t = (-1)^b r^2 \bmod n$-et. B visszaküld egy $e_1, \cdots, e_l$ bitsorozatot, amire A $u = r\prod_{i=1}^{l} s_i^{e_i} \bmod n$-nel válaszol. B ellenőrzi, hogy teljesül-e a $\pm u^2 \prod_{i=1}^{l} v_i^{e_i} \bmod n = t$ egyenlőség. Ha nem, akkor nem fogadja el a bejelentkezőt A-ként, ellenkező esetben újabb kört kezdeményezhet, vagy elfogad.

A protokollban a b_i bitek szerepe, hogy az összes olyan szám előfordulhasson, amelynek a Jacobi-szimbóluma 1, ne csak a kvadratikus maradékok. Ily módon az n-hez relatív prím, nála kisebb nem negatív egészek fele előfordulhat, míg a kvadratikus maradékok száma ennek csupán a fele.

Az identifikáció csak egy adott pillanatban, egy rövid ideig azonosít egy személyt, míg az integritás biztosítása önmagában egyáltalán nem biztosítja az adott dokumentum hitelességét. Ezt a feladatot az aláírás oldja meg. Az aláírással szembeni elvárásaink az alábbiak:

- legyen hiteles;
- legyen hamisíthatatlan;
- ne lehessen újra felhasználni;
- ne lehessen az aláírt dokumentumot megváltoztatni;
- ne lehessen az aláírást letagadni.

A digitális aláírás lényegesen különbözik a hagyományos aláírástól. Az utóbbi független a dokumentum tartalmától, és éppen azt várják el az aláírótól, hogy különböző időpontban más és más dokumentumon elhelyezett kézjegye nagyjából legyen azonos. Ezzel szemben az elektronikus aláírás tartalomfüggő, vagyis az aláírás különböző dokumentumokon szinte biztosan más lesz, és ez jelentősen megnehezíti a hamisító dolgát. A másik oldalon viszont a kézírásos aláírás a hordozóhoz rögzített, míg a kriptográfiai aláírás bármikor áthelyezhető egy adathordozóról egy másikra, ezért nagyon lényeges, hogy tényleg erősen függjön az aláírás az aláírt dokumentum tartalmától.

A klasszikus rejtjelezés esetén a titkosítás egyben aláírás is, hiszen csak a feladó ismerhette a rejtjelező kulcsot (feltéve, hogy minden kulcsot csak egy küldő és egy fogadó ismer). A nyilvános kulcsú rendszer esetén viszont a titkosítás semmilyen kapcsolatot nem biztosít a kulcs gazdájával, hiszen az nyilvános, bárki által hozzáférhető, ezért itt a titkosítás nem jelent egyben hitelesítést is.

A digitális aláírásnak két nagy csoportja van:

- toldalékos;
- üzenet-visszanyeréses.

Az előbbinél nem a teljes üzenetet írjuk alá, hanem annak csak a kivonatát, ami gyorsítja az eljárást. Ekkor az üzenettel együtt elküldjük az aláírt kivonatot is, és a címzett a megkapott üzenet kivonatát egybevetheti a kapott, aláírt kivonattal. A második módszer esetén a teljes üzenetet írjuk alá. Ekkor azonban megfelelő óvintézkedést kell tennünk. Tegyük fel, hogy az aláírásra a jól ismert *RSA*-t használjuk, „fordított” üzemmódban. Ekkor az m üzenet A által aláírt példánya $m' = m^{d_A} \bmod n_A$, amit valóban csak a legális küldő tud kiszámítani, és amiből a címzett könnyen ellenőrizni tudja, hogy tényleg A küldte-e, és időközben nem módosult-e az üzenet. Ehhez A nyilvános kulcsát kell használnia, hiszen $m'^{e_A} \bmod n_A = m$, de ha a számítást nem A titkos kulcsával végezték, vagy módosították az aláírt üzenetet, akkor már (szinte biztosan) nem fog teljesülni az egyenlőség. Igen ám, de az a baj, hogy most B nem tudja, mi volt m, így nem tudja ellenőrizni, hogy nem történt-e változás. A megoldás, hogy az aláírás előtt redundanciát viszünk az üzenetbe, olyan redundanciát, amelyet az aláírt, vagy hamisan aláírt üzenettel nem lehet (vagy csak nagyon vak tyúk alapon lehet) elérni. Tipikusan ilyen redundancia, hogy az üzenetet „dadogósan”, kétszer egymás mellé másolva írjuk le, és ezt írjuk alá, erre alkalmazzuk a titkos kitevőnket.

