SZÉCHENYI ISTVÁN
EGYETEM
GYŐR

KÓDOLÁSELMÉLET

Nagy Szilvia

1. Lineáris blokk-kódok

Shannon hírközlési modellje

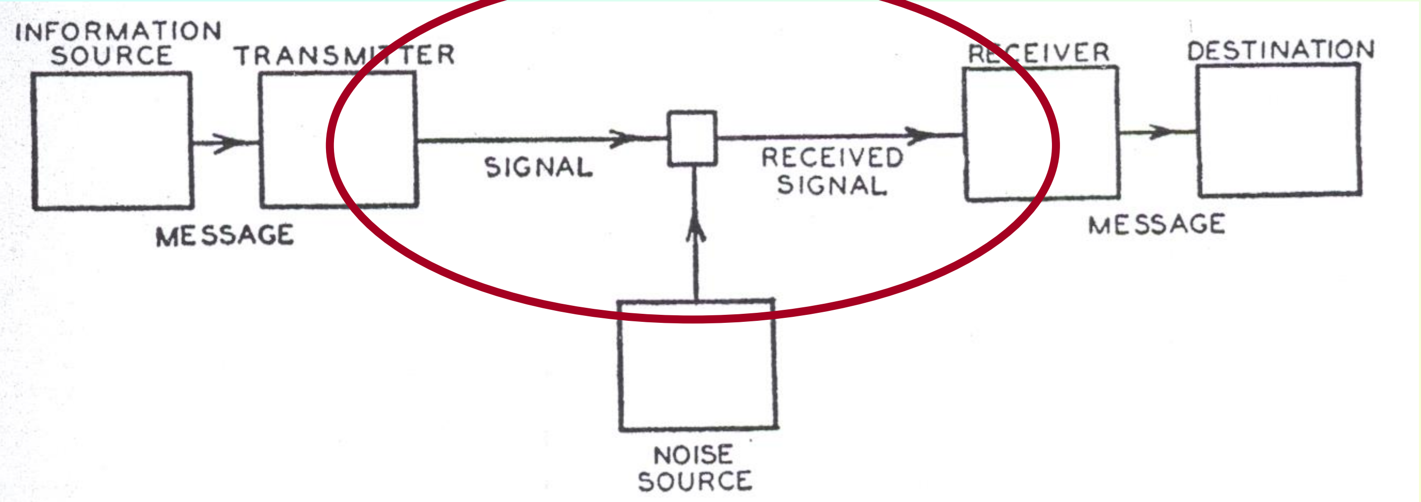


Fig. 1—Schematic diagram of a general communication system.

információforrás
adó

csatorna – zajforrás

vevő
rendeltetési hely

Kódok halmaza, csatornakódolás

A C^n tér azon K részhalmazát, amelyet a kódszavak alkotnak, **kód**nak nevezik.

- Csatornakódolás:

$$F : B^l \mapsto K$$

- Dekódolás:
  - döntés:

$$G : C^n \mapsto K$$

  - a kódolás inverze:

$$F^{-1} : K \mapsto B^l$$

Kódtávolság, javítható hibák száma

Egy *K* kód **kódtávolsága**:

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

a kódszavak közötti Hamming-távolság minimuma.

Hibajelzés lehetséges, ha a **c** kódszavunkból keletkezett **v** nem egy másik érvényes kódszó: $\mathbf{v}\notin K$. Ha ν a hibák száma, akkor $\nu < d_{\min}$ hibát lehet biztosan jelezni.

Törléses hibák javítása: $\nu < d_{\min}$

Egyszerű hibák javítása: $\nu < (d_{\min}-1)/2$

Kódtávolság, javítható hibák száma

Egy K kód **kódtávolsága**:

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

a kódszavak közötti Hamming-távolság minimuma.

Hibajelzés lehetséges, ha a **c** kódszavunkból keletkezett **v** nem egy másik érvényes kódszó: $\mathbf{v} \notin K$. Ha ν a hibák száma, akkor $\nu < d_{\min}$ hibát lehet biztosan jelezni. Hibajelzés után általában megismétlik az üzenetet.

