Teoria de la Informació i la Codificació

Jordi Quer

7 de febrer de 2024

Aquestes notes són una recopilació d'apunts i llistes de problemes de cursos impartits per l'autor en titulacions d'Informàtica, Matemàtiques i Ciència i Enginyeria de Dades de la UPC. En part provenen de material el·laborat per altres professors, especialment Josep Grané, Josep Maria Brunat, José Luís Ruíz, Fernando Martínez i Josep Vidal.

Es tracta de material de treball en el que seguerament hi ha moltes errades tipogràfiques, gramaticals i de contingut: imprecisions, inconsistències i errors. S'agrairà qualsevol suggeriment o correcció.

Índex

Introducció

La *Teoria de la Informació* neix l'any 1948 quan l'enginyer i matemàtic *Claude Shannon* (1916–2001) publica l'article *"A mathematical theory of information"* [27] a la *revista científica* sobre temes de *telecomunicacions* publicada pels laboratoris Bell.

En aquest article Shannon proposa un model matemàtic per al concepte "informació" i una manera de mesurar-la. La informació es tradueix en dades, les quals es transmeten a traves de canals de comunicació. Shannon estudia i resol dos dels problemes principals dels canals de comunicació: la seva eficiència (quantitat de dades que es poden transmetre a través d'un canal donat) i la seva fiabilitat (com transmetre les dades per evitar els errors que introdueix el canal). Les seves solucions venen donades en els dos resultats principals de l'article: el *teorema de codificació de font* i el *teorema de codificació de canal*.

En un article posterior [29] Shannon estudia i resol un altre problema dels canals de comunicació: la privacitat (en el sentit de mantenir el secret en la comunicació), demostrant que el sistema de xifrat conegut com a *OTP* és l'únic que garanteix la seguretat perfecta en les comunicacions.

Informació i entropia. Shannon suggereix agafar com a model matemàtic de la informació una variable aleatòria X i proposa que la quantitat d'informació continguda en aquesta variable es mesuri com l'esperança de la variable aleatòria $H(X) = -\log\big(p(X)\big)$, que anomena entropia de la variable. Agafar el logaritme en una base o una altra correspon a canviar la unitat de mesura. Habitualment s'agafen els logaritmes en base 2 i en aquest cas la informació es mesura en bits. A la secció 2 s'estudiarà aquesta teoria: es veurà en quin sentit la variable modelitza la informació i perquè l'entropia és una bona mesura, i es veuran propietats de l'entropia a partir de la teoria de la probabilitat.

Canal de comunicacions. La informació es tradueix a dades. Per aclarir les idees aquí es consideraran només dades discretes, que són seqüències de lletres d'un alfabet. El cas més importat és el de l'alfabet binari: aquí les dades són simplement seqüències de zeros i uns, que representen per exemple un text (en ASCII), una imatge (els píxels), un so digital, etc. A la secció 1 es veurà la manera com la informació es codifica o transforma en dades discretes usant codis sobre un alfabet.

L'objectiu de la comunicació és enviar les dades d'un emissor a un receptor usant un canal. Això se sol representar de la forma següent:

En la pràctica els canals de comunicació físics tenen deficiències que cal intentar superar: tenen limitacions de capacitat, introdueixen errors en les dades transmeses i poden ser interferits per usuaris no autoritzats. La solució a aquests problemes són la compressió de dades, la correcció d'errors i la criptografia.

Compressió de dades. L'objectiu de les tècniques de compressió és representar una informació amb seqüències binàries el més curtes possible. Per fer-ho es fan servir codis de longitud variable (secció 1.3) construïts de tal manera que les dades més probables es codifiquen amb paraules curtes i les menys probables amb paraules més llargues. Shannon, en el seu teorema de codificació de font, estableix el límit teòric fins on es pot comprimir una informació, en funció de la seva entropia. Hi ha tècniques de codificació que permeten apropar-se molt a aquest límit. A la secció 3 es demostrarà el teorema i es veuran alguns dels mètodes de codificació més importants.

La compressió que es considera en teoria de la informació és la compressió sense pèrdua. Hi ha altres tècniques de compressió en què les dades recuperades en descomprimir no són exactament iguals que les dades originals, que s'anomenen de compressió amb pèrdua. Estan basades en altres consideracions que tenen a veure amb la percepció humana del so i la imatge i usen tècniques d'anàlisi de Fourier.

Correcció d'errors. Els codis correctors d'errors afegeixen redundància a la informació de tal manera que això permeti corregir els errors introduïts pel canal durant la transmissió (secció 1.5). Shannon defineix el concepte de capacitat d'un canal de comunicacions i demostra el seu teorema de codificació de canal, en què veu l'existència de codis correctors d'errors que són capaços de corregir essencialment tots els errors a canvi d'afegir una proporció de redundància que només depèn de la capacitat. A la secció 4 s'estudia la capacitat dels canals de comunicació i s'enuncia i demostra el teorema.

El teorema de Shannon no és constructiu: no dona una manera de construir a la pràctica codis amb bones propietats correctores. La teoria de codis s'ocupa d'això: trobar codis correctors d'errors amb molta capacitat correctora i també dissenyar algorismes eficients de codificació i descodificació. A les seccions 5 i 6 es veuran les idees d'aquesta teoria i alguns dels codis més importants que es fan servir en les aplicacions pràctiques.

Criptografia. Hi ha tres aspectes principals de la seguretat en les comunicacions: (1) la privacitat requereix que algú que tingui accés a les dades enviades a través del canal no pugui saber la informació que aquestes dades codifiquen; això s'aconsegueix amb tècniques de *xifrat* en què emissor i receptor *comparteixen una clau*; (2) la integritat consisteix en què tota modificació introduïda en les dades mentre passen pel canal sigui detectada pel receptor; s'aconsegueix usant funcions *hash criptogràfiques*; (3) l'autenticitat permet verificar la identitat de l'emissor; s'aconsegueix amb *firmes digitals*, que són tècniques de *criptografia de clau pública* en què emissor i receptor tenen claus privades que no comparteixen.

La teoria de la informació de Shannon s'aplica només al secret en les comunicacions. Al seu article [29] Shannon determina les condicions que ha de complir un *sistema de xifrat* per poder ser totalment invulnerable a l'espionatge. A més defineix el concepte de *distància d'unicitat*, que és la mínima quantitat de dades que necessita un adversari per poder descobrir la clau usada en una transmissió, i diu com es calcula.

0 Preliminars

Aquesta secció conté resums de temes de matemàtiques que tenen un paper important en la teoria de la informació i la codificació. La majoria del que es necessita es veu en cursos bàsics de fonaments, matemàtica discreta, àlgebra lineal i probabilitat durant els primers anys d'un grau universitari de l'àmbit de les ciències o l'enginyeria. Alguns aspectes de l'aritmètica de polinomis, en especial les congruències, i la construcció dels cossos finits de nombre d'elements no primer, són segurament els que seran menys familiars.

Pot ser útil per repassar aquests conceptes estudiar i fer els exercicis dels capítols 1, 8 i 9 del text de Brunat-Ventura [3].

0.1 Probabilitat, variables aleatòries i processos

Es consideren *variables aleatòries* discretes que prenen un nombre finit de valors. Els valors de la variable no necessàriament són nombres, sinó elements d'un conjunt finit $\mathcal{X}$ qualsevol. En teoria de la informació el conjunt $\mathcal{X}$ és un *alfabet* i els seus elements s'anomenen *lletres* o *símbols*. Donar una variable aleatòria X vol dir donar

- el conjunt $\mathcal{X}$ dels seus valors;
- una distribució de probabilitat sobre aquest conjunt: per a cada $x \in \mathcal{X}$ un nombre $\Pr(X = x) = p(x) \geqslant 0$ tal que $\sum_{x\in\mathcal{X}} p(x) = 1$.

Tot subconjunt $A \subseteq \mathcal{X}$ és un *esdeveniment*, amb probabilitat

$$p(A) = \Pr(X \in A) = \sum_{x\in A} p(x).$$

Es denotarà $|\mathcal{X}|$ o $\#\mathcal{X}$ el *cardinal* (nombre d'elements) del conjunt $\mathcal{X}$.

Moltes vegades els elements de $\mathcal{X}$ es denoten en forma indexada $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ amb $n = |\mathcal{X}|$. En aquest cas, quan calgui escriure una N-tupla o vector d'elements de $\mathcal{X}$, o sigui, un element de $\mathcal{X}^N$, es denotarà $(x_{i_1}, x_{i_2}, \ldots, x_{i_N})$ per no confondre el subíndex que indica de quin element de $\mathcal{X}$ es tracta amb el que indica la posició dins de la tupla, on els subíndexs $i_1, \ldots, i_N$ són elements de $\{1, \ldots, n\}$.

En la majoria d'aplicacions els valors de la variable no són importants; només ho és el nombre de valors possibles. Si convé es pot suposar que el conjunt $\mathcal{X}$ és el dels n nombres enters $\{1, 2, \ldots, n\}$ o un altre conjunt convenient. En teoria de la informació l'únic que compta és el nombre de lletres, no la forma o l'aspecte que tenen ni el signe que els representa.

Les probabilitats $p(x)$ poden ser iguals a zero. El subconjunt de $\mathcal{X}$ format pels valors que pren la variable amb probabilitat $p(x) \neq 0$ s'anomena *suport* de la variable. De vegades es poden reduir els problemes a considerar variables sense probabilitats zero; o sigui, variables que tenen com a suport el conjunt de tots els seus valors.

Les variables aleatòries discretes amb infinits valors prenen un nombre de valors numerable. La distribució de probabilitat és una successió infinita de nombres $\geqslant 0$ que han de sumar 1, en el sentit de la suma de la sèrie de nombres reals. Tot funciona de manera anàloga al cas finit tenint en compte que les sumes, en el cas numerable, són sumes de sèries. Algunes esperances: mitjana, moments, entropia, etc. poden no estar definides perquè les sèries no siguin convergents.

Exemples. Alguns exemples importants de variables aleatòries són:

- Constant o *degenerada*. Només pren un valor $x \in \mathcal{X}$ amb probabilitat no nul·la: $p(x) = 1$ i $p(y) = 0$ per a tot $y \neq x$.
- *De Bernoulli* o binària. Pren només dos valors. Se sol agafar $\mathcal{X} = \{0, 1\}$. Les probabilitats són $p(1) = p \in [0, 1]$ i $p(0) = q = 1 - p$. Es denota $X \sim \text{Ber}(p)$.
- *Uniforme*. Pren tots els valors amb la mateixa probabilitat. Si $|\mathcal{X}| = n$ ha de ser $p(x) = \frac{1}{n}$ per a tot $x \in \mathcal{X}$. Es denota $X \sim \text{Unif}(n)$.
- *Binomial*. Pren $n+1$ valors $\mathcal{X} = \{0, 1, \ldots, n\}$ amb probabilitats $p(k) = \binom{n}{k} p^k (1-p)^{n-k}$ per a un $p \in [0, 1]$. Es denota $X \sim \text{Binom}(n, p)$.

Funcions de variables aleatòries. Donada una variable aleatòria X, que pren valors en un conjunt $\mathcal{X}$, sigui $g\colon \mathcal{X} \to \mathcal{Y}$ una *funció* definida sobre els elements del conjunt $\mathcal{X}$ amb valors en un conjunt $\mathcal{Y}$.

Usant aquesta funció es pot definir una nova variable aleatòria $Y = g(X)$ que pren valors en $\mathcal{Y}$ agafant la distribució de probabilitat

$$p(y) = \Pr(Y = y) := \sum_{\substack{x \in \mathcal{X} \\ g(x) = y}} p(x),$$

entenent que $p(y)$ és la suma buida, igual a zero, quan $y \in \mathcal{Y}$ no és de la imatge de g. Que això és efectivament una distribució de probabilitat es comprova veient que

$$\sum_{y \in \mathcal{Y}} p(y) = \sum_{y \in \mathcal{Y}} \sum_{\substack{x \in \mathcal{X} \\ g(x) = y}} p(x) = \sum_{x \in \mathcal{X}} p(x) = 1.$$

Esperança. Quan una funció d'una variable aleatòria pren valors en el conjunt dels nombres reals: $\mathcal{Y} \subset \mathbb{R}$, es defineix la seva *esperança* o *valor esperat* com

$$\mathbb{E}\big[g(X)\big] = \sum_{x \in \mathcal{X}} p(x) g(x).$$

És a dir, és la mitjana dels valors de la funció agafats d'acord amb les probabilitats corresponents.

En el cas particular que la variable mateixa X ja prengui valors reals: $\mathcal{X} \subset \mathbb{R}$, s'anomenen *moments* les esperances de les seves potències

$$\mathbb{E}[X^k] = \sum_{x \in \mathcal{X}} p(x) x^k.$$

El primer moment μ s'anomena *mitjana* i el segon moment de la variable centrada $X - \mu$ és la *variància*: el quadrat de la *desviació tipus* σ:

$$\mu = \mu_X = \mathbb{E}[X] = \sum_{x \in \mathcal{X}} p(x) x, \qquad \sigma^2 = \sigma_X^2 = \mathbb{E}\big[(X - \mu)^2\big] \sum_{x \in \mathcal{X}} p(x)(x - \mu)^2.$$

Aquests invariants satisfan les desigualtats bàsiques següents:

- *Desigualtat de Markov*. Sigui X una variable aleatòria que pren valors reals no negatius amb esperança $\mu = \mathbb{E}[X]$. Per a tot real $a > 0$,

$$\Pr(X \geqslant a) \leqslant \frac{\mu}{a}.$$

- *Desigualtat de Chebyshev*. Sigui X una variable aleatòria que pren valors reals amb esperança $\mu = \mathbb{E}[X]$ finita i variància $\sigma^2 \neq 0$ també finita. Per a tot real $a > 0$,

$$\Pr\left(|X - \mu| \geqslant a\sigma\right) \leqslant \frac{1}{a^2}.$$

  Agafant el valor a/σ en comptes de a s'obté la formulació equivalent

$$\Pr\left(|X - \mu| \geqslant a\right) \leqslant \frac{\sigma^2}{a^2}.$$

Distribucions de probabilitat. S'anomena *distribució de probabilitat* (finita) de n valors una n-tupla $\boldsymbol{p} = (p_1, p_2, \ldots, p_n)$ de n nombres reals $p_i \geqslant 0$ amb suma $\sum_{i=1}^n p_i = 1$.

Tota variable aleatòria dona lloc a una distribució de probabilitat posant $p_i = p(x_i)$. Recíprocament, donada una distribució de probabilitat es pot definir una variable aleatòria tal que la distribució de probabilitat corresponent sigui la donada, agafant un conjunt $\mathcal{X}$ de n elements qualsevol.

Aquesta situació en què $\Pr(X = x_i) = p_i$ es denota $X \sim \boldsymbol{p}$, per indicar que la distribució de probabilitat de la variable X és la que correspon al vector $\boldsymbol{p}$.

Les distribucions de probabilitat de n nombres són punts en l'espai afí n-dimensional $\mathbb{R}^n$ i formen el símplex amb vèrtexs els n vectors $\boldsymbol{e}_i$ de la base canònica: els vectors que tenen una coordenada igual a 1 i les demés iguals a zero. Els vèrtexs del símplex són les distribucions de probabilitat que corresponen a les variables constants: les variables amb distribució $X \sim \boldsymbol{e}_i$ que prenen només el valor x_i amb probabilitat 1 i els altres amb probabilitat zero.

Diverses variables. En la descripció i estudi d'un fenomen aleatori apareixen sovint diverses variables aleatòries relacionades entre elles a través d'una *distribució conjunta*. En el cas de dues variables, un *parell de variables aleatòries* es pot pensar com:

- Donades dues variables aleatòries X i Y, que prenen valors en conjunts $\mathcal{X}$ i $\mathcal{Y}$, respectivament, *amb una distribució de probabilitat sobre el conjunt producte*:

$$p(x, y) = \Pr(X = x, Y = y), \qquad \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p(x, y) = 1,$$

  anomenada *distribució de probabilitat conjunta*, de tal manera que les distribucions de probabilitat de cadascuna d'elles, anomenades *distribucions marginals*, són:

$$p(x) = \Pr(X = x) = \sum_{y\in\mathcal{Y}} p(x, y), \qquad p(y) = \Pr(Y = y) = \sum_{x\in\mathcal{X}} p(x, y).$$

- Un *vector aleatori* o variable aleatòria multivariant: una variable $\boldsymbol{X}$ que pren *valors vectorials* en un conjunt $\boldsymbol{\mathcal{X}} = \mathcal{X}_1 \times \mathcal{X}_2$ que és el producte cartesià de dos conjunts $\mathcal{X}_i$. La variable $\boldsymbol{X}$ té dues components X_1 i X_2, que són variables aleatòries que prenen valors en els conjunts $\mathcal{X}_1$ i $\mathcal{X}_2$, respectivament, que són les components dels valors de $\boldsymbol{X}$. De fet les X_i són les funcions de $\boldsymbol{X}$ obtingudes en aplicar a $\boldsymbol{\mathcal{X}}$ les projeccions sobre les components: $\pi_i \colon \boldsymbol{\mathcal{X}} \twoheadrightarrow \mathcal{X}_i$.

  La distribució de probabilitat de la variable vectorial $\boldsymbol{X}$ és la *distribució de probabilitat conjunta* i d'ella es dedueixen les *distribucions marginals* de les dues variables:

$$p_1(x) = \Pr(X_1 = x) = \sum_{y \in X_2} \Pr\big(\boldsymbol{X} = (x, y)\big).$$

Donat un parell de variables aleatòries X i Y es poden definir *variables condicionades*:

- Per a cada $x \in \mathcal{X}$ amb probabilitat $p(x) \neq 0$ es té una variable condicionada $Y|X = x$, que pren valors en el conjunt $\mathcal{Y}$, amb distribució de probabilitats

$$p(y|x) = \Pr(Y = y|X = x) = \frac{p(x, y)}{p(x)}.$$

- De manera anàloga es tenen variables $X|Y = y$ per a cada $y \in \mathcal{Y}$ amb probabilitat no nul·la.
- També es poden definir variables condicionades a esdeveniments: si $A \subseteq \mathcal{X}$ té $p(A) \neq 0$ es defineix la variable $Y|X \in A$ a partir de la distribució de probabilitats:

$$p(y|X \in A) = \Pr(Y = y|X \in A) = \frac{p(A, y)}{p(A)}.$$

Tal com s'han definit les probabilitats condicionades es tenen igualtats

$$p(x, y) = p(x)p(y|x) = p(y)p(x|y). \tag{1}$$

que valen també quan $p(x) = 0$ i quan $p(y) = 0$. Tenint en compte que quan $p(x) = 0$ aleshores necessàriament $p(x, y) = 0$ per a tot $y \in \mathcal{Y}$, i anàlogament si $p(y) = 0$, aquestes igualtats romanen certes en els casos de probabilitat zero donant valors arbitraris a les probabilitats condicionades. Així, des del punt de vista de la relació entre probabilitats conjuntes i marginals, les probabilitats condicionades es poden deixar indefinides o també es poden definir amb els valors que es vulgui sempre que les probabilitats marginals (i, per tant, també les conjuntes) siguin zero.

Es coneix pel nom de *teorema de Bayes* la relació entre les probabilitats condicionades en canviar l'ordre de les variables que es dedueix de la igualtat (1): si $p(x) \neq 0$ aleshores

$$p(y|x) = \frac{p(y)p(x|y)}{p(x)}$$

Donat un parell de variables aleatòries X i Y que prenen valors en els dos conjunts $\mathcal{X} = \{x_1, x_2, \ldots, x_n\}$ i $\mathcal{Y} = \{y_1, y_2, \ldots, y_m\}$, respectivament, les probabilitats conjuntes i marginals es poden organitzar en forma de taula de la manera següent:

(X,Y)	y_1	y_2	$\cdots$	y_m	X
x_1	$p(x_1,y_1)$	$p(x_1,y_2)$	$\cdots$	$p(x_1,y_m)$	$p(x_1)$
x_2	$p(x_2,y_1)$	$p(x_2,y_2)$	$\cdots$	$p(x_2,y_m)$	$p(x_2)$
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	$p(x_n,y_1)$	$p(x_n,y_2)$	$\cdots$	$p(x_n,y_m)$	$p(x_n)$
Y	$p(y_1)$	$p(y_2)$	$\cdots$	$p(y_m)$	1

Les entrades $p(x_i, y_j)$ són les probabilitats conjuntes. En sumar per files s'obtenen les probabilitats marginals $p(x_i)$ de la variable X i en sumar per columnes les de la variable Y. Les nm entrades de la taula han de sumar 1, així com les de la fila i de la columna del final.

Les probabilitats condicionades també es poden posar en forma de taula:

$Y\|X$	y_1	y_2	$\cdots$	y_m	
x_1	$p(y_1\|x_1)$	$p(y_2\|x_1)$	$\cdots$	$p(y_m\|x_1)$	1
x_2	$p(y_1\|x_2)$	$p(y_2\|x_2)$	$\cdots$	$p(y_m\|x_2)$	1
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
x_n	$p(y_1\|x_n)$	$p(y_2\|x_n)$	$\cdots$	$p(y_m\|x_n)$	1

Aquí les files de la taula són les distribucions de probabilitat de les variables condicionades $Y|X = x_i$, i per tant han de sumar 1, almenys quan $p(x_i) \neq 0$, que és quan estan definides.

Una matriu que contingui distribucions de probabilitat a les seves files s'anomena *matriu de probabilitat* o *matriu estocàstica*.

De manera anàloga es poden considerar més de dues variables alhora, o, dit d'una altra manera, vectors aleatoris $\boldsymbol{X} = (X_1, X_2, \ldots, X_k)$ amb més de dues components, que prenen valors en conjunts $\boldsymbol{\mathcal{X}} = \mathcal{X}_1 \times \mathcal{X}_2 \times \cdots \times \mathcal{X}_k$ producte cartesià de diversos conjunts $\mathcal{X}_i$. Aquestes (famílies de) variables venen descrites per una distribució de probabilitat conjunta

$$p(\boldsymbol{x}) = p(x_1, x_2, \ldots, x_k) = \Pr(X_1 = x_1, X_2 = x_2, \ldots, X_k = x_k),$$

per a cada $\boldsymbol{x} = (x_1, x_2, \ldots, x_k) \in \boldsymbol{\mathcal{X}}$ amb suma $\sum_{\boldsymbol{x} \in \boldsymbol{\mathcal{X}}} p(\boldsymbol{x}) = 1$.

La distribució conjunta d'un vector aleatori dona lloc a moltes distribucions marginals: no només les distribucions marginals de cadascuna de les variables components X_i sinó de cada subconjunt format per algunes d'aquestes variables. També es tenen moltes possibilitats per a les probabilitats condicionades: d'una variable respecte d'una altra o d'un subconjunt de variables qualsevol respect d'un altre subconjunt de variables disjunt.

Processos estocàstics. Es consideren *processos estocàstics* discrets de *temps discret*: successions infinites de variables aleatòries discretes $\boldsymbol{X} = (X_j)_{j\geqslant 1} = X_1, X_2, X_3, \ldots$ que prenen valors en un mateix conjunt $\mathcal{X}$. El procés queda caracteritzat per les probabilitats

$$p(x_1, \ldots, x_k) = \Pr(X_1 = x_{i_1}, \ldots, X_k = x_{i_k}), \quad x_{i_j} \in \mathcal{X}, \quad k \geqslant 1,$$

a partir de les quals es poden calcular les probabilitats marginals de les variables de la successió i de qualsevol conjunt finit d'aquestes variables.

Es pot pensar com el model matemàtic d'un experiment que produeix seqüències d'elements del conjunt $\mathcal{X}$ de manera aleatòria, de manera que els elements de la seqüència no tenen perquè ser independents. S'anomena *estat* del procés en temps k al valor $x \in \mathcal{X}$ de la variable X_k. Així, un procés va canviant d'estat en instants de temps discrets.

Per abreujar algunes notacions, i seguint la notació usada sovint a [4], per a cada $k \geqslant 1$ es denotarà X^k el vector aleatori $(X_1, \ldots, X_k)$ format per les primeres k variables i es denotarà $\boldsymbol{x}^k = (x_{i_1}, \ldots, x_{i_k})$ un element del conjunt $\mathcal{X}^k$: una k-tupla d'elements de $\mathcal{X}$.

De vegades es consideren també processos infinits en totes dues direccions, amb les variables indexades en els nombres enters: $\boldsymbol{X} = (X_j)_{j\in\mathbb{Z}} = \ldots, X_{-2}, X_{-1}, X_0, X_1, X_2, \ldots$.

Procés com a variable. Un procés es pot pensar també com una variable aleatòria $\boldsymbol{X}$ definida en un conjunt $\boldsymbol{\mathcal{X}} = \prod_{j=1}^{\infty} \mathcal{X}$ que és el producte cartesià d'infinites còpies d'un mateix conjunt discret $\mathcal{X}$. Els elements d'aquest conjunt són successions $\boldsymbol{x} = (x_i)_{i\geqslant 1}$ d'elements $x_i \in \mathcal{X}$. Per tant els valors que pren la variable $\boldsymbol{X}$ són successions com aquestes.

Ara s'ha de definir bé la probabilitat, perquè aquest conjunt $\mathcal{X}$ és no numerable i la variable no és discreta. S'agafa com a *σ-àlgebra* en el conjunt $\mathcal{X}$ la generada pels *conjunts cilíndrics*: subconjunts $S = \prod_{j=1}^{\infty} S_j \subseteq \mathcal{X}$ producte cartesià de $S_j \subseteq \mathcal{X}$ que són tots iguals al conjunt total: $S_j = \mathcal{X}$, excepte un nombre finit.

Donades variables aleatòries X_j a valors en $\mathcal{X}$ es pot definir una probabilitat en aquesta σ-àlgebra dient que la probabilitat del conjunt cilíndric $\prod S_j$ és el producte de totes les probabilitats $\Pr(X_j \in S_j)$, que està ben definit ja que totes elles són iguals a 1 excepte un nombre finit, gràcies a la condició que els S_j siguin gairebé tots igual a $\mathcal{X}$.

Recíprocament, a partir d'una probabilitat sobre la σ-àlgebra dels conjunts cilíndrics s'obté un procés estocàstic determinat per les probabilitats

$$\Pr(X_1 = x_1, \ldots, X_n = x_n) = \Pr\left(\{x_1\} \times \cdots \times \{x_n\} \times \prod_{k\geqslant n} \mathcal{X}\right)$$

Tipus de processos. Algunes característiques importants que poden tenir els processos estocàstics són les següents:

- Un procés es diu *estacionari* si és invariant respecte dels desplaçaments temporals en el sentit següent: les distribucions de probabilitat dels vectors finits de variables consecutives són independents d'on comenci el vector:

  $$(X_r, X_{r+1}, \ldots, X_{r+k}) \sim (X_s, X_{s+1}, \ldots, X_{s+k}), \quad \forall r, s, \quad \forall k \geqslant 0.$$

  En particular, totes les variables han de tenir mateixa distribució: $X_r \sim X_s \ \forall r, s$.

- Un procés $\boldsymbol{X} = (X_j)_{j\geqslant 0}$ es diu *cadena de Markov* o *procés de Markov* si cada variable depèn només de la immediatament anterior:

  $$\Pr\left(X_{k+1} = x_{i_{k+1}} | X_k = x_{i_k}, X_{k-1} = x_{i_{k-1}}, \ldots, X_1 = x_{i_1}\right) = \Pr\left(X_{k+1} = x_{i_{k+1}} | X_k = x_{i_k}\right)$$

  per a tot $k \geqslant 1$. Una cadena de Markov queda determinada donant la distribució de probabilitat de la primera variable $X_0 \sim \boldsymbol{p} = (p_1, \ldots, p_n)$ i una família de *matrius de transició de probabilitats* $\mathbf{P}_k = \left[p_{ij}^{(k)}\right]$ que continguin les probabilitats condicionades de cada variable respecte de l'anterior: $p_{ij}^{(k)} = \Pr(X_{k+1} = x_j | X_k = x_i)$ per a $k \geqslant 0$. Amb això ja es poden calcular totes les probabilitats de la cadena. Si $X_k \sim \boldsymbol{p}_k$ aleshores $\boldsymbol{p}_{k+1} = \boldsymbol{p}_k \cdot \mathbf{P}_k$ i la distribució de probabilitats de cada variable X_k s'obté a partir de la inicial $\boldsymbol{p} = \boldsymbol{p}_0$ i les matrius $\mathbf{P}_k$ com el producte $X_n \sim \boldsymbol{p} \cdot \prod_{i=1}^{k} \mathbf{P}_i$.

- Una *cadena de Markov* d'ordre $m \geqslant 1$ és un procés en què cada variable depèn només de les m variables anteriors:

  $$\begin{aligned}\Pr\left(X_{k+1} = x_{i_{k+1}} | X_k = x_{i_k}, \ldots, X_1 = x_{i_1}\right) \\ = \Pr\left(X_{k+1} = x_{i_{k+1}} | X_k = x_{i_k}, \ldots, X_{k-m-1} = x_{i_{k-m+1}}\right) \quad \forall k \geqslant m.\end{aligned}$$

- Una cadena de Markov $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ es diu *invariant respecte del temps* si les probabilitats condicionades de cada variable respecte l'anterior són les mateixes:

  $$\Pr\left(X_{r+1} = y | X_r = x\right) = \Pr\left(X_{s+1} = y | X_s = x\right) \quad \forall r, s \geqslant 1.$$

  És a dir, si les matrius de transició de probabilitats són totes iguals: $\mathbf{P}_k = \mathbf{P}$ per a tot $k \geqslant 0$. En aquest cas s'anomena *distribució estacionària* una distribució de probabilitats inicial $X_0 \sim \boldsymbol{p}$ que indueixi la mateixa distribució en totes les altres variables: $X_k \sim \boldsymbol{p}$ per a tot $k \geqslant 0$. Aquesta condició és equivalent a què $\boldsymbol{p}$ sigui un vector propi de valor propi 1 per l'esquerra de la matriu de transició $\mathbf{P}$. El *teorema de Perron-Frobenius* dona informació sobre l'existència i unicitat de distribucions estacionàries.

- Un procés *ergodic* és un procés estocàstic estacionari en què les distribucions de probabilitat es poden deduir a partir d'una única observació.

  Aquesta condició es descriu formalment de la manera següent: sigui $k \geqslant 1$ i sigui $\boldsymbol{x} = (x_{i_1}, \ldots, x_{i_k}) \in \mathcal{X}^k$ un vector de k elements de $\mathcal{X}$. Per a cada $m \geqslant k$ es denota $N_{\boldsymbol{x}}(X_1, \ldots, X_m)$ la variable aleatòria que compta el nombre de vegades que el vector $\boldsymbol{x}$ apareix com una seqüència de valors consecutius dins del vector de m elements que pren com a valor la variable vectorial $\boldsymbol{X}^m = (X_1, \ldots, X_m)$. El procés és ergodic quan

  $$\lim_{m\to\infty} \frac{1}{m} N_{\boldsymbol{x}}(X_1, \ldots, X_m) = p(\boldsymbol{x}) = \Pr(X_1 = x_{i_1}, \ldots, X_k = x_{i_k}), \qquad \forall k \geqslant 1, \quad \forall \boldsymbol{x} \in \mathcal{X}^k.$$

L'exemple més senzill, però molt important, de procés estocàstic és el

Exemple 0.1 (Procés i.i.d.)**.** *Un procés estocàstic $\boldsymbol{X} = X_1, X_2, \ldots$ és independent i idènticament distribuït (abreujat i.i.d.) si les variables són independents les unes de les altres i tenen totes la mateixa distribució: $X_i \sim X_j$ per a tot i, j.*

Donada una variable aleatòria X qualsevol es pot construir un procés i.i.d. simplement agafant una successió de variables independents totes amb la distribució de X.

Convergència de variables aleatòries. Recordi's també que donada una successió de variables aleatòries $(X_n)_{n\geqslant 1}$, o sigui un procés estocàstic en temps discret, i una variable X, on totes les variables involucrades prenen valors en els nombres reals, es diu que la successió X_n tendeix cap a X

- *en probabilitat* si per a tot $\epsilon > 0$ es té

$$\lim_{n\to\infty} \Pr\left(|X_n - X| \geqslant \epsilon\right) = 0;$$

- *gairebé segur* o *amb probabilitat* 1 si es té

$$\Pr\left(\lim_{n\to\infty} X_n = X\right) = 1;$$

- *en mitjana* si

$$\lim_{n\to\infty} \mathbb{E}[X_n - X] = 0;$$

- *en mitjana quadràtica* $(k = 2)$ o d'ordre $k \geqslant 1$ si

$$\lim_{n\to\infty} \mathbb{E}\left[|X_n - X|^k\right] = 0;$$

- *en distribució* si les funcions de distribució de les X_n tendeixen a la de X.

Llei dels grans nombres. Les *lleis dels grans nombres*: donen límits de la successió de les mitjanes aritmètiques de variables aleatòries i.i.d. amb valors reals.

Sigui $\boldsymbol{X} = X_1, X_2, X_3, \ldots$ una successió de variables i.i.d. que prenen valors a $\mathbb{R}$. Siguin $\mathbb{E}[X_i] = \mu$ la seva esperança i $\mathrm{Var}(X_i) = \sigma^2$ la seva variància. Es defineixen les variables $\bar{X}_n = \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$ com les mitjanes aritmètiques de les n primeres variables X_i. Aquestes variables tenen la mateixa esperança que les X_n: $\mathbb{E}\left[\bar{X}_n\right] = \mu$ i, en canvi, la seva variància és $\mathrm{Var}\left(\bar{X}_n\right) = \frac{\sigma^2}{n}$. Aleshores,

- *Llei dèbil dels grans nombres.* La successió $\bar{X}_n$ tendeix cap a (la variable constant) μ en probabilitat: per a tot $\epsilon > 0$ es té

$$\lim_{n\to\infty} \bar{X}_n = \mu.$$

  O sigui, per a tot $\delta > 0$ existeix un N tal que

$$n \geqslant N \quad \Rightarrow \quad \Pr\left(|\bar{X}_n - \mu| \geqslant \epsilon\right) < \delta.$$

  La demostració es fa aplicant la desigualtat de Chebyshev, de la qual es dedueix que

$$\Pr\left(|\bar{X}_n - \mu| \geqslant \epsilon\right) \leqslant \frac{\sigma^2}{n\epsilon^2},$$

  i això tendeix a zero quan n tendeix a infinit.

- *Llei forta dels grans nombres.* La successió $\bar{X}_n$ tendeix cap a μ gairebé segur:

$$\Pr\left(\lim_{n\to\infty} \bar{X}_n = \mu\right) = 1.$$

La llei dels grans nombres val també, només amb la condició que les variables siguin independents i tinguin totes la mateixa esperança i variància, encara que no tinguin la mateixa distribució.

Un altre resultat important per a aquest tipus de successions de variables aleatòries i.i.d. és el *teorema del límit central*: si es consideren les variables normalitzades $Y_n = \frac{X_n - \mu}{\sigma}$ aleshores les variables $\frac{1}{\sqrt{n}}(Y_1 + \cdots + Y_n)$ convergeixen *en distribució* cap a una variable aleatòria amb distribució normal estàndard.

Problemes

0.1. *Generació d'unes variables a partir d'unes altres.* Digueu com generar una variable amb distribució $X \sim \mathrm{Ber}(p)$ tirant una moneda. Calculeu el nombre esperat de llançaments necessari per generar un valor de X.

Més endavant, en parlar de la codificació aritmètica, es veurà com generalitzar això construint una variable aleatòria a partir d'una altra.

0.2 Aritmètica de nombres enters i congruències

Els *nombres enters* són els nombres

$$\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, \ldots\}.$$

Al conjunt dels nombres enters hi ha definides dues operacions: la *suma* i el *producte*, que li donen estructura d'*anell commutatiu*.

Divisibilitat. Donats nombres enters a i b amb $b \neq 0$, es diu que b *divideix* a si existeix un enter q tal que $a = bq$. En aquest cas es diu que b es un *divisor* de a i que a es un *múltiple* de b. L'enter q es diu *quocient* de la divisió de a per b.

Per exemple, els divisors positius de l'enter 40 són $1, 2, 4, 5, 8, 10, 20$ i 40, i els divisors positius de 37 són només 1 i 37. Tot enter positiu n té almenys dos divisors positius: 1 i ell mateix.

Nombres primers. Un enter $p > 1$ és *primer* si els seus únics divisors positius són 1 i p. Els enters $n > 1$ que no són primers s'anomenen *compostos*.

Els enters compostos es caracteritzen per tenir una *descomposició* $n = ab$ com a producte de dos enters positius a i b tots dos més grans que 1. Per exemple, $n = 40$ és compost ja que admet descomposicions $40 = 2 \cdot 20 = 4 \cdot 10 = 5 \cdot 8$ i l'enter $n = 37$ és primer ja que l'única descomposició com a producte d'enters positius és $37 = 1 \cdot 37$.

Dels enters menors que 30,

- $2, 3, 5, 7, 11, 13, 17, 19, 23$ i 29 són primers i
- $4, 6, 8, 9, 10, 12, 14, 15, 16, 18, 20, 21, 22, 24, 25, 26, 27$ i 28 són compostos.

De nombres primers n'hi ha infinits.

Descomposició única. Tot enter positiu *descompon de manera única* en producte de primers. En general, la descomposició en primers d'un enter positiu és de la forma

$$n = p_1^{m_1} \cdot p_2^{m_2} \cdots p_r^{m_r} = \prod_{i=1}^{r} p_i^{m_i}$$

on els p_i són els nombres primers diferents que divideixen n i els exponents $m_i \geqslant 1$ indiquen la màxima potència de cada primer que divideix n. Per exemple, l'enter $n = 9775822320$ té descomposició $n = 2^4 \cdot 3^2 \cdot 5 \cdot 11^3 \cdot 101^2$.

Màxim comú divisor. El *màxim comú divisor* dels enters a i b (no tots dos zero) és l'enter positiu més gran que els divideix tots dos. Es caracteritza també com l'únic enter positiu que divideix tots els divisors comuns de a i b. Es defineix de manera anàloga el màxim comú divisor d'enters $a_1, a_2, \ldots, a_r$. El màxim comú divisor dels enters a i b acostuma a denotar-se (a, b) o també $\gcd(a, b)$ si pot haver-hi confusió amb el significat del parèntesi.

Els enters a i b es diuen *coprimers* (o *relativament primers* o *primers entre ells*) si tenen màxim comú divisor igual a 1. Per exemple,

$$\gcd(54, 66) = 6, \qquad \gcd(21, 50) = 1.$$

Per tant els enters 54 i 66 no són relativament primers i els enters 21 i 50 sí que ho són.

Divisió entera o euclidiana. Donats enters $a \in \mathbb{Z}$ i $b > 0$ existeixen dos enters q i r tals que

$$a = bq + r \qquad \text{i} \qquad 0 \leqslant r < b.$$

Aquesta expressió s'anomena *divisió euclidiana* (o *divisió amb reste* o *divisió entera*) de a per b i els enters q i r, que són únics amb aquesta propietat, són el *quocient* i el *reste* de la divisió. L'enter a es divideix per b si, i només si, el reste de la divisió euclidiana és $r = 0$.

La divisió euclidiana d'un enter per un altre es pot fer amb l'algorisme de divisió que s'aprèn a l'escola.

Algorisme d'Euclides. Donats enters $a \in \mathbb{Z}$ i $b > 0$ l'algorisme d'Euclides consisteix en

- inicialitzar $r_0 = |a|$, $r_1 = b$ i $k = 1$, i
- per a tot $k \geqslant 1$, i mentre $r_k \neq 0$, fer la divisió euclidiana de r_{k-1} per r_k obtenint així dos enters q_k i r_{k+1} determinats per la identitat:

$$r_{k-1} = r_k q_k + r_{k+1}, \qquad 0 \leqslant r_{k+1} < r_k.$$

Com que a partir de r_1 els nombres r_k decreixen estrictament i són sempre $\geqslant 0$, arribarà un moment en que un sigui zero, i aquí l'algorisme acaba.

La utilitat principal de l'algorisme d'Euclides és la de calcular el màxim comú divisor de dos enters: el seu màxim comú divisor és l'últim reste no nul de l'algorisme d'Euclides.

Identitat de Bézout. Donats enters a i b i $d = \gcd(a, b)$ el seu màxim comú divisor s'anomena *identitat de Bézout* una expressió de la forma

$$ax + by = d, \qquad x, y \in \mathbb{Z}.$$

La identitat de Bézout sempre té solucions $x, y \in \mathbb{Z}$. De fet en té infinites que es poden obtenir a partir d'una solució (x, y) qualsevol com $(x - k\frac{b}{d}, y + k\frac{a}{d})$ amb paràmetre $k \in \mathbb{Z}$.

L'*Algorisme d'Euclides estès* permet calcular una solució de la identitat de Bézout usant els quocients successius q_k que van apareixent en fer l'algorisme d'Euclides de la manera següent: es calculen parells d'enters (x_k, y_k) recursivament amb les fórmules:

$$\begin{aligned} x_0 &= 1, \qquad x_1 = 0, \qquad x_{k+1} = x_{k-1} - q_k x_k, \\ y_0 &= 0, \qquad y_1 = 1, \qquad y_{k+1} = y_{k-1} - q_k y_k. \end{aligned}$$

Aquests nombres satisfan les identitats $ax_k + by_k = r_k$ per a tot $k \geqslant 0$ i per tant, com que l'últim reste no nul, diguem-li r_n, és el màxim comú divisor de a i b, l'últim parell (x_n, y_n) és una solució de la identitat de Bézout:

$$ax_n + by_n = r_n = d.$$

Exemple. A la taula següent hi ha les dades corresponents a l'algorisme d'Euclides estès per als nombres enters $a = 232123452$ i $b = 111217921$.

k	r_k	q_k	x_k	y_k
0	232123452		1	0
1	111217921	2	0	1
2	9687610	11	1	−2
3	4654211	2	−11	23
4	379188	12	23	−48
5	103955	3	−287	599
6	67323	1	884	−1845
7	36632	1	−1171	2444
8	30691	1	2055	−4289
9	5941	5	−3226	6733
10	986	6	18185	−37954
11	25	39	−112336	234457
12	11	2	4399289	−9181777
13	3	3	−8910914	18598011
14	2	1	31132031	−64975810
15	1	2	−40042945	83573821
16	0			

El màxim comú divisor d'aquests dos enters és 1 i una solució de la identitat de Bézout és

$$(232123452)\cdot(-40042945)+(111217921)\cdot(83573821)=1.$$

Congruències. Sigui n un enter positiu, que s'anomenarà *mòdul*. Es diu que dos enters $a,b\in\mathbb{Z}$ són *congruents mòdul n* si

$$n \quad \text{divideix la diferència} \quad b-a$$

i es denota $a\equiv b \pmod n$. És equivalent a dir que es pot passar d'un dels enters a l'altre sumant o restant diverses vegades el mòdul n:

$$\text{existeix un enter} \quad t\in\mathbb{Z} \quad \text{tal que} \quad b=a+nt.$$

Classes de restes o classes de congruència. La relació *ser congruents mòdul n* classifica els nombres en *classes de congruència*. Per exemple, les congruències mòdul 2 classifiquen els nombres en dues classes: la dels nombres parells i la dels nombres senars. Les congruències mòdul 10 els classifiquen en 10 classes diferents; la classe d'un enter positiu queda determinada per l'última xifra decimal. Les congruències mòdul 2^k permeten distingir entre 2^k classes de nombres enters; quan s'agafen les cadenes binaries de k bits per escriure aquests enters, la notació posicional ordinaria en base 2 els identifica amb el representant de l'interval $[0,2^{n-1})$; i la notació en complement a 2 amb el representant de l'interval $[-2^{n-1},2^{n-1}-1)$.

Cada classe de congruència conté un únic enter a de l'interval $0\leqslant a\leqslant n-1$, que se sol agafar com a *representant canònic* de la classe. De vegades es fa servir la notació $[a]_n$ (o també $[a]$ o $\bar a$ si el mòdul n se sobreentén) per indicar la classe de congruència d'un enter a, o també el representant canònic d'aquesta classe. Si $[a]_n$ denota el representant de la classe, aleshores Cada enter $a\in\mathbb{Z}$ representa la classe $[a]_n\in\mathbb{Z}_n$, formada per tots els enters que donen un mateix reste en fer la divisió euclidiana per n, un dels quals és el mateix a.

Es denota $\mathbb{Z}_n$ (també és habitual la notació $\mathbb{Z}/n\mathbb{Z}$) el conjunt de *classes de restes* o *classes de congruència* mòdul n. Se li diu també, simplement, conjunt dels *enters mòdul n*. A la pràctica se sol identificar amb els seus representants canònics, els nombres enters entre 0 i $n-1$, i es posa:

$$\mathbb{Z}_n\approx\{0,1,2,\ldots,n-2,n-1\}.$$

Aritmètica de congruències: reducció. El conjunt $\mathbb{Z}_n$ té estructura natural d'anell commutatiu, amb les operacions de sumar i multiplicar definides a partir de representants de les classes posant:

$$[a]_n+[b]_n:=[a+b]_n,\qquad [a]_n\cdot[b]_n:=[ab]_n.$$

A la pràctica les operacions es fan a partir dels representants canònics $0\leqslant a\leqslant n-1$, reduint el resultat mòdul n després de cada operació de sumar o multiplicar. Per exemple, $[19]_{25}[11]_{25}$ es calcula fent el producte $19\cdot11=209$ i fent la divisió euclidiana $209=25\cdot8+9$, de manera que el resultat és $[209]_{25}=[9]_{25}$.

El 0 és el neutre de la suma i la classe de $-a$ és l'invers de la classe de a. L'1 és el neutre del producte però no totes les classes diferents de zero tenen invers.

Per exemple, si $n = 39$, l'invers per la suma de 5 és 34 i l'invers de 17 és 23; l'invers pel producte de 11 és 32 ja que $11 \cdot 32 = 352 = 39 \cdot 9 + 1 \equiv 1 \pmod{39}$, l'invers de 4 és 10 ja que $4 \cdot 10 = 40 = 39 \cdot 1 + 1 \equiv 1 \pmod{39}$ però en canvi 12 i 26 no tenen invers mòdul 39.

Els llenguatges de programació acostumen a tenir implementada aquesta reducció. Per exemple a `Python` l'expressió `a%n` retorna el representant canònic de $[\texttt{a}]_{\texttt{n}}$: el reste de la divisió euclidiana de `a` per `n` (ull amb els convenis respecte del signe, però).

Elements invertibles. Un enter a és *invertible mòdul* n quan la congruència

$$aX \equiv 1 \pmod{n}$$

té solució. És el mateix que dir que la classe $[a]_n \in \mathbb{Z}_n$ té invers pel producte. En aquest cas, l'*invers de* a *mòdul* n és un enter b tal que $ab \equiv 1 \pmod{n}$ i es denota $b \equiv a^{-1} \pmod{n}$. Per exemple, $32 \equiv 11^{-1} \pmod{39}$ i 12 no és invertible mòdul 39.

Els enters invertibles mòdul n es caracteritzen com aquells que són relativament primers amb el mòdul n. És a dir, un enter $a \in \mathbb{Z}$ és invertible mòdul n si, i només si, $(a, n) = 1$. Si a és un enter relativament primer amb n, l'invers de a mòdul n es pot calcular fàcilment com la classe mòdul n d'un enter x que satisfaci la identitat de Bézout $ax + ny = 1$.

El grup multiplicatiu. El subconjunt de $\mathbb{Z}_n$ format pels elements invertibles mòdul n es diu *grup multiplicatiu mòdul n*. Es denota $\mathbb{Z}_n^*$ o també $(\mathbb{Z}/n\mathbb{Z})^*$. Es pot identificar amb el conjunt dels enters entre 0 i $n-1$ que són relativament primers amb n. Per exemple,

$$\mathbb{Z}_7^* \approx \{1, 2, 3, 4, 5, 6\}, \qquad \mathbb{Z}_{12}^* \approx \{1, 5, 7, 11\}.$$

En el cas que $n = p$ és primer el grup multiplicatiu $\mathbb{Z}_p^*$ conté tots els enters $\{1, 2, \ldots, p-1\}$: totes les classes excepte la classe del zero tenen invers pel producte. Un anell que té aquesta propietat que tots els elements no nuls tenen invers pel producte es diu un *cos*. S'acostuma a usar la notació $\mathbb{F}_p$ o també $\mathrm{GF}(p)$ per indicar el cos finit $\mathbb{Z}_p$ de p elements.

Indicador d'Euler. L'indicador d'Euler d'un enter positiu n és el nombre d'enters entre 0 i $n-1$ que són relativament primers amb n; o sigui, és el nombre d'elements del grup multiplicatiu $\mathbb{Z}_n^*$. Es denota $\varphi(n)$.

L'indicador d'Euler es pot calcular a partir de la descomposició en primers $n = \prod_{i=1}^{r} p_i^{m_i}$ amb la fórmula següent

$$\varphi(n) = \prod_{i=1}^{r} p_i^{m_i - 1}(p_i - 1) = n \prod_{i=1}^{r} \left(1 - \frac{1}{p_i}\right).$$

En particular es tenen els dos casos importants següents:

$$\varphi(p) = p - 1, \quad \text{si } p \text{ és un nombre primer,}$$
$$\varphi(N) = (p-1)(q-1), \quad \text{si } N = pq \text{ és un producte de dos primers diferents.}$$

El primer correspon als cossos finits i el segon als mòduls usats en el sistema criptogràfic de clau pública *RSA*.

Teorema d'Euler. Si a i n són enters relativament primers i $\varphi(n)$ és l'indicador d'Euler, *aleshores*

$$a^{\varphi(n)} \equiv 1 \pmod{n}.$$

Com a conseqüència del Teorema d'Euler, el valor d'una expressió

$$a^k \pmod{n}$$

depèn només del valor de a mòdul n i del valor de k mòdul $\varphi(n)$. O sigui,

$$(a+nu)^{k+\varphi(n)v} \equiv a^k \pmod{n}$$

per a enters $u, v \in \mathbb{Z}$ qualssevol.

Teorema de Fermat. Com a cas particular del teorema d'Euler, corresponent al mòdul primer, es té el (petit) *Teorema de Fermat*: Si p és un nombre primer, per tot enter a no divisible per p es compleix

$$a^{p-1} \equiv 1 \pmod{p},$$

o, en una versió més general, per a tot enter $a \in \mathbb{Z}$ es compleix

$$a^p \equiv a \pmod{p}.$$

0.3 Polinomis

Sigui $\mathbb{K}$ un *cos*. Per exemple els nombres racionals $\mathbb{Q}$, reals $\mathbb{R}$, complexos $\mathbb{C}$, o les classes de congruència $\mathbb{Z}_p$ amb mòdul primer p són cossos. L'*anell de polinomis* amb coeficients en $\mathbb{K}$, en la *variable* X, és el conjunt

$$\mathbb{K}[X] = \left\{ f(X) = a_0 + a_1X + a_2X^2 + \cdots + a_nX^n = \sum_{i=0}^{n} a_iX^i : a_i \in \mathbb{K} \right\}$$

amb les operacions suma i producte definides de la manera natural. Els a_i són els *coeficients* del polinomi; si convé considerar coeficients per a índexs $i > n$ s'agafa $a_i = 0$. Els polinomis que només tenen el coeficient a_0 s'anomenen *constants* (o *escalars*). Així, el cos $\mathbb{K}$ es considera contingut dins de l'anell de polinomis com el subconjunt dels polinomis constants.

Des del punt de vista de l'aritmètica l'anell de polinomis s'assembla molt a l'anell dels nombres enters, amb conceptes i propietats anàlogues, que són compartides per tots els anomenats *anells euclidians*: divisibilitat, *divisió euclidiana* (amb reste), elements *primers*, *descomposició única* en primers, *màxim comú divisor*, algorisme d'Euclides, identitat de Bézout, algorisme d'Euclides estès, congruències, anells de classes de restes, etc.

En aplicacions a les tecnologies digitals es fan servir sobretot polinomis amb coeficients binaris: polinomis sobre el cos $\mathbb{K} = \mathbb{Z}_2 = \{0, 1\}$ de dos elements. En aquest cos la suma i el producte són les dels enters mòdul 2 i corresponen a les operacions booleanes XOR i AND. Com que els coeficients només poden ser zeros o uns els polinomis s'escriuen com a suma de potències X^i de la variable, posant-hi les potències que tenen coeficient 1 i sense posar-hi les

que tenen coeficient 0. Per exemple el polinomi $X + X^3 + X^7 + X^9 \in \mathbb{Z}_2[X]$ és el que té coeficients $a_1 = a_3 = a_7 = a_9 = 1$ i $a_0 = a_2 = a_4 = a_5 = a_6 = a_8 = 0$ (i s'entén que tots els altres coeficients d'índex $\geqslant 10$ també són zeros).

Els polinomis de $\mathbb{Z}_2[X]$ poden identificar-se amb seqüències de dígits binaris:

$$a_0 + a_1X + a_2X^2 + \cdots + a_{n-1}X^{n-1} + a_nX^n \in \mathbb{Z}_2[X] \quad \approx \quad a_0a_1a_2\cdots a_{n-1}a_n \in \{0,1\}^{n+1}.$$

D'aquesta manera les cadenes binàries es poden considerar com a seqüències no de bits sinó de blocs formats per uns quants bits (per exemple d'octets: els bytes) els quals es poden operar entre ells amb les operacions definides per l'estructura de cos.

Grau. Tot polinomi no nul es pot escriure com $f(X) = \sum_{i=0}^{n} a_iX^i$ amb coeficient $a_n \neq 0$ el coeficient d'índex $n \geqslant 0$ més gran que sigui diferent de zero. Aquest coeficient a_n es diu *coeficient líder* del polinomi i l'exponent n és el *grau* del polinomi, que es denota $\deg f$. Per exemple, els polinomis de grau zero són les constants no nul·es. El grau dels polinomis es comporta de la manera següent respecte el producte i la suma:

- $\deg(fg) = \deg f + \deg g$,
- $\deg(f + g) \leqslant \max\{\deg f, \deg g\}$, amb igualtat si $\deg f \neq \deg g$.

El grau del polinomi 0 es defineix per conveni com $\deg 0 = -\infty$, de manera que per a aquest polinomi també valguin les fórmules anteriors.

Polinomis de grau $< n$. Es denota $\mathbb{K}[X]_n$ el conjunt dels polinomis de grau $< n$. Amb la suma i el producte per escalars és un $\mathbb{K}$ espai vectorial de dimensió n, on se sol agafar com a base canònica els monomis $1, X, X^2, \ldots, X^{n-1}$.

Identificant cada polinomi amb el vector dels seus coeficients es té una identificació (isomorfisme canònic) entre $\mathbb{K}[X]_n$ i l'espai vectorial $\mathbb{K}$:

$$\mathbb{K}[X]_n \approx \mathbb{K}^n \qquad \text{posant} \quad f(X) = a_0 + a_1X + \cdots + a_{n-1}X^{n-1} \;\approx\; (a_0, a_1, \ldots, a_{n-1}).$$

Els monomis de la base canònica de l'espai $\mathbb{K}[X]_n$ corresponen als vectors $\boldsymbol{e}_i$ de la base canònica de l'espai $\mathbb{K}^n$.

Polinomis mònics. Un polinomi no nul es diu *mònic* si el seu coeficient líder és 1. Aquests polinomis fan el mateix a l'anell de polinomis que els enters positius fan a l'anell $\mathbb{Z}$ dels enters, en el sentit següent: els nombres enters que tenen invers pel producte són només ± 1 i tot enter no nul es pot escriure de manera única com un d'aquests dos enters per un de positiu; anàlogament els polinomis que tenen invers pel producte són les constants no nul·les (els polinomis de grau zero) i tot polinomi no nul es pot escriure de manera única com una constant no nul·la per un polinomi mònic.

Polinomis primers. Un polinomi mònic no constant és primer si els dos únics polinomis mònics que el divideixen són 1 i ell mateix. Altrament es diu *compost*: els polinomis compostos són els polinomis que es poden posar com un producte $f(X) = u(X)v(X)$ amb u i v polinomis de grau $\geqslant 1$. Els polinomis primers es diuen també *irreductibles*.

De polinomis primers a l'anell $\mathbb{K}[X]$ n'hi ha sempre infinits, sigui quin sigui el cos $\mathbb{K}$. Això es pot demostrar amb el mateix *argument* d'Euclides per als enters $\mathbb{Z}$: multiplicant nombres (polinomis) primers i sumant 1 al resultat s'obté un nombre (polinomi) no divisible per cap dels primers que s'han multiplicat.

Els polinomis mònics de grau 1, de la forma $X + a$ per a $a \in \mathbb{K}$, sempre són primers. La descripció general dels polinomis primers de l'anell $\mathbb{K}[X]$ depèn del cos de coeficients. Per exemple (ull! demostrar aquestes afirmacions no sempre és fàcil):

- A $\mathbb{Q}[X]$ hi ha polinomis primers de tots els graus. Per exemple, si p és un enter primer el polinomi $X^n + p$ és primer per a tot grau $n \geqslant 1$.
- A $\mathbb{R}[X]$ els polinomis primers són els mònics de grau 1 i els mònics de grau 2 amb discriminant negatiu. No hi ha polinomis primers de grau $\geqslant 3$.
- A $\mathbb{C}[X]$ els únics polinomis primers són els mònics de grau 1. Això equival al teorema fonamental de l'àlgebra: tot polinomi no constant de $\mathbb{C}[X]$ té alguna arrel. Els cossos amb aquesta propietat es diuen *algebraicament tancats*.
- A $\mathbb{Z}_p[X]$ hi ha polinomis primers de tots els graus. No hi ha fórmules o expressions generals que proporcionin sempre polinomis primers de qualsevol graus. A la pràctica, tal com es fa amb els nombres enters, per saber si un polinomi és primer s'ha de comprovar que no es divideixi per polinomis de grau més petit (n'hi ha només un nombre finit), tot i que, igualment com a $\mathbb{Z}$, hi ha *tests de primalitat* i algorismes probabilístics que simplifiquen molt els càlculs.
- A $\mathbb{Z}_2[X]$ els polinomis primers són:
  - de grau 1 n'hi ha dos: X i $1 + X$;
  - de grau 2 n'hi ha només un: només $1 + X + X^2$;
  - de grau 3 n'hi ha dos: $1 + X + X^3$ i $1 + X^2 + X^3$;
  - de grau 4 n'hi ha tres: $1 + X + X^4$, $1 + X^3 + X^4$ i $1 + X + X^2 + X^3 + X^4$; ...

Descomposició única. Tot polinomi mònic $f(X)$ descompon, de manera única, com un producte de polinomis primers:

$$f(X) = P_1(X)^{m_1} \cdot P_2(X)^{m_2} \cdots P_r(X)^{m_r} = \prod_{i=1}^{r} P_i(X)^{m_i}.$$

Quan el polinomi no és mònic té una descomposició anàloga on, a més, hi apareix com a factor el seu coeficient líder.

Divisió euclidiana. Donats polinomis f i g amb g no nul, existeixen polinomis q i r únics tals que

$$f(X) = g(X)q(X) + r(X), \qquad -\infty \leqslant \deg r < \deg g,$$

que s'anomenen quocient i reste de la *divisió euclidiana* de f per g. El polinomi g divideix f quan el reste sigui zero (corresponent al cas en què $\deg r = -\infty$).

La divisió euclidiana de polinomis es pot fer amb un algorisme anàleg al de la divisió euclidiana de nombres enters, on els coeficients dels polinomis fan el mateix paper que els dígits dels nombres. De fet, en aquest cas l'algorisme és encara més senzill, ja que no hi ha problemes de ròssec (*carry*).

El *màxim comú divisor*, l'algorisme d'Euclides, la identitat de Bézout i l'algorisme d'Euclides estès funcionen exactament igual amb polinomis que amb nombres enters, i les fórmules i expressions corresponents són les mateixes.

Congruències. Tot funciona exactament igual que amb nombres enters. S'agafa un polinomi mònic $N(X)$ de grau $n = \deg N$: el *mòdul*. Dos polinomis $f(X), g(X) \in K[X]$ són *congruents mòdul* N si

$$N(X) \quad \text{divideix la diferència} \quad f(X) - g(X).$$

Equivalentment, es pot passar d'un polinomi a l'altre sumant-li un múltiple del mòdul: $g(X) = f(X) + N(X)q(X)$ per a algun polinomi $q(X) \in \mathbb{K}[X]$.

La relació de congruència classifica els polinomis de $\mathbb{K}[X]$ en classes de congruència. En aquest cas es poden agafar com a representants canònics els polinomis de grau menor que el grau del mòdul $n = \deg N(X)$: tota classe de congruència $[f(X)]_{N(X)}$ conté un únic polinomi de grau $< n$: el reste de la divisió euclidiana de qualsevol polinomi que pertanyi a la classe per $N(X)$.

Es denota $\mathbb{K}[X]_{N(X)}$, o també $\mathbb{K}[X]/N(X)\mathbb{K}[X]$ o $\mathbb{K}[X]/\langle N(X)\rangle$, l'anell de classes de congruència, o de classes de restes, format per les classes de congruència de tots els polinomis de $\mathbb{K}[X]$. Agafant representants canònics, de la mateixa manera com $\mathbb{Z}_n$ se sol identificar amb el conjunt dels enters entre 0 i $n-1$, en aquest cas l'anell de classes de restes es pot identificar amb els polinomis de grau $< n$:

$$\mathbb{K}[X]_{N(X)} \approx \{a_0 + a_1X + \cdots + a_{n-1}X^{n-1} : a_i \in \mathbb{K}\}.$$

Per fer aritmètica amb congruències mòdul $N(X)$ es poden sumar els representants canònics directament, i el resultat ja és el representant canònic de la suma, que també té grau $< n$. En canvi el producte de representants tindrà en general grau $\geqslant n$ i per tant es requereix un procés de reducció que rebaixi aquest grau: s'ha de fer la divisió euclidiana del polinomi producte pel mòdul $N(X)$ i canviar-lo pel reste d'aquesta divisió.

Anàlogament al cas de les congruències amb nombres enters, una classe $[f(X)]_{N(X)}$ té invers pel producte si, i només si, $\gcd\big(f(X), N(X)\big) = 1$. L'invers es pot calcular com la classe del polinomi $f(X)$ en una solució de la identitat de Bézout $f(X)g(X)+N(X)h(X) = 1$: en aquest cas l'invers de $[f(X)]_{N(X)}$ és $[g(X)]_{N(X)}$. En particular es té el:

Teorema. *L'anell $\mathbb{K}[X]_{N(X)}$ de classes de congruència és un cos si, i només si, el mòdul $N(X)$ és un polinomi primer.*

Arrels. Tot polinomi $f(X) \in \mathbb{K}[X]$ indueix una aplicació d'*avaluació* $f\colon \mathbb{K} \to \mathbb{K}$ que envia cada element $\alpha \in \mathbb{K}$ a l'element $f(\alpha) \in \mathbb{K}$ que s'obté en substituir la variable X per α i fer les operacions corresponents.

Els $\alpha \in \mathbb{K}$ tals que $f(\alpha) = 0$ són les *arrels* (o *zeros*) del polinomi f. La *regla de Ruffini* assegura que α és arrel del polinomi f si, i només si, el polinomi $X - \alpha$ divideix $f(X)$ a $\mathbb{K}[X]$. En aquest cas, si m és l'exponent més gran tal que $(X-\alpha)^m$ divideix el polinomi $f(X)$ es diu que α és una arrel *de multiplicitat* m.

La descomposició única en primers a l'anell $\mathbb{K}[X]$ implica que un polinomi de grau n pot tenir, com a màxim, n arrels, comptant-les amb les seves multiplicitats.

Interpolació. Un polinomi $f(X) \in \mathbb{K}[X]_n$ de grau $< n$ queda unívocament determinat donant el seu valor en n elements diferents del cos: donats n elements diferents $\alpha_1, \ldots, \alpha_n$ de $\mathbb{K}$ i uns altres n elements qualssevol $\beta_1, \ldots, \beta_n$ existeix un únic polinomi $f(X) \in \mathbb{K}[X]$ tal que $f(\alpha_i) = \beta_i$ per a tot $i = 1, \ldots, n$.

Aquest polinomi s'anomena *polinomi interpolador* i hi ha diverses maneres de calcular-lo; en particular es pot fer usant *polinomis de Lagrange* o *matrius de Vandermonde*.

Elements algebraics i polinomi mínim. Es considera una *extensió de cossos*: dos cossos tals que un està contingut en l'altre $\mathbb{K} \subseteq \mathbb{E}$. Per exemple, $\mathbb{Q} \subset \mathbb{R}$ o $\mathbb{R} \subset \mathbb{C}$ o també en el cas de cossos finits (veure secció 0.4) es tenen les extensions $\mathbb{F}_q \subset \mathbb{F}_{q^e}$.

Els elements del cos gran $\alpha \in \mathbb{E}$ es poden avaluar en polinomis amb coeficients en el cos petit $f(X) \in \mathbb{K}[X]$. Un element $\alpha \in \mathbb{E}$ es diu *algebraic* sobre $\mathbb{K}$ si és arrel d'algun polinomi no nul de $\mathbb{K}[X]$.

Per exemple, $\sqrt{2}, \sqrt[3]{2} \in \mathbb{R}$ i $e^{2\pi i/n} \in \mathbb{C}$ són algebraics sobre $\mathbb{Q}$ ja que són arrels dels polinomis $X^2 - 2$, $X^3 - 2$ i $X^n - 1 \in \mathbb{Q}[X]$, respectivament. En canvi (això és difícil de demostrar) els nombres reals π i e no són algebraics sobre $\mathbb{Q}$.

En el cas d'una extensió de cossos finits tots els elements de $\mathbb{F}_{q^e}$ són algebraics sobre $\mathbb{F}_q$.

S'anomena *polinomi mínim* o polinomi irreductible d'un element algebraic $\alpha \in \mathbb{E}$ el polinomi mònic de grau més petit del qual és arrel. És un polinomi primer $P(X) \in \mathbb{K}[X]$ amb la propietat que els polinomis de $\mathbb{K}[X]$ que s'anul·len en α són exactament els seus múltiples.

Per exemple, $X^2 - 2$ i $X^3 - 2$ són els polinomis mínims de $\sqrt{2}, \sqrt[3]{2}$ sobre $\mathbb{Q}$ però $X^n - 1$ no és el polinomi mínim de $e^{2\pi i/n}$; en aquest cas (no és fàcil de veure-ho) el polinomi mínim és un divisor estricte de $X^n - 1$ que té grau $\varphi(n)$ anomenat *polinomi ciclotòmic* n-èsim.

Problemes

0.2. Siguin $f(X), g(X) \in \mathbb{K}[X]$ dos polinomis no tots dos nuls. Siguin

$$f(X)u_i(X) + g(X)v_i(X) = r_i(X)$$

les identitats que s'obtenen en fer l'algorisme d'Euclides estès, amb $r_n = \gcd(f, g)$ i $r_{n+1} = 0$. Demostreu que

1. $u_{n+1} = \alpha \frac{g}{\gcd(f,g)}$, $v_{n+1} = -\alpha \frac{f}{\gcd(f,g)}$ per a algun $\alpha \in \mathbb{K}^*$;
2. $\deg u_i = \deg g - \deg r_{i-1}$ i $\deg v_i = \deg u - \deg r_{i-1}$ per a $i = 1, \ldots, n+1$;

3. u_i i v_i són els polinomis de grau més petit tals que $f(X)u_i(X)+g(X)v_i(X) = r_i(X)$;

0.3. Es considera l'anell de polinomis $\mathbb{K}[X]$ amb coeficients en un cos qualsevol. Siguin $\boldsymbol{r}_0(X), \boldsymbol{r}_1(X) \in \mathbb{K}[X]$ dos polinomis amb $\deg \boldsymbol{r}_0 \geqslant \deg \boldsymbol{r}_1 \geqslant 0$. S'aplica l'algorisme d'Euclides estès obtenint-se successions de polinomis $\boldsymbol{r}_i, \boldsymbol{u}_i$ i $\boldsymbol{v}_i$ amb $1 \leqslant i \leqslant n+1$ fins que $\boldsymbol{r}_n$ és el màxim comú divisor dels dos polinomis inicials i $\boldsymbol{r}_{n+1} = 0$, de manera que se satisfan les igualtats

$$\boldsymbol{r}_0(X)\boldsymbol{u}_i(X) + \boldsymbol{r}_1(X)\boldsymbol{v}_i(X) = \boldsymbol{r}_i(X).$$

Demostreu que:

1. $\deg \boldsymbol{v}_i = \deg \boldsymbol{r}_0 - \deg \boldsymbol{r}_{i-1}$ per a tot $i = 1, \ldots, n+1$;
2. $\deg \boldsymbol{u}_i = \deg \boldsymbol{r}_1 - \deg \boldsymbol{r}_{i-1}$ per a tot $i = 2, \ldots, n+1$.

0.4. Siguin $S(X), N(X) \in \mathbb{K}[X]$ polinomis de graus $\deg S < \deg N$. Siguin n, m enters amb $n + m = \deg N$. Demostreu que existeixen polinomis $L(X)$ i $A(X)$ de graus $\deg L < n$ i $\deg A \leqslant m$ tals que $S(X)L(X) \equiv A(X) \pmod{N}(X)$, i que aquests polinomis són únics.

0.5. Es considera l'anell de polinomis $\mathbb{K}[X]$ amb coeficients en un cos. Siguin

- $\boldsymbol{r}_0, \boldsymbol{r}_1 \in \mathbb{K}[X]$ no tots dos nuls; $d_r = \max\{\deg \boldsymbol{r}_0, \deg \boldsymbol{r}_1\}$;
- $\varepsilon_0, \varepsilon_1 \in \mathbb{K}[X]$; $d_\varepsilon = \max\{\deg \varepsilon_0, \deg \varepsilon_1\}$;
- $f \in \mathbb{K}[X]$ de grau $\deg f > d_r + d_\varepsilon$;
- $\boldsymbol{s}_0 = f\boldsymbol{r}_0 + \varepsilon_0$; $\boldsymbol{s}_1 = f\boldsymbol{r}_1 + \varepsilon_1$.

Es fa l'algorisme d'Euclides estès amb els polinomis $\boldsymbol{r}_0$ i $\boldsymbol{r}_1$ obtenint-se:

- $\boldsymbol{r}_{i-1} = \boldsymbol{r}_i\boldsymbol{q}_i + \boldsymbol{r}_{i+1}$ (divisió euclidiana) per a $i = 1, \ldots, n$;
- $\boldsymbol{r}_n = \gcd(\boldsymbol{r}_0, \boldsymbol{r}_1)$ i $\boldsymbol{r}_{n+1} = 0$;
- coeficients $\boldsymbol{u}_i$ i $\boldsymbol{v}_i$ amb $\boldsymbol{r}_0\boldsymbol{u}_i + \boldsymbol{r}_1\boldsymbol{v}_i = \boldsymbol{r}_i$ per a $i = 0, \ldots, n+1$.

Es fa ara l'algorisme d'Euclides estès amb els polinomis $\boldsymbol{s}_0$ i $\boldsymbol{s}_1$ obtenint-se:

- $\boldsymbol{s}_{i-1} = \boldsymbol{s}_i\boldsymbol{q}'_i + \boldsymbol{s}_{i+1}$ (divisió euclidiana) per a $i = 1, \ldots, m$;
- coeficients $\boldsymbol{u}'_i$ i $\boldsymbol{v}'_i$ amb $\boldsymbol{s}_0\boldsymbol{u}'_i + \boldsymbol{s}_1\boldsymbol{v}'_i = \boldsymbol{s}_i$ per a $i = 0, \ldots, m+1$.

Demostreu que

1. $\boldsymbol{q}'_i = \boldsymbol{q}_i$ per a $i = 1, \ldots, n$;
2. $\boldsymbol{u}'_i = \boldsymbol{u}_i$ i $\boldsymbol{v}'_i = \boldsymbol{v}_i$ per a $i = 0, \ldots, n+1$;
3. $\boldsymbol{s}_n$ és l'últim reste de grau $\deg \boldsymbol{s}_n \geqslant \deg f$;
4. $\boldsymbol{s}_{n+1}$ és el primer reste de grau $\deg \boldsymbol{s}_{n+1} \leqslant d_r + d_\epsilon$;
5. $\boldsymbol{u}'_{n+1} = \alpha\frac{\boldsymbol{r}_1}{\boldsymbol{r}_n}$ i $\boldsymbol{v}'_{i+1} = -\alpha\frac{\boldsymbol{r}_0}{\boldsymbol{r}_n}$ per a algun $\alpha \in \mathbb{F}^*$.