Az elektronikus aláírásról szóló törvény a digitális aláírás biztonsága szempontjából három fokozatot különböztet meg:

- elektronikus aláírás;
- fokozott biztonságú elektronikus aláírás;
- minősített elektronikus aláírás.

A törvény megfogalmazása szerint

- *Elektronikus aláírás:* elektronikus dokumentumhoz azonosítás céljából logikailag hozzárendelt és azzal elválaszthatatlanul összekapcsolt elektronikus adat, illetőleg dokumentum.

- *Fokozott biztonságú elektronikus aláírás:* elektronikus aláírás, amely megfelel a következő követelményeknek:

  a. alkalmas az aláíró azonosítására, és egyedülállóan hozzá köthető,
  b. olyan eszközzel hozták létre, amely kizárólag az aláíró befolyása alatt áll,
  c. a dokumentum tartalmához olyan módon kapcsolódik, hogy minden – az aláírás elhelyezését követően az iraton, illetve dokumentumon tett – módosítás érzékelhető.

- *Minősített elektronikus aláírás:* olyan – fokozott biztonságú – elektronikus aláírás, amely biztonságos aláírás-létrehozó eszközzel készült, és amelynek hitelesítése céljából minősített tanúsítványt bocsátottak ki.

A törvényhez kiadott irányelvek szerint a minősített aláíráshoz alkalmazott aláíráshoz az *RSA*-t és a *DSS*-t (***D**igital **S**ignature **S**tandard*, a *DSA* mint szabvány neve) ajánlott alkalmazni, ez utóbbinak az *elliptikus görbés* változatát is. Az *RSA* üzenet-visszanyeréses, bár az ilyen típus mindig használható a másik üzemmódban is, míg a második algoritmus toldalékos, hiszen fix hosszúságon dolgozik. Az üzenet-visszanyeréses technikánál, ha szükséges, az aláírt üzenetet titkosíthatjuk a címzett nyilvános kulcsával, míg a másik módszer esetén ilyenkor az üzenethez láncolt aláírással együtt titkosítjuk az üzenetet. Az *RSA* esetén tehát ekkor $E_{e_B}\left(D_{d_A}(m)\right) = \left(m^{d_A} \bmod n_A\right)^{e_B} \bmod n_B$-t küldi A a másik félnek, B-nek. Ha a két modulus különböző, márpedig majdnem mindig ez a helyzet, akkor $n_B < n_A$ esetén problémás a titkosítás, amint azt könnyű végiggondolni. Ezzel most nem foglalkozunk, hanem egy más kérdést vizsgálunk egészen röviden. Első ránézésre úgy tűnhet, hogy az előbbi transzformáció helyett $D_{d_A}\left(E_{e_B}(m)\right) = (m^{e_B} \bmod n_B)^{d_A} \bmod n_A$-t is küldhetné A, B ugyanúgy helyre tudná állítani az eredeti üzenetet. Ekkor azonban egy aktív támadó A nyilvános kulcsával kiszámolhatná $E_{e_B}(m)$-et, és utána a saját kulcsával aláírva a levelet továbbítaná azt B-nek. Ha a titkosított szöveg nem utal A-ra, akkor B azt hiszi, hogy C volt a feladó, és például válaszolva neki, C fontos információk birtokába juthat, vagyis lényeges a két transzformáció sorrendje.

A toldalékos módszerhez két algoritmust ismertetünk.

Elsőként a *DSA*-t nézzük. Az elnevezés a ***D**igital **S**ignature **A**lgorithm* (Digitális Aláírási Algoritmus) kezdőbetűiből származik. Maga az eljárás szabványosított, a szabvány neve ***D**igital **S**ignature **S**tandard* (Digitális Aláírási Szabvány). Az eljárás hash-függvényként a 160-bites SHA-1 algoritmust alkalmazza.