Kódtávolság, javítható hibák száma

Blokk-kódok

Shannon hírközlési modellje

Kódtávolság

Singleton-korlát

Hamming-korlát

Csatorna-kódolási tétel

LDPC kódok

<u>Törléses hiba javítása</u>: ezesetben tudjuk a hibák helyét. A **v** hibás vett vektort abba a kódszóba javítjuk, amelyik a hibás pozícióktól eltekintve azonos **v**-vel. Ha több ilyen van, nem tudunk javítani. Ha a két legközelebbi kódszóból d_{min} komponenst a megfelelő helyről törlünk, akkor azonos maradékot kapunk, ennél kevesebb elem törlésével sehogy sem kaphatunk azonos maradékot.
Így $\nu \le d_{min} - 1$ törléses hiba javítható.

4 ■ 5 0 1 3 0 ■ ■ 4 6 ■ 1 0
4 ■ 5 0 1 3 0 ■ ■ 4 6 ■ 1 0

ν=1 hiba javítható: a két vektor különbözik

ν=4 nem javítható

Kódtávolság, javítható hibák száma

Egyszerű hiba javítása: nem tudjuk a hibák helyét. A **v** hibás vett vektort abba a **c** kódszóba javítjuk, amelyikre $d(\mathbf{v},\mathbf{c})$ a legkisebb. Ha több ilyen van, nem tudunk javítani. A javíthatóság feltétele:

$$d(\mathbf{c},\mathbf{v}) < d(\mathbf{c}',\mathbf{v}).$$

A háromszög-egyenlőtlenség szerint:

$$d(\mathbf{c},\mathbf{c}') \le d(\mathbf{c},\mathbf{v}) + d(\mathbf{v},\mathbf{c}') = d(\mathbf{c},\mathbf{v}) + d(\mathbf{c}',\mathbf{v})$$

$$d(\mathbf{c},\mathbf{c}') - d(\mathbf{c},\mathbf{v}) \le d(\mathbf{c}',\mathbf{v})$$

$$d(\mathbf{c},\mathbf{v}) < d(\mathbf{c},\mathbf{c}') - d(\mathbf{c},\mathbf{v})$$

$$2d(\mathbf{c},\mathbf{v}) < d(\mathbf{c},\mathbf{c}') \quad \Rightarrow \quad d(\mathbf{c},\mathbf{v}) < \frac{1}{2} d_{\min}$$

A javítható egyszerű hibák száma $\nu \le \dfrac{d_{\min} - 1}{2}$

Singleton-korlát

Legyen a kódábécé elemszáma r , a kódszavak hossza n , száma M , a kódtávolság pedig $d_{\min}$. A Singleton-korlát szerint

$$M \le r^{n-d_{\min}+1}$$

Bizonyítás: Az r elemből felépülő k hosszúságú sorozatok száma r^k .
Legyen $r^{k-1} < M \le r^k$.
Több kódszó van (M db) mint ahány $k-1$ hosszú sorozat, így
$\exists\, \mathbf{c}_i , \mathbf{c}_j \in K$,
melyeknek az első $k-1$ eleme azonos.
Ezekre
$d(\mathbf{c}_i , \mathbf{c}_j) < n-(k-1)$, így $d_{\min} < n-(k-1)$.

Singleton-korlát

Legyen a kódábécé elemszáma r , a kódszavak hossza n , száma M , a kódtávolság pedig $d_{\min}$. A Singleton-korlát szerint

$$M \le r^{n-d_{\min}+1}$$

Bizonyítás: Az r elemből felépülő k hosszúságú sorozatok száma r^k .

Legyen $r^{k-1} < M \le r^k$.

$$d_{\min} \le n - k + 1$$

$$k \le n - d_{\min} + 1$$

$$M \le r^k \le r^{n-d_{\min}+1}$$

Singleton-korlát

M egyértelműen megadja *k*-t, az $r^{k-1} < M \le r^k$ -nak egyetlen egész megoldása van: $\log_r M$ egészrésze. A Singleton-korlát szerint

$$M \le r^{n-d_{\min}+1}$$

Behelyettesítve *k*-t: $r^k \le r^{n-d_{\min}+1}$

majd *r* -alapú logaritmust véve: $k \le n - d_{\min} + 1$

és átrendezve a kódtávolságnak a kódszavak számától függő maximumát kapjuk:

$$d_{\min} \le n - k + 1 \le n - \log_r M + 1$$

Singleton-korlát

$$d_{min} \le n - k + 1 \le n - \log_r M + 1$$

Az olyan kódok, amelyeknél mindkét helyen egyenlőség áll, **maximális távolságú kód**ok (MDS – Maximum Distance Seprable)

A *k* szám és *n*, a kódszavak hossza szokott a kód két paramétere lenni.