0.4 Cossos finits

Els *cossos* són conjunts on hi ha definides dues operacions, una suma i un producte, amb les mateixes propietats que tenen la suma i el producte de nombres reals: la suma és *associativa*, *commutativa*, té *element neutre* (el zero) i tot element té *invers* ($-a$ és l'invers de a); el producte és associatiu, commutatiu, té element neutre (l'u) i tot element diferent de zero té invers (a^{-1} és l'invers de $a \neq 0$); a més, totes dues operacions estan relacionades a través de la *propietat distributiva*, que és la que permet treure factor comú en una suma de termes en què tots tenen un mateix factor multiplicant.

Els més coneguts, que surten en la majoria de cursos de matemàtiques bàsiques, són el cos $\mathbb{R}$ dels nombres reals i el cos $\mathbb{C}$ dels nombres complexos. També és habitual el cos $\mathbb{Q}$ dels nombres racionals: les fraccions amb numerador i denominador nombres enters.

En aplicacions tecnològiques es treballa molt amb *cossos finits*: conjunts finits on hi ha definides una suma i un producte que els donen estructura de cos. A la secció 0.2 s'han vist exemples de cossos finits: els anells $\mathbb{Z}_p$ de classes de congruència d'enters mòdul un nombre primer p. A la secció 0.3 s'ha vist com construir nous cossos a partir d'uns altres: anells $\mathbb{K}[X]_{P(X)}$ de classes de congruència de polinomis mòdul un polinomi primer $P(X)$. Quan el cos base $\mathbb{K}$ és un cos finit $\mathbb{Z}_p$ el cos $\mathbb{Z}_p[X]_{P(X)}$ és també un cos finit, que conté $q = p^e$ elements, amb $e = \deg P$. Com que més endavant s'hauran de considerar polinomis amb coeficients en cossos finits, en la construcció dels cossos finits es farà servir la lletra $\mathtt{z}$ en comptes de X per denotar la variable: el cos finit de $q = p^e$ elements és l'anell quocient $\mathbb{Z}_p[\mathtt{z}]_{P(\mathtt{z})}$ mòdul un polinomi primer $P(\mathtt{z})$ de grau e.

En certa manera aquestes dues construccions donen tots els cossos finits possibles. Qualsevol cos finit és essencialment un d'aquests:

- Per a cada nombre primer $p \in \mathbb{Z}$ existeix un cos finit de p elements: el cos $\mathbb{Z}_p$ de les classes de congruència d'enters mòdul p. Els elements d'aquest cos es poden representar amb els enters $0, 1, \ldots, p-1$ i les operacions de suma i producte es fan reduint el resultat mòdul p. L'invers pel producte d'un enter a coprimer amb p s'obté com la x d'una solució de la identitat de Bézout $ax + py = 1$.
- Per a cada potència de nombre primer $q = p^e$ amb $e > 1$ existeix un cos finit de p^e elements: el cos $\mathbb{Z}_p[\mathtt{z}]_{P(\mathtt{z})}$ on $P(\mathtt{z})$ és un polinomi primer de grau e. Els elements d'aquest cos es poden representar amb els polinomis $u(\mathtt{z}) = u_0 + u_1\mathtt{z} + \cdots + u_{e-1}\mathtt{z}^{e-1} \in \mathbb{Z}_p[\mathtt{z}]$ de grau $< e$, els quals es poden identificar amb seqüències o vectors $(u_0, u_1, \ldots, u_{e-1})$ de e elements de $\mathbb{Z}_p$; és a dir, amb seqüències de e nombres $0 \leqslant u_i \leqslant p-1$. Les operacions de suma i producte es fan reduint la suma (que en realitat no cal) i el producte de polinomis de $\mathbb{Z}_p[\mathtt{z}]$ mòdul $P(\mathtt{z})$.

Així doncs, per a cada enter $q = p^e$ que sigui potència d'un nombre primer hi ha un (únic) cos finit de q elements. Aquest cos s'acostuma a denotar $\mathbb{F}_q$ o també $\mathrm{GF}(q)$ (Galois Field, en honor d'*Évariste Galois*, que va construir els cossos finits amb nombre d'elements no primer). El primer p s'anomena *característica* del cos finit $\mathbb{F}_{p^e}$. Una propietat important, i que pot portar a confusió fins que un s'hi acostuma, és que en un cos finit de característica p el nombre p és igual a zero: en sumar p vegades l'1, element neutre del producte, dona igual a zero. En particular, en els cossos $\mathbb{F}_{2^e}$ el 2 (i tot nombre parell) és igual a zero.

Grup multiplicatiu. Els elements no nuls d'un cos finit $\mathbb{F}_q$ formen un grup de $q-1$ elements amb l'operació producte. S'anomena *grup multiplicatiu* del cos i se sol denotar $\mathbb{F}_q^*$.

El teorema de Fermat per a congruències mòdul p es generalitza als cossos finits i diu que per tot element $\alpha \in \mathbb{F}_q^*$ la seva potència $(q-1)$-èsima és $\alpha^{q-1} = 1$.

Els elements d'un cos que una potència seva és igual a 1 s'anomenen *arrels de la unitat* del cos. Els elements $\alpha \in \mathbb{K}$ tals que $\alpha^n = 1$ són les *arrels n-èsimes de la unitat*. Per exemple, a $\mathbb{Q}$ i $\mathbb{R}$ les úniques arrels de la unitat són els dos nombres ± 1. A $\mathbb{C}$ per a tot $n \geqslant 1$ hi ha n arrels n-èsimes de la unitat: els nombres $e^{2\pi i k/n}$ per a $k = 0, \ldots, n-1$. El teorema de Fermat assegura que en un cos finit tots els elements diferents de zero són arrels de la unitat: són arrels $(q-1)$-èsimes de la unitat.

Donat un element qualsevol $\beta \in \mathbb{F}_q^*$ les seves potències successives $\beta, \beta^2, \beta^3, \ldots$ són elements de $\mathbb{F}_q^*$, que són tots diferents fins a arribar a una potència $\beta^n = 1$ a partir de la qual es van repetint. Aquest enter n, el més petit amb $\beta^n = 1$, s'anomena *ordre* de β.

Naturalment, com que $\mathbb{F}_q^*$ només té $q-1$ elements diferents, l'ordre de tot element ha de ser necessàriament $\leqslant q-1$. De fet, és fàcil veure que ha de ser un divisor de $q-1$. Un resultat important assegura que

Teorema. *En tot cos finit $\mathbb{F}_q$ existeix algun element $\zeta \neq 0$ d'ordre $q-1$. És a dir, un element tal que les seves potències $1 = \zeta^0, \zeta = \zeta^1, \zeta^2, \ldots, \zeta^{q-2}$ són* tots *els elements no nuls del cos.*

Els elements de $\mathbb{F}_q^*$ d'ordre $q-1$ s'anomenen *elements primitius* del cos. El teorema anterior afirma que tot cos té elements primitius. Es pot veure que en el cos $\mathbb{F}_q$ d' elements primitius n'hi ha $\varphi(q-1)$: el valor de la funció *indicador d'Euler* en $q-1$.

Fixat un element primitiu ζ del cos $\mathbb{F}_q^*$ s'anomenen exponencial discreta i *logaritme discret* de base ζ les aplicacions

$$\exp_\zeta \colon \mathbb{Z}_{q-1} \to \mathbb{F}_q^*, \quad \exp_\zeta(r) = \zeta^r,$$

i la seva inversa

$$\log_\zeta \colon \mathbb{F}_q^* \to \mathbb{Z}_{q-1}, \quad \log_\zeta(\alpha) = r \text{ tal que } \alpha = \zeta^r.$$

Aquestes aplicacions són isomorfismes de grups: són aplicacions bijectives que respecten les operacions en tots dos grups: sumar elements de $\mathbb{Z}_{q-1}$ equival a multiplicar els elements de $\mathbb{F}_q^*$ que els corresponen. L'exponencial i el logaritme discret són la base de molts algorismes *criptogràfics* de *clau pública*: *intercanvi de claus*, *signatura digital*, etc.

Usant un element primitiu $\zeta \in \mathbb{F}_q^*$ es poden representar tots elements no nuls del cos com a potències seves, i això dona una representació alternativa que simplifica els càlculs quan s'han de fer moltes operacions de multiplicar. En efecte, tots els elements no nuls del cos són de la forma ζ^r per a algun enter r, i el producte de dos elements com aquests és $\zeta^r \zeta^s = \zeta^{r+s}$, de manera que per multiplicar elements només cal sumar els seus exponents (que estan determinats només mòdul l'enter $q-1$). En canvi usant aquesta representació la suma és més complicada: donades potències ζ^r i ζ^s s'ha de trobar un enter t tal que $\zeta^r + \zeta^s = \zeta^t$. Algunes implementacions d'aritmètica en cossos finits fan servir l'anomenat *logaritme de Zech* per a calcular sumes d'elements expressats com a potències d'un element primitiu.

Característica 2. Com ja s'ha dit els cossos finits més usats en aplicacions per a les tecnologies digitals són els cossos de característica 2, que tenen 2^e elements. Els elements d'un cos de característica 2 es representen amb cadenes binàries:

$$\mathbb{F}_{2^e} \approx \big\{u_0u_1\cdots u_{e-1} : u_i \in \{0,1\}\big\}.$$

La suma de dos elements és molt senzilla: consisteix simplement en fer XOR bit a bit. En canvi per multiplicar-los les cadenes s'han de veure com a polinomis binaris

$$u_0u_2\cdots u_{e-1} \rightsquigarrow u_0 + u_1X + \cdots + u_{e-1}X^{e-1} \in \mathbb{Z}_2[X],$$

multiplicar-los normalment com a polinomis, i finalment reduir el resultat mòdul $P(X)$, el polinomi binari primer de grau e usat per construir el cos $\mathbb{F}_{2^e}$. Els microprocessadors actuals incorporen *instruccions* que donen el producte de polinomis binaris en termes de les paraules binàries corresponents, a partir de les quals és fàcil implementar el producte en un cos finit. L'aritmètica en cossos finits binaris també té relació amb els *LFSR* (registre de desplaçament amb retroalimentació lineal).

És important observar que per donar un cos finit de característica 2 no n'hi ha prou a dir quants elements té (quina potència $q = 2^e$ és el nombre d'elements), sinó que s'ha d'especificar quin polinomi $P(X)$ es fa servir com a mòdul, ja que l'operació producte depèn d'això: els estàndards de tecnologies digitals que usen cossos finits han de dir en les seves especificacions quin polinomi primer fan servir com a mòdul.

Exemple: $\mathbb{F}_8$. Per construir el cos $\mathbb{F}_q$ de $q = 8 = 2^3$ elements s'ha d'agafar un polinomi binari primer de grau 3. D'aquests polinomis n'hi ha dos: $= \mathtt{z}^3 + \mathtt{z} + 1$ i $\mathtt{z}^3 + \mathtt{z}^2 + 1$. Es tria per exemple el primer: $P(\mathtt{z}) = \mathtt{z}^3 + \mathtt{z} + 1$. El cos finit $\mathbb{F}_8$ està format per les cadenes binàries de longitud 3, identificades amb els polinomis binaris de grau menor que 3:

$$\begin{aligned}\mathbb{F}_8 = \mathbb{Z}_2[\mathtt{z}]_{P(\mathtt{z})} &\approx \{\mathtt{000}, \mathtt{001}, \mathtt{010}, \mathtt{011}, \mathtt{100}, \mathtt{101}, \mathtt{110}, \mathtt{111}\}\\ &\approx \{0, \mathtt{z}^2, \mathtt{z}, \mathtt{z}+\mathtt{z}^2, 1, 1+\mathtt{z}^2, 1+\mathtt{z}, 1+\mathtt{z}+\mathtt{z}^2\}.\end{aligned}$$

La suma es fa sumant els polinomis tenint en compte que els coeficients són elements del cos binari $\mathbb{Z}_2$, i correspon a fer *XOR* amb les cadenes binàries. La "taula de sumar" és:

+	000	001	010	011	100	101	110	111
000	000	001	010	011	100	101	110	111
001	001	000	011	010	101	100	111	110
010	010	011	000	001	110	111	100	101
011	011	010	001	000	111	110	101	100
100	100	101	110	111	000	001	010	011
101	101	100	111	110	001	000	011	010
110	110	111	100	101	010	011	000	001
111	111	110	101	100	011	010	001	000

Pel producte s'han de multiplicar els polinomis, amb resultat un polinomi de grau $\leqslant 4$, que s'ha de reduir mòdul $P(\mathtt{z})$ canviant-lo pel reste de la divisió euclidiana, que ja serà un polinomi de grau $\leqslant 2$. Per exemple, per multiplicar 111 per 101 es calcula

$$(\mathtt{z}^2+\mathtt{z}+1)(\mathtt{z}^2+1) = \mathtt{z}^4+\mathtt{z}^3+\mathtt{z}+1 = P(\mathtt{z})(\mathtt{z}+1)+\mathtt{z}^2+\mathtt{z} \equiv \mathtt{z}^2+\mathtt{z} \pmod{P(\mathtt{z})}$$

obtenint-se el resultat 011. La taula de multiplicar queda:

+	000	001	010	011	100	101	110	111
000	000	000	000	000	000	000	000	000
001	000	011	110	101	001	010	111	100
010	000	110	001	111	010	100	011	101
011	000	101	111	010	011	110	100	001
100	000	001	010	011	100	101	110	111
101	000	010	100	110	101	111	001	011
110	000	111	011	100	110	001	101	010
111	000	100	101	001	111	011	010	110

El cos té $\varphi(7) = 6$ elements primitius: ho són tots els elements no nuls diferents de 1. Agafant per exemple l'element $\alpha = 010 = \mathtt{z}$ es té la taula següent amb la representació dels elements del cos com a polinomi, cadena binària o potència de l'element primitiu:

polinòmica	binària	exponencial
0	000	
1	100	α^0
$\mathtt{z}$	010	α^1
$\mathtt{z}^2$	001	α^2
$1+\mathtt{z}$	110	α^3
$\mathtt{z}+\mathtt{z}^2$	011	α^4
$1+\mathtt{z}+\mathtt{z}^2$	111	α^5
$1+\mathtt{z}^2$	101	α^6

Exemple: $\mathbb{F}_{256}$. El cos de 256 elements és un dels més usats en la pràctica ja que els seus elements es poden identificar amb octets (bytes), que són blocs binaris que es fan servir molt sovint en processar informació digital.

Per construir aquest cos s'ha d'agafar un polinomi primer de grau 8. Per exemple, es pot agafar el polinomi

$$P(\mathtt{z}) = 1 + \mathtt{z}^2 + \mathtt{z}^3 + \mathtt{z}^6 + \mathtt{z}^8.$$

Per multiplicar l'element $\alpha = 01010101 = \mathtt{z} + \mathtt{z}^3 + \mathtt{z}^6 + \mathtt{z}^7$ per l'element $\beta = 00001111 = \mathtt{z}^4 + \mathtt{z}^5 + \mathtt{z}^6 + \mathtt{z}^7$ es multipliquen els polinomis i es fa la divisió euclidiana del producte pel mòdul $P(\mathtt{z})$:

$$\begin{aligned}(\mathtt{z} + \mathtt{z}^3 + \mathtt{z}^6 + \mathtt{z}^7)(\mathtt{z}^4 + \mathtt{z}^5 + \mathtt{z}^6 + \mathtt{z}^7) &= \mathtt{z}^5 + \mathtt{z}^6 + \mathtt{z}^{13} + \mathtt{z}^{14} \\ &= P(\mathtt{z})(1 + \mathtt{z}^3 + \mathtt{z}^4 + \mathtt{z}^5 + \mathtt{z}^6) + 1 + \mathtt{z}^6\end{aligned}$$

obtenint-se així el producte $\alpha\beta = 10000010 = 1 + \mathtt{z}^6$.

Polinomis sobre cossos finits. Es donen a continuació algunes propietats de polinomis i arrels en cossos finits: donada una extensió $\mathbb{E}/\mathbb{F}$ de grau e i un element $\alpha \in \mathbb{E}$,

- α és arrel d'algun polinomi de $\mathbb{F}[X]$; per exemple, sempre ho és de $X^{q^e} - X$;

- el polinomi $P_\alpha(X) \in \mathbb{F}[X]$ de grau més petit tal que $P_\alpha(\alpha) = 0$ és primer: és el *polinomi mínim* de α sobre $\mathbb{F}$ i el seu grau $\nu = \deg P_\alpha(X)$ divideix e;
- els polinomis de $\mathbb{F}[X]$ que s'anul·len en α són els múltiples de $P_\alpha(X)$;
- les potències $\alpha, \alpha^q, \alpha^{q^2}, \ldots, \alpha^{q^{e-1}} \in \mathbb{E}$ són totes les arrels de $P_\alpha(X)$.

Problemes

0.6. Sigui $\mathbb{F} = \mathbb{F}_q$ un cos finit de q elements. Sigui $\zeta \in \mathbb{F}^*$ un element primitiu. Per a cada element $m \in \mathbb{Z}_{q-1}$ amb $\zeta^m \neq -1$ es defineix el *logaritme de Zech* $Z(m)$ com l'únic enter mòdul $q-1$ tal que

$$1 + \zeta^m = \zeta^{Z(m)}.$$

1. Comproveu que l'aplicació $Z\colon \mathbb{Z}_{q-1} \smallsetminus \{\log_\zeta(-1)\} \to \mathbb{Z}_{q-1}$ està ben definida.
2. Comproveu que se satisfà la igualtat

$$\zeta^m + \zeta^n = \zeta^{m+Z(n-m)}, \qquad \text{sempre que} \quad \zeta^m \neq -\zeta^n.$$

3. En quin element $m \in \mathbb{Z}_{q-1}$ no està definit el seu logaritme de Zech?
   OBSERVACIÓ: depèn de si la característica és 2 o és senar.
4. Quin valor de $\mathbb{Z}_{q-1}$ no és logaritme de Zech de ningú?
5. Comproveu que en característica 2 el logaritme de Zech és una bijecció del conjunt dels enters $1 \leqslant m \leqslant q-1$ en ell mateix.
6. Construïu el logaritme de Zech en els cossos $\mathbb{F}_8$ i $\mathbb{F}_9$.

0.7. Sigui $q = p^e$ una potència de primer i sigui $\mathbb{F}_q = \mathbb{F}_p[\mathtt{z}]_{P(\mathtt{z})}$ el cos finit construït amb congruències mòdul un polinomi primer $P(\mathtt{z}) = \mathtt{p}_0 + \mathtt{p}_1\mathtt{z} + \cdots + \mathtt{p}_{e-1}\mathtt{z}^{e-1}$. Es denotarà $\mathtt{x} = [\mathtt{z}]_{P(\mathtt{z})} \in \mathbb{F}_q$

El cos $\mathbb{F}_q$ es pot veure com un $\mathbb{F}_p$-espai vectorial de dimensió e amb base $1, \mathtt{x}, \ldots, \mathtt{x}^{e-1}$.

1. Comproveu que l'aplicació $m_\alpha\colon \mathbb{F}_q \to \mathbb{F}_q$ definida com $m_\alpha(x) = \alpha{\cdot}x$, que multiplica la variable per un element $\alpha \in \mathbb{F}_q$ fixat, és una aplicació $\mathbb{F}_p$-lineal.
2. Sigui $M_\alpha \in \mathrm{Mat}_t(\mathbb{F}_p)$ la matriu de m_α en la base de les potències de $\mathtt{x}$. Calculeu la matriu $M_\mathtt{x}$ i doneu una fórmula per a les matrius M_α en funció de les coordenades $\mathtt{a}_i$ de l'element $\alpha = \mathtt{a}_0 + \mathtt{a}_1\mathtt{x}_1 + \cdots + \mathtt{a}_{e-1}\mathtt{x}^{e-1}$ i de la matriu $m_\mathtt{x}$.
3. Vegeu que l'aplicació $\alpha \mapsto M_\alpha\colon \mathbb{F}_q \to \mathrm{Mat}_t(\mathbb{F}_p)$ és un monomorfisme d'anells que permet veure el cos $\mathbb{F}_q$ com un subanell de l'anell de les matrius.

0.5 Álgebra lineal

En els cursos d'àlgebra lineal dels graus de ciències i enginyeria es treballa sobretot en el cos $\mathbb{R}$ dels nombres reals com a cos d'escalars, i de vegades també en el cos $\mathbb{C}$ dels nombres complexos.

En teoria de codis l'àlgebra lineal es fa principalment sobre cossos finits. En aquesta secció es recorden alguns resultats bàsics d'àlgebra lineal, fent èmfasi en aspectes que són importants quan el cos d'escalars és un cos finit.

Es denotarà $\mathbb{K}$ un cos qualsevol i $\mathbb{F} = \mathbb{F}_q$ un cos finit de característica p i $q = p^e$ d'elements.

Els $\mathbb{K}$-*espais vectorials* són conjunts de *vectors* on hi ha definides dues operacions: la suma de vectors i el *producte per escalars*, que tenen les propietats habituals.

L'exemple més important d'espai vectorial és el producte cartesià

$$\mathbb{K}^n = \{\boldsymbol{x} = (x_1, x_2, \ldots, x_n) : x_i \in \mathbb{K}\}$$

format per les *n-tuples* d'elements del cos amb la suma i el producte per escalars definits component a component.

Combinacions lineals, independents, generadors i bases. Una *combinació lineal* de vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ amb *coeficients* els escalars $x_1, \ldots, x_k$ és un vector expressat com la suma $\boldsymbol{v} = \sum_{i=1}^{k} x_i \boldsymbol{v}_i$ de múltiples dels vectors $\boldsymbol{v}_i$ amb coeficients multiplicadors els escalars x_i.

Els vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$ es diuen *generadors* si tot vector de l'espai és combinació lineal seva; es diuen *independents* si l'única manera d'obtenir el vector zero com a combinació lineal seva és agafant tots els coeficients iguals a zero (de manera equivalent, si els vectors que són combinació lineal seva ho són de manera única); són una *base* si són alhora generadors i independents.

Tot espai vectorial V té bases i totes les bases tenen el mateix nombre d'elements, que es diu la *dimensió* de l'espai, i es denota $\dim V$.

Fixar una base $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ d'un espai vectorial V permet identificar els seus vectors $\boldsymbol{v} \in V$ amb n-tuples d'escalars $(x_1, \ldots, x_n)$. Els x_i són les *coordenades* del vector: els coeficients de la combinació lineal (única) dels vectors de la base que dona $\boldsymbol{v}$:

$$\boldsymbol{v} \approx (x_1, x_2, \ldots, x_n), \qquad \boldsymbol{v} = x_1\boldsymbol{v}_1 + x_2\boldsymbol{v}_2 + \cdots + x_n\boldsymbol{v}_n.$$

L'espai $\mathbb{K}^n$ té dimensió n i admet com a base els vectors $\boldsymbol{e}_i = (0, \ldots, 0, 1, 0, \ldots, 0)$ que tenen totes les components iguals a zero excepte una que val 1, la i-èsima. En aquesta base, anomenada *base canònica*, les coordenades d'un vector $\boldsymbol{x} = (x_1, \ldots, x_n)$ són directament les seves components x_i, ja que $\boldsymbol{x} = \sum_{i=1}^{n} x_i \boldsymbol{e}_i$.

L'anell de polinomis $\mathbb{K}[X]$ amb la suma i el producte per escalars és un espai vectorial de dimensió infinita que admet com a *base canònica* els monomis $1, X, X^2, X^3, \ldots$. El subconjunt $\mathbb{K}[X]_n$ dels polinomis de grau $< n$ és un subespai vectorial, amb base $1, X, \ldots, X^{n-1}$. Si $N(X)$ és un polinomi de grau n aleshores el conjunt de classes de congruència $\mathbb{K}[X]_{N(X)}$ s'identifica com a espai vectorial amb $\mathbb{K}[X]_n$, agafant com a representant de cada classe l'únic polinomi de grau $< n$ que conté, però ara en aquest conjunt hi ha una nova operació: el producte de polinomis mòdul $N(X)$, que li dona estructura d'anell.

Quan el cos d'escalars és el cos finit $\mathbb{F}_q$ de q elements els espais vectorials de dimensió finita contenen només una quantitat finita de vectors. En representar-los com a combinacions lineals $\sum_{i=1}^{n} x_i \boldsymbol{v}_i$ de vectors d'una base cada coeficient es pot triar arbitràriament entre els q escalars possibles. Es dedueix el fet molt important següent, que no té un anàleg quan es fa àlgebra lineal sobre cossos infinits:

Proposicio. *Un $\mathbb{F}_q$-espai vectorial de dimensió n conté exactament q^n vectors.*

Subespais. Un *subespai vectorial* d'un espai V és un subconjunt tancat per la suma i el producte per escalars; és a dir, tal que tota combinació lineal de vectors del subconjunt pertany al subconjunt.

Els subespais se solen donar de dues maneres:

- Amb generadors: el subespai consisteix en les combinacions lineals d'una família de vectors donats; el *rang* dels vectors és la dimensió del subespai que generen.
- Amb equacions: les coordenades dels vectors del subespai són les solucions d'un sistema d'equacions lineals homogeni; la dimensió del subespai de solucions és la diferència entre el nombre de variables i el rang de la matriu del sistema (el nombre d'equacions independents).

En les aplicacions de l'àlgebra lineal sovint s'ha de passar d'una descripció d'un subespai a l'altra. La manera de fer-ho es veu en tots els cursos bàsics: tan en un sentit com en l'altre, s'ha de resoldre un sistema d'equacions lineals.

Transformacions elementals i reducció gaussiana. S'anomena *transformació elemental* d'una família de vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_r$ una transformació d'un dels tres tipus següents:

- *tipus I:* intercanviar dos dels vectors;
- *tipus II:* multiplicar un dels vectors per un escalar no nul;
- *tipus III:* sumar a un dels vectors un múltiple qualsevol d'un altre vector.

En aplicar transformacions elementals a una família de vectors es conserven les propietats de ser linealment independents, generadors o base, i també el subespai que generen.

Donades dues bases qualsevol d'un espai vectorial sempre es pot passar de l'una a l'altra fent transformacions elementals.

Dues matrius $\mathbf{A}, \mathbf{B} \in \mathrm{Mat}_{m\times n}(\mathbb{K})$ es diuen *equivalents per files* si es pot passar de l'una a l'altra fent transformacions elementals de files (resp. columnes). És el mateix que dir que les files de totes dues matrius generen el mateix subespai de $\mathbb{K}^n$, o també que existeix una matriu invertible $\mathbf{P} \in \mathrm{Mat}_m(\mathbb{K})$ tal que $\mathbf{B} = \mathbf{P} \cdot \mathbf{A}$.

S'anomena *eliminació gaussiana* (o reducció gaussiana) o *eliminació de Gauss-Jordan* la reducció d'una matriu a *forma esglaonada* o a *forma esglaonada reduïda* aplicant transformacions elementals dels seus vectors-fila.

Tota matriu és equivalent a diverses matrius en forma esglaonada però és equivalent a *una única matriu* en forma esglaonada reduïda: la forma esglaonada reduïda és un representant canònic de la classe d'equivalència per files d'una matriu.

L'aplicació més important de la reducció gaussiana és la resolució de sistemes d'equacions lineals: com que les solucions del sistema es veuen directament quan la seva matriu està en forma esglaonada reduïda per resoldre'l n'hi ha prou a reduir la matriu del sistema donat. També es fa servir per calcular la inversa d'una matriu o el rang d'una família de vectors, entre altres aplicacions. En els codis correctors d'errors la forma esglaonada reduïda d'una matriu generadora dona una matriu generadora sistemàtica, quan aquesta existeix (veure secció 5.2).

Norma de Hamming. Donat un $\mathbb{K}$-espai vectorial V de dimensió finita sobre un cos $\mathbb{K}$ qualsevol i fixada una base $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n$ es pot definir la norma de Hamming (o *pes de Hamming*) d'un vector $\boldsymbol{v} = \sum_{i=1}^{n} x_i \boldsymbol{v}_i$ com el nombre de coordenades diferents de zero d'aquest vector:

$$\|\boldsymbol{v}\| = \|\boldsymbol{v}\|_H = w(\boldsymbol{v}) := \big|\{i : 1 \leqslant i \leqslant n, x_i \neq 0\}\big|.$$

La norma de Hamming és un enter entre zero i n i satisfà la desigualtat triangular:

$$\|\boldsymbol{u} + \boldsymbol{v}\| \leqslant \|\boldsymbol{u}\| + \|\boldsymbol{v}\|.$$

Com es fa amb les normes en espais vectorials reals o complexos, aquí també es pot definir una distància amb la mateixa expressió. La *distància de Hamming* es defineix com

$$d(\boldsymbol{u}, \boldsymbol{v}) = \|\boldsymbol{u} - \boldsymbol{v}\|.$$

El valor de la distància de Hamming $d\colon V \times V \to \mathbb{R}$ és el nombre de coordenades diferents que tenen dos vectors i té les propietats d'una *distància*:

- $d(\boldsymbol{u}, \boldsymbol{v}) \geqslant 0$ i $d(\boldsymbol{u}, \boldsymbol{v}) = 0 \Leftrightarrow \boldsymbol{u} = \boldsymbol{v}$ (positivitat);
- $d(\boldsymbol{v}, \boldsymbol{u}) = d(\boldsymbol{u}, \boldsymbol{v})$ (simetria);
- $d(\boldsymbol{u}, \boldsymbol{v}) \leqslant d(\boldsymbol{u}, \boldsymbol{w}) + d(\boldsymbol{w}, \boldsymbol{v})$ (desigualtat triangular).

Producte bilineal. En els cursos d'àlgebra lineal s'estudien productes escalars en espais vectorials reals i productes hermítics en espais vectorials complexos. Aquestes estructures introdueixen conceptes geomètrics en els espais: la mida dels vectors (norma) i l'angle que formen. En particular també el concepte d'ortogonalitat.

Quan el cos d'escalars és un cos qualsevol $\mathbb{K}$ no hi ha construccions equivalents. El problema està en la condició de positivitat que s'exigeix en el cas real o complex.

Tot i així es poden considerar productes bilineals simètrics que tenen un comportament anàleg en alguns aspectes. Donat un $\mathbb{K}$-espai vectorial de dimensió finita i fixada una base $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_n$ es defineix una aplicació $V \times V \to \mathbb{K}$ que assigna a cada parell de vectors un escalar de la manera següent:

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = x_1 y_1 + x_2 y_2 + \cdots + x_n y_n, \qquad \boldsymbol{u} = \sum x_i \boldsymbol{v}_i, \quad \boldsymbol{v} = \sum y_i \boldsymbol{v}_i.$$

Aquesta aplicació, que també s'anomenarà *producte escalar*, és clarament bilineal (lineal respecte de cadascuna de les dues variables) i simètrica.

Dos vectors es diuen *ortogonals* si el seu producte escalar és zero: $\langle \boldsymbol{u}, \boldsymbol{v} \rangle = 0$, i es denota $\boldsymbol{u} \perp \boldsymbol{v}$. El conjunt de tots els vectors que són ortogonals a un vector donat, o a un conjunt de vectors, és un subespai vectorial: $S^\perp = \{\boldsymbol{v} \in V : \boldsymbol{u} \perp \boldsymbol{v}\ \forall \boldsymbol{u} \in S\}$ si $S \subseteq V$.

Igual que en el cas real o complex la dimensió dels subespais ortogonals de subespais $V \subseteq \mathbb{K}^n$ satisfà $\dim V + \dim V^\perp = n$. En canvi, a diferència del cas real o complex, no es pot assegurar que aquests dos espais estiguin en suma directa, ja que hi ha vectors no nuls ortogonals a ells mateixos. Per exemple sobre el cos binari $\mathbb{K} = \mathbb{F}_2$ tot vector $(1, 1, \ldots, 1) \neq \mathbf{0}$ de longitud parell és ortogonal a ell mateix.

Matrius de Vandermonde. Tots els conceptes i propietats estudiats per a matrius reals o complexes funcionen exactament igual en un cos qualsevol: operacions, rang, determinant, matriu inversa, polinomi característic, vectors i valors propis, diagonalització, etc.

En teoria de codis apareixen sovint *matrius de Vandermonde*, que tenen aplicacions en moltes branques de les matemàtiques: donada una n-tupla $\boldsymbol{a} = (a_1, a_2, \ldots, a_n)$ d'elements d'un cos $\mathbb{K}$ es considera la matriu:

$$\mathbf{V}_{\boldsymbol{a}} = \begin{bmatrix} 1 & a_1 & a_1^2 & \cdots & a_1^{n-1} \\ 1 & a_2 & a_2^2 & \cdots & a_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & a_n & a_n^2 & \cdots & a_n^{n-1} \end{bmatrix}.$$

El determinant d'aquesta matriu es pot calcular aplicant les propietats del determinant respecte les transformacions elementals de files i columnes, i dona:

$$\det \mathbf{V}_{\boldsymbol{a}} = \prod_{i>j} (a_i - a_j).$$

En particular, la matriu de Vandermonde és invertible si, i només si, els a_i són tots diferents.

La matriu de Vandermonde és la matriu de l'aplicació lineal $\mathrm{av}_{\mathbf{a}} \colon \mathbb{K}[X]_n \to \mathbb{K}^n$ que avalua els polinomis en els a_i

$$\mathrm{av}_{\mathbf{a}}\big(\boldsymbol{v}(X)\big) = \big(\boldsymbol{v}(a_1), \boldsymbol{v}(a_2), \ldots, \boldsymbol{v}(a_n)\big).$$

L'aplicació lineal inversa és l'aplicació d'interpolació $\mathrm{itp}_{\mathbf{a}} \colon \mathbb{K}^n \to \mathbb{K}[X]_n$ que dona el polinomi interpolador

$$\mathrm{itp}_{\mathbf{a}}(\boldsymbol{x}) = \boldsymbol{v}(X) \quad \Leftrightarrow \quad \boldsymbol{v}(a_i) = x_i.$$

Per tant, la matriu inversa $\mathbf{V}_{\boldsymbol{a}}^{-1}$, que és la matriu de l'aplicació $\mathrm{itp}_{\boldsymbol{a}}$, té per columnes els coeficients dels polinomis interpoladors dels vectors de la base canònica, que són els polinomis de Lagrange:

$$\mathrm{itp}_{\mathbf{a}}(\boldsymbol{e}_i) = \ell_i(X) = \prod_{j \neq i} \frac{X - a_j}{a_i - a_j}.$$

Transformada de Fourier discreta. Un cas particular especialment important s'obté en agafar un element $\zeta \in \mathbb{K}$ que sigui una *arrel primitiva N-èsima de la unitat*: les potències $1 = \zeta^0, \zeta = \zeta^1, \zeta^2, \zeta^3, \ldots, \zeta^{N-1}$ són totes diferents i $\zeta^N = 1$. En aquest cas el seu invers $\zeta^{-1} = \zeta^{N-1}$ també té la mateixa propietat. La matriu de Vandermonde construïda amb els elements a_i que són les potències de ζ^{-1} s'anomena matriu de la *transformada de Fourier discreta*:

$$\mathcal{F}(\zeta) = \begin{bmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & \zeta^{-1} & \zeta^{-2} & \zeta^{-3} & \cdots & \zeta^{-(N-1)} \\ 1 & \zeta^{-2} & \zeta^{-4} & \zeta^{-6} & \cdots & \zeta^{-2(N-1)} \\ 1 & \zeta^{-3} & \zeta^{-6} & \zeta^{-9} & \cdots & \zeta^{-3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & & \vdots \\ 1 & \zeta^{-(N-1)} & \zeta^{-2(N-1)} & \zeta^{-3(N-1)} & \cdots & \zeta^{-(N-1)^2} \end{bmatrix} = \big[\zeta^{-ij}\big]_{0 \leqslant i,j \leqslant N-1}.$$

Aquesta matriu té la propietat important que assegura que la seva inversa és essencialment (llevat d'un factor N) la matriu $\mathcal{F}(\zeta^{-1})$. És a dir, es té la identitat $\mathcal{F}(\zeta)\mathcal{F}(\zeta^{-1}) = N\mathbf{I}_N$.

En termes d'avaluació i interpolació el que diu això és que avaluar en les potències de ζ^{-1} equival (llevat d'un factor N) a interpolar en les potències de ζ: donat un polinomi $\boldsymbol{v}(X) = \sum v_i X^i \in \mathbb{F}[X]_n$, el polinomi $P(X) = \frac{1}{N}\sum \boldsymbol{v}(\zeta^{-i})X^i$ és el polinomi interpolador amb valors $P(\zeta^i) = v_i$.

Siguin $\boldsymbol{x} = (x_0, x_1, \ldots, x_{N-1}) \in \mathbb{K}^n$ i $\boldsymbol{X} = (X_0, X_1, \ldots, X_{N-1}) = \mathcal{F}(\zeta)(\boldsymbol{x})$. La relació entre les components x_i d'un vector i les components X_i de la seva transformada és:

$$X_i = \sum_{j=0}^{N-1} x_j \zeta^{-ij}, \quad x_i = \frac{1}{N}\sum_{j=0}^{N-1} X_i \zeta^{ij},$$

que són expressions anàlogues a les de la transformada de Fourier ordinaria en els complexos.

Problemes

0.8. *Determinant de Vandermonde.* Comproveu la identitat següent, coneguda amb el nom de *determinant de Vandermonde*:

$$\begin{vmatrix} 1 & a_1 & a_1^2 & \cdots & a_1^{n-1} \\ 1 & a_2 & a_2^2 & \cdots & a_2^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & a_n & a_n^2 & \cdots & a_n^{n-1} \end{vmatrix} = \prod_{i>j}(a_i - a_j)$$

0.9. *Transformada de fourier inversa.* Sigui $\mathbb{K}$ un cos qualsevol. Sigui $\zeta \in \mathbb{K}$ una arrel N-èsima primitiva de la unitat: un element no nul tal que les seves primeres N potències $1 = \zeta^0, \zeta = \zeta^1, \zeta^2, \ldots, \zeta^{N-1}$ són totes diferents i amb $\zeta^N = 1$.

Demostreu les *fórmules d'ortogonalitat*

$$\sum_{i=0}^{N-1} \zeta^{ik} = \begin{cases} N, & k = 0, \\ 0, & k \neq 0. \end{cases}$$

i a partir d'elles calculeu la inversa de la transformada discreta de Fourier:

$$\mathcal{F}(\zeta) = \left[\zeta^{ij}\right]_{0 \leqslant i,j \leqslant N-1} \quad \Rightarrow \quad \mathcal{F}(\zeta)^{-1} = \frac{1}{N}\mathcal{F}(\zeta^{-1}).$$

0.10. *Fita de Singleton.* Sigui $\mathbb{K}$ un cos qualsevol. Per a cada subespai $V \subseteq \mathbb{K}^n$ es defineix

$$w_{\min}(V) = \min\{w(\boldsymbol{v}) : \boldsymbol{v} \in V : \boldsymbol{v} \neq \boldsymbol{0}\}, \quad \text{on } w \text{ denota el pes de Hamming.}$$

1. Demostreu que si $\dim V = k$ aleshores

$$w_{\min}(V) \leqslant n - k + 1.$$

2. Siguin $\alpha_1, \alpha_2, \ldots, \alpha_n$ elements diferents de $\mathbb{K}$. Comproveu que per a tot polinomi $f(X) \in \mathbb{K}[X]$ es té

$$w(\boldsymbol{v}) \geqslant n - \deg f \quad \text{per al vector} \quad \boldsymbol{v} = \big(f(\alpha_1), \ldots, f(\alpha_n)\big).$$

3. Per a cada k amb $1 \leqslant k \leqslant n$ construïu un subespai $V \subseteq \mathbb{K}^n$ amb

$$w_{\min}(V) = n - k + 1.$$

0.11. *Propietat MDS i àlgebra lineal.* Sigui $\mathbb{K}$ un cos qualsevol. Un subespai $V \subseteq \mathbb{K}^n$ de dimensió k es diu MDS si $w_{\min}(V) = n - k + 1$, on el pes mínim $w_{\min}$ s'ha definit al problema **0.10**. Una matriu "apaïsada" $\mathbf{A} \in \mathrm{Mat}_{r\times n}(\mathbb{K})$ amb $r \leqslant n$ es diu MDS si tot subconjunt de r columnes de la matriu és independent (com a vectors de $\mathbb{K}^r$).

Sigui $V \subseteq \mathbb{K}^n$ un subespai de dimensió r. Sigui $\mathbf{G} \in \mathrm{Mat}_{k\times n}(\mathbb{K})$ una matriu que té per files una base de V. Sigui $\mathbf{H} \in \mathrm{Mat}_{m\times n}(\mathbb{K})$ la matriu d'un sistema lineal homogeni que té per solucions l'espai V, amb $m = n - k$. Demostreu que són equivalents:

1. V és MDS;
2. $\mathbf{G}$ és MDS;
3. $\mathbf{H}$ és MDS.

i deduïu que per a tot parell de matrius $\mathbf{A} \in \mathrm{Mat}_{r\times n}(\mathbb{K})$ i $\mathbf{B} \in \mathrm{Mat}_{s\times n}(\mathbb{K})$ de rangs màxims r i s amb $r + s = n$ i tals que $\mathbf{A} \cdot \mathbf{B}^{\mathsf{T}} = \mathbf{0}$, una d'elles és MDS si, i només si, ho és l'altra.

0.12. *Propietat MDS i Vandermonde.* Sigui $\mathbb{K}$ un cos qualsevol. Siguin $r \leqslant n$. En el problema **0.11** s'ha definit matriu MDS. Demostreu que

1. Si $\alpha_1, \ldots, \alpha_n$ són elements diferents del cos $\mathbb{K}$ la matriu

$$\begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \alpha_3^2 & \cdots & \alpha_n^2 \\ \vdots & \vdots & \vdots & & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \alpha_3^{k-1} & \cdots & \alpha_n^{k-1} \end{bmatrix} \qquad \text{és MDS.}$$

2. Si $\beta \in \mathbb{K}$ té les primeres n potències $1 = \beta^0, \beta = \beta^1, \beta^2, \ldots, \beta^{n-1}$ totes diferents, la matriu

$$\begin{bmatrix} 1 & \beta & \beta^2 & \ldots & \beta^{n-1} \\ 1 & \beta^2 & \beta^4 & \ldots & \beta^{2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \beta^r & \beta^{r2} & \ldots & \beta^{r(n-1)} \end{bmatrix} \qquad \text{és MDS.}$$

0.13. *Pes mínim en termes de la matriu de control.* Sigui $\mathbb{K}$ un cos qualsevol. Sigui $\mathbf{H} \in \mathrm{Mat}_{m\times n}(\mathbb{K})$ una matriu, amb $m \leqslant n$. Sigui $V \subseteq \mathbb{K}^n = \{\boldsymbol{v} \in \mathbb{K}^n : \mathbf{H} \cdot \boldsymbol{v} = \mathbf{0}\}$ el subespai vectorial determinat pel sistema lineal homogeni de matriu $\mathbf{H}$. Sigui $w_{\min}(V)$ com en el problema anterior, i sigui k la seva dimensió.

1. Justifiqueu que $k \leqslant n - m$.
2. Demostreu que $w_{\min}(V) = w$ si, i només si,
   (a) menys de w columnes de la matriu $\mathbf{H}$ són sempre linealment independents, i
   (b) existeixen w columnes de $\mathbf{H}$ que són linealment dependents.
3. Demostreu que $w_{\min}(V) \leqslant n - k + 1$.

1 Codis i codificació

Aquí el terme *codificació* s'entén com una manera de representar informació en forma de seqüències de símbols d'un alfabet, habitualment com una seqüència de dígits binaris `0` i `1`. La codificació de la informació persegueix objectius diversos:

- Adequació al medi. Diferents classes d'informació (text, so, imatge, ...) es tradueixen a un format apropiat per al seu tractament i processat en dispositius d'emmagatzematge o comunicacions. Típicament es codifica en binari. Per exemple, un text es *codifica en binari* amb el codi ASCII o un so es digitalitza a través d'un procés de mostreig i quantització que el converteix en una seqüència de dígits binaris.
- Eficiència (compressió). Es treu redundància de les dades per tal d'aconseguir representar la informació amb seqüències el més curtes possible de símbols d'un alfabet el més petit possible.
- Fiabilitat (correcció d'errors). Afegint redundància a les dades es poden detectar i corregir errors que es produeixen en transmetre-les a través d'un canal de comunicacions o en emmagatzemar-les en un dispositiu de memòria.
- Seguretat (xifrat). Amb tècniques criptogràfiques es codifica la informació usant una clau, de manera que sense conèixer aquesta clau no es pugui recuperar la informació només a partir de les dades xifrades.

1.1 Alfabets, paraules i codis

Referència: Brunat-Ventura [3, Capítol 3]

Els *alfabets* són els conjunts de símbols o lletres que representen certs *fonemes* del llenguatge oral. Aquí es considera una accepció més general d'aquesta paraula:

Definició 1.1 (Alfabet)**.** *Un alfabet és un conjunt finit no buit:*

$$\mathbb{A} = \{\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_q\}.$$

Els seus elements s'anomenen lletres o símbols*. El nombre de lletres es denota* $q = |\mathbb{A}|$*.*

Exemples. Alguns exemples importants d'alfabets són:

- L'*alfabet llatí* estàndard de $q = 26$ lletres:

  ```
  a, b, c, d, e, f, g, h, i, j, k, l, m,
  n, o, p, q, r, s, t, u, v, w, x, y, z.
  ```

  La majoria de llengües dels països occidentals s'escriuen usant *variants* d'aquest alfabet.
- L'*alfabet grec* té $q = 24$ lletres:

  $$\boldsymbol{\alpha},\ \boldsymbol{\beta},\ \boldsymbol{\gamma},\ \boldsymbol{\delta},\ \boldsymbol{\epsilon},\ \boldsymbol{\zeta},\ \boldsymbol{\eta},\ \boldsymbol{\theta},\ \boldsymbol{\iota},\ \boldsymbol{\kappa},\ \boldsymbol{\lambda},\ \boldsymbol{\mu},\ \boldsymbol{\nu},\ \boldsymbol{\xi},\ \boldsymbol{o},\ \boldsymbol{\pi},\ \boldsymbol{\rho},\ \boldsymbol{\sigma},\ \boldsymbol{\tau},\ \boldsymbol{\upsilon},\ \boldsymbol{\phi},\ \boldsymbol{\chi},\ \boldsymbol{\psi},\ \boldsymbol{\omega}.$$

- El sistema de numeració *Hindú-Àrab* escriu els nombres naturals amb un alfabet de $q = 10$ lletres: els dígits decimals:

  $$\texttt{0, 1, 2, 3, 4, 5, 6, 7, 8, 9.}$$

  Usant també el signe menys $-$ per indicar nombres negatius, la barra / per indicar una fracció, i el *punt decimal* per separar la part entera de la part fraccionària es poden representar nombres enters, nombres racionals i (aproximacions de) *nombres reals*.

  Els nombres es poden escriure també en *binari* usant l'alfabet que només té els dos dígits `0` i `1`.

  El sistema de numeració *hexadecimal* fa servir un alfabet de 16 símbols: els deu primers dígits hexadecimals són els mateixos dígits decimals $\texttt{0}, \ldots, \texttt{9}$ i els sis següents, entre 10 i 15, es representen amb les primeres lletres de l'alfabet llatí: $\texttt{a} = 10$; $\texttt{b} = 11$; $\texttt{c} = 12$; $\texttt{d} = 13$; $\texttt{e} = 14$ i $\texttt{f} = 15$.

  En una base $q > 1$ qualsevol els nombres s'escriuen usant un alfabet de q dígits, que representen els nombres entre 0 i $q - 1$.

- L'*alfabet DNA* $\mathbb{A} = \{\texttt{A}, \texttt{C}, \texttt{T}, \texttt{G}\}$ té $q = 4$ lletres, que corresponen als quatre nucleòtids presents en el els *àcids desoxiribonucleics*: adenina, citosina, timina i guanina. Es fa servir en *genètica* per representar el *genoma* dels éssers vius.

- L'*alfabet Braille* consisteix en $q = 64$ símbols diferents que es formen marcant en relleu alguns dels sis punts en un caixetí rectangular de tres files i dues columnes.

- Una imatge digital s'escriu en l'*alfabet dels píxels*. Cada píxel es representa usant un nombre fixat d de bits que determinen el seu nivell de gris o bé les seves coordenades de color. Es tracta, per tant, d'un alfabet de $q = 2^d$ símbols. El nombre d s'anomena *profunditat* de color.

- El *so digital* s'obté amb un procés de mostreig i quantització de l'ona de so contínua. A intervals de temps regulars es mesura l'*ona sonora* i es representa el valor obtingut usant un nombre d de bits prefixat: la *profunditat* del so digital. Així, el so es converteix en una seqüència de símbols d'un alfabet de $q = 2^d$ elements.

- L'alfabet usat en les tecnologies digitals és l'*alfabet binari*

  $$\mathbb{A} = \{\texttt{0}, \texttt{1}\}, \qquad \text{de} \quad q = 2 \text{ lletres.}$$

  La *revolució digital* de les darreres dècades ha comportat que pràcticament tota la informació que es transmet i emmagatzema avui dia sigui informació digital: informació codificada amb una seqüència de dígits de l'alfabet binari.

  Tot i el risc de confondre amb la unitat de mesura de la informació, tal i com es defineix en Teoria de la Informació, és habitual anomenar *bits* les lletres `0` i `1` d'aquest alfabet.

Nombre de lletres. L'únic important d'un alfabet és el nombre de lletres $q = |\mathbb{A}|$, i no pas quines siguin les lletres concretes. Un alfabet de q lletres es diu *alfabet q-ari*.

L'alfabet de dues lletres és l'*alfabet binari*. Normalment s'agafen com a lletres d'aquest alfabet els dos nombres `0` i `1`. En sistemes de comunicacions digitals de vegades *es representen* els elements de l'alfabet binari amb els nombres reals 1 i -1 o amb els angles 0 i π. També es pot considerar l'alfabet binari $\{\mathtt{T}, \mathtt{F}\}$ format per les variables booleanes `T`="cert" i `F`="fals" de la lògica booleana, pels resultats "cara" i "creu" de llençar una moneda o per qualsevol altre parell de símbols diferents. Fer servir un alfabet o un altre no té cap rellevància des del punt de vista de la codificació.

Estructura algebraica. En moltes aplicacions es treballa amb un alfabet $\mathbb{A}$ que té estructura algebraica, de manera que les lletres es poden sumar i multiplicar entre elles. Els casos més importants són:

- Congruències $\mathbb{A} = \mathbb{Z}_q$: s'agafa com a alfabet q-ari el conjunt $\mathbb{A} = \mathbb{Z}_q$ de les classes de congruència mòdul q; les lletres són els nombres entre 0 i $q-1$, que es poden representar amb altres símbols si convé. En aquest conjunt hi ha definides la suma i el producte mòdul q, que tenen força bones propietats, però quan q no és un nombre primer hi ha elements diferents de zero que no tenen invers pel producte. Es fa servir aquest alfabet sobretot en codis detectors d'errors usats per afegir dígits de verificació (secció 1.6).
- Cossos finits $\mathbb{A} = \mathbb{F}_q$: quan el nombre de lletres és $q = p^e$ una potència d'un nombre primer existeixen conjunts finits $\mathbb{F}_q$ amb una suma i un producte que tenen les propietats d'un cos. En aquest cas es pot agafar com a alfabet aquest cos finit: $\mathbb{A} = \mathbb{F}_q$. Quan $q = p$ és directament un nombre primer el cos finit és $\mathbb{F}_p = \mathbb{Z}_p$. Quan és una potència d'exponent > 1 aleshores la construcció del cos finit $\mathbb{F}_q$ es fa de manera anàloga però fent congruències amb polinomis a coeficients a $\mathbb{F}_p$ en comptes de congruències amb nombres enters (secció 0.4). Gairebé tots els codis correctors d'errors s'agafen amb aquest alfabet: són subespais vectorials de l'espai vectorial $\mathbb{F}_q^n$ de les n-tuples d'elements del cos finit $\mathbb{F}_q$. Els casos més importants i més usats en tecnologies digitals són l'alfabet binari, amb nombre d'elements $q = 2 = 2^1$. i alfabets amb nombre d'elements que és $q = 2^e$ una potència de 2: per exemple, els *bytes* poden considerar-se com les $q = 256 = 2^8$ lletres d'un alfabet que és un cos finit, els *píxels* amb profunditat de color e, representats amb e bits, poden pensar-se com les lletres d'un alfabet de $q = 2^e$ elements, etc.

  En canvi als alfabets llatí o grec no es pot definir cap estructura de cos perquè el nombre de lletres (26 i 24, respectivament) no és una potència d'un nombre primer.

Seqüències. Les seqüències de lletres d'un alfabet $\mathbb{A}$ s'anomenen també *paraules*, *blocs*, *cadenes*, *textos*, *missatges*, etc. En termes matemàtics són elements del producte cartesià de diverses còpies de l'alfabet. S'escriuen com una seqüència o una n-tupla de lletres:

$$\boldsymbol{x} = \mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_n \qquad \text{o bé} \qquad \boldsymbol{x} = (\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_n) \qquad \text{amb} \quad \mathtt{a}_i \in \mathbb{A}.$$

Si $\mathbb{A} = \mathbb{F}$ és un cos finit les paraules es poden pensar com a vectors dels espais vectorials $\mathbb{F}^n$.

- Es denota $\mathbb{A}^n$ el conjunt de totes les seqüències de n lletres

$$\mathbb{A}^n = \{\boldsymbol{x} = \mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_n : \mathtt{a}_i \in \mathbb{A}\},$$

  que conté $|\mathbb{A}^n| = |\mathbb{A}|^n = q^n$ elements.
- Es considera també la paraula buida λ, que no té cap lletra, de manera que $\mathbb{A}^0 = \{\lambda\}$.
- Es denota $\mathbb{A}^*$ el conjunt infinit de les seqüències de qualsevol nombre de lletres

$$\mathbb{A}^* = \{\boldsymbol{x} = \mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_n : \mathtt{a}_i \in \mathbb{A}, n \geqslant 0\} = \bigcup_{n\geqslant 0} \mathbb{A}^n.$$

  Es denota $\mathbb{A}^+ = \mathbb{A}^* \setminus \{\lambda\}$ el conjunt de les paraules diferents de la paraula buida.
- En aquest conjunt $\mathbb{A}^*$ hi ha definida una operació: la *concatenació* (o composició, o juxtaposició) de seqüències, que consisteix simplement en posar una seqüència a continuació de l'altra:

$$\boldsymbol{xy} = \mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_n\mathtt{b}_1\mathtt{b}_2\cdots\mathtt{b}_m \qquad \text{si} \qquad \boldsymbol{x} = \mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_n, \quad \boldsymbol{y} = \mathtt{b}_1\mathtt{b}_2\cdots\mathtt{b}_m.$$

  La concatenació s'acostuma a denotar sense posar cap símbol, com el producte de nombres. De vegades, si convé fer èmfasi en el fet que es tracta d'una concatenació, es denotarà $\boldsymbol{x}\|\boldsymbol{y}$. En alguns llenguatges de programació, per exemple `Python`, la concatenació de strings es denota amb el signe $+$.

  La concatenació és associativa: $(\boldsymbol{xy})\boldsymbol{z} = \boldsymbol{x}(\boldsymbol{yz})$ però no commutativa: en general $\boldsymbol{yx} \neq \boldsymbol{xy}$. La seqüència buida λ és l'element neutre: $\lambda\boldsymbol{x} = \boldsymbol{x}\lambda = \boldsymbol{x}$.
- Donats subconjunts $\mathcal{S}, \mathcal{T} \subseteq \mathbb{A}^*$ es denota

$$\mathcal{S}\mathcal{T} = \{\boldsymbol{xy} : \boldsymbol{x} \in \mathcal{S}, \boldsymbol{y} \in \mathcal{T}\}$$

  el conjunt de totes les concatenacions de paraules de l'un i de l'altre.
- $\ell(\boldsymbol{x})$ és la *longitud* o *mida* d'una cadena $\boldsymbol{x} \in \mathbb{A}^*$: el nombre de lletres que la formen. És additiva respecte de la concatenació: $\ell(\boldsymbol{xy}) = \ell(\boldsymbol{x}) + \ell(\boldsymbol{y})$.
- Una expressió de la forma

$$\boldsymbol{x} = \boldsymbol{x}_1\boldsymbol{x}_2\cdots\boldsymbol{x}_r \quad (\text{o} \quad \boldsymbol{x} = \boldsymbol{x}_1\|\boldsymbol{x}_2\|\cdots\|\boldsymbol{x}_r), \qquad \boldsymbol{x}, \boldsymbol{x}_i \in \mathbb{A}^*$$

  és una *descomposició* de la cadena $\boldsymbol{x}$ com a concatenació de les cadenes $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_r$.
- Si $\mathcal{S}$ és un subconjunt de $\mathbb{A}^*$ es defineix el subconjunt $\mathcal{S}^* \subseteq \mathbb{A}^*$ com el conjunt de tots els elements de $\mathbb{A}^*$ que s'obtenen concatenant paraules de $\mathcal{S}$. És a dir, les seqüències de $\mathbb{A}^*$ que tenen una descomposició com a concatenació de paraules de $\mathcal{S}$:

$$\mathcal{S}^* = \{\boldsymbol{x} \in \mathbb{A}^* : \boldsymbol{x} = \boldsymbol{s}_1\boldsymbol{s}_2\cdots\boldsymbol{s}_r, \ \boldsymbol{s}_i \in \mathcal{S}, \ r \geqslant 0\}.$$

  Per exemple, si $\mathcal{S} = \mathbb{A}^n$ aleshores $\mathcal{S}^*$ està format per totes les seqüències de $\mathbb{A}^*$ que tenen longitud múltiple de n.

- De vegades les repeticions consecutives de lletres o paraules es denoten com una exponenciació. Per exemple,

$$\mathtt{a}^4\mathtt{ba}^2\mathtt{b}^3 = \mathtt{aaaabaabbb} \in \mathbb{A}^*, \qquad \mathtt{a}, \mathtt{b} \in \mathbb{A};$$

$$\boldsymbol{x}^3\boldsymbol{y}\boldsymbol{z}^2\boldsymbol{x}^2 = \boldsymbol{xxxyzzxx} \in \mathbb{A}^*, \qquad \boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z} \in \mathbb{A}^*.$$

- Donades seqüències $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{A}^*$ es diu que $\boldsymbol{x}$ és un *prefix* de $\boldsymbol{y}$ si existeix una paraula $\boldsymbol{z} \in \mathbb{A}^*$ tal que $\boldsymbol{xz} = \boldsymbol{y}$. Tota paraula és prefix d'ella mateixa; un prefix diferent de la paraula mateixa es diu *propi*. En particular tot prefix ha de complir $\ell(\boldsymbol{x}) \leqslant \ell(\boldsymbol{y})$. Per exemple, les paraules `p,pa,par,para,parau,paraul,paraula` són tots els prefixos de la paraula `paraula` de l'alfabet llatí.

  Es defineix anàlogament *sufix*.

  Es diu que $\boldsymbol{x}$ és un *factor* de $\boldsymbol{y}$ si existeixen paraules $\boldsymbol{u}, \boldsymbol{v} \in \mathbb{A}^*$ amb $\boldsymbol{y} = \boldsymbol{uxv}$.

- Donats subconjunts $\mathcal{S}, \mathcal{T} \subseteq \mathbb{A}^*$ es denota

$$\mathcal{S}^{-1}\mathcal{T} = \{\boldsymbol{z} \in \mathbb{A}^* : \boldsymbol{xz} = \boldsymbol{y} \text{ amb } \boldsymbol{x} \in \mathcal{S}, \boldsymbol{y} \in \mathcal{T}\}$$

  el conjunt dels sufixos de paraules de $\mathcal{T}$ tals que el prefix corresponent és una paraula de $\mathcal{S}$. Es denota de manera anàloga $\mathcal{S}^{-1}\mathcal{T}\mathcal{U}^{-1}$ el conjunt dels factors de paraules de $\mathcal{T}$ tals que el prefix corresponent és de $\mathcal{S}$ i el sufix és de $\mathcal{U}$.

- Es diu que un subconjunt de $\mathbb{A}^*$ és un *conjunt prefix* si cap paraula del subconjunt és prefix d'una altra. Per exemple, el conjunt $\{\mathtt{0}, \mathtt{10}, \mathtt{110}, \mathtt{111}\} \subset \{\mathtt{0}, \mathtt{1}\}^*$ és prefix i tot subconjunt de $\mathbb{A}^n$ (format per paraules de la mateixa mida) també és prefix.

  Es defineix anàlogament subconjunt sufix. Un subconjunt *bifix* és un subconjunt en què cap paraula no és ni prefix ni sufix d'una altra.

- Ocasionalment pot ser necessari treballar amb elements del conjunt

$$\mathbb{A}^\infty = \{\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3\cdots : \mathtt{a}_i \in \mathbb{A}\}$$

  de les successions infinites $(\mathtt{a}_i)_{i\geqslant 1}$ de lletres de $\mathbb{A}$. Observi's que el conjunt $\mathbb{A}^*$ és infinit numerable, però, en canvi, el conjunt $\mathbb{A}^\infty$ és infinit no numerable.

  També es poden considerar successions infinites per tots dos costats:

$$\mathbb{A}^{\mathbb{Z}} = \{\cdots \mathtt{a}_{-2}\mathtt{a}_{-1}\mathtt{a}_0\mathtt{a}_1\mathtt{a}_2\cdots : \mathtt{a}_i \in \mathbb{A}, i \in \mathbb{Z}\}.$$

Notació amb subíndexos. En estudiar alfabets i paraules els subíndexos es fan servir per a dues coses:

- Enumerar les lletres de l'alfabet: quan es posa $\mathbb{A} = \{\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_q\}$ aleshores $\mathtt{a}_i$ denota la lletra i-èsima de les de l'alfabet, i l'índex varia entre 1 i $q = |\mathbb{A}|$. Quan $i \neq j$ les lletres $\mathtt{a}_i$ i $\mathtt{a}_j$ són diferents.

- Identificar les components d'un vector o les lletres d'una paraula: quan s'escriu $\boldsymbol{a} = (\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_n)$ o $\boldsymbol{a} = \mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_n$ per a un $\boldsymbol{a} \in \mathbb{A}^n$ aleshores $\mathtt{a}_i$ representa la lletra de l'alfabet que està a la coordenada i-èsima de la n-tupla $\boldsymbol{a}$, o la lletra i-èsima de la paraula $\boldsymbol{a}$. Aquests subíndexos varien entre 1 i n. En aquest cas diferents components del vector poden ser la mateixa lletra: pot ser que $\mathtt{a}_i = \mathtt{a}_j$ per a índexs diferents; fins i tot totes les lletres $\mathtt{a}_i$ podrien ser iguals.

Aquests dos usos no s'han de confondre. En general el context permet saber de quin dels dos es tracta. Si alguna vegada s'han de fer servir tots dos alhora es pot usar la notació més recarregada $\boldsymbol{a} = \mathtt{a}_{i_1}\mathtt{a}_{i_2}\cdots\mathtt{a}_{i_n}$ amb subíndexos dobles $i_j \in \{1, 2, \ldots, q\}$ per a $j = 1, 2, \ldots, n$ que indiquen que la lletra j-èsima de la paraula $\boldsymbol{a}$ és la lletra i_j-èsima de l'alfabet $\mathbb{A}$, segons l'enumeració prefixada d'aquest alfabet.

Definició 1.2 (Codi)**.** *Un subconjunt no buit $\mathcal{C}$ de $\mathbb{A}^*$ és un codi si tot element de $\mathcal{C}^* \subseteq \mathbb{A}^*$ descompon* de manera única *com a concatenació d'elements de $\mathcal{C}$.*

Els elements de $\mathcal{C}$ es diuen paraules codi.

És a dir, el subconjunt $\mathcal{C} \subset \mathbb{A}^*$ és un codi si, sempre que es concatenen dues famílies de paraules del conjunt $\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_r \in \mathcal{C}$ i $\boldsymbol{c}'_1, \boldsymbol{c}'_2, \ldots, \boldsymbol{c}'_s \in \mathcal{C}$ i s'obté com a resultat la mateixa seqüència de $\mathbb{A}^*$, aleshores necessàriament totes dues famílies coincideixen:

$$\boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_r = \boldsymbol{c}'_1\boldsymbol{c}'_2\cdots\boldsymbol{c}'_s \quad \Leftrightarrow \quad r = s \quad \text{i} \quad \boldsymbol{c}_i = \boldsymbol{c}'_i \ \forall i.$$

Exemple 1.3 (Codi de bloc)**.** *Tot subconjunt $\mathcal{C} \subseteq \mathbb{A}^n$, format per paraules totes de la mateixa mida n, és un codi.*

Prova: Tot element de $\mathcal{C}^*$ és una seqüència de longitud múltiple de n i l'única descomposició possible com a concatenació de paraules de $\mathcal{C}$ s'obté trencant-la en trossos de mida n. □

Aquests codis s'anomenen *codis de bloc* o *codis de longitud fixa*. Són els que es fan servir per a la detecció i correcció d'errors. Els codis en què no s'exigeix que totes les paraules tinguin necessàriament la mateixa mida s'anomenen *codis de longitud variable* i es fan servir en la compressió de dades.

Exemple 1.4 (Codi prefix)**.** *Tot subconjunt prefix de $\mathbb{A}^*$ és un codi. S'anomena codi prefix o també* codi instantani.

Prova: Sigui $\mathcal{C} \subset \mathbb{A}^*$ un subconjunt prefix. Es parteix d'una doble descomposició d'un element $\boldsymbol{x} \in \mathbb{A}^*$ com a concatenació de paraules de $\mathcal{C}$

$$\boldsymbol{x} = \boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_r = \boldsymbol{c}'_1\boldsymbol{c}'_2\cdots\boldsymbol{c}'_s, \qquad \text{amb} \quad \boldsymbol{c}_i, \boldsymbol{c}'_j \in \mathcal{C}.$$

Si fos $\ell(\boldsymbol{c}_1) < \ell(\boldsymbol{c}'_1)$ la paraula $\boldsymbol{c}_1$ seria prefix de la paraula $\boldsymbol{c}'_1$, en contradicció amb la hipòtesi que el codi és prefix. Anàlogament tampoc pot passar que $\ell(\boldsymbol{c}_1) > \ell(\boldsymbol{c}'_1)$. Per tant ha de ser $\ell(\boldsymbol{c}_1) = \ell(\boldsymbol{c}'_1)$. Sigui n la longitud comú d'aquestes paraules. Com que tant $\boldsymbol{c}_1$ com $\boldsymbol{c}'_1$ són les primeres n lletres de la cadena $\boldsymbol{x}$ han de ser iguals: $\boldsymbol{c}_1 = \boldsymbol{c}'_1$. Eliminant totes dues paraules del començament de $\boldsymbol{x}$ s'obté una igualtat entre les concatenacions de les altres. Per recurrència es dedueix que totes dues descomposicions són iguals. □

Els codis de bloc són codis prefixos.

El nom de *codi instantani* per als codis prefixos es refereix a què les seqüències de $\mathcal{C}^* \subseteq \mathbb{A}^*$ es poden descompondre en paraules de $\mathcal{C}$ a mesura que es van llegint les lletres que en formen part. Donada una seqüència $\boldsymbol{x} = \mathtt{a}_1\mathtt{a}_2\mathtt{a}_3\mathtt{a}_4\mathtt{a}_5\cdots\mathtt{a}_k \in \mathbb{A}^*$ que sigui concatenació de paraules codi: $\boldsymbol{x} = \boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_r \in \mathcal{C}^*$ es poden anar obtenint les paraules $\boldsymbol{c}_i$ de la manera següent. S'agafen les paraules formades per les primeres lletres $\mathtt{a}_1$, $\mathtt{a}_1\mathtt{a}_2$, $\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3$, ... de la seqüència $\boldsymbol{x}$ fins a trobar una paraula que sigui del codi $\mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_n \in \mathcal{C}$. La condició de subconjunt prefix assegura que aquesta paraula necessàriament ha de ser $\boldsymbol{c}_1$. A continuació s'eliminen aquestes lletres del començament de seqüència i es continua de la mateixa manera amb la resta $\mathtt{a}_{n+1}\mathtt{a}_{n+2}\cdots\mathtt{a}_k$. D'aquesta manera es pot anar descodificant sobre la marxa sense haver de mirar la seqüència $\boldsymbol{x}$ tota sencera per poder començar a fer la descodificació.

No tots els codis són prefixos:

Exemple 1.5. *Els conjunts següents són codis no prefixos:*

1. $\mathcal{C} = \{\mathtt{1}, \mathtt{10}, \mathtt{00}\} \subseteq \{\mathtt{0},\mathtt{1}\}^*$;
2. $\mathcal{C} = \{\mathtt{0}, \mathtt{010}\} \subset \{\mathtt{0},\mathtt{1}\}^*$.

Prova: S'ha de comprovar que són codis ja que clarament són conjunts no prefixos.

1. És un subconjunt en què cap paraula és sufix d'una altra: un "codi sufix" que admet una descodificació instantània començant pel final. Usant aquest codi, si s'ha de descompondre una seqüència $\boldsymbol{x} = \boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_k \in \mathcal{C}^* \subseteq \{\mathtt{0},\mathtt{1}\}^*$ que comença amb un $\mathtt{1}$ seguit de molts zeros, de la forma $\boldsymbol{x} = \mathtt{10000000000}\ldots$, no es pot decidir si la primera paraula codi $\boldsymbol{c}_1$ és $\mathtt{1}$ o és $\mathtt{10}$ fins que es trobi un altre $\mathtt{1}$ a $\boldsymbol{x}$ o s'arribi al final. Si entre el primer $\mathtt{1}$ i el següent o el final de $\boldsymbol{x}$ hi ha un nombre parell de zeros aleshores la primera paraula és $\boldsymbol{c}_1 = \mathtt{1}$; si, en canvi, n'hi ha un nombre senar és $\boldsymbol{c}_1 = \mathtt{10}$.
2. L'única manera de descompondre una seqüència binària de $\mathcal{C}^* \subset \{\mathtt{0},\mathtt{1}\}^*$ en paraules codi és la següent: cada $\mathtt{1}$ que aparegui a la seqüència ha d'estar entremig de dos $\mathtt{0}$ i indica la presència de la paraula $\mathtt{010}$. Un cop identificades totes les aparicions de $\mathtt{010}$ tots els demés símbols de la seqüència han de ser per força zeros, i cadascun d'ells correspon a l'altra paraula $\mathtt{0} \in \mathcal{C}$. □

Exemple 1.6. *Els conjunts següents no són codis:*

1. $\mathcal{C} = \{\mathtt{010}, \mathtt{10}, \mathtt{101}\} \subset \{\mathtt{0},\mathtt{1}\}^*$;
2. $\mathcal{C} = \{\mathtt{000}, \mathtt{00000}\} \subset \{\mathtt{0},\mathtt{1}\}^*$;
3. $\mathcal{C} = \{\mathtt{a}, \mathtt{c}, \mathtt{ad}, \mathtt{abb}, \mathtt{bad}, \mathtt{deb}, \mathtt{bbcde}\} \subset \mathbb{A}^*$ *amb* $\mathbb{A} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}, \mathtt{e}\}$.

Prova: Per veure-ho s'ha de trobar una seqüència de $\mathcal{C}^*$ amb una doble descomposició com a concatenació de paraules codi.