Ha A egy résztvevő a rendszerben, akkor választ egy 160-bites q_A prímszámot, azaz egy olyan prímet, amelyre $2^{159} < q_A < 2^{160}$. Csebisev tétele szerint minden szám és a kétszerese között van prímszám, így ilyen prímet lehet találni. Választ egy $8 \geq t_A \in \mathbb{N}$ egészt, amely a rendszer bonyolultságát határozza meg, A választásától függően, majd választ egy $2^{511+64t_A} < p_A < 2^{512+64t_A}$ prímet, vagyis egy, a t_A-tól függő nagyságú, legalább 512, legfeljebb 1024 bites prímet úgy, hogy $q_A | p_A - 1$. Most a $\mathbb{Z}_{p_A}$ egy tetszőleges, $g \neq 0$ elemével kiszámolja $g^{\frac{p_A-1}{q_A}} \bmod p_A$-t. Ha ez 1, akkor új g-vel számolunk, amíg 1-től különböző értéket nem kapunk. Legyen ez α_A. Most A választ egy $q_A > a \in \mathbb{N}^+$ egészt, és kiszámolja $y_A = \alpha_A^a \bmod p_A$-t. A titkos kulcsa a, míg az aláírása ellenőrzéséhez szükséges nyilvános adatok $(p_A, q_A, \alpha_A, y_A)$. Ezzel a rendszer paraméterei rendelkezésre állnak.

Legyen az A által aláírandó üzenet m. A választ egy véletlen $q_A > k \in \mathbb{N}^+$ egészt. k választásánál ügyelni kell arra, hogy ne ismétlődjön, mert ismétlődés esetén, ha valaki észreveszi, fel tudja törni a rendszert, hozzá tud jutni A titkos adatához, meg tudja határozni a-t, amint majd később, az általánosított AlGamal rendszernél ezt belátjuk. k választása után kiszámítja $r = \left(\alpha_A^k \bmod p_A\right) \bmod q_A$-t valamint $\kappa = k^{-1} \bmod q_A$-t, és végül $s = \kappa(h(m) + ar) \bmod q_A$-t (ügyelve arra, hogy ez utóbbi ne legyen 0). Az aláírás ekkor m-re az (r, s) pár.

Az előbbiekből következik, hogy ha $w = s^{-1} \bmod q_A$, akkor $k = w(h(m) + ar) \bmod q_A$, tehát $\alpha_A^k \equiv \alpha_A^{wh(m)} y_A^{wr}\ (p_A)$, és így $r = \left(\alpha_A^{wh(m)} y_A^{wr} \bmod p_A\right) \bmod q_A$

Ha egy $\big(m', (r', s')\big)$ pár állítólag egy üzenet A aláírásával, akkor az ellenőrzésnél először megnézzük, hogy teljesül-e $q_A > r' \in \mathbb{N}^+$ és $q_A > s' \in \mathbb{N}^+$. Ha nem, akkor nem fogadjuk el az aláírást. Ellenkező esetben kiszámítjuk $w = {s'}^{-1} \bmod q_A$-t, $u_1 = wh(m') \bmod q_A$-t és $u_2 = r'w \bmod q_A$-t, majd ellenőrizzük, hogy teljesül-e az $\big(\alpha_A^{u_1} y_A^{u_2} \bmod p_A\big) \bmod q_A = r'$ egyenlőség. Az előbbi bekezdés alapján akkor és csak akkor fogadjuk el (r', s')-t A aláírásának m'-re, ha az előbbi egyenlőség fennáll.

A második toldalékos aláírás, amellyel foglalkozunk, az **általánosított AlGamal rendszer**. Ismét A rendszerét vizsgáljuk. Legyen G az α által generált n-edrendű ciklikus csoport, $h: C^* \to \mathbf{Z}_n$, ahol C az üzenetekhez használt ábécé és $f: G \to C^*$ egy injektív függvény. A továbbiakban, a rövidség kedvéért, legyen $\varphi = hf$. A választ egy $n > a \in \mathbb{N}^+$ egészt, és kiszámítja $y = \alpha^a$-t. A nyilvános kulcsa $(\mathsf{G}, f, \alpha, y)$, míg az aláíráshoz használt titkos kulcsa a.

Amikor A aláírja az m üzenetet, akkor választ egy véletlen, n-hez relatív prím, $n > k \in \mathbb{N}^+$ egészt, és kiszámítja $r = \alpha^k$-t valamint $\kappa = k^{-1} \bmod n$-et, továbbá $s = \kappa(h(m) - a\varphi(r) \bmod n)$-et. Az A m-hez tartozó aláírása az (r, s) pár, az elküldött, aláírt üzenet $\big(m, (r, s)\big)$ lesz.