Hamming-korlát

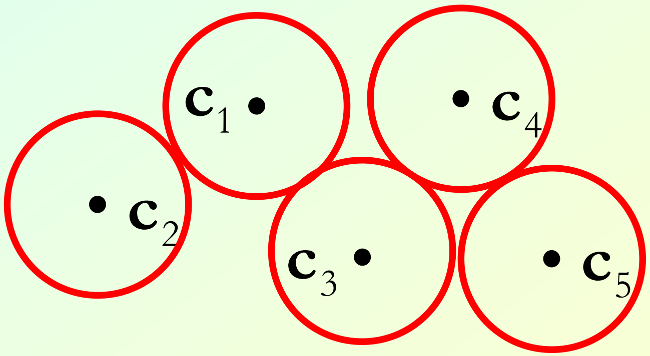
Legyen a kódábécé elemszáma r , a kód paraméterei (n, k) , a javítandó hibák száma ν. A Hamming-korlát (gömbpakolási korlát) szerint

$$r^k \sum_{i=1}^{\nu} \binom{n}{i} \cdot (r-1)^i \leq r^n$$

Bizonyítás: A C^n térben a $\mathbf{c}_i \in K$ kódszavak pontok; egymástól minél távolabb vannak, $d_{\min}$ annál nagyobb, így annál több hibát tudunk javítani.

Hamming-korlát

Akkor javítunk egy $v \in C^n$ hibás vektort a $\mathbf{c}_i$ kódszóba, ha az a $\mathbf{c}_i$ körüli ν sugarú gömbön belül van. Ezek a gömbök nem fedhetnek át, azaz az összes gömbben levő elemek száma nem lehet nagyobb, mint r^n, C^n elemszáma.



A $\mathbf{c}$ kódszótól pontosan i helyen, a $j_1, \ldots, j_i$-edik helyeken eltérő vektorok száma:

$$(r-1)^i$$

Hamming-korlát

A $j_1, \ldots, j_i$ pozíciók megválasztása $\binom{n}{i}$ -féleképpen lehet.

A **c** kódszótól <u>legfeljebb</u> ν helyen eltérő vektorok száma: $\sum_{i=1}^{\nu} \binom{n}{i} \cdot (r-1)^i$

Összesen r^k darab kódszó van, mindegyik körül egy-egy $\sum_{i=1}^{\nu} \binom{n}{i} \cdot (r-1)^i$ sugarú gömb.

Egyetlen olyan vektor sincs, amely több gömbbben is benne lenne, így a gömbök elemszámainak összege nem lehet több, mint a teljes C^n halmaz elemszáma, r^n:

$$r^k \sum_{i=1}^{\nu} \binom{n}{i} \cdot (r-1)^i \le r^n$$

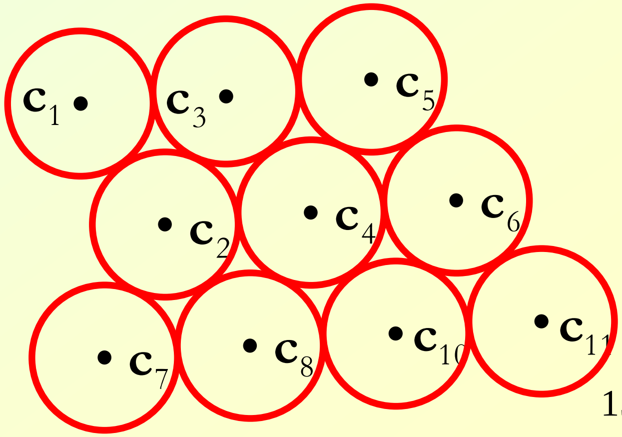
Perfekt kódok

Azokat a *K* kódokat, amelyekre a Hamming-korlátban egyenlőség teljesül, azaz

$$r^k \sum_{i=1}^{v} \binom{n}{i} \cdot (r-1)^i = r^n$$

perfekt kódoknak nevezzük.