1. Es té una doble descomposició de la paraula $\mathtt{101010}$ en paraules codi

$$\mathtt{101}\|\mathtt{010} = \mathtt{101010} = \mathtt{10}\|\mathtt{10}\|\mathtt{10}.$$

2. La seqüència formada per quinze zeros seguits té una doble descomposició com a concatenació de cinc vegades `000` o de tres vegades `00000`. També la seqüència de vuit zeros té dues descomposicions diferents com a concatenació de les dues paraules del conjunt, agafant-les en els dos ordres possibles.
3. Es denoten $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4, \boldsymbol{x}_5, \boldsymbol{x}_6, \boldsymbol{x}_7\} = \{\mathtt{a}, \mathtt{c}, \mathtt{ad}, \mathtt{abb}, \mathtt{bad}, \mathtt{deb}, \mathtt{bbcde}\}$ les paraules del conjunt. No és un codi ja que es té una doble descomposició:

$$\boldsymbol{x}_1\boldsymbol{x}_7\boldsymbol{x}_5 = \mathtt{a}\|\mathtt{bbcde}\|\mathtt{bad} = \mathtt{abbcdebad} = \mathtt{abb}\|\mathtt{c}\|\mathtt{deb}\|\mathtt{ad} = \boldsymbol{x}_4\boldsymbol{x}_2\boldsymbol{x}_6\boldsymbol{x}_3.$$

de la paraula $\mathtt{abbcdebad} \in \mathcal{C}^*$ com a concatenació de paraules de $\mathcal{C}$. □

Tot i que en la majoria d'aplicacions interessen sobretot els codis que són conjunts finits, i en aquestes notes moltes vegades se suposarà sense dir-ho que un codi és finit, també es consideren de vegades codis infinits. Per exemple:

Exemple 1.7 (Codificació unària)**.** *El el conjunt infinit format per les paraules que comencen amb zeros i acaben amb un u és un codi binari prefix:*

$$\mathcal{C} = \{\mathtt{1}, \mathtt{01}, \mathtt{001}, \mathtt{0001}, \mathtt{00001}, \dots\} \subset \{\mathtt{0}, \mathtt{1}\}^*.$$

És clar que tot subconjunt no buit d'un codi q-ari també és un codi q-ari. Hi ha codis que no estan estrictament continguts en cap altre codi. Són els:

Definició 1.8 (Codis maximals)**.** *Un codi q-ari $\mathcal{C} \subseteq \mathbb{A}^*$ és un codi maximal si per a tot altre codi $\mathcal{C}' \subseteq \mathbb{A}^*$ es té*

$$\mathcal{C} \subseteq \mathcal{C}' \quad \Rightarrow \quad \mathcal{C}' = \mathcal{C}.$$

Exemple 1.9. *Els codis $\mathcal{C} = \mathbb{A}^n$ són maximals.*

Prova: Sigui $\mathcal{C} \subseteq \mathcal{C}' \subseteq \mathbb{A}^*$. Suposi's que $\mathcal{C}' \neq \mathcal{C}$. Sigui $\boldsymbol{c}' \in \mathcal{C}'$ una paraula amb $k = \ell(\boldsymbol{c}') \neq n$, que existeix ja que $\mathcal{C}$ conté totes les paraules de longitud n. Aleshores es té una doble descomposició

$$(\boldsymbol{c}')^n = \boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_k$$

de la seqüència de kl lletres obtinguda concatenant $\boldsymbol{c}'$ amb ella mateixa n vegades: com que aquesta seqüència té longitud múltiple de n és també la concatenació de n paraules del codi $\mathcal{C}$. □

Com saber si un subconjunt de $\mathbb{A}^*$ és un codi. En general no hi ha una manera senzilla de veure immediatament si un subconjunt de $\mathbb{A}^*$ és o no és un codi. Hi ha una manera de decidir això, que es coneix com l'*algorisme de Sardinas–Patterson*:

Teorema 1.10 (Sardinas-Patterson)**.** *Donat un subconjunt no buit $\mathcal{S} \subset \mathbb{A}^*$ es consideren els conjunts*

$$\mathcal{U}_1 = \mathcal{S}^{-1}\mathcal{S} \setminus \{\lambda\}, \qquad \mathcal{U}_{n+1} = \mathcal{S}^{-1}\mathcal{U}_n \cup \mathcal{U}_n^{-1}\mathcal{S}, \quad \text{per a} \quad n \geqslant 1.$$

Aleshores $\mathcal{S}$ és un codi si, i només si, cap dels conjunts $\mathcal{U}_n$ conté la paraula buida.

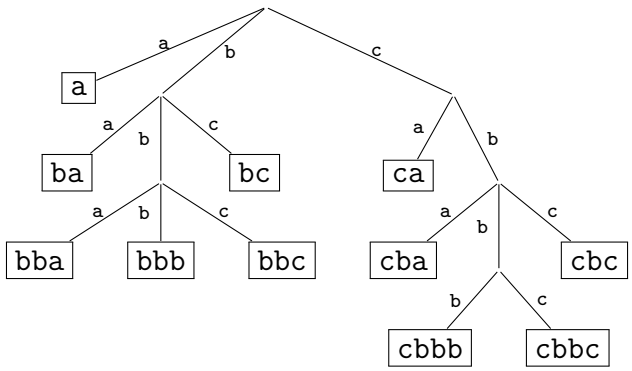
PROVA: Veure [7, Theorem 3.1]. □

En el cas que $\mathcal{S}$ és un conjunt finit aleshores hi ha només un nombre finit de conjunts $\mathcal{U}_n$ possibles, ja que tots els elements dels $\mathcal{U}_n$ són factors d'elements de $\mathcal{S}$, dels quals n'hi ha només un nombre finit. Si hi ha una repetició $\mathcal{U}_n = \mathcal{U}_{n+k}$ aleshores en endavant aquests conjunts es van repetint periòdicament: $\mathcal{U}_{n+k+r} = \mathcal{U}_{n+r}$ per a tot $r \geqslant 0$. Per tant, una de dues: o bé es troba la paraula buida en algun dels conjunts λ, i per tant $\mathcal{S}$ no és un codi, o bé s'arriba a una repetició sense haver-la trobat, i per tant no es pot trobar mai més, i $\mathcal{S}$ sí que és un codi.

En la demostració del teorema es veu una manera de trobar una doble descomposició d'una seqüència de $\mathbb{A}^*$ com a concatenació de paraules de $\mathcal{S}$ a partir d'haver trobat λ en un dels conjunts $\mathcal{U}_n$.

Problemes

1.1. Considereu la correspondència entre codis q-aris prefixos i arbres q-aris (arbres amb una única arrel i tal que tot node pare té com a molt q nodes fills) a partir de l'exemple següent: el codi ternari $\mathcal{C} = \{\mathtt{a}, \mathtt{ba}, \mathtt{bc}, \mathtt{ca}, \mathtt{bba}, \mathtt{bbb}, \mathtt{bbc}, \mathtt{cba}, \mathtt{cbc}, \mathtt{cbbb}, \mathtt{cbbc}\}$ es representa amb l'arbre següent:



Representeu en forma d'arbre el codi binari prefix

$$\mathcal{C} = \{\mathtt{00}, \mathtt{010}, \mathtt{011}, \mathtt{1000}, \mathtt{1001}, \mathtt{1011}, \mathtt{110}, \mathtt{111}\}$$

i digueu com serien els arbres que representen el codi de bloc $\mathcal{C} = \mathbb{A}^n$, el codi de repetició $\mathcal{C} \subset \mathbb{A}^n$ format per les paraules que tenen la mateixa lletra repetida n vegades, i el codi binari parell $\mathcal{C} \subset \{\mathtt{0}, \mathtt{1}\}^n$ format per les paraules que tenen un nombre parell de uns.

1.2. Demostreu que si a un codi prefix $\mathcal{C} \subset \mathbb{A}^*$ de M paraules s'agafa una paraula qualsevol $\boldsymbol{c} \in \mathcal{C}$ i se li afegeixen r lletres diferents $\mathtt{a}_1, \ldots, \mathtt{a}_r \in \mathbb{A}$ s'obté com a resultat un conjunt

$$\big(\mathcal{C} \smallsetminus \{\boldsymbol{c}\}\big) \sqcup \big\{\boldsymbol{c}\|\mathtt{a}_1, \boldsymbol{c}\|\mathtt{a}_2, \ldots, \boldsymbol{c}\|\mathtt{a}_r\big\} \subset \mathbb{A}^*$$

de $M - 1 + r$ paraules que també és un codi prefix.

Interpreteu-ho en termes dels arbres del problema **1.1**.

1.2 Codificació

Definició 1.11 (Codificació)**.** *Una* codificació *dels elements d'un conjunt finit $\mathcal{M}$ amb paraules d'un alfabet $\mathbb{A}$ és una aplicació injectiva* $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$ *que envia cada element* $\mathtt{m} \in \mathcal{M}$ *a una paraula* $\mathrm{enc}(\mathtt{m}) \in \mathbb{A}^*$ *tal que la imatge* $\mathcal{C} := \mathrm{enc}(\mathcal{M}) = \mathrm{Im}(\mathrm{enc}) \subset \mathbb{A}$ *sigui un codi.*

La codificació s'estén de manera natural a una aplicació $\mathrm{enc}^*\colon \mathcal{M}^* \to \mathcal{C}^* \subseteq \mathbb{A}^*$ *que envia cada seqüència* $\boldsymbol{m} = \mathtt{m}_1\mathtt{m}_2\cdots\mathtt{m}_r \in \mathcal{M}^*$ *a la concatenació de les imatges corresponents:*

$$\mathrm{enc}^*(\boldsymbol{m}) = \mathrm{enc}^*(\mathtt{m}_1\mathtt{m}_2\cdots\mathtt{m}_r) = \mathrm{enc}(\mathtt{m}_1)\|\,\mathrm{enc}(\mathtt{m}_2)\|\cdots\|\,\mathrm{enc}(\mathtt{m}_r) = \boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_r.$$

Així, una codificació estableix una bijecció entre els elements d'un conjunt $\mathcal{M}$ i els elements d'un codi $\mathcal{C} \subset \mathbb{A}^*$ sobre l'alfabet $\mathbb{A}$. Es denotarà $\mathrm{dec}\colon \mathcal{C} \to \mathcal{M}$ l'aplicació inversa: $\mathrm{dec} = \mathrm{enc}^{-1}$.

La codificació $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$ transforma elements de $\mathcal{M}$ en seqüències de lletres de l'alfabet $\mathbb{A}$, i l'extensió $\mathrm{enc}^*\colon \mathcal{M}^* \to \mathbb{A}^*$ fa el mateix amb elements de $\mathcal{M}^*$. La imatge de l'extensió és el subconjunt $\mathrm{enc}^*(\mathcal{M}^*) = \mathcal{C}^* \subseteq \mathbb{A}^*$.

Els elements del conjunt $\mathcal{M}$ o les cadenes de $\mathcal{M}^*$ juguen aquí el paper de la informació: s'interpreten com a *missatges* que s'han de codificar en forma de seqüències de lletres de l'alfabet $\mathbb{A}$ (típicament l'alfabet binari) per al seu emmagatzematge en un dispositiu o transmissió a través d'un canal de comunicacions.

De vegades aquest conjunt $\mathcal{M}$ de missatges és també un subconjunt de $\mathbb{A}^*$, de manera que la codificació transforma unes cadenes de $\mathbb{A}^*$ en altres cadenes de $\mathbb{A}^*$. Per exemple, un codi de Huffman per comprimir un text en català transforma octets $\mathtt{m} \in \mathcal{M} = \{\mathtt{0},\mathtt{1}\}^8$, que representen les lletres en codi ASCII, en una paraula binària $\mathrm{enc}(\mathtt{m}) \in \{\mathtt{0},\mathtt{1}\}^*$ de longitud variable, més curta o més llarga segons la freqüència amb què $\mathtt{m}$ apareix en el text a comprimir. Un codi corrector d'errors transforma blocs $\mathtt{m} \in \mathcal{M} = \{\mathtt{0},\mathtt{1}\}^k$ de mida k en blocs més llargs $\mathrm{enc}(\mathtt{m}) \in \{\mathtt{0},\mathtt{1}\}^n$ de mida $n \geqslant k$, de tal manera que la redundància afegida en augmentar la mida dels blocs es pot usar per corregir errors de transmissió. L'estàndard de xifrat de clau secreta *AES* transforma cadenes binàries de longitud 128 en unes altres cadenes de la mateixa longitud usant una clau.

En teoria de codis s'acostuma a fer servir el terme *codi* tant per indicar un subconjunt $\mathcal{C} \subseteq \mathbb{A}^*$ que és un codi com per indicar una codificació $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$ d'elements d'un conjunt $\mathcal{M}$ usant paraules del codi $\mathcal{C} = \mathrm{enc}(\mathcal{M})$.

Descodificació única. Que l'aplicació enc sigui injectiva i que la imatge $\mathrm{enc}(\mathcal{M})$ sigui un codi $\mathcal{C} \subseteq \mathbb{A}^*$ garanteixen que a partir de la seqüència que dona la codificació $\mathrm{enc}^*(\boldsymbol{m}) \in \mathbb{A}^*$ es pot recuperar la seqüència $\mathtt{m}_1\mathtt{m}_2\cdots\mathtt{m}_r$ d'elements de $\mathcal{M}$ que formen el missatge $\boldsymbol{m}$:

Proposició 1.12. *Per a tota codificació* $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$ *existeix una (única) aplicació de descodificació* $\mathrm{dec}^*\colon \mathcal{C}^* \to \mathcal{M}^*$ *tal que* $\mathrm{dec}^*\big(\mathrm{enc}^*(\boldsymbol{m})\big) = \boldsymbol{m}$ *per a tot* $\boldsymbol{m} \in \mathcal{M}^*$.

Prova: Sigui $\mathcal{C} = \mathrm{enc}(\mathcal{M})$ el codi imatge i sigui $\mathrm{dec}\colon \mathcal{C} \to \mathcal{M}$ l'aplicació de descodificació d'elements de $\mathcal{M}$: $\mathrm{dec} = \mathrm{enc}^{-1}$ amb $\mathrm{enc}\colon \mathcal{M} \to \mathcal{C}$ bijectiva.

Donada una paraula $\boldsymbol{c} \in \mathcal{C}^* \subseteq \mathbb{A}^*$, per ser $\mathcal{C}$ un codi existeix una única descomposició $\boldsymbol{c} = \boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_r$ amb $\boldsymbol{c}_i \in \mathcal{C}$. Aleshores es defineix $\mathrm{dec}^*(\boldsymbol{c}) = \mathrm{dec}(\boldsymbol{c}_1)\|\,\mathrm{dec}(\boldsymbol{c}_2)\|\cdots\|\,\mathrm{dec}(\boldsymbol{c}_r)$.

Aquesta aplicació és una inversa per l'esquerra de enc^*: $\mathrm{dec}^* \circ \mathrm{enc}^* = \mathrm{Id}_{\mathcal{M}^*}$, tal com diu l'enunciat. En efecte, donada una paraula $\boldsymbol{m} = \mathtt{m}_1\mathtt{m}_2\cdots\mathtt{m}_r \in \mathcal{M}^*$ la seva imatge és $\boldsymbol{c} = \boldsymbol{c}_1\boldsymbol{c}_2\cdots\boldsymbol{c}_r \in \mathcal{C}^*$ amb $\boldsymbol{c}_i = \mathrm{enc}(\mathtt{m}_i)$ i en calcular $\mathrm{dec}^*(\boldsymbol{c})$ de la manera indicada es recupera el missatge $\boldsymbol{m}$ que s'havia codificat. □

En alguns textos, per exemple a [4] o a l'entrada de la *wikipèdia*, s'anomena *codificació* a una aplicació qualsevol $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$. Quan aquesta aplicació és injectiva se li diu *codificació no singular* i quan existeix una aplicació de descodificació que permet recuperar $\boldsymbol{m}$ a partir de $\mathrm{enc}^*(\boldsymbol{m})$ se li diu *de descodificació única*. En realitat, les "codificacions" sense descodificació única no es fan servir mai a la pràctica, i tenen un interès purament teòric, relacionat amb demostracions del teorema de codificació de canal. Aquí se suposarà per defecte que totes les codificacions són sempre de descodificació única.

El lema següent dona caracteritzacions d'aquesta condició:

Lema 1.13. *Sigui* $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$ *una aplicació qualsevol amb extensió* $\mathrm{enc}^*\colon \mathcal{M}^* \to \mathbb{A}^*$. *Sigui* $\mathcal{C} = \mathrm{enc}(\mathcal{M}) \subset \mathbb{A}^*$ *la seva imatge. Les condicions següents són equivalents:*

1. *Existeix una aplicació de descodificació* $\mathrm{dec}^*\colon \mathcal{C}^* \to \mathcal{M}^*$ *tal que* $\mathrm{dec}^*\big(\mathrm{enc}^*(\boldsymbol{m})\big) = \boldsymbol{m}$ *per a tot* $\boldsymbol{m} \in \mathcal{M}^*$.
2. *L'aplicació* enc^* *és injectiva.*
3. *L'aplicació* enc *és injectiva i la seva imatge* $\mathcal{C} = \mathrm{enc}(\mathcal{M}) \subset \mathbb{A}^*$ *és un codi.*

Prova: Observi's que en l'enunciat s'agafa una aplicació $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$ qualsevol: no es demana que sigui injectiva ni s'imposa cap propietat a la seva imatge.

- (1) ⇒ (2). Donada una aplicació $f\colon X \to Y$ entre conjunts, existeix una aplicació $g\colon Y \to X$ tal que $g \circ f = \mathrm{Id}_X$ si, i només si, f és injectiva. En efecte, si f es injectiva es defineix g posant $g(y) = x$ si $f(x) = y$ i donant-li un valor qualsevol si y no és de la imatge de f; recíprocament, si existeix una aplicació g amb $g \circ f = \mathrm{Id}_X$ aleshores $f(x) = f(y) \Rightarrow x = g\big(f(x)\big) = g\big(f(y)\big) = y$.
- (2) ⇒ (3). Si enc^* és injectiva aleshores enc, que és la seva restricció al subconjunt $\mathcal{M} \subset \mathcal{M}^*$, també ho és. Per tant té inversa $\mathrm{dec}\colon \mathcal{C} \to \mathcal{M}$. Suposi's que $\mathcal{C}$ no fos un codi. Aleshores existeix una cadena a $\mathcal{C}^*$ amb doble descomposició: $\boldsymbol{c} = \boldsymbol{c}_1\cdots\boldsymbol{c}_r = \boldsymbol{c}'_1\cdots\boldsymbol{c}'_s$ amb $\boldsymbol{c}_i, \boldsymbol{c}_j \in \mathcal{C}$. En aquest cas els missatges $\boldsymbol{m} = \mathtt{m}_1\cdots\mathtt{m}_r$ i $\boldsymbol{m}' = \mathtt{m}'_1\cdots\mathtt{m}'_s$ amb lletres $\mathtt{m}_i = \mathrm{dec}(\boldsymbol{c}_i)$ i $\mathtt{m}'_j = \mathrm{dec}(\boldsymbol{c}'_j)$ són diferents però tenen la mateixa imatge $\mathrm{enc}^*(\boldsymbol{m}) = \mathrm{enc}(\boldsymbol{m}') = \boldsymbol{c}$, i això contradiu la injectivitat de enc^*.
- (3) ⇒ (1). La condició (3) és la definició de codificació i aquesta implicació és la proposició 1.12. □

Exemples. A continuació es donen alguns exemples de codis usats per escriure la informació en un format adequat per a la transmissió i processat en diferents sistemes de comunicació i en tecnologies digitals o per millorar l'eficiència i la fiabilitat de la transmissió:

- El *codi ASCII* es crea als anys 60 com a estàndard per processar informació en computadors i comunicacions digitals. És una *codificació* sobre l'alfabet binari $\mathbb{A} = \{0, 1\}$ d'un conjunt $\mathcal{M}$ de 128 símbols diferents, que inclouen lletres majúscules i minúscules, dígits decimals, signes de puntuació i caràcters de control. El codi més bàsic és $\mathcal{C} = \text{enc}(\mathcal{M}) = \{0, 1\}^7$, format per totes les paraules de 7 bits. També s'usa la versió estesa amb un bit de paritat $\mathcal{C} = \text{enc}(\mathcal{M}) \subset \{0, 1\}^8$ format per les 128 paraules de 8 bits que tenen un nombre parell d'uns. Aquesta versió permet detectar un error en cada paraula.
- ASCII ha evolucionat cap a *Unicode*, que permet codificar una gran quantitat de caràcters, signes i símbols. Inclou els alfabets amb què s'escriuen totes les llengües del món i molts símbols que tenen un significat en determinades disciplines o aplicacions. És un codi binari format per paraules de mida 8, 16, 24 o 32. Per exemple, l'estàndard *UTF-8* és un codi prefix que permet codificar 1 112 064 símbols usant paraules de 8 bits (que comencen per `0`), de 16 bits (que comencen per `110`), de 24 bits (que comencen per `1110`) i de 32 bits (que comencen per `11110`). En totes les paraules de més de 8 bits els bytes que segueixen al byte inicial comencen per `10` per permetre la sincronització. Amb aquestes restriccions sobre els prefixos dels bytes s'obtindria un codi de $2^7 + 2^{11} + 2^{16} + 2^{21} = 2\,164\,864$ paraules, però no totes es fan servir.
- El *codi de Baudot* és un predecessor del codi ASCII creat a finals del segle XIX. Va ser usat per enviar informació a traves de línies de telègraf. Codifica un conjunt $\mathcal{M}$ de 32 símbols (lletres, signes de puntuació i caràcters de control) amb paraules binàries de mida 5. En la pràctica es treballava amb *cintes de paper* perforades que es podien llegir a gran velocitat amb dispositius mecànics, els quals transformaven les dades escrites a la cinta en senyals elèctrics. Això minimitzava el temps d'ús efectiu de la línia telegràfica comparat amb la introducció manual de les dades per part del telegrafista.
- El *codi Morse* va ser creat per enviar informació de text a través de línies telegràfiques. Codifica un conjunt $\mathcal{M}$ de 36 caràcters format per les 26 lletres de l'alfabet llatí i els 10 dígits decimals. L'alfabet $\mathbb{A}$ és ternari i consisteix en "punt", "ratlla" i "espai". És un codi de longitud variable en què es té en compte la freqüència de les lletres en l'anglès per assignar la paraula codi a cadascuna. Per exemple, la lletra més frequent, que és la `e`, es codifica amb la paraula més curta: un punt.

  Les paraules codi de Morse no són un conjunt prefix. En realitat, per poder considerar Morse com un codi amb descodificació única s'ha de tenir en compte que per a codificar un text escrit en alfabet llatí (i dígits) dues lletres consecutives dins d'una mateixa paraula se separen amb tres espais i dues paraules se separen amb set espais.
- El codi *Base-64* es fa servir per transmetre informació binària a través de canals de comunicacions que no interpreten correctament els bytes no imprimibles del codi ASCII. Codifica elements del conjunt $\mathcal{M} = \{0, 1\}^6$ usant un alfabet $\mathbb{A}$ format per 64 caràcters: les 26 lletres majúscules, les 26 lletres minúscules, els 10 dígits decimals i els dos símbols `/` i `+`. Per transmetre els elements de $\mathbb{A}$ es codifiquen en ASCII convertint-los en octets. El resultat és que cada 3 bytes originals donen lloc a $24 = 3 \times 8$ bits, que es descomponen en quatre elements de $\mathcal{M}$ ($24 = 4 \times 6$ bits), cadascun els quals es

codifica amb un caràcter ASCII imprimible. Això incrementa en un terç la mida de les seqüències a transmetre (cada tres bytes es converteixen en quatre) però a canvi no s'han d'enviar bytes que el canal pugui interpretar malament.

- El codi de Hamming de longitud 7 codifica blocs binaris de longitud 4 en blocs binaris de longitud 7 afegint-los tres bits de paritat: a cada bloc $\mathtt{b}_1\mathtt{b}_2\mathtt{b}_3\mathtt{b}_4$ se li afegeixen els tres bits $\mathtt{b}_2+\mathtt{b}_3+\mathtt{b}_4$, $\mathtt{b}_1+\mathtt{b}_3+\mathtt{b}_4$ i $\mathtt{b}_1+\mathtt{b}_2+\mathtt{b}_4$. A canvi de l'increment en la mida de les dades codificades aquest codi permet corregir tots els errors que afectin només un bit de cada paraula.
- L'estàndard grup 4 de codificació de FAX codifica les seqüències binàries generades en l'escanejat d'un full amb dos codis prefixos que es fan servir per codificar les cadenes binàries de zeros o uns seguits, respectivament. Com que en el resultat d'escanejar un full de text es crea una seqüència binària amb moltes cadenes llargues de zeros i uns seguits, aquesta codificació dona com a resultat una compressió de les dades.

Problemes

1.3. *Run-length encoding*. Proposeu una codificació binària $\mathrm{enc}\colon \{\mathtt{0},\mathtt{1}\}^* \to \{\mathtt{0},\mathtt{1}\}^*$ que representi les seqüències de zeros i uns consecutius amb el nombre que n'hi ha.

1.4. Doneu una codificació binària $\mathrm{enc}\colon \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\} \to \{\mathtt{0},\mathtt{1}\}^*$ que codifiqui una seqüència $\boldsymbol{a} \in \{\mathtt{A},\mathtt{C},\mathtt{G},\mathtt{T}\}^{1000}$ amb una cadena binària el més curta possible, usant un codi de bloc o el codi $\mathcal{C} = \{\mathtt{0},\mathtt{10},\mathtt{110},\mathtt{111}\}$, segons si la cadena $\boldsymbol{a}$ conté

1. 250 vegades cada lletra;
2. 500 vegades una lletra, 300 una altra i 100 vegades cadascuna de les altres dues;
3. 400 vegades una lletra i 200 vegades cadascuna de les altres;
4. 100 vegades una lletra i 300 vegades cadascuna de les altres.

1.3 Codis de longitud variable

Referència: Brunat-Ventura [3, Capítol 3]

Els *codis de longitud variable* són codis $\mathcal{C} \subset \mathbb{A}^*$ en què no s'exigeix que totes les paraules hagin de tenir la mateixa longitud. Aquí es consideren només codis finits: formats per un nombre finit de paraules. La majoria de resultats valen, però, també en el cas de codis infinits.

Els codis de longitud variable es fan servir sobretot en compressió de dades de la manera següent: els elements del conjunt de missatges $\mathcal{M}$ que s'han de codificar tenen associades probabilitats, de manera que uns elements són més freqüents que uns altres. Un exemple típic són les lletres en un text en anglès, català, castellà, etc. on les unes són molt més freqüents que les altres. Es construeix una codificació $\mathrm{enc}\colon \mathcal{M} \to \mathbb{A}^*$ que codifiqui els elements més freqüents amb paraules curtes i els menys freqüents amb paraules més llargues, de manera que, en mitjana, les paraules del codi $\mathcal{C} = \mathrm{enc}(\mathcal{M}) \subset \mathbb{A}^*$ usades en la codificació siguin el més curtes possible.

Així, l'objectiu és construir codis que permetin codificar els missatges de $\mathcal{M}$ amb les longituds de les paraules depenent de les freqüències dels missatges que codifiquen.

L'objectiu d'aquesta secció és estudiar una desigualtat fonamental que satisfan les longituds de les paraules d'un codi de longitud variable, anomenada *desigualtat de Kraft-McMillan*. Es recorda que la notació $\ell(\boldsymbol{c})$ denota la longitud d'una paraula $\boldsymbol{c} \in \mathbb{A}^*$. En tota la secció es considerarà un alfabet q-ari $\mathbb{A}$ i un subconjunt finit qualsevol $\mathcal{C} \subset \mathbb{A}^*$ que conté M paraules $\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_M$ de longituds $\ell_i = \ell(\boldsymbol{c}_i)$.

Teorema 1.14 (Desigualtat de Kraft). *Si $\mathcal{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M\}$ és un codi q-ari prefix amb paraules de longituds $\ell_i = \ell(\boldsymbol{c}_i)$, aleshores*

$$\sum_{i=1}^{M} q^{-\ell_i} \leqslant 1.$$

Recíprocament, donats enters positius $\ell_1, \ldots, \ell_M$ que satisfacin la desigualtat anterior existeix un codi q-ari prefix format per M paraules $\boldsymbol{c}_i$ de longituds $\ell(\boldsymbol{c}_i) = \ell_i$.

Prova: Donada una paraula $\boldsymbol{c} \in \mathbb{A}^*$ de longitud $\ell(\boldsymbol{c})$ i un enter $n \geqslant \ell(\boldsymbol{c})$ hi ha $q^{n-\ell(\boldsymbol{c})}$ paraules de longitud n que comencen per $\boldsymbol{c}$: són de la forma $\boldsymbol{c}\|\boldsymbol{x}$ per a alguna paraula $\boldsymbol{x} \in \mathbb{A}^{n-\ell(\boldsymbol{c})}$.

Reordenant si cal les paraules del codi es pot suposar que els nombres ℓ_i estan ordenats en ordre creixent: $\ell_1 \leqslant \ell_2 \leqslant \cdots \leqslant \ell_M$. Es denota $\ell = \ell_M$ la longitud màxima.

Suposi's que els ℓ_i són les longituds de les paraules d'un codi q-ari prefix.

Per a cada índex i Sigui A_i el conjunt de les $q^{\ell-\ell_i}$ paraules que comencen per $\boldsymbol{c}_i$.

Per a $i \neq j$ els conjunts A_i i A_j són disjunts ja que, altrament, una de les dues paraules $\boldsymbol{c}_i$ o $\boldsymbol{c}_j$ seria prefix de l'altra (la de longitud més curta seria prefix de l'altra). La reunió disjunta de tots els conjunts A_i està continguda en $\mathbb{A}^\ell$. Dividint per q^ℓ es dedueix la desigualtat:

$$\bigsqcup_{i=1}^{M} A_i \subseteq \mathbb{A}^\ell \quad \Rightarrow \quad \sum_{i=1}^{M} q^{\ell-\ell_i} \leqslant q^\ell \quad \Rightarrow \quad \sum_{i=1}^{M} q^{-\ell_i} \leqslant 1.$$

Recíprocament, suposi's que les longituds ℓ_i satisfan la desigualtat de l'enunciat. S'ha de veure que existeixen paraules $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M \in \mathbb{A}^*$ amb les longituds ℓ_i donades i tals que cap és prefix d'una altra.

Es demostrarà per inducció sobre el nombre M de paraules. Si $M = 1$ s'agafa com a $\boldsymbol{c}_1$ una paraula qualsevol de longitud ℓ_1. Suposi's demostrat per a $M-1$ i que es donen M longituds satisfent la desigualtat. Aleshores $\sum_{i=1}^{M} q^{-\ell_i} \leqslant 1 \Rightarrow \sum_{i=1}^{M-1} q^{-\ell_i} < 1$. Per hipòtesi d'inducció existeix un codi prefix $\{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{M-1}\}$ amb paraules de longituds $\ell_1, \ldots, \ell_{M-1}$.

El nombre total de paraules de longitud $\ell = \ell_M$ és q^ℓ. D'aquestes, n'hi ha $q^{\ell-\ell_1}$ que comencen per $\boldsymbol{c}_1$, i en general $q^{\ell-\ell_i}$ que comencen per $\boldsymbol{c}_i$. Per tant el nombre de paraules de longitud ℓ que comencen per alguna de les paraules $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{M-1}$ és menor o igual (de fet, és igual) a

$$\sum_{i=1}^{M-1} q^{\ell-\ell_i}$$

Multiplicant per q^ℓ la desigualtat $\sum_{i=1}^{M-1} q^{-\ell_i} < 1$ es dedueix que aquest nombre és estrictament menor que q^ℓ, el nombre total de paraules de longitud ℓ. Per tant existeix alguna paraula $\boldsymbol{c}_M$ de longitud $\ell = \ell_M$ que no té cap de les altres com a prefix. Afegint aquesta paraula a les que ja es tenien, s'obté el codi prefix que es volia.

A la *wiki* això es demostra amb un argument anàleg usant arbres. □

Teorema 1.15 (Desigualtat de McMillan). *Si $\mathcal{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M\}$ és un codi q-ari qualsevol, aleshores*

$$\sum_{i=1}^{M} q^{-\ell(\boldsymbol{c}_i)} \leqslant 1.$$

Prova: Siguin

$$\alpha = \sum_{i=1}^{M} q^{-\ell(\boldsymbol{c}_i)} \qquad \text{i} \qquad \ell = \max\big\{\ell(\boldsymbol{c}_i) : i = 1, \ldots, M\big\}.$$

Aleshores

$$\alpha^2 = \left(\sum_{i=1}^{M} q^{-\ell(\boldsymbol{c}_i)}\right)\left(\sum_{j=1}^{M} q^{-\ell(\boldsymbol{c}_j)}\right) = \sum_{i,j=1}^{M} q^{-\big(\ell(\boldsymbol{c}_i)+\ell(\boldsymbol{c}_j)\big)}$$

i el mateix càlcul per a un exponent $k \geqslant 1$ qualsevol dona

$$\alpha^k = \sum_{i_1,\ldots,i_k} q^{-\big(\ell(\boldsymbol{c}_{i_1})+\cdots+\ell(\boldsymbol{c}_{i_k})\big)} = \sum_{r=1}^{\ell k} \sum_{\ell(\boldsymbol{c}_{i_1})+\cdots+\ell(\boldsymbol{c}_{i_k})=r} q^{-r}$$

on el primer sumatori de la dreta només arriba fins a ℓk ja que aquesta és la longitud màxima que es pot assolir concatenant k paraules del codi. A l'últim sumatori sempre se suma el mateix valor q^{-r}, i cal sumar-lo tantes vegades com paraules hi hagi a $\mathbb{A}^r$ que s'obtinguin concatenant k paraules de $\mathcal{C}$. Pel fet que $\mathcal{C}$ és un codi cada paraula de $\mathbb{A}^r$ pot aparèixer com a màxim una vegada, per tant el sumatori contindrà com a màxim q^r sumands. Es té

$$\alpha^k \leqslant \sum_{r=1}^{\ell k} q^r q^{-r} = \sum_{r=1}^{\ell k} 1 = \ell k.$$

Agafant l'arrel k-èssima a cada costat s'obté la desigualtat

$$\alpha \leqslant \sqrt[k]{\ell}\sqrt[k]{k},$$

que val per a tot enter $k \geqslant 1$. Fent tendir k a infinit el terme de la dreta tendeix a 1, i això demostra la desigualtat de l'enunciat. □

Naturalment, com que les desigualtats en tots dos teoremes són la mateixa, la suficiència del primer teorema també es pot aplicar al segon, de manera que la desigualtat és condició suficient per a l'existència d'algun codi. En endavant aquesta desigualtat s'anomenarà *desigualtat de Kraft-McMillan*.

Aplicant tots dos teoremes es dedueix immediatament el

Corol·lari 1.16. *Si existeix un codi q-ari format per paraules de longituds $\ell_1, \ldots, \ell_M$ també existeix un codi q-ari* que és prefix *amb paraules de les mateixes longituds.*

Problemes

1.5. Digueu quin és el mínim nombre q de lletres d'un alfabet $\mathbb{A}$ per tal que existeixi un codi amb:

1. sis paraules de longituds $1, 1, 2, 2, 3, 3$;
2. onze paraules de longituds $1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3$;
3. cinquanta paraules totes de longitud 3.

i construïu un codi q-ari prefix que tingui paraules d'aquestes longituds.

Quantes paraules de longitud 3 es podrien afegir a aquest codi de manera que segueixi sent un codi?

1.6. Sigui $\mathcal{C}$ un codi q-ari no maximal format per M paraules de longituds ℓ_i. Sigui $\ell = \max(\ell_i)$ la longitud màxima d'aquestes paraules. Sigui $\Sigma = \sum_{i=1}^{M} q^{-\ell_i}$. L'objectiu d'aquest problema és estudiar la manera de "millorar" un codi no complet afegint-li paraules o bé escurçant algunes de les seves paraules.

1. Demostreu que $1 - \Sigma = \frac{k}{q^\ell}$ per a un enter k amb $1 \leqslant k \leqslant q^\ell - 1$.
2. Construïu, per a cada enter $\ell \geqslant 1$, un codi bloc amb $1 - \Sigma = \frac{k}{q^\ell}$ on k és un enter donat de l'interval $1 \leqslant k \leqslant q^\ell - 1$.
3. Demostreu que existeix un codi amb $M+1$ paraules, M amb les mateixes longituds que les de l'apartat anterior i una altra paraula de longitud ℓ.
4. Demostreu que existeix un codi maximal amb M paraules de les mateixes longituds que les de $\mathcal{C}$ i totes les altres paraules de longitud ℓ. Quantes?
5. Existeix sempre un codi amb M paraules de les mateixes longituds de les de $\mathcal{C}$ excepte una, que té una lletra menys?
   INDICACIÓ: la resposta depèn de si $q = 2$ o bé $q > 2$.

1.4 Codis de bloc

Referència: Brunat-Ventura [3, Capítol 6].

Els *codis de bloc* o *codis de longitud fixa* són codis en què totes les paraules tenen la mateixa longitud:

Definició 1.17 (Codi de bloc)**.** *Un codi de bloc de longitud n sobre un alfabet $\mathbb{A}$ és un subconjunt no buit*

$$\mathcal{C} \subseteq \mathbb{A}^n.$$

Si l'alfabet té $|\mathbb{A}| = q$ lletres es diu codi q-ari. *Es denota $M = |\mathcal{C}|$ el nombre de paraules que formen el codi.*

Es té $1 \leqslant M \leqslant q^n$. Els codis formats per una única paraula no tenen aplicació pràctica i s'acostuma a suposar que el codi conté més d'una paraula: $M \geqslant 2$.

Els codis de bloc se solen agafar sobre alfabets $\mathbb{A} = \mathbb{F}$ que tenen una estructura de cos finit. D'aquesta manera les paraules es poden veure com a vectors d'un $\mathbb{F}$-espai vectorial o també com a polinomis a coeficients en $\mathbb{F}$, el qual dona eines d'àlgebra lineal i àlgebra de polinomis per treballar amb elles. Aquesta estructura addicional es veurà a les seccions 5 de codis lineals i 6 de codis polinomials. El cardinal d'un cos finit és sempre una potència d'un nombre primer: $q = p^e$. A la pràctica són especialment importants el cos $\mathbb{F}_2$ de dos elements i els cossos de nombre d'elements $q = 2^e$ que sigui potència de 2.

Definició 1.18 (Codi lineal)**.** *Un codi lineal de longitud n sobre el cos finit $\mathbb{F}$ és un subespai vectorial*

$$\mathcal{C} \subseteq \mathbb{F}^n.$$

Es denota $k = \dim_{\mathbb{F}}(\mathcal{C})$ la seva dimensió. Es té $M = |\mathcal{C}| = q^k$.

Prova: Per calcular el nombre de paraules d'un codi lineal s'agafa una base $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_k$, que està formada per k elements on k és la dimensió. Tota paraula codi $\boldsymbol{v} \in \mathcal{C}$ és una combinació lineal de la base de manera única

$$\boldsymbol{v} = \lambda_1 \boldsymbol{v}_1 + \lambda_2 \boldsymbol{v}_2 + \cdots + \lambda_k \boldsymbol{v}_k$$

on els coeficients $\lambda_i \in \mathbb{F}$ són escalars, que poden prendre q valors diferents. Per tant el nombre de paraules possibles és q^k. □

El concepte de *dimensió* es pot generalitzar a codis de bloc qualsevol, no necessàriament lineals, definint $k := \log_q M$. En aquest cas general, no necessàriament lineal, la dimensió és un nombre real amb $0 \leqslant k \leqslant n$. En àlgebra lineal s'acostuma a anomenar *codimensió* d'un subespai vectorial $V \subseteq \mathbb{K}^n$ de dimensió k a la diferència $n - k$. Així, es defineix la

Definició 1.19 (Codimensió)**.** *La* codimensió *d'un codi de bloc és la diferència entre la seva longitud i la seva dimensió $m := n - k$.*

Exemples 1.20. *Sigui $\mathbb{A}$ un alfabet q-ari, que pot ser un cos finit.*

1. *Un* codi trivial *és un codi que conté només una paraula de $\mathbb{A}^n$. Aquests són codis poc interessants i moltes vegades se suposa per defecte que els codis que es consideren són no trivials, sense que calgui dir-ho explícitament. Quan $\mathbb{A} = \mathbb{F}$ és un cos, l'únic codi trivial que és lineal és el format per la paraula zero.*
2. *El* codi total $\mathcal{C} = \mathbb{A}^n$*, format per totes les q^n paraules de longitud n. Quan $\mathbb{A} = \mathbb{F}$ és un cos, és un codi lineal de dimensió n.*
3. *El* codi de repetició $\mathrm{Rep}_q(n)$*, format per totes les paraules de longitud n que s'obtenen en repetir una mateixa lletra de l'alfabet $\mathcal{C} = \{\boldsymbol{c} = \mathtt{c}\mathtt{c}\cdots\mathtt{c} \in \mathbb{A}^n : \mathtt{c} \in \mathbb{A}\}$, que conté $M = q$ paraules. Quan $\mathbb{A} = \mathbb{F}$ és un cos, és un codi lineal de dimensió* 1*.*
4. *El* codi binari parell $\mathrm{Par}_2(n)$ *format per totes les paraules de longitud n sobre l'alfabet binari $\mathbb{A} = \mathbb{F}_2 = \{\mathtt{0}, \mathtt{1}\}$ que tenen un nombre parell d'uns. Aquest codi conté $M = 2^{n-1}$ paraules. És un codi lineal de dimensió $n - 1$.*

   També es pot considerar el codi format per les paraules amb nombre senar d'uns, que conté el mateix nombre de paraules, però aquest no és lineal.

5. *Anàlogament, per a tot nombre enter $q > 1$ es pot considerar el codi q-ari* $\mathrm{Par}_q(n)$ *sobre l'alfabet* $\mathbb{A} = \mathbb{Z}_q$ *de les classes de congruència mòdul q format per totes les paraules* $\boldsymbol{x} = (\mathtt{x}_1, \ldots, \mathtt{x}_n) \in \mathbb{Z}_q^n$ *amb lletres que sumen zero:* $\sum_{i=1}^n \mathtt{x}_i \equiv \mathtt{0} \pmod q$. *Aquest codi conté* q^{n-1} *paraules que es poden obtenir afegint a totes les paraules* $(\mathtt{x}_1, \ldots, \mathtt{x}_{n-1}) \in \mathbb{Z}_q^{n-1}$ *de longitud* $n-1$ *una lletra al final calculada com* $\mathtt{x}_n \equiv -\sum_{i=1}^{n-1} \mathtt{x}_i \pmod q$.

   Quan l'alfabet és un cos finit $\mathbb{F}_q$ *aquest codi és un codi lineal de dimensió* $n-1$.

6. *El codi binari de longitud* 5 *format per les quatre paraules següents:*

$$\mathcal{C} = \{\mathtt{00000}, \mathtt{11100}, \mathtt{00111}, \mathtt{11011}\} \subset \mathbb{F}_2^5$$

   és un codi lineal de dimensió 2.

7. *El codi binari de longitud* 6 *format per les vuit paraules següents:*

$$\mathcal{C} = \{\mathtt{000000}, \mathtt{001011}, \mathtt{010101}, \mathtt{100110}, \mathtt{011110}, \mathtt{101101}, \mathtt{110011}, \mathtt{111000}\} \subset \mathbb{F}_2^6$$

   és un codi lineal de dimensió 3.

En el conjunt de les paraules de longitud n es pot definir una distància que mesura com de diferents són dues paraules. És la

Definició 1.21 (Distància de Hamming)**.** *Donades paraules* $\boldsymbol{x} = \mathtt{x}_1 \cdots \mathtt{x}_n$ *i* $\boldsymbol{y} = \mathtt{y}_1 \cdots \mathtt{y}_n$ *de* $\mathbb{A}^n$*, la* distància de Hamming *entre totes dues es defineix com el nombre de posicions en què tenen lletres diferents*

$$d(\boldsymbol{x}, \boldsymbol{y}) = \big|\{i : \mathtt{x}_i \neq \mathtt{y}_i\}\big|.$$

Per exemple,

$$d(\mathtt{1010101}, \mathtt{1010110}) = 2, \quad d(\mathtt{0000000}, \mathtt{1011100}) = 4, \quad d(\mathtt{paraula}, \mathtt{partida}) = 3.$$

Si S és un subconjunt de $\mathbb{A}^n$, es defineix la distància d'una paraula $\boldsymbol{x} \in \mathbb{A}^n$ al conjunt S com $d(\boldsymbol{x}, S) = \min\big\{d(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{y} \in S\big\}$. Es defineix anàlogament la distància entre dos subconjunts com $d(S, T) = \min\big\{d(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x} \in S, \boldsymbol{y} \in T\big\}$.

Proposició 1.22. *La distància de Hamming compleix les propietats següents:*

1. $d(\boldsymbol{x}, \boldsymbol{y}) \geqslant 0$; $\quad d(\boldsymbol{x}, \boldsymbol{y}) = 0 \;\Leftrightarrow\; \boldsymbol{x} = \boldsymbol{y}$ $\qquad$ *(positivitat);*
2. $d(\boldsymbol{x}, \boldsymbol{y}) = d(\boldsymbol{y}, \boldsymbol{x})$ $\qquad$ *(simetria);*
3. $d(\boldsymbol{x}, \boldsymbol{z}) \leqslant d(\boldsymbol{x}, \boldsymbol{y}) + d(\boldsymbol{y}, \boldsymbol{z})$ $\qquad$ *(desigualtat triangular).*

Prova: Les dues primeres propietats són evidents a partir de la definició. La desigualtat triangular es pot comprovar veient-la primer per a paraules d'una sola lletra:

$$d(\mathtt{x}, \mathtt{y}) = \begin{cases} 0, & \mathtt{x} = \mathtt{y} \\ 1, & \mathtt{x} \neq \mathtt{y}, \end{cases} \qquad \mathtt{x}, \mathtt{y} \in \mathbb{A} = \mathbb{A}^1.$$

En aquest cas la desigualtat $d(\mathtt{x},\mathtt{z}) \leqslant d(\mathtt{x},\mathtt{y}) + d(\mathtt{y},\mathtt{z})$ es comprova per casos: si $\mathtt{x} = \mathtt{z}$ aleshores $d(\mathtt{x},\mathtt{z}) = 0$ i no s'ha de comprovar res, i si $\mathtt{x} \neq \mathtt{z}$ aleshores $d(\mathtt{x},\mathtt{z}) = 1$ i com que necessàriament $\mathtt{x} \neq \mathtt{y}$ o bé $\mathtt{y} \neq \mathtt{z}$ dels dos sumands de la dreta almenys un val 1. El cas general es dedueix aleshores d'aquest observant que

$$d(\boldsymbol{x},\boldsymbol{z}) = \sum_{i=1}^{n} d(\mathtt{x}_i,\mathtt{z}_i) \leqslant \sum_{i=1}^{n} \big(d(\mathtt{x}_i,\mathtt{y}_i) + d(\mathtt{y}_i,\mathtt{z}_i)\big)$$
$$= \sum_{i=1}^{n} d(\mathtt{x}_i,\mathtt{y}_i) + \sum_{i=1}^{n} d(\mathtt{y}_i,\mathtt{z}_i) = d(\boldsymbol{x},\boldsymbol{y}) + d(\boldsymbol{y},\boldsymbol{z}).$$

□

Aquestes tres propietats de la distància de Hamming són les propietats matemàtiques que es demanen quan es vol donar una estructura d'*espai mètric* a un conjunt, i permeten usar al conjunt $\mathbb{A}^n$ conceptes geomètrics relacionats amb el fet de tenir definida la distància entre paraules. En particular, com en tot espai mètric, es té el concepte de bola de radi r, que són els punts a distància $\leqslant r$ d'un punt central donat:

Definició 1.23 (Bola)**.** *Per a tot $r \geqslant 0$ i cada paraula $\boldsymbol{x} \in \mathbb{A}^n$ la* bola (o disc) de radi r *centrada en la paraula $\boldsymbol{x}$ és el conjunt de les paraules a distància $\leqslant r$ de $\boldsymbol{x}$:*

$$B_r(\boldsymbol{x}) = \big\{\boldsymbol{y} \in \mathbb{A}^n : d(\boldsymbol{y},\boldsymbol{x}) \leqslant r\big\}$$

Naturalment, només té sentit considerar boles de radis $r \leqslant n$ ja que a partir de radi $r = n$ tota bola centrada en una paraula qualsevol és tot el conjunt $\mathbb{A}^n$.

Lema 1.24. *El nombre d'elements d'una bola de radi r és*

$$|B_r(\boldsymbol{x})| = \sum_{i=0}^{r} \binom{n}{i} (q-1)^i.$$

Prova: En efecte, els elements de la bola són els que difereixen de $\boldsymbol{x}$ en $0 \leqslant i \leqslant r$ posicions i, per a cada tria de les i posicions, de les quals n'hi ha $\binom{n}{i}$, es pot canviar la lletra de la paraula $\boldsymbol{x}$ per qualsevol lletra diferent, i cada lletra en té $q-1$, de lletres diferents. □

Una propietat molt senzilla però útil de la distància de Hamming és que es pot anar d'una paraula a una altra qualsevol que estigui a distància $d = d_1 + d_2$ passant per una tercera paraula que estigui a distància d_1 de la primera i a distància d_2 de la segona:

Lema 1.25. *Siguin $\boldsymbol{x},\boldsymbol{y} \in \mathbb{A}^n$ paraules a distància $d = d(\boldsymbol{x},\boldsymbol{y})$ l'una de l'altra. Si $d = d_1 + d_2$ aleshores existeix una paraula $\boldsymbol{z} \in \mathbb{A}^n$ amb $d(\boldsymbol{x},\boldsymbol{z}) = d_1$ i $d(\boldsymbol{z},\boldsymbol{y}) = d_2$.*

Prova: Siguin $\boldsymbol{x} = \mathtt{x}_1 \cdots \mathtt{x}_n$ i $\boldsymbol{y} = \mathtt{y}_1 \cdots \mathtt{y}_n$. Siguin $i_1,\ldots,i_d$ els índexs que indiquen les posicions en què $\boldsymbol{x}$ i $\boldsymbol{y}$ són diferents: $\mathtt{x}_{i_j} \neq \mathtt{y}_{i_j}$ per a $j = 1,\ldots,d$ però, en canvi, $\mathtt{x}_k = \mathtt{y}_k$ per a tot índex k diferent dels i_j.

Sigui $\boldsymbol{z} = \mathtt{z}_1 \dots \mathtt{z}_n$ la paraula amb lletres definides per:

$$\mathtt{z}_k = \begin{cases} \mathtt{x}_k = \mathtt{y}_k, & k \neq i_1, \dots, i_d, \\ \mathtt{y}_k, & k = i_1, \dots, i_{d_1}, \\ \mathtt{x}_k, & k = i_{d_1+1}, \dots, i_d. \end{cases}$$

Aquesta paraula $\boldsymbol{z}$ satisfà òbviament les condicions requerides. □

Definició 1.26 (Distància mínima)**.** *La distància mínima d'un codi de bloc $\mathcal{C}$ es defineix com el mínim de les distancies entre paraules diferents del codi:*

$$d = d(\mathcal{C}) = \min\big\{d(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}, \boldsymbol{x} \neq \boldsymbol{y}\big\}.$$

En general $d \geqslant 1$, ja que la distància entre dues paraules diferents és com a minim 1. D'altra banda és clar que $d \leqslant n$ ja que dues paraules de longitud n només poden diferir en un màxim de n posicions.

Per exemple, el codi total $\mathbb{A}^n$ té distància mínima 1, el codi de repetició $\mathrm{Rep}_q(n)$ té distància mínima n, el codi binari parell $\mathrm{Par}_2(n)$ (de qualsevol longitud $n \geqslant 2$) té distància mínima 2. Es pot veure que els dos darrers codis dels exemples 1.20, de longituds 5 i 6, tenen tots dos distància mínima igual a 3. Per fer-ho s'haurien de comparar tots els parells de paraules diferents del codi i veure que les distàncies entre elles són sempre $\geqslant 3$, i que hi ha algun parell a distància 3, però tenint en compte que són lineals això es pot comprovar molt més fàcilment a partir del pes de Hamming de les seves paraules, tal com es veurà a la secció de codis lineals.

Per a un codi que conté una única paraula la definició de distància mínima no té sentit, ja que defineix $d(\mathcal{C})$ com el mínim del conjunt buit. De vegades pot ser convenient posar $d(\mathcal{C}) = n + 1$ per a aquests codis, on n és la longitud de l'única paraula.

Definició 1.27 (Radi de tangència)**.** *Es defineix el radi de tangència d'un codi a partir de la seva distància mínima com*

$$\tau = \left\lfloor \frac{d-1}{2} \right\rfloor$$

Observi's que es té $d = 2\tau + 1$ o $2\tau + 2$ segons si d és senar o parell.

Proposició 1.28 (Radi de tangència)**.** *El radi de tangència d'un codi és el radi més gran tal que les boles d'aquest radi centrades en paraules del codi són totes disjuntes:*

$$\tau = \max\big\{r \geqslant 0 : B_r(\boldsymbol{c}) \cap B_r(\boldsymbol{c}') = \varnothing,\ \forall\ \boldsymbol{c}, \boldsymbol{c}' \in \mathcal{C}, \boldsymbol{c} \neq \boldsymbol{c}'\big\}.$$

Prova: És clar que τ compleix la propietat ja que si la intersecció de dues boles fos no buida els seus centres estarien a distància $\leqslant d - 1$. És el més petit ja que si s'agafa $\tau + 1$ aleshores agafant dues paraules del codi a distància mínima d i una paraula a distància $d_1 := \tau + 1$ de l'una i $d_2 = d - d_1$ de l'altra, aleshores aquesta paraula pertanyeria a la intersecció de les dues boles. □

Paràmetres d'un codi. A un codi de bloc $\mathcal{C} \subseteq \mathbb{A}^n$ se li associen els paràmetres següents. Els més importants són el nombre q de lletres de l'alfabet, la longitud n, la dimensió k i la distància mínima d; els demés s'obtenen a partir d'aquests.

- el *nombre de lletres* de l'alfabet $q = |\mathbb{A}|$;
- la *longitud* n;
- el *nombre de paraules* $M = |\mathcal{C}|$;
- la *distància mínima* $d = d(\mathcal{C})$;
- la *dimensió* $k = \log_q M$, que per als codis lineals és un enter;
- la *codimensió* $m = n - k$;
- les *taxes d'informació*[1] $R = \frac{\log_q M}{n} = \frac{k}{n}$ i de *redundància* $1 - R = \frac{n-k}{n} = \frac{m}{n}$;
- la *capacitat correctora* o *radi de tangència*, definida en termes de d com $\tau = \lfloor \frac{d-1}{2} \rfloor$;
- la *distància mínima relativa* $\delta = \delta(\mathcal{C}) = \frac{d(\mathcal{C})}{n}$.

Atenent als seus paràmetres en teoria de codis s'acostuma a dir que el codi és

$$\textit{de tipus} \quad (n, M, d)_q \qquad \text{o} \qquad \textit{de tipus} \quad [n, k, d]_q$$

segons si el nombre de paraules es vol indicar directament o a través de la dimensió. El subíndex q no es posa si es dona per sobreentès i la distància mínima no es posa quan es desconeix o no és rellevant.

Optimització de paràmetres. Un dels problemes fonamentals de la teoria de codis és l'optimització d'un dels tres paràmetres n, M i d quan els altres dos estan fixats: es tracta de trobar codis de longitud curta (n petit) que continguin moltes paraules (M gran) que estiguin molt separades les unes de les altres (d gran). Naturalment, això dependrà del nombre de lletres q de l'alfabet. Se sol plantejar en termes de maximitzar M quan n i d (i q) estan fixats i es fa servir la notació següent:

Definició 1.29. *Es defineix:*

$$A_q(n, d) := \max\big\{M : \text{existeix un codi de tipus } (n, M, d)_q\big\}.$$

No hi ha una fórmula simple que doni aquest valor en funció dels tres paràmetres q, n, d. Calcular-lo en casos particulars és un problema combinatori de gran dificultat computacional. Es coneixen només uns quants *valors* de $A_q(n, d)$ per a q, n i d petits. En general, però, només es tenen fites. Una de les més senzilles és la

Proposició 1.30 (*Fita de Singleton*)**.** *Una condició necessària per a tot codi de tipus* $(n, M, d)_q$ *és que*

$$M \leqslant q^{n-d+1}.$$

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

[1] En l'enunciat del teorema de codificació de canal es fa servir el "ratio binari" $R = \frac{\log M}{n}$, amb el logaritme agafat en base 2.

PROVA: Sigui $\mathcal{C} \subseteq \mathbb{A}^n$ un codi q-ari de M paraules, longitud n i distància mínima d. Es considera el conjunt $\mathcal{C}'$ obtingut retallant les $d-1$ últimes lletres de les paraules de $\mathcal{C}$:

$$\mathcal{C}' = \left\{ \boldsymbol{c}' = (\mathsf{c}_1, \mathsf{c}_2, \ldots, \mathsf{c}_{n-d+1}) : \boldsymbol{c} = (\mathsf{c}_i) \in \mathcal{C} \right\} \subseteq \mathbb{A}^{n-d+1}.$$

Les paraules $\boldsymbol{c}'$ que s'obtenen d'aquesta manera són totes diferents, ja que, com que dues paraules diferents de $\mathcal{C}$ difereixen en almenys d lletres, en treure'n només $d-1$ les paraules resultants han de diferir en almenys una lletra. Per tant,

$$M = |\mathcal{C}| = |\mathcal{C}'| \leqslant q^{n-d+1}$$

i es dedueix la fita de l'enunciat. □

Moltes vegades, i en especial per als codis lineals, aquesta fita s'escriu com una relació entre longitud, dimensió i distància mínima en la forma equivalent següent:

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

Per a codis lineals la fita de Singleton es pot deduir també amb arguments d'àlgebra lineal a partir de matrius generadores i de control del codi, tal com es veu en el problema **0.13**.

Els codis que assoleixen la fita de Singleton s'anomenen:

Definició 1.31 (Codis MDS)**.** *Un codi MDS (de distància de separació màxima) és un codi que assoleix la fita de Singleton. És a dir, un codi de tipus* $(n, M, d)_q$ *amb* $M = q^{n-d+1}$*; o, equivalentment, un codi de tipus* $[n, k, d]$ *amb* $d = n - k + 1$*.*

Exemples 1.32. *Els codis següents són codis MDS:*

1. *el codi total* $\mathbb{A}^n$*;*
2. *el codi de repetició* $\mathrm{Rep}_q(n)$*;*
3. *el codi de bit de paritat* $\mathrm{Par}_q(n)$*.*

PROVA: Es comprova calculant els paràmetres:

1. és de tipus $[n, n, 1]$ i $d = 1 = n - n + 1 = n - k + 1$;
2. és de tipus $[n, 1, n]$ i $d = n = n - 1 + 1 = n - k + 1$;
3. és de tipus $[n, n-1, 2]$ i $d = 2 = n - (n-1) + 1 = n - k + 1$. □

Es pot veure que en el cas dels codis binaris aquests exemples són els únics codis MDS que existeixen. Sobre alfabets més grans es poden construir codis MDS amb altres paràmetres. Concretament, en el cas d'alfabets $\mathbb{A} = \mathbb{F}$ amb estructura de cos finit sempre hi ha codis MDS de tipus $[n, k, d]_q$ per a tot $n \leqslant q + 1$ amb paràmetres $k, d \geqslant 1$ qualssevol tals que $d + k = n + 1$. Els codis de Reed-Solomon, que es veuran a la secció 6.1, són codis MDS que assoleixen aquests paràmetres. Entre els codis més importants usats en les aplicacions pràctiques hi ha els codis de Reed-Solomon sobre l'alfabet $\mathbb{F}_{256}$.

Usant el nombre de punts de les boles en l'espai $\mathbb{A}^n$ s'obté la fita següent, coneguda també com a fita de l'*empaquetament d'esferes*:

Teorema 1.33 (*Fita de Hamming*)**.**

$$A_q(n,d) \leqslant \frac{q^n}{\sum_{i=0}^{\tau}\binom{n}{i}(q-1)^i}, \qquad \tau = \left\lfloor \frac{d-1}{2} \right\rfloor.$$

Prova: Observi's que el denominador de l'expressió de la fita és simplement el nombre de punts de les boles de radi τ calculat al lema 1.24.

Sigui $\mathcal{C}$ un codi qualsevol de tipus $(n, M, d)_q$. Les boles de centre les paraules del codi i radi τ son subconjunts de $\mathbb{A}^n$ que són disjunts dos a dos (proposició 1.28) i cadascun conté $\beta = \sum_{i=0}^{\tau}\binom{n}{i}(q-1)^i$ elements. La reunió de tots aquests conjunts conté $M\beta$ elements i és un subconjunt de $\mathbb{A}^n$. Per tant, $M\beta \leqslant q^n \Rightarrow M \leqslant \frac{q^n}{\beta}$. □

Els codis que assoleixen la fita de Hamming s'anomenen:

Definició 1.34 (Codi perfecte)**.** *Un codi perfecte és un codi amb nombre de paraules que assoleix la fita de Hamming.*

És a dir, un codi perfecte és un codi en el qual la reunió de totes les boles de radi $\tau = \lfloor \frac{d-1}{2} \rfloor$, que són disjuntes, és tot l'espai $\mathbb{A}^n$: les boles són una *partició* d'aquest espai.

Exemple 1.35. *Els codis següents són perfectes:*

1. *el codi total* $\mathbb{A}^n$*;*
2. *els codis binaris de repetició* $\mathrm{Rep}_2(n)$ *de longitud* n *senar;*
3. *els codis binaris de Hamming* $\mathrm{Ham}_2(m)$*, que són de tipus* $[2^m-1, 2^m-m-1, 3]_2$*.*

Prova: Els codis de Hamming encara no s'han definit però per veure que són codis perfectes n'hi ha prou a saber quins són els seus paràmetres, que són els donats a l'enunciat.

1. El codi total té distància mínima $d = 1$ i per tant $\tau = 0$. Les boles de radi zero contenen només un punt: el seu centre. La reunió d'aquestes boles centrades en punts del codi $\mathcal{C} = \mathbb{A}^n$ és tot l'espai.
2. El fet que la longitud sigui senar permet posar $n = d = 2\tau + 1$. El codi conté només dues paraules. Les boles de radi τ contenen 2^{n-1} paraules. Es té

$$2^n = (1+1)^n = \sum_{i=0}^{n}\binom{n}{i} = \sum_{i=0}^{\tau}\binom{n}{i} + \sum_{i=\tau+1}^{n}\binom{n}{i} = 2\sum_{i=0}^{\tau}\binom{n}{i} \Rightarrow \sum_{i=0}^{n}\binom{n}{i} = 2^{n-1}.$$

   El segon sumatori coincideix amb el primer ja que $\binom{n}{i} = \binom{n}{n-i}$ i quan i recorre els enters entre $\tau+1$ i n la diferència $n-i$ recorre els enters entre 0 i $n-(\tau+1) = \tau$. Es dedueix que entre les dues boles recobreixen tot l'espai $\{0,1\}^n$.
3. Es té $\tau = 1$. Les boles de radi 1 contenen $1 + n = 1 + 2^m - 1 = 2^m$ punts. Com que el codi conté 2^{2^m-m-1} paraules la reunió de les boles de radi τ centrades en aquestes paraules conté

$$2^m \times 2^{2^m-m-1} = 2^{2^m-1} = 2^n$$

   paraules: les boles recobreixen totalment l'espai $\{0,1\}^n$, i per tant el codi és un codi perfecte. □

Codis asimptòticament bons. Els codis d'una successió $(\mathcal{C}_i)_{i\geqslant 1}$ de codis sobre un alfabet q-ari de longituds n_i creixents es diuen asimptòticament bons si tant els seus ratios com les seves distàncies mínimes relatives es mantenen fitats inferiorment per nombres positius. Més específicament, si $[n_i, k_i, d_i]_q$ són els paràmetres d'aquests codis aleshores existeixen nombres positius $R > 0$ i $\delta > 0$ tals que les quantitats $R_i = R(\mathcal{C}_i) = \frac{k_i}{n_i}$ i $\delta_i = \delta(\mathcal{C}_i) = \frac{d_i}{n_i}$ satisfan:

$$R_i \geqslant R \quad \text{i} \quad \delta_i \geqslant \delta, \qquad \forall i \geqslant 1.$$

La construcció de successions de codis asimptòticament bons és difícil.

Problemes

1.7. *Codis binaris.* En el conjunt $\{0,1\}^n$ de les paraules binaries de longitud n es considera la *suma bit a bit*. Es denotarà $\mathbf{0}$ la paraula formada per n zeros. Per a $\mathbf{x} \in \{0,1\}^n$ es denotarà $w(\mathbf{x})$ o $\|\mathbf{x}\|$ el nombre de uns que té (*pes de Hamming*). Demostreu que

1. un codi $\mathcal{C} \subseteq \{0,1\}^n$ és lineal si, i només si, és tancat per la suma i conté $\mathbf{0}$;
2. una aplicació $\{0,1\}^n \to \{0,1\}^m$ és lineal si, i només si, envia sumes a sumes i envia el zero al zero;
3. $\boldsymbol{x} + \boldsymbol{x} = \mathbf{0}$ per a tot $\boldsymbol{x} \in \{0,1\}^n$;
4. per a tot parell $\boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^n$ es té $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}$ amb $\boldsymbol{e} \in \{0,1\}^n$ i $d(\boldsymbol{x}, \boldsymbol{y}) = w(\boldsymbol{e})$;
5. la distància mínima d'un codi lineal és $d(\mathcal{C}) = \min\{w(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{C}, \boldsymbol{x} \neq \mathbf{0}\}$;
6. $w(\boldsymbol{x} + \boldsymbol{y}) \equiv w(\boldsymbol{x}) + w(\boldsymbol{y}) \pmod 2$.

1.8. *Paritat.* Es considera la paritat del nombre d'uns d'una paraula binària. Les paraules parells són aquelles que la suma dels seus bits (mòdul 2) és zero. Donada una paraula $\boldsymbol{x} = (\mathtt{x}_1, \ldots, \mathtt{x}_n) \in \{0,1\}^n$ es defineix la seva *extensió parell* com la paraula obtinguda en afegir-li un bit de manera que el resultat sigui una paraula parell. Comproveu que:

$$\mathrm{ev}(\boldsymbol{x}) = (\mathtt{x}_1, \ldots, \mathtt{x}_n, \mathtt{x}_{n+1}) \in \{0,1\}^{n+1}, \qquad \mathtt{x}_{n+1} = \sum_{i=1}^{n} \mathtt{x}_i \pmod 2.$$

1. el conjunt $\mathrm{Par}_2(n)$ de les paraules parells de longitud n són un subespai vectorial de $\{0,1\}^n$ i doneu bases d'aquest subespai;
2. l'aplicació $\mathrm{ev}\colon \{0,1\}^n \to \{0,1\}^{n+1}$ és una aplicació lineal injectiva;
3. per a tota paraula $\boldsymbol{x} \in \{0,1\}^n$,

$$w\big(\mathrm{ev}(\boldsymbol{x})\big) = \begin{cases} w(\boldsymbol{x}), & w(\boldsymbol{x}) \ \text{ parell}, \\ w(\boldsymbol{x}) + 1, & w(\boldsymbol{x}) \ \text{ senar}. \end{cases}$$

4. per a tot parell de paraules $\boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^n$,

$$d\big(\mathrm{ev}(\boldsymbol{x}), \mathrm{ev}(\boldsymbol{y})\big) = \begin{cases} d(\boldsymbol{x}, \boldsymbol{y}), & d(\boldsymbol{x}, \boldsymbol{y}) \ \text{ parell}, \\ d(\boldsymbol{x}, \boldsymbol{y}) + 1, & d(\boldsymbol{x}, \boldsymbol{y}) \ \text{ senar}. \end{cases}$$

1.9. *Extensió parell.* Sigui $\mathcal{C} \subseteq \{0,1\}^n$ un codi binari de longitud n. Es construeix un altre codi agafant les extensions parells de totes les paraules de $\mathcal{C}$, que s'anomena extensió parell del codi donat.

$$\mathcal{C}^{\text{ev}} = \text{ev}(\mathcal{C}) = \big\{\text{ev}(\boldsymbol{c}) : \boldsymbol{c} \in \mathcal{C}\big\}$$

Calculeu els paràmetres d'aquest codi en funció dels de $\mathcal{C}$.

1.10. Es defineix el codi complementari d'un codi binari $\mathcal{C} \subseteq \{0,1\}^n$ com el codi que s'obté intercanviant els zeros i els uns. Quins paràmetres té?

1.11. Sigui τ el radi de tangència d'un codi de distància mínima d. Comproveu que $d = 2\tau+1$ o bé $2\tau + 2$ segons la seva paritat.

1.12. Recordeu que $A_q(n,d)$ és el nombre màxim de paraules que pot tenir un codi q-ari de longitud n i distància mínima d:

$$A_q(n,d) = \max\big\{M = |\mathcal{C}| : \mathcal{C} \subseteq \mathbb{A}^n, d(\mathcal{C}) = d\big\},$$

on $\mathbb{A}$ és un alfabet q-ari: $|\mathbb{A}| = q$.

1. Quant valen $A_q(n,1)$ i $A_q(n,n)$?
2. Demostreu que per a tot $n \geqslant d \geqslant 1$ es té $A_q(n+1,d+1) \leqslant A_q(n,d)$.
   INDICACIÓ: Agafeu un codi que maximitzi $A_q(n+1,d+1)$ i retalleu una lletra adequada.
3. Demostreu que per a $q = 2$ i d senar la desigualtat de l'apartat anterior és una igualtat. INDICACIÓ: Afegiu un bit de paritat.

1.13. Demostreu que $A_2(3,2) = 4$ i $A_3(3,2) = 9$. Calculeu $A_q(3,2)$ per a q qualsevol.

1.14. Comproveu que codis de tipus $[23,12,7]_2$ i de tipus $[11,6,5]_3$ són perfectes.

Existeixen codis d'aquests tipus, anomenats *codi de Golay binari* i *codi de Golay ternari*.

1.15. Demostreu que la distància mínima d'un codi perfecte ha de ser senar.

1.5 Descodificació de codis de bloc

Referència: Brunat-Ventura [3, Capítol 6]

La principal aplicació dels codis de bloc és la *detecció i correcció d'errors*. Els canals de comunicacions i els dispositius d'emmagatzematge del mon real introdueixen soroll que de vegades provoca errors en les dades transmeses o guardades. Per detectar aquests errors, i fins i tot corregir-los, es pot enviar la informació codificada amb un codi de bloc. El receptor, en rebre una paraula que no sigui del codi, sap que s'han produït errors en algunes lletres d'aquesta paraula: detecta que la paraula s'ha rebut erròniament. També pot intentar corregir aquests errors canviant la paraula rebuda per una paraula codi, procurant maximitzar la probabilitat que aquesta sigui la paraula codi que s'havia enviat.

La detecció i correcció d'errors es veu clarament en els dos exemples prototípics següents:

- Usant un codi binari parell $\mathrm{Par}_2(n)$ es detecten els errors que afectin només un bit de cada paraula. Quan la transmissió canvia un dels bits de la paraula codi enviada, que té un nombre parell de uns, la converteix en una paraula amb un nombre senar de uns, que no és del codi, i així el receptor detecta que s'ha produït un error.

  Si, en canvi, en una mateixa paraula es produeixen dos errors, això no es detectarà perquè el resultat seguirà sent una paraula parell. Més precisament, usant aquest codi es detecten tots els errors en una mateixa paraula que afectin un nombre senar de bits però cap dels que afectin un nombre parell.

  Anàlogament, usant un codi q-ari parell $\mathrm{Par}_q(n)$ es detecten tots els errors que afectin només una lletra en cada paraula, però només alguns (però no tots) els errors que afectin més d'una lletra.

- Usant un codi de repetició $\mathrm{Rep}_q(3)$ de longitud 3 es poden corregir tots els errors que afectin només una lletra en cada paraula: les paraules enviades són les de la forma $\boldsymbol{a} = \mathtt{aaa} \in \mathbb{A}^3$ amb $\mathtt{a} \in \mathbb{A}$: les que repeteixen la mateixa lletra tres vegades. Si una de les tres lletres `a` es transmet erròniament canviant-la per $\mathtt{b} \neq \mathtt{a}$ (o sigui, es rep `baa` o bé `aba` o bé `aab`) aleshores el receptor pot assegurar que la paraula enviada és `aaa`, agafant la lletra que apareix dues vegades a la paraula rebuda. Si, en canvi, en una paraula es produeixen dos o tres errors el receptor no podrà corregir-los.