Az előbbi kifejezésből kapjuk, hogy $ks + a\varphi(r) \bmod n = h(m)$, és ebből $r^s y^{\varphi(r)} = \alpha^{h(m)}$.

Ellenőrzésnél az $\big(m', (r', s')\big)$ pár első eleméből meghatározzuk $h(m')$-t, (r', s')-ből $\varphi(r')$-t, majd ezekkel $v_1 = {r'}^{s'} y^{\varphi(r')}$-t és $v_2 = \alpha^{h(m')}$-t. Akkor és csak akkor fogadjuk el (r', s')-t mint A aláírását m'-re, ha $v_1 = v_2$.

Két lényeges kérdést említünk a rendszer használatával kapcsolatban.

Először tegyük fel, hogy A két különböző üzenetet ír alá úgy, hogy ugyanazt a k egészt használja. Ekkor a k-ból számított r-ek is megegyeznek. Legyen a két üzenet m_1 és m_2, és legyen a két aláírás (r, s_1) és (r, s_2). Most $k(s_1 - s_2) \equiv h(m_1) - h(m_1)\ (n)$, és ebben a kongruenciában egyedül k nem ismert, amely így meghatározható (a kongruenciának biztosan van megoldása, például az eredeti k). Ha viszont ismerjük k-t, akkor például $s_1 \equiv \kappa\big(h(m_1) - a\varphi(r)\big)\ (n)$-ből megkapjuk a-t, A titkos kulcsát.

A másik probléma, ha nem használunk hash-függvényt, vagyis ha $h(m) = m$. Legyen ekkor u és v két pozitív egész, az utóbbi relatív prím n-hez. Legyen most $r = \alpha^u y^v$, $s = -\varphi(r)v^{-1} \bmod n$ és $m = su \bmod n$. Ezekkel az értékekkel $v_1 = (\alpha^u y^v)^{-\varphi(r)v^{-1}} y^{\varphi(r)} = \alpha^{-u\varphi(r)v^{-1}} = \alpha^{us} = \alpha^m = v_2$, és az így meghatározott (r, s) pár A érvényes aláírása m-re, jóllehet A feltehetően nem is látta m-et.

Az eredeti AlGamal rendszert úgy kapjuk ebből az általánosított rendszerből, hogy a ciklikus csoport $\mathbb{Z}_p^*$ valamilyen p prímszámmal, h most $\mathbb{Z}_p$-be képez, és φ az identikus leképezés, továbbá $y = \alpha^a \bmod p$ és aláírásnál $r = \alpha^k \bmod p$. Az ellenőrzés annyiban módosul, hogy a $v_1 \equiv v_1\ (p)$ feltételt kell ellenőrizni.

A rendszernél ellenőrzéskor lényeges megnézni, hogy teljesül-e a $p - 1 > r \in \mathbb{N}^+$ feltétel, mert ellenkező esetben lehetőség van csalásra. Tegyük ugyanis fel, hogy (r, s) A egy legális aláírása m-re. Választunk egy tetszőleges m' üzenetet, majd kiszámítjuk $h(m')$-t, és ha $(h(m), p - 1) = 1$, akkor $u = h(m')\big(h(m)\big)^{-1} \bmod (p - 1)$-et. Ha $s' = su \bmod (p - 1)$, és r' olyan, hogy $r' \equiv ru\ (p - 1)$ és $r' \equiv r\ (p)$, akkor $y^{r'} {r'}^{s'} \equiv \Big(\alpha^{(ar + ks)(h(m))^{-1} \bmod (p-1)}\Big)^{h(m')} \equiv \alpha^{h(m')}\ (p)$, vagyis (r', s') A érvényes aláírása m'-re, amennyiben nem ellenőrizzük, hogy $p - 1 > r' \in \mathbb{N}^+$ (r'-t hatvány alapjaként r-rel, míg kitevőként ru-val helyettesítjük). Egyébként hasonló támadás a DSA-nál is lehetséges, így ott is fontos a megfelelő ellenőrzés.

Az ismertetett aláírási rendszereket használják elliptikus görbékkel is.