Az ilyen kódoknál a teljes C^n teret kitöltik a gömbök, szorosan illeszkednek egymáshoz, a kódszavak egyenletesen helyezkednek el a téren belül (Hamming-távolságot véve), adott *n* szóhosszra maximális számú kódszót tartalmaznak.



Kódsebesség (jelsebesség)

Az információátvitel gyorsasága jellemez-hető a

$$R = \frac{H(K)}{n}$$

kódsebességgel, avagy **jelsebesség**gel.
(egy szimbólumra jutó átlagos információ)

Kódsebesség (jelsebesség)

Legyen a kódszavak előfordulási valószínűsége azonos, 1/M. Az entrópia ekkor

$$H(K) = -\sum_{i=1}^{M} \frac{1}{M} \log_2 \frac{1}{M} = \log_2 M,$$

a kódsebesség pedig

$$R = \frac{\log_2 M}{n}.$$

Ha a kódnak csak a legáltalánosabb paraméterei (szóhossz, betűszám) ismertek, ez jó felső becslés a jelsebességre.

Shannon csatornakódolási tétele

Ha egy C kapacitású diszkrét, memória-mentes csatornán

- $R < C$, akkor lehet olyan n kódszóhosszt találni, hogy a hibás dekódolás valószínűsége tetszőleges $\varepsilon > 0$ számnál kisebb legyen. <u>Az n növelésével</u> csökken ε minimuma, azaz <u>csökken a hibás dekódolás valószínűsége.</u>
- $R > C$, akkor nem lehet olyan n kódszóhosszt találni, hogy a hibás dekódolás valószínűsége tetszőlegesen kicsi legyen.

Shannon csatornakódolási tétele

$R>C$ esetén n növelésével a hibás dekódolás valószínűsége,

$$1-\frac{C}{R}-\frac{1}{nR}$$

is nő.

A tétel nem ad meg módszert jó csatornakódok létrehozására, csak azt mondja ki, hogy jelsebességük kisebb, mint a csatornakapacitás.

Shannon csatornakódolási tétele

$R>C$ esetén n növelésével a hibás dekódolás valószínűsége,

$$1-\frac{C}{R}-\frac{1}{nR}$$

is nő.

A tétel nem ad meg módszert jó csatornakódok létrehozására, csak azt mondja ki, hogy jelsebességük kisebb, mint a csatornakapacitás.

Shannon csatornakódolási tétele

Bizonyítás:

Kölcsönös tipikus sorozatok: x és y kölcsönösen tipikus, egy P(X), P(Y), P(X,Y) valószínűségeloszlásra, ha

$$\left|\frac{1}{N}\log\frac{1}{P(\mathbf{x})} - H(X)\right| < \beta,$$

$$\left|\frac{1}{N}\log\frac{1}{P(\mathbf{y})} - H(Y)\right| < \beta,$$

$$\left|\frac{1}{N}\log\frac{1}{P(\mathbf{x},\mathbf{y})} - H(X,Y)\right| < \beta.$$

Ha x, y véletlen választott sorozatok, kölcsönös tipikusságuk vsze→1 ha $N\to\infty$

$$|J_{N\beta}| \le 2^{N(H(X,Y)+\beta)};$$

$$P((\mathbf{x}',\mathbf{y}') \in J_{N\beta}) \le 2^{-N(I(X;Y)-3\beta)}.$$

LDPC kódok

Low Density Parity Check

Gallager 1960

w_s = egyesek száma a sorokban

w_o = egyesek száma az oszlopokban

Pl:

$$\mathbf{H} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

itt $w_s = 4$ és $w_o = 2$ minden sorban és oszlopban. (nem LD, de reguláris)