  Anàlogament, usant un codi de repetició de longitud senar $n = 2\tau+1$ es podran corregir tots els errors que afectin com a màxim a $\tau = \lfloor \frac{d-1}{2} \rfloor$ lletres d'una paraula. La correcció es fa per majoria: el receptor ha d'agafar simplement la paraula codi que correspon a la lletra que surti més vegades en la paraula rebuda.

Així, la manera de fer servir un codi de bloc per corregir errors consisteix a usar una aplicació de descodificació $\mathrm{ccdec}\colon \mathbb{A}^n \to \mathcal{C}$. En les aplicacions dels codis de bloc a la codificació de canal (veure secció 4) s'anomena codi de canal al parell format per un codi de bloc $\mathcal{C}$ junt amb una aplicació de descodificació com aquesta.

El receptor, en rebre una paraula $\boldsymbol{x} \in \mathbb{A}^n$, la converteix en una paraula codi usant aquesta aplicació $\boldsymbol{c} = \mathrm{ccdec}(\boldsymbol{x}) \in \mathcal{C}$, i assumeix que aquesta és la paraula que l'emissor li ha enviat. Aquest procés es pot representar de la manera següent:

$$\mathcal{M} \overset{\mathrm{enc}}{\longleftarrow} \mathcal{C} \subseteq \mathbb{A}^n \overset{\mathrm{canal}}{\longrightarrow} \mathbb{A}^n \overset{\mathrm{ccdec}}{\longrightarrow} \mathcal{C} \overset{\mathrm{dec}}{\longleftarrow} \mathcal{M}.$$

Naturalment, l'aplicació de descodificació ccdec s'ha de triar de manera que corregeixi el màxim nombre d'errors possible.

Aquesta accepció de la paraula *descodificació* no s'hauria de confondre amb la descodificació corresponent a la definició 1.11, que es denota dec, la qual recupera un missatge que s'ha codificat usant un codi, i que només està definida en el subconjunt de $\mathcal{C} \subseteq \mathbb{A}^n$ de les paraules codi. Les aplicacions ccdec estan definides en tots els elements de $\mathbb{A}^n$ i han de tenir la propietat que la paraula codi $\mathrm{ccdec}(\boldsymbol{x})$ sigui la que ha estat enviada amb probabilitat el més gran possible. Aquestes aplicacions s'estudiaran a la secció 4.2.

Una de les aplicacions de descodificació de codis de bloc més importants, i que és la millor opció en la majoria de canals del mon real, és la descodificació per proximitat, que envia la paraula rebuda $\boldsymbol{x}$ a una paraula codi que estigui a distància mínima:

Definició 1.36 (Descodificació per proximitat)**.** *Sigui $\mathcal{C} \subseteq \mathbb{A}^n$ un codi de bloc de longitud n. S'anomena descodificació per proximitat (o descodificació pel veí més pròxim) una aplicació* $\mathrm{nndec}\colon \mathbb{A}^n \to \mathcal{C}$ *que envia cada paraula $\boldsymbol{x} \in \mathbb{A}^n$ a una paraula codi $\boldsymbol{c} \in \mathcal{C}$ que estigui el més a prop possible de $\boldsymbol{x}$:*

$$\mathrm{nndec}(\boldsymbol{x}) = \boldsymbol{c} \in \mathcal{C} \qquad \textit{tal que} \qquad d(\boldsymbol{x}, \boldsymbol{c}) = d(\boldsymbol{x}, \mathcal{C}) = \min\big\{d(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{y} \in \mathcal{C}\big\}.$$

Aquesta paraula més pròxima de vegades es denota com

$$\mathrm{nndec}(\boldsymbol{x}) = \arg\min_{\boldsymbol{c}\in\mathcal{C}} d(\boldsymbol{x}, \boldsymbol{c}).$$

En estudiar la codificació de canal a la secció 4.2 es veurà que, en la majoria de canals d'interès pràctic, aquesta aplicació de descodificació és una descodificació per *màxima versemblança*, que es denotarà $\mathrm{mldec}\colon \mathbb{A}^n \to \mathcal{C}$, en el sentit que quan s'envia a través del canal una paraula codi $\boldsymbol{c} \in \mathcal{C}$ i es rep una paraula $\boldsymbol{x} \in \mathbb{A}^n$, possiblement amb algunes lletres errònies, la paraula codi $\widehat{\boldsymbol{c}} = \mathrm{nndec}(\boldsymbol{x})$ obtinguda en descodificar per proximitat minimitza la probabilitat d'error $\Pr(\widehat{\boldsymbol{c}} \neq \boldsymbol{c})$:

$$\mathrm{mldec}(\boldsymbol{x}) = \arg\min_{\widehat{\boldsymbol{c}}\in\mathcal{C}} \Pr(\widehat{\boldsymbol{c}} \neq \boldsymbol{c}).$$

En aquesta definició les probabilitats venen donades per una distribució de probabilitats al conjunt de les paraules codi, que diu la probabilitat de fer servir cada paraula en la transmissió, i per una matriu estocàstica associada al canal, que dona les probabilitats que el canal introdueixi errors en cada lletra transmesa.

Descodificació incompleta. També es poden usar aplicacions de descodificació

$$\mathrm{ccdec}\colon \mathbb{A}^n \to \mathcal{C} \sqcup \{*\}$$

que no sempre converteixen la paraula rebuda en una paraula codi, sinó que de vegades declaren que s'ha produït un error en la comunicació que no se sap corregir de manera raonable, i descodifiquen la paraula rebuda amb un valor indeterminat $\mathrm{ccdec}(\boldsymbol{x}) = *$.

Per exemple, una descodificació per proximitat incompleta $\mathrm{nndec}\colon \mathbb{A}^n \to \mathcal{C}\sqcup\{*\}$ dona com a resultat una paraula codi $\mathrm{nndec}(\boldsymbol{x}) = \boldsymbol{c}$ quan existeix una única paraula $\boldsymbol{c} \in \mathcal{C}$ que estigui a distància mínima de la paraula rebuda $\boldsymbol{x}$. En canvi quan hi ha dues o més paraules codi a distància mínima aleshores l'aplicació de descodificació declara que s'ha produït un error: $\mathrm{ccdec}(\boldsymbol{x}) = *$. Aquesta idea es generalitza en l'estratègia de detecció-correcció simultànies, que es descriu més endavant.

Definició 1.37 (Capacitat detectora i correctora)**.** *Es diu que un codi és*

- capaç de detectar *(fins a) s errors si detecta tots els errors que afecten fins a un màxim de s lletres d'una paraula;*
- capaç de corregir *(fins a) t errors si la descodificació per proximitat corregeix tots els errors que afecten fins a un màxim de t lletres d'una paraula.*

Els valors màxims respectius s'anomenen capacitat detectora *i* capacitat correctora *del codi.*

Observi's que el codi $\mathcal{C}$ és capaç de detectar fins a s errors si per a tota paraula codi $\boldsymbol{c} \in \mathcal{C}$ i tota paraula $\boldsymbol{x} \in \mathbb{A}^n$ es té $1 \leqslant d(\boldsymbol{x}, \boldsymbol{c}) \leqslant s \Rightarrow \boldsymbol{x} \notin \mathcal{C}$. És capaç de corregir fins a t errors si per a tota paraula codi $\boldsymbol{c} \in \mathcal{C}$ i tota paraula $\boldsymbol{x} \in \mathbb{A}^n$ es té $d(\boldsymbol{x}, \boldsymbol{c}) \leqslant t \Rightarrow \text{nndec}(\boldsymbol{x}) = \boldsymbol{c}$.

Proposició 1.38. *Sigui $\mathcal{C} \subset \mathbb{A}^n$ un codi de bloc amb distància mínima $d = d(\mathcal{C})$.*

- *La seva capacitat detectora és $d - 1$, i*
- *la seva capacitat correctora és $\tau = \lfloor \frac{d-1}{2} \rfloor$.*

PROVA: Sigui $\boldsymbol{c} \in \mathcal{C}$ i sigui $\boldsymbol{x} \in \mathbb{A}^n$. Suposi's que $1 \leqslant d(\boldsymbol{x}, \boldsymbol{c}) \leqslant d - 1$. Aleshores $\boldsymbol{x} \notin \mathcal{C}$ ja que no pot ser la paraula $\boldsymbol{c}$ per ser la distància $\geqslant 1$ i tampoc pot ser una altra paraula diferent per ser la distància $< d$.

Aquest és el valor més gran possible: d ja no pot ser ja que agafant dues paraules codi a distància mínima es pot agafar com a $\boldsymbol{c}$ l'una i com a $\boldsymbol{x}$ l'altra no es corregirien els errors.

Quant a la correcció, si el nombre d'errors produïts en passar de $\boldsymbol{c}$ a $\boldsymbol{x}$ és $k \leqslant \tau$ aleshores la paraula descodificada $\widehat{\boldsymbol{c}} = \text{nndec}(\boldsymbol{x})$ és igual a $\boldsymbol{c}$. En efecte, en aquest cas es té $d(\boldsymbol{x}, \boldsymbol{c}) = k$ i $d(\widehat{\boldsymbol{c}}, \boldsymbol{x}) = d(\boldsymbol{x}, \mathcal{C}) \leqslant d(\boldsymbol{x}, \boldsymbol{c}) = k$. Per la desigualtat triangular:

$$d(\widehat{\boldsymbol{c}}, \boldsymbol{c}) \leqslant d(\widehat{\boldsymbol{c}}, \boldsymbol{x}) + d(\boldsymbol{x}, \boldsymbol{c}) \leqslant k + k \leqslant d - 1,$$

i això només pot passar si totes dues paraules codi són iguals: $\boldsymbol{c} = \widehat{\boldsymbol{c}}$.

Per veure el recíproc s'ha de veure que el codi no sempre pot corregir $\tau + 1$ errors. Siguin $\boldsymbol{c}, \boldsymbol{c}' \in \mathcal{C}$ dues paraules a distància d. Sigui $d = d_1 + d_2$ amb $d_1 = \tau + 1$. Aleshores

$$d \leqslant 2\tau + 2 \quad \Rightarrow \quad d_2 = d - d_1 \leqslant 2\tau + 2 - \tau - 1 \leqslant \tau + 1 = d_1.$$

Es té, per tant, que $d_2 \leqslant d_1$. Aplicant el lema 1.25 existeix una paraula $\boldsymbol{x} \in \mathbb{A}^n$ "entremig" de les dues paraules agafades, a distàncies $d(\boldsymbol{c}, \boldsymbol{x}) = d_1$ i $d(\boldsymbol{c}', \boldsymbol{x}) = d_2$.

Suposi's que s'ha enviat la paraula $\boldsymbol{c}$ i que s'ha rebut la paraula $\boldsymbol{x}$, de manera que s'han produït $d_1 = \tau + 1$ errors de transmissió. Aleshores la descodificació per proximitat produirà un error ja que la paraula $\boldsymbol{x}$ té una altra paraula del codi $\boldsymbol{c}' \neq \boldsymbol{c}$ a distància d_2 menor o igual que la de la paraula enviada. □

Estratègies de detecció i correcció d'errors. En general es pot fer servir un codi de bloc i l'aplicació de descodificació per proximitat corresponent per detectar i/o corregir errors seguint una de les estratègies següents:

- *Detecció d'errors.* Quan la paraula rebuda $\boldsymbol{x} \in \mathbb{A}^n$ no és del codi el receptor declara que aquesta paraula conté errors i la descarta, o bé, quan això és possible, demana a l'emissor que torni a enviar la informació.
- *Correcció d'errors.* Es fa servir l'aplicació $\text{nndec}\colon \mathbb{A}^n \to \mathcal{C}$ de descodificació per proximitat i, per a cada paraula $\boldsymbol{x}$ rebuda, es decideix que la paraula enviada ha estat $\widehat{\boldsymbol{c}} = \text{nndec}(\boldsymbol{x}) \in \mathcal{C}$.

- *Mixta detecció-correcció simultànies.* Es fixa un llindar t que sigui $0 \leqslant t \leqslant \tau = \lfloor \frac{d-1}{2} \rfloor$. Per a cada paraula rebuda $\boldsymbol{x} \in \mathbb{A}^n$ es calcula $d(\boldsymbol{x}, \mathcal{C})$. Si aquesta distància és $\leqslant t$ es corregeixen els errors usant l'aplicació de descodificació per proximitat; altrament la paraula no es descodifica i es declara que s'ha produït un error de transmissió que el sistema és incapaç de corregir.

Proposició 1.39. *Sigui $\mathcal{C} \in \mathbb{A}^n$ un codi de bloc amb distància mínima $d = d(\mathcal{C})$. Donats enters $s \geqslant t \geqslant 0$ l'estratègia mixta de correcció-detecció d'errors pot corregir t errors i detectar-ne s simultàniament si, i només si, $s + t < d$.*

Prova: Observi's que amb la hipòtesi $s \geqslant t \geqslant 0$ la condició $s + t < d$ implica $2t \leqslant s + t \leqslant d - 1 \leqslant 2\tau + 1$ i, per paritat, ha de ser $2t \leqslant 2\tau \Rightarrow t \leqslant \tau$.

Suposi's que es compleixen la hipòtesi de l'enunciat. Aleshores com que $t \leqslant \tau$ el codi sempre pot corregir t errors, de manera que la part de correcció d'errors de l'algorisme de descodificació funciona bé.

Sigui $\boldsymbol{c}$ la paraula codi enviada i suposi's que el nombre $d(\boldsymbol{x}, \boldsymbol{c})$ d'errors en la transmissió satisfà $t < d(\boldsymbol{x}, \boldsymbol{c}) \leqslant s$. Aleshores, com que $s < d - t \leqslant d$ la paraula rebuda $\boldsymbol{x}$ no pot ser una paraula codi: es té $1 \leqslant d(\boldsymbol{x}, \boldsymbol{c}) < d$. Per tant, l'algorisme detecta fins a s errors. Això assegura que la part de detecció d'errors de l'estratègia de descodificació també funciona correctament.

Ara es veu la implicació recíproca. Si la descodificació mixta sempre pot corregir t errors ha de ser necessàriament $t \leqslant \tau$, ja que altrament es produirien errors de correcció. Suposi's doncs que $t \leqslant \tau$ però que $s + t \geqslant d$. Aleshores $s \geqslant d - t$. Siguin $\boldsymbol{c}$ i $\boldsymbol{c}'$ paraules codi a distància $d(\boldsymbol{c}, \boldsymbol{c}') = d$. Aleshores pel lema 1.25 existeix alguna paraula $\boldsymbol{x} \in \mathbb{A}^n$ "entremig" d'elles amb $d(\boldsymbol{x}, \boldsymbol{c}) = d - t$ i $d(\boldsymbol{x}, \boldsymbol{c}') = t$. Suposi's que en enviar $\boldsymbol{c}$ es rep la paraula $\boldsymbol{x}$. S'han produït $d(\boldsymbol{c}, \boldsymbol{x}) = d - t \leqslant s$ errors i, per tant, l'estratègia de correcció-detecció els hauria de (corregir, que no és el cas, o) detectar. Però no ho fa: com que $d(\boldsymbol{x}, \boldsymbol{c}') = t$ l'estratègia de correcció-detecció decideix descodificar erròniament la paraula $\boldsymbol{x}$ com $\boldsymbol{c}'$. □

Correcció de ràfegues: entrellaçament. Els errors que es produeixen en un canal de comunicació o un dispositiu d'emmagatzemament són de dos tipus: errors *aïllats* i errors en *ràfega*. Els primers afecten només un símbol aïllat, els segons afecten una seqüència de símbols consecutius. Per exemple, en el cas de'un disc òptic (CD, DVD, ...) els errors aleatoris provenen de defectes en el procés de fabricació del motlle, partícules o bombolles microscòpiques en el plàstic del recobriment, etc. En canvi, els errors de ràfega són deguts al desgast del suport amb l'ús (ratllades, brutícia, ...), i també a errors del dispositiu de lectura (moviments bruscos, ...).

Els codis correctors són apropiats per corregir errors aleatoris però no ràfegues d'errors.

Hi ha procediments per transmetre o emmagatzemar la informació codificada amb un codi corrector d'errors que aprofiten la capacitat del codi de corregir errors aleatoris per corregir ràfegues d'errors. Un cop codificada, la informació consisteix en una seqüència de paraules codi

$$\boldsymbol{c}_1 \| \boldsymbol{c}_2 \| \boldsymbol{c}_3 \ldots, \qquad \boldsymbol{c}_i = \mathsf{c}_{i,1} \mathsf{c}_{i,2} \cdots \mathsf{c}_{i,n} \in \mathcal{C}.$$

La manera més natural d'enviar o emmagatzemar aquesta informació és fer-ho consecutivament, primer els de la primera paraula, després els de la segona, i així successivament:

$$\mathsf{c}_{1,1} \quad \mathsf{c}_{1,2} \quad \cdots \quad \mathsf{c}_{1,n} \quad \mathsf{c}_{2,1} \quad \mathsf{c}_{2,2} \quad \cdots \quad \mathsf{c}_{2,n} \quad \mathsf{c}_{3,1} \quad \mathsf{c}_{3,2} \quad \cdots$$

El procediment d'*entrellaçament* (*interleaving*) de profunditat s transmet les paraules del codi per paquets de s paraules. Cada paquet de s paraules es transmet de la manera següent: primer el primer símbol de totes les s paraules, a continuació el segon símbol de totes les paraules, i així successivament fins a transmetre l'últim símbol de totes les paraules:

$$\begin{array}{cccccccccccccc}
\mathsf{c}_{1,1} & \mathsf{c}_{2,1} & \cdots & \mathsf{c}_{s,1} & \mathsf{c}_{1,2} & \mathsf{c}_{2,2} & \cdots & \mathsf{c}_{s,2} & \cdots & \mathsf{c}_{1,n} & \mathsf{c}_{2,n} & \cdots & \mathsf{c}_{s,n} & \\
\mathsf{c}_{s+1,1} & \mathsf{c}_{s+2,1} & \cdots & \mathsf{c}_{2s,1} & \mathsf{c}_{s+1,2} & \mathsf{c}_{s+2,2} & \cdots & \mathsf{c}_{2s,2} & \cdots & \mathsf{c}_{s+1,n} & \mathsf{c}_{s+2,n} & \cdots & \mathsf{c}_{2s,n} & \cdots
\end{array}$$

Es pot representar escrivint les paraules en una taula amb s files de la manera següent:

$$\begin{array}{ccccccccccc}
\mathsf{c}_{1,1} & \mathsf{c}_{1,2} & \cdots & \mathsf{c}_{1,n} & \mathsf{c}_{s+1,1} & \mathsf{c}_{s+1,2} & \cdots & \mathsf{c}_{s+1,n} & \mathsf{c}_{2s+1,1} & \cdots \\
\mathsf{c}_{2,1} & \mathsf{c}_{2,2} & \cdots & \mathsf{c}_{2,n} & \mathsf{c}_{s+2,1} & \mathsf{c}_{s+2,2} & \cdots & \mathsf{c}_{s+2,n} & \mathsf{c}_{2s+2,1} & \cdots \\
\vdots & \vdots & & \vdots & \vdots & \vdots & & \vdots & \vdots & \\
\mathsf{c}_{s,1} & \mathsf{c}_{s,2} & \cdots & \mathsf{c}_{s,n} & \mathsf{c}_{2s,1} & \mathsf{c}_{2s,2} & \cdots & \mathsf{c}_{2s,n} & \mathsf{c}_{3s,1} & \cdots
\end{array}$$

i fent la transmissió per columnes.

Si el codi de bloc $\mathcal{C}$ usat té distancia mínima d i capacitat correctora $\tau = \lfloor (d-1)/2 \rfloor$, l'entrellaçat de profunditat s permet corregir ràfegues de fins a ds errors consecutius en la seqüència transmesa, sempre que cada paraula del codi estigui afectada només per una d'aquestes ràfegues.

Entrellaçament amb retard. L'*entrellaçament amb retard* (*delayed interleaving*) és una variant que permet distribuir les ràfegues d'errors entre un nombre encara més gran de paraules codi. Sigui n la longitud del codi $\mathcal{C}$. Ara les paraules s'escriuen en n files posant a la fila i-èsima el símbol i-èsim de cada paraula, però fent lliscar les files un cert nombre s de posicions, anomenat *retard*. Un cop organitzada així la informació, la transmissió es fa per columnes.

Per exemple, amb paraules de longitud $n = 4$, en una transmissió amb entrellaçament amb retard de $s = 2$ es crea una taula de la forma següent i després es transmeten les columnes consecutivament

$$\begin{array}{cccccccccc}
\mathsf{c}_{1,1} & \mathsf{c}_{2,1} & \mathsf{c}_{3,1} & \mathsf{c}_{4,1} & \mathsf{c}_{5,1} & \mathsf{c}_{6,1} & \mathsf{c}_{7,1} & \mathsf{c}_{8,1} & \mathsf{c}_{9,1} & \mathsf{c}_{10,1} & \cdots \\
* & * & \mathsf{c}_{1,2} & \mathsf{c}_{2,2} & \mathsf{c}_{3,2} & \mathsf{c}_{4,2} & \mathsf{c}_{5,2} & \mathsf{c}_{6,2} & \mathsf{c}_{7,2} & \mathsf{c}_{8,2} & \cdots \\
* & * & * & * & \mathsf{c}_{1,3} & \mathsf{c}_{2,3} & \mathsf{c}_{3,3} & \mathsf{c}_{4,3} & \mathsf{c}_{5,3} & \mathsf{c}_{6,3} & \cdots \\
* & * & * & * & * & * & \mathsf{c}_{1,4} & \mathsf{c}_{2,4} & \mathsf{c}_{3,4} & \mathsf{c}_{4,4} & \cdots
\end{array}$$

on els $*$ representa un padding de símbols de l'alfabet que es poden inicialitzar de manera convinguda (per exemple amb zeros en un codi lineal), i que després el receptor eliminarà un cop feta la correcció d'errors.

En general la transmissió de les paraules d'un codi de longitud n amb entrellaçament amb retard de s posicions segueix l'esquema següent:

$$\cdots \quad \mathsf{c}_{t,1} \quad \mathsf{c}_{t-s,2} \quad \mathsf{c}_{t-2s,3} \quad \cdots \quad \mathsf{c}_{t-(n-1)s,n} \quad \mathsf{c}_{t+1,1} \quad \mathsf{c}_{t+1-s,2} \quad \mathsf{c}_{t+1-2s,3} \quad \cdots \quad \mathsf{c}_{t+1-(n-1)s,n} \quad \cdots$$

on en aquesta expressió quan el primer subíndex del símbol és < 1 i, per tant, no correspon a una lletra a transmetre, s'hi posa la lletra $*$ convinguda.

Utilitzant aquest tipus de transmissió es poden corregir ràfegues d'errors de longitud fins a $t(sn+1)$, on t és la capacitat correctora del codi $\mathcal{C}$; sempre, és clar, que cada paraula codi estigui afectada només per una ràfega d'errors.

Esborralls. S'anomenen canals amb esborralls els canals que transmeten lletres d'un alfabet $\mathbb{A}$ i es poden rebre lletres d'un alfabet $\mathbb{A} \sqcup \{*\}$ on s'ha afegit un símbol $*$, anomenat esborrall. La idea és que en un canal com aquest el receptor, en rebre una lletra $\mathtt{a} \in \mathbb{A}$ pot estar segur que s'havia enviat aquesta mateixa lletra. Si, en canvi, rep l'esborrall $*$ no sap res sobre la lletra de $\mathbb{A}$ que s'havia enviat.

És a dir, en enviar una seqüència de lletres de l'alfabet $\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3\mathtt{a}_4\mathtt{a}_5\mathtt{a}_6\mathtt{a}_7\mathtt{a}_8\cdots$ es rep una seqüència del tipus $\mathtt{a}_1 * \mathtt{a}_3\mathtt{a}_4\mathtt{a}_5 * * \mathtt{a}_8 \cdots$ Les lletres primera, tercera, quarta, cinquena, vuitena, etc. s'han rebut correctament; en canvi de les lletres segona, sisena, setena, etc. s'ha rebut un esborrall i el receptor no sap quines lletres s'havien enviat.

Els codis de bloc es poden fer servir per corregir els esborralls intentant, a partir de la paraula rebuda, recuperar la paraula enviada. L'aplicació de descodificació és anàloga al cas d'un canal amb errors: s'agafa la paraula codi que estigui més a prop de la paraula rebuda

$$\mathrm{nndec}\colon \big(\mathbb{A} \sqcup \{*\}\big)^n \to \mathbb{C}, \qquad \mathrm{nndec}(\boldsymbol{x}) = \arg\min_{\boldsymbol{c}\in\mathbb{C}} d(\boldsymbol{x}, \boldsymbol{c}).$$

En aquest cas, saber quina és la lletra de la paraula codi en totes les posicions en què no hi ha hagut esborrall permet una capacitat correctora superior:

Proposició 1.40. *Un codi amb distància mínima d és capaç de corregir fins a $d-1$ esborralls.*

Prova: Sigui $\boldsymbol{c} = \mathtt{c}_1\mathtt{c}_2\cdots\mathtt{c}_n$ la paraula enviada i $\boldsymbol{x} = \mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n$ la paraula rebuda. Suposi's que s'han produït $r < d$ esborralls. Siguin $i_1, i_2, \ldots, i_r$ els índexs corresponents. És a dir, $\mathtt{x}_i = \mathtt{c}_i$ per a $i \neq i_1, \ldots, i_r$ i $\mathtt{x}_i = *$ per a $i = i_1, \ldots, i_r$. Aleshores $d(\boldsymbol{c}, \boldsymbol{x}) = r$. Suposi's que $\boldsymbol{c}'$ és una altra paraula codi i suposi's que $d(\boldsymbol{c}', \boldsymbol{x}) \leqslant r$. Aleshores, com que $\boldsymbol{x}$ té r lletres iguals a $*$, que no forma part de l'alfabet $\mathbb{A}$, necessàriament la distància ha de ser igual a r i les lletres diferents són les de les posicions i_j. Es dedueix que $\boldsymbol{c}_i = \boldsymbol{x}_i = \boldsymbol{c}'_i$ per a tot $i \neq i_j$ i per tant $d(\boldsymbol{c}, \boldsymbol{c}') \leqslant r < d \Rightarrow \boldsymbol{c}' = \boldsymbol{c}$. Per tant,

$$\boldsymbol{c} = \arg\min_{\boldsymbol{c}'\in\mathbb{C}} d(\boldsymbol{x}, \boldsymbol{c}')$$

i la paraula rebuda es descodifica correctament.

No es poden corregir d esborralls. En efecte, donades dues paraules codi $\boldsymbol{c}, \boldsymbol{c}'$ a distància mínima $d(\boldsymbol{c}, \boldsymbol{c}') = d$ sigui $\boldsymbol{x} \in \big(\mathbb{A} \sqcup \{\infty\}\big)^n$ la paraula amb $\mathtt{x}_i = \mathtt{c}_i$ si $\mathtt{c}_i = \mathtt{c}'_i$ i $\mathtt{x}_i = *$ si $\mathtt{c}_i \neq \mathtt{c}'_i$. Aleshores la paraula $\boldsymbol{x}_i$ conté d esborralls i $d(\boldsymbol{x}, \boldsymbol{c}) = d(\boldsymbol{x}, \boldsymbol{c}') = d$, de manera que la descodificació per proximitat no és capaç d'assignar cap paraula codi a la paraula rebuda $\boldsymbol{x}$. □

Modulació i descodificació tova. La correcció d'errors que s'ha descrit fins ara pressuposa que la informació rebuda són seqüències binàries (o de lletres d'un alfabet finit qualsevol) on alguns bits poden ser incorrectes per culpa del soroll en el canal.

En realitat la situació a la pràctica sol ser una mica més complexa. Les dades discretes se solen transmetre a través de canals de comunicació que operen amb informació de natura contínua: senyals elèctrics, electromagnètics, òptics, etc. Per fer això, les dades s'envien *modulant* una *ona portadora*, amb opcions diverses: modulació d'*amplitud*, de *freqüència* o de *fase*. En aplicacions pràctiques se solen codificar alfabets que consisteixen en paraules binàries curtes, de 2 o 3 bits, representant-les amb punts del pla anomenats *constel·lacions*.

Aquesta situació es tradueix en el model següent: en el procés de modulació les lletres de l'alfabet es converteixen en nombres reals. Per exemple els dígits binaris `0` i `1` es poden transformar en 1 i -1 quan es modula l'amplitud, o bé en 0 i π quan es modula la fase. El soroll del canal pertorbarà l'ona i a la sortida els valors hauran canviat per altres nombres reals, en general propers als enviats. Un procés de *demodulació* recupera les dades digitals a partir dels valors rebuts. Per exemple, es pot agafar el bit que correspongui al valor més proper entre els triats per representar-los: si els bits s'han representat amb els nombres reals 1 i -1 aleshores quan el receptor mesura un valor positiu s'agafa el bit `0` i quan es mesura un valor negatiu s'agafa el bit `1`. També es pot optar per una estratègia mixta que inclou esborralls: els valor rebuts de valor absolut $\leqslant \frac{1}{2}$ es declaren com a esborrall i els valors rebuts de valor absolut $> \frac{1}{2}$ es demodulen amb el bit corresponent al seu signe.

En aquest context, quan es transmet informació codificada amb un codi de bloc corrector d'errors $\mathcal{C} \subseteq \mathbb{A}^n$, el procés de comunicació és el següent:

- les dades discretes a transmetre es codifiquen amb paraules del codi $\boldsymbol{c} \in \mathcal{C}$;
- la modulació transforma les paraules $\boldsymbol{c}$ en vectors n-dimensionals $\boldsymbol{x} \in \mathbb{R}^n$;
- els vectors $\boldsymbol{x}$ s'envien a través del canal i es reben vectors $\boldsymbol{y} \in \mathbb{R}^n$ on possiblement les coordenades han sofert canvis per culpa del soroll del canal;
- cada vector $\boldsymbol{y}$ es converteix en una paraula codi $\widehat{\boldsymbol{c}} \in \mathcal{C}$ procurant minimitzar la probabilitat d'error $\Pr(\widehat{\boldsymbol{c}} \neq \boldsymbol{c})$.

Aquesta descodificació de la paraula $\boldsymbol{y}$ rebuda en una paraula codi es pot fer de dues maneres:

1. primer es demodula $\boldsymbol{y}$ convertint-la en una paraula de $\mathbb{A}^n$, amb components que ja són lletres de l'alfabet $\mathbb{A}$ i no pas nombres reals, i després es descodifica aquesta paraula convertint-la en una del codi, per exemple usant descodificació per proximitat; o bé
2. es descodifica $\boldsymbol{y}$ directament en una paraula codi, sense un pas previ de demodulació.

Aquestes dues opcions es coneixen en teoria de codis amb els noms de *descodificació dura* (hard decoding, o hard-decision decoding) i *descodificació tova* (soft decoding o *soft-decision decoding*). Es poden representar de la manera següent:

$$\mathcal{M} \xleftrightarrow{\text{enc}} \mathcal{C} \subseteq \mathbb{A}^n \xrightarrow{\text{modul.}} \mathbb{R}^n \xrightarrow{\text{canal}} \mathbb{R}^n \xrightarrow{\text{demod.}} \mathbb{A}^n \xrightarrow{\text{ccdec}} \mathcal{C} \xleftrightarrow{\text{dec}} \mathcal{M}$$

és la descodificació dura i

$$\mathcal{M} \xleftrightarrow{\text{enc}} \mathcal{C} \subseteq \mathbb{A}^n \xrightarrow{\text{modul.}} \mathbb{R}^n \xrightarrow{\text{canal}} \mathbb{R}^n \xrightarrow{\text{ccdec}} \mathcal{C} \xleftrightarrow{\text{dec}} \mathcal{M}$$

és la descodificació tova.

La descodificació tova aprofita millor la informació que conté la paraula rebuda $\boldsymbol{y} \in \mathbb{R}^n$, que es perd en el procés de demodulació quan es converteix en una paraula de $\mathbb{A}^n$. Per exemple, quan es representen els bits amb els nombres 1 i -1, si una component de la paraula rebuda és $\mathtt{y}_i \approx 0.99312$ i una altra és $\mathtt{y}_j \approx 0.17123$, totes dues es demodularien assignant el bit $\mathtt{1}$, però és molt més probable que hi hagi hagut un error en el segon que no pas en el primer d'aquests dos bits, ja que el nombre 0.99312 és molt proper a 1 i, en canvi el nombre 0.17123, tot i ser positiu, no està gaire més a prop de 1 que de -1.

La descodificació tova que se sol aplicar per a la correcció d'errors és la descodificació per proximitat, anàloga a la descodificació dura per proximitat. Aquí, però, es parteix d'un vector $\boldsymbol{y} \in \mathbb{R}^n$ i es busca el vector de $\mathbb{R}^n$ més proper que sigui una paraula del codi, vist com a subconjunt $\mathcal{C} \subset \mathbb{R}^n$ a través de la modulació. En aquest cas no s'agafa la distància de Hamming sinó la distància euclidiana ordinària

$$\operatorname{nndec}(\boldsymbol{x}) = \operatorname*{arg\,min}_{\boldsymbol{c} \in \mathcal{C} \subset \mathbb{R}^n} \|\boldsymbol{x} - \boldsymbol{c}\|.$$

Com que l'aplicació elevar al quadrat és creixent, en realitat els càlculs es poden fer amb el quadrat de la norma euclidiana $\|\boldsymbol{x} - \boldsymbol{c}\|^2 = \sum(\mathtt{x}_i - \mathtt{c}_i)^2$ estalviant calcular l'arrel quadrada.

Problemes

1.16. Es considera el codi de bloc binari

$$\mathcal{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3, \boldsymbol{c}_4\} = \{\mathtt{000000}, \mathtt{001111}, \mathtt{111100}, \mathtt{110011}\} \subseteq \{\mathtt{0}, \mathtt{1}\}^6$$

i les paraules $\boldsymbol{x}_1 = \mathtt{010100}$, $\boldsymbol{x}_2 = \mathtt{111110}$, $\boldsymbol{x}_3 = \mathtt{000100}$, $\boldsymbol{x}_4 = \mathtt{111100}$, $\boldsymbol{x}_5 = \mathtt{111111}$.

1. Calculeu la distància mínima d i el radi de tangència τ del codi $\mathcal{C}$.
2. Apliqueu descodificació per proximitat nndec: $\{\mathtt{0}, \mathtt{1}\}^6 \to \mathcal{C}$ a les paraules $\boldsymbol{x}_i$.
3. Apliqueu descodificació incompleta per proximitat nndec: $\{\mathtt{0}, \mathtt{1}\}^6 \to \mathcal{C} \cup \{*\}$ a les paraules $\boldsymbol{x}_i$.

1.17. Per al codi binari de repetició $\operatorname{Rep}_2(n)$ digueu com es poden calcular la descodificació completa per proximitat, la descodificació incompleta per proximitat i la tècnica mixta de correcció/detecció que corregeixi només els errors que afecten $t \leqslant \tau$ lletres en funció del nombre d'uns de la paraula rebuda.

1.18. *Descodificació tova.* Es transmet informació binària a través d'un canal amb soroll usant el codi de repetició de longitud 5:

$$\operatorname{Rep}_2(5) = \{\mathtt{00000}, \mathtt{11111}\}.$$

Apliqueu descodificació tova per proximitat en els exemples següents:

1. L'alfabet binari $\{\mathtt{0}, \mathtt{1}\}$ es representa amb els nombres reals 1 i -1 i es rep
   (a) $\boldsymbol{y} = (1.05, 1.5, -0.5, -1.3, -0.7)$;

(b) $\boldsymbol{y} = (1, 1, -1, 1, 1)$;
(c) $\boldsymbol{y} = (0.8, -0.8, 0.8, -0.4, -0.4)$.

2. L'alfabet binari $\{\mathtt{0}, \mathtt{1}\}$ es representa amb els angles 0 i π i es rep
   (a) $\boldsymbol{y} = (\frac{\pi}{5}, \frac{\pi}{6}, \frac{\pi}{7}, \frac{7\pi}{8}, \frac{3\pi}{4})$;
   (b) $\boldsymbol{y} = (\frac{\pi}{4}, \frac{\pi}{4}, \pi, \pi, \frac{7\pi}{4})$.

En cada cas, calculeu també la descodificació dura per proximitat si primerament es demodula la paraula rebuda.

1.6 Dígits de verificació

Referència: Brunat-Ventura [3, Capítol 7]

En moltes situacions quotidianes es fan servir identificadors alfanumèrics: número de DNI o passaport, números de compte corrent o de targeta de crèdit, ISBN d'un llibre, identificadors d'articles de supermercat en codis de barres, etc. Per evitar confondre aquests identificadors quan es donen a algú altre de paraula o per escrit, s'introdueixen usant un teclat d'un aparell electrònic, o es llegeixen per procediments mecànics o òptics, se'ls afegeixen un o més dígits o símbols redundants que es coneixen com a *dígits de verificació*.

Exemples ben coneguts són els codis d'identificació de persones (número de *DNI* o passaport), números de comptes de bancs (*IBAN*: International Bank Account Number), números de targetes de crèdit, codis d'identificació de llibres (*ISBN*: International Standard Book Number) o codis de barres que identifiquen productes (*UPC*: Universal Product Code; *EAN*: European Article Number), etc.

En afegir els dígits de verificació a un identificador s'obté un codi detector d'errors. Aquests codis són molt semblants als codis de paritat $\mathrm{Par}_q(n)$ dels exemples 1.20 tot i que aquí en comptes de sumar directament els dígits se solen fer combinacions lineals amb coeficients prefixats. Són codis sobre alfabets q-aris $\mathbb{A}$ que s'identifiquen amb l'anell $\mathbb{Z}_q$ de les classes de congruència mòdul q, de manera que les lletres es poden sumar i multiplicar entre elles, i que es defineixen a partir d'una congruència lineal de la manera següent:

$$\mathcal{C} = \left\{ (\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_n) \in \mathbb{Z}_q^n : \sum_{i=1}^{n} \mathtt{c}_i \mathtt{x}_i \equiv 0 \pmod{q} \right\}$$

on els coeficients $\mathtt{c}_i \in \mathbb{Z}_q$ estan fixats. El cas del codi de paritat $\mathrm{Par}_q(n)$ correspon a agafar tots els coeficients $\mathtt{c}_i$ iguals a 1.

Els $n-1$ primers dígits $\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_{n-1}$ de les paraules codi són l'identificador i l'últim símbol $\mathtt{x}_n$ és el dígit de verificació, que es pot calcular a partir dels altres com

$$\mathtt{x}_n = \mathtt{c}_n^{-1} \sum_{i=1}^{n-1} -\mathtt{c}_i \mathtt{x}_i \pmod{q}.$$

Observi's que per poder fer això cal que el coeficient $\mathtt{c}_n$ sigui invertible mòdul q.

En alguns casos els identificadors s'escriuen en un alfabet que té menys de q lletres i per tant el dígit de verificació pot pertànyer a un alfabet més gran. Per exemple, el NIF té com a

identificador el DNI, que és un nombre de 8 dígits decimals (una paraula de longitud 8 sobre un alfabet de 10 lletres) i el mòdul és $q = 23$; el dígit de verificació és una lletra de l'alfabet llatí diferent de I, O i U. En l'ISBN de 10 símbols l'identificador és un nombre de 9 dígits decimals i el mòdul és $q = 11$; el dígit de verificació és un dígit decimal o bé la lletra X, que correspon al $10 \in \mathbb{Z}_{11}$.

La mateixa idea es fa servir en els *CRC* (cyclic redundancy check) que detecten errors en memòries i sistemes de comunicacions digitals, on en comptes de fer congruències amb mòdul un nombre enter q es fan congruències mòdul polinomis binaris. Els CRC són exemples de codis polinomials i es discutiran més endavant a la secció 6.2.

Codi EAN. El codi *EAN*: European article number o International article number és un identificador d'articles comercials que s'imprimeix sobre l'article amb en un codi de barres de manera que pugui ser llegit fàcilment de manera automàtica amb un lector òptic. Com que en la lectura òptica de codis de barres es produeixen errors freqüents el codi incorpora un dígit de verificació.

El codi EAN d'un article té 13 dígits decimals $\mathtt{x}_1\mathtt{x}_2 \dots \mathtt{x}_{12}\mathtt{x}_{13}$, de manera que $q = 10$. Els primers 12 dígits identifiquen el producte. L'últim dígit és un dígit de verificació, que es calcula amb una congruència mòdul 10. L'equació lineal que determina el codi té coeficients $\mathtt{c}_i = 1$ i 3 alternats en les posicions senars i parells, respectivament:

$$\mathtt{x}_1 + 3\mathtt{x}_2 + \mathtt{x}_3 + 3\mathtt{x}_4 + \mathtt{x}_5 + \cdots + 3\mathtt{x}_{12} + \mathtt{x}_{13} = \sum_{i=0}^{6} \mathtt{x}_{2i+1} + 3\sum_{\mathtt{i}=1}^{6} \mathtt{x}_{2\mathtt{i}} \equiv 0 \pmod{10}.$$

Per tant el dígit de verificació es pot calcular a partir de l'identificador amb la fórmula

$$\mathtt{x}_{13} = 9\sum_{\mathtt{i}=0}^{5} \mathtt{x}_{2\mathtt{i}+1} + 7\sum_{\mathtt{i}=1}^{6} \mathtt{x}_{2\mathtt{i}} \pmod{10},$$

tenint en compte que $-1 \equiv 9 \pmod{10}$ i que $-3 \equiv 7 \pmod{10}$.

El codi EAN forma part de l'estàndard *GTEM* (Global trade item number) definit per l'organització *GS1*.

Codi ISBN 10. El codi *ISBN* 10 (International standard book number) és un codi de deu símbols assignat a cada llibre que es publica. L'identificador del llibre està escrit en dígits decimals, però es treballa amb congruències mòdul $q = 11$. Per representar l'element $10 \in \mathbb{Z}_{11}$, que no correspon a cap dígit decimal, es fa servir el signe X, que només pot aparèixer en l'última posició del codi, que és la corresponent al dígit de verificació.

Els nou dígits decimals de l'identificador estan separats en tres blocs per guions. Corresponen a la llengua en que està escrita la publicació, l'editorial, i, finalment, el llibre mateix entre tots els publicats per aquesta mateixa editorial. El desè símbol del codi és el dígit de verificació; pot ser un dígit decimal o també el símbol X.

Per exemple, `0-486-66521-6` és el codi ISBN del llibre "Information Theory" de Robert B. Ash. El primer nombre `0` correspon a la llengua anglesa, el segon `486` a l'editorial Dover

Publications i el tercer nombre 66521 identifica el llibre. El dígit 6 del final és el dígit de verificació.

La fórmula per calcular el dígit de verificació és la següent: siguin $\mathtt{x}_1\mathtt{x}_2\mathtt{x}_3\mathtt{x}_4\mathtt{x}_5\mathtt{x}_6\mathtt{x}_7\mathtt{x}_8\mathtt{x}_9$ els 9 dígits decimals que identifiquen el llibre i sigui $\mathtt{x}_{10}$ el dígit de verificació. Aleshores:

$$\mathtt{x}_{10} = \sum_{i=1}^{9} i\mathtt{x}_i = \mathtt{x}_1 + 2\mathtt{x}_2 + 3\mathtt{x}_3 + 4\mathtt{x}_4 + 5\mathtt{x}_5 + 6\mathtt{x}_6 + 7\mathtt{x}_7 + 8\mathtt{x}_8 + 9\mathtt{x}_9 \pmod{11}.$$

És a dir, els coeficients $\mathtt{c}_i$ són $\mathtt{c}_i = i$. Com que $\mathbb{Z}_{11}$ conté 11 elements diferents es necessiten 11 símbols per representar-los. Per això apart dels 10 dígits decimals 0,...,9 es fa servir el símbol X quan la suma anterior és 10 mòdul 11. Per exemple, el codi ISBN 0-471-60836-X correspon al llibre "Introduction to Number Theory" de D.E. Flath, editat per John Wiley & Sons.

Sigui $\mathbb{F}$ l'alfabet de 11 símbols 0,1,2,3,4,5,6,7,8,9,X, identificat amb el cos finit de 11 elements. Tenint en compte que $-1 \equiv 10 \pmod{11}$, el fet que una cadena de 10 lletres $\mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_9\mathtt{x}_{10}$ correspongui a un codi ISBN correcte es tradueix en la identitat

$$\sum_{i=1}^{10} i\mathtt{x}_i \equiv 0 \pmod{11}.$$

Per tant, els codis ISBN vàlids pertanyen al subespai de $\mathbb{F}^{10}$ de dimensió 9 format per les paraules que satisfan l'equació lineal anterior. No totes les paraules del subespai són, però, identificadors vàlids, ja que els 9 primers dígits han de ser sempre diferents de X.

Hi ha una altra versió anomenada ISBN 13 que segueix el format dels codis EAN.

Codi DNI/NIF. El número de Document Nacional d'Identitat espanyol és un nombre de (fins a) 8 dígits decimals. El NIF s'obté afegint al DNI un dígit de verificació que és una lletra de l'alfabet llatí diferent de I, O i U. Es treballa a l'anell $\mathbb{Z}_q$ de congruències mòdul $q = 23$. Els dígits decimals s'identifiquen amb ells mateixos com a elements de $\mathbb{Z}_{23}$. Les lletres del dígit de verificació s'identifiquen amb els elements de $\mathbb{Z}_{23}$ segons la *taula següent*:

0	T	4	G	8	P	12	N	16	Q	20	C
1	R	5	M	9	D	13	J	17	V	21	K
2	W	6	Y	10	X	14	Z	18	H	22	E
3	A	7	F	11	B	15	S	19	L		

La lletra es calcula reduint el número de DNI mòdul 23 i usant aquesta taula.

Codi CCC i IBAN. Els codis *IBAN* són un estàndard internacional per a identificar comptes bancaris. Són seqüències de símbols alfanumèrics de fins a 34 caràcters: els dos primers són lletres que identifiquen el país (ES correspon a Espanya); després hi ha dos dígits decimals que són dígits de verificació; finalment ve una seqüència de fins a 30 símbols que identifica el compte en un format que depèn del país i que pot contenir o no dígits de verificació, anomenat BBAN (Basic bank account number). A Espanya el BBAN es coneix

com a CCC (*Codi compte client*) i està format per 20 dígits decimals dos dels quals són dígits de verificació.

El format del CCC és el següent: els quatre primers dígits identifiquen l'entitat bancària, els quatre següents la sucursal, després venen dos dígits de verificació i, finalment, els deu últims dígits identifiquen el compte dins de la oficina de la manera que cada banc determini. El CCC és la concatenació de dues paraules que són codis mòdul 11, l'un de longitud 9 i l'altre de longitud 11. Escrivint-lo de la manera següent:

$$\mathtt{x}_2\mathtt{x}_3\mathtt{x}_4\mathtt{x}_5\,\mathtt{x}_6\mathtt{x}_7\mathtt{x}_8\mathtt{x}_9\,\mathtt{x}_{10}\,\mathtt{y}_{10}\,\mathtt{y}_0\mathtt{y}_1\mathtt{y}_2\mathtt{y}_3\mathtt{y}_4\mathtt{y}_5\mathtt{y}_6\mathtt{y}_7\mathtt{y}_8\mathtt{y}_9$$

aleshores els dígits de verificació $\mathtt{x}_{10}$ i $\mathtt{y}_{10}$ queden determinats mòdul 11 per les identitats

$$\sum_{i=2}^{10} 2^i \mathtt{x}_i \equiv 0 \pmod{11}, \qquad \sum_{i=0}^{10} 2^i \mathtt{y}_i \equiv 0 \pmod{11}.$$

Com que no està previst cap símbol per al nombre 10 mòdul 11 com a l'ISBN, aquí no totes les paraules $\mathtt{x}_2 \ldots \mathtt{x}_9$ es poden fer servir per identificar parells banc-sucursal ni totes les paraules $\mathtt{y}_0 \ldots \mathtt{y}_9$ es poden fer servir per identificar números de compte de clients.

Finalment els dos dígits de verificació que formen part de l'IBAN es calculen de la manera següent: les lletres que identifiquen el país s'afegeixen al final del BBAN i després s'hi posen dos zeros; les lletres que apareixen en aquesta seqüència es transformen en nombres usant la regla $\mathtt{A} \mapsto 10, \mathtt{B} \mapsto 11, \ldots, \mathtt{Z} \mapsto 35$ (cada lletra s'ha transformat en un nombre de dos dígits); es redueix el nombre que resulta d'aquesta substitució mòdul 97 escrivint el resultat amb dos dígits, posant zeros a l'esquerra, si cal. Els dígits de verificació s'obtenen restant de 98 aquest nombre.

Per exemple, es considera el compte a la Caixa (codi d'entitat $\mathtt{x}_2\mathtt{x}_3\mathtt{x}_4\mathtt{x}_5 = \mathtt{2100}$) de l'oficina de Diagonal 530 (codi de sucursal $\mathtt{x}_6\mathtt{x}_7\mathtt{x}_8\mathtt{x}_9 = \mathtt{5000}$) número $\mathtt{y}_0\mathtt{y}_1\mathtt{y}_2\mathtt{y}_3\mathtt{y}_4\mathtt{y}_5\mathtt{y}_6\mathtt{y}_7\mathtt{y}_8\mathtt{y}_9 = \mathtt{0200227591}$. Tenint en compte que $2^{10} \equiv 1 \pmod{11}$ i, per tant, els coeficients de $\mathtt{x}_{10}$ i $\mathtt{y}_{10}$ en l'equació lineal que defineix el codi és igual a 1, els dígits de verificació es calculen com:

$$\mathtt{x}_{10} \equiv -\sum_{i=2}^{9} 2^i \mathtt{x}_i \equiv -(2^2 2 + 2^3 1 + 2^6 5) \equiv 5 \pmod{11},$$

$$\mathtt{y}_{10} \equiv -\sum_{i=0}^{9} 2^i \mathtt{y}_i \equiv -(2^1 2 + 2^4 2 + 2^5 2 + 2^6 7 + 2^7 5 + 2^8 9 + 2^9 1) \equiv 0 \pmod{11}.$$

Per tant, el CCC és $\mathtt{2100\,5000\,5002\,0022\,7591}$. Ara, per calcular l'IBAN s'ha d'afegir a la dreta d'aquest nombre les lletres $\mathtt{ES}$, convertides en els nombres $\mathtt{14}$ i $\mathtt{28}$ seguides de dos zeros i calcular:

$$210050005002002275911428 00 \pmod{97} = 15, \qquad 98 - 15 = 83.$$

Per tant els dígits de verificació són $\mathtt{83}$ i l'IBAN del compte és

$$\mathtt{ES83\,2100\,5000\,5002\,0022\,7591}.$$

Problemes

1.19. Es considera un sistema de dígit de verificació mòdul q que converteix identificadors de $n-1$ dígits en codis de n dígits que satisfan la congruència

$$\sum_{i=1}^{n} \mathtt{c}_i \mathtt{x}_i \equiv 0 \pmod{q}, \qquad \mathtt{c}_i, \mathtt{x}_i \in \mathbb{Z}_q.$$

1. Demostreu que si $\mathtt{c}_r$ és invertible mòdul q aleshores es detecten tots els errors que afectin només el símbol $\mathtt{x}_r$.
2. Demostreu que si $\mathtt{c}_r - \mathtt{c}_s$ és invertible mòdul q aleshores es detecten totes les transposicions de símbols diferents $\mathtt{x}_r \leftrightarrow \mathtt{x}_s$.
3. Quan $\mathtt{c}_r$ o $\mathtt{c}_r - \mathtt{c}_s$ no són invertibles mòdul q, quins errors i transposicions no es detecten?

Discutiu les implicacions d'això en els codis CCC, ISBN i EAN. Compareu CCC amb l'*algorisme de Luhn*, que es fa servir per exemple en els nombres de targetes de crèdit.

1.20. Es considera un codi

$$\mathcal{C} = \left\{ (\mathtt{x}_1, \ldots, \mathtt{x}_n) \in \mathbb{Z}_q^n : \sum_{i=1}^{n} \mathtt{c}_i \mathtt{x}_i \equiv 0 \pmod{q}, \right\}$$

Calculeu els paràmetres d'aquest codi en el cas que existeixi un índex r tal que el coeficient $\mathtt{c}_r$ és invertible mòdul q.

1.21. Calculeu el nombre $\mathtt{x}_9$ mòdul 23 que determina la lletra del DNI amb una expressió de la forma

$$\mathtt{x}_9 = \sum_{i=1}^{8} \mathtt{c}_i \mathtt{x}_i,$$

on el número de DNI és $n = \mathtt{x}_1\mathtt{x}_2\mathtt{x}_3\mathtt{x}_4\mathtt{x}_5\mathtt{x}_6\mathtt{x}_7\mathtt{x}_8 = \sum_{i=1}^{8} \mathtt{x}_i \cdot 10^{8-i}$, amb els $\mathtt{x}_i$ els seus dígits decimals.

1.22. En escriure un NIF usant un teclat s'ha comès un error en teclejar un dels dígits: s'ha picat la tecla que està a la dreta o a l'esquerra de la correcta, augmentant o disminuint el dígit en 1. Si el NIF erroni entrat és `41586126E`, quin és el NIF correcte?

Comproveu que aquest tipus d'error en un NIF (un error de magnitud 1 en un dígit) sempre es podrà corregir.

1.23. Comproveu els dígits de verificació de l'IBAN d'un compte vostre i calculeu l'IBAN del compte número `0001062212` de la sucursal `0603` de l'entitat `0081`.

1.7 Problemes Complementaris

1.24. *Preparació per a la construcció de codis de Huffman q-aris.* Sigui $\mathbb{A}$ un alfabet q-ari i sigui $\mathcal{C} \subseteq \mathbb{A}^*$ un codi. Sigui $m = \max\{\ell(\boldsymbol{c}) : \boldsymbol{c} \in \mathcal{C}\}$ la longitud màxima de les seves paraules i sigui $N_m = \big|\{\boldsymbol{c} \in \mathcal{C} : \ell(\boldsymbol{c}) = m\}\big|$ el nombre de paraules que tenen aquesta longitud. Demostreu que

1. si $\mathcal{C}$ no és maximal se li pot afegir almenys una paraula de longitud m de tal manera que segueixi sent un codi;
2. es pot arribar a un codi maximal afegint paraules de longitud m al codi $\mathcal{C}$;
3. si el codi és maximal aleshores N_m és divisible per q;
4. si $\mathcal{C}$ és prefix i complet, per a cada paraula $\boldsymbol{c} \in \mathcal{C}$ de longitud màxima m, el codi ha de contenir totes les q paraules amb el mateix prefix de longitud $m-1$ que $\boldsymbol{c}$;
5. si $\mathcal{C}$ és maximal aleshores $|\mathcal{C}| \equiv 1 \pmod{q-1}$;
6. existeix un codi prefix amb paraules de les mateixes longituds que $\mathcal{C}$ tals que si $N_m = qt + r$ és la divisió euclidiana, amb quocient $t \geqslant 0$ i reste $0 \leqslant r < q$, les paraules de longitud m d'aquest codi prefix estan agrupades en
   - t conjunts, cadascun de q paraules que només difereixen en l'última lletra, i
   - si $r > 0$ un conjunt de r paraules que només difereixen en l'última lletra.

1.25. *Retallar i escurçar.* Sigui $\mathcal{C} \subseteq \mathbb{A}^n$ un codi de bloc de longitud n. El *codi retallat* (punctured code) en la darrera posició és el subconjunt

$$\big\{(\mathtt{a}_1, \ldots, \mathtt{a}_{n-1}) : (\mathtt{a}_1, \ldots, \mathtt{a}_n) \in \mathcal{C}\big\} \subseteq \mathbb{A}^{n-1}$$

format per les paraules de longitud $n-1$ que s'obtenen en treure a les paraules del codi $\mathcal{C}$ la seva última lletra.

Fixat un element $\mathtt{a} \in \mathbb{A}$ el *codi escurçat* (shortened code) en la darrera posició respecte aquesta lletra és el subconjunt

$$\big\{(\mathtt{a}_1, \ldots, \mathtt{a}_{n-1}) : (\mathtt{a}_1, \ldots, \mathtt{a}_n) \in \mathcal{C}, \mathtt{a}_n = \mathtt{a}\big\} \subseteq \mathbb{A}^{n-1}$$

format per les paraules de longitud $n-1$ tals que en afegir-los al final la lletra $\mathtt{a}$ resulta una paraula del codi $\mathcal{C}$.

De manera anàloga es defineixen els codis retallats o escurçats en una altra o en varies posicions.

Discutiu els paràmetres dels codis retallats i escurçats en funció dels del codi $\mathcal{C}$.

1.26. *Codis equivalents.* Dos codis de bloc $\mathcal{C}, \mathcal{C}' \subseteq \mathbb{A}^n$ es diuen *equivalents* (notació $\mathcal{C} \sim \mathcal{C}'$) si es pot obtenir un a partir de l'altre fent transformacions dels tipus següents:

- permutar les lletres en una posició fixada: $\mathtt{x}_1 \cdots \mathtt{x}_i \cdots \mathtt{x}_n \mapsto \mathtt{x}_1 \cdots \phi(\mathtt{x}_i) \cdots \mathtt{x}_n$, amb $\phi\colon \mathbb{A} \to \mathbb{A}$ una permutació (bijecció) de $\mathbb{A}$;

– permutar les lletres de les diferents posicions: $\mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n \mapsto \mathtt{x}_{\sigma(1)}\mathtt{x}_{\sigma(2)}\cdots\mathtt{x}_{\sigma(n)}$, on $\sigma \in \mathfrak{S}_n$ és una permutació de $\{1, 2, \ldots, n\}$.

1. Demostreu que $\mathcal{C} \sim \mathcal{C}'$ si, i només si, existeixen n permutacions $\phi_i \colon \mathbb{A} \to \mathbb{A}$ de l'alfabet i una permutació $\sigma \in \mathfrak{S}_n$ dels índexs $\{1, \ldots, n\}$ tals que

$$\mathcal{C}' = \left\{\phi_1(\mathtt{x}_{\sigma(1)})\cdots\phi_i(\mathtt{x}_{\sigma(i)})\cdots\phi_n(\mathtt{x}_{\sigma(n)}) : \mathtt{x}_1\cdots\mathtt{x}_i\cdots\mathtt{x}_n \in \mathcal{C}_1\right\}.$$

2. Demostreu que dos codis equivalents són del mateix tipus $(n, M, d)_q$.

1.27. Trobeu tots els codis equivalents al codi binari de repetició i al codi binari parell.

1.28. Les equivalències que converteixen un codi en ell mateix s'anomenen *automorfismes* del codi. Comproveu que els automorfismes d'un codi formen un grup i calculeu el grup d'automorfismes del codi de repetició i del codi binari parell.

1.29. Calculeu $A_2(4, 3)$.

1.30. Calculeu $A_2(5, 3)$.

1.31. Demostreu la *Fita de Gilbert-Varshamov*:

$$A_q(n, d) \geqslant \frac{q^n}{\sum_{i=0}^{d-1} \binom{n}{i}(q-1)^i}.$$

2 Informació i entropia

En *Teoria de la Informació* el concepte d'*informació* es defineix com una mesura de la *incertesa* que es té sobre el resultat d'un experiment o, de manera equivalent, una mesura de la quantitat d'informació que s'obté en conèixer quin ha estat aquest resultat.

Si l'experiment està modelitzat per una variable aleatòria X la informació que s'obté en saber que el resultat pertany a un esdeveniment A és una funció decreixent de la seva probabilitat $\Pr(X \in A)$: esdeveniments poc probables proporcionen molta informació i esdeveniments molt probables proporcionen poca informació. Per mesurar-la s'agafa la funció logaritme amb signe negatiu. Aquesta és l'elecció natural si es vol relacionar la informació amb la mida de les dades que calen per representar-la, tal i com queda palès en el teorema de codificació de font, que es veurà a la secció 3.2.

Definició 2.1 (Informació)**.** *Sigui X una variable aleatòria. La quantitat d'*informació *que s'obté d'un esdeveniment A és el logaritme de l'invers de la seva probabilitat:*

$$\log \frac{1}{p(A)} = -\log\big(p(A)\big), \qquad p(A) = \Pr(X \in A).$$

El logaritme s'acostuma a agafar en base 2 i, en aquest cas, la informació es mesura en *bits*. Agafar el logaritme en una altra base equival a canviar d'unitat de mesura. Si s'agafa el logaritme natural (*neperià*) la informació es mesura en *nats* i si s'agafa el logaritme en base 10 es mesura en *dècits*. Mentre no es digui explícitament el contrari se suposarà que tots els logaritmes són en base 2, i, per tant, que la informació s'està mesurant en bits.

Com que $p(A) \in [0,1]$, la informació pren valors $-\log p(A) \in [0,\infty]$. Un esdeveniment segur (probabilitat 1) no dona cap informació; un esdeveniment impossible (probabilitat 0, i que per tant no passarà mai) donaria infinita informació; un esdeveniment de probabilitat $\frac{1}{2}$, per exemple el resultat de llaçar una moneda, dona un bit d'informació; el resultat d'una tirada d'un dau dona $-\log \frac{1}{6} \approx 2.585$ bits d'informació; donat un nombre decimal de cinc dígits, saber que és el premi gros de la loteria dona $-\log 10^{-5} \approx 16.61$ bits d'informació i saber que no ho és dona $-\log(1-10^{-5}) \approx 0.0000144$ bits d'informació.

2.1 Entropia

Referències: Cover-Thomas [4, Chapter 2], Brunat-Ventura[3, Capítol 2], Ball[2, Chapter 1].

Definició 2.2 (Entropia)**.** *L'entropia d'una variable aleatòria és l'esperança de la informació que proporcionen els diferents valors de la variable:*

$$H(X) := \mathbb{E}\big[-\log p(X)\big]$$

L'entropia d'una variable aleatòria s'interpreta com una mesura de la quantitat d'informació que hom obté en observar el resultat d'un experiment governat per aquesta variable. Es pot pensar com la quantitat d'informació "que conté" la variable. Per exemple la variable tirar una moneda dona un bit d'informació; la variable tirar un dau en dona ≈ 2.685; la variable que diu el nombre de tirades de moneda fins a la primera cara en dona 2; la variable que diu si t'ha tocat la grossa de la loteria en dona ≈ 0.00018; la variable que agafa una lletra aleatòriament en un text anglès en dona ≈ 4.09; etc.

Variables discretes. En aquestes notes se suposarà que les variables aleatòries són discretes i prenen només un nombre finit de valors, no necessàriament numèrics. Es denotara $\mathcal{X}$ aquest conjunt de valors. El cas de variables discretes amb nombre numerable de valors és totalment anàleg, canviant sumes finites per sumes de sèries numèriques, amb la diferència que en aquest cas l'entropia pot ser infinita. El cas de les variables contínues és força diferent i s'ha de considerar apart (secció 2.3).

Definició 2.3 (Entropia d'una variable discreta)**.** *Sigui X una variable aleatòria discreta que pren valors en un conjunt $\mathcal{X}$ amb probabilitats $p(x)$ per a $x \in \mathcal{X}$. La seva entropia és*

$$H(X) := \mathbb{E}[-\log p(X)] = \sum_{x\in\mathcal{X}} p(x) \log \frac{1}{p(x)} = -\sum_{x\in\mathcal{X}} p(x) \log p(x).$$

Mentre no es digui el contrari s'agafaran els logaritmes en base 2, de manera que l'entropia d'una variable aleatòria està donada en bits. Algunes vegades, i especialment en l'enunciat del teorema de codificació de font per a fonts q-àries, s'ha de calcular l'entropia amb logaritmes en una altra base q. Es denotarà:

$$H_q(X) = \sum_{x\in\mathcal{X}} p(x) \log_q \frac{1}{p(x)}.$$

Convenis. En estudiar l'entropia apareixeran sovint expressions de la forma $a \log \frac{b}{a}$ amb $a, b \geqslant 0$ i també $a \log \frac{a}{b}$, que per les propietats del logaritme ha de ser igual a $-a \log \frac{b}{a}$. En el cas $a = b = 0$ es convé que aquestes expressions prenen el valor zero. Quan només un dels dos nombres a i b és igual a zero s'agafa el límit corresponent. Així, es té que $0 \log \frac{b}{0} = 0 \log \frac{0}{b} = 0$ per a tot $b > 0$ i que $a \log \frac{0}{a} = -\infty$ i $a \log \frac{a}{0} = \infty$ si $a > 0$.

En particular si hi ha valors $x \in \mathcal{X}$ que són impossibles l'entropia de la variable no canvia si s'eliminen aquests valors del conjunt $\mathcal{X}$ i es canvia la variable per una altra que prengui valors només en el suport de X.

Exemple 2.4 (Distribució uniforme)**.** *L'entropia d'una variable que pren n valors amb distribució uniforme $X \sim \mathrm{Unif}(n) \sim (\frac{1}{n}, \ldots, \frac{1}{n})$ és*

$$H(X) = \log n.$$

Prova: En efecte,

$$H(X) = \sum_{x\in\mathcal{X}} p(x) \log \frac{1}{p(x)} = \sum_{x\in\mathcal{X}} \frac{1}{n} \log n = n\frac{1}{n} \log n = \log n.$$

Es veurà més endavant que aquesta és l'entropia màxima que pot assolir una variable que prengui n valors diferents. □

Exemple 2.5. *Una* variable constant *és una variable que només pren un valor amb probabilitat $\neq 0$. L'entropia d'una variable és zero si, i només si, és una variable constant.*

Prova: En el sumatori de l'entropia tots els sumands són $\geqslant 0$. Si la variable és constant els sumands en $H(X)$ són $-0 \log 0 = 0$ o bé $-1 \log 1 = 0$. Recíprocament, si la variable no és constant pren algun valor amb probabilitat $p \neq 0, 1$ i en aquest cas algun dels sumands és $-p \log p > 0$, de manera que la suma total ha de ser > 0. □

Entropia d'una distribució de probabilitat. La natura dels valors que pren la variable X és irrellevant. L'únic important és la distribució de probabilitats corresponent. Per això es pot veure l'entropia com una quantitat associada a una distribució de probabilitats: una n-tupla $\boldsymbol{p} = (p_i)_{1\leqslant i\leqslant n} = (p_1, p_2, \ldots, p_n)$ de nombres reals $p_i \in [0,1] \subset \mathbb{R}$ amb suma $\sum_{i=1}^n p_i = 1$. Es denota

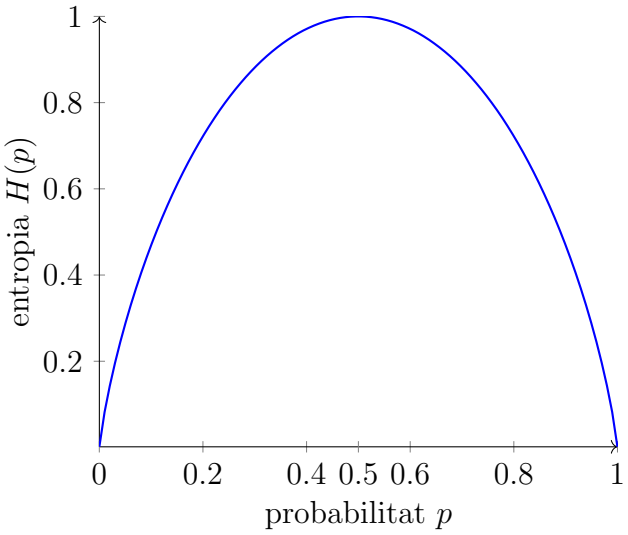
$$H(\boldsymbol{p}) := H(p_1, p_2, \ldots, p_n) = \sum_{i=1}^n p_i \log \frac{1}{p_i} = -\sum_{i=1}^n p_i \log p_i,$$

i es veu H com una funció definida en el símplex $(n-1)$-dimensional format pels punts de $\mathbb{R}^n$ de coordenades no negatives amb suma 1. De fet, en realitat està definida en la reunió de tots aquests símplex per a $n \geqslant 2$.

Per a $n = 2$ s'acostuma a fer un abús de notació i escriure

$$H(p) := H(p, 1-p) = -p \log p - (1-p)\log(1-p), \qquad p \in [0,1].$$

En aquest cas binari l'entropia H es veu com una funció d'una variable p que està definida sobre l'interval $[0,1]$. La seva gràfica està dibuixada a continuació:



Proposició 2.6 (Propietats de l'entropia)**.** *L'entropia, considerada com a funció de distribucions de probabilitat, té les propietats següents:*

1. *Continuïtat: $H(p_1, \ldots, p_n)$ és una funció contínua de les variables $p_1, \ldots, p_n$.*
2. *Simetria: $H(p_{\sigma(1)}, \ldots, p_{\sigma(n)}) = H(p_1, \ldots, p_n)$ per a tota permutació $\sigma \in \mathfrak{S}_n$.*
3. *Positivitat: $H(p_1, \ldots, p_n) \geqslant 0$. Es compleix la igualtat si i només si la variable és constant: $p_i = 1$ per a algun i i per tant $p_j = 0$ per a tot $j \neq i$.*
4. *Irrellevància dels successos impossibles: $H(p_1, \ldots, p_n, 0) = H(p_1, \ldots, p_n)$;*
5. *En distribucions equiprobables, és creixent respecte del nombre de casos:*

$$H\left(\frac{1}{n}, \ldots, \frac{1}{n}\right) < H\left(\frac{1}{n+1}, \ldots, \frac{1}{n+1}\right).$$

6. *Additivitat: si* $(p_1,\dots,p_n)$ *i* $(q_1,\dots,q_m)$ *són dues distribucions,*

$$H(p_1q_1,\dots,p_1q_m,\dots,p_nq_1,\dots,p_nq_m) = H(p_1,\dots,p_n) + H(q_1,\dots,q_m).$$

7. *Recurrència: si* $p = p_1 + \cdots + p_k$ *i* $q = p_{k+1} + \cdots + p_n$,

$$H(p_1,\dots,p_n) = pH\left(\frac{p_1}{p},\dots,\frac{p_k}{p}\right) + qH\left(\frac{p_{k+1}}{q},\dots,\frac{p_n}{q}\right) + H(p,q).$$

PROVA: La propietat (1) es dedueix de la continuïtat de la funció logaritme sobre $(0,1]$ i del fet que la funció $x\log x$ té limit zero en tendir x a zero per la dreta. En efecte, per l'Hôpital,

$$\lim_{x\to 0^+} x\log x = \lim_{x\to 0^+} \frac{\log x}{1/x} = \lim_{x\to 0^+} \frac{1/x}{-1/x^2} = \lim_{x\to 0^+} -x = 0.$$

Les propietats (2), (3) i (4) són immediates tenint en compte els convenis adoptats per als valors de l'expressió $p\log\frac{1}{p}$ quan $p = 0$.

Per comprovar (5) només cal observar que $H(\frac{1}{n},\dots,\frac{1}{n}) = \log n$ i tenir en compte que la funció logaritme és estrictament creixent.