19. Etimológia

kripto- *gör* előtagként vminek a titkos v. rejtett voltát jelöli; titkos-, rejtett
κρύπτω (krüpto) a ***κρυπτός*** (krüptosz) *rejtett*, *titkos* görög szóból
κρύπτω (főnévi igenév: ***κρύπτειν*** krüptein) *elrejt*

-lógia, -logia *gör-lat* **1.** utótagként jelöl: vmilyen tudományt; -tan, -tudomány (pl. *geológia*) **2.** utótagként jelöl: (számnevekkel) az összetevők számát (pl. *tetralógia*) **3.** utótagként jelöl: vmilyen beszéd- v. előadásmódot (pl. *tautológia*)
λόγος (logosz) *szó*, *beszéd*, *magyarázat*, *fogalom*, *tudomány* szóból eredeti görög képzésű utótag
λόγιος, ***λογία***, ***λόγιον*** (logiosz, logia, logion) *értelmes*; *tudománnyal kapcsolatos*
λέγω (lego) (főnévi igenév: ***λέγειν*** legein) *mond*

kriptológia *gör el.* a rejtjelfejtés elmélete és gyakorlata
rejtett dolgok tudománya

-gráfia (-graphia) *gör* **1.** utótagként jelöl: vmely tudományágat (pl. *geográfia*) **2.** utótagként jelöl: vmely írás- v. más rögzítési módot (pl. *fotográfia*) **3.** utótagként jelöl: vmely nyomdászati eljárást; -nyomás (pl. *litográfia*)
γραφή (grafé) *írás* szóból görög képzésű utótag
γράφω (grafo) (főnévi igenév: ***γράφειν*** grafein) *vés*; *ír*

kriptográfia *gör el.* titkosírás, rejtjeles írás, ill. ennek rendszere, kulcsa
fn *Tud*|Titkosírások készítésének és megfejtésének módszertana.|Titkosírás. [nk: gör el.]
titkosírás (mestersége)

analízis *gör* **1.** elemzés; részekre, elemekre való bontás mint tudományos kutató módszer **2.** *mat* a matematika azon ágainak összessége, amelyek a függvény, a határérték, a differenciál és az integrál fogalmával szervesen összefüggnek, arra épülnek **3.** vegyelemzés **4.** lélekelemzés, pszichoanalízis
ανάλυσις (analüszisz) *feloldás, megoldás, darabokra szedés, megfejtés*
αναλύω (analüo) (főnévi igenév: ***αναλύειν*** analüein) *feloszt, feldarabol*
ανα- (ana-) *föl* + ***λύω*** (lüo) (főnévi igenév: ***λύειν*** lüein) *old*

kriptoanalízis *gör* **cryptanalysis** *fn* titkosírás megfejtése
titkos írás, titkos jelek megfejtése

-gram, -gramma *gör* **1.** utótagként jelent: vmilyen *-gráf* utótagú műszer által lerajzolt görbét (pl. *szeizmogram*) **2.** utótagként jelent: vmilyen ábrát v. görbét (pl. *diagram*) **3.** utótagként jelent: vmely írásművet (pl. *epigramma*)
γράμμα (gramma) *vésett*, *betű*; *rajzolás*, *írás*, *feljegyzés* ***γράφ|ω***-ból a ***-μα*** képzővel

kriptogramma *gör el.* titkosírásos v. rejtett értelmű felirat, szöveg(részlet)

entrópia: *Shannon* javaslatára entrópiának nevezzük az információ átlagos hírértékét. Az entrópia eredetileg a hőtanban használt állapotjelző neve, melyet *Rudolf Clausius* vezetett be a termodinamikai folyamatok megfordíthatóságának mértékeként. Az *εντρεπειν* (*entrepein*) görög szó, jelentése: *megfordít*. Az információelméleti és termodinamikai entrópia rokonsága a matematikai modell azonosságára vezethető vissza.

entrópia *gör* **1.** *fiz* anyagi rendszerek molekuláris rendezetlenségi fokának, ill. állapotuk termodinamikai valószínűségének mértéke **2.** *fiz* az energia hasznosíthatóságának, munkavégző képességének mértéke termikus folyamatokban **3.** *inf* a bizonytalanságnak a kapott információkkal csökkenő

arányszáma **4.** *fil* a termodinamikai állapotfüggvények hatályának hibás kiterjesztése révén létrehozott elmélet a világ hőhaláláról