LD, ha $w_s << n$, és $w_o << (n - k)$

Reguláris, ha w_s és w_o azonos minden sorban/oszlopban

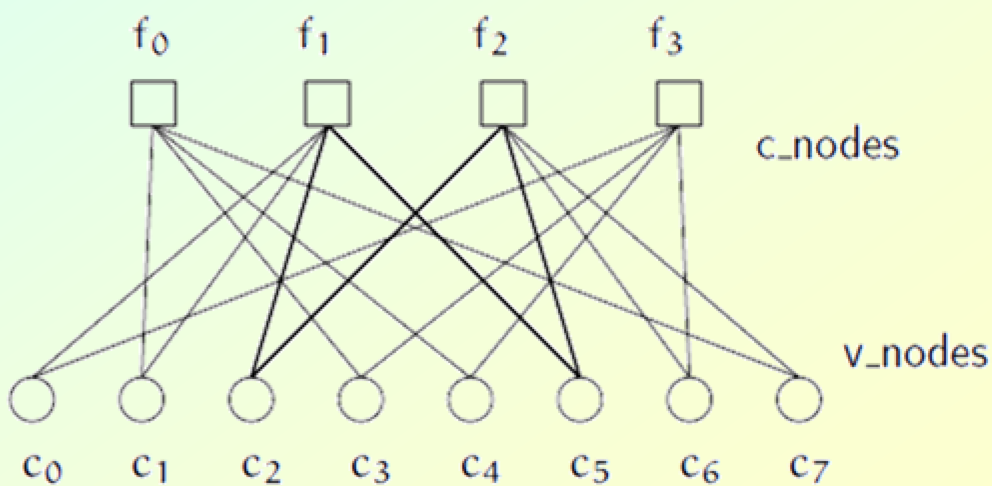
LDPC kódok

Blokk-kódok

- Shannon hírközlési modellje
- Kódtávolság
- Singleton-korlát
- Hamming-korlát
- Csatorna-kódolási tétel
- LDPC kódok

Tanner-gráf: Az LDPC kódok egy leírása, egy bipartit gráf, változó csomópontok (v) és ellenőrző csomópontok (c)

$$\mathbf{H} = \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$



LDPC kódok

Generálás: Többnyire szisztematikus kód, paritásmátrixot általában álvéletlen módon hozzák létre, sőt a jó kódok irregulárisak (pl. 0.04 dB-vel a Shannon-korláton belüli R, 10^{-6} BER, $n \approx 10^7$)

Nagy blokkhossz miatt iteratív kódolás – divide and conquer

LDPC kódok

Dekódolás: „belief propagation” (avagy message passing avagy sum-product) algoritmussal

1. minden ellenőrző csomópont kap egy-egy üzenetet a hozzájuk tartozó változó csomóponttól arról, hogy szerintük, milyen bit tartozik hozzájuk.
2. az ellenőrző csomópontok kiszámolnak egy választ: milyen bitnek kellett volna tőle jönni, ha a többi változó csomópont jó értéket küldött.

LDPC kódok

Dekódolás: „belief propagation” (avagy message passing avagy sum-product) algoritmussal

1. minden ellenőrző csomópont kap egy

c-node	received/sent				
f_0	received:	$c_1 \to 1$	$c_3 \to 1$	$c_4 \to 0$	$c_7 \to 1$
	sent:	$0 \to c_1$	$0 \to c_3$	$1 \to c_4$	$0 \to c_7$
f_1	received:	$c_0 \to 1$	$c_1 \to 1$	$c_2 \to 0$	$c_5 \to 1$
	sent:	$0 \to c_0$	$0 \to c_1$	$1 \to c_2$	$0 \to c_5$
f_2	received:	$c_2 \to 0$	$c_5 \to 1$	$c_6 \to 0$	$c_7 \to 1$
	sent:	$0 \to c_2$	$1 \to c_5$	$0 \to c_6$	$1 \to c_7$
f_3	received:	$c_0 \to 1$	$c_3 \to 1$	$c_4 \to 0$	$c_6 \to 0$
	sent:	$1 \to c_0$	$1 \to c_3$	$0 \to c_4$	$0 \to c_6$

2.

jó értéket küldött.

LDPC kódok

Dekódolás: „belief propagation” (avagy message passing avagy sum-product) algoritmussal

1. minden ellenőrző csomópont kap egy-egy üzenetet a hozzájuk tartozó változó csomóponttól arról, hogy szerintük, milyen bit tartozik hozzájuk.
2. az ellenőrző csomópontok kiszámolnak egy választ: milyen bitnek kellett volna tőle jönni, ha a többi változó csomópont jó értéket küldött.
3. a változó csomópontok eldöntik, vajon jó bitet kaptak-e → új üzenet

LDPC kódok

Blokk-kódok

Shannon hírközlési modellje

Kódtávolság

Singleton-korlát

Hamming-korlát

Csatorna-kódolási tétel

LDPC kódok

Dekódolás: „belief propagation” (avagy message passing avagy sum-product) algoritmussal