Per veure (6) es calcula

$$\begin{aligned}\sum_{i,j} p_iq_j \log\frac{1}{p_iq_j} &= \sum_{i,j} p_iq_j\left(\log\frac{1}{p_i} + \log\frac{1}{q_j}\right) = \sum_{i,j} p_iq_j\left(\log\frac{1}{p_i}\right) + \sum_{i,j} p_iq_j\left(\log\frac{1}{q_j}\right)\\ &= \Big(\sum_j q_j\Big)\sum_i p_i\log\frac{1}{p_i} + \Big(\sum_i p_i\Big)\sum_j q_j\log\frac{1}{q_j} = \sum_i p_i\log\frac{1}{p_i} + \sum_j q_j\log\frac{1}{q_j}.\end{aligned}$$

Finalment, per veure (7),

$$pH\left(\frac{p_1}{p},\dots,\frac{p_k}{p}\right) = p\sum_{i=1}^k \frac{p_i}{p}\log\frac{p}{p_i} = \sum_{i=1}^k p_i\left(\log\frac{1}{p_i} + \log p\right) = p\log p + \sum_{i=1}^k p_i\log\frac{1}{p_i},$$

i, anàlogament,

$$qH\left(\frac{p_{k+1}}{q},\dots,\frac{p_n}{q}\right) = q\log q + \sum_{i=k+1}^n p_i\log\frac{1}{p_i}.$$

Sumant totes dues expressions amb $H(p,q) = -p\log p - q\log q$ s'obté $H(p_1,\dots,p_n)$. □

Divergència. En teoria de la probabilitat sovint convé comparar dues distribucions de probabilitat en un mateix conjunt i buscar maneres de mesurar com de diferents són l'una de l'altra. Hi ha diferents maneres de fer-ho. Una de les més importants és la *divergència de Kullback-Leibler*, també coneguda com a *entropia relativa*. En el cas discret es defineix de la manera següent:

Definició 2.7 (Divergència)**.** *Si* $\boldsymbol{p} = (p_i)_{1\leqslant i\leqslant n}$ *i* $\boldsymbol{q} = (q_i)_{1\leqslant i\leqslant n}$ *són dues distribucions de probabilitat la seva* divergència *o* entropia relativa *es defineix com:*

$$D(\boldsymbol{p}\|\boldsymbol{q}) = \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}.$$

Aquesta divergència no té les propietats de les distàncies en matemàtiques: no és ni simètrica, ja que en general $D(\boldsymbol{p}\|\boldsymbol{q}) \neq D(\boldsymbol{q}\|\boldsymbol{p})$, ni tampoc satisfà sempre la desigualtat triangular. En canvi sí que és *definida positiva*: pren valors no negatius i només pot ser zero quan totes dues distribucions de probabilitat són iguals. Per comprovar això, que no és immediat, convé demostrar primer un resultat tècnic molt útil per establir propietats d'expressions que apareixen sovint en teoria de la informació:

Lema 2.8 (Lema de Gibbs)**.** *Per a tot parell de distribucions de probabilitat* $\boldsymbol{p} = (p_i)_{1\leqslant i\leqslant n}$ *i* $\boldsymbol{q} = (q_i)_{1\leqslant i\leqslant n}$ *es compleix la Desigualtat de Gibbs:*

$$\sum p_i \log \frac{p_i}{q_i} \geqslant 0, \tag{2}$$

i se satisfà la igualtat si, i només si, $\boldsymbol{p} = \boldsymbol{q}$*; o sigui, quan* $p_i = q_i$ *per a tot índex* i*.*

L'enunciat també és cert si als nombres q_i *només se'ls demana la condició més general que siguin tots* $\geqslant 0$ *i que tinguin suma* $\sum_{i=1}^{n} q_i \leqslant 1$*, no necessàriament igual a* 1*.*

Prova: Naturalment, en aquest enunciat s'adopten els convenis habituals per a expressions de la forma $a \log \frac{a}{b}$ per a nombres reals a, b quan algun dels dos és zero.

En primer lloc s'observa que si l'enunciat és cert agafant el logaritme en una base qualsevol aleshores també és cert en qualsevol altra base. Això és degut a què canviar de base la funció logaritme equival a multiplicar per un nombre estrictament positiu, i en fer això les afirmacions de l'enunciat es mantenen.

Es demostrarà la desigualtat per a la funció logaritme neperià ln: el logaritme en base e, que és la que té derivada $\ln'(x) = \frac{1}{x}$. Aquesta funció satisfà $\ln x \leqslant x - 1$ per a tot $x > 0$, amb igualtat si, i només si, $x = 1$. En efecte, la funció $\ln x - x + 1$ té derivada $\frac{1}{x} - 1$; per tant la funció és estrictament creixent quan $x < 1$ (derivada estrictament positiva) i és estrictament decreixent quan $x > 1$ (derivada estrictament negativa). Com que val zero en $x = 1$ és una funció estrictament negativa en tots els $x \neq 1$.

Per a tot parell de nombres $a, b \geqslant 0$ es té la desigualtat

$$a \ln \frac{b}{a} \leqslant b - a,$$

amb igualtat si i només si $a = b$. En efecte, quan algun dels dos és zero es té

$$0 = 0 \ln \frac{0}{0} = 0 - 0; \quad 0 = 0 \ln \frac{b}{0} < b - 0; \quad -\infty = a \ln \frac{0}{a} < 0 - a,$$

i quan tots dos són diferents de zero, agafant $x = \frac{b}{a} > 0$ es té

$$\ln \frac{b}{a} \leqslant \frac{b}{a} - 1 \Leftrightarrow a \ln \frac{b}{a} \leqslant a\left(\frac{b}{a} - 1\right) = a\frac{b}{a} - a = b - a,$$

amb igualtat si, i només si, $x = 1 \Leftrightarrow a = b$.

Aplicant-ho a cadascun dels parells p_i i q_i es té

$$-\sum p_i \ln \frac{p_i}{q_i} = \sum p_i \ln \frac{q_i}{p_i} \leqslant \sum (q_i - p_i) = \sum q_i - \sum p_i = 1 - 1 = 0, \tag{3}$$

amb igualtat si, i només si, $p_i = q_i$ per a tot índex i. Canviant el signe s'obté la desigualtat de Gibbs.

Una demostració alternativa es pot obtenir directament a partir de la *desigualtat de Jensen*: per a tota variable aleatòria X i funció convexa f es compleix

$$\mathbb{E}\big[f(X)\big] \geqslant f\big(\mathbb{E}[X]\big).$$

Si la funció f és estrictament convexa aleshores la desigualtat és una igualtat si, i només si, la variable X és constant: pren un únic valor amb probabilitat 1.

S'agafa la variable aleatòria X que pren cadascun els n valors $\frac{q_i}{p_i}$ amb probabilitat p_i i la funció estrictament convexa $f(x) = -\log x$. Aplicant la desigualtat de Jensen es té

$$\sum p_i \left(-\log \frac{q_i}{p_i}\right) = \mathbb{E}[f(X)] \geqslant f\big(\mathbb{E}[X]\big) = -\log\left(\sum p_i \frac{q_i}{p_i}\right) = -\log\left(\sum q_i\right) = -\log 1 = 0,$$

amb desigualtat estricta excepte si X és constant, que equival a què prengui l'únic valor $\frac{p_i}{q_i} = 1$, equivalent a què $p_i = q_i$ per a tot i.

En el cas més general en què els q_i tenen suma $q = \sum_{i=1}^{n} q_i \leqslant 1$, es pot veure que la demostració també val observant que al final de 3 s'obté $\leqslant q - 1 \leqslant 0$. Alternativament, també es pot veure la validesa en aquest cas més general considerant les distribucions de probabilitat amb $n+1$ valors obtingudes afegint $p_{n+1} = 0$ i $q_{n+1} = 1 - q$ i aplicant-les-hi la desigualtat per a distribucions de probabilitat ja demostrada. □

Moltes vegades la *Desigualtat de Gibbs* s'enuncia de la manera equivalent següent: donades dues distribucions de probabilitat (p_i) i (q_i) es té la desigualtat

$$\sum p_i \log \frac{1}{p_i} \leqslant \sum p_i \log \frac{1}{q_i}, \quad \text{amb igualtat si, i només si,} \quad p_i = q_i \ \text{ per a tot } \ i. \tag{4}$$

Proposició 2.9. *Propietats de l'entropia relativa:*

1. $0 \leqslant D(\boldsymbol{p}\|\boldsymbol{q}) \leqslant \infty$*; a més,*
   - (a) $D(\boldsymbol{p}\|\boldsymbol{q}) = 0$ *si, i només si,* $\boldsymbol{p} = \boldsymbol{q}$*;*
   - (b) $D(\boldsymbol{p}\|\boldsymbol{q}) = \infty$ *si, i només si, existeix un índex* i *tal que* $q_i = 0$ *però* $p_i \neq 0$*;*
2. *En general,* $D(\boldsymbol{p}\|\boldsymbol{q}) \neq D(\boldsymbol{q}\|\boldsymbol{p})$.

Prova: Els apartats (1) i (1)-(a) són conseqüència immediata del lema de Gibbs. L'apartat (1)-(b) és clar del fet que els sumands del sumatori que defineix l'entropia relativa són nombres reals o bé valen ∞. La suma val ∞ si, i només si, algun sumand té aquest valor, el qual és equivalent a la condició de l'enunciat.

Per veure el segon apartat, la manca de simetria de l'entropia relativa en general, n'hi ha prou a comprovar-la en un exemple. Per exemple, si $\boldsymbol{q}$ és la distribució de Bernoulli uniforme i $\boldsymbol{p}$ és una altra distribució de Bernoulli, les dues divergències són sempre diferents. □

Una conseqüència immediata i molt important de la desigualtat de Gibbs és el següent:

Teorema 2.10. *L'entropia és màxima per a la distribució uniforme: per a tota distribució de probabilitat $\boldsymbol{p} = (p_1, \dots, p_n)$ es compleix*

$$H(\boldsymbol{p}) \leqslant \log n, \qquad \textit{amb igualtat si i només si} \quad p_i = \frac{1}{n} \quad \textit{per a tot} \quad i.$$

Prova: Aplicant la desigualtat de Gibbs amb $\boldsymbol{q} = (q_1, \dots, q_n)$ i $q_i = \frac{1}{n}$ es té

$$0 \leqslant \sum p_i \log \frac{p_i}{q_i} = \sum p_i \log np_i = \sum p_i \log n + \sum p_i \log p_i = \log n - H(\boldsymbol{p})$$

d'on es dedueix la desigualtat de l'enunciat. La igualtat es compleix només quan ho preveu la desigualtat de Gibbs; és a dir, quan totes dues distribucions de probabilitat coincideixen: $\boldsymbol{p}$ és uniforme. □

Problemes

2.1. Calculeu quanta informació s'obté en conèixer el resultat dels experiments següents:

1. tirar tres daus;
2. tirar dos daus i sumar els seus valors;
3. tirar quatre monedes i comptar el nombre de cares;
4. tirar un dau i després tirar tantes monedes com digui el resultat del dau.

2.2. Es consideren quatre estats del temps: pluja, sol, núvols i boira. En una ciutat les probabilitats del quatre estats són les mateixes, i en una altra són $\frac{1}{4}, \frac{1}{8}, \frac{1}{8}$ i $\frac{1}{2}$, respectivament. En quina de les dues ciutats l'estat del temps dona més informació?

2.3. Siguin X i Y dues variables aleatòries, que prenen valors en conjunts disjunts $\mathcal{X}$ i $\mathcal{Y}$. Sigui Z la variable aleatòria que pren valors en la reunió $\mathcal{Z} := \mathcal{X} \sqcup \mathcal{Y}$ de la manera següent: es tria una de les dues variables X o Y amb probabilitats p i $1-p$, respectivament, i s'agafa el valor d'aquesta variable triada. Demostreu que

$$H(Z) = H(p, 1-p) + pH(X) + (1-p)H(Y).$$

Generalitzeu-ho a considerar n variables aleatòries $X_1, \dots, X_n$ i la variable Z que consisteix en: primer es tria una de les variables X_i, segons una probabilitat p_i i aleshores s'agafa el valor d'aquesta variable.

2.4. Sigui X la variable aleatòria que compta el nombre de tirades d'una moneda que calen fins a obtenir la primera cara. És una variable discreta que pren valors en el conjunt numerable $\mathcal{X} = \mathbb{N} = \{1, 2, 3, \dots\}$. Calculeu la seva entropia $H(X)$.

Quines preguntes amb resposta binària si/no es poden fer a algú que coneix el resultat de la variable per descobrir aquest resultat amb el mínim nombre de preguntes?

Indicació: Recordeu la suma de la sèrie geomètrica i la derivació de sèries.

2.5. Sigui X una variable aleatòria que pren $n+1$ valors $\mathcal{X} = \{0, 1, 2, \ldots, n\}$ amb distribució de probabilitat "geomètrica finita"

$$X \sim (q, pq, p^2q, \ldots, p^{n-1}q, p^n),$$

on $\mathrm{Ber}(p) \sim (q, p)$ és una distribució de Bernoulli. Aquesta variable compta el nombre d'èxits inicials abans del primer fracàs o d'arribar al final en una seqüència de n mostres d'una variable de Bernoulli amb la distribució donada.

Calculeu la seva entropia i calculeu també l'entropia de la distribució geomètrica sobre els naturals

$$\Pr(X = k) = p^k q, \qquad k \in \mathbb{N}.$$

INDICACIÓ: Podeu fer servir la fórmula següent (que es comprova per inducció):

$$\sum_{k=0}^{n-1} kx^k = \frac{x(1-x^n) - nx^n(1-x)}{(1-x)^2}, \quad n \geqslant 1, \quad x \in \mathbb{R} \setminus \{1\}.$$

2.6. Donats nombres no negatius $a_1, \ldots, a_n$ i $b_1, \ldots, b_n$ amb sumes $A = \sum a_i$ i $B = \sum b_i$, demostreu (amb els convenis habituals sobre $a \log \frac{a}{b}$) la *log sum inequality*:

$$\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geqslant A \log \frac{A}{B},$$

amb igualtat si, i només si, $a_i B = b_i A$ per a tot i.

En particular, quan $A = 1$ i $B \leqslant 1$ la desigualtat és el cas general de la desigualtat de Gibbs: $\sum_{i=1}^{n} a_i \log \frac{a_i}{b_i} \geqslant 0$.

2.7. Demostreu que si es canvien dues probabilitats en una distribució de manera que el resultat s'apropi a la uniforme aleshores l'entropia augmenta: si $(p_1, p_2, p_3, \ldots, p_n)$ és una distribució de probabilitat amb $p_1 < p_2$ i s'agafa $0 < \Delta \leqslant \frac{p_2 - p_1}{2}$ aleshores

$$H(p_1 + \Delta, p_2 - \Delta, p_3, \ldots, p_n) > H(p_1, p_2, p_3, \ldots, p_n).$$

2.2 Diverses variables

Referències: Cover-Thomas [4, Chapter 2], Brunat-Ventura [3, Capítol 2], Ball[2, Chapter 1].

A continuació es considera l'entropia de parells de variables aleatòries i, en general, de variables amb valors vectorials.

Definició 2.11 (Entropia conjunta)**.** *Sigui X, Y un parell de variables aleatòries que prenen valors en els conjunts $\mathcal{X}$ i $\mathcal{Y}$, respectivament. La seva entropia conjunta és*

$$H(X, Y) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \frac{1}{p(x, y)} = \mathbb{E}\big[-\log p(X, Y) \big].$$

És a dir, l'entropia conjunta no és res més que l'entropia de la variable bivaluada corresponent al parell de variables $\boldsymbol{X} = (X, Y)$, de manera que $H(X, Y) = H(\boldsymbol{X})$.

Es defineix l'entropia conjunta anàlogament per a n-tuples de variables aleatòries: si $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ és una variable vectorial amb components les variables X_i que prenen valors en els conjunts $\mathcal{X}_i$ la seva entropia és

$$H(\boldsymbol{X}) = H(X_1, \ldots, X_n) = \sum_{x_i \in \mathcal{X}_i} p(x_1, \ldots, x_n) \log \frac{1}{p(x_1, \ldots, x_n)}.$$

Proposició 2.12. *$H(X, Y) \leqslant H(X) + H(Y)$ amb igualtat si, i només si, X i Y són independents.*

Prova: Aplicant la desigualtat de Gibbs es té

$$H(X, Y) = \sum_{i,j} p(x_i, y_j) \log \frac{1}{p(x_i, y_j)} \leqslant \sum_{i,j} p(x_i, y_j) \log \frac{1}{p(x_i)p(y_j)}.$$

Separant al sumatori de la dreta el logaritme del producte com a suma de logaritmes, aplicant la propietat distributiva i la relació entre probabilitats conjuntes i marginals, es dedueix que el sumatori de la dreta és $H(X) + H(Y)$. La condició d'igualtat es dedueix de la condició corresponent de la desigualtat de Gibbs. La identitat $p(x_i, y_j) = p(x_i)p(y_j)$ per a tot i, j correspon a la independència de les variables aleatòries. □

Definició 2.13 (Entropia condicionada)**.** *Siguin X, Y un parell de variables aleatòries que prenen valors en conjunts $\mathcal{X}$ i $\mathcal{Y}$, respectivament. Per a cada $x \in \mathcal{X}$ es pot considerar la variable aleatòria Y condicionada a aquest valor $Y|X = x$, que té entropia*

$$H(Y|X = x) = \sum_{y \in \mathcal{Y}} p(y|x) \log \frac{1}{p(y|x)}.$$

L'entropia de la variable Y condicionada a la variable X es defineix com la mitjana d'aquestes entropies per a tots els elements $x \in \mathcal{X}$:

$$H(Y|X) = \mathbb{E}_X\big[H(Y|X)\big] = \sum_{x \in \mathcal{X}} p(x) H(Y|X = x).$$

Lema 2.14. *L'entropia condicionada ve donada per l'expressió següent:*

$$H(Y|X) = \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \frac{1}{p(y|x)} = \mathbb{E}_{(X,Y)}\big[- \log p(Y|X)\big].$$

Prova: Desenvolupant la definició es té:

$$\begin{aligned} H(Y|X) = \mathbb{E}_X\big[H(Y|X)\big] = \sum_{x \in \mathcal{X}} p(x) H(Y|X = x) &= \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} p(x) p(y|x) \log \frac{1}{p(y|x)} \\ &= \sum_{x \in \mathcal{X}, y \in \mathcal{Y}} p(x, y) \log \frac{1}{p(y|x)} = \mathbb{E}_{(X,Y)}\big[- \log p(Y|X)\big], \end{aligned}$$

i això demostra la fórmula de l'enunciat. □

L'entropia condicionada $H(Y|X)$ s'interpreta com la quantitat d'informació addicional que proporciona saber el valor de la variable Y quan ja es coneix el valor de la variable X. De vegades se li diu també *equivocació*, que correspon a la interpretació següent: es vol obtenir tanta informació com es pugui sobre Y a partir del valor de X; l'equivocació $H(Y|X)$ és la quantitat d'informació sobre Y que no s'obté quan s'estima el seu valor coneixent només X.

Proposició 2.15 (Regla de la cadena)**.**

$$H(X,Y) = H(X) + H(Y|X) = H(Y) + H(X|Y).$$

PROVA: En l'expressió per a $H(X,Y)$ a dins del logaritme es canvia $p(x,y)$ per $p(x)p(y|x)$, se separa en suma de dos i s'obté la primera igualtat. La segona s'obté anàlogament. □

Aplicant reiteradament aquesta igualtat s'obté la regla de la cadena per a entropies conjuntes de més de dues variables:

$$\begin{aligned} H(X_1, X_2, \ldots, X_n) &= H(X_1) + H(X_2|X_1) + H(X_3|X_1, X_2) + \cdots + H(X_n|X_1, X_2, \ldots, X_{n-1}) \\ &= \sum_{k=1}^{n} H(X_k|X_1, \ldots, X_{k-1}) \end{aligned}$$

Proposició 2.16. *$H(X|Y) \leqslant H(X)$ amb igualtat si, i nomes si, X i Y són independents.*

PROVA: Conseqüència de la regla de la cadena i de la relació entre l'entropia conjunta i la suma de les entropies marginals de la proposició 2.12:

$$H(Y) + H(X|Y) = H(X,Y) \leqslant H(X) + H(Y)$$

i restant a cada costat $H(Y)$ s'obté la desigualtat, que és una igualtat si, i només si, la desigualtat anterior ho és, que per la proposició 2.12 passa exactament quan les dues variables siguin independents. □

Exemple 2.17. *Es considera l'experiment de tirar dos daus i les variables X i Y que donen el resultat d'un dau i la suma de tots dos. Aleshores es té:*

- $X \sim \text{Unif}(6)$*;* $H(X) = \log 6 \approx 2.58496$*;*
- $Y \sim \frac{1}{36}(1,2,3,4,5,6,5,4,3,2,1)$*;* $H(Y) \approx 3.2744$*;*
- (X,Y) *té només probabilitats* $\frac{1}{36}$ *i* 0*;* $H(X,Y) = \log 36 = 2\log 6 \approx 5.16993 < 5.85936 = H(X) + H(Y)$*;*
- $Y|X = x$ *té només probabilitats* $\frac{1}{6}$ *i* 0*;* $H(Y|X = x) = \log 6 \approx 2.58496$*;*
- $X|Y = y$ *té només probabilitats* $\frac{1}{y-1}$ *i* 0 *per a* $y = 2,3,4,5,6,7$*, i les altres són simètriques;* $H(X|Y = y) = \log(y-1)$ *per a* $2 \leqslant y \leqslant 7$*, i les altres simètriques.*
- $H(Y|X) = \log 6 \approx 2.58496$*;*
- $H(X|Y) \approx 1.89552$.

Variables que són funció d'altres. Donada una variable X que pren valors en un conjunt $\mathcal{X}$ i una aplicació $g\colon \mathcal{X} \to \mathcal{Y}$ en un conjunt $\mathcal{Y}$ es defineix la variable $Y := g(X)$ com la que pren valors en $\mathcal{Y}$ amb distribució conjunta

$$\Pr(X = x, Y = y) = \begin{cases} \Pr(X = x), & \text{si} \quad y = g(x), \\ 0, & \text{si} \quad y \neq g(x). \end{cases}$$

En particular, la distribució de probabilitat de la variable Y és

$$\Pr(Y = y) = \sum_{\substack{x \in \mathcal{X} \\ g(x) = y}} \Pr(X = x).$$

En aquest cas es diu que Y és funció (determinista) de X.

Lema 2.18. *La variable Y és funció de X si, i només si, totes les variables $Y|X = x$ són constants per a tot $x \in \mathcal{X}$ amb $\Pr(X = x) \neq 0$.*

Prova: En efecte, suposi's que $Y = g(X)$. Sigui $x \in \mathcal{X}$ amb $p(x) \neq 0$. Aleshores del fet que $p(x) = \sum_{y \in \mathcal{Y}} p(x, y)$ i que per definició $p(x, y)$ només pugui ser $p(x)$ o zero es dedueix que existeix un únic element $y \in \mathcal{Y}$ amb $y = g(x)$. Aleshores la variable $Y|X = x$ és constant amb valor $y = g(x) \in \mathcal{Y}$:

$$\Pr(Y = y|X = x) = \frac{\Pr(X = x, Y = y)}{\Pr(X = x)} = \begin{cases} 1, & \text{si} \quad y = g(x), \\ 0, & \text{si} \quad y \neq g(x). \end{cases}$$

Recíprocament, suposi's que totes les variables condicionades són constants. Es defineix una funció $g\colon \mathcal{X} \to \mathcal{Y}$ de la manera següent: per als x amb probabilitat $\Pr(X = x) = 0$ es pot definir de manera arbitraria, triant un valor qualsevol; per als x amb $\Pr(X = x) \neq 0$ es defineix $g(x)$ com el valor constant d'aquesta variable: l'únic element $y \in \mathcal{Y}$ tal que $\Pr(Y = y|X = x) = 1$. Es vol veure que la variable Y és igual a $g(X)$. En efecte, per a tot parell $(x, y) \in \mathcal{X} \times \mathcal{Y}$ es té:

$$\Pr(X = x, Y = y) = \Pr(X = x) \Pr(Y = y|X = x) = \begin{cases} \Pr(X = x), & \text{si} \quad y = g(x), \\ 0, & \text{si} \quad y \neq g(x). \end{cases}$$

Això val en tots els casos: si $p(x) \neq 0$ aleshores $Y|X = x$ és constant i el producte $p(x)p(y|x)$ només pot ser u o zero, segons si $y = g(x)$ o no; altrament, si $p(x) = 0$ aleshores les probabilitats $p(y|x)$ no estan definides però $p(x, y)$ és sempre zero, que coincideix amb els valors $p(x)$ i zero.

Proposició 2.19. *Donades variables aleatòries X i Y, són equivalents:*

1. *Y és funció determinista de X: $Y = g(X)$;*
2. *$H(Y|X) = 0$;*
3. *$H(X, Y) = H(X)$.*

En aquest cas es té una desigualtat $H(Y) \leqslant H(X)$ amb igualtat si, i només si, g és injectiva en el suport de X.

PROVA: L'equivalència entre (1) i (2) és conseqüència de la definició d'entropia condicionada: $H(Y|X)$ és l'esperança de les entropies $H(Y|X = x)$, que és zero si i només si aquestes entropies són totes zero per a tots els valors de x amb $\Pr(X = x) \neq 0$. Una variable té entropia zero si i només si és constant.

L'equivalència entre (2) i (3) és la regla de la cadena.

Finalment, si es compleixen (2) i (3) també es compleix (1) ja que de la condició

$$H(Y|X) = \sum_{x \in \mathcal{X}} p(x) H(Y|X = x) = 0,$$

i tenint en compte que l'entropia d'una variable és sempre no negativa, es dedueix que per a tot $x \in \mathcal{X}$ amb probabilitat no nul·la ha de ser $H(Y|X = x) = 0$, que pel lema 2.18 equival a què Y sigui funció de X.

Si es compleixen les condicions aleshores $H(Y) \leqslant H(X, Y) = H(X)$. Per la condició (3) la igualtat es compleix si, i només si, X també és funció de Y. Això vol dir que per a cada $y \in \mathcal{Y}$ amb probabilitat no nul·la, que equival a dir que sigui de la imatge de g en el suport de $\mathcal{X}$, la variable $X|Y = y$ és constant: existeix un únic element $x \in \mathcal{X}$ amb imatge y. □

De forma similar a com es va fer per a la definició 2.1, podem definir la informació entre dues variables aleatòries (no s'ha de confondre amb l'entropia conjunta):

Definició 2.20 (Informació per a dues variables aleatòries)**.** *Siguin X, Y dues variables aleatòries. Si observem $Y = y$, la quantitat d'informació de la que disposem sobre l'esdeveniment $X = x$ és*

$$I(Y = y; X = x) = \log \frac{p(x, y)}{p(x)p(y)}$$

Proposició 2.21. *Donades dues variables aleatòries X i Y, es compleix $I(Y = y; X = x) = 0$ quan X i Y són independents.*

PROVA: Si X i Y són independents, $p(x, y) = p(x)p(y)$. □

Si promitgem sobre els possibles valors de X i de Y podem definir el concepte d'*informació mútua* de l'una respecte l'altra. Representa la quantitat d'informació promig que s'obté sobre una d'elles en observar l'altra.

Definició 2.22 (Informació mútua)**.** *Siguin X, Y un parell de variables aleatòries que prenen valors en els conjunts $\mathcal{X}$ i $\mathcal{Y}$, respectivament. La informació mútua de l'una respecte l'altra es defineix com:*

$$I(X; Y) = \sum_{(x,y) \in \mathcal{X} \times \mathcal{Y}} p(x, y) \log \frac{p(x, y)}{p(x)p(y)} = \mathbb{E}_{(X,Y)} \left[\log \frac{p(X, Y)}{p(X)p(Y)} \right].$$

Proposició 2.23 (Propietats de la informació mútua)**.** *La informació mútua satisfà les propietats següents:*

1. $I(X;Y) = H(X) + H(Y) - H(X,Y)$;
2. $I(X;Y) = H(X) - H(X|Y) = H(Y) - H(Y|X)$;
3. $I(X;Y) = I(Y;X)$;
4. $I(X;Y) \geqslant 0$; $I(X;Y) = 0$ *si, i només si,* X *i* Y *són independents;*
5. $I(X;X) = H(X)$;
6. $I(X;Y) = D\big(p(x,y) \| p(x)p(y)\big)$.

PROVA:

1. Separant el logaritme en suma de tres termes i passant en dos dels sumatoris de la probabilitat conjunta a les probabilitats marginals queden tres sumatoris que corresponen als termes de la dreta.
2. A partir de la identitat anterior usant la regla de la cadena.
3. L'expressió de la definició és simètrica i la de l'apartat (1) també.
4. És immediat a partir de la desigualtat de Gibbs. També es dedueix tant del primer com del segon apartat a partir de les propietats de l'entropia conjunta i l'entropia condicionada.
5. Es veu amb qualsevol dels dos primers apartats.
6. És simplement llegir la definició de la informació mútua com la divergència entre dues distribucions de probabilitat en el conjunt $\mathcal{X} \times \mathcal{Y}$: la probabilitat conjunta de les variables X i Y donades i la probabilitat producte de les probabilitats marginals corresponents, que és la distribució que correspondria a un parell de variables aleatòries independents amb les mateixes distribucions marginals que les donades. □

Definició 2.24 (Informació mútua condicionada)**.** *Siguin* X, Y *un parell de variables aleatòries que prenen valors en els conjunts* $\mathcal{X}$ *i* $\mathcal{Y}$*. Donada una variable* Z *que pren valors en* $\mathcal{Z}$*, la informació conjunta entre* X *i* Y *es defineix com:*

$$I(X;Y|Z) = \sum_{(x,y,z)\in\mathcal{X}\times\mathcal{Y}\times\mathcal{Z}} p(x,y,z) \log \frac{p(x,y|z)}{p(x|z)p(y|z)} = \mathbb{E}_{(X,Y,Z)}\left[\log \frac{p(X,Y|Z)}{p(X|Z)p(Y|Z)}\right].$$

Proposició 2.25. *La informació mútua condicionada satisfà les següents propietats:*

1. $I(Y;X|Z) = H(Y|Z) - H(Y|X,Z)$
2. $I(X;Y|Z) = I(Y;X|Z)$
3. $I(X;X|Z) = H(X|Z)$
4. $I(X;Y|Z) \geqslant 0$*, i* $I(X;Y|Z) = 0$ *si* X *condicionada a* Z *i* Y *condicionada a* Z *són variables aleatòries independents.*
5. *Regla de la cadena:* $I(X_1, X_2; Y) = I(X_1;Y) + I(X_2;Y|X_1)$

PROVA:

1. Cal separar el logaritme en dos termes, tenir en compte que $p(X|Y,Z) = p(X,Y|Z)/p(Y|Z)$ i identificar les entropies a partir de les probabilitats condicionades marginals.
2. Comproveu que l'expressió de la definició es simètrica en X i Y.
3. Es veu a partir del primer apartat.
4. Es pot deduir del primer apartat.
5. A partir del segon apartat de la proposició 2.23 i de la regla de la cadena per l'entropia conjunta, podem escriure

$$\begin{aligned} I(X_1, X_2; Y) &= H(X_1, X_2) - H(X_1, X_2|Y) \\ &= H(X_1) + H(X_2|X_1) - H(X_1|Y) - H(H_2|X_1, Y) \\ &= I(X_1; Y) + I(X_2; Y|X_1). \end{aligned}$$

□

Problemes

2.8. Les *sèries mundials* són una competició de beisbol entre dos equips A i B en què el vencedor és el primer que guanya quatre partits. Per exemple, si A guanya els quatre primers partits s'acaba el torneig i queda guanyador; en canvi, si guanyen alternadament s'han de jugar set partits fins a poder donar per acabada la competició, amb vencedor aquell que n'ha guanyat quatre. Se suposa que tots dos equips tenen les mateixes probabilitats de guanyar en cada partit.

Sigui X la variable aleatòria que dona tots els resultats possibles de la sèrie (la seqüència d'equips guanyadors: $AAAA$, $BBBB$, $ABAAA$, $BABABAB$, ...) i sigui Y la variable que dona el nombre de partits que s'han jugat (entre quatre i set).

Calculeu les entropies $H(X)$, $H(Y)$, $H(X,Y)$, $H(Y|X=x)$, $H(Y|X)$, $H(X|Y=y)$ i $H(X|Y)$ i la informació mútua $I(X;Y)$.

2.9. Un servei de meteorologia prediu el temps. Siguin X la variable aleatòria que indica la predicció i Y la variable aleatòria que diu el temps que realment ha fet. Totes dues prenen valors en el conjunt $\{\mathtt{P}, \mathtt{N}\}$, on $\mathtt{P}$ = "plou" i $\mathtt{N}$ = "no plou". Observant l'històric de prediccions i temps real s'obté la distribució següent per al parell de variables (X, Y):

$$\begin{aligned} \Pr(X=\mathtt{P}, Y=\mathtt{P}) &= \frac{1}{12}, \quad \Pr(X=\mathtt{P}, Y=\mathtt{N}) = \frac{1}{6}, \\ \Pr(X=\mathtt{N}, Y=\mathtt{P}) &= \frac{1}{12}, \quad \Pr(X=\mathtt{N}, Y=\mathtt{N}) = \frac{2}{3}. \end{aligned}$$

1. Amb quina probabilitat encerta aquest servei el temps que farà?
2. Un espavilat es dona compte que si prediu sempre que no plourà l'encerta més vegades que el servei, i s'ofereix a substituir-lo a meitat de preu; amb quina probabilitat encerta?

3. Quina de les dues prediccions dona més informació sobre el temps que farà? Convé acceptar l'oferta de l'espavilat?

2.10. Considereu els textos següents sobre l'alfabet binari $\{\texttt{0},\texttt{1}\}$:

1. $\texttt{txt} = \texttt{0101} \overset{n)}{\cdots} \texttt{0101}$ de longitud $2n$;
2. $\texttt{txt} = \texttt{00110011} \overset{n)}{\cdots} \texttt{0011}$ de longitud $4n$;
3. $\texttt{txt} = \texttt{000111000111} \overset{n)}{\cdots} \texttt{000111}$ de longitud $6n$;
4. el mateix amb blocs alternats de zeros i uns de mida k en lloc de 2 o 3;
5. $\texttt{txt} = \texttt{0111} \overset{n-1)}{\cdots} \texttt{1}$ de longitud n;
6. $\texttt{txt} = \texttt{010011000111} \cdots \texttt{0} \overset{n)}{\cdots} \texttt{01} \overset{n)}{\cdots} \texttt{1}$ (quina longitud té?);
7. $\texttt{txt} = \texttt{010010001} \cdots \texttt{0} \overset{n)}{\cdots} \texttt{01}$ (quina longitud té?);
8. $\texttt{txt} = \texttt{11.}0010010000111111011010101000100\cdots$ (representació binària de π). INDICACIÓ: es conjectura que π és un *"nombre normal"*.

Sigui $\boldsymbol{X}$ la variable aleatòria amb valors en $\{\texttt{00},\texttt{01},\texttt{10},\texttt{11}\}$ corresponent a agafar dos dígits consecutius del text i siguin Y i Z les variables amb valors en $\{\texttt{0},\texttt{1}\}$ que donen el primer i el segon dígit de $\boldsymbol{X}$; és a dir, $\boldsymbol{X} = (Y, Z)$. Per tal que es puguin agafar dos dígits consecutius començant en qualsevol punt del text se suposa que el dígit que va a continuació de l'últim de `txt` és el primer.

Per a cadascun dels textos calculeu la distribució de probabilitats de les tres variables $\boldsymbol{X}, Y, Z$ i les distribucions de probabilitat condicionades de $Z|Y = \texttt{0}$ i $Z|Y = \texttt{1}$. Calculeu les entropies $H(\boldsymbol{X}), H(Y), H(Z)$, les entropies condicionades $H(Z|Y)$ i la informació mútua $I(Y;Z)$.

2.11. Una ciutat està dividida en dos barris A i B. En passar una enquesta sobre intenció de vot hi ha ciutadans que responen la veritat, altres que responen una mentida i altres que no responen. Sigui X la variable aleatòria que diu en quin barri viu un ciutadà i sigui Y la variable aleatòria que diu com respon les enquestes: veritat, fals o sense resposta.

Suposi's que $\Pr(X = A) = p \in [0,1]$, que un 20% dels habitants de la ciutat no responen les enquestes i que:

- dels habitants del barri A el 50% responen la veritat i el 30% responen una mentida;
- al barri B els percentatges s'intercanvien: 50% menteix i 30% respon la veritat.

1. Calculeu l'entropia condicionada $H(Y|X)$.
2. Calculeu l'entropia $H(Y)$ en funció de p.
3. Trobeu els valors de p que fan que la informació mútua $I(X;Y)$ sigui màxima i mínima, i digueu quant val aquesta informació mútua en cada cas.

2.12. Siguin X una variable aleatòria que pren valors en $\mathcal{X} \subset \mathbb{R}$ i sigui $Y = g(X)$. Digueu quina relació hi ha entre les entropies $H(X)$ i $H(Y)$ per a les tres funcions $Y = 2^X$, $Y = |X|$ i $Y = \lfloor X \rfloor$ i doneu exemples de totes les possibilitats segons el conjunt de valors $\mathcal{X}$ de la variable X.

2.13. *Fita per a l'entropia d'una binomial.* Demostreu que l'entropia d'una variable aleatòria amb distribució binomial X, que pren valors en el conjunt $\{0, 1, \ldots, n\}$ amb probabilitats $\Pr(X = k) = \binom{n}{k} p^k (1-p)^{n-k}$ està fitada de la manera següent:

$$H(X) \leqslant nH(p).$$

2.14. Sigui $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ un vector aleatori de variables binàries (no necessàriament independents ni amb la mateixa distribució). Sigui $\boldsymbol{R}$ la variable aleatòria que dona la seqüència del nombre de zeros i uns consecutius en el resultat de $\boldsymbol{X}$; per exemple, si $\boldsymbol{X} = (1, 1, 1, 0, 1, 1, 0, 0, 0, 0)$ aleshores $\boldsymbol{R} = (3, 1, 2, 4)$. Demostreu que

$$H(\boldsymbol{R}) \leqslant H(\boldsymbol{X}) \leqslant H(\boldsymbol{R}) + \min\big(H(X_i) : i = 1, \ldots, n\big).$$

Digueu en quins casos la desigualtat de l'esquerra és una desigualtat estricta.

2.3 Entropia diferencial

Referència: Cover-Thomas[4, Chapter 8].

Fins ara s'han considerat només variables discretes. En aquesta secció es donen les definicions i propietats més importants de la teoria de la informació en el cas de les variables contínues. Per a aquestes variables se sol usar el terme *entropia diferencial*, tot i que moltes vegades es diu simplement entropia.

L'entropia es defineix exactament igual per a tota classe de variables aleatòries X com l'esperança de la funció $-\log f_X$, on f_X és la funció de *densitat de probabilitat* de la variable. En el cas continu aquesta esperança ve donada per una integral en comptes d'una suma. A continuació es defineixen conceptes i es donen fórmules on apareixen integrals; aquestes integrals poden no existir de manera que els conceptes només estan definits i les fórmules només valen quan totes les integrals involucrades existeixin.

Definició 2.26 (Entropia diferencial)**.** *Sigui X una variable aleatòria contínua amb densitat de probabilitat f_X. La seva entropia (diferencial) és*

$$h(X) := \mathbb{E}\big[-\log f_X\big] = -\int_{-\infty}^{\infty} f_X(x) \log f_X(x)\, dx.$$

Tradicionalment es fa servir la lletra h minúscula per denotar l'entropia diferencial, en comptes de la H majúscula de l'entropia de variables discretes.

En la definició, de manera anàloga al cas discret, es considera que l'expressió $f_X(x) \log f_X(x)$ val zero per als x amb $f_X(x) = 0$. Es pot obviar aquest conveni si es defineix l'entropia com la integral sobre el *suport* de la funció f_X: el subconjunt del domini de definició de la variable amb densitat de probabilitat no nul·la.

A diferència del cas discret l'entropia diferencial pot agafar també valors negatius.

Exemples 2.27. *Les entropies diferencials d'algunes variables bàsiques són:*

1. *una variable uniforme sobre l'interval* $[a, b]$ *té entropia* $\log(b - a)$*;*
2. *una variable normal de variància* σ^2 *té entropia* $\log(\sigma\sqrt{2\pi e})$*;*
3. *una variable exponencial de paràmetre* λ *té entropia* $1 - \log\lambda$*.*

Teorema 2.28. *Les distribucions de probabilitat que maximitzen l'entropia són, segons el suport de la variable, les següents:*

1. *amb suport en l'interval finit* $[a, b]$ *la distribució uniforme;*
2. *amb suport a tot* $\mathbb{R}$ *la distribució normal;*
3. *amb suport a* $[0, \infty)$ *la distribució exponencial.*

Proposició 2.29 (Desplaçament i escalat)**.** *L'entropia diferencial es comporta de la manera següent en relació al desplaçament i l'escalat de la variable aleatòria:*

1. $h(X + s) = h(X)$ *per a tot* $s \in \mathbb{R}$*;*
2. $h(aX) = h(X) + \log|a|$ *per a tot* $a \neq 0$*.*

Variables multivaluades. Com en el cas discret es tenen també els conceptes d'entropia diferencial conjunta, condicionada i d'informació mútua:

$$h(X, Y) = -\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x, y)\log f_{X,Y}(x, y)\,dx\,dy,$$

$$h(Y|X) = -\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x, y)\log\frac{f_{X,Y}(x, y)}{f_X(x)}\,dx\,dy,$$

$$I(X;Y) = -\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f_{X,Y}(x, y)\log\frac{f_{X,Y}(x, y)}{f_X(x)f_Y(y)}\,dx\,dy.$$

També se satisfà la regla de la cadena i les relacions entre entropia i informació mútua són les mateixes que en el cas discret:

$$h(X, Y) = h(X) + h(Y|X) = h(Y) + h(X|Y);$$
$$I(X;Y) = h(X) - h(X|Y) = h(Y) - h(Y|X) = h(X, Y) - h(X) - h(Y).$$

Relació amb l'entropia discreta. Es pot discretitzar una variable contínua i establir una relació entre les entropies de totes dues variables.

Sigui X una variable aleatòria contínua amb funció de densitat de probabilitat f_X. Es fixa una amplada Δ i se subdivideix $\mathbb{R}$, o el domini de definició de la variable, en classes d'amplada Δ: intervals $[k\Delta, (k+1)\Delta]$ per a $k \in \mathbb{Z}$. Gràcies al *teorema del valor mitjà per a integrals* en cada classe existeix un x_k tal que

$$\frac{1}{\Delta}\int_{k\Delta}^{(k+1)\Delta} f_X(x)\,dx = f_X(x_k).$$

Sota bones condicions de convergència es té $\sum_{k\in\mathbb{Z}} f_X(x_k)\Delta = \int_{-\infty}^{\infty} f_X(x)\,dx = 1$, de manera que els nombres $p_k = f_X(x_k)\Delta$ són una distribució de probabilitat discreta sobre el conjunt numerable dels nombres x_k i això permet definir una variable discreta X^Δ sobre aquest conjunt posant

$$\Pr(X^\Delta = x_k) := p_k.$$

Per a les variables X amb suport finit la variable discretitzada X^Δ pren només un nombre finit de valors.

Les entropies de les variables X i X^Δ estan relacionades de la manera següent:

Proposició 2.30 (Relació entre entropies)**.** *Si X^Δ és la discretització de la variable X en classes d'amplada Δ es té:*

$$H(X^\Delta) = -\sum_{k\in\mathbb{Z}} p_k \log p_k = -\sum_{k\in\mathbb{Z}} f_X(x_k)\Delta \log f_X(x_k) - \log\Delta.$$

En l'expressió anterior el sumatori del primer terme és una suma de Riemann per a la integral sobre $\mathbb{R}$ de la funció $-f_X(x)\log f_X(x)$, corresponent a la partició de $\mathbb{R}$ en classes d'amplada Δ. En bones condicions d'integrabilitat aquestes sumes de Riemann tendeixen a la integral $-\int_{-\infty}^{\infty} f_X(x)\log f_X(x)\,dx$, que és l'entropia diferencial $h(X)$. Per tant, com a conseqüència de la proposició s'obté el valor del límit:

$$\lim_{\Delta\to 0} H(X^\Delta) + \log\Delta = h(X).$$

Aquesta identitat se sol interpretar de la manera següent: si es quantitza la variable amb classes d'amplada $\Delta = 2^{-n}$, que equival a identificar els valors de la variable amb una precisió de n bits, aleshores la quantitat d'informació que conté la variable discretitzada X^Δ és essencialment igual a $h(X) + n$ bits, per a un valor prou gran de n.

2.4 Processos

Referència: Cover-Thomas [4, Chapter 4]

L'entropia es defineix d'entrada per a una variable aleatòria. Per a diverses variables (variables multivaluades o vectors aleatoris), tal com s'ha vist en la secció anterior, la definició és la mateixa, i apareixen els nous conceptes d'entropies conjuntes, marginals i condicionades i la informació mútua.

Ara es vol definir una entropia associada a un procés estocàstic $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ que mesuri d'alguna manera la quantitat d'informació continguda en el procés. En aquest cas se la sol anomenar *taxa d'entropia* del procés, tot i que moltes vegades se li diu simplement d'entropia. La idea es pot formalitzar de dues maneres: (1) s'agafen vectors finits $(X_1,\ldots,X_n)$, es calcula la seva entropia, que és la informació continguda en tot el vector, es divideix pel nombre de variables n que formen el vector i s'agafa el límit en créixer n; (2) es consideren les entropies relatives $H(X_n|X_1,X_2,\ldots,X_{n-1})$, que representen la informació afegida en conèixer cada nova variable quan ja es coneixen totes les anteriors, i s'agafa el límit en créixer n.

Definició 2.31 (Taxa d'entropia)**.** *Sigui $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ un procés estocàstic. La seva taxa d'entropia, o simplement entropia, es pot definir de les dues maneres següents:*

$$H(\boldsymbol{X}) = \lim_{n\to\infty} \frac{H(X_1, X_2, \ldots, X_n)}{n}, \tag{5}$$

o bé posant

$$H(\boldsymbol{X}) = \lim_{n\to\infty} H(X_n|X_1, X_2, \ldots, X_{n-1}). \tag{6}$$

Els processos i.i.d. tenen taxa d'entropia igual a l'entropia de qualsevol de les variables del procés.

Exemple 2.32 (Processos i.i.d.)**.** *Sigui $\boldsymbol{X} = X_1, X_2, \ldots$ un procés estocàstic independent i idènticament distribuït amb variables $X_i \sim X$. Els dos límits de la definició 2.31 existeixen i són iguals a l'entropia $H(X)$ de la variable X.*

Prova: En efecte, tenint en compte la independència i les distribucions $X_i \sim X$ es té

$$\frac{H(X_1, X_2, \ldots, X_n)}{n} = \frac{H(X_1) + \cdots + H(X_n)}{n} = \frac{nH(X)}{n} = H(X)$$

és una successió constant amb límit $H(X)$. L'altra successió

$$H(X_n|X_1, X_2, \ldots, X_{n-1}) = H(X_n) = H(X)$$

també és constant amb el mateix valor. □

En general els límits de la definició 2.31 poden no existir, de manera que un procés pot no tenir definida una taxa d'entropia. El que no pot passar mai és que tinguin valors diferents. De fet, si el segon existeix aleshores el primer també existeix i tots dos són iguals:

Lema 2.33. *Si el límit* (6) *existeix aleshores el límit* (5) *també existeix i són iguals.*

Prova: És conseqüència d'un resultat de successions que assegura que una successió convergent té suma de Cesàro igual al seu límit: si $(a_n)_{n\geqslant 1}$ és una successió qualsevol, es considera la successió $s_n = \frac{1}{n}\sum_{k=1}^{n} a_k$ de les seves mitjanes aritmètiques. El resultat diu que si (a_n) és convergent aleshores (s_n) també ho és i té el mateix límit:

$$\lim_{n\to\infty} a_n = a \quad \Rightarrow \quad \lim_{n\to\infty} s_n = a.$$

Hi ha successions a_n que no tenen límit però tals que s_n si en té. Per exemple $a_n = (-1)^n$ no és convergent però les mitjanes aritmètiques donen la successió amb termes $s_n = 0$ o $\frac{1}{n}$ segons si n és parell o senar, que té límit zero.

La demostració de l'afirmació anterior és la següent: donat un $\epsilon > 0$ existeix un N tal que $n \geqslant N \Rightarrow |a_n - a| < \frac{\epsilon}{2}$. Es considera el nombre $A = \sum_{k=1}^{N} |a_k - a|$. Existeix un M tal que $n \geqslant M \Rightarrow \frac{1}{n}A \leqslant \frac{\epsilon}{2}$. Aleshores, per a tot $n \geqslant \max\{N, M\}$, es té

$$\begin{aligned} |s_n - a| = \left|\frac{1}{n}\sum_{k=1}^{n} a_k - a\right| = \left|\frac{1}{n}\sum_{k=1}^{n} a_k - \frac{1}{n}\sum_{k=1}^{n} a\right| = \frac{1}{n}\left|\sum_{k=1}^{n}(a_k - a)\right| \leqslant \frac{1}{n}\sum_{k=1}^{n}|a_k - a| \\ = \frac{1}{n}\sum_{k=1}^{N}|a_k - a| + \frac{1}{n}\sum_{k=N+1}^{n}|a_k - a| \leqslant \frac{1}{n}A + \frac{1}{n}(n-N)\frac{\epsilon}{2} < \frac{\epsilon}{2} + \frac{\epsilon}{2} = \epsilon. \end{aligned}$$

L'enunciat és simplement un cas particular de límit de successió de mitjanes aritmètiques: per a la successió (a_n) del límit (6) la seva successió de mitjanes aritmètiques (s_n) és la successió del límit (5). En efecte, el fet que $s_n = \frac{1}{n}\sum_{k=1}^{n} a_n$ és la regla de la cadena. □

Per a una classe prou general de processos, els *processos estacionaris*, la taxa d'entropia existeix sempre:

Teorema 2.34. *En un procés estacionari la successió $H(X_n|X_1, X_2, \ldots, X_{n-1})$ de les entropies condicionades és decreixent; com que és de termes positius és convergent. Es dedueix que tot procés estacionari té taxa d'entropia.*

Prova: En efecte, es té

$$H(X_{n+1}|X_1, \ldots, X_n) \leqslant H(X_{n+1}|X_2, \ldots, X_n) = H(X_n|X_1, \ldots, X_{n-1}).$$

La primera desigualtat degut a què en augmentar la informació prèvia coneguda la quantitat d'informació que sobreviu en la variable X_{n+1} disminueix, tal com s'ha vist a la proposició 2.16, i la segona igualtat degut a l'estacionarietat del procés.

Com que tota successió decreixent fitada inferiorment (per zero: tots els termes són no negatius) té límit, la taxa d'entropia definida pel límit (6) existeix. □

Usant el lema 2.33 es dedueix:

Corol·lari 2.35. *En un procés estacionari el límit* (5) *existeix i és igual a* (6).

De fet, és fàcil veure que, per a processos estacionaris, la successió de (5) també és decreixent, igual com ho és la de (6), i això dona una altra justificació de l'existència del límit (5) sense haver de recórrer a la convergència de (6) i al lema 2.33.

Exemple 2.36 (Taxa d'una cadena de Markov estacionària)**.** *L'entropia d'una cadena de Markov estacionària és $H(\boldsymbol{X}) = H(X_2|X_1)$. En particular, si $\boldsymbol{P} = \big[p(y|x)\big]_{(x,y)\in\mathcal{X}\times\mathcal{X}}$ és la matriu de transició del procés, l'entropia ve donada per l'expressió:*

$$H(\boldsymbol{X}) = -\sum_{x,y\in\mathcal{X}} p(x)p(y|x)\log p(y|x).$$

Prova: En efecte, la propietat de Markov assegura que totes les entropies condicionades a les variables anteriors són $H(X_n|X_1, \ldots, X_{n-1}) = H(X_n|X_{n-1})$, i la estacionarietat fa que totes aquestes siguin iguals a $H(X_2|X_1)$, i per tant el seu límit és aquest mateix nombre. □

Problemes

2.15. Sigui $\boldsymbol{X} = (X_i)_{i\in I}$ un procés estocàstic estacionari, amb conjunt d'índexs els naturals $I = \mathbb{N}$ o bé els enters $I = \mathbb{Z}$. Demostreu que per a tot índex $n \in I$ i tot $k \geqslant 1$ ($k > n$ si $I = \mathbb{N}$ per tal que les X_{n-i} existeixin) es té

$$H(X_n|X_{n-1}, X_{n-2}, \ldots, X_{n-k}) = H(X_n|X_{n+1}, X_{n+2}, \ldots, X_{n+k}).$$

Aquesta propietat s'interpreta de la manera següent: *En un procés estacionari la incertesa coneixent el passat és la mateixa que coneixent el futur.*

2.16. Sigui $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ el procés estocàstic amb variables de Bernoulli següent: les variables d'índex senar són independents entre elles i tenen distribució uniforme $X_{2n-1} \sim \mathrm{Ber}(\frac{1}{2})$; les variables d'índex parell coincideixen amb la variable anterior $X_{2n} = X_{2n-1}$.

1. És de Markov?
2. És estacionari?
3. Calculeu, si existeix, $\lim_{n\to\infty} \frac{1}{n} H(X_1, \ldots, X_n)$.
4. Calculeu, si existeix, $\lim_{n\to\infty} H(X_n | X_1, \ldots, X_{n-1})$.
5. Quina és la taxa d'entropia del procés?
6. Vegeu que existeixen processos $\boldsymbol{Y} = (Y_n)_{n\geqslant 1}$ amb taxa d'entropia $H(\boldsymbol{Y})$ un nombre real $H \geqslant 0$ donat per als quals el límit $\lim_{n\to\infty} H(Y_n | Y_1, \ldots, Y_{n-1})$ no existeix.

2.17. Es considera un passeig aleatori sobre els enters de la forma següent: es comença al zero; el primer pas es fa a dreta o esquerra amb la mateixa probabilitat; en els passos següents s'avança en la mateixa direcció que en el pas anterior amb probabilitat $\frac{9}{10}$ i es canvia de direcció amb probabilitat $\frac{1}{10}$. Sigui $X_0 = 0, X_1, X_2, X_3, \ldots, X_n, \ldots$ la successió de variables aleatòries que donen la posició sobre $\mathbb{Z}$ quan s'han fet n passos.

1. Comproveu que els X_i són una cadena de Markov d'ordre 2.
2. És estacionària?
3. Calculeu $H(X_0, X_1, X_2, \ldots, X_n)$.
4. Calculeu l'entropia del procés.
5. Calculeu el nombre esperat de passos abans de canviar de direcció.

2.18. Considereu una cadena de Markov $\boldsymbol{X} = X_1, X_2, \ldots$ d'ordre 1 amb variables que prenen valors en el conjunt $\mathcal{X} = \{0, 1, 2\} = \mathbb{Z}_3$, la primera variable X_1 té distribució uniforme i la matriu de transició entre estats és

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{bmatrix}.$$

Considereu també el procés estocàstic $\boldsymbol{Z} = Z_1, Z_2, \ldots$ amb $Z_1 = X_1$ i $Z_i = X_i - X_{i-1} \pmod 3$ per a $i \geqslant 2$.

1. La cadena X_i, és estacionària?
2. Calculeu la seva entropia.
3. Calculeu $H(Z_1, \ldots, Z_n)$ i l'entropia del procés $\boldsymbol{Z}$.
4. Calculeu $H(X_n)$ i $H(Z_n)$.
5. Calculeu $H(Z_n | Z_{n-1})$.
6. Les variables Z_n i Z_{n-1}, són independents?

2.19. Sigui $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ un procés estocàstic i.i.d. amb distribució $X_n \sim \mathrm{Ber}(p)$. Considereu el procés $\boldsymbol{Y} = (Y_n)_{n\geqslant 1}$ tal que Y_n dona el nombre d'uns seguits que apareixen fins a X_n. Per exemple, si $X^n = 1011100110\ldots$ aleshores $Y^n = 1012300120\ldots$.

1. Calculeu les taxes d'entropia de tots dos processos.
2. El procés $\boldsymbol{Y}$, és estacionari? és de Markov?
3. Calculeu $H(Y_n|Y_{n-1})$.
4. Quina variable conté més informació, X_n o Y_n?
5. Calculeu $H(Y_n)$.

2.5 Equipartició asimptòtica

Referència: Cover-Thomas [4, Chapter 3]

Una propietat important que poden tenir els processos estocàstics, des del punt de vista de la teoria de la informació, és la

Definició 2.37 (Equipartició asimptòtica)**.** *Un procés estocàstic $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ té la propietat d'equipartició asimptòtica si satisfà:*

$$\lim_{n\to\infty} -\frac{1}{n}\log p(X_1,\ldots,X_n) = H(\boldsymbol{X}) \quad \textit{en probabilitat.}$$

Exemple 2.38. *Tot procés i.i.d. té la propietat d'equipartició asimptòtica.*

Prova: Llei dels grans nombres aplicada a la successió de variables i.i.d. $-\log p(X_i)$. □

Es pot demostrar també que l'equipartició asimptòtica la satisfà un tipus més general de processos, tal com diu el:

Teorema 2.39 (Shanon-McMillan-Breimann)**.** *Tot procés estacionari ergodic té la propietat d'equipartició asimptòtica.*

Aquesta propietat és important ja que hi ha una demostració dels dos teoremes fonamentals de la teoria de la informació, el teorema de codificació de font i el teorema de codificació de canal, que s'aplica a tot procés estocàstic que tingui la propietat d'equipartició asimptòtica. Tot i això, hi ha altres demostracions que són més senzilles i útils ja que donen una idea de com obtenir bones codificacions a la pràctica, especialment pel que fa a la codificació de font.

El concepte principal que es fa servir en les demostracions que usen la propietat d'equipartició asimptòtica és el de *seqüència típica*, que són les que apareixen amb probabilitat alta entre totes les seqüències que pot generar el procés.

Definició 2.40 (Seqüència típica)**.** *Sigui $\boldsymbol{X}$ un procés estocàstic de variables X_n que prenen valors en el conjunt finit $\mathcal{X}$. Una seqüència ε-típica és un element $x^n \in \mathcal{X}^n$ tal que*

$$2^{-n(H(\boldsymbol{X})+\varepsilon)} \leqslant p(x^n) \leqslant 2^{-n(H(\boldsymbol{X})-\varepsilon)}.$$

Es denotarà $\mathrm{TS}_\varepsilon^{(n)} \subseteq \mathcal{X}^n$ el conjunt de les seqüències ε-típiques de longitud n. Quan un procés satisfà la propietat d'equipartició asimptòtica aquest conjunt conté un nombre petit d'elements i, en canvi, la probabilitat que la seqüència que genera el procés sigui típica s'acosta a 1. Més concretament:

Proposició 2.41. *Sigui $\boldsymbol{X}$ un procés estocàstic que satisfà la propietat d'equipartició asimptòtica. Sigui $\varepsilon > 0$. Aleshores*

1. $\Pr\left(\mathrm{TS}_\varepsilon^{(n)}\right) > 1-\varepsilon$ *per a n suficientment gran;*
2. $\left|\mathrm{TS}_\varepsilon^{(n)}\right| \leqslant 2^{n(H(\boldsymbol{X})+\varepsilon)}$ *per a tot $n \geqslant 1$;*
3. $(1-\varepsilon)2^{-n(H(\boldsymbol{X})-\varepsilon)} \leqslant \left|\mathrm{TS}_\varepsilon^{(n)}\right|$ *per a n suficientment gran.*

Prova: Agafant logaritmes i dividint per n la condició de seqüència típica equival a

$$\left|-\frac{1}{n}\log p(x^n) - H(\boldsymbol{X})\right| \leqslant \varepsilon.$$

El fet que $\boldsymbol{X}$ satisfaci la propietat d'equipartició asimptòtica equival a què

$$\Pr\left(X^n \in \mathrm{TS}_\varepsilon^{(n)}\right) = \Pr\left(\left|-\frac{1}{n}\log p(X^n) - H(\boldsymbol{X})\right| \leqslant \varepsilon\right) \to 1 \quad \text{quan} \quad n \to \infty.$$

1. Agafant el nombre $1-\varepsilon < 1$ existeix un enter N tal que per a tot $n \geqslant N$ es té

$$\Pr\left(X^n \in \mathrm{TS}_\varepsilon^{(n)}\right) > 1-\varepsilon.$$

2. Per a tot n es té

$$1 = \Pr(X^n) \geqslant \Pr\left(\mathrm{TS}_\varepsilon^{(n)}\right) = \sum_{x^n \in \mathrm{TS}_\varepsilon^{(n)}} p(x^n) \geqslant \sum_{x^n \in \mathrm{TS}_\varepsilon^{(n)}} 2^{-n(H(\boldsymbol{X})+\varepsilon)} = \left|\mathrm{TS}_\varepsilon^{(n)}\right| 2^{-n(H(\boldsymbol{X})+\varepsilon)},$$

   i d'aquí es dedueix la desigualtat de l'enunciat multiplicant per $2^{n(H(\boldsymbol{X})+\varepsilon)}$.

3. Per a n suficientment gran es compleix la desigualtat del primer apartat i es té

$$1-\varepsilon < \Pr\left(\mathrm{TS}_\varepsilon^{(n)}\right) = \sum_{x^n \in \mathrm{TS}_\varepsilon^{(n)}} p(x^n) \leqslant \sum_{x^n \in \mathrm{TS}_\varepsilon^{(n)}} 2^{-n(H(\boldsymbol{X})-\varepsilon)} = \left|\mathrm{TS}_\varepsilon^{(n)}\right| 2^{-n(H(\boldsymbol{X})-\varepsilon)},$$

   i es dedueix com abans la desigualtat de l'enunciat. □

Seqüències típiques conjuntes. Es considera ara un parell de processos $\boldsymbol{X} = X_1, X_2, \ldots$ i $\boldsymbol{Y} = Y_1, Y_2, \ldots$ que prenen valors en alfabets $\mathcal{X}$ i $\mathcal{Y}$, respectivament. Per a cada $n \geqslant 1$ es consideren els parells de vectors aleatoris X^n i Y^n amb probabilitats conjuntes $p(x^n, y^n)$, marginals $p(x^n)$ i $p(y^n)$ i condicionades $p(y^n|x^n)$.

Donat un llindar $\varepsilon > 0$ es defineix el conjunt de seqüències típiques conjuntes com:

Definició 2.42 (Seqüències típiques conjuntes)**.** *Les seqüències ε-típiques conjuntes es defineixen com els parells $(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n$ que compleixen alhora les tres propietats següents:*

- *x^n és una seqüència ε-típica per a X^n;*
- *y^n és una seqüència ε-típica per a Y^n;*
- *(x^n, y^n) és una seqüència ε-típica per al parell de variables X^n, Y^n.*

Es denotarà $\mathrm{JTS}_\varepsilon^{(n)} \subseteq \mathcal{X}^n \times \mathcal{Y}^n$ el conjunt d'aquestes seqüències típiques conjuntes. La seva descripció precisa és:

$$\begin{aligned}\mathrm{JTS}_\varepsilon^n = \big\{&(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \\ &2^{-n(H(\boldsymbol{X})+\varepsilon)} \leqslant p(x^n) \leqslant 2^{-n(H(\boldsymbol{X})-\varepsilon)}, 2^{-n(H(\boldsymbol{Y})+\varepsilon)} \leqslant p(y^n) \leqslant 2^{-n(H(\boldsymbol{Y})-\varepsilon)}, \\ &2^{-n(H(\boldsymbol{X},\boldsymbol{Y})+\varepsilon)} \leqslant p(x^n, y^n) \leqslant 2^{-n(H(\boldsymbol{X},\boldsymbol{Y})-\varepsilon)}\big\} \\ = \big\{&(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n : \\ &\left|\frac{-\log p(x^n)}{n} - H(\boldsymbol{X})\right| \leqslant \varepsilon, \left|\frac{-\log p(y^n)}{n} - H(\boldsymbol{Y})\right| \leqslant \varepsilon, \\ &\left|\frac{-\log p(x^n, y^n)}{n} - H(\boldsymbol{X}, \boldsymbol{Y})\right| \leqslant \varepsilon\Big\}.\end{aligned}$$

L'aplicació més important de les seqüències típiques conjuntes és per poder donar un esquema de decisió per a la descodificació de codis de canal i demostrar el teorema de codificació de canal usant aquest esquema, tal com es veurà a la secció 4.4. Aquesta és la demostració original de Shannon del teorema. En la proposició següent es veuen les propietats que es faran servir en aquesta demostració.

Proposició 2.43 (Propietats)**.** *El conjunt de les seqüències típiques conjuntes satisfà:*

- $\Pr\big(\mathrm{JTS}_\varepsilon^{(n)}\big) \geqslant 1 - \varepsilon$*;*
- $\big|\mathrm{JTS}_\varepsilon^{(n)}\big| \leqslant 2^{n(H(X,Y)+\varepsilon)}$*;*
- $\Pr\big((\widetilde{X}^n, \widetilde{Y}^n) \in \mathrm{JTS}_\varepsilon^{(n)}\big) \leqslant 2^{-n(I(X;Y)-3\varepsilon)}$*, si $\widetilde{X}$ i $\widetilde{Y}$ són un parell de variables independents amb les mateixes distribucions marginals que X i Y.*

Prova: Cover-Thomas [4, Theorem 7.6.1]. □

2.6 Problemes Complementaris

2.20. *Fitació dels coeficients binomials.* Siguin $\boldsymbol{p} = (p, 1-p)$ una distribució de Bernoulli i $\boldsymbol{q} = (\frac{1}{2}, \frac{1}{2})$ la distribució uniforme. Demostreu que $D(\boldsymbol{q}\|\boldsymbol{p}) \geqslant D(\boldsymbol{p}\|\boldsymbol{q})$ amb igualtat si, i només si, la distribució $\boldsymbol{p}$ també és la uniforme.

2.21. *Fitació dels coeficients binomials.* Demostreu les desigualtats següents:

$$\frac{1}{n+1}2^{nH(p)} \leqslant \binom{n}{k} \leqslant 2^{nH(p)} \quad \text{amb} \quad p = \frac{k}{n} \qquad \text{per a tot} \quad n \geqslant 1,\ 0 \leqslant k \leqslant n.$$

Indicació: considereu l'expansió binomial de $\big(p + (1-p)\big)^n$ i vegeu que els seus termes creixen fins al k-èsim i després decreixen.

2.22. Demostreu que per a tot $\lambda < \frac{1}{2}$ el nombre de paraules que conté una bola de radi $n\lambda$ dins de $\{\mathtt{0},\mathtt{1}\}^n$ està fitat per

$$\sum_{i=0}^{\lfloor n\lambda \rfloor} \binom{n}{i} \leqslant 2^{nH(\lambda)}.$$

2.23. Calculeu l'entropia de la variable aleatòria X amb valors en els nombres naturals $\mathcal{X} = \mathbb{N} = \{1, 2, 3, \dots\}$ que compta el nombre de tirades d'una moneda trucada amb probabilitat p de treure una cara fins a obtenir la primera cara.

2.24. Es considera una variable aleatòria contínua X amb distribució uniforme sobre l'interval $[0, 1]$ que pren valors en el conjunt de les distribucions de Bernoulli donant la probabilitat del valor 1. Calculeu l'esperança de l'entropia $\mathbb{E}\big[H(X)\big]$.

2.25. Demostreu que si X, Y són variables independents que prenen valors en un mateix conjunt amb la mateixa distribució de probabilitats, aleshores

$$\Pr(X = Y) \geqslant 2^{-H(X)}$$

amb igualtat si, i només si, la distribució és uniforme.
INDICACIÓ: Podeu usar la *desigualtat de Jensen*.

Demostreu també que si les variables independents tenen distribucions $\boldsymbol{p}$ i $\boldsymbol{q}$ aleshores

$$\Pr(X = Y) \geqslant \max\left\{2^{-H(X)-D(\boldsymbol{p}\|\boldsymbol{q})}, 2^{-H(Y)-D(\boldsymbol{q}\|\boldsymbol{p})}\right\}.$$

2.26. Sigui $X_1, X_2, \dots$ una successió de variables aleatòries i.i.d. Calculeu

$$\lim_{n\to\infty} \sqrt[n]{p(X_1, X_2, \dots, X_n)}.$$

Digueu quant val aquest límit si la distribució de les variables és $\mathrm{Ber}(p)$ i dibuixeu la gràfica dels seus valors per a $p \in [0, 1]$.

2.27. Considereu la cadena binària `11011100101110111` formada concatenant els dígits binaris dels enters entre 1 i $7 = 2^3 - 1$. Considereu les variables aleatòries $\boldsymbol{X}$ que corresponen a agafar dos dígits consecutius de la seqüència, i Y i Z que donen el primer i segon dígit de $\boldsymbol{X}$, respectivament. Com en el problema **2.10** es convé que el dígit següent de l'últim és el primer.

1. calculeu les entropies $H(\boldsymbol{X})$, $H(Y)$, i $H(Z)$;
2. calculeu les entropies condicionades $H(Z|Y = x)$ per a $x = 0$ i $x = 1$, i $H(Z|Y)$;
3. calculeu la informació mútua $I(Y; Z)$;
4. calculeu les entropies relatives $D(\boldsymbol{p}\|\boldsymbol{q})$ i $D(\boldsymbol{q}\|\boldsymbol{p})$, on $\boldsymbol{p}$ i $\boldsymbol{q}$ són les distribucions de probabilitat de les variables Y i Z, respectivament;
5. calculeu les entropies relatives $D(\boldsymbol{p}\|\boldsymbol{q})$ i $D(\boldsymbol{q}\|\boldsymbol{p})$, on $\boldsymbol{p}$ i $\boldsymbol{q}$ són les distribucions de probabilitat de les variables $Z|Y = 0$ i $Z|Y = 1$, respectivament;

6. demostreu que, per a la cadena binària formada de la mateixa manera concatenant els dígits binaris dels enters entre 1 i $2^n - 1$, es té

$$H(Y) = H\left(\frac{2^{n-1}(n-2)+1}{2^n(n-1)+1}, \frac{2^{n-1}n}{2^n(n-1)+1}\right)$$

i calculeu $\lim_{n\to\infty} H(Y)$.

2.28. En una urna hi ha boles de tres colors diferents; siguin a, b i c el nombre de boles de cada color. Sigui X la variable aleatòria corresponent a extraure una bola de l'urna. Siguin Y i Z les variables aleatòries corresponents a extraure una segona bola, en la variable Y amb reemplaçament i en la variable Z sense reemplaçament.

1. Calculeu les entropies $H(X)$, $H(Y)$ i $H(Z)$.
2. Calculeu les entropies condicionades $H(Y|X)$ i $H(Z|X)$.
3. Trobeu els valors en el cas particular amb nombres de boles 3, 6 i 1.

Discutiu la situació si en comptes de tres colors hi ha boles de més colors diferents i també si en comptes de fer dues extraccions es fan més extraccions, en un cas amb reemplaçament i en un altre sense.

2.29. Demostreu que $H(X,Y|Z) = H(X|Z) + H(Y|X,Z)$. Justifiqueu-ho primer amb un diagrama de Venn que representi les entropies de les tres variables en joc i després demostreu-ho rigorosament a partir del sumatori que defineix l'entropia condicionada i també aplicant la regla de la cadena.

Generalitzeu-ho al cas de més variables:

$$H(X_1,\ldots,X_n|Z) = \sum_{i=1}^{n} H(X_i|X_1,\ldots,X_{i-1},Z)$$

2.30. L'objectiu d'aquest problema és demostrar que l'increment de les condicions disminueix l'entropia de les variables condicionades: donades tres variables aleatòries X, Y, Z es compleix la desigualtat

$$H(Z|X,Y) \leqslant H(Z|Y).$$

Per obtenir aquesta desigualtat, demostreu primer que

$$H(Y,Z|X) \leqslant H(Y|X) + H(Z|X).$$

INDICACIÓ: podeu considerar $H(Y,Z|X)$ com una esperança respecte la variable X o també comparar la distribució conjunta de totes tres variables amb la distribució $p(x)p(y|x)p(z|x)$.

2.31. Sigui $n \geqslant 3$. Siguin $X_1,\ldots,X_{n-1}$ variables aleatòries binàries independents amb distribució uniforme. Sigui $X_n = 1$ si $\sum_{i=1}^{n-1} X_i$ és senar i $X_n = 0$ si és parell.

1. Vegeu que X_i i X_j són independents per a tot parell $i \neq j$ amb $1 \leqslant i, j \leqslant n$.
2. Calculeu $H(X_i, X_j)$ si $i \neq j$.
3. Calculeu $H(X_1, \ldots, X_n)$.

2.32. *Desigualtat de Fano.* Siguin X i Y dues variables aleatòries que prenen valors en el mateix conjunt $\mathcal{X}$. Sigui $P_e = \Pr(X \neq Y) \in [0,1]$ la probabilitat d'error en estimar una variable usant l'altra. Demostreu que

$$H(X|Y) \leqslant H(P_e) + P_e\big(\log|\mathcal{X}| - 1\big).$$

2.33. *Entropia relativa condicionada.* Donades dues distribucions de probabilitat conjuntes $p(x,y)$ i $q(x,y)$ en el mateix conjunt $\mathcal{X} \times \mathcal{Y}$, producte cartesià de dos conjunts, es defineix l'entropia relativa condicionada com

$$D\big(p(y|x)\|q(y|x)\big) = \sum_{(x,y)\in\mathcal{X}\times\mathcal{Y}} p(x,y)\log\frac{p(y|x)}{q(y|x)}.$$

Demostreu les propietats següents:

1. Positivitat: $D\big(p(y|x)\|q(y|x)\big) \geqslant 0$;
2. Regla de la cadena: $D\big(p(x,y)\|q(x,y)\big) = D\big(p(x)\|q(x)\big) + D\big(p(y|x)\|q(y|x)\big)$.