εντροπία (entropia) *fordulat*

εντροπή (entropé) *fordulat*, *belefordulás*, *meghajlás, lealacsonyodás* szóból valószínűleg latin szavak mintájára képzett szó

εν- (en) *-ban*, *-ben* + **τροπή** (tropé) *fordulat* a ***τρέπω*** (trepo) (főnévi igenév:***τρέπειν*** trepein) *fordít* igéből

redundancia: 1. hétköznapi értelemben felesleg, vagyis az a többlet, amelyet a cél eléréséhez mindenképpen szükséges eszközökön túl használnak. 2. Számítástechnikai vonatkozásban az információ ábrázolására rendelkezésre bocsátott, de fel nem használt terület. Ha egy karakterlánc például 256 karakteres, de aktuális értéke csak 6 bájt hosszú, 250 redundáns bájtot tartalmaz, hiszen az eredetileg a karakterlánc számára lefoglalt tárterület nagysága nem változik. 3. Az információforrás redundanciája az egyenletes eloszlású forrás maximális entrópiájához viszonyított relatív entrópia komplementere: $R_S = 1 - H(S)/\log V$. Szemléletesen a forrás egy hírében, üzenetében rejlő, de információt, tehát újdonságot nem tartalmazó közlés arányát, a hír banalitásának fokát fejezi ki. 4. A kód redundanciája az átlagos szóhossznak az információt ténylegesen nem hordozó, vagyis a feltétlenül szükséges minimális szóhosszt meghaladó része. A szeparálható kódok esetében a minimális szóhosszt pontosan ismerjük, ilyenkor ennek és a tényleges szóhossz arányának a komplementere a kód redundanciája. Mivel $L_0 = H(S)/\log B$ (*Shannon* II. tétele), ezért $R_E = 1 - L_0/L = 1 - H(S)/(L\log B)$.

redundancia *lat* **1.** *inf* újabb információt nem adó felesleg a közleményben, amely nélkül azonban a megértés nehezebbé válna **2.** vminek redundáns volta

fn *Távk* Közlésben az egyértelmű megértéshez elvileg elegendő minimumom felüli többlet. [nk: lat]

redundantia (*túlzott*) *bővelkedés*

redundáns *lat* **1.** *inf* új információt, érdemleges közlést már nem tartalmazó **2.** terjengős, fölösleges elemeket tartalmazó

redundant-, ***redundans*** (*lat*) jelenidejű melléknévi igenév a *redundo*, *redundare túlcsordul* igéből

re-, ***red***- (fokozás)- + ***unda*** *hullám*

sifre (*fr→ném*) titkosírás, rejtjel

sifríroz *fr→ném* titkosírással ír, rejtjelez

desifríroz *fr→ném* kibetűz, titkos- v. rejtjelezett írást megfejt

chiffre *h fn* **1.** szám(jegy) **3.** titkos írásjel, rejtjel, sifrírozás; **en chiffres** sifrírozva; rejtjelben; **le Chiffre** a külügyminisztérium rejtjelosztálya **4.** rejtjel- v. sifre-kódex; titkosírás rendszere *[ábécéje]*

chiffre *n. m.* (XV^{e}, «écriture secrète»; *cifre*, 1220; lat. médiév. *cifra* «zéro», de l'arabe *sifr* «vide», *ch-* d'apr. it. *cifra*) II. 1. Caractères numériques de convention employés dans une écriture secrète (V. **Cryptographie**). *Écrire en chiffres* (*opposé à* écrire en clair). – *Par anal*. Tout signe de convention servant à correspondre secraitement, et absolt. *Le chiffre*, l'ensemble de ses signes. V. **Code**. *Changer de chiffre*. *Avoir le secret*, *la clef du chiffre*. V. **Chiffrer**, **déchiffrer**. *Service du chiffre*: bureau civil ou militaire où l'on chiffre et déchiffre les dépêches secrètes. *Être affecté au chiffre*.