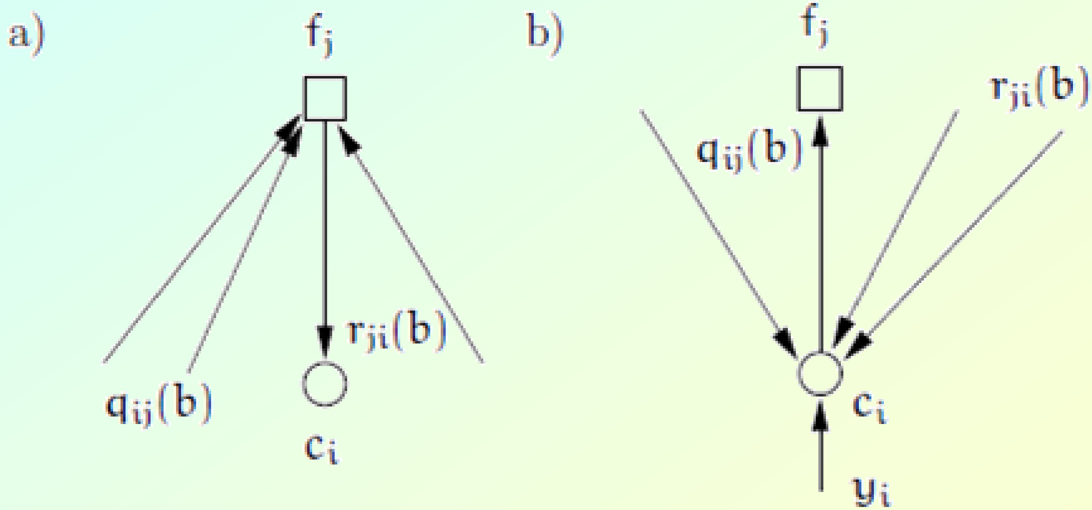
1. minden ellenőrző csomópont kap egy-egy üzenetet a hozzájuk tartozó változó csomóponttól arról, hogy szerintük, milyen bit tartozik hozzájuk.
2. az ellenőrző csomópontok kiszámolnak … ont …
3. … jó

v-node	y_i received	messages from check nodes		decision
c_0	1	$f_1 \to 0$	$f_3 \to 1$	1
c_1	1	$f_0 \to 0$	$f_1 \to 0$	0
c_2	0	$f_1 \to 1$	$f_2 \to 0$	0
c_3	1	$f_0 \to 0$	$f_3 \to 1$	1
c_4	0	$f_0 \to 1$	$f_3 \to 0$	0
c_5	1	$f_1 \to 0$	$f_2 \to 1$	1
c_6	0	$f_2 \to 0$	$f_3 \to 0$	0
c_7	1	$f_0 \to 1$	$f_2 \to 1$	1

LDPC kódok

Lágy döntéssel:

1. $q_{ij}(1) = P_i \quad q_{ij}(0) = 1 - P_i.$



2. $$r_{ji}(0) = \frac{1}{2} + \frac{1}{2} \prod_{i' \in V_j \setminus i} (1 - 2q_{i'j}(1))$$

$$r_{ji}(1) = 1 - r_{ji}(0)$$

LDPC kódok

Lágy döntéssel:

1. $q_{ij}(1) = P_i$ $\quad q_{ij}(0) = 1 - P_i.$

2. $$r_{ji}(0) = \frac{1}{2} + \frac{1}{2} \prod_{i' \in V_j \setminus i} (1 - 2q_{i'j}(1))$$
$$r_{ji}(1) = 1 - r_{ji}(0)$$

3. $$q_{ij}(0) = K_{ij}(1 - P_i) \prod_{j' \in C_i \setminus j} r_{j'i}(0) \qquad Q_i(0) = K_i(1 - P_i) \prod_{j \in C_i} r_{ji}(0)$$
$$q_{ij}(1) = K_{ij} P_i \prod_{j' \in C_i \setminus j} r_{j'i}(1) \qquad Q_i(1) = K_i P_i \prod_{j \in C_i} r_{ji}(1)$$
$$\hat{c}_i = \begin{cases} 1 & \text{if } Q_i(1) > Q_i(0), \\ 0 & \text{else} \end{cases}$$

4. gt 2.