2.34. La *informació mútua condicionada* es defineix posant

$$I(X;Y|Z) := \sum_{x,y,z} p(x,y,z)\log\frac{p(x,y|z)}{p(x|z)p(y|z)} = H(X|Z) - H(X|Y,Z).$$

1. Demostreu que $I(X;Y|Z) \geqslant 0$.
2. Comproveu la igualtat entre les dues expressions que es donen a la definició.
3. Demostreu la regla de la cadena $I(X,Y;Z) = I(X;Z) + I(Y;Z|X)$.
4. Generalitzeu la regla de la cadena a més variables trobant una expressió per a la informació mútua $I(X_1, X_2, \ldots, X_n; Y)$ i demostrant que és correcta.
5. Doneu exemples de variables per a les quals
   (a) $I(X;Y|Z) < I(X;Y)$;
   (b) $I(X;Y|Z) = I(X;Y)$;
   (c) $I(X;Y|Z) > I(X;Y)$.
6. Discutiu la conveniència de fer servir diagrames de Venn per representar entropies quan es treballa amb tres o més variables.

2.35. Siguin X, Y, Z tres variables que formen una cadena de Markov: $p(z|x,y) = p(z|y)$. Això equival a dir que Z només depèn de Y però no de X. Demostreu la desigualtat

$$I(Z;X) \leqslant I(Y;X)$$

i digueu perquè es coneix pel nom de *desigualtat del processament de dades*.

2.36. Sigui $\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$ un vector aleatori de variables binàries que no pren cap valor de $\mathcal{X}^n$ amb un nombre senar d'uns i pren tots els valors amb nombre parell d'uns amb la mateixa probabilitat. Calculeu

$$I(X_2; X_1), \quad I(X_3; X_2|X_1), \quad \ldots \quad I(X_n; X_{n-1}|X_1, \ldots, X_{n-2})$$

2.37. De les afirmacions següents digueu quines són certes i quines falses; les certes, demostreu-les, i, de les falses, doneu-ne un contraexemple.

1. $I(X;Y) = H(Y)$ si, i només si, Y és funció de X: $Y = g(X)$;
2. si X pren valors a $\mathcal{X}$ i Y pren valors a $\mathcal{Y}$ i es té una funció $g\colon \mathcal{X} \to \mathcal{Y}$ exhaustiva i $Y = g(X)$ aleshores $H(X) = H(Y)$;
3. Si X i Y prenen valors reals i $Z = X + Y$ aleshores $I(X;Y|Z) = I(X;Y)$;
4. Si X i Y prenen valors reals i $Z = X + Y$ aleshores $H(X|Z) = H(Y|Z)$;
5. Si X i Y prenen valors reals i $Z = X + Y$ aleshores $H(Z|X) = H(Z|Y)$.

2.38. Sigui $\boldsymbol{X} = X_1, X_2, \ldots, X_n, X_{n+1}, \ldots$ un procés estocàstic estacionari. Es denota $X^n = X_1, X_2, \ldots, X_n$ i $H(\boldsymbol{X})$ la taxa d'entropia del procés.

1. Quin és el valor de $H(\boldsymbol{X})$ si les variables són independents?
2. Justifiqueu que $H(X^n) = H(X_n|X^{n-1}) + H(X^{n-1})$.
3. Demostreu que $H(X_n|X^{n-1}) \leqslant H(X_i|X^{i-1})$ per a tot $i = 1, \ldots, n-1$.
4. Feu servir els resultats anteriors per demostrar que

$$\frac{H(X^n)}{n} \leqslant \frac{H(X^{n-1})}{n-1} \quad \text{per a} \quad n = 2, 3, \ldots$$

El quart apartat diu que la successió $\frac{1}{n}H(X^n)$ és decreixent. Com que l'entropia condicionada és no negativa això implica que la successió té límit i dona una nova demostració de què els processos estacionaris tenen taxa d'entropia.

2.39. Sigui $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ un procés estocàstic i.i.d. amb distribució $X_n \sim \mathrm{Ber}(p)$ que genera seqüències binàries. A partir d'aquest procés es construeixen altres quatre processos tal com es descriu a continuació:

- $\boldsymbol{Y} = (Y_n)_{n\geqslant 1}$ genera les seqüències d'enters $\geqslant 0$ que s'obtenen a partir de les seqüències binàries generades per $\boldsymbol{X}$ deixant els mateixos zeros i canviant les cadenes d'uns seguits per la seva longitud: el nombre d'uns consecutius, que és un nombre enter $\geqslant 1$;
- $\boldsymbol{Z} = (Z_n)_{n\geqslant 1}$ genera les seqüències binàries que s'obtenen a partir de les seqüències d'enters generades per $\boldsymbol{Y}$ deixant els zeros iguals i canviant els enters $\geqslant 1$ per uns;
- $\boldsymbol{R} = (R_n)_{n\geqslant 1}$ genera les seqüències d'enters $\geqslant 0$ que s'obtenen a partir de les seqüències binàries generades per $\boldsymbol{X}$ posant successivament els nombres de zeros i d'uns seguits que van apareixent;

– $\boldsymbol{S} = (S_n)_{n\geqslant 1}$ genera les seqüències d'enters $0 \leqslant k \leqslant N$ que s'obtenen a partir de les seqüències binàries generades per $\boldsymbol{X}$ posant els nombres de zeros i d'uns seguits que van apareixent, però ara amb un nombre N fixat que s'agafa com a fita per al valor màxim de les variables S_n.

Els processos $\boldsymbol{R}$ i sobretot $\boldsymbol{S}$ (ja que a la pràctica s'ha de posar una fita N) corresponen a la tècnica de codificació coneguda pel nom de *run-length encoding*.

Per entendre millor la relació entre els processos i com s'obtenen tots a partir de $\boldsymbol{X}$ es dona a continuació un exemple de seqüència generada per $\boldsymbol{X}$ i les seqüències que li corresponen en els altres processos:

$$\begin{aligned}\boldsymbol{X} &= 1101001110000010110\ldots\\ \boldsymbol{Y} &= 201003000001020\ldots\\ \boldsymbol{Z} &= 101001000001010\ldots\\ \boldsymbol{R} &= 0211235112\ldots\\ \boldsymbol{S} &= 021123302112\ldots\end{aligned}$$

1. Demostreu que, si existeixen, les taxes d'entropia d'aquests processos satisfan les desigualtats següents:

$$H(\boldsymbol{Z}) \leqslant H(\boldsymbol{X}) \leqslant H(\boldsymbol{Y}) \leqslant H(\boldsymbol{R}), \quad H(\boldsymbol{X}) \leqslant H(\boldsymbol{S}) \leqslant H(\boldsymbol{R}).$$

2. Calculeu
   (a) la distribució de probabilitat de les variables Z_n;
   (b) la distribució de probabilitat de les variables Y_n;
   (c) la distribució de probabilitat de les variables R_n;
   (d) la distribució de probabilitat de les variables S_n,

   i digueu quina és l'entropia de cadascuna d'aquestes variables.
3. Estudieu les relacions de dependència de les variables en tots quatre processos, dient si les variables són independents i si el procés és de Markov.
4. Calculeu les taxes d'entropia de tots quatre processos.

3 Codificació de font

En Teoria de la Informació una *font d'informació* (discreta) és un dispositiu que emet informació en forma de seqüències de símbols d'un alfabet d'acord amb unes distribucions de probabilitat determinades.

Les fonts d'informació es diuen amb memòria o sense memòria segons si les probabilitats dels símbols de la seqüència emesa depenen o no dels símbols anteriors. Com a model matemàtic s'agafa el següent:

- El model d'una *font sense memòria* és una variable aleatòria X que pren valors en un conjunt finit $\mathcal{X}$: l'alfabet de la font. La font emet aleatòriament seqüències de símbols d'aquest alfabet $\mathtt{x}_1\mathtt{x}_2\mathtt{x}_3\cdots$ de manera independent amb probabilitat de cada lletra determinada per la distribució de la variable: $p(\mathtt{x}_i) = \Pr(X = \mathtt{x}_i)$.
- El model d'una *font amb memòria* és un procés estocàstic $\boldsymbol{X} = (X_n)_{n\geqslant 1}$ de variables aleatòries X_i que prenen valors en un mateix conjunt finit $\mathcal{X}$: l'alfabet de la font. La font emet aleatòriament seqüències $\mathtt{x}_1\mathtt{x}_2\mathtt{x}_3\cdots$ de símbols de $\mathcal{X}$ amb les probabilitats de cada paraula $p(\mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n)$ determinades per la distribució del vector $X^n = (X_1, \ldots, X_n)$.

El cas sense memòria es pot veure també com un procés estocàstic i.i.d. amb variables independents totes amb la mateixa distribució de la variable X.

Se suposarà que les variables aleatòries en una font no són constants.

3.1 Codis de font

Referència: Brunat-Ventura [3, Capítol 4]

L'objectiu de la codificació de font és codificar de manera eficient les seqüències de símbols de $\mathcal{X}$ emeses per una font amb un codi sobre un alfabet $\mathbb{A}$ donat. Sempre que no es digui explícitament el contrari se suposarà que es codifica sobre l'alfabet binari $\mathbb{A} = \{\mathtt{0}, \mathtt{1}\}$, que és la situació més habitual a la pràctica. És a dir, es consideraran codis de font binaris.

Aquí, el conjunt $\mathcal{X}$ dels símbols emesos per la font fa el paper del conjunt $\mathcal{M}$ de missatges a codificar en la notació de la secció 1.2. Es denotarà $M = |\mathcal{X}|$ el nombre de símbols emesos per la font (el nombre d'elements de l'alfabet de la font) i la lletra q denotarà el nombre de símbols de l'alfabet del codi $\mathbb{A}$. Per tant el codi de font codifica els símbols d'un alfabet M-ari $\mathcal{X}$ amb paraules d'un alfabet q-ari $\mathbb{A}$.

En la major part d'aquesta secció es consideraran fonts sense memòria, donades en la forma d'una variable aleatòria X que pren valors en un alfabet $\mathcal{X}$.

Definició 3.1 (Codi de font)**.** *Un* codi de font *(binari) per a la variable aleatòria X (font sense memòria) és un codi $\mathcal{C}_X \subset \{\mathtt{0}, \mathtt{1}\}^*$ amb una codificació* $\mathrm{enc}\colon \mathcal{X} \to \mathcal{C}_X$.

Els termes *codi de font* i *codificació de font* se solen fer servir de manera intercanviable per referir-se tant a l'aplicació de codificació enc com al codi binari que és la imatge d'aquesta aplicació: el codi $\mathcal{C}_X = \mathrm{enc}(\mathcal{X}) \subset \{\mathtt{0}, \mathtt{1}\}^*$.

Com amb qualsevol codificació, un codi de font es pot fer servir per codificar seqüències $\boldsymbol{x} = \mathtt{x}_1\cdots\mathtt{x}_r \in \mathcal{X}^*$ concatenant les paraules codi corresponents:

$$\mathrm{enc}^*(\boldsymbol{x}) = \mathrm{enc}^*(\mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_r) = \mathrm{enc}(\mathtt{x}_1) \| \mathrm{enc}(\mathtt{x}_2) \| \cdots \| \mathrm{enc}(\mathtt{x}_r),$$

on $\mathrm{enc}^*\colon \mathcal{X}^* \to \{0,1\}^*$ denota l'aplicació de codificació de seqüències que estén la codificació de símbols enc concatenant les paraules codi de cada símbol.

Per treballar amb codis de font sovint s'adopta el conveni de notació següent, que simplifica moltes expressions. A través dels subíndexos s'enumeren de manera coherent els símbols $\mathtt{x}_1, \mathtt{x}_2, \cdots, \mathtt{x}_M \in \mathcal{X}$ emesos per la font, les seves probabilitats $p_1, p_2, \ldots, p_M$, les paraules del codi de font corresponents $\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_M \in \mathcal{C}_X$, i les seves longituds $\ell_1, \ell_2, \ldots, \ell_M$, de manera que $p_i = p(\mathtt{x}_i) = \Pr(X = \mathtt{x}_i)$, $\boldsymbol{c}_i = \mathrm{enc}(\mathtt{x}_i)$ i $\ell_i = \ell(\boldsymbol{c}_i)$. De vegades s'agafen els valors $\mathtt{x}_i$ ordenats de tal manera que les probabilitats i/o les longituds de les paraules codi quedin en un ordre determinat (com per exemple al lema 3.10 o al teorema 3.11).

Exemple 3.2 (Codi de bloc)**.** *Una font X que emet $M = |\mathcal{X}|$ símbols diferents es pot codificar amb un codi de bloc $\mathcal{C}_X \subseteq \{0,1\}^n$ format per un conjunt qualsevol de paraules de longitud n, per a tot $n \geqslant \lceil \log M \rceil$.*

Prova: Com que $\{0,1\}^n$ conté 2^n paraules, la condició sobre n per tal que hi hagi almenys M paraules d'aquesta longitud és que $M \leqslant 2^n \Leftrightarrow \log M \leqslant n \Leftrightarrow \lceil \log M \rceil \leqslant n$. Naturalment, per poder codificar amb un codi de bloc q-ari la condició seria que $n \geqslant \lceil \log_q M \rceil$. □

Exemple 3.3 (*Codi de Shannon*)**.** *Sigui X una font amb probabilitats no nul·les $p_i \neq 0$. Siguin $\ell_i := \lceil -\log p_i \rceil$. Un* codi de Shannon *per a la font X és un codi amb paraules de longituds $\ell(\boldsymbol{c}_i) = \ell_i$.*

Prova: Per veure que aquesta definició és correcta s'ha de comprovar que existeixen codis amb paraules de les longituds previstes en la definició: s'ha de veure que les longituds $\ell_i = \lceil -\log p_i \rceil$ satisfan la desigualtat de Kraft-McMillan. Com que la part entera per excés és més gran o igual que el nombre, es té

$$\ell_i = \lceil -\log p_i \rceil \geqslant -\log p_i \quad \Rightarrow \quad -\ell_i \leqslant \log p_i \quad \Rightarrow \quad 2^{-\ell_i} \leqslant p_i \quad \Rightarrow \quad \sum 2^{-\ell_i} \leqslant \sum p_i = 1.$$

Per tant existeixen codis (prefixos si es vol) amb paraules d'aquestes longituds.

Anàlogament es podrien construir codis de Shannon q-aris (prefixos si es vol) agafant paraules de longituds $\ell_i = \lceil -\log_q p_i \rceil$.

Observi's que els codis de Shannon no estan definits en fonts que tenen probabilitats zero, ja que les longituds de les paraules corresponents haurien de ser infinites. □

Definició 3.4 (Longitud mitjana)**.** *La* longitud mitjana *d'un codi de font $\mathcal{C}_X$ es defineix com l'esperança de la longitud de la paraula binària que codifica cada símbol emès per la font:*

$$\widetilde{\ell}(\mathcal{C}_X) = \mathbb{E}\big[\ell(\mathrm{enc}(X))\big] = \sum_{\mathtt{x} \in \mathcal{X}} p(\mathtt{x}) \ell\big(\mathrm{enc}(\boldsymbol{x})\big) = \sum_{i=1}^{M} p_i \ell_i.$$

Exemple 3.5. *Sigui X una font que emet $M = 4$ símbols amb probabilitats $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}$.*

- *El codi de bloc $\mathcal{C}_X = \{00, 01, 10, 11\}$ té longitud mitjana 2;*
- *El codi de Shannon té paraules de longituds $1, 2, 3$ i 3; per exemple es pot agafar el codi $\mathcal{C}_X = \{0, 10, 110, 111\}$. Té longitud mitjana $\frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 2 + \frac{1}{8} \cdot 3 + \frac{1}{8} \cdot 3 = \frac{7}{4} = 1.75$.*

Definició 3.6 (Longitud mitjana mínima)**.** *La* longitud mitjana mínima *d'una font X es defineix com l'ínfim de les longituds mitjanes de tots els codis de font possibles:*

$$\widetilde{\ell}(X) = \inf\left\{\widetilde{\ell}(\mathcal{C}_X) : \mathcal{C}_X \text{ codi de font per a } X\right\}.$$

Definició 3.7 (Codi òptim)**.** *Un* codi òptim *per a la font X és un codi de font amb longitud mitjana igual a la longitud mitjana mínima de la font:*

$$\widetilde{\ell}(\mathcal{C}_X) = \sum_{i=1}^{M} p_i \ell_i = \widetilde{\ell}(X)$$

i tal que no existeix cap altre codi amb alguna paraula de longitud menor.

La segona condició de la definició demana que no existeix cap altre codi de font $\mathcal{C}'_X$ amb $\widetilde{\ell}(\mathcal{C}'_X) = \widetilde{\ell}(X)$ format per paraules de longituds ℓ'_i, amb $\ell'_i \leqslant \ell_i$ per a tot i, però amb almenys un índex j tal que $\ell'_j < \ell_j$. Aquesta condició només pot afectar codis amb probabilitats zero: en un codi que la satisfaci ha de ser necessàriament $p_j = 0$. (exercici **3.1**). Per tant, per a fonts que prenen valors en el seu suport l'única condició per a ser òptim és que la longitud mitjana sigui la mínima possible.

Teorema 3.8. *Tota font té codis òptims.*

Prova: Observi's que en la definició de longitud mitjana mínima s'ha agafat l'ínfim ja que hi ha infinits codis de font possibles i no és immediat que aquest ínfim l'assoleixi la longitud mitjana d'algun codi de font. Només cal veure que aquest mínim sí que s'assoleix sempre.

S'agafa un codi de font $\mathcal{C}$. Per exemple, sempre es pot agafar un codi de bloc. Aquest codi tindrà una longitud mitjana $\ell(\mathcal{C})$.

Sigui $\mathcal{C}_X$ un codi de font qualsevol amb paraules de longituds ℓ_i. Suposi's que té longitud mitjana més petita: $\widetilde{\ell}(\mathcal{C}_X) \leqslant \widetilde{\ell}(\mathcal{C})$. Per a cada probabilitat no nul·la $p_k \neq 0$ es té

$$p_k \ell_k \leqslant \sum_{i=1}^{M} p_i \ell_i = \widetilde{\ell}(\mathcal{C}_X) \leqslant \widetilde{\ell}(\mathcal{C}) \quad \Rightarrow \quad \ell_k \leqslant \frac{\widetilde{\ell}(\mathcal{C})}{p_k}.$$

Per tant, les longituds ℓ_k de les paraules $\boldsymbol{c}_k \in \mathcal{C}_X$ que codifiquen símbols amb probabilitat no nul·la estan fitades.

Si totes les probabilitats són no nul·les es poden agafar tots els codis possibles amb paraules de longituds

$$\ell_k \leqslant \frac{\widetilde{\ell}(\mathcal{C})}{p_k}, \qquad k = 1, \ldots, M.$$

D'aquests codis n'hi ha només un nombre finit. Entre tots ells, els que tinguin longitud mitjana mínima són òptims. Aquí com que no hi ha probabilitats zero la segona condició de la definició dels codis òptims no s'aplica.

Suposi's ara que hi ha probabilitats nul·les. Ordenant-les en ordre decreixent es pot suposar que les $M' < M$ primeres són no nul·les i les últimes $M - M'$ són zero. S'agafen tots els codis *no maximals* de M' paraules amb

$$\ell_k \leqslant \frac{\widetilde{\ell}(\mathcal{C})}{p_k}, \qquad k = 1, \ldots, M'.$$

D'aquests codis n'hi ha un nombre finit. Entre tots ells s'agafen els que tinguin longitud mitjana mínima. Cadascun d'aquests codis es pot estendre a un codi de M paraules afegint-li $M - M'$ paraules: a un codi no maximal se li poden afegir tantes paraules com es vulgui tenint també un codi. Per exemple, agafant un ℓ tal que $(M - M')q^{-\ell} \leqslant 1 - \sum_{k=1}^{M'} 2^{-\ell_k}$ es podrien afegir $M - M'$ paraules totes de la mateixa longitud ℓ. Aquestes paraules afegides són les que codifiquen els símbols de probabilitat zero. Ara, per a cadascun d'aquests codis es busquen tots els codis que tinguin les primeres M' paraules de la mateixa longitud i les altres $M - M'$ de longitud més petita que les del codi obtingut. Quan cap d'aquestes $M - M'$ paraules ja no es pugui escurçar s'ha arribat a un codi òptim. □

Corol·lari 3.9. *Tota font té codis òptims que són codis prefix.*

PROVA: És conseqüència del fet que un codi sigui òptim o no només depèn de les longituds de les seves paraules i que donat un codi qualsevol amb paraules de longituds determinades existeix un codi prefix amb paraules de les mateixes longituds. □

L'objectiu a continuació és trobar una manera eficient de construir codis òptims.

Lema 3.10. *Sigui X una font de $M \geqslant 2$ símbols amb probabilitats $p_1 \geqslant p_2 \geqslant \cdots \geqslant p_M$. Existeix un codi òptim prefix per a X tal que les dues paraules $\boldsymbol{c}_{M-1}$ i $\boldsymbol{c}_M$ que codifiquen els dos últims símbols difereixen només en el seu últim dígit: són de la forma*

$$\boldsymbol{c}_{M-1} = \boldsymbol{c}\|0, \qquad \boldsymbol{c}_M = \boldsymbol{c}\|1 \qquad \textit{per a una} \quad \boldsymbol{c} \in \{0,1\}^*.$$

PROVA: Es parteix d'un codi òptim prefix $\mathcal{C}_X$ qualsevol per a la font, que existeix gràcies al teorema 3.8. Es veurà que es pot trobar un codi que satisfaci les propietats de l'enunciat simplement reordenant les paraules de $\mathcal{C}_X$.

Observi's que reordenar les paraules codi associades a símbols de la mateixa probabilitat no afecta la longitud mitjana, que tampoc queda afectada si es reordenen paraules de la mateixa longitud. Per tant aquestes reordenacions no afecten el fet que el codi sigui òptim (en el cas de fonts amb probabilitat zero les reordenacions tampoc afectarien a la segona condició).

Primer s'observa que les longituds ℓ_i de les paraules codi que codifiquen símbols de probabilitat diferent tenen longituds ordenades en l'ordre contrari de les probabilitats: si $p_i > p_{i+1}$ aleshores $\ell_i \leqslant \ell_{i+1}$. En efecte, es considera el codi $\mathcal{C}'_X$ format per les mateixes paraules de $\mathcal{C}_X$ però intercanviant les paraules codi $\boldsymbol{c}_i$ i $\boldsymbol{c}_{i+1}$. Com que $\mathcal{C}_X$ és òptim la longitud mitjana d'aquest altre codi de font ha de ser $\geqslant$ i per tant:

$$\widetilde{\ell}(\mathcal{C}'_X) - \widetilde{\ell}(\mathcal{C}_X) = p_i\ell_{i+1} + p_{i+1}\ell_i - p_i\ell_i - p_{i+1}\ell_{i+1} = (p_i - p_{i+1})(\ell_{i+1} - \ell_i) \geqslant 0.$$

Tenint en compte que $p_i - p_{i+1} > 0$ es dedueix que ha de ser $\ell_{i+1} \geqslant \ell_i$.

Ara es reordenen les paraules codi associades als símbols amb la mateixa probabilitat de manera que les longituds quedin en ordre creixent. D'aquesta manera s'ha obtingut un altre codi (òptim i prefix) amb longituds ordenades de la forma $\ell_1 \leqslant \ell_2 \leqslant \cdots \leqslant \ell_M$.

Observi's que fins aquí només s'ha fet servir que el codi és òptim i no s'ha usat la condició de ser prefix.

Sigui $\mathtt{a} \in \{\mathtt{0},\mathtt{1}\}$ la darrera lletra de la paraula $\boldsymbol{c}_M$ i sigui $\boldsymbol{c}_M = \boldsymbol{c}\|\mathtt{a}$ la descomposició com a concatenació del prefix més llarg i l'última lletra amb $\boldsymbol{c} \in \{\mathtt{0},\mathtt{1}\}^*$ Es considera el conjunt $\mathcal{C}'_X = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{M-1}, \boldsymbol{c}\} \subset \{\mathtt{0},\mathtt{1}\}^*$. Aquest conjunt conté M elements: la paraula $\boldsymbol{c}$ no pot ser igual a cap de les $\boldsymbol{c}_i$ perquè en aquest cas $\boldsymbol{c}_i$ seria un prefix de $\boldsymbol{c}_M$, que contradiu la hipòtesi que $\mathcal{C}_X$ és un codi prefix. Si $\mathcal{C}'_X$ fos un conjunt prefix seria un codi i es podria agafar com a codi de font per a X. Es tindria:

- si $p_M \neq 0$ aleshores seria $\widetilde{\ell}(\mathcal{C}'_X) = \widetilde{\ell}(\mathcal{C}_X) - p_M < \widetilde{\ell}(\mathcal{C}_X)$, que contradiu el fet que $\mathcal{C}_X$ tingui longitud mitjana mínima, i
- si $p_M = 0$ la paraula $\boldsymbol{c}$ assignada al símbol $\mathtt{x}_M$ de probabilitat zero seria més curta que $\boldsymbol{c}_M$, i això també contradiu el fet que $\mathcal{C}_X$ sigui òptim.

Per tant $\mathcal{C}'_X$ no pot ser prefix: alguna paraula ha de ser prefix d'una altra. Com que el conjunt $\mathcal{C}_X$ sí que és prefix i l'única possibilitat és que $\boldsymbol{c}$ sigui prefix d'una $\boldsymbol{c}_i$ amb $i < M$. Aquesta paraula només pot ser $\boldsymbol{c}_i = \boldsymbol{c}\|\mathtt{b}$ amb $\mathtt{b} \neq \mathtt{a}$ l'altre bit. En particular les dues paraules $\boldsymbol{c}_i$ i $\boldsymbol{c}_M$ tenen la mateixa longitud, i com que estaven ordenades en ordre creixent de longituds totes les que estan entre una i altra tenen també aquesta longitud. Reordenant les paraules des de la i-èsima endavant es poden posar al final les paraules $\boldsymbol{c}\|\mathtt{0}$ i $\boldsymbol{c}\|\mathtt{1}$. Aquesta reordenació no afecta la longitud mitjana del codi que, per tant, segueix sent òptim i ja satisfà la condició demanada a l'enunciat. □

Teorema 3.11. *Donada una font X de $M \geqslant 2$ símbols amb probabilitats $p_1 \geqslant p_2 \geqslant \ldots \geqslant p_M$ sigui Y la font amb un símbol menys i distribució de probabilitats*

$$p_1, p_2, \ldots, p_{n-2}, p_{M-1} + p_M.$$

Sigui $\mathcal{C}_Y = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{M-1}\}$ un codi òptim prefix per a la font Y. Aleshores el codi

$$\mathcal{C}_X := \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{M-2}, \boldsymbol{c}_{M-1}\|\mathtt{0}, \boldsymbol{c}_{M-1}\|\mathtt{1}\}$$

és un codi òptim prefix per a la font X.

Prova: Primerament s'observa que el conjunt $\mathcal{C}_X$ és un codi prefix, gràcies a què el conjunt $\mathcal{C}_Y$ ho és. Això és immediat, i s'ha vist en general al problema **1.2**.

Per tant l'únic que cal veure ara és que el codi $\mathcal{C}_X$ és òptim per a la font X.

Sigui $\mathcal{C}'_X = \{\boldsymbol{c}'_1, \boldsymbol{c}'_2, \ldots, \boldsymbol{c}'_M\}$ un codi òptim prefix per a X que satisfaci les condicions del lema 3.10: les dues últimes paraules són de la forma $\boldsymbol{c}'_{M-1} = \boldsymbol{c}\|\mathtt{0}$ i $\boldsymbol{c}'_M = \boldsymbol{c}\|\mathtt{1}$.

Es considera el codi de font $\mathcal{C}'_Y = \{\boldsymbol{c}'_1, \ldots, \boldsymbol{c}'_{M-2}, \boldsymbol{c}\}$ per a la font Y, que és clarament un conjunt prefix i per tant un codi. La seves longituds mitjanes satisfan:

$$\begin{aligned}
\widetilde{\ell}(X) = \widetilde{\ell}(\mathcal{C}'_X) &= p_1\ell(\boldsymbol{c}'_1) + \cdots + p_{M-2}\ell(\boldsymbol{c}'_{M-2}) + p_{M-1}\ell(\boldsymbol{c}'_{M-1}) + p_M\ell(\boldsymbol{c}'_M) \\
&= p_1\ell(\boldsymbol{c}'_1) + \cdots + p_{M-2}\ell(\boldsymbol{c}'_{M-2}) + (p_{M-1} + p_M)\ell(\boldsymbol{c}) + (p_{M-1} + p_M) \\
&= \widetilde{\ell}(\mathcal{C}'_Y) + (p_{M-1} + p_M).
\end{aligned}$$

Per altra banda, de manera anàloga, la longitud mitjana del codi $\mathcal{C}_X$ és

$$\begin{aligned}\widetilde{\ell}(\mathcal{C}_X) &= p_1\ell(\boldsymbol{c}_1) + \cdots + p_{M-2}\ell(\boldsymbol{c}_{M-2}) + p_{M-1}\ell(\boldsymbol{c}_{M-1}\|\mathtt{0}) + p_M\ell(\boldsymbol{c}_{M-1}\|\mathtt{1}) \\ &= p_1\ell(\boldsymbol{c}_1) + \cdots + p_{M-2}\ell(\boldsymbol{c}_{M-2}) + (p_{M-1}+p_M)\ell(\boldsymbol{c}_{M-1}) + (p_{M-1}+p_M) \\ &= \widetilde{\ell}(\mathcal{C}_Y) + (p_{M-1}+p_M) = \widetilde{\ell}(Y) + (p_{M-1}+p_M) \\ &\leqslant \widetilde{\ell}(\mathcal{C}'_Y) + (p_{M-1}+p_M) = \widetilde{\ell}(X).\end{aligned}$$

Per minimalitat aquesta desigualtat ha de ser necessàriament una igualtat i per tant el codi $\mathcal{C}_X$ és un codi òptim per a la variable X. □

Aquest resultat suggereix una manera de construir codis òptims de forma recursiva. Els codis que s'obtenen així s'anomenen *codis de Huffman*:

Definició 3.12 (Codi de Huffman binari)**.** *Sigui X una font de $M \geqslant 2$ valors amb probabilitats $p_1 \geqslant p_2 \geqslant \cdots \geqslant p_M$. Un codi de Huffman $\mathcal{C}_X$ per a X és:*

- *Un codi amb paraules $\boldsymbol{c}_1 = \mathtt{0}$ i $\boldsymbol{c}_2 = \mathtt{1}$ si $M = 2$;*
- *Un codi $\boldsymbol{c}_1, \cdots, \boldsymbol{c}_{M-2}, \boldsymbol{c}_{M-1}\|\mathtt{0}, \boldsymbol{c}_{M-1}\|\mathtt{1}$, on $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{M-1}$ és un codi de Huffman per a una font de $M-1$ valors amb probabilitats $p_1, p_2, \ldots, p_{M-1}+p_M$.*

Un codi de Huffman per a una font no té perquè ser únic; en molts casos hi ha diversos símbols amb la mateixa probabilitat mínima i aleshores es pot triar quins dos se sumen per construir la font anterior a partir de la qual es construirà el codi següent. A part d'això, també es consideren codis de Huffman aquells que s'obtenen triant en cada pas l'ordre en què es posen els dígits zero i u. En canviar aquests ordres s'obtenen codis amb paraules de les mateixes longituds.

Corol·lari 3.13. *Els codis de Huffman són codis òptims prefixos.*

Prova: Es demostra per inducció com a conseqüència immediata del teorema 3.11. Per a fonts de dos valors, sigui quina sigui la distribució de probabilitat, és clar que el codi $\{\mathtt{0},\mathtt{1}\}$ és un codi òptim (i l'únic altre codi òptim és $\{\mathtt{1},\mathtt{0}\}$).

Suposant que els codis de Huffman per a fonts de $M-1$ valors són òptims el teorema 3.11 assegura que els codis de Huffman per a fonts de M valors també ho són. □

La construcció en la pràctica dels codis de Huffman per a una font X de M valors amb probabilitats p_i es fa generant un arbre binari ple (tot node pare té exactament dos fills) amb les característiques següents:

- té una única arrel i M fulles;
- els nodes estan etiquetats amb nombres de l'interval $[0,1]$; l'etiqueta de l'arrel és el nombre 1 i les de les fulles són les probabilitats p_i;
- les dues arestes que surten de cada node pare estan etiquetades amb els dígits `0` i `1`.

Una vegada construït l'arbre les paraules codi $\boldsymbol{c}_i \in \{\mathtt{0},\mathtt{1}\}^*$ que codifiquen cada símbol $\mathtt{x}_i$ s'obtenen com la seqüència binària formada per les etiquetes de les arestes que van des del node arrel fins a la fulla etiquetada amb la probabilitat p_i.

L'arbre de Huffman es construeix de la manera següent:

Algorisme 3.14 (Arbre de Huffman)**.** *Sigui X una font de M valors amb probabilitats p_i.*

INICIALITZACIÓ: *S'inicialitza un arbre amb M nodes i cap aresta. Els nodes s'etiqueten amb les probabilitats $p_i = p(\mathtt{x}_i)$.*

REPETIR: *Mentre a l'arbre quedi més d'un node sense pare:*

NOU NODE: *S'agafen dos dels nodes sense pare que tinguin les etiquetes més petites i s'afegeix un nou node com a pare d'aquests dos. El nou node pare s'etiqueta amb la suma de les etiquetes dels dos fills. Les arestes que van del nou node pare als dos fills s'etiqueten una amb un* `0` *i l'altra amb un* `1`*.*

Com que en cada pas el nombre de nodes sense pare disminueix en una unitat l'algorisme acaba quan només en queda un, que serà l'únic node arrel de l'arbre i portarà com a etiqueta la suma de totes les etiquetes inicials, que és 1.

Codi de Huffman q-ari. La construcció de codis de Huffman que s'ha explicat en aquesta secció és específica per a codificació binària. En el cas d'alfabets amb més de dos símbols es poden construir codis prefixos òptims q-aris de manera anàloga, amb una diferència que és que es demana sempre que l'alfabet de la font tingui un nombre M de lletres que satisfaci la congruència $M = |\mathcal{X}| \equiv 1 \pmod{q-1}$. Això s'aconsegueix afegint valors a la variable X amb probabilitat zero fins que es compleix la tenir aquesta congruència. El motiu d'aquesta condició és assegurar que al final l'arrel serà el pare d'exactament q fills.

Per a una font com aquesta la construcció de codis de Huffman q-aris és completament anàloga al cas binari a partir d'un arbre q-ari ple que es genera amb l'algorisme següent:

INICIALITZACIÓ: S'inicialitza un arbre amb M nodes i cap aresta. Els nodes s'etiqueten amb les probabilitats $p_i = p(\mathtt{x}_i)$.

REPETIR: Mentre a l'arbre quedi més d'un node sense pare:

NOU NODE: S'agafen q dels nodes sense pare que tinguin les etiquetes més petites i s'afegeix un nou node com a pare d'aquests dos. El nou node pare s'etiqueta amb la suma de les etiquetes dels q fills. Les arestes que van del nou node pare als fills s'etiqueten amb les diferents lletres de l'alfabet q-ari.

En cas que s'hagin hagut d'afegir valors de probabilitat zero les paraules que els haurien de codificar, que s'han obtingut a partir de l'arbre, es descarten. Són algunes de les més llargues del codi construït.

La demostració que d'aquesta manera s'obtenen codis òptims és anàloga a la del cas binari però més elaborada. En els problemes **3.22** i **3.21** es donen indicacions per fer aquesta demostració de manera rigorosa; són les versions q-àries del lema 3.10 i el teorema 3.11 que s'han usat per demostrar el cas binari.

Problemes

3.1. Sigui X una font que pot tenir probabilitats nul·les. Sigui $\mathcal{C}_X$ un codi de font per a X amb paraules de longituds ℓ_i que tingui longitud mitjana mínima: $\widetilde{\ell}(\mathcal{C}_X) = \widetilde{\ell}(X)$. Sigui $\mathcal{C}'_X$ un altre codi de font per a X amb longituds $\ell'_i = \ell_i$ excepte $\ell'_j < \ell_j$ per a un índex j. Demostreu que aleshores ha de ser $p_j = 0$.

3.2. Trobeu un codi de Huffman binari per a una font de cinc lletres amb distribució de probabilitats $\boldsymbol{p} = (\frac{1}{3}, \frac{1}{5}, \frac{1}{5}, \frac{2}{15}, \frac{2}{15})$. Comproveu que aquest codi també és òptim per a la distribució uniforme sobre les cinc lletres.

3.3. Per a la distribució de probabilitats $\boldsymbol{p} = (\frac{1}{3}, \frac{1}{3}, \frac{1}{4}, \frac{1}{12})$,

1. trobeu dos codis de Huffman amb longituds de paraules diferents i comproveu que les seves longituds mitjanes coincideixen;
2. trobeu un codi de Shannon i calculeu la seva longitud mitjana;
3. compareu les longituds mitjanes anteriors amb l'entropia.

3.4. Construïu un codi ternari per a una variable que prengui quatre valors diferents fent un arbre ternari de manera anàloga a l'algorisme de Huffman binari. Comproveu que aquest codi no és òptim però sí que s'obté un codi òptim si en el primer pas s'uneixen només dos nodes.

Interpreteu això com afegir un valor a la variable amb probabilitat zero.

3.5. Trobeu codis de Huffman binaris i ternaris per a la distribució de probabilitat

$$\boldsymbol{p} = (0.49, 0.26, 0.12, 0.04, 0.04, 0.03, 0.02).$$

Calculeu les seves longituds mitjanes i compareu-les amb l'entropia.

3.6. Trobeu codis de Huffman binaris i ternaris per a una variable aleatòria amb distribució de probabilitats $\boldsymbol{p} = \frac{1}{21}(1, 2, 3, 4, 5, 6)$, calculeu les longituds mitjanes corresponents i compareu amb l'entropia.

3.7. Digueu quins dels codis següents poden ser codis òptims per a alguna font:

1. $\{\mathtt{0}, \mathtt{10}, \mathtt{11}\}$;
2. $\{\mathtt{00}, \mathtt{01}, \mathtt{10}, \mathtt{110}\}$;
3. $\{\mathtt{01}, \mathtt{10}\}$.

3.8. Es considera un procés estocàstic $\boldsymbol{X} = X_1, X_2, \ldots$ que emet símbols d'un alfabet de tres elements $\mathcal{X} = \{\mathtt{s}_1, \mathtt{s}_2, \mathtt{s}_3\}$. Se suposa que és una cadena de Markov (d'ordre 1) estacionària amb matriu de transició

$$\boldsymbol{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}.$$

Es construeixen tres codis prefixos binaris $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3$ amb els quals es codifiquen les seqüències emeses per la cadena segons l'esquema següent: el símbol emès per la primera variable X_1 es codifica amb algun dels tres codis $\mathcal{C}_i$; un cop codificat fins al símbol n-èsim s'agafa el seu valor i si $X_n = \mathtt{s}_i$, aleshores el símbol emès per la variable següent X_{n+1} es codifica usant el codi $\mathcal{C}_i$.

1. Calculeu els codis $\mathcal{C}_i$ que minimitzin la longitud d'aquesta codificació.
2. Calculeu la longitud mitjana per símbol d'aquesta codificació.
3. Compareu el resultat anterior amb la taxa d'entropia del procés.

3.9. Demostreu que tot codi de font binari òptim és complet però que això no es compleix per a codis q-aris.

3.2 Teorema de codificació de font

Referència: Brunat-Ventura [3, Capítol 4]

El resultat més important de la teoria de la informació pel que fa a codificació de font és el *teorema de codificació de font*, que dona fites per a la longitud mitjana dels codis òptims en termes de l'entropia de la variable aleatòria corresponent. En el cas més simple d'una font sense memòria el teorema assegura que:

Teorema 3.15 (Teorema de codificació de font sense memòria)**.** *Per a tota font sense memòria X es tenen desigualtats*

- $H(X) \leqslant \widetilde{\ell}(\mathcal{C}_X)$ *per a tot codi de font* $\mathcal{C}_X$*;*
- $\widetilde{\ell}(\mathcal{C}_X) \leqslant H(X) + 1$ *per a tot codi de font òptim* $\mathcal{C}_X$*.*

Per tant la longitud $\widetilde{\ell}(X)$ dels codis de font òptims satisfà la doble desigualtat:

$$H(X) \leqslant \widetilde{\ell}(X) \leqslant H(X) + 1.$$

Per a fonts no constants la segona desigualtat és estricta.

Prova: Sigui $\boldsymbol{p} = (p_1, \ldots, p_M)$ la distribució de probabilitats de la variable X.

Sigui $\mathcal{C}_X$ un codi de font qualsevol per a X, amb paraules de longituds ℓ_i. Per la desigualtat de Kraft, $s := \sum 2^{-\ell_i} \leqslant 1$. S'agafa la distribució de probabilitat formada pels nombres $q_i = \frac{1}{s} 2^{-\ell_i}$, que tenen suma 1. La desigualtat de Gibbs dona:

$$\begin{aligned} H(X) = \sum_{i=1}^{M} p_i \log \frac{1}{p_i} &\leqslant \sum_{i=1}^{M} p_i \log \frac{1}{q_i} = \sum_{i=1}^{M} p_i \log s 2^{\ell_i} \\ &= \sum_{i=1}^{M} p_i \log s + \sum_{i=1}^{M} p_i \log 2^{-\ell_i} = \log s \sum_{i=1}^{M} p_i + \sum_{i=1}^{M} p_i \ell_i = \log s + \widetilde{\ell}(\mathcal{C}_X) \leqslant \widetilde{\ell}(\mathcal{C}_X) \end{aligned}$$

ja que $s \leqslant 1 \Rightarrow \log s \leqslant 0$.

Fent servir la versió més general de la desigualtat de Gibbs, que val per a nombres $q_i \geqslant 0$ amb $\sum q_i \leqslant 1$ aleshores s'obté el mateix més directament, agafant simplement $q_i = 2^{-\ell_i}$:

$$H(X) = \sum p_i \log \frac{1}{p_i} \leqslant \sum p_i \log \frac{1}{2^{-\ell_i}} \sum p_i \ell_i = \widetilde{\ell}(\mathcal{C}_X).$$

Per demostrar la segona desigualtat n'hi ha prou a construir un codi de font $\mathcal{C}$ que tingui longitud mitjana $\widetilde{\ell}(\mathcal{C}) \leqslant H(X)+1$, ja que tot codi òptim $\mathcal{C}_X$ tindrà longitud mitjana inferior i també satisfarà la desigualtat: $\widetilde{\ell}(\mathcal{C}_X) \leqslant \widetilde{\ell}(\mathcal{C}) \leqslant H(X)+1$.

Se suposa primer no té probabilitats zero i és no constant (tampoc té probabilitat 1). S'agafa el codi de Shannon de l'exemple 3.3, format per paraules de longitud $\ell_i = \lceil -\log p_i \rceil$, la part entera per excés de $-\log p_i$. Aquest enter satisfà la desigualtat estricta $\ell_i < -\log p_i + 1$. Multiplicant per p_i i sumant sobre tots els índexs es té

$$\widetilde{\ell}(\mathcal{C}) = \sum_{i=1}^{M} p_i \ell_i < \sum_{i=1}^{M} p_i\big(-\log p_i + 1\big) = H(X) + \sum_{i=1}^{M} p_i = H(X)+1.$$

En aquest cas, per tant, es té una desigualtat estricta.

Si la font és constant sense probabilitats zero només agafa un valor amb probabilitat 1, que s'ha de codificar amb una paraula de longitud 1. Té entropia zero i el codi té longitud mitjana 1, i per tant es compleix la desigualtat $\widetilde{\ell}(\mathcal{C}_X) = 1 = H(X)+1$.

Finalment, el cas de les fonts amb alguna probabilitat igual a zero es pot reduir a l'anterior de la manera següent: sigui X una font de M' símbols amb $M < M'$ probabilitats no nul·les $p_1 \geqslant p_2 \geqslant \cdots \geqslant p_M$ i les altres zero. Sigui $p \in (0,1)$ i sigui $q = 1-p$. Es considera la font Y de $M+1$ símbols i probabilitats $p_1, p_2, \cdots, p_{M-1}, pp_M, qp_M$, que té com a probabilitats més petites les dues últimes. Aquesta font té entropia $H(Y) = H(X) + p_M H(p)$. Sigui $\mathcal{C}_Y$ un codi òptim per a la font Y amb les dues últimes paraules de la forma $\boldsymbol{c}\|\mathtt{0}$, $\boldsymbol{c}\|\mathtt{1}$. Existeix gràcies al lema 3.10, o simplement es pot agafar un codi de Huffman, que satisfarà aquesta condició. Com que Y té totes les probabilitats no nul·les se satisfà la desigualtat $\widetilde{\ell}(\mathcal{C}_Y) \leqslant H(Y)+1$ (de fet, en realitat se satisfà la desigualtat estricta). Aleshores el conjunt prefix $\mathcal{C}_X$ format per les primeres $M-1$ paraules de $\mathcal{C}_Y$, la paraula $\boldsymbol{c}\|\mathtt{0}$ i tantes paraules amb prefix $\boldsymbol{c}\|\mathtt{1}$ com valors amb probabilitat zero es pot agafar com a codi de font per a X i es té

$$\begin{aligned}\widetilde{\ell}(\mathcal{C}_X) = \sum_{i=1}^{M-1} p_i\ell_i + p_M \ell(\boldsymbol{c}\|\mathtt{0}) = \sum_{i=1}^{M-1} p_i\ell_i + pp_M\ell(\boldsymbol{c}\|\mathtt{0}) + qp_M\ell(\boldsymbol{c}\|\mathtt{1}) \\ = \widetilde{\ell}(\mathcal{C}_Y) \leqslant H(Y)+1 = H(X)+1+p_M H(p).\end{aligned}$$

Com que quan $p \to 0$ (o també quan $p \to 1$) l'entropia $H(p)$ tendeix a zero i la desigualtat anterior es compleix per a tot $p \in (0,1)$ es dedueix que ha de ser $\widetilde{\ell}(\mathcal{C}_X) \leqslant H(X)+1$.

En el cas que X sigui no constant ha de ser $M \geqslant 2$ i per tant $p_1 \in (0,1)$. Aleshores es té $\widetilde{\ell}(\mathcal{C}_Y) \leqslant H(Y)+1-\alpha$ amb $\alpha = p_1(-\log p_1 + 1 - \ell_1) > 0$ (de fet, la desigualtat és estricta ja que hi ha altres probabilitats en la mateixa situació, però no cal per a l'argument). El mateix argument d'abans dona la desigualtat $\widetilde{\ell}(\mathcal{C}_X) \leqslant H(X)+1-\alpha+pH(p_M)$. Fent tendir p a zero o a u es dedueix que $\widetilde{\ell}(\mathcal{C}_X) \leqslant H(X)+1-\alpha < H(X)+1$. □

El teorema s'ha vist per a codificacions binàries de fonts. Quan es fan codificacions amb alfabets q-aris es té un resultat totalment anàleg, que es demostra exactament igual, amb l'única diferència que s'ha de canviar l'entropia en bits habitual per l'entropia amb els logaritmes agafats en base q:

$$H_q(X) = -\sum_{\mathtt{x}\in\mathcal{X}} p(\mathtt{x}) \log_q \big(p(\mathtt{x})\big) = \frac{H(X)}{\log q}.$$

Aleshores la desigualtat per codis de font q-aris es converteix en

$$H_q(X) \leqslant \widetilde{\ell}(X) < H_q(X) + 1.$$

Extensions d'una font: codificació per blocs. Un codi de font $\mathcal{C}_X$ codifica els símbols emesos per la font X d'un en un. Una altra manera de codificar la informació que emet la font és descompondre la seqüència generada en blocs formats per diversos símbols i considerar aquests blocs com l'alfabet. És a dir, es fixa un $n > 1$ i es considera la variable aleatòria vectorial $X^n = (X_1, X_2, \ldots, X_n)$ amb variables independents $X_i \sim X$ que pren com a valors paraules $\mathtt{x}^n \in \mathcal{X}^n$ de n lletres. Com que les X_i són independents, ja que es considera un font sense memòria, la distribució de probabilitat d'aquesta variable X^n s'obté a partir de la de la variable X multiplicant les probabilitats de les lletres que formen la paraula $\mathtt{x}^n$.

La variable X^n es pot considerar com una nova font sense memòria. Se li diu *extensió n-èsima* de la font X. Aplicant el teorema 3.15 a aquesta variable s'obté la desigualtat

$$H(X^n) \leqslant \widetilde{\ell}\big(X^n\big) < H(X^n) + 1.$$

Com que les components de X^n són independents es té $H(X^n) = nH(X)$. Per altra banda, en codificar els valors de la font X^n, que són elements de $\mathcal{X}^n$, cada paraula del codi $\mathrm{enc}(\mathtt{x}^n)$ serveix per codificar n símbols de $\mathcal{X}$ alhora, de manera que el nombre de bits usat per a cada símbol s'ha de dividir per n. Així es té el

Corol·lari 3.16. *Si $\mathcal{C}_{X^n}$ és un codi òptim per a l'extensió X^n d'una font X,*

$$H(X) \leqslant \frac{1}{n}\widetilde{\ell}\big(\mathcal{C}_{X^n}\big) < H(X) + \frac{1}{n}.$$

Aquest resultat indica com es pot millorar l'eficiència de la codificació de font: si quan es codifiquen els símbols emesos per la font d'un en un amb un codi òptim $\mathcal{C}_X$ la longitud mitjana està massa lluny de l'entropia, una alternativa és codificar les seqüències de símbols emesos per la font agrupats en blocs de n símbols amb un codi òptim $\mathcal{C}_{X^n}$ per a la variable X^n. D'aquesta manera la longitud mitjana per símbol codificat del codi, que és igual a $\frac{1}{n}\widetilde{\ell}(\mathcal{C}_{X^n})$, es pot aconseguir que estigui tan a prop de l'entropia de X com es vulgui, amb la condició d'augmentar n, el nombre de símbols que es codifiquen alhora. Naturalment, la construcció de codis òptims per a les variables X^n quan n augmenta és molt més complexa que per a la variable X, ja que aquesta pren $|\mathcal{X}|$ valors i aquella en pren $|\mathcal{X}|^n$.

Exemple: extensions d'un font binària. Es considera una font X que emet símbols de l'alfabet binari $\mathcal{X} = \{\mathtt{0}, \mathtt{1}\}$ amb probabilitats $p(\mathtt{0}) = 0.9$ i $p(\mathtt{1}) = 0.1$. La seva entropia és

$$H(X) = -0.9\log(0.9) - 0.1\log(0.1) \approx 0.469 \text{ bits.}$$

Per tant cada símbol que emet X proporciona una informació de menys de mig bit.

Un codi de Huffman $\mathcal{C}$ per a aquesta font és, simplement, el que envia $\mathtt{0} \mapsto \mathtt{0}, \mathtt{1} \mapsto \mathtt{1}$. La seva longitud mitjana és $\widetilde{\ell}(\mathcal{C}) = 1$, que entra dins de la desigualtat del teorema de codificació de font:

$$0.469 \approx H(X) \quad \leqslant \quad 1 = \ell(\mathcal{C}) \quad < \quad 1.469 \approx H(X) + 1.$$

Aquesta codificació no comprimeix gens la informació però, tal com s'ha indicat, agafant extensions de la font es pot arribar a codificar la informació emesa usant un nombre de bits per símbol tan proper a l'entropia com es vulgui.

Es considera la 2-extensió X^2, que emet símbols del conjunt $\mathcal{X}^2 = \{\mathtt{00}, \mathtt{01}, \mathtt{10}, \mathtt{11}\}$ amb probabilitats

$$p(\mathtt{00}) = 0.81, \quad p(\mathtt{01}) = p(\mathtt{10}) = 0.09, \quad p(\mathtt{11}) = 0.01$$

En fer l'arbre i construir un codi de Huffman s'obté el codi prefix $\mathcal{C}_{X^2}$ següent:

$\mathtt{x}^2$	$\boldsymbol{c}$
00	0
01	10
10	110
11	111

que té longitud mitjana igual a

$$\begin{aligned}\widetilde{\ell}\big(\mathcal{C}_{X^2}\big) &= p(\mathtt{00})\ell(\mathtt{0}) + p(\mathtt{01})\ell(\mathtt{10}) + p(\mathtt{10})\ell(\mathtt{110}) + p(\mathtt{11})\ell(\mathtt{111}) \\ &= 0.81 \cdot 1 + 0.09 \cdot 2 + 0.09 \cdot 3 + 0.01 \cdot 3 = 1.29\end{aligned}$$

Com que cada paraula d'aquest codi codifica dos símbols binaris de la font X, per codificar cada símbol de X es fan servir, en mitjana, 0.645 símbols binaris.

Es pot també considerar la 3-extensió de la font X^3, que genera blocs de 3 símbols. Les probabilitats, el codi de Huffman $\mathcal{C}_{X^3}$ i la seva longitud mitjana són:

$\mathtt{x}^3$	$p(\mathtt{x}^3)$	$\boldsymbol{c}$
000	0.729	0
001	0.081	100
010	0.081	101
011	0.009	11100
100	0.081	110
101	0.009	11101
110	0.009	11110
111	0.001	11111

$$\widetilde{\ell}\big(\mathcal{C}_{X^3}\big) = 0.729 \cdot 1 + 0.081 \cdot (3+3+3) + 0.009 \cdot (5+5+5) + 0.001 \cdot 5 = 1.598.$$

Per codificar cada símbol de X aquest codi fan servir, en mitjana, 0.5327 símbols binaris.

Usant un codi de Huffman per codificar la quarta extensió X^4 calen 0.49255 bits per símbol, etc. Així, fent extensions de la font cada vegada més llargues i calculant els seus codis de Huffman les longituds mitjanes per símbol de X codificat es van apropant a l'entropia $H(X) \approx 0.469$ tant com es vulgui.

Codificació de fonts amb memòria. El teorema 3.15 té conseqüències també per a la codificació de fonts amb memòria. Es considera una font qualsevol donada per un procés estocàstic $\boldsymbol{X}$: una successió de variables aleatòries $X_1, X_2, X_3, \ldots$ que prenen valors en un mateix conjunt $\mathcal{X}$.

Per a cada $n \geqslant 1$ es denota $X^n = (X_1, \ldots, X_n)$ el vector aleatori format per les n primeres variables. Se suposa que la font té taxa d'entropia $H(\boldsymbol{X}) = \lim_{n\to\infty} \frac{1}{n} H(X^n)$. Per exemple, si el procés és estacionari aleshores la seva taxa d'entropia està definida. Del teorema de codificació de font es dedueix el

Corol·lari 3.17. *Si $\mathcal{C}_{X^n}$ és un codi de font òptim per a la variable X^n per a cada $n \geqslant 1$,*

$$\lim_{n\to\infty} \frac{1}{n} \widetilde{\ell}\big(\mathcal{C}_{X^n}\big) = H(\boldsymbol{X}).$$

Prova: El teorema assegura que, per a cada $n \geqslant 1$, es té

$$H(X^n) \leqslant \widetilde{\ell}(\mathcal{C}_{X^n}) < H(X^n) + 1.$$

Dividint per n i fent tendir n a infinit els dos extrems de la igualtat tendeixen a l'entropia de la font $\boldsymbol{X}$. □

Així, el Teorema de Codificació de Font torna a dir en aquest cas el mateix: la informació emesa per una font qualsevol $\boldsymbol{X}$ que tingui entropia $H(\boldsymbol{X})$ es pot codificar en binari usant essencialment $H(\boldsymbol{X})$ bits per símbol emès. Això si, és imprescindible fer la codificació directament sobre tota la seqüència que correspon a la informació que es vol codificar. Tot i que com a resultat teòric és interessant, en aquesta situació no té sentit portar-ho a la pràctica ja que la construcció de codis de Huffman per a extensions cada vegada més llargues de la font acaba sent inviable.

Joc de les preguntes. Els codis de font òptims tenen relació amb el *joc de les preguntes* amb resposta binària si/no: es tracta d'endevinar el resultat d'una variable aleatòria X fent preguntes binàries a algú que coneix aquest resultat. L'objectiu és aconseguir saber el resultat de la variable fent el mínim nombre de preguntes possible.

Quan es juga el joc moltes vegades, per a diversos resultats de la variable, el nombre de preguntes que cal fer és, en mitjana, la longitud mitjana dels codis de font òptims $\mathcal{C}_X$. L'arbre de Huffman permet dissenyar el conjunt de preguntes a fer: es van seguint els nodes de l'arbre començant per l'arrel i, en cada pas, es pregunta si el resultat de la variable pertany al subconjunt de $\mathcal{X}$ format per fulles de l'arbre a les quals s'arriba seguint una de les dues arestes que surten del node.

Problemes

3.10. El teorema de codificació de font assegura que tot codi òptim per a una variable X té longitud mitjana $\widetilde{\ell}(\mathcal{C}_X)$ fitada de la forma $H(X) \leqslant \widetilde{\ell}(\mathcal{C}_X) < H(X)+1$. Doneu exemples de variables i codis òptims en els dos extrems: uns amb $\widetilde{\ell}(\mathcal{C}_X) = H(X)$ i altres amb $\widetilde{\ell}(\mathcal{C}_X) = H(X) + 1 - \epsilon$ per a un nombre $\epsilon > 0$ donat tan petit com es vulgui.

3.11. Sigui X una font de M símbols amb distribució de probabilitats $\boldsymbol{p} = (p_1, \ldots, p_M)$ que són totes potències (negatives) de 2: $p_i = 2^{-n_i}$ per a tot i amb $n_i \in \mathbb{N}$. Sigui $\mathcal{C}_X$ un codi de Huffman per a la variable X, amb paraules de longituds ℓ_i. Demostreu que

1. almenys dues de les probabilitats més petites són iguals;
2. $\ell_i = n_i$;
3. $\widetilde{\ell}(\mathcal{C}_X) = H(X)$.

3.12. Sigui $\mathcal{C}_X$ un codi de font binari per a la variable X format per n paraules de longituds ℓ_i. Sigui $\Sigma = \sum_{i=1}^{n} 2^{-\ell_i}$. Demostreu que

$$\widetilde{\ell}(\mathcal{C}_X) = H(X) + D(\boldsymbol{p}\|\boldsymbol{q}) - \log \Sigma,$$

on $\boldsymbol{q}$ és la distribució de probabilitat amb valors $\frac{1}{\Sigma} 2^{-\ell_i}$.

Deduïu que els codis òptims $\mathcal{C}_X$ per a la font X assoleixen la fita inferior $\widetilde{\ell}(\mathcal{C}_X) = H(X)$ del teorema de codificació de font si, i només i, les probabilitats amb què la font emet els símbols són totes potències de 2.

3.13. Construïu un codi de Huffman binari per a la distribució de probabilitats

$$\boldsymbol{p} = \frac{1}{10}(3, 3, 2, 1, 1).$$

Calculeu la seva longitud mitjana i compareu-la amb l'entropia.

Trobeu una distribució de probabilitats $\boldsymbol{q} = (q_1, q_2, q_3, q_4, q_5)$ tal que el codi construït abans tingui longitud mitjana igual a l'entropia d'aquesta distribució.

3.14. Construïu codis de Huffman i de Shannon binaris per a la distribució de probabilitats $\boldsymbol{p} = (0.6, 0.3, 0.1)$ i compareu les seves longituds mitjanes amb l'entropia. Digueu per a quins enters $q \geqslant 2$ el codi de Shannon q-ari és òptim.

3.15. Considereu una variable que pren cinc valors amb distribució de probabilitats $\boldsymbol{p} = \frac{1}{32}(12, 6, 5, 5, 4)$. Calculeu codis de Shannon, *Shannon-Fano* i Huffman binaris per a aquesta variable. Calculeu les longituds mitjanes i compareu amb l'entropia.

3.16. Considereu una variable que pren 19 valors amb distribució de probabilitats

$$\boldsymbol{p} = (0.4, 0.26, 0.02, 0.02, 0.02, \ldots, 0.02).$$

Trobeu codis de Shannon, Shannon-Fano i Huffman binaris per a aquesta variable, calculeu les seves longituds mitjanes i compareu amb l'entropia.

Compareu les mides de les paraules en tots tres codis.

3.17. Sigui X una font binària amb distribució de Bernoulli $\mathrm{Ber}(p)$ amb probabilitat $p = \frac{1}{10}$. Trobeu l'extensió mínima de la font que es pot codificar amb un codi que tingui longitud mitjana per símbol $\leqslant \frac{1}{2}$.

3.18. Sigui X una font binària amb distribució de Bernoulli $\mathrm{Ber}(p)$ amb probabilitat $p = \frac{1}{10}$. Es codifica una cadena binària emesa per aquesta font un codi de Hufmman binari per a una extensió. Calculeu la distribució de probabilitat dels dígits en la cadena codificada per a les extensions primera, segona, tercera i quarta.

Interpreteu els valors obtinguts en termes de la taxa d'entropia del procés estocàstic que genera els dígits binaris de la seqüència codificada.

3.19. Sigui $\mathcal{C}$ un codi binari format per n paraules de longituds $\ell_1, \ldots, \ell_n$. Demostreu que

$$\ell_1 + \ell_2 + \cdots + \ell_n \geqslant n \log n.$$

INDICACIÓ: Useu aquest codi per codificar una font d'informació apropiada.

3.20. Sigui $\mathcal{C}$ un codi de Huffman binari format per n paraules de longituds $\ell_1, \ldots, \ell_n$. Demostreu que

$$\ell_1 + \ell_2 + \cdots + \ell_n \leqslant \frac{n^2 + n - 2}{2}.$$

INDICACIÓ: La paraula més llarga ha de tenir longitud $\ell \leqslant n - 1$.

3.3 Compressió

El motiu pel qual la informació es pot comprimir és la presència de *redundància*. La idea bàsica prové de l'article *"A mathematical theory of communication"* [27] publicat el 1948 per Claude Shannon, on s'introdueix la Teoria de la Informació i es defineix el concepte fonamental d'*entropia*. Donada una seqüència de símbols generada per una font l'entropia de la font és una mesura de la quantitat d'informació continguda en cada símbol d'aquesta seqüència. Dona un límit inferior de la màxima compressió possible. Els mètodes de compressió procuren fer una codificació binària de la seqüència amb nombre de bits tan a prop com es pugui de l'entropia.

La *compressió de dades* consisteix en canviar el format de les dades perquè ocupin menys espai. Naturalment, això s'ha de fer de manera reversible: a partir de les dades comprimides s'han de poder recuperar les dades originals (exactament o aproximada). Les tècniques de compressió de dades més habituals es classifiquen de la manera següent:

Sense pèrdua. Quan en descomprimir es recupera la informació original *exactament*. És imprescindible fer servir aquest tipus de tècnica per comprimir text, software, bases de dades, i en general sempre que l'original s'hagi de recuperar sense permetre cap canvi. Els mètodes pertanyen a dues grans famílies:

Mètodes probabilistes. Exploten l'estructura del missatge des del punt de vista de les probabilitats: freqüències de lletres, digrames, trigrames, freqüències relatives,

etc. Es basen en el concepte la teoria de la informació i el teorema de codificació de font, estudiats a les seccions 2 i 3.2, respectivament. Els més coneguts són els codis de Huffman vistos a la secció 3.1 i la codificació aritmètica que es descriurà a la secció 3.4.

Els codis de Huffman són una de les tècniques de compressió més usades en la pràctica, moltes vegades com una de les parts constituents d'un procediment més complex.

Mètodes de diccionari. Es coneixen també com a mètodes LZ en referència al nom dels inventors: Lempel i Ziv. En processar la seqüència que es vol comprimir es va creant un diccionari amb les paraules noves que es van trobant; quan una paraula que ja és al diccionari es torna a trobar es codifica amb una referència a la posició que ocupa en el diccionari. Es descriuran a la secció 3.5

Observi's que en el primer cas és essencial conèixer les propietats estocàstiques de la font que ha generat la seqüència a comprimir i fer-les servir en l'algorisme de compressió. En canvi, en el segon no cal, tot i que el resultat de la compressió dependrà d'aquestes propietats; és per això que aquests mètodes de vegades s'anomenen compressors universals, en el sentit que no depenen de la font que ha generat la seqüència.

Amb pèrdua. Anomenada també codificació perceptual. La informació es recupera només aproximadament: en el procés de codificació-descodificació es produeix una pèrdua. Es fan servir sobretot per comprimir audio, imatge i vídeo. En aquest tipus d'informació es tenen en compte les característiques fisiològiques de la vista i la oïda humanes per tal que els canvis introduïts en els processos de compressió-descompressió siguin gairebé imperceptibles. La tècnica principal per a la compressió amb pèrdua es basa en la teoria de Fourier: es transformen les dades entre els dominis *temporal* (so) o espacial (imatge) i el *domini freqüencial*. Això es fa amb transformades discretes: típicament la *transformada de cosinus* i *transformades wavelet*.

Aquí no es parlarà d'aquestes tècniques de *compressió amb pèrdua*.

Model per als mètodes probabilistes. La compressió persegueix codificar una seqüència de lletres d'un alfabet $\mathcal{X}$ amb una cadena binària el més curta possible. Els mètodes probabilistes assumeixen que la seqüència ha estat emesa per una font d'informació, de manera que les seves lletres han estat generades d'acord amb determinades probabilitats, de manera independent o no segons que la font sigui sense o amb memòria.

La font és un procés estocàstic $\boldsymbol{X} = X_1, X_2, X_3, \ldots$ que s'agafa com a model matemàtic per fer la compressió. Naturalment, per tal que la cosa funcioni, les lletres de la seqüència a comprimir han de reflectir bé les propietats estocàstiques d'aquest procés, i tan l'emissor (compressor) com el receptor (descompressor) han de saber quin és aquest model.

En alguns casos el model està clar. Per exemple, si es vol dissenyar un mètode de compressió/descompressió que s'aplicarà només a textos en una llengua (català, castellà, anglès, etc.) es pot fixar un model que tingui en compte les particularitats estocàstiques de la llengua, que pot ser molt simple tenint en compte només freqüències de lletres, o més sofisticat, tenint en compte freqüències de digrames, trigrames, freqüències relatives, etc. Un cop acordat el

model (la font) tant compressor com descompressor el faran servir. Si es vol aplicar aquest mètode a una seqüència de lletres que correspon a un text en una altra llengua o un altre tipus d'informació aleshores no funcionarà bé.

En la majoria de situacions, però, només es té una seqüència $\boldsymbol{x} \in \mathcal{X}^*$ que no és una mostra d'un procés estocàstic conegut. Tot i que sempre es pot usar un mètode de diccionari, si es vol aplicar a aquesta seqüència un mètode probabilista s'ha de fer una hipòtesi sobre quin podria ser un model adequat: la distribució de probabilitats de la font s'infereixen a partir de la seqüència $\boldsymbol{x}$ calculat la freqüència de les lletres (model sense memòria), digrames (model de Markov de primer ordre), trigrames (Markov de segon ordre), etc. en aquesta seqüència. Hi ha dues maneres de fer això, que es coneixen amb els noms de

Codificació estàtica. La font és la mateixa durant tot el procés de compressió o descompressió: abans de comprimir es calculen les freqüències a partir de la seqüència $\boldsymbol{x}$, es decideix quin és el model: les probabilitats del procés estocàstic $\boldsymbol{X}$, i es codifica d'acord amb aquest model.

El descodificador ha de rebre també la informació sobre el model $\boldsymbol{X}$ usat per poder fer la descompressió.

Codificació adaptativa. La font es va modificant sobre la marxa durant la compressió i la descompressió, adaptant-la a les dades a mesura que es van processant, tant en la compressió com en la descompressió.

Es parteix d'un model inicial arbitrari (per exemple una distribució uniforme) i les probabilitats del procés es van recalculant cada vegada que es processa un símbol de la seqüència (o un nombre prefixat de símbols).

En aquest cas no és necessari donar-li al descompressor la informació sobre el model, ja que ell mateix l'anirà creant i modificant exactament com ho ha fet el compressor, perquè en cada moment només depèn del tros de text que ja s'ha processat.

Aquestes dues opcions es coneixen també amb el nom de *codificació amb dues passades* i *codificació amb una passada*, respectivament. Això fa referència a què en el primer cas la seqüència $\boldsymbol{x}$ s'ha de processar completa dues vegades: la primera serveix per calcular les probabilitats de la font $\boldsymbol{X}$ i la segona per fer la codificació a partir d'aquestes probabilitats; en el segon cas només cal processar la seqüència $\boldsymbol{x}$ una única vegada.

En situacions en què un flux de lletres de $\mathcal{X}$ s'ha d'anar codificant sobre la marxa, transformant-lo en un flux binari que correspon a la versió comprimida a mesura que es van processant les lletres, de manera que mai es disposa de la seqüència $\boldsymbol{x}$ completa, només es pot fer aplicar la compressió adaptativa.

Exemple: llenguatge natural. Quan es vol justificar la utilitat dels codis de Huffman se sol posar com a exemple el dels llenguatges naturals: en totes les llengües les lletres apareixen en el text amb freqüències diferents. Per codificar un text es fa servir un codi de Huffman que tingui en compte les probabilitats de les lletres: les més freqüents es codifiquen amb cadenes binaries curtes i les menys freqüents amb cadenes més llargues.

Les acadèmies encarregades de vetllar per cada llengua acostumen a compilar llistes de texts que són representatius de la llengua, anomenats *corpora*. A partir d'aquestes llistes es poden obtenir les probabilitats de cada lletra en una llengua donada. Alternativament, es pot agafar un text concret (per exemple el text que es vol comprimit) per crear un model estocàstic de la llengua. Aquest model serà el que comprimeixi millor el text en qüestió, tot i que potser en altres textos la compressió no sigui prou bona.

Les freqüències de lletres en idiomes diferents es poden trobar en molts llocs, per exemple a la *wiki*, tot i que, com s'ha explicat, aquestes freqüències depenen del model agafat per modelitzar la llengua, i les taules que es donen en diferents llocs varien lleugerament.

Per exemple, s'agafa com a model de l'anglès la novel·la Moby Dick, de *Herman Melville*, que conté 1 168 421 caràcters de l'alfabet anglès de 26 lletres més l'espai. La taula següent dona el nombre de vegades que apareix cada lletra i un codi de Huffman per a aquesta font:

LLETRA	#	CODI	LLETRA	#	CODI	LLETRA	#	CODI
espai	216026	111	i	65471	0110	r	52173	0001
a	77948	1010	j	1084	1011111010	s	64256	0101
b	16886	100101	k	8058	10111111	t	88037	1101
c	22523	110011	l	42804	11000	u	26707	00001
d	38233	10110	m	23285	00000	v	8606	1011110
e	117141	001	n	65645	0111	w	22227	110010
f	20846	101110	o	69357	1000	x	1034	1011111001
g	20827	100111	p	17265	100110	y	16877	100100
h	62917	0100	q	1556	1011111011	z	632	1011111000

L'entropia d'aquesta font és $H(X) \approx 4.094353$ i la d'una font uniforme amb el mateix alfabet seria $\log_2(27) \approx 4.75489$. La longitud mitjana del codi de Huffman de la taula és $\widetilde{\ell}(\mathcal{C}_X) \approx 4.134070$ bits per símbol.