cifra, **ziphra**, **zifera** (*jel*, *számjel*, *nulla*, *titkos írásjel*) késő-/középlatin szó. A klasszikusban érthetően nem létezik, mert az arab *ṣifr* XIII. századi átvétele a latin matematikai műnyelvbe. Eredetileg az arab szó a szanszkrit *sūnya* tükörfordítása. Számos európai nyelvben jelen van: ***Ziffer*** (ném., ciffer), ***chiffre*** (fr., sifr), ***cifra*** (ol., csifra)... A *titkos írásjel* értelme a diplomácia köreiből fejlődött, ugyanis itt gyakran alkalmaztak számjegyeket titkosított írásokban. A magyarba is eredetileg hasonló

értelemben került be, majd a zérus, a kis kör forma díszítőelemként való alkalmazása elvezetett a *cifra*, *díszes*, *tarka* értelemhez is.

zéró, zérus (*arab→ol*) **1.** nulla, semmi **2.** *biz* senki; jelentéktelen ember

صِفْر (szifr) **صُفْر** (szufr) **صَفِر** (szafir) **صُفْر** (szufur) **صَفْر** (szafr) többesszáma: **أصْفار** (’ászfár) *üres*, *haszontalan*, *értéktelen*, *mentes* (**مِن** (min) *vmitől*)

صِفْر (szifr) *zérus*, *zéro*, *nulla*, *semmi*

kommunikáció *lat* **1.** tájékoztatás, (hír) közlés **2.** *inf* információk közlése v. cseréje vmilyen erre szolgáló eszköz, ill. jelrendszer (nyelv, média, gesztusok stb.) útján **3.** *ritk* közlemény **4.** összeköttetés, kapcsolat, érintkezés

communicatio *közlés, a közlés folyamata*

communis, communæ *közös*

communico, communicare *megoszt, közöl, közössé tesz*

kommunikál *lat* közöl (vmit vmivel, vkivel); értesít (vkit)

információ *lat* **1.** felvilágosítás, tájékoztatás **2.** hírközlés **3.** értesülés, adat **4.** híranyag, a közlés tárgya **5.** *inf* elektronikus úton továbbított jel; hír

informatio *formába öntés*; *közlés átvitele*

informo, ***informare***, ***informavi***, ***informatum*** az ***in*** (*-ba*, *-be*, *-ban*, *-ben*) praepositio – igekötő – és a ***forma***, ***formae*** f(emininum) (*alak*) összetétele. A *forma* szó a ***fero ferre tuli latum*** (*hoz*, *visz*) ige gyökének minőségi hangmásulás (qualitatív ablaut – gyakori jelenség az indoeurópai nyelvekben) szenvedett alakja és egy főnévképző (*ma* suffixum) egyesüléséből van. *Informo = alakot ad*, *formába önt*, képletesen *szavakban*, *szavakkal formál meg*, azaz *közöl*.

kód (*lat→fr*) **1.** *inf* megállapodás szerinti jelek v. szimbólumok rendszere, amellyel vmely információ továbbítható és visszaalakítható **2.** *biol* → genetikai kód **3.** rejtjeles ábécé kulcsa **4.** *inf* jelábécé (sürgönynél, távírónál stb.)

(fn) **1**. *Tud* Megállapodás szerinti jelek v. szimbólumok rendszere, amellyel vmely információ egyértelműen visszaadható. **2**. *ritk* Jelkulcs [nk:fr<lat]

code *n. m.* (1220; lat. jur. *codex* «planchette, recueil»). **1**. Recueil de lois. ... **4**. Recueil de conventions ; dictionnaire des équivalences entre deux langages (*spécialt*. un langage naturel et un langage non naturel). *Code de signaux*. *Code secret*. V. **Chiffre**

caudex, codex (*fa*)*törzs*, *rönk*, *tuskó*, *tönk*; *dokumentum*, eredetileg *fa írótábla*-ból. Ebből származik a magyar ***kódex*** szó.