Ara s'agafa com a model el Don Quijote de la Mancha, de *Miguel de Cervantes*, que té 2 021 834 caràcters de l'alfabet de 25 lletres més l'espai (la lletra `k` no surt i la lletra `w` surt només dues vegades, en el nom del rei got Wamba). La taula corresponent és:

LLETRA	#	CODI	LLETRA	#	CODI	LLETRA	#	CODI
espai	381208	00	i	90077	11101	s	125728	1010
a	200499	010	j	10530	11110100	y	61749	10011
b	24147	1011111	l	89143	11100	u	79560	11010
c	59437	10010	m	44658	111100	v	17856	1011011
d	87240	11011	n	112683	1000	w	2	11110101000
e	229191	011	o	162514	1100	x	377	11110101001
f	7581	111101011	p	35465	100110	y	25115	1111011
g	17225	1011010	q	32483	101100	z	6491	1111010101
h	19920	1011110	r	100955	11111			

L'entropia de la font és $H(X) \approx 3.968888 < \log_2(26) \approx 4.70044$. La longitud mitjana del codi de Huffman és $\widetilde{\ell}(\mathcal{C}_X) \approx 4.017647$ bits per símbol.

Exemple: el fax. Aquest és un en exemple interessant d'ús de codis de Huffman. El *fax* (facsímil) és un sistema per transmetre informació impresa a través d'una línia telefònica. La idea es va començar a fer servir a mitjans del segle XIX tot i que no és fins a la segona meitat del segle XX que el seu ús es generalitza en tota mena de relacions comercials i administratives. En els darrers anys gairebé ha desaparegut, substituït pel correu electrònic i l'adjunció de fitxers en formats pdf o altres. Aquí s'expliquen versions digitals introduïdes a finals dels anys 60 que comprimeixen les dades usant codis de Huffman.

El funcionament és el següent: un escàner llegeix els fulls seguint línies horitzontals i converteix cadascuna d'aquestes línies en una seqüència binària de longitud 1728 on els zeros representen punts on el paper és blanc i els uns representen els punts on hi ha tinta. El nombre de línies per full varia entre unes 1000 i unes 4000 en un full A4 segons la qualitat que es demani. Així, la quantitat d'informació a transmetre per full està entre 1.7 i 6.5 megabits. A una velocitat de 9600 bauds, típica de les línies telefòniques, això representa entre 3 i 12 minuts d'ús de la línia. Per disminuir el temps de transmissió les dades es comprimeixen amb un procediment que barreja la tècnica anomenada *run length encoding* amb els codis de Huffman. Amb aquesta tècnica la mida de les dades es redueix en mitjana a 1/7 de l'original (o sigui, la factura de telèfon es divideix per 7) i l'enviament d'un fax normalment no triga més d'un minut.

El run length encoding es fa servir per comprimir dades en què hi ha llargues cadenes de símbols repetits; per exemple, la cadena `aaaaaaaabbbbrrrrrrrrr` es codifica posant `8a4b9r`. En el cas del fax, com que els símbols són només zeros i uns, no cal posar-los cada vegada: una seqüència de zeros i uns es tradueix simplement en una seqüència de nombres enters que s'interpreten com a nombre de zeros i d'uns de manera alternada. Aquests nombres es codifiquen separadament amb dos codis de Huffman, un per als que representen cadenes de zeros i un altre per als que representen cadenes d'uns. La taula següent conté una part dels codis de Huffman que formen part de l'estàndard *Grup 4*:

#	zeros	uns	#	zeros	uns
0	00110101	0000110111	61	00110010	000001011010
1	000111	010	62	00110011	000001100110
2	0111	11	63	00110100	000001100111
3	1000	10	...	...	...
4	1011	011	128	10010	000011001000
5	1100	0011	192	010111	000011001001
6	1110	0010	256	0110111	000001011011
...	...	...	...	...	...

Per exemple, la descodificació de la seqüència `101110101101100110011111`... és la cadena binària amb quatre zeros, tres uns, quatre zeros, quatre uns, seixanta-un zeros, dos uns, etc.

3.4 Codificació aritmètica

Referències: Salomon-Motta [16, Sections 5.9, 5.10], Sayood [17].

Aquesta tècnica de codificació probabilista permet codificar la informació emesa per una font sense memòria X assolint el límit inferior del teorema de codificació de font, de manera que es facin servir $H(X)$ bits per cada símbol codificat, sense necessitat d'haver de considerar extensions de la font, com passa quan es fan servir codis de Huffman.

La idea és molt simple: es tracta només de fer un canvi de base en la representació d'un nombre. Això sí, agafant el concepte més general de "bases amb pesos".

Notació posicional. Es recorda el funcionament de la *notació posicional* per representar nombres. S'agafa un nombre $q \geqslant 2$, la *base* de numeració, i un alfabet $\mathbb{D}$ de q símbols diferents, els dígits en base q, que representen els nombres entre 0 i $q-1$. Per simplificar notacions s'agafaran com a dígits els nombres mateixos: $\mathbb{D} = \{\mathtt{0}, \mathtt{1}, \mathtt{2}, \ldots, \mathtt{q}-\mathtt{1}\}$.

Els nombres naturals s'escriuen com una cadena finita de dígits $\mathtt{d}_n\mathtt{d}_{n-1}\ldots\mathtt{d}_2\mathtt{d}_1\mathtt{d}_0$, que representa el nombre natural $\mathtt{d}_0 + \mathtt{d}_1 q + \mathtt{d}_2 q^2 + \cdots + \mathtt{d}_{n-1} q^{n-1} + \mathtt{d}_n q^n$.

Aquí interessa la representació dels nombres reals de l'interval $[0,1)$. Aquests nombres s'escriuen com una successió infinita de dígits $\mathtt{0.d_1d_2d_3}\ldots$ després del punt decimal, que representa el nombre real

$$\alpha = \frac{\mathtt{d}_1}{q} + \frac{\mathtt{d}_2}{q^2} + \cdots + \frac{\mathtt{d}_k}{q^k} + \cdots = \sum_{k=1}^{\infty} \frac{\mathtt{d}_k}{q^k} \in [0,1).$$

Aquesta representació és essencialment única: només s'han de prohibir les successions que repeteixin el dígit $\mathtt{q}-\mathtt{1}$ d'un lloc endavant, ja que els nombres que els corresponen també es poden representar amb successions que repeteixen el dígit $\mathtt{0}$ d'un lloc endavant. Per exemple, en base 10 es té $\mathtt{0.237999}\cdots = \mathtt{0.238000}\ldots$

A la pràctica es treballa amb cadenes finites $\mathtt{0.d_1d_2}\ldots\mathtt{d_n}$ que representen aproximacions del nombre real α: donar aquesta cadena equival a situar α en un interval de longitud $\frac{1}{q^n}$:

$$\alpha \in [\alpha_n, \beta_n), \qquad \alpha_n = \sum_{k=1}^{n} \frac{\mathtt{d}_k}{q^k}, \quad \beta_n = \alpha_n + \frac{1}{q^n}.$$

L'interval $[\alpha_n, \beta_n)$ conté tots els nombres reals $\alpha \in [0,1)$ amb representació en base q que comenci amb aquests mateixos n primers dígits: $\alpha = \mathtt{0.d_1d_2}\ldots\mathtt{d_n}\ldots$

Així, agafar els n primers dígits dels nombres correspon a fer una subdivisió de $[0,1)$ en q^n subintervals, tots de la mateixa longitud $\frac{1}{q^n}$, i cada cadena finita de n dígits determina un d'aquests subintervals.

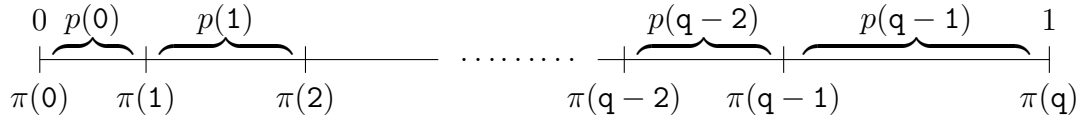
Notació posicional amb pesos. Es pot generalitzar la idea de la notació posicional admetent que cada dígit representi una proporció diferent de nombres de l'interval, d'acord amb uns *pesos* o probabilitats assignades a cadascun: a cada dígit $\mathtt{d} \in \{\mathtt{0}, \mathtt{1}, \mathtt{2}, \ldots, \mathtt{q}-\mathtt{1}\}$ se li associa un nombre $p(\mathtt{d}) \geqslant 0$, de manera que $\sum_{\mathtt{d}=\mathtt{0}}^{\mathtt{q}-\mathtt{1}} p(\mathtt{d}) = 1$. És a dir, s'agafa una distribució de probabilitats en el conjunt dels dígits. Es consideren les probabilitats acumulades

$$\pi(\mathtt{0}) = \mathtt{0},\ \pi(\mathtt{1}) = \mathtt{p(0)},\ \pi(\mathtt{2}) = \mathtt{p(0)} + \mathtt{p(1)},\ \pi(\mathtt{3}) = \mathtt{p(0)} + \mathtt{p(1)} + \mathtt{p(2)}, \quad \pi(\mathtt{d}) = \sum_{\mathtt{d}'<\mathtt{d}} \mathtt{p}(\mathtt{d}'),$$

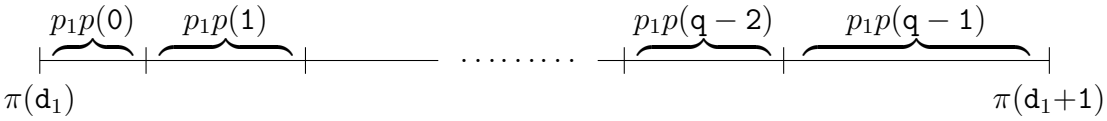
amb $\pi(\mathtt{q}) = \sum_{\mathtt{d}<\mathtt{q}} p(\mathtt{d}) = 1$. Els nombres $\pi(\mathtt{d})$ determinen una partició de l'interval $[0,1)$ en q subintervals

$$\big[\pi(\mathtt{0}),\pi(\mathtt{1})\big), \quad \big[\pi(\mathtt{1}),\pi(\mathtt{2})\big), \quad \big[\pi(\mathtt{q}-\mathtt{1}),\pi(\mathtt{q})\big)$$

de longituds $p(\mathtt{0})$, $p(\mathtt{1})$, …, $p(\mathtt{q}-\mathtt{1})$, respectivament:



Ara els nombres de l'interval $[0,1)$ es poden representar, igual com abans, de manera essencialment única, com a successions infinites de dígits de la forma $\mathtt{0.d_1d_2d_3}\ldots$, que determinen el nombre seguint la mateixa idea: el primer dígit $\mathtt{d_1}$ determina a quin dels q subintervals $\big[\pi(\mathtt{d_1}),\pi(\mathtt{d_1}{+}\mathtt{1})\big)$ pertany el nombre. Un cop determinat aquest interval, que té longitud igual a $p_1 = p(\mathtt{d_1})$, s'el subdivideix anàlogament en subintervals de longituds $p_1 p(\mathtt{d})$ proporcionals a les probabilitats dels dígits:



Amb això el nombre quedarà determinat dins d'un interval que comença en el nombre

$$\pi(\mathtt{d_1}) + p_1 \sum_{\mathtt{d}'<\mathtt{d_2}} p(\mathtt{d}') = \pi(\mathtt{d_1}) + p(\mathtt{d_1})\pi(\mathtt{d_2})$$

i acaba en el nombre $\pi(\mathtt{d_1}) + p(\mathtt{d_1})\pi(\mathtt{d_2}+1)$, de longitud $p(\mathtt{d_1})p(\mathtt{d_2})$, que és un dels q^2 subintervals en què es descompon $[0,1)$ segons els dos primers dígits. Aquests q^2 subintervals $[\alpha_2,\beta_2)$ tenen longituds $p(\mathtt{d_1})p(\mathtt{d_2})$ per a tots els parells de dígits $\mathtt{d_1},\mathtt{d_2} \in \mathbb{D}$.

Fent el mateix amb els digits següents, s'obté una successió $\mathtt{0.d_1d_2}\ldots$ que representa el nombre $\alpha \in [0,1)$ com la suma de la sèrie següent:

$$\begin{aligned}\alpha = \pi(\mathtt{d_1}) + p(\mathtt{d_1})\pi(\mathtt{d_2}) + p(\mathtt{d_1})p(\mathtt{d_2})\pi(\mathtt{d_3}) + \cdots + \big(p(\mathtt{d_1})\cdots p(\mathtt{d}_{n-1})\big)\pi(\mathtt{d}_n) + \cdots \\ = \sum_{k=1}^{\infty} \left(\prod_{i=1}^{n-1} p(\mathtt{d}_i)\right) \pi(\mathtt{d}_k),\end{aligned}$$

on cada producte $\prod_{i=1}^{k-1} p(\mathtt{d}_i)$ és la longitud de l'interval $[\alpha_{k-1},\beta_{k-1})$ que conté α, determinat pels primers $k-1$ dígits $\mathtt{d}_i$, i la probabilitat acumulada $\pi(\mathtt{d}_k)$ indica on comença el subinterval $[\alpha_k,\beta_k) \subset [\alpha_{k-1},\alpha_k)$ que conté α, determinat pel dígit següent $\mathtt{d}_k$. Per a cada $n \geqslant 1$ el nombre α pertany a l'interval $[\alpha_n,\beta_n)$ d'extrems

$$\alpha_n = \sum_{k=1}^{n} \left(\prod_{k=1}^{n-1} p(\mathtt{d}_i)\right) \pi(\mathtt{d}_k), \qquad \beta_n = \alpha_n + \prod_{i=1}^{n} p(\mathtt{d}_i)$$

Codificació aritmètica. La codificació aritmètica d'una font X sense memòria consisteix a considerar els símbols $\mathtt{x} \in \mathcal{X}$ d'un alfabet de $q = |\mathcal{X}|$ lletres com els dígits en base q, i assignar a cadascun el pes donat per la probabilitat $p(\mathtt{x}) = \Pr(X = \mathtt{x})$.

Quan es vol codificar una seqüència $\boldsymbol{x} = \mathtt{x}_1\mathtt{x}_2\mathtt{x}_3\cdots\mathtt{x}_n \in \mathcal{X}^n$ de n símbols emesos per la font simplement es calcula l'interval $[\alpha_n, \beta_n)$ que conté els nombres reals que comencen per aquests dígits. Aquest interval conté els nombres reals $\alpha \in [0, 1)$ tals que el seu desenvolupament q-ari amb pesos $p(\mathtt{x})$ comença per aquesta seqüència de dígits $\alpha = \mathtt{0.x}_1\mathtt{x}_2\ldots\mathtt{x}_n\ldots$

La longitud de l'interval és $\beta_n - \alpha_n = p(\mathtt{x}_1)p(\mathtt{x}_2)\cdots p(\mathtt{x}_n)$. Com que la font és sense memòria els símbols que emet són independents i, per tant, aquesta longitud és la probabilitat $p(\boldsymbol{x}) = \Pr(X^n = \boldsymbol{x})$ que la font emeti la seqüència $\boldsymbol{x} = \mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n$.

La *codificació aritmètica* d'aquesta seqüència $\boldsymbol{x}$ consisteix en la representació en base 2 (aquí la representació posicional ordinària, sense pesos) d'un nombre qualsevol γ que pertanyi a aquest interval: $\gamma \in [\alpha_n, \beta_n)$. Aquest nombre determina completament l'interval entre tots els possibles q^n subintervals en què es divideix $[0, 1)$ a partir dels n primers dígits d'un nombre, i per tant permet recuperar la seqüència $\boldsymbol{x}$ calculant els n primers dígits del seu desenvolupament q-ari amb pesos.

La representació binària d'un nombre és en principi infinita però si l'únic que es vol és assegurar que el nombre pertany a l'interval $[\alpha_n, \beta_n)$ i no a cap altre n'hi ha prou a donar una aproximació de γ amb un nombre de dígits suficient. Com que el dígit binari en la posició ℓ-èsima correspon a un subinterval de longitud $\frac{1}{2^\ell}$ n'hi ha prou a donar ℓ dígits amb

$$\frac{1}{2^\ell} \leqslant \beta_n - \alpha_n = p(\boldsymbol{x}) \Leftrightarrow -\ell \leqslant \log\big(p(\boldsymbol{x})\big) \Leftrightarrow \ell \geqslant -\log\big(p(\boldsymbol{x})\big).$$

Així, per codificar cada cadena $\boldsymbol{x} \in \mathcal{X}^*$ n'hi ha prou en donar les primeres

$$\ell(\boldsymbol{x}) := \big\lceil -\log\big(p(\boldsymbol{x})\big)\big\rceil$$

xifres binàries del nombre γ, que són suficients per determinar completament quin és l'interval $[\alpha_n, \beta_n)$ que el conté entre tots els q^n intervals en què se subdivideix $[0, 1)$, i sabent aquest interval es pot recuperar la cadena $\boldsymbol{x}$.

Això és essencialment la quantitat d'informació que proporciona saber el resultat $\boldsymbol{x}$ de la variable aleatòria X^n. La longitud esperada per símbol d'aquesta codificació, si es tenen en compte totes les cadenes binàries de longitud n que a priori es podrien haver de codificar, és igual a l'entropia de la variable X:

$$\frac{1}{n}\sum_{\boldsymbol{x}\in\mathcal{X}^n} p(\boldsymbol{x})\ell(\boldsymbol{x}) \approx \frac{1}{n}\sum_{\boldsymbol{x}\in\mathcal{X}^n} -p(\boldsymbol{x})\log\big(p(\boldsymbol{x})\big) = \frac{1}{n}H(X^n) = \frac{1}{n}nH(X) = H(X).$$

Per tant la codificació aritmètica, efectivament, codifica seqüències emeses per una font X usant en mitjana $H(X)$ bits per símbol. És essencialment equivalent a codificar blocs arbitràriament llargs de símbols emesos per la font (extensions de la font) usant codis de Huffman.

Descodificació. La descodificació segueix un procés anàleg de canvi de base: es tracta de passar de la base 2 sense pesos a la base dels símbols de $\mathcal{X}$ amb pesos.

Es parteix d'un nombre $\gamma = \sum_{k=1}^{m} \frac{\mathtt{d}_k}{2^k} \in [0,1)$ donat en termes dels seus m primers dígits binaris $\mathtt{d}_k \in \{\mathtt{0},\mathtt{1}\}$. Aquest nombre pertany a un únic subinterval $[\alpha_n, \beta_n) \in [0,1)$ dels q^n subintervals de longituds diferents en què se subdivideix segons totes les q^n cadenes de n símbols de l'alfabet $\mathcal{X}$. Per a tot $k \leqslant n$ aquest subinterval està contingut en un únic $[\alpha_k, \beta_k)$.

Es comença determinant el primer símbol $\mathtt{x}_1$ de la seqüència $\boldsymbol{x}$, que s'obté veient a quin subinterval $\big[\pi(\mathtt{x}_1), \pi(\mathtt{x}_1{+}\mathtt{1})\big)$ pertany γ. Per determinar el segon símbol s'ha de mirar a quin subinterval $\big[\pi(\mathtt{x}_1) + p(\mathtt{x}_1)\pi(\mathtt{x}_2), \pi(\mathtt{x}_1) + p(\mathtt{x}_1)\pi(\mathtt{x}_2{+}\mathtt{1})\big)$ pertany, i així successivament es van recuperant tots els n símbols $\mathtt{x}_k$ de la seqüència $\boldsymbol{x}$.

Exemple: font binària $\mathrm{Ber}(\frac{1}{10})$**.** Es considera una font binària sense memòria amb probabilitats $p(\mathtt{0}) = 0.9$ i $p(\mathtt{1}) = 0.1$. A continuació es veuen les codificacions de les seqüències binàries de longituds $n \leqslant 3$. Per a cadascuna es dona: la seqüència $\boldsymbol{x} \in \{\mathtt{0},\mathtt{1}\}^n$, l'interval $[\alpha_n, \beta_n) \subset [0,1)$ corresponent, el nombre binari $\gamma = \sum_{k=1}^{\ell} \frac{\mathtt{b}_k}{2^k} \in [\alpha_n, \beta_n)$ amb menys dígits que determina unívocament aquest interval, la seqüència de dígits corresponent, que és la codificació $\boldsymbol{c} = \mathrm{enc}(\boldsymbol{x}) = \mathtt{b}_1\mathtt{b}_2\cdots\mathtt{b}_\ell \in \{\mathtt{0},\mathtt{1}\}^\ell$, el nombre ℓ, i la informació $-\log\big(p(\boldsymbol{x})\big)$ continguda en la seqüència que es codifica.

$\boldsymbol{x}$	$[\alpha_n, \beta_n)$	γ	$\boldsymbol{c}$	ℓ	$-\log(p(\boldsymbol{x}))$
0	$[0, 0.9)$	0.5	1	1	0.152
1	$[0.9, 1)$	0.9375	1111	4	3.322
00	$[0, 0.81)$	0.5	1	1	0.304
01	$[0.81, 0.9)$	0.8125	1101	4	3.474
10	$[0.9, 0.99)$	0.9375	1111	4	3.474
11	$[0.99, 1)$	0.992188	1111111	7	6.644
000	$[0, 0.729)$	0.5	1	1	0.456
001	$[0.729, 0.81)$	0.75	11	2	3.626
010	$[0.81, 0.891)$	0.8125	1101	4	3.626
011	$[0.891, 0.9)$	0.898438	1110011	7	6.796
100	$[0.9, 0.981)$	0.9375	1111	4	3.626
101	$[0.981, 0.99)$	0.984375	111111	6	6.796
110	$[0.99, 1)$	0.992188	1111111	7	6.796
111	$[0.999, 1)$	0.9990023	1111111111	10	9.966

Es pot observar que la longitud del codi $\boldsymbol{c}$ s'assembla a la informació $-\log(p(\boldsymbol{x}))$.

Per veure la compressió es pot calcular, per exemple, la codificació de la cadena binària $\boldsymbol{x} = \mathtt{000100000100000}$ de longitud 15, amb una distribució de zeros i uns propera a la de la font considerada: en aquesta cadena hi ha $\frac{2}{15}$ uns i $\frac{13}{15}$ zeros. Es té $p(\boldsymbol{x}) \approx 0.00254$ i $-\log\big(p(\boldsymbol{x})\big) \approx 8.62$. Per tant la seva codificació aritmètica hauria d'usar de l'ordre de 8 o 9 bits, produint-se una compressió. Els extrems de l'interval són $\alpha_{15} = 0.6948420489$ i $\beta_{15} = 0.697383914728329$. El nombre $\gamma = 0.6953125$ en binari té 7 bits i determina unívocament l'interval. La codificació de la cadena és $\boldsymbol{c} = \mathtt{1011001}$, de longitud $\ell(\boldsymbol{c}) = 7$.

A la taula es veu que no n'hi ha prou amb la paraula codi $\boldsymbol{c}$ per recuperar $\boldsymbol{x}$. En efecte, cada nombre binari $\gamma = \sum_{k=1}^{\ell} \frac{\mathtt{b}_k}{2^k} \in [\alpha_n, \beta_n)$ determina, per a cada $m \geqslant n$, un dels subintervals $[\alpha_m, \beta_m) \subset [\alpha_n, \beta_n)$. Per tant, si $\boldsymbol{x}$ és una descodificació de la seqüència $\boldsymbol{c}$, aquesta mateixa

seqüència té altres infinites descodificacions possibles, de totes les longituds més grans que la de $\boldsymbol{x}$, totes començant per $\boldsymbol{x}$. En la taula, per exemple, la seqüència de quatre bits `1111` descodifica en `1`, `10`, `100`, etc. però també correspon a altres infinites seqüències binàries, una per a cada longitud > 3.

Per poder recuperar $\boldsymbol{x}$ el descodificador usa el codi $\boldsymbol{c}$, però també necessita la mida del text $\boldsymbol{x}$, per saber quan ha de parar de calcular més símbols. Aquest problema es pot evitar, si cal, introduint un símbol *EOF* més en l'alfabet $\mathcal{X}$, amb una probabilitat molt petita ja que només es farà servir una vegada, i afegir aquest símbol al final de $\boldsymbol{x}$ per tal que el descodificador sàpiga que ja té el text $\boldsymbol{x}$ complet.

Exemple. Es considera la seqüència

$$\boldsymbol{x} = \texttt{setzejutgesdunjutjatmengenfetgedunpenjat}$$

de longitud $\ell(\boldsymbol{x}) = 40$. Es construeix una font adaptada a aquest text que tingui en compte les lletres que hi surten i les seves probabilitats. Amb la distribució d'aquesta font es té $p(\boldsymbol{x}) \approx 10^{-41}$ i $-\log\big(p(\boldsymbol{x})\big) \approx 135.0028$. Els extrems de l'interval corresponent són

$$\alpha_{40} \approx 0.688719964480347026291437918864705210969822405022 6\ldots$$
$$\beta_{40} \approx 0.688719964480347026291437918864705210969845319050 4\ldots$$

Agafant el nombre binari finit més curt que determina l'interval es troba el codi

$$\begin{aligned}\boldsymbol{c} =& \texttt{10110000010011111111001110011011100010111001 1}\\ & \texttt{10110000001101110111111111001000101101010010 0}\\ & \texttt{00111101100000110111100010110000011001101 01}\end{aligned}$$

de longitud $\ell(\boldsymbol{c}) = 133$ bits.

Problemes de precisió. Aritmètica entera. La descripció que s'ha fet de la codificació aritmètica dona la idea general. A la practica, implementar-la exactament com s'ha dit requeriria treballar amb nombres decimals de precisió molt gran si es tracta de codificar seqüències de símbols de $\mathcal{X}$ molt llargues. Tot i que això en principi és possible, adaptant la precisió dels càlculs a la mida de la cadena a codificar, a la pràctica resultaria extraordinàriament costós i no és viable.

Amb petites modificacions els algorismes de codificació i descodificació es poden adaptar per treballar només amb aritmètica de nombres enters que es poden escriure amb un nombre k de bits prefixat (per exemple enters de 16 o de 32 bits) de manera que només calgui fer operacions d'aritmètica entera (suma, producte i divisió euclidiana) amb aquests nombres.

La idea bàsica és que en anar calculant els extrems de l'interval $[\alpha_n, \beta_n)$ aquests nombres es van apropant entre ells i cada vegada tenen més dígits en comú al començament, de manera que la diferència entre tots dos queda reflectida en els dígits d'un lloc endavant. Els dígits inicials iguals romanen iguals en tots els passos següents. Per tant el que es fa és anar eliminant aquests dígits (el qual equival a reescalar l'interval), que ja formen part del codi

$\boldsymbol{c}$, i tota l'estona treballar només amb els primers k dígits dels dos extrems de l'interval que siguin diferents.

En comptes de considerar els nombres reals de l'interval $[0, 1)$ es treballa amb els nombres enters de l'interval $[0, 2^k - 1)$, que són els que es poden representar amb k bits. Això equival a fer un escalat de factor $2^k - 1$ i una quantització que consisteix a aproximar els nombres reals amb la seva part entera per defecte.

En el procés de codificació es van actualitzant variables α, β que representen nombres enters $\alpha, \beta \in [0, 2^{k-1})$ de manera anàloga a com es faria treballant amb precisió infinita, de manera que en cada pas l'interval $[\alpha, \beta]$ correspon a la darrera lletra de $\boldsymbol{x}$ codificada. Els nombres α i β es representen sempre usant exactament k bits, amb zeros a l'esquerra si cal.

Algorisme de codificació aritmètica. Les dades són la cadena $\boldsymbol{x} \in \mathcal{X}^*$ sobre l'alfabet q-ari $\mathcal{X} = \{\mathtt{x}_1, \ldots, \mathtt{x}_q\}$ i la distribució de probabilitats de la font. L'algorisme retorna una cadena binària $\boldsymbol{c}$.

Inicialització: Es posa a $\boldsymbol{c}$ la cadena binària buida i s'inicialitza:

$$\alpha \leftarrow 0 = \mathtt{000}\overset{k)}{\cdots}\mathtt{00}, \qquad \beta \leftarrow 2^k - 1 = \mathtt{111}\overset{k)}{\cdots}\mathtt{11}.$$

Repetir: Per a cada lletra $\mathtt{x}$ de la cadena,

- Descomposició de l'interval: Es descompon l'interval $[\alpha, \beta]$ en n subintervals disjunts:
$$\big[\, \alpha + \lfloor \delta\pi_{i-1} \rfloor,\ \alpha + \lfloor \delta\pi_i \rfloor - 1 \,\big], \quad i = 1, \ldots, n,$$
on $\lfloor \cdot \rfloor$ denota la part entera per defecte d'un nombre real i $\delta = \beta - \alpha + 1$.
- Actualització de l'interval: L'índex i corresponent a la lletra $\mathtt{x}$ que s'ha de codificar determina quin dels q subintervals s'ha d'agafar: si $\mathtt{x} = \mathtt{x}_i \in \mathcal{X}$ s'actualitzen els valors de α i β a:
$$\alpha \leftarrow \alpha + \lfloor \delta\pi_{i-1} \rfloor, \qquad \beta \leftarrow \alpha + \lfloor \delta\pi_i \rfloor - 1.$$
- Reescalat i codificació:[2] Si el primer bit de α i β coincideix s'elimina d'elles i s'afegeix al codi $\boldsymbol{c}$, i a α i β se'ls afegeixen un $\mathtt{0}$ i un $\mathtt{1}$ al final respectivament. És a dir, si $\alpha = \mathtt{a}_1\mathtt{a}_2\cdots\mathtt{a}_k$ i $\beta = \mathtt{b}_1\mathtt{b}_2\cdots\mathtt{b}_k$ amb $\mathtt{a}_1 = \mathtt{b}_1$, aleshores
$$\alpha \leftarrow \mathtt{a}_2\mathtt{a}_3\cdots\mathtt{a}_k\mathtt{0}, \qquad \beta \leftarrow \mathtt{b}_2\mathtt{b}_3\cdots\mathtt{b}_k\mathtt{1}, \qquad \boldsymbol{c} \leftarrow \boldsymbol{c}||\mathtt{a}_1.$$
Es repeteix això tantes vegades com calgui fins que sigui $\mathtt{a}_1 \neq \mathtt{b}_1$. Observi's que, en acabar aquest pas, es tindrà necessàriament $\mathtt{a}_1 = \mathtt{0}$ i $\mathtt{b}_1 = \mathtt{1}$.

L'algorisme acaba quan s'ha processat l'última lletra de $\boldsymbol{x}$.

En acabar s'ha d'afegir un bit $\mathtt{1}$ al codi $\boldsymbol{c}$ abans de retornar-lo per garantir que en la descodificació es recuperarà correctament l'última lletra de $\boldsymbol{x}$: els últims valors de α i β seran de la forma $\mathtt{0a}_2\cdots\mathtt{a}_k$ i $\mathtt{1b}_2\cdots\mathtt{b}_k$ i qualsevol nombre que comenci en binari amb un $\mathtt{1}$ serà més gran que el primer i menor o igual que el segon, i per tan pertany a l'interval $[\alpha, \beta]$.

[2] Ull! això és una versió incompleta que s'haurà de modificar.

Desbordament. L'algorisme que s'acaba de descriure no sempre funciona correctament ja que es poden produir problemes de desbordament (*underflow*). En l'algorisme es dona per suposat que els subintervals de la descomposició són no buits (altrament el reescalat no funcionaria), però això no sempre està garantit. La condició de ser no buit equival a què $\lfloor \delta\pi_{i-1} \rfloor \neq \lfloor \delta\pi_i \rfloor$; és a dir, que els nombres $\delta\pi_{i-1}$ i $\delta\pi_i$ pertanyin a dos intervals diferents entre enters consecutius.

Això es podria garantir imposant que $\delta(\pi_i - \pi_{i-1}) = \delta p_i$ sigui $\geqslant 1$ per a tot $i = 1, \ldots, n$. És a dir, que δ no sigui mai més petit que l'invers de la probabilitat més petita en la distribució. Però garantir això no és possible: les probabilitats venen donades per la font i el valor de δ pot ser molt petit en tota codificació. El cas pitjor es dona quan l'interval correspon als nombres $\beta = \mathtt{100\cdots0}$ i $\alpha = \mathtt{011\cdots1}$ en què es té $\delta = \beta - \alpha + 1 = 2$.

Per solucionar aquesta pega s'ha de complicar una mica l'algorisme, de la manera següent: en cada pas, després del procés de reescalat es tindrà $\alpha = \mathtt{0a_2a_3\cdots a_k}$ i $\beta = \mathtt{1b_2b_3\cdots b_k}$. Aquests nombres estan massa a prop quan els primers dígits de α després del zero siguin uns i els primers dígits de β després de l'u siguin zeros.

S'introdueix una variable $\mathtt{u}$, que s'inicialitza amb el valor $\mathtt{u} = 0$ i s'afegeix un pas més a l'algorisme, que caldrà fer sempre després de cada pas de reescalat i codificació:

PREVENCIÓ DE DESBORDAMENT: Siguin $\alpha = \mathtt{0a_2\cdots a_k}$ i $\beta = \mathtt{1b_1\cdots b_k}$ els valors que provenen del reescalat que s'acaba de fer. Suposi's que $\mathtt{a_2} = \mathtt{1}$ and $\mathtt{b_2} = \mathtt{0}$. Aleshores s'eliminen aquests segons bits de α i β i se'ls afegeixen al final un $\mathtt{0}$ i un $\mathtt{1}$, respectivament, i s'incrementa el comptador $\mathtt{u}$:

$$\alpha \leftarrow \mathtt{0a_3a_4\cdots a_k0}, \qquad \beta \leftarrow \mathtt{1b_3b_4\cdots b_k1}, \qquad \mathtt{u} \leftarrow \mathtt{u} + 1.$$

Això es repeteix tantes vegades com calgui. Al final, o bé la variable α començarà amb dos zeros o bé la variable β començarà amb dos uns.

El comptador $\mathtt{u}$ contindrà el nombre de vegades que s'ha fet aquest això.

El pas de reescalat i codificació s'ha de modificar per tenir en compte això, i queda de la manera següent:

REESCALAT I CODIFICACIÓ: Si els primers bits de α i β coincideixen es modifiquen igual que en la versió anterior, les vegades que calgui; a més, en cada pas:

- si el primer bit de α i β era un $\mathtt{0}$ s'afegeixen $\mathtt{u}$ uns a $\boldsymbol{c}$;
- si el primer bit de α i β era un $\mathtt{1}$ s'afegeixen $\mathtt{u}$ zeros a $\boldsymbol{c}$;
- es posa el comptador a zero: $\mathtt{u} \leftarrow 0$.

Observi's que si cal reescalar diverses vegades només en la primera el comptador pot ser $\neq 0$ i pot haver-se de modificar $\boldsymbol{c}$ afegint els $\mathtt{u}$ bits que corresponguin. Això compensa la eliminació de bits de α i β sense afectar $\boldsymbol{c}$ que s'hagin pogut fer en l'anterior pas de prevenció de desbordament.

Lema 3.18. *Sigui $p = \min\{p_i : 1 \leqslant i \leqslant n\}$ la probabilitat més petita de la distribució de la font. L'algorisme de codificació aritmètica anterior funciona correctament sempre que es treballi amb aritmètica amb nombre de bits*

$$k \geqslant -\log p + 2.$$

PROVA: El funcionament del pas de prevenció de desbordament és el següent: quan els extrems de l'interval tenen la forma següent:

$$\alpha = \mathtt{0111a}_5\mathtt{a}_6\cdots\mathtt{a}_k, \qquad \beta = \mathtt{1000b}_5\mathtt{b}_6\cdots\mathtt{b}_k$$

no es pot saber quin serà el següent bit que s'afegirà al codi $\boldsymbol{c}$.

Si es treballés amb precisió infinita, després de codificar unes quantes lletres més de $\boldsymbol{x}$ els valors que es tindrien α' i β', que es van acostant, acabarien tenint el mateix bit inicial, Com que es treballa amb precisió finita, només amb k bits, això podria passar quan els valors s'hagin de calcular amb més de k bits.

Com que $\alpha \leqslant \alpha' < \beta' \leqslant \beta$, si el bit més significant acaba sent un zero aleshores l'expansió binària de α' i β' serà de la forma:

$$\alpha' = \mathtt{0111a}'_5\mathtt{a}'_6\mathtt{a}'_7\cdots, \qquad \beta' = \mathtt{0111b}'_5\mathtt{b}'_6\mathtt{b}'_7\cdots,$$

i si acaba sent un `1`, aleshores aquestes expansions serien de la forma:

$$\alpha' = \mathtt{1000a}'_5\mathtt{a}'_6\mathtt{a}'_7\cdots, \qquad \beta' = \mathtt{1000b}'_5\mathtt{b}'_6\mathtt{b}'_7\cdots.$$

Per aquesta raó, tan aviat com es decideixi quin és el primer bit, es pot estar segur de quins seran els `u` bits que vindran a continuació, que corresponen als bits que s'han tret de α i de β en el pas de prevenció de desbordament: uns si el primer bit s'ha estabilitzat en un `0` i zeros si s'ha estabilitzat en un `1`.

Un cop fet aquest pas l'interval $[\alpha, \beta]$ se separa en n subintervals i se'n tria un d'acord amb la lletra $\mathtt{x}_i$ que s'hagi de codificar. Els primers dos bits de les noves α i β extrems del subinterval no poden ser els parells `01` i `10`.

- si són `00` i `11` aleshores $\delta \geqslant 2^{k-1}$;
- si són `01` i `11` aleshores $\delta \geqslant 2^{k-2}$.

En qualsevol dels dos casos es té $\delta \geqslant 2^{k-2}$ i per tant per assegurar la condició $\delta p \geqslant 1$ per a tota probabilitat p n'hi ha prou que sigui $2^{k-2} \geqslant p^{-1}$, que és equivalent a la condició de l'enunciat. □

Algorisme de descodificació. El descodificador, que naturalment ha de tenir la informació sobre la font, rep com a input el codi $\boldsymbol{c} \in \{\mathtt{0},\mathtt{1}\}^*$ i la longitud del missatge $\boldsymbol{x} \in \mathcal{X}^*$ que codifica. Observi's que el codi $\boldsymbol{c}$ es pot allargar afegint-li tants zeros com es vulgui sense afectar la descodificació, ja que el nombre de $[0, 1)$ que li correspon a aquest altre desenvolupament binari amb més zeros al final no canvia. Recordi's que tal com s'ha acabat la codificació, l'últim bit de $\boldsymbol{c}$ sempre serà un `1`.

El procés de descodificació queda controlat per tres variables, α, β i γ que són cadenes binaries de longitud k i representen enters de l'interval $[0, 2^k)$ de la manera habitual (els bits més significatius a l'esquerra). Les primeres dues representen els extrems d'un interval no buit i la tercera és un enter dins d'aquest interval, de manera que sempre s'han de satisfer les desigualtats $\alpha < \beta$ i $\alpha \leqslant \gamma \leqslant \beta$.

INICIALITZACIÓ: Es posa a $\boldsymbol{x}$ la cadena buida sobre l'alfabet de la font, i les variables s'inicialitzen amb els valors $\alpha \leftarrow 0 = \mathtt{000\cdots0}$, $\beta \leftarrow 2^k - 1 = \mathtt{111\cdots1}$ i $\gamma \leftarrow \mathtt{c_1c_2c_3\cdots c_k}$, els k primers bits de $\boldsymbol{c}$.

DESCOMPOSICIÓ DE L'INTERVAL: Com en la codificació es considera la descomposició de $[\alpha, \beta]$ en n subintervals no buits disjunts:

$$\big[\, \alpha + \lfloor \delta\pi_{i-1} \rfloor,\ \alpha + \lfloor \delta\pi_i \rfloor - 1 \,\big], \quad i = 1, \ldots, n,$$

DESCODIFICACIÓ: El subinterval que determina quina és la lletra següent de $\boldsymbol{x}$ és el que conté γ: si i és l'índex del subinterval tal que

$$\gamma \in \big[\, \alpha + \lfloor \delta\pi_{i-1} \rfloor,\ \alpha + \lfloor \delta\pi_i \rfloor - 1 \,\big]$$

aleshores s'afegeix la lletra $\mathtt{x}_i \in \mathcal{X}$ corresponent a aquest índex a la descodificació $\boldsymbol{x}$ i s'actualitzen els valors de α i β als extrems d'aquest subinterval.

Si $\boldsymbol{x}$ ja té la longitud demanada, la descodificació es completa: es retorna $\boldsymbol{x}$ i s'acaba.

REESCALAT: Suposi's que el primer bit de α i β coincideixen. En aquest cas el primer bit de γ també ha de ser el mateix. Aleshores s'eliminen aquests bits de totes tres variables i se'ls afegeix un `0` a α, un `1` a β i el primer bit de $\boldsymbol{c}$ no usat encara a γ. Es repeteix això tantes vegades com sigui necessari fins que α comenci amb `0` i β amb `1`.

DESBORDAMENT: Suposi's que α comença amb `01` i β comença amb `10`. Aleshores γ ha de començar amb `01` o bé amb `10`. En aquest cas s'elimina el segon bit de totes tres variables i se'ls afegeix un bit més de la mateixa manera que en el pas anterior: un `0` a α, un `1` a β i el primer bit de $\boldsymbol{c}$ no usat encara a γ. Es repeteix això fins que almenys una de les dues variables α o β comenci amb `00` o amb `11`, respectivament. Aleshores es torna al pas de descomposició de l'interval.

Així, el control del desbordament és més senzill en la descodificació que no pas en la codificació: no és necessari guardar en una variable `u` el nombre de vegades que es fa el pas corresponent.

Exemple. Es considera una font de Bernoulli amb probabilitats $p(\mathtt{0}) = 0.9$ i $p(\mathtt{1}) = 0.1$ Com que $-\log(0.1) \approx 3.32$ és suficient treballar amb $k = 6$ bits. Es vol codificar la cadena $\boldsymbol{x} = \mathtt{1010000000}$ de longitud 10. A la taula següent es donen les dades en el procés de codificació indicant la lletra `x` que es codifica en cada pas, els extrems α i β de l'interval que li correspon, els extrems després dels reescalats que s'han fet, els extrems després de la

prevenció de desbordament quan hagi calgut, i què conté el codi binari $\boldsymbol{c}$ després de cada pas:

x	α, β	reescalat	desbordament	$\boldsymbol{c}$
inici	000000, 111111	–	–	$\varnothing$
1	111001, 111111	001000, 111111	–	111
0	001000, 111001	–	–	111
1	110101, 111001	010100, 100111	001000, 101111	11111
0	001000, 101011	–	–	11111
0	001000, 100111	–	–	11111
0	001000, 100011	–	–	11111
0	001000, 100000	–	–	11111
0	001000, 011101	010000, 111011	–	1111101
0	010000, 110110	–	–	1111101
0	001000, 110010	–	–	1111101

Observi's que només en un cas s'ha hagut de fer el pas de prevenció de desbordament (una sola vegada): en codificar el tercer bit de $\boldsymbol{x}$, després de reescalar dues vegades, els valors de α i β comencen amb 01 i 10, de manera que estan massa pròxims. La recuperació del bit que s'ha perdut no es fa fins que es codifica el vuitè bit de $\boldsymbol{x}$, ja que entre el tercer i el vuitè no cal fer cap reescalat. Aquest bit que es recupera és el que està en vermell a la $\boldsymbol{c}$ de la taula.

Finalment s'afegeix un 1 i es retorna el codi $\boldsymbol{c} = 11111011$ de longitud 8. Les dades del procés de descodificació es donen a la taula següent, on s'indiquen en blau els dígits binaris que es van eliminant en cada pas de reescalat i en vermell els que s'eliminen en recuperar-se de la prevenció de desbordament:

α, β	γ	x	reescalat	desbordament
000000, 111111	111110	$\varnothing$	–	–
111001, 111111	111110	1	001000, 111111	–
001000, 111001	110110	10	–	–
110101, 111001	110110	101	010100, 100111	001000, 101111
001000, 101011	010000	1010	–	–
001000, 100111	010000	10100	–	–
001000, 100011	010000	101000	–	–
001000, 100000	010000	1010000	–	–
001000, 011101	010000	10100000	010000, 111011	–
010000, 110110	100000	101000000	–	–
001000, 110010	100000	1010000000	–	–

Un cop s'han obtingut ja $10 = \ell(\boldsymbol{x})$ dígits la descodificació s'atura.

3.5 Mètodes de diccionari

Referències: Salomon-Motta [16, Chapter 6], Sayood [17, Chapter 5].

Els anys 1977 i 1978 *Abraham Lempel* i *Jacob Ziv* (Technion Institute of Technology) proposen mètodes de compressió universals on no cal saber la font a partir de la qual s'han generat les dades, tot i que, això sí, la compressió assolida sí que dependrà de les seves propietats estocàstiques, i per tant de l'entropia d'una font que serveixi de model. Aquests mètodes es coneixen amb els acrònims *LZ77 i LZ78*, formats amb les inicials dels creadors i l'any de publicació.

De seguida van aparèixer propostes de variacions i millores que han donat lloc a nombroses variants que es coneixen amb el nom genèric de *mètodes de diccionari*, i s'han convertit en els preferits per a la compressió de fitxers en ordinadors: es fan servir als compressors `zip`, `pkzip`, `compress`, `7zip`, etc.

El nom "mètode de diccionari" fa referència a la idea general subjacent a tots dos: es crea un diccionari de paraules (blocs de text) que es van trobant a mesura que es processa el text a comprimir, de manera que quan més endavant tornen a sortir aquestes paraules es poden codificar simplement fent referència a la posició que ocupen en el diccionari.

Això es pot interpretar de la manera següent: es vol comprimir una seqüència $\boldsymbol{x} \in \mathcal{X}^*$ de lletres d'un alfabet. Per fer-ho, primerament es descompon $\boldsymbol{x}$ en blocs, no necessàriament de la mateixa mida, d'acord amb un procediment que s'ha d'especificar per a cada mètode. A continuació cada bloc es tradueix en una estructura de dades, que es coneix amb el nom de "token", formada per nombres enters i/o caràcters de l'alfabet $\mathcal{X}$. Els nombres enters dels tokens són punters cap a altres blocs anteriors, que es pensen com les paraules en un diccionari.

En una segona fase els tokens es codifiquen en binari, per exemple amb un codi de bloc de longitud suficient, o millor usant un codi de Huffman.

Per descodificar primer es recuperen els tokens a partir del codi binari i després es cadascun d'ells es converteix en el bloc de lletres que representa.

LZ77. Aquest mètode, conegut també amb el nom de *finestra lliscant*, va ser proposat per Lempel i Ziv en el seu article [24]. Sigui $\boldsymbol{x} = \mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n \in \mathcal{X}^n$ la cadena que es vol comprimir. Es fixen dos enters s i t i en el procés de codificació es consideren finestres de la forma:

$$\cdots\, \mathtt{x}_{r-s-1} \boxed{\mathtt{x}_{r-s}\cdots\mathtt{x}_{r-2}\,\mathtt{x}_{r-1}} \boxed{\mathtt{x}_r\,\mathtt{x}_{r+1}\cdots\mathtt{x}_{r+t-1}} \mathtt{x}_{r+t}\,\cdots \tag{7}$$

de longitud $s+t$ dividides en dues parts:

- l'esquerra conté els s últims caràcters $\mathtt{x}_{r-s}\cdots\mathtt{x}_{r-2}\,\mathtt{x}_{r-1}$ que ja s'han codificat;
- la dreta conté els t primers caràcters $\mathtt{x}_r\,\mathtt{x}_{r+1}\cdots\mathtt{x}_{r+t-1}$ que encara s'han de codificar.

En aquest mètode el diccionari no es construeix de manera explícita sinó que queda determinat implícitament per la finestra (7): les paraules del diccionari són totes les subcadenes del text contingudes dins de la finestra que comencin en alguna lletra de la part esquerra (search buffer) i tinguin longitud $\leqslant t$.

Els tokens són triplets $(\theta, \lambda, \mathtt{x})$ que contenen dos enters θ i λ i una lletra $\mathtt{x} \in \mathcal{X}$. Els enters, anomenats *desplaçament* i *longitud*, han de satisfer $1 \leqslant \theta \leqslant s$ i $0 \leqslant \lambda < t$, i determinen la paraula $\mathtt{x}_{r-\theta}\mathtt{x}_{r-\theta+1}\cdots\mathtt{x}_{r-\theta+\lambda-1}$ del diccionari: el desplaçament serveix per indicar on comença la paraula i la longitud indica quantes lletres la formen.

Algorisme de codificació. L'entrada és una seqüència $\boldsymbol{x} = \mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n \in \mathcal{X}^*$. Un comptador r indica quina part del text ha estat processada: en cada moment és l'índex de la primera lletra que no ha estat codificada. El resultat és una llista de tokens.

INICIALITZACIÓ: s'agafa com a llista de tokens la llista buida, i es posa $r \leftarrow 1$.

REPETIR: mentre $r \leqslant n$,

MATCH: es busca la coincidència més llarga entre les lletres del començament de la part dreta de la finestra (7) amb alguna de les paraules del diccionari: el valor més llarg de $\lambda \geqslant 0$ tal que

$$\mathtt{x}_r \cdots \mathtt{x}_{r+\lambda-1} = \mathtt{x}_{r-\theta} \cdots \mathtt{x}_{r-\theta+\lambda-1}$$

per a algun desplaçament θ.

CODIFICACIÓ: s'afegeix $(\theta, \lambda, \mathtt{x})$ a la llista de tokens, on $\mathtt{x} = \mathtt{x}_{r+\lambda}$ és la primera lletra que no coincideix. Observi's que si $\lambda = 0$, és a dir, si la lletra $\mathtt{x}_r$ no apareix a la finestra esquerra, aleshores es pot agafar el valor del desplaçament θ que es vulgui.

COMPTADOR: S'avança el comptador en $\lambda + 1$ unitats $r \leftarrow r + \lambda + 1$.

La codificació acaba quan el comptador és $r > n$.

La codificació binària dels tokens es pot fer usant $\lceil \log s \rceil + \lceil \log t \rceil + \lceil \log |\mathcal{X}| \rceil$ bits per cadascun amb un codi de bloc, però això es pot millorar si es fa servir un codi de Huffman adaptat a la freqüència dels diferents tokens en la codificació.

La mida de la finestra es fixa en funció de la compressió que es vulgui assolir i del temps que es vulgui invertir en fer-ho: com més gran sigui més possibilitats de coincidències més llargues, i per tant de comprimir, però també es tardarà més ja que s'ha de buscar en un diccionari més gran i també la codificació binària dels tokens serà més llarga.

Naturalment, al començament de la compressió la part esquerra de la finestra conté menys de s lletres i per tant la mida del diccionari és més petita i la possibilitat de coincidències és menor. De fet, per a seqüències molt curtes el mètode expandeix les dades, però si les seqüències són prou llargues i les lletres del text tenen molta correlació, aleshores de seguida es comencen a produir coincidències llargues, amb la compressió consegüent. El motiu que els tokens tinguin un caràcter en la tercera component és que al començament no hi ha coincidències ni tan sols d'una lletra i per tant no es pot codificar només amb un punter a una paraula del diccionari.

Algorisme de descodificació. És un algorisme molt simple i d'execució rapidíssima, comparat amb la compressió. L'entrada és una llista de tokens de la forma $(\theta, \lambda, \mathtt{x})$.

INICIALITZACIÓ: s'inicialitza el text $\boldsymbol{x} \in \mathcal{X}^*$ amb la paraula buida, que no conté cap lletra.

REPETIR: per a cada token $(\theta, \lambda, \mathtt{x})$,

DESCODIFICACIÓ: Sigui $r = \ell(\boldsymbol{x})+1$ la longitud de la seqüència que ja s'ha descodificat més 1. S'afegeix a $\boldsymbol{x}$ la paraula $\mathtt{x}_{r-\theta}\mathtt{x}_{r-\theta+1}\cdots\mathtt{x}_{r-\theta+\lambda-1}$ i després la lletra $\mathtt{x}$.

Per exemple, el text

$$\texttt{En\_un\_lugar\_de\_la\_mancha, \_de\_cuyo}$$

es codificaria amb la seqüència de tokens

$$(1,0,\texttt{E}),(1,0,\texttt{n}),(1,0,\texttt{\_}),(1,0,\texttt{u}),(3,2,\texttt{l}),(1,0,\texttt{u}),(1,0,\texttt{g}),(1,0,\texttt{a}),(1,0,\texttt{r}),$$
$$(1,0,\texttt{\_}),\texttt{d},\texttt{e},(9,2,\texttt{a}),\texttt{m},\texttt{a},\texttt{n},\texttt{c},\texttt{h},\texttt{a},,,(13,4,\texttt{c}),\texttt{u},\texttt{y},\texttt{o}$$

on les lletres $\boldsymbol{x}$ corresponen als tokens del tipus $(\lambda, 0, \boldsymbol{x})$ que surten quan no es troba cap coincidència al diccionari.

Variants. L'any 1982 Storer i Szymanski proposen una millora, coneguda com a *LZSS*, que consisteix a considerar tokens de dos tipus diferents, els quals es poden codificar després en binari separadament, per exemple usant un bit de flag. Les coincidències amb el diccionari que tinguin de longitud $\lambda \geqslant 2$ (o més gran que un $k > 1$ prefixat) es tradueixen en el token (θ, λ) i quan no es troba cap coincidència prou llarga s'agafa com a token simplement el primer caràcter que s'ha de codificar.

Més endavant, a finals dels 80, *Phil Katz* proposa una nova millora sobre LZSS, que bàsicament consisteix en una manera astuta de codificar els tokens usant codis de Huffman. Es tracta de fer servir dos codis: un codifica simultàniament les longituds λ i les lletres de l'alfabet $\mathcal{X}$, i un altre codifica la mida en bits del desplaçament θ. Això a més estalvia l'ús del bit de flag rebaixant en un bit el codi de cada token ja que si la descodificació del primer codi és una longitud λ se sap que el token és d'un tipus i si és una lletra de $\mathcal{X}$ se sap que és de l'altre. En cas que correspongui a un token de tipus (λ, θ) a continuació s'afegeix el codi del nombre de bits de θ seguit del seu valor usant aquest nombre de bits.

Aquest algorisme es coneix amb el nom de *deflate* i és el que implementa un dels compressors més populars: `pkzip`.

LZ78. Mètode, proposat per Lempel i Ziv en el seu article [25].

Aquest mètode va creant un diccionari amb les paraules que es van trobant a mesura que es codifica/descodifica, que es fa servir per codificar paraules que surtin més endavant en el text: quan s'han de codificar paraules que ja han sortit abans, i per tant ja estan al diccionari, es codifiquen amb un punter cap a la posició corresponent. El diccionari comença buit i cal anar omplint-lo tant per comprimir com per descomprimir.

Els tokens són parells $(\imath, \boldsymbol{x})$ formats per un enter $\imath$, que indica una posició en el diccionari, i una lletra $\boldsymbol{x} \in \mathcal{X}$, que és la primera lletra del text a codificar després de les que coincideixen amb la paraula $\imath$-èsima del diccionari.

Algorisme de codificació. L'entrada és una seqüència $\boldsymbol{x} = \mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n \in \mathcal{X}^*$. Un comptador r indica quina part del text ha estat processada: en cada moment és l'índex de la primera lletra que no ha estat codificada. En el procés es treballa amb el "diccionari" que és una llista ordenada de paraules de $\mathcal{X}^*$. El resultat és una llista de tokens.

INICIALITZACIÓ: s'agafa com a llista de tokens la llista buida, el diccionari conté només una paraula, la paraula buida $\lambda \in \mathcal{X}^0$, i es posa $r \leftarrow 1$.

REPETIR: mentre $r \leqslant n$,

MATCH: es busca la coincidència més llarga entre el text a codificar i una paraula del diccionari: el nombre $k \geqslant 0$ més gran tal que $\mathtt{x}_r\mathtt{x}_{r+1}\cdots\mathtt{x}_{r+k-1}$ és una paraula del diccionari.

CODIFICACIÓ: s'afegeix el token $(\imath, \mathtt{x}_{r+k})$ a la llista, on $\imath$ és la posició de la paraula del diccionari; observi's que sempre hi haurà coincidència ja que la paraula buida coincideix amb la paraula a codificar de longitud zero.

DICCIONARI: S'actualitza el diccionari afegint la paraula $\mathtt{x}_r\mathtt{x}_{r+1}\cdots\mathtt{x}_{r+k-1}\mathtt{x}_{r+k}$ que s'acaba de codificar.

COMPTADOR: S'avança el comptador en $k+1$ unitats $r \leftarrow r+k+1$.

La codificació acaba quan el comptador és $r > n$.

El pas de codificació s'ha de modificar lleugerament quan s'arriba al final del text: si la paraula més llarga que coincideix acaba en l'última lletra, o sigui si $r+k-1 = n$, aleshores es considera la coincidència només amb el prefix de $k-1$ lletres (que també serà al diccionari perquè el diccionari conté tots els prefixos de les paraules que conté) per així poder posar l'última lletra $\mathtt{x}_n$ en el token corresponent.

Observi's que els tokens de la forma $(\imath, \mathtt{x})$ amb punter $\imath = 0$ que apunta a la paraula buida calen quan la lletra $\mathtt{x}$ apareix per primera vegada a $\boldsymbol{x}$. Per altra banda, el fet que els tokens continguin la primera lletra que no coincideix és necessari per tal que el descodificador pugui afegir la paraula corresponent al diccionari, de manera que els diccionaris del codificador i el descodificador siguin iguals en cada moment del processat: estiguin sincronitzats.

Algorisme de descodificació. L'entrada és una llista de tokens de la forma $(\imath, \mathtt{x})$. En el procés es construeix un diccionari que és el mateix que en la codificació.

INICIALITZACIÓ: s'inicialitza el text $\boldsymbol{x} \in \mathcal{X}^*$ amb la paraula buida $\boldsymbol{x} = \lambda \in \mathcal{X}^0$ i el diccionari posant-hi només aquesta paraula.

REPETIR: per a cada token $(\imath, \mathtt{x})$,

DESCODIFICACIÓ: S'afegeix a $\boldsymbol{x}$ la paraula $\imath$-èsima del diccionari i després la lletra $\mathtt{x}$.

DICCIONARI: S'actualitza el diccionari afegint la paraula que s'acaba d'afegir a $\boldsymbol{x}$: la paraula $\imath$-èsima del diccionari amb la lletra $\mathtt{x}$ afegida al final.

Mida del diccionari. El nombre de paraules que conté el diccionari és un paràmetre important: d'una banda la codificació binària del nombre $\imath$ requerirà un nombre de bits que en depèn; d'una altra mantenir el diccionari durant la codificació/descodificació consumeix recursos de memòria. Es consideren diverses estratègies per controlar això:

- Augmentar la mida quan s'omple. Es comença fixant la mida del diccionari en per exemple (fins a) 256 paraules, i per tant $\imath$ es codifica amb vuit bits. Cada vegada que

el diccionari s'omple es duplica la seva mida i, a partir d'aleshores, $\imath$ es comença a codificar usant un bit més.

- Deixar d'actualitzar el diccionari un cop està ple i, a partir d'aleshores, treballar només amb les paraules que estan en aquest diccionari fix.
- Esborrar el diccionari quan s'omple i tornar a començar amb un de buit.
- Alliberar espai traient algunes de les paraules que s'hagin usat menys de manera que en puguin entrar de noves.

Fins i tot algunes implementacions controlen la compressió que es va produint durant la codificació i decideixen canvis d'estratègia si és massa pobra.

Variants. El 1984 Terry Welch va proposar la variant *LZW*, que essencialment persegueix eliminar la lletra `x` dels tokens deixant-los reduïts només al punter cap al diccionari. Això presenta dues dificultats: com introduir noves lletres que no havien sortit abans, que en la versió original corresponen als tokens $(\mathtt{0}, \mathtt{x})$, i com sincronitzar el diccionari en la descodificació.

La primera s'evita inicialitzant el diccionari amb totes les paraules de longitud 1, corresponents a totes les lletres de $\mathcal{X}$. Per exemple, en compressors per a fitxers d'ordinador genèrics se sol agafar com a alfabet $\mathcal{X}$ el conjunt dels bytes i aleshores tant compressor com descompressor inicialitzen el diccionari amb les 256 lletres: totes les paraules de longitud 1 (i la paraula buida ja no cal).

La segona es pot solucionar de la manera següent: en descodificar un token s'afegeix al diccionari *provisionalment* la paraula tot just descodificada (que és la mateixa que una paraula anterior del diccionari) i es passa al token següent. La paraula a la qual apunta aquest token pot ser qualsevol, incloent la paraula provisional tot just afegida. Aleshores, abans de descodificar el token, s'afegeix a la paraula provisional *la primera lletra* de la paraula a què apunta el nou token, de manera que el diccionari ja queda sincronitzat, i ara es descodifica normalment.

Aquest és el mètode que fa servir la utilitat `compress` de UNIX, que funciona de la manera següent: l'alfabet són els bytes; el diccionari s'inicialitza fixant la mida en 512 paraules, posant els bytes en les 256 primeres deixant lloc per unes altres 256; al començament els tokens, que són simplement punters al diccionari, es codifiquen amb 9 bits; cada vegada que el diccionari s'omple es duplica la seva mida i a partir d'aleshores cal un bit més per codificar els tokens en binari; quan s'arriba a omplir el diccionari de $2^{16} = 65536$ paraules ja no es duplica més i se segueix amb aquest mateix diccionari fins al final de la compressió/descompressió del fitxer sense afegir-li més paraules; de fet hi ha un control de la compressió assolida que, quan considera que no és prou bona, decideix esborrar el diccionari i tornar a començar amb un de 256 paraules, amb l'esperança que vagi millor.

El format d'imatge *GIF* també fa servir LZW en l'opció de compressió sense pèrdua.

3.6 Problemes Complementaris

3.21. *Codis q-aris òptims.* Sigui X una font que emet M símbols. Sigui $\mathcal{X}$ un alfabet q-ari. Sigui r la solució més petita de la congruència $r \equiv 1 - M \pmod{q-1}$, amb $0 \leqslant r < q - 1$.

Demostreu que existeix algun codi de font q-ari prefix òptim tal que hi ha $q-r$ paraules associades a elements de $\mathcal{X}$ de probabilitat mínima que tenen totes la mateixa longitud i difereixen només en l'última lletra.

3.22. *Codis de Huffman q-aris.* Sigui $\mathcal{X}$ un alfabet q-ari. Sigui X una font que emet M símbols amb probabilitats $p_1 \geqslant p_2 \geqslant \cdots \geqslant p_M$. Sigui r la solució més petita de la congruència $r \equiv 1 - M \pmod{q-1}$, amb $0 \leqslant r < q-1$. Sigui $s = q - r$ i $k = M - s + 1$. Sigui Y la font que emet k símbols amb probabilitats $p_1, \ldots, p_{k-1}$ i $\sum_{i=0}^{s-1} p_{k+i}$. Sigui $\mathcal{C}_Y = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{k-1}, \boldsymbol{c}_k\}$ un codi prefix òptim per a la font Y.

Demostreu que el conjunt

$$\mathcal{C}_X := \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{k-1}, \boldsymbol{c}_k \| \mathtt{a}_1, \boldsymbol{c}_k \| \mathtt{a}_2, \ldots, \boldsymbol{c}_k \| \mathtt{a}_s\}$$

és un codi prefix òptim per a la font X sobre l'alfabet $\mathcal{X}$, on $\mathtt{a}_1, \ldots, \mathtt{a}_s$ són s lletres diferents d'aquest alfabet. A partir d'això comproveu que l'algorisme donat al text per construir codis de Huffman q-aris dona efectivament codis òptims.

3.23. Digueu com es pot simular una variable discreta qualsevol X fent llançaments d'una moneda usant la representació de nombres de l'interval $[0, 1)$ en bases amb pesos.

Expliqueu com simular una variable discreta qualsevol a partir d'una altra.

4 Codificació de canal

En Teoria de la Informació un *canal de comunicació* és un model matemàtic per a la *transmissió* de dades a través d'un medi que pot introduir *soroll*, de manera que les dades enviades es poden rebre amb *errors*. El canal transmet les dades entre un *emissor* i un *receptor*. Situacions típiques en què s'aplica aquest model són:

- *Sistemes de comunicacions* diversos: transmissió guiada per cable *trenat*, cable *coaxial*, *fibra òptica*, etc. o no guiada amb ones de *ràdio*, etc.

  Per exemple en una conversa telefònica dos usuaris, que es van intercanviant el paper d'emissor i receptor, es comuniquen informació de veu; una emissora de ràdio o televisió emet informació d'àudio o video que pot ser captada per molts receptors; en el protocol `http` d'internet client i servidor intercanvien paquets d'informació, que es transmeten entre un i altre passant per múltiples punts d'enllaç de la xarxa.

- *Emmagatzematge* de dades en un disc o cinta magnètica, un disc òptic, una unitat d'estat sòlid, o qualsevol altre dispositiu anàleg. En aquest cas l'emissor és qui introdueix les dades en el dispositiu i el receptor qui les recupera més endavant, una o més vegades. Sovint un mateix usuari del dispositiu fa tots dos papers en moments diferents.

Matemàticament el canal (discret) es modelitza com un dispositiu que admet com a entrada (seqüències de) símbols d'un alfabet finit $\mathcal{X}$ i produeix a la sortida (seqüències de) símbols d'un alfabet finit $\mathcal{Y}$. La sortida depèn de l'entrada de manera probabilista, segons unes determinades probabilitats condicionades $p(\boldsymbol{y}|\boldsymbol{x})$ per a $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X} \times \mathcal{Y}$, les entrades d'una matriu estocàstica $\mathbf{P} = \big[p(\boldsymbol{y}|\boldsymbol{x})\big]$.

L'objectiu és enviar informació lliure d'errors a través del canal: aconseguir que el receptor aconsegueixi determinar exactament quina ha estat la seqüència enviada $\mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_r$ d'entre totes les seqüències possibles que donarien lloc a la seqüència $\mathtt{y}_1\mathtt{y}_2\cdots\mathtt{y}_r$ rebuda a la sortida.

Per usar el canal primer la informació que es vol enviar es codifica amb una cadena de símbols de l'alfabet d'entrada $\mathcal{X}$. Naturalment, és convenient fer això amb una cadena el més curta possible, el qual es pot aconseguir amb un codi de font. Com a resultat les dades a transmetre seran cadenes de símbols de $\mathcal{X}$ que apareixen tots amb la mateixa probabilitat. Cadascun d'ells conté $\log q$ bits d'informació, on $q = |\mathcal{X}|$ és la mida de l'alfabet. Es pot pensar, doncs, que la informació a transmetre està generada per una variable aleatòria $M \sim \mathrm{Unif}(q)$ que pren valors en l'alfabet $\mathcal{X}$. La sortida del canal correspon a una variable aleatòria Y amb valors en l'alfabet $\mathcal{Y}$ i distribució de probabilitat que s'obté com el producte de la distribució de M per la matriu$\mathbf{P}$.

La (esperança de la) informació que conté cada lletra rebuda sobre la lletra enviada és $I(Y;M)$, que en general és un nombre $\leqslant H(M) = \log q$. Per tant la lletra rebuda no conté en general prou informació per saber exactament quina és la lletra enviada.

La solució consisteix en el següent: la seqüència de dades a transmetre es descompon en blocs $\boldsymbol{m} \in \mathcal{X}^k$ i cadascun es codifica amb un bloc $\boldsymbol{c} \in \mathcal{X}^n$ usant un codi de bloc que afegeix redundància a les dades. D'aquesta manera la quantitat d'informació enviada és $k \log q = \log q^k$ i la quantitat d'informació rebuda és $nI(Y;M)$. Per tal que amb la informació

rebuda a la sortida es pugui saber exactament la informació enviada cal que

$$nI(Y;M) \geqslant k \log q \quad \Leftrightarrow \quad \frac{k \log q}{n} \leqslant I(Y;M).$$

Això dona una fita superior per al ratio (binari) d'informació del codi: la informació mútua entre una variable aleatòria uniforme d'entrada i la variable de sortida.

La fita es pot millorar si el codi $\mathcal{C} \subseteq \mathcal{X}^n$ s'agafa de manera que les lletres de les seves paraules segueixin la distribució d'una variable X amb valors en $\mathcal{X}$ que maximitzi el valor de la informació mútua $I(Y;X)$ de la variable de sortida corresponent. Aquest valor màxim de $I(Y;X)$, que només depèn de la matriu $\mathbf{P}$, i que s'assoleix per a una variable d'entrada X, s'anomena capacitat del canal.

Shannon demostra que sí que es pot enviar a través del canal informació lliure d'errors codificant amb codis de ratio igual a la capacitat. La formulació precisa assegura que es pot transmetre informació amb probabilitat d'error tan petita com es vulgui usant codis amb ratio tan a prop com es vulgui de la capacitat. Això sí, la longitud dels codis pot haver de ser molt gran, segons com de petita es vulgui la probabilitat d'error i com de gran es vulgui el ratio (sempre per sota de la capacitat).

4.1 Capacitat de canal

Formalment, en el cas discret i sense memòria, es defineix un canal com:

Definició 4.1 (Canal)**.** *Un* canal *és en una terna* $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ *formada per un* alfabet d'entrada *q-ari* $\mathcal{X} = \{\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_q\}$*, un* alfabet de sortida *r-ari* $\mathcal{Y} = \{\mathtt{y}_1, \mathtt{y}_2, \ldots, \mathtt{y}_r\}$ *i una* matriu de canal

$$\mathbf{P} = \big[p(\mathtt{y}|\mathtt{x})\big]_{(\mathtt{x},\mathtt{y})\in\mathcal{X}\times\mathcal{Y}} = \begin{bmatrix} p(\mathtt{y}_1|\mathtt{x}_1) & p(\mathtt{y}_2|\mathtt{x}_1) & \cdots & p(\mathtt{y}_r|\mathtt{x}_1) \\ \vdots & \vdots & & \vdots \\ p(\mathtt{y}_1|\mathtt{x}_q) & p(\mathtt{y}_2|\mathtt{x}_q) & \cdots & p(\mathtt{y}_r|\mathtt{x}_q) \end{bmatrix} \in \mathrm{Mat}_{q\times r}(\mathbb{R}),$$

que és una matriu estocàstica: té entrades $p(\mathtt{y}|\mathtt{x}) \geqslant 0$ *i la suma de cada fila és igual a* 1*.*

Un canal discret transforma distribucions de probabilitat $\boldsymbol{p} = (p_1, \ldots, p_q)$ sobre l'alfabet d'entrada $\mathcal{X}$ en distribucions de probabilitat $\boldsymbol{q} = (q_1, \ldots, q_r)$ sobre l'alfabet de sortida $\mathcal{Y}$ (ull amb el doble ús de la lletra q en les notacions, per denotar el nombre de lletres de l'alfabet $\mathcal{X}$ i les probabilitats de l'alfabet de sortida, que no hauria de portar a confusió). En forma matricial la transformació ve donada per la identitat $\boldsymbol{q} = \boldsymbol{p} \cdot \mathbf{P}$. Dit d'una altra manera, un canal transforma variables aleatòries X que prenen valors en el conjunt $\mathcal{X}$ en variables aleatòries Y que prenen valors en el conjunt $\mathcal{Y}$. En termes de *fonts*, transforma fonts d'entrada X, que emeten q símbols amb les probabilitats donades per $\boldsymbol{p}$, en fonts de sortida Y, que emeten r símbols amb les probabilitats donades per $\boldsymbol{q}$.

L'única dada essencial, que determina completament el canal, és la matriu $\mathbf{P}$, que dona la dependència de la sortida respecte l'entrada. Els conjunts $\mathcal{X}$ i $\mathcal{Y}$ són en realitat irrellevants i l'únic important són els seus cardinals $|\mathcal{X}| = q$ i $|\mathcal{Y}| = r$, que corresponen al nombre de files i

columnes de la matriu, respectivament. Per tant, moltes vegades es donarà només la matriu $\mathbf{P}$ com a única dada per descriure un canal i es parlarà simplement del "canal de matriu $\mathbf{P}$".

Un canal es fa servir per enviar seqüències d'elements de $\mathcal{X}$: a l'entrada s'envien paraules $\boldsymbol{x} = \mathtt{x}_1\mathtt{x}_2\cdots\mathtt{x}_n \in \mathcal{X}^n$, i a la sortida es reben paraules $\boldsymbol{y} = \mathtt{y}_1\mathtt{y}_2\cdots\mathtt{y}_n \in \mathcal{Y}^n$. En general es consideraran només canals *sense memòria*, en què se suposa que la transmissió de les lletres de $\mathcal{X}$ que formen part d'una seqüència té lloc de manera independent: les probabilitats condicionades que dona la matriu de canal són les mateixes per a cadascuna de les lletres de la seqüència. Això es tradueix en què la probabilitat de rebre una paraula $\boldsymbol{y} = (\mathtt{y}_i) \in \mathcal{Y}^n$ quan s'envia la paraula $\boldsymbol{x} = (\mathtt{x}_i) \in \mathcal{X}^n$ vingui donada pel producte $p(\boldsymbol{y}|\boldsymbol{x}) = \prod_{i=1}^n p(\mathtt{y}_i|\mathtt{x}_i)$.

De manera anàloga a les extensions d'una font això es pot interpretar com una "extensió del canal": consisteix a agafar un altre canal amb variables d'entrada i sortida X^n i Y^n, que són vectors amb n components i.i.d. amb les distribucions de X i de Y, que prenen valors en els conjunts $\mathcal{X}^n$ i $\mathcal{Y}^n$ de les paraules de longitud n, i amb matriu de canal $\big[p(\boldsymbol{y}|\boldsymbol{x})\big]$ per a $(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{X}^n \times \mathcal{Y}^n$ amb entrades $p(\boldsymbol{y}|\boldsymbol{x}) = \prod p(\mathtt{y}_i|\mathtt{x}_i)$.

L'invariant més important d'un canal és la seva:

Definició 4.2 (Capacitat)**.** *La capacitat (o capacitat d'informació, o capacitat teòrica) d'un canal és la màxima informació mútua entre dues variables aleatòries X i Y d'entrada i sortida, agafada sobre totes les possibles variables d'entrada X. O, el que és el mateix, sobre totes les distribucions de probabilitat $\boldsymbol{p}$ en el conjunt $\mathcal{X}$:*

$$C = C(\mathbf{P}) = \max_X \big\{I(X;Y)\big\} = \max_{\boldsymbol{p}} \big\{I(X;Y)\big\}.$$

Observi's que la capacitat d'un canal té associada també una distribució de probabilitat sobre la variable d'entrada, que és amb la qual assoleix aquesta capacitat.

Proposició 4.3. *Propietats de la capacitat de canal:*

1. *està ben definida: existeix el màxim de $I(X;Y)$ sobre totes les distribucions $\boldsymbol{p}$;*
2. *$C \geqslant 0$;*
3. *$C \leqslant \min\big\{\log|\mathcal{X}|, \log|\mathcal{Y}|\big\}$.*

Prova: La capacitat s'ha definit com el màxim de les informacions mútues per a les diferents distribucions de probabilitat per a la variable d'entrada. S'ha de justificar que aquest màxim existeix. D'entrada només es podria assegurar que existeix el suprem, ja que les informacions mútues estan fitades superiorment, per exemple per $I(X;Y) \leqslant H(X) \leqslant \log|\mathcal{X}|$.

1. $I(X;Y)$ és una funció contínua de les components $p(\mathtt{x})$ de la distribució $\boldsymbol{p}$ de la variable X. En efecte, $I(X;Y) = \sum p(\mathtt{x},\mathtt{y}) \log \frac{p(\mathtt{x},\mathtt{y})}{p(\mathtt{x})p(\mathtt{y})}$ és una funció contínua de les probabilitats conjuntes $p(\mathtt{x},\mathtt{y})$ i marginals $p(\mathtt{x})$ i $p(\mathtt{y})$. La marginal $p(\mathtt{y})$ s'obté com el producte $\boldsymbol{p}\cdot\mathbf{P}$, que és una funció contínua de les components de $\boldsymbol{p}$. La conjunta és $p(\mathtt{x},\mathtt{y}) = p(\mathtt{x})p(\mathtt{y}|\mathtt{x})$ i és funció contínua de $p(\mathtt{x})$. Les distribucions $\boldsymbol{p}$ formen un conjunt compacte dins de $\mathbb{R}^n$: el símplex amb vèrtexs els vectors de la base canònica $\boldsymbol{e}_i$. Tota funció contínua sobre un compacte pren un valor màxim.
2. Totes les $I(X;Y)$ són $\geqslant 0$. Per tant el màxim també.

3. $I(X;Y) \leqslant \min\{H(X), H(Y)\} \leqslant \min\{\log|\mathcal{X}|, \log|\mathcal{Y}|\}$ per a tot parell de variables d'entrada i sortida X, Y. Per tant, el màxim també està fitat per aquest nombre. □

Calcular la capacitat d'un canal de matriu $\mathbf{P} = \big[p(\mathtt{y}_j|\mathtt{x}_i)\big]$ consisteix a resoldre un problema d'optimització amb restriccions. Considerant les components de la distribució d'entrada $\boldsymbol{p} = (p_1, \ldots, p_q)$ com a variables que satisfan les restriccions

$$p_i \geqslant 0 \quad \text{per a tot} \quad i = 1, \ldots, q \qquad \text{i} \qquad p_1 + p_2 + \cdots + p_q = 1$$

es tracta de trobar els valors d'aquestes variables que maximitzen el valor de la funció contínua que dona la informació mútua $I(X;Y)$ entre una variable d'entrada amb distribució $\boldsymbol{p}$ i la variable de sortida corresponent amb distribució $\boldsymbol{p} \cdot \mathbf{P}$:

$$I(X,Y) = \sum_{\mathtt{x},\mathtt{y}} p(\mathtt{x},\mathtt{y}) \log \frac{p(\mathtt{x},\mathtt{y})}{p(\mathtt{x})p(\mathtt{y})} = \sum_{i=1}^{q}\sum_{j=1}^{r} p_i\, p(\mathtt{y}_j|\mathtt{x}_i) \log \frac{p(\mathtt{y}_j|\mathtt{x}_i)}{\sum_{k=1}^{q} p_k\, p(\mathtt{y}_j|\mathtt{x}_k)},$$

que és una funció contínua de les q variables p_i en el compacte determinat per les restriccions i derivable en el seu interior.

Per a molts canals d'interès pràctic la capacitat es pot calcular a partir d'una de les dues expressions per a la informació mútua següents:

$$I(X;Y) = H(Y) - H(Y|X) = H(X) - H(X|Y).$$

Si, per exemple, l'entropia condicionada $H(X|Y)$ no depèn de la distribució de X aleshores la capacitat s'obté amb la distribució uniforme per a X, que és la que maximitza la diferència $H(X) - H(X|Y)$ amb valor $\log|\mathcal{X}| - H(X|Y)$. Si, en canvi, l'entropia condicionada $H(Y|X)$ no depèn de la distribució de X, aleshores la capacitat s'obté amb la distribució de X que maximitzi $H(Y)$, que pot ser la uniforme però no sempre ho és; per exemple, per al canal binari asimètric de l'exemple 4.14 aquesta distribució no és la uniforme.

Exemples bàsics de canals. A continuació es veuran alguns exemples de canals en què el càlcul de la capacitat és relativament senzill. Alguns d'ells són models molt acurats de canals físics molt importants en el mon real.

Canal sense soroll. El *canal binari sense soroll* és el canal amb alfabets $\mathcal{X} = \mathcal{Y} = \{\mathtt{0}, \mathtt{1}\}$, o qualsevol altre parell d'alfabets binaris, i matriu de canal

$$\mathbf{P} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = \mathbf{I}_2.$$

En aquest canal la variable de sortida és igual a la d'entrada: $Y = X$. Per tant $H(X|Y) = 0$ i la informació mútua és

$$I(X;Y) = H(X) - H(X|Y) = H(X).$$

El valor màxim d'aquesta entropia s'assoleix agafant la distribució uniforme:

$$X \sim \mathrm{Ber}(\tfrac{1}{2}) \sim (\tfrac{1}{2}, \tfrac{1}{2}), \qquad C(\mathbf{I}_2) = H(X) = 1.$$

En usar aquest canal es pot enviar la informació directament sense codificar aprofitant al màxim la seva capacitat: es rep 1 bit d'informació per a cada símbol binari enviat, ja que el canal no introdueix errors.

Naturalment, per tal d'assolir la capacitat d'un bit per símbol transmès, cal que l'emissor enviï símbols binaris generats per una font uniforme. Si la informació que es vol enviar està generada per una font binària amb una altra distribució, sempre es pot fer primer una codificació de font, que la convertirà en una font uniforme.

Més en general, es poden considerar canals del mateix tipus amb alfabets arbitraris:

Definició 4.4 (Canal sense soroll)**.** *Un* canal sense soroll *és un canal amb el mateix alfabet q-ari $\mathcal{X}$ d'entrada i sortida i la matriu identitat $\mathbf{P} = \mathbf{I}_q$ com a matriu del canal.*

La capacitat d'aquest canal es pot obtenir exactament igual com s'ha fet per al binari:

Proposició 4.5 (Canal sense soroll)**.** *Un canal sense soroll de matriu $\mathbf{I}_q$ té capacitat $\log q$ bits, que s'assoleix amb la distribució uniforme sobre la variable d'entrada.*

Canal amb sortida no superposada. És un canal en què l'entrada depèn completament de la sortida, tot i que cada entrada pot donar lloc a sortides diferents. Equival a dir que $X = g(Y)$: la variable d'entrada és funció de la variable de sortida. Aquesta condició es caracteritza en termes de la matriu de canal $\mathbf{P}$ en el fet que cada columna té com a màxim una component diferent de zero.

Per exemple, siguin $\mathcal{X} = \{\mathtt{0}, \mathtt{1}\}$, $\mathcal{Y} = \{\mathtt{a}, \mathtt{e}, \mathtt{i}, \mathtt{o}, \mathtt{u}\}$ i el canal amb la matriu següent:

$$\mathbf{P} = \begin{bmatrix} \frac{1}{4} & 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{2}{3} & \frac{1}{3} & 0 & 0 \end{bmatrix}.$$

La sortida determina unívocament l'entrada: si es rep una `a`, una `o` o una `u` l'entrada ha estat un `0`, i si es rep una `e` o una `i` l'entrada ha estat un `1`. Aquest canal permet clarament enviar sense errors un bit per símbol transmès, que correspon a la seva capacitat, donada en la proposició 4.7.

Un canal amb $X = g(Y)$, tot i que és un canal amb soroll, ja que un mateix símbol d'entrada es pot rebre a la sortida de maneres diferents, es pot fer servir per enviar informació completament lliure d'errors sense necessitat de codificar: cada símbol de $\mathcal{X}$ que s'envia es recupera a la sortida usant la funció determinista $X = g(Y)$.

Definició 4.6 (Canal amb sortida no superposada)**.** *Un* canal amb sortida no superposada *és un canal en què la variable d'entrada és funció de la variable de sortida: $X = g(Y)$. De manera equivalent, un canal amb matriu $\mathbf{P}$ que té com a màxim una component no nul·la en cada columna.*

Proposició 4.7 (Canal amb sortida no superposada)**.** *La capacitat d'un canal amb sortida no superposada és igual a $\log |\mathcal{X}|$, i s'assoleix amb la distribució uniforme sobre la variable d'entrada.*

PROVA: Com que $X = g(Y)$ es té $H(X|Y) = 0 \Rightarrow I(X;Y) = H(X) - H(X|Y) = H(X)$. La distribució de X que maximitza aquest valor és la uniforme.

El canal de l'exemple donat abans té capacitat $C = \log |\mathcal{X}| = 1$ bit. □

Canal inútil. Els canals inútils són els que no poden transmetre cap informació.

Definició 4.8 (Canal inútil)**.** *Un canal inútil és un en què totes les files de la matriu de canal* $\mathbf{P}$ *són iguals.*

Lema 4.9. *Totes les files de* $\mathbf{P}$ *són iguals si, i només si, tots els parells* X, Y *de variables d'entrada i sortida són independents.*

PROVA: Suposi's que els parells de variables són sempre independents. Agafant una variable d'entrada qualsevol amb totes les probabilitats $p(\mathtt{x}_i)$ no nul·les es té

$$p(\mathtt{y}_j|\mathtt{x}_i) = \frac{p(\mathtt{x}_i, \mathtt{y}_j)}{p(\mathtt{x}_i)} = p(\mathtt{y}_j),$$

de manera que totes les files de $\mathbf{P}$ contenen la distribució de probabilitat de Y.

Recíprocament, suposi's que totes les files de $\mathbf{P}$ són iguals a una certa distribució. Aleshores sigui quina sigui la variable d'entrada X la variable de sortida té sempre la distribució donada per aquestes files comunes:

$$p(\mathtt{y}_j) = \sum_{i=1}^{r} p(\mathtt{x}_i, \mathtt{y}_j) = \sum_{i=1}^{r} p(\mathtt{x}_i) p(\mathtt{y}_j|\mathtt{x}_i) = p(\mathtt{y}_j|\mathtt{x}_i) \sum_{i=1}^{r} p(\mathtt{x}_i) = p(\mathtt{y}_j|\mathtt{x}_i),$$

ja que els valors $p(\mathtt{y}_j|\mathtt{x}_i)$ no depenen de l'índex i. Aleshores, per a tota variable d'entrada,

$$p(\mathtt{x}_i, \mathtt{y}_j) = p(\mathtt{x}_i) p(\mathtt{y}_j|\mathtt{x}_i) = p(\mathtt{x}_i) p(\mathtt{y}_j),$$

de manera que les variables d'entrada i sortida són independents. □

Proposició 4.10 (Canal inútil)**.** *Un canal és inútil si, i només si, té capacitat zero: no pot transmetre gens d'informació.*

PROVA: En efecte, la capacitat és igual a zero si, i només si, per a tot parell de variables d'entrada i sortida la informació mútua és igual a zero, que equival a què aquestes variables siguin independents. □

Canal binari simètric. Aquest és probablement el canal més important ja que serveix com a model de molts sistemes de comunicacions i emmagatzematge de dades de la vida real.

És un canal amb alfabets binaris $\mathcal{X} = \mathcal{Y} = \{\mathtt{0}, \mathtt{1}\}$ i matriu de canal

$$\mathbf{P} = \begin{bmatrix} 1-p & p \\ p & 1-p \end{bmatrix}.$$

És a dir, la probabilitat que el canal enviï un símbol erroni és p, independentment de si aquest símbol és un zero o un u, i la probabilitat que enviï el símbol correctament és $1 - p$.

Es té $H(Y|X=\mathtt{x})=H(p)$ per a tots dos símbols $\mathtt{x}\in\{\mathtt{0},\mathtt{1}\}$ i, per tant, $H(Y|X)=H(p)$. Aleshores $I(X;Y)=H(Y)-H(Y|X)=H(Y)-H(p)$. Aquest valor es maximitza si Y té distribució uniforme, el qual es pot aconseguir agafant la distribució uniforme a X. Per tant, la capacitat del canal binari simètric és

$$C(\mathbf{P})=1-H(p).$$

Aquesta expressió és simètrica respecte $p=\frac{1}{2}$. Sempre es pot suposar que $p\in[0,\frac{1}{2}]$: és més probable que el bit arribi correctament que no pas erròniament. Altrament a la sortida es podrien intercanviar els zeros amb els uns i es tindria un altre canal binari simètric amb aquesta propietat.

La capacitat és màxima, igual a un bit per símbol transmès, si $p=0$ (o 1). En aquest cas és un canal sense soroll. La capacitat és mínima, igual a zero, si $p=\frac{1}{2}$. En aquest cas és un canal inútil.

La situació es pot generalitzar a canals amb alfabets no binaris:

Definició 4.11 (Canal simètric)**.** *Un canal es diu* simètric *si les files de* $\mathbf{P}$ *són permutacions una de l'altra, i el mateix passa amb les seves columnes.*

Proposició 4.12 (Capacitat d'un canal simètric)**.** *La capacitat d'un canal simètric és*

$$C=\log|\mathcal{Y}|-H(\boldsymbol{p})$$

on $\boldsymbol{p}$ *és la distribució de probabilitat de qualsevol fila de la matriu de canal.*

Prova: Com que les files de $\mathbf{P}$ són permutacions les unes de les altres, l'entropia de totes les variables condicionades $H(Y|X=\mathtt{x})$ és la mateixa per a tot $\mathtt{x}\in\mathcal{X}$, igual a $H(\boldsymbol{p})$. Per tant $H(Y|X)=\sum p(\mathtt{x})H(Y|X=\mathtt{x})=\sum p(\mathtt{x})H(\boldsymbol{p})=H(\boldsymbol{p})$.

La informació mútua $I(X;Y)=H(Y)-H(Y|X)=H(Y)-H(\boldsymbol{p})$ es maximitza quan Y té la distribució uniforme.

Agafant X amb la distribució uniforme, les probabilitats de cada valor de Y són la probabilitat d'un valor de X multiplicat per la suma de la columna de $\mathbf{P}$ corresponent. Com que les columnes són permutacions les unes de les altres han de sumar sempre el mateix, i es dedueix que les probabilitats dels valors de Y són totes iguals: Y té la distribució uniforme

Per tant la capacitat del canal és $C=\log|\mathcal{X}|-H(\boldsymbol{p})$ i s'assoleix amb la distribució uniforme a l'entrada. □

Es diu *canal dèbilment simètric* un canal amb matriu $\mathbf{P}$ tal que les seves files són permutacions les unes de les altres i les seves columnes sumen totes el mateix, però no cal que siguin permutacions les unes de les altres. Aquests canals es comporten exactament igual que els canals simètrics. Per exemple, els canals de matrius

$$\mathbf{P}=\begin{bmatrix}\frac{1}{5}&\frac{1}{5}&\frac{3}{10}&\frac{3}{10}\\[2pt]\frac{3}{10}&\frac{3}{10}&\frac{1}{5}&\frac{1}{5}\end{bmatrix}\qquad\text{o}\qquad\mathbf{P}=\begin{bmatrix}\frac{1}{3}&\frac{1}{6}&\frac{1}{2}\\[2pt]\frac{1}{3}&\frac{1}{2}&\frac{1}{6}\end{bmatrix}$$

són simètric i dèbilment simètric, respectivament.

Canal binari amb esborralls. És un canal amb $\mathcal{X} = \{\mathtt{0}, \mathtt{1}\}$, $\mathcal{Y} = \{\mathtt{0}, \mathtt{1}, \mathtt{e}\}$ i matriu

$$\mathbf{P} = \begin{bmatrix} 1-p & 0 & p \\ 0 & 1-p & p \end{bmatrix}.$$

Si es rep un zero o un u, aleshores se sap segur que aquest és el símbol enviat, però si rep el símbol `e` (esborrall) no es pot saber res sobre què s'ha enviat: tant zero com u tenen la mateixa probabilitat d'haver estat enviats.

Proposició 4.13 (Canal binari amb esborralls)**.** *La capacitat d'un canal binari amb esborrall de probabilitat p és*

$$H = 1 - p,$$

la qual s'assoleix amb la distribució uniforme sobre la variable d'entrada.

Prova: Es té $H(Y|X = \mathtt{x}) = H(p)$ independentment de quin sigui $\mathtt{x} \in \{\mathtt{0}, \mathtt{1}\}$. Per tant es té $H(Y|X) = H(p)$ i es dedueix que $I(X;Y) = H(Y) - H(Y|X) = H(Y) - H(p)$. Per maximitzar això s'ha de maximitzar $H(Y)$ però no està clar que es pugui aconseguir obtenir la distribució uniforme sobre Y a partir d'alguna distribució sobre X. De fet, no es pot. Sigui $\pi = \Pr(X = 0)$. La distribució de probabilitats de Y és

$$\big(\pi(1-p), (1-\pi)(1-p), p\big)$$

i la seva entropia és

$$H\big(\pi(1-p), (1-\pi)(1-p), p\big) = H(p) + (1-p)H(\pi) + pH(1)$$

que pren el valor màxim igual a $H(p) + 1 - p$ quan X té la distribució uniforme: $\pi = \frac{1}{2}$, que dona $H(\pi) = 1$ i $H(Y) = H(p) + 1 - p$.

Es dedueix que la capacitat del canal és $C = H(p) + 1 - p - H(p) = 1 - p$ bits per símbol transmès, que s'assoleix per a la distribució uniforme sobre la l'entrada. □

Aquest valor per a la capacitat concorda amb la intuïció: una proporció $1 - p$ dels bits transmesos arriba correctament; en canvi, els altres p es perden completament.

El canal amb esborralls es pot generalitzar a un alfabet d'entrada qualsevol $\mathcal{X}$, amb alfabet de sortida $\mathcal{Y} = \mathcal{X} \sqcup \{\mathtt{e}\}$, i la matriu $\mathbf{P}$ obtinguda afegint a la matriu $(1-p)\mathbf{I}_{|\mathcal{X}|}$ una columna que conté p en totes les components. En aquest cas general es pot veure anàlogament que la capacitat és $(1-p)\log|\mathcal{X}|$, i que s'assoleix també amb la distribució uniforme sobre la variable d'entrada.

Canal binari asimètric. És un canal binari que es comporta de manera diferent per als dos símbols d'entrada. Un cas extrem és l'exemple que es veurà a continuació, en què el zero es transmet sempre correctament i, en canvi, l'u es transmet de la pitjor manera possible:

Exemple 4.14 (Canal binari asimètric)**.** *El canal binari asimètric de matriu*

$$\mathbf{P} = \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

té capacitat $C = \log\frac{5}{4} \approx 0.322$, que assoleix amb la distribució $X \sim (\frac{3}{5}, \frac{2}{5})$.

PROVA: Sigui $X \sim (q,p)$ la distribució de probabilitats de la variable d'entrada X, que és una Bernoulli amb $\Pr(X = \mathtt{1}) = p$. Usant la matriu de canal es calcula la distribució corresponent de la variable de sortida Y com el producte de matrius:

$$Y \sim \begin{bmatrix} q & p \end{bmatrix} \begin{bmatrix} 1 & 0 \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} = \begin{bmatrix} q + \frac{p}{2} & \frac{p}{2} \end{bmatrix}.$$

L'entropia relativa és

$$H(Y|X) = q\, H(Y|X = \mathtt{0}) + p\, H(Y|X = \mathtt{1}) = q \cdot 0 + p \cdot 1 = p$$

i l'entropia de la variable Y és

$$H(Y) = H\left(q + \frac{p}{2}, \frac{p}{2}\right) = H\left(\frac{p}{2}\right).$$

Per tant $I(X;Y) = H(\frac{p}{2}) - p$. Per maximitzar aquest valor es considera com una funció de $p \in [0,1]$ i s'iguala la derivada a zero:

$$I(X;Y)' = -\left[\frac{1}{2}\log\frac{p}{2} + \frac{p}{2}\frac{\log e}{\frac{p}{2}}\frac{1}{2}\right] - \left[\frac{-1}{2}\log\left(1 - \frac{p}{2}\right) + \left(1 - \frac{p}{2}\right)\frac{\log e}{1 - \frac{p}{2}}\frac{-1}{2}\right] - 1$$
$$= \frac{1}{2}\log\frac{1 - \frac{p}{2}}{\frac{p}{2}} - 1 = 0.$$

Això equival a què

$$\log\frac{1 - \frac{p}{2}}{\frac{p}{2}} = 2 \Leftrightarrow \frac{1 - \frac{p}{2}}{\frac{p}{2}} = 4 \Leftrightarrow 1 - \frac{p}{2} = 4\frac{p}{2} \Leftrightarrow 5\frac{p}{2} = 1 \Leftrightarrow p = \frac{2}{5}.$$

Per tant la capacitat del canal s'assoleix quan la variable X té distribució $X \sim (\frac{3}{5}, \frac{2}{5})$. A aquesta distribució de X li correspon la distribució de sortida $Y \sim (\frac{4}{5}, \frac{1}{5})$. La distribució de probabilitat conjunta del parell (X,Y) és

$p(x,y)$	0	1	$p(x)$
0	$\frac{3}{5}$	0	$\frac{3}{5}$
1	$\frac{1}{5}$	$\frac{1}{5}$	$\frac{2}{5}$
$p(y)$	$\frac{4}{5}$	$\frac{1}{5}$	1

La capacitat del canal corresponent a aquesta distribució és:

$$I(X;Y) = H\left(\frac{2/5}{2}\right) - \frac{2}{5} = H\left(\frac{1}{5}\right) - \frac{2}{5} = \log 5 - 2 \approx 0.321928 \quad \square$$

Problemes

4.1. Calculeu, per a cada $a \in \mathbb{R}$, la capacitat d'un canal amb entrada X que pren valors en $\mathcal{X} = \{0, 1\} \subset \mathbb{R}$ i sortida $Y = X + Z$ que pren valors en un subconjunt $\mathcal{Y} \subset \mathbb{R}$, on Z és la variable que pren valors en $\mathcal{Z} = \{0, a\} \subset \mathbb{R}$ amb distribució uniforme.

4.2. Es considera un canal que transforma una variable d'entrada X amb valors a $\mathcal{X} = \mathbb{Z}_{12}$ en una variable de sortida Y amb valors en el mateix alfabet a través de la igualtat $Y = X + Z \pmod{12}$, on Z representa el "soroll" i és una variable (independent de X) que pren valors en el subconjunt $\mathcal{Z} = \{0, 1, 2\} \subset \mathbb{Z}_{12}$ amb distribució uniforme.

1. Digueu quina és la matriu d'aquest canal i quina propietat important té com a matriu de canal.
2. Calculeu la capacitat del canal i per a quina distribució de X s'assoleix.
3. Digueu com es pot usar el canal per assolir la capacitat amb codis de qualsevol longitud de ratio igual a la capacitat i probabilitat d'error igual a zero.
4. Discutiu la situació si s'agafen variables X, Y amb valors a $\mathbb{Z}_{13}$ amb la mateixa relació entre elles, però ara vista mòdul 13.
5. Perquè canals d'aquest tipus es coneixen amb el nom de "màquina d'escriure amb soroll" (noisy typewriter)?

4.3. *Canal amb soroll additiu.* Calculeu la capacitat d'un canal amb alfabets d'entrada i sortida $\mathcal{X} = \mathcal{Y} = \mathbb{Z}_q$, el conjunt dels nombres enters mòdul q, amb variables d'entrada i sortida relacionades de la forma $Y = X + Z$ per a una variable Z que també pren valors dins de $\mathbb{Z}_q$ (independent de les altres) que modela el soroll.

4.4. *Concatenació de canals.* Siguin $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ i $(\mathcal{Y}, \mathbf{Q}, \mathcal{Z})$ dos canals amb l'alfabet de sortida del primer igual a l'alfabet d'entrada del segon. Es poden enviar dades (seqüències de símbols de l'alfabet $\mathcal{X}$) a través dels dos canals successivament.

Comproveu que el resultat és equivalent a considerar un únic canal $(\mathcal{X}, \mathbf{PQ}, \mathcal{Z})$ amb matriu el producte de les dues matrius.

4.5. Es construeix un canal que consisteix en una concatenació de n canals binaris simètrics (vegeu problema **4.4**), tots amb la mateixa probabilitat d'error p.

Comproveu que el resultat també és un canal binari simètric amb probabilitat d'error

$$p_n = \frac{1 - (1 - 2p)^n}{2}.$$

Quina capacitat té aquesta concatenació de n canals quan n creix?

4.6. *Canal binari asimètric.* Estudieu el canal binari asimètric de matriu

$$\mathbf{P} = \begin{bmatrix} 1 & 0 \\ 1 - p & p \end{bmatrix}.$$

4.7. Es considera un canal ternari amb alfabets d'entrada i sortida $\mathcal{X} = \{\mathtt{x}_1, \mathtt{x}_2, \mathtt{x}_3\}$ i matriu de canal determinada per la taula:

$p(\mathtt{y}\|\mathtt{x})$	$\mathtt{x}_1$	$\mathtt{x}_2$	$\mathtt{x}_3$
$\mathtt{x}_1$	$\frac{2}{3}$	$\frac{1}{3}$	0
$\mathtt{x}_2$	$\frac{1}{3}$	$\frac{1}{3}$	$\frac{1}{3}$
$\mathtt{x}_3$	0	$\frac{1}{3}$	$\frac{2}{3}$

1. Calculeu la seva capacitat i digueu per a quina distribució de la variable d'entrada s'assoleix.
2. Interpreteu el fet que la distribució que assoleix la capacitat té $\Pr(X = \mathtt{x}_2) = 0$ per comparar aquest canal amb un altre tipus de canal ben conegut.

4.8. Es considera un canal binari on es produeixen errors (amb probabilitat p) i també esborralls (amb probabilitat ϵ).

1. feu un diagrama que representi el canal;
2. digueu quina és la matriu del canal;
3. calculeu la seva capacitat;
4. compareu-la amb les capacitats dels casos $p = 0$ (canal binari amb esborrall de probabilitat ϵ) i $\epsilon = 0$ (canal binari simètric amb probabilitat d'error p);
5. digueu quina és la capacitat per als valors $p = \frac{1}{4}$ i $\epsilon = \frac{1}{4}$.

4.9. Calculeu la capacitat d'un canal ternari amb esborrall de probabilitat α: un canal amb alfabet d'entrada $\{\mathtt{1}, \mathtt{2}, \mathtt{3}\}$, alfabet de sortida $\{\mathtt{1}, \mathtt{2}, \mathtt{3}, \mathtt{e}\}$ i matriu determinada per la taula

$p(\mathtt{y}\|\mathtt{x})$	1	2	3	e
1	$1-\alpha$	0	0	α
2	0	$1-\alpha$	0	α
3	0	0	$1-\alpha$	α

i digueu per a quina distribució de probabilitat d'entrada s'assoleix aquesta capacitat.

4.2 Codis de canal

L'objectiu de la *codificació de canal* és transmetre informació de manera fiable a través d'un canal sorollós que introdueix errors en les dades transmeses. Per aconseguir això es codifica la informació amb un codi de bloc sobre l'alfabet d'entrada, que permeti corregir tants errors com sigui possible, entre els que s'hagin produït durant la transmissió. La descripció formal de la situació és la següent:

Definició 4.15 (Codi de canal)**.** *Sigui* $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ *un canal. S'anomena* codi de canal *un parell* $(\mathcal{C}, \mathrm{ccdec})$ *format per un codi de bloc* $\mathcal{C} \subseteq \mathcal{X}^n$ *i una aplicació de descodificació* $\mathrm{ccdec}\colon \mathcal{Y}^n \to \mathcal{C}$.

En molts s'agafa una aplicació de descodificació seguint un criteri (esquema de decisió) que no depèn de quin sigui el codi $\mathcal{C}$ concret sinó només del canal: la matriu $\mathbf{P}$ i, potser, la distribució de probabilitat d'entrada. S'acostuma donar només el codi $\mathcal{C}$ per donar un codi de canal, tot i que sempre l'aplicació de descodificació estarà implícita en el context.

També es pot considerar una *descodificació incompleta* $\mathrm{ccdec}\colon \mathcal{Y}^n \to \mathcal{C} \sqcup \{*\}$, en què l'aplicació de descodificació pot prendre un valor $*$ que no és una paraula codi, que s'interpreta com que s'accepta un error en la transmissió que no es pot corregir.

Observi's que donar una aplicació de descodificació $\mathrm{ccdec}\colon \mathcal{Y}^n \to \mathcal{C}$ per a un codi de M paraules $\mathcal{C} = \{\boldsymbol{c}_1, \dots, \boldsymbol{c}_M\}$ equival a donar una partició de $\mathcal{Y}^n$ en M subconjunts disjunts $\mathcal{D}_i$, de la manera següent: a cada $\boldsymbol{c}_i \in \mathcal{C}$ li correspon un subconjunt $\mathcal{D}_i \subseteq \mathcal{Y}^n$, de manera que $\mathrm{ccdec}(\boldsymbol{y}) = \boldsymbol{c}_i \Leftrightarrow \boldsymbol{y} \in \mathcal{D}_i$; o sigui, $\mathcal{D}_i = \mathrm{ccdec}^{-1}(\boldsymbol{c}_i)$. A partir d'una família qualsevol de subconjunts disjunts $\mathcal{D}_i \subseteq \mathcal{Y}^n$ es pot definir una descodificació assignant a cada paraula $\boldsymbol{y} \in \mathcal{D}_i$ la paraula codi $\boldsymbol{c}_i$ i declarant un error en les paraules $\boldsymbol{y} \in \mathcal{Y}^n \smallsetminus \sqcup_{i=1}^M \mathcal{D}_i$ que no pertanyen a cap dels $\mathcal{D}_i$ (descodificació incompleta) o bé simplement descodificant aquestes paraules amb una paraula codi arbitrària agafada aleatòriament.

Un codi format per $M = |\mathcal{C}|$ paraules de longitud n es dirà *de tipus* (n, M).

El ratio d'un codi q-ari s'ha definit com $R = \frac{\log_q M}{n}$. Com que en l'estudi dels codis de canal el ratio s'ha de comparar amb la capacitat del canal, que se sol mesurar en bits, aquí és més natural és considerar el *ratio binari* $R = \frac{\log M}{n}$, calculat amb el logaritme en base 2. El ratio binari és un nombre de l'interval $[0, \log q]$ i el ratio q-ari pertany a $[0, 1]$. La relació entre tots dos s'obté multiplicant o dividint per $\log q$. En tota aquesta secció el ratio d'un codi es referirà sempre a ratio binari, si no es diu el contrari.

Naturalment, per a l'eficiència de la transmissió cal que el ratio sigui el més gran possible.

El ratio i el cardinal es relacionen a través de la identitat $M = 2^{nR}$.

Codificació de canal. Els codis de canal es fan servir per corregir errors de transmissió que introdueixen els canals amb soroll, de la manera següent:

- La informació que s'ha d'enviar es tradueix en seqüències d'elements d'un conjunt de missatges $\mathcal{M}$. Els elements de $\mathcal{M}$ es codifiquen amb un codi de canal $\mathcal{C} \subseteq \mathcal{X}^n$ mitjançant una aplicació de codificació bijectiva $\mathrm{enc}\colon \mathcal{M} \to \mathcal{C}$.
- Es considera que els missatges els emet una font d'informació, de manera que tenen associades probabilitats. Això es tradueix en què cada paraula de $\mathcal{C}$ s'haurà de transmetre a través del canal d'acord amb unes certes probabilitats.
- Es denotarà S la variable aleatòria corresponent, que pren valors en $\mathcal{C}$. En alguns arguments se suposa que S té distribució uniforme. Quan aquesta no sigui la distribució de probabilitat natural de la font de missatges $\mathcal{M}$ es pot aconseguir que ho sigui passant prèviament la informació per una codificació de font.
- En transmetre una paraula $\boldsymbol{c} \in \mathcal{C}$ es rebrà a la sortida del canal una paraula $\boldsymbol{y} \in \mathcal{Y}^n$, d'acord amb les probabilitats $\Pr(Y^n = \boldsymbol{y} | S = \boldsymbol{c}) = \prod p(\mathtt{y}_i | \mathtt{c}_i)$ determinades per la matriu de canal $\mathbf{P}$.
- El receptor descodifica la paraula rebuda usant l'aplicació ccdec associada al codi de canal, i assumeix que $\widehat{\boldsymbol{c}} = \mathrm{ccdec}(\boldsymbol{y}) \in \mathcal{C}$ és la paraula que s'havia enviat.

- Es recupera el missatge enviat com $\widehat{\boldsymbol{m}} = \mathrm{dec}(\widehat{\boldsymbol{c}})$, on $\mathrm{dec}\colon \mathcal{C} \to \mathcal{M}$ és la inversa de enc.
- La comunicació s'ha produït amb èxit si $\widehat{\boldsymbol{c}} = \boldsymbol{c} \Leftrightarrow \widehat{\boldsymbol{m}} = \boldsymbol{m}$ i hi ha hagut error en la transmissió del missatge $\boldsymbol{m} \in \mathcal{M}$ en cas contrari: quan $\widehat{\boldsymbol{c}} \neq \boldsymbol{c}$.

En molts casos la informació s'escriu com una cadena de símbols del mateix alfabet d'entrada; per exemple, l'alfabet binari. En aquesta situació la codificació es fa de la manera següent: la cadena de $\mathcal{X}^*$ que representa la informació es descompon en blocs de longitud k, que fan el paper de missatges: els elements de $\mathcal{M}$. Cada bloc de $\mathcal{M} = \mathcal{X}^k$ de longitud k es codifica amb un bloc de longitud n usant un codi de bloc de tipus $[n, k]$ (o (n, M) amb $M = q^k$). amb una codificació $\mathrm{enc}\colon \mathcal{X}^k \to \mathcal{C} \subseteq \mathcal{X}^n$, que és la que s'envia a través del canal.

D'aquesta manera per a cada k símbols d'informació se'n transmeten n a través del canal, amb ratio (q-ari) $\frac{k}{n}$. El ratio binari ve donat per $\frac{\log q^k}{n} = \frac{k}{n} \log q$, amb un factor $\log q$ respecte al ratio q-ari $\frac{k}{n}$.

El teorema de codificació de canal diu que existeixen codis amb qualsevol ratio que es vulgui, sempre que sigui inferior a la capacitat del canal, que permeten transmetre la informació (essencialment) lliure d'errors. Això sí, pagant el preu d'haver de treballar amb codis de longituds n prou grans, depenent aquesta mida de com de prop de la capacitat es vol arribar i de com de petita es vol que sigui la probabilitat d'error.

Esquemes de decisió. És clar que en l'aplicació dels codis de canal a la correcció d'errors l'aplicació de descodificació juga un paper essencial. S'anomenen *esquemes de decisió* diferents criteris que es poden usar per construir aquest tipus d'aplicacions de descodificació. Els més habituals i importants són els següents:

- *màxima versemblança* (*maximum likelihood*): la paraula $\boldsymbol{c} = \mathrm{mldec}(\boldsymbol{y})$ maximitza la probabilitat condicionada de rebre la paraula $\boldsymbol{y}$ suposant que s'ha enviat aquesta paraula $\boldsymbol{c}$:
$$\mathrm{mldec}(\boldsymbol{y}) = \arg\max_{\boldsymbol{c}\in\mathcal{C}} \Pr(Y^n = \boldsymbol{y} | S = \boldsymbol{c}).$$
- observador ideal (*ideal observer*) o *màxim a posteriori*: la paraula $\boldsymbol{c} = \mathrm{iodec}(\boldsymbol{y})$ maximitza la probabilitat condicionada que s'hagi enviat aquesta paraula suposant que s'ha rebut la paraula $\boldsymbol{y}$:
$$\mathrm{iodec}(\boldsymbol{y}) = \arg\max_{\boldsymbol{c}\in\mathcal{C}} \Pr(S = \boldsymbol{c} | Y^n = \boldsymbol{y}).$$
- *proximitat* (*nearest neighbour*): la paraula $\boldsymbol{c} = \mathrm{nndec}(\boldsymbol{y})$ minimitza la seva distància al codi:
$$\mathrm{nndec}(\boldsymbol{y}) = \arg\min_{\boldsymbol{c}\in\mathcal{C}} d(\boldsymbol{y}, \boldsymbol{c}).$$

La descodificació per màxima versemblança està definida per a qualsevol canal, independentment de la probabilitat de l'alfabet d'entrada: només depèn de la matriu $\mathbf{P}$ del canal.

La descodificació per observador ideal depèn no només de la matriu del canal sinó també de la distribució de probabilitat sobre els missatges que es transmeten.

La descodificació per proximitat només es pot aplicar quan els alfabets d'entrada i sortida són el mateix, ja que es consideren distàncies entre paraules $\boldsymbol{c} \in \mathcal{C} \subseteq \mathcal{X}^n$ i $\boldsymbol{y} \in \mathcal{Y}^n$ de

tots dos alfabets, que no estarien definides si els alfabets fossin diferents. Apart d'aquest requisit, la descodificació per proximitat no depèn de cap distribució de probabilitats: ni de les probabilitats de canal (matriu $\mathbf{P}$) ni de les de la variable d'entrada.

Proposició 4.16. *Si la distribució sobre els missatges és la uniforme, l'observador ideal coincideix amb l'esquema de decisió per màxima versemblança.*

PROVA: Si la distribució dels missatges és equiprobable cada paraula codi es fa servir amb probabilitat $\Pr(S = \boldsymbol{c}) = \frac{1}{M}$. Aleshores,

$$p(\boldsymbol{c}|\boldsymbol{y}) = \frac{p(\boldsymbol{c},\boldsymbol{y})}{p(\boldsymbol{y})} = \frac{p(\boldsymbol{y}|\boldsymbol{c})p(\boldsymbol{c})}{p(\boldsymbol{y})} = \frac{p(\boldsymbol{y}|\boldsymbol{c})}{Mp(\boldsymbol{y})}.$$

Per tant,

$$\max_{\boldsymbol{c}\in\mathcal{C}}\{p(\boldsymbol{c}|\boldsymbol{y})\} = \max_{\boldsymbol{c}\in\mathcal{C}}\left\{\frac{p(\boldsymbol{y}|\boldsymbol{c})}{Mp(\boldsymbol{y})}\right\} = \frac{1}{Mp(\boldsymbol{y})}\max_{\boldsymbol{c}\in\mathcal{C}}\{p(\boldsymbol{y}|\boldsymbol{c})\}.$$

De manera que, per a cada paraula $\boldsymbol{y} \in \mathcal{Y}^n$ rebuda, una paraula $\boldsymbol{c} \in \mathcal{C}$ maximitza el valor $p(\boldsymbol{c}|\boldsymbol{y})$ si, i només si, maximitza el valor $p(\boldsymbol{y}|\boldsymbol{c})$. □

Proposició 4.17. *En un canal binari simètric amb probabilitat d'error $p \leqslant \frac{1}{2}$ l'esquema de decisió per proximitat coincideix amb l'esquema per màxima versemblança.*

PROVA: Suposi's que s'envia la paraula $\boldsymbol{c} \in \mathcal{C}$ i es rep la paraula $\boldsymbol{y} \in \{\mathtt{0},\mathtt{1}\}^n$. La probabilitat que passi això és:

$$\Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}) = \prod p(y_i|x_i) = p^d(1-p)^{n-d} = (1-p)^n\left(\frac{p}{1-p}\right)^d, \quad \text{amb} \;\; d = d(\boldsymbol{x},\boldsymbol{c}).$$

Com que $p \leqslant \frac{1}{2} \Leftrightarrow 1-p \geqslant \frac{1}{2}$ es té $\frac{p}{1-p} \leqslant 1$. Per tant la potència $\left(\frac{p}{1-p}\right)^d$ decreix amb la distància d. Per a tota altra paraula del codi $\boldsymbol{c}' \in \mathcal{C}$ es té

$$d(\boldsymbol{y},\boldsymbol{c}) < d(\boldsymbol{y},\boldsymbol{c}') \quad \Leftrightarrow \quad \Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}) > \Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}'),$$

i la paraula codi que minimitza la distància a $\boldsymbol{y}$ és la que maximitza la versemblança. □

Com que en el canal binari simètric la distribució que assoleix la capacitat és la uniforme, quan es fa servir aquesta distribució d'entrada en el canal binari simètric tots tres esquemes de decisió coincideixen.

Definició 4.18 (Probabilitats d'error)**.** *Donat un codi de canal $\mathcal{C} \subseteq \mathcal{X}^n$ amb aplicació de descodificació* ccdec$\colon \mathcal{Y}^n \to \mathcal{C}$ *es defineixen les probabilitats d'error següents:*

- *probabilitat d'error en transmetre una paraula del codi $\boldsymbol{c} \in \mathcal{C}$:*

$$\mathscr{E}(\mathcal{C}|S = \boldsymbol{c}) = \Pr\left(\widehat{\boldsymbol{c}} \neq \boldsymbol{c}\right) = \sum_{\substack{\boldsymbol{y}\in\mathcal{Y}^n \\ \mathrm{ccdec}(\boldsymbol{y})\neq\boldsymbol{c}}} p(\boldsymbol{y}|\boldsymbol{c});$$

- *probabilitat d'error màxima:*

$$\mathscr{E}_{\max}(\mathcal{C}) = \max_{\boldsymbol{c}\in\mathcal{C}} \mathscr{E}(\mathcal{C}|S=\boldsymbol{c});$$

- *probabilitat d'error mitjana:*

$$\mathscr{E}_{\mathrm{av}}(\mathcal{C}) = \frac{1}{M}\sum_{\boldsymbol{c}\in\mathcal{C}} \mathscr{E}(\mathcal{C}|S=\boldsymbol{c}).$$

4.3 Teorema de codificació de canal

El teorema de codificació de canal va ser enunciat i demostrat per Shannon en el seu article fundacional [27]. Assegura que existeixen codis de canal amb qualsevol ratio inferior a la capacitat del canal, tals que la probabilitat màxima d'error en fer-los servir es pot fet tan petita com es vulgui. Això té una contrapartida: la longitud d'aquests codis pot haver de ser molt gran per assegurar que la probabilitat d'error sigui menor que un $\epsilon > 0$ prefixat. El teorema té un recíproc, que diu que fent servir codis amb ratio superior a la capacitat del canal mai es podrà aconseguir que la probabilitat d'error sigui tan petita com es vulgui. A continuació es donen enunciats precisos d'aquestes afirmacions.

Teorema 4.19 (Teorema de codificació de canal)**.** *Sigui $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ un canal de capacitat C. Siguin R un nombre $R < C$ i ϵ un nombre $\epsilon > 0$.*

Aleshores existeix un enter $N \geqslant 1$ tal que per a tot $n \geqslant N$ existeixen codis de canal $\mathcal{C}_n \subseteq \mathcal{X}^n$ de longitud n i ratio $R(\mathcal{C}_n) \geqslant R$ amb probabilitat màxima d'error $\mathscr{E}_{\max}(\mathcal{C}_n) < \epsilon$.

Es donarà una demostració del teorema basada Ash [1, pags. 69-72]. Es comença amb un parell de lemes. El primer és simplement una aplicació directa de la llei dels grans nombres. El segon és on hi ha l'important de la demostració: dona una construcció de codis que, agafant els paràmetres adequats, són els que permetran acabar la demostració.

Lema 4.20. *Siguin X i Y variables d'entrada i sortida per a un canal $(\mathcal{X}, \boldsymbol{P}, \mathcal{Y})$. Per a cada $n \geqslant 1$ i nombre real a es defineix l'esdeveniment següent, en relació al parell (X^n, Y^n):*

$$\mathcal{A}_a^{(n)} = \left\{ (\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X}^n \times \mathcal{Y}^n : \log \frac{p(\boldsymbol{y}|\boldsymbol{x})}{p(\boldsymbol{y})} > na \right\} \subseteq \mathcal{X}^n \times \mathcal{Y}^n.$$

Aleshores, per a tot $a < I(X;Y)$ es té $\lim_{n\to\infty} \Pr\left(\mathcal{A}_a^{(n)}\right) = 1$.

Prova: És una aplicació directa de la llei dels grans nombres. Com que se suposa que el canal és sense memòria les variables X^n i Y^n són vectors $(X_1, \ldots, X_n)$ i $(Y_1, \ldots, Y_n)$ amb components independents que tenen la mateixa distribució que X i Y.

Descomponent les probabilitats per components es té

$$\log \frac{p(\boldsymbol{y}|\boldsymbol{x})}{p(\boldsymbol{y})} = \log \prod_{i=1}^{n} \frac{p(\mathtt{y}_i|\mathtt{x}_i)}{p(\mathtt{y}_i)} = \sum_{i=1}^{n} \log \frac{p(\mathtt{y}_i|\mathtt{x}_i)}{p(\mathtt{y}_i)}.$$

Siguin Z_i les variables $\log \frac{p(Y_i|X_i)}{p(Y_i)}$ i siguin $\bar{Z}_n = \frac{1}{n}\sum_{i=1}^n Z_i$ les variables que donen les seves mitjanes aritmètiques. La probabilitat del conjunt de l'enunciat és:

$$\Pr\left(\mathcal{A}_a^{(n)}\right) = \Pr(\bar{Z}_n > a).$$

Les Z_i són i.i.d. amb esperança la informació mútua $I(X;Y)$, ja que

$$I(X;Y) = \mathbb{E}_{(X,Y)}\left[\log \frac{p(X,Y)}{p(X)p(Y)}\right] = \mathbb{E}_{(X,Y)}\left[\log \frac{p(Y|X)}{p(Y)}\right].$$

La llei dels grans nombres assegura que $\Pr\left(\left|\bar{Z}_n - I(X;Y)\right| \leqslant \epsilon\right) \to 1$ per a tot $\epsilon > 0$.

Per tant, $\Pr\left(\bar{Z}_n > I(X;Y) - \epsilon\right) \to 1$ per a tot $\epsilon > 0$ i agafant $\epsilon = I(X;Y) - a > 0$ es dedueix el límit de l'enunciat. □

Lema 4.21 ([1, Lemma 3.5.2])**.** *Siguin X i Y variables d'entrada i sortida per a un canal $(\mathcal{X}, \boldsymbol{P}, \mathcal{Y})$. Siguin $n \geqslant 1$, $a \in \mathbb{R}$ i $\mathcal{A}_a^{(n)}$ com al lema 4.20.*

Per a cada enter $M \geqslant 1$ tal que el nombre $\epsilon_n := M2^{-na} + 1 - \Pr\left(\mathcal{A}_a^{(n)}\right)$ sigui $\epsilon_n < 1$ existeix un codi de bloc $\mathcal{C} \subseteq \mathcal{X}^n$ amb $|\mathcal{C}| = M$ i

$$\mathscr{E}_{\max}(\mathcal{C}) \leqslant \epsilon_n.$$

Prova: Per a cada $n \geqslant 1$ es denotaran X^n i Y^n les variables vectorials, amb valors en $\mathcal{X}^n$ i $\mathcal{Y}^n$, que tenen components i.i.d. amb les distribucions de X i Y.

Per a cada $\boldsymbol{x} \in \mathcal{X}^n$ es denotarà $\mathcal{A}_{\boldsymbol{x}}$ l'esdeveniment següent, en relació a la variable $\mathcal{Y}$:

$$\mathcal{A}_{\boldsymbol{x}} = \left\{\boldsymbol{y} \in \mathcal{Y}^n : (\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{A}_a^{(n)}\right\} \subseteq \mathcal{Y}^n.$$

És una "llesca" del conjunt $\mathcal{A}_a^{(n)}$ del lema 4.20, que s'obté en fixar el valor de la primera coordenada igual a $\boldsymbol{x} \in \mathcal{X}^n$.

La demostració es farà donant un algorisme amb el qual es construeix un codi de longitud n format per $M' \geqslant M$ paraules $\mathcal{C}' = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_{M'}\} \subseteq \mathcal{X}^n$, junt amb una aplicació de descodificació ccdec: $\mathcal{Y}^n \to \mathcal{C}'$ per al qual es compleix la fita $\mathscr{E}_{\max}(\mathcal{C}') \leqslant \epsilon_n$. La descodificació ccdec es definirà dient quins són els subconjunts $\mathcal{D}_i := \text{ccdec}^{-1}(\boldsymbol{c}_i)$ per a cada índex i.

A partir d'aquest codi s'obté immediatament un codi $\mathcal{C}$ com el que demana l'enunciat simplement quedant-se amb només M paraules qualssevol, per exemple les M primeres, i canviant la descodificació a una descodificació incompleta, assumint errors en els elements de $\mathcal{Y}^n$ que anaven a paraules de $\mathcal{C}'$ descartades, o bé a una descodificació completa assignant a aquests elements paraules de $\mathcal{C}$ de qualsevol manera arbitraria.

El codi es construirà amb un algorisme que comença amb un codi buit al qual es van afegint paraules $\boldsymbol{c}_k$ i, per a cadascuna d'elles, es diu quin és el conjunt $\mathcal{D}_k \subseteq \mathcal{Y}^n$ corresponent. Durant tot el procés es denotarà $\mathcal{D}$ la reunió dels $\mathcal{D}_i$ corresponents a les paraules codi que es tinguin en aquell moment: per a construir la paraula $\boldsymbol{c}_k$ es farà servir $\mathcal{D} = \sqcup_{i=1}^{k-1}\mathcal{D}_i$.

Es comença amb un codi buit, que no conté cap paraula, i amb $\mathcal{D} = \varnothing \subset \mathcal{X}^n$. Sigui k un comptador que indica l'índex de la propera paraula que s'ha de construir, que s'inicialitza amb el valor $k = 1$. L'algorisme és el següent:

REPETIR: Mentre sigui possible,

AFEGIR PARAULA $\boldsymbol{c}_k$: Sigui $\boldsymbol{x} \in \mathcal{X}^n$ una paraula qualsevol (si existeix) tal que

$$\Pr\left(Y^n \in \mathcal{A}_{\boldsymbol{x}} \setminus \mathcal{D} \middle| X^n = \boldsymbol{x}\right) > 1 - \epsilon_n.$$

S'agafa aquesta paraula com a nova paraula codi: $\boldsymbol{c}_k = \boldsymbol{x}$.

DEFINIR $\mathcal{D}_k$: Es defineix $\mathcal{D}_k = \mathcal{A}_{\boldsymbol{c}_k} \setminus \mathcal{D}$ i s'actualitza $\mathcal{D}$ afegint-li $\mathcal{D}_k$. D'aquesta manera quedarà $\mathcal{D} = \sqcup_{i=1}^{k} \mathcal{D}_i$.

COMPTADOR: S'incrementa el comptador en una unitat i es torna al primer pas per provar d'afegir una nova paraula.

L'algorisme acabarà quan en el primer pas ja no existeixi cap paraula $\boldsymbol{x}$ amb

$$\Pr\left(Y^n \in \mathcal{A}_{\boldsymbol{x}} \setminus \mathcal{D} \middle| X^n = \boldsymbol{x}\right) > 1 - \epsilon_n.$$

Observi's que, a priori, això podria passar fins i tot en el primer pas, de manera que el "codi" construït fos buit. Això no és possible ja que es veurà que en acabar l'algorisme el codi conté $M' \geqslant M \geqslant 1$ paraules.

Les noves paraules que es van afegint al codi no poden ser mai una de les anteriors ja que, en el primer pas, si s'agafa una $\boldsymbol{x} = \boldsymbol{c}_i$ amb $i < k$ es té $\mathcal{A}_{\boldsymbol{x}} = \mathcal{A}_{\boldsymbol{c}_i} \subseteq \mathcal{D}$. Aleshores el conjunt $\mathcal{A}_{\boldsymbol{x}} \setminus \mathcal{D}$ és el conjunt buit i no pot tenir una probabilitat positiva $> 1 - \epsilon_n$.

Segur que l'algorisme acaba en un nombre finit de passos ja que el conjunt $\mathcal{X}^n$ té un nombre finit de paraules, i en cada pas s'afegeix una paraula nova al codi $\mathcal{C}'$.

Sigui $M' = |\mathcal{C}'|$ el nombre de paraules del codi que s'obté un cop s'ha acabat l'execució de l'algorisme. S'ha de demostrar que $M' \geqslant M$. Per fer-ho es fitarà la probabilitat $\Pr\left(\mathcal{A}_a^{(n)}\right)$ obtenint-se una desigualtat que relaciona aquest nombre M' amb ϵ_n. Es té:

$$\begin{aligned}\Pr\left(\mathcal{A}_a^{(n)}\right) &= \sum_{(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y}) = \sum_{\boldsymbol{x} \in \mathcal{X}^n} p(\boldsymbol{x}) \sum_{\boldsymbol{y} \in \mathcal{A}_{\boldsymbol{x}}} p(\boldsymbol{y}|\boldsymbol{x}) \\ &= \sum_{\boldsymbol{x} \in \mathcal{X}^n} p(\boldsymbol{x}) \sum_{\boldsymbol{y} \in \mathcal{A}_{\boldsymbol{x}} \cap \mathcal{D}} p(\boldsymbol{y}|\boldsymbol{x}) + \sum_{\boldsymbol{x} \in \mathcal{X}^n} p(\boldsymbol{x}) \sum_{\boldsymbol{y} \in \mathcal{A}_{\boldsymbol{x}} \cap \mathcal{D}^c} p(\boldsymbol{y}|\boldsymbol{x}).\end{aligned}$$

El primer d'aquests sumatoris es pot fitar de la manera següent:

$$\begin{aligned}\sum_{\boldsymbol{x} \in \mathcal{X}^n} p(\boldsymbol{x}) \sum_{\boldsymbol{y} \in \mathcal{A}_{\boldsymbol{x}} \cap \mathcal{D}} p(\boldsymbol{y}|\boldsymbol{x}) &\leqslant \sum_{\boldsymbol{x} \in \mathcal{X}^n} p(\boldsymbol{x}) \sum_{\boldsymbol{y} \in \mathcal{D}} p(\boldsymbol{y}|\boldsymbol{x}) = \sum_{\boldsymbol{x} \in \mathcal{X}^n} \sum_{\boldsymbol{y} \in \mathcal{D}} p(\boldsymbol{x},\boldsymbol{y}) \\ &= \sum_{\boldsymbol{y} \in \mathcal{D}} \sum_{\boldsymbol{x} \in \mathcal{X}^n} p(\boldsymbol{x},\boldsymbol{y}) = \sum_{\boldsymbol{y} \in \mathcal{D}} p(\boldsymbol{y}) = \Pr(\mathcal{D}) = \sum_{k=1}^{M'} \Pr(\mathcal{D}_k) \leqslant \sum_{k=1}^{M'} \Pr(\mathcal{A}_{\boldsymbol{c}_k}),\end{aligned}$$

on la primera desigualtat es compleix perquè $\mathcal{A}_{\boldsymbol{x}} \cap \mathcal{D} \subseteq \mathcal{D}$ i la darrera perquè $\mathcal{D}_k \subseteq \mathcal{A}_{\boldsymbol{c}_k}$.

Aleshores

$$\boldsymbol{y} \in \mathcal{A}_{\boldsymbol{c}_k} \;\Leftrightarrow\; (\boldsymbol{c}_k, \boldsymbol{y}) \in \mathcal{A}_a^{(n)} \quad \Rightarrow \quad \log \frac{p(\boldsymbol{y}|\boldsymbol{c}_k)}{p(\boldsymbol{y})} > na \;\Leftrightarrow\; p(\boldsymbol{y}) < p(\boldsymbol{y}|\boldsymbol{c}_k) 2^{-na},$$

i d'aquí es dedueix que

$$\Pr(\mathcal{A}_{\boldsymbol{c}_k}) = \sum_{\boldsymbol{y}\in\mathcal{A}_{\boldsymbol{c}_k}} p(\boldsymbol{y}) \leqslant \sum_{\boldsymbol{y}\in\mathcal{A}_{\boldsymbol{c}_k}} p(\boldsymbol{y}|\boldsymbol{c}_k)2^{-na} \leqslant 2^{-na} \sum_{\boldsymbol{y}\in\mathcal{Y}^n} p(\boldsymbol{y}|\boldsymbol{c}_k) = 2^{-na}.$$

Per tant, el primer sumatori queda fitat per $M'2^{-na}$.

Com que l'algorisme s'ha acabat, en el primer pas no es pot afegir cap paraula més. Per tant, per a tota $\boldsymbol{x} \in \mathcal{X}^n$ ha de ser

$$\Pr\left(Y^n \in \mathcal{A}_{\boldsymbol{x}} \setminus \mathcal{D} \middle| X^n = \boldsymbol{x}\right) = \sum_{\boldsymbol{y}\in\mathcal{A}_{\boldsymbol{x}}\cap\mathcal{D}^c} p(\boldsymbol{y}|\boldsymbol{x}) \leqslant 1-\epsilon_n,$$

i s'obté la fita següent per al segon sumatori:

$$\sum_{\boldsymbol{x}\in\mathcal{X}^n} p(\boldsymbol{x}) \sum_{\boldsymbol{y}\in\mathcal{A}_{\boldsymbol{x}}\cap\mathcal{D}^c} p(\boldsymbol{y}|\boldsymbol{x}) \leqslant \sum_{\boldsymbol{x}\in\mathcal{X}^n} p(\boldsymbol{x})(1-\epsilon_n) = 1-\epsilon_n.$$

Usant totes dues fites s'obté, finalment, la fita:

$$\Pr\left(\mathcal{A}_a^{(n)}\right) \leqslant M'2^{-na} + 1 - \epsilon_n.$$

A partir d'això es dedueix la desigualtat:

$$M2^{-na} + 1 - \Pr\left(\mathcal{A}_a^{(n)}\right) = \epsilon_n \leqslant M'2^{-na} + 1 - \Pr\left(\mathcal{A}_a^{(n)}\right) \quad \Rightarrow \quad M \leqslant M'.$$

Per tant en l'algorisme s'ha creat un codi $\mathcal{C}'$ que conté $M' \geqslant M$ paraules. Agafant el codi $\mathcal{C}$ com el format per les primeres M paraules $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_M$ de $\mathcal{C}'$ (o qualsevol subconjunt de M paraules) i l'aplicació de descodificació $\mathcal{Y}^n \to \mathcal{C}$ determinada pels conjunts $\mathcal{D}_i$ corresponents, s'obté un codi que satisfà les condicions de l'enunciat: conté M paraules i la probabilitat d'error en enviar cadascuna és

$$\mathscr{E}(\mathcal{C}|S=\boldsymbol{c}_i) = \Pr(Y^n \notin \mathcal{D}_i) = 1 - \Pr(\mathcal{D}_i) \leqslant 1 - (1-\epsilon_n) = \epsilon_n$$

per a tota paraula codi $\boldsymbol{c}_i$, de manera que $\mathscr{E}_{\max}(\mathcal{C}) \leqslant \epsilon_n$. □

Ara, agafant un valor de a que permeti usar el lema 4.20 i per al qual les fites ϵ_n del lema 4.21 tendeixin a zero es demostra el teorema de codificació de canal.

Prova: Sigui C la capacitat del canal i sigui R un ratio (binari) amb $0 < R < C$. Per a cada $n \geqslant 1$ s'agafa el nombre $M = \lceil 2^{nR} \rceil$, que satisfà $2^{nR} \leqslant M < 2^{nR} + 1$. Els codis de tipus (n, M) tenen ratio $\frac{\log_2 M}{n} \geqslant R$. Sigui $a = \frac{1}{2}(R+C)$, que satisfà $R < a < C$.

S'agafa la variable d'entrada X amb la distribució que assoleix la capacitat del canal, de manera que $I(X;Y) = C$. Aleshores, com que $a < I(X;Y)$ el lema 4.20 assegura que $\lim_{n\to\infty} \Pr\left(\mathcal{A}_a^{(n)}\right) = 1$. La fita d'error màxim dels codis del lema 4.21 satisfà:

$$\epsilon_n = M2^{-na} + 1 - \Pr\left(\mathcal{A}_a^{(n)}\right) < 2^{n(R-a)} + 2^{-na} + 1 - \Pr\left(\mathcal{A}_a^{(n)}\right)$$

i aquest nombre tendeix a zero quan $n \to \infty$ ja que $R - a < 0$, $a > 0$ i la probabilitat del conjunt $\mathcal{A}_a^{(n)}$ tendeix a 1.

Aleshores, donat un $\epsilon > 0$ qualsevol, que es pot suposar < 1, es té $\epsilon_n \leqslant \epsilon$ per a tots els n prou grans. Per a aquests n, com que $\epsilon_n < 1$, el lema 4.21 assegura que existeixen codis de tipus (n, M) amb probabilitat màxima d'error $\leqslant \epsilon$. □

Teorema 4.22 (Recíproc del teorema de codificació de canal)**.** *Sigui $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ un canal de capacitat C. Sigui R un nombre $R > C$. Existeix un nombre $\epsilon > 0$ tal que tot codi de canal $\mathcal{C}$ de ratio $\geqslant R$ té probabilitat d'error $\mathscr{E}_{\max}(\mathcal{C}) > \epsilon$.*

Aquest enunciat del recíproc és la *versió dèbil*. El que diu és que amb codis de ratio per damunt de la capacitat del canal mai es podrà fer la probabilitat màxima d'error tan petita com es vulgui. Hi ha una versió més forta d'aquest enunciat que assegura que agafant codis amb ratio per damunt de la capacitat la probabilitat màxima d'error sempre tendirà a 1 quan els codis es van agafant de longitud cada vegada més gran.

Teorema 4.23 (Recíproc en versió forta)**.** *Sigui $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ un canal de capacitat C. Sigui R un nombre $R > C$. Per a tota successió de codis de canal $(\mathcal{C}_i)_{i\geqslant 1}$ de longituds $n(\mathcal{C}_i) \to \infty$ i ratios $R(\mathcal{C}_i) \geqslant R$ es té $\lim_{i\to\infty} \mathscr{E}_{\max}(\mathcal{C}_i) = 1$.*

Ratio assolible. Els teoremes es poden enunciar també en termes del concepte de

Definició 4.24 (Ratio assolible)**.** *Un ratio R es diu* assolible *en un canal $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ si existeix una successió $\mathcal{C}_n$ de codis de longitud n i ratio $R(\mathcal{C}_n) \geqslant R$ tals que la probabilitat màxima d'error tendeix a zero: $\lim_{n\to\infty} \mathscr{E}_{\max}(\mathcal{C}_n) = 0$.*

Aleshores el teorema de codificació de canal i el seu recíproc es poden enunciar dient que:

Teorema 4.25. *Tot ratio que sigui inferior a la capacitat del canal és assolible i cap ratio que sigui superior a la capacitat del canal ho és.*

Reducció de l'error màxim a l'error mitjà. Moltes demostracions del teorema de codificació de canal donen aquest resultat per a l'error mitjà i no per a l'error màxim. En realitat les afirmacions per als dos tipus d'error són equivalents. Com que $\mathscr{E}_{\text{av}}(\mathcal{C}) \leqslant \mathscr{E}_{\max}(\mathcal{C})$ està clar que si l'enunciat és cert per a l'error màxim també ho és per a l'error mitjà. L'afirmació recíproca es demostra a la següent:

Proposició 4.26. *Suposi's que el teorema 4.19 és cert per a la probabilitat d'error mitjana dels codis. Aleshores també és cert per a la probabilitat d'error màxima.*

Prova: Donat un ratio $R < C$ (estrictament) inferior a la capacitat del canal es considera una altre ratio R' amb $R < R' < C$. Donat un $\epsilon > 0$ s'aplica el teorema de codificació de canal amb probabilitat d'error mitjana al ratio R' i al nombre $\epsilon' = \frac{\epsilon}{2}$. Sigui N' l'enter corresponent que dona el teorema de codificació de canal per a R' i ϵ'.

Per a cada $n \geqslant N'$ sigui $\mathcal{C}'_n$ un codi de tipus (n, M') amb ratio $\geqslant R'$ tal que la probabilitat mitjana d'error és $\mathscr{E}_{\text{av}}(\mathcal{C}'_n) \leqslant \epsilon' = \frac{\epsilon}{2}$, i sigui $\text{ccdec}'_n \colon \mathcal{Y}^n \to \mathcal{C}'_n$ l'aplicació de descodificació corresponent.

Sigui $\mathcal{C}_n$ el codi de tipus (n, M) format per les paraules $\boldsymbol{c} \in \mathcal{C}'_n$ que tenen probabilitat d'error $\mathscr{E}(\mathcal{C}'_n | S = \boldsymbol{c}) \leqslant \epsilon$. Per a aquests codis s'agafa l'aplicació de descodificació amb $\text{ccdec}_n(\boldsymbol{x}) = \text{ccdec}'_n(\boldsymbol{x})$ sempre que $\text{ccdec}'_n(\boldsymbol{x}) \in \mathcal{C}_n$ i definint $\text{ccdec}_n(\boldsymbol{x})$ arbitràriament altrament. És clar que $\mathscr{E}(\mathcal{C}_n | S = \boldsymbol{c}) = \mathscr{E}(\mathcal{C}'_n | S = \boldsymbol{c}) \leqslant \epsilon$ per a tota paraula $\boldsymbol{c} \in \mathcal{C}_n$, i per tant aquest codi té probabilitat màxima d'error $\mathscr{E}_{\max}(\mathcal{C}_n) \leqslant \epsilon$.

El codi $\mathcal{C}_n$ ha de contenir almenys la meitat de les paraules del codi $\mathcal{C}'_n$; o sigui, $M \geqslant \frac{M'}{2}$. En efecte, si no fos així aleshores $\mathcal{C}'_n$ contindria més de la meitat de paraules $\boldsymbol{c}$ amb error $\mathscr{E}(\mathcal{C}'_n|S = \boldsymbol{c}) > \frac{\epsilon}{2}$ de manera que $\mathscr{E}_{\text{av}}(\mathcal{C}'_n) = \frac{1}{M'}\mathscr{E}(\mathcal{C}'_n|S = \boldsymbol{c}) > \epsilon$, en contradicció amb la hipòtesi. Observi's que aquí s'està suposant que totes les paraules codi es fan servir amb la mateixa probabilitat: la variable S té distribució uniforme.

El ratio del codi $\mathcal{C}_n$ és $\frac{\log M}{n} \geqslant \frac{\log M'-1}{n} \geqslant R' - \frac{1}{n}$. Aquest ratio és $\geqslant R$ si, i només si, $n \geqslant \frac{1}{R'-R}$. Per tant agafant $N = \max(N', \frac{1}{R'-R})$ els codis $\mathcal{C}_n$ satisfan la condició del teorema amb probabilitat màxima per a tot $n \geqslant N$. □

4.4 Mètode probabilistic

El *mètode probabilistic* consisteix a demostrar l'existència d'un objecte veient que la probabilitat que existeixi és positiva. La demostració original de Shannon del teorema de codificació de canal és un exemple paradigmàtic d'aquesta tècnica. Moltes altres demostracions que han anat apareixent després segueixen aquesta mateixa idea.

L'argument és el següent: donat un canal $(\mathcal{X}, \mathbf{P}, \mathcal{Y})$ de capacitat C i un ratio R amb $0 < R < C$,

- S'agafa una variable d'entrada X amb la distribució que assoleix la capacitat del canal i la variable de sortida Y corresponent. Aquestes distribucions determinen distribucions en els vectors X^n i Y^n, amb components independents, per a tot $n \geqslant 1$.
- Es fixa un esquema de decisió que es pugui fer servir per definir aplicacions de descodificació ccdec: $\mathcal{Y}^n \to \mathcal{C}$ per a tots els codis $\mathcal{C} \subseteq \mathcal{X}^n$. Per exemple, la descodificació per màxima versemblança, observador ideal o proximitat. En la demostració original es fa servir la descodificació per seqüències típiques, que es veurà també més endavant.
- Fixat un enter $n \geqslant 1$, s'agafa $M = \lceil 2^{nR} \rceil$ i es consideren tots els codis de bloc $\mathcal{C}$ de tipus (n, M). Sigui Γ_n una variable aleatòria que pren com a valors aquests codis, amb les probabilitats agafades de manera que cada paraula de $\mathcal{X}^n$ forma part d'algun codi agafat per la variable Γ_n segons la probabilitat que li correspon respecte la variable X^n.

  La probabilitat d'error mitjà $\mathscr{E}_{\text{av}}(\Gamma_n)$ és una variable aleatòria, funció de Γ_n, que pren valors a l'interval $[0, 1]$.
- Es troben fites ϵ_n per a l'esperança d'aquestes variables

$$\mathbb{E}\big[\mathscr{E}_{\text{av}}(\Gamma_n)\big] \leqslant \epsilon_n \qquad \text{tals que} \quad \lim_{n\to\infty} \epsilon_n = 0.$$

  Aquest és el punt clau de la demostració, on hi ha la dificultat.
- Aleshores, per a cada $\epsilon > 0$ existeix un N tal que $\mathbb{E}\big[\mathscr{E}_{\text{av}}(\Gamma_n)\big] < \epsilon$ per a tot $n \geqslant N$. Es dedueix que per a cada $n \geqslant N$ hi ha d'haver algun codi $\mathcal{C}$ de tipus (n, M), és a dir, algun valor de la variable Γ_n, amb $\mathscr{E}_{\text{av}}(\mathcal{C}) < \epsilon$, ja que altrament l'esperança de la variable Γ_n hauria de ser $\geqslant \epsilon$.

Aquesta és una demostració no constructiva, en el sentit que no explica com es pot trobar directament un codi amb les propietats que es demanen. És clar que es sempre podrien

agafar tots els codis d'un tipus (n, M) donat, i anar-los examinant un per un fins a trobarne un amb probabilitat mitjana d'error $< \epsilon$. Aquesta idea es pot traslladar a un algorisme probabilistic de Monte-Carlo eficient. Tot i això, la natura aleatòria dels codis que es trobarien d'aquesta manera els faria poc adequats per fer-los servir a la pràctica, per falta de mètodes de codificació i descodificació eficients.

Els codis útils a la pràctica tenen estructura addicional que permet usar àlgebra lineal (les paraules són vectors), aritmètica de polinomis, teoria de grafs i altres eines matemàtiques per dissenyar algorismes eficients de codificació, descodificació i