code *h fn*, **1.** *jog* törvénykönyv, jogszabálygyűjtemény, kódex ... **2.** kód, jel-/betűkódex; **code binaire** bináris kód; **code biquinaire** bikvináris kód; **code cyclique** ciklikus kód; **code détecteur d'erreurs** hibakereső kód; **code génétique** genetikai kód; **code points-traits** Morze-ábécé; **code télégraphique** sürgönyjel-ábécé; *vill* **code temporel** időkód; **code à adresses multiples** többcímű kód; **code à redondance** redundáns kód; **code de contrôle** ellenőrző kód; **code de correction d'erreurs** hibajavító kód; **code de graph** gráfkód; **code de signaux** jelzési utasítás; **télégramme en code** rejtjeles, sifrírozott sürgöny; **code d'instructions** utasításrendszer; **mettre en code** kódol, rejtjelez **3.** kód(szám); **code-barre**, **code à barres** termékkód; **code génétique** genetikai kód; **code postal** irányítószám; **code de comptes** folyószámla (kód)száma; **code pour carte bancaire** PIN-kód **4. le code** a szabályzat

Tárgymutató

M

N

O,Ó

P

R

S

Sz

T

V

Z

Irodalomjegyzék

Berlekamp, E. R.
Algebraic Coding Theory
McGraw Hill, 1968.

Csiszár, I., Fritz, J.
Információelmélet
Tankönyvkiadó, 1986.

Gonda, J.
Véges testek
compalg.inf.elte.hu/material/DOWNLOAD/vt.pdf, 2007.

Gonda, J.
Hibakorlátozás
compalg.inf.elte.hu/material/DOWNLOAD/hibakor.pdf, 2007.

Gonda, J.
A rejtjelezés néhány kérdése
http://www.inf.elte.hu/karunkrol/digitkonyv/Jegyzetek2010/A_rejtjelezes_nehany_kerdese.pdf
2010.

Györfi, L., Győri, S., Vajda, I.
Információ- és kódelmélet
Typotex Kiadó, 2000.

Györfi, L., Vajda, I.
A hibajavító kódolás és a nyilvános kulcsú rejtjelezés elemei
Műegyetemi Kiadó, 1990.

Huffman, W. C., Pless, V.
Fundamentals of Error-Correcting Codes
Cambridge University Press, 2003.

Linder, T., Lugosi, G.
Bevezetés az információelméletbe
Műegyetemi Kiadó, 1993.

Lint, J. H. van
Introduction to Coding Theory
Springer Verlag, 1982.

Lucky, R. W., Salz, J., Weldon, E. J.
Adatátvitel
Műszaki Könyvkiadó, 1973.

MacWilliams, F. J., Sloane, M. J. A.
The Theory of Error-Correcting Codes
North-Holland, 1977

Reza, F. M.
Információelmélet
Műszaki Könyvkiadó,

Roman, S.
Coding and Information Theory
Springer Verlag, 1992.

Shannon, C. E.
A mathematical theory of communication
Bell System Technical Journal vol. 27., 1948.

Shannon, C. E.
Communication theory of secrecy system
Bell System Technical Journal vol. 28., 1948.

Shannon, C. E., Weaver, W.
A kommunikáció matematikai elmélete
Országos Műszaki Információs Központ és Könyvtár, Budapest, 1986.

Tsfasman, M. A., Vlăduț, S. G.
Algebraic-Geometric Codes
Kluwer Academic Publishers, 1991.

Vajda, I.
Hibajavító kódolás és műszaki alkalmazásai
BME Mérnöki Továbbképző Intézete, 1982.

Beutelspacher, A.
Cryptology
The Mathematical Association of America, 1994.

Brassard, G.
Cryptology
Springer, 1989.

Buttyan, L., Vajda, I.
Kriptográfia és alkalmazásai
Typotex, 2004.

Diffie, W., Hellman, M. E.
New directions in cryptography
IEEE Trans. on Info. Theory, IT-22, 1976.

Kahn, D.
The codebreakers. The story of secret writing
McMillan, 1967.

Ködmön, J.
Kriptográfia
Computerbooks, 1999.

Menezes, A. J., Ooorshot, P. C., Vanstone, S. A.
Handbook of Applied Cryptography
CRC Press, 1996.

Nemetz, T., Vajda, I.
Algoritmusos adatvédelem
Akadémiai Kiadó, 1991.

Révay, Z.
Titkosírások
Zrínyi Kiadó, 1978.

Salomaa, A.
Public-key Cryptography
Springer, 1990.

Shing, A.
Kódkönyv. A rejtjelezés és a rejtjelfejtés története
Park Könyvkiadó, 2001.

Tilborg, H. C. A. van
An introduction to cryptology
Kluwer, 1988.

Virasztó, T.
Titkosítás és adatrejtés
NetAkadémia, 2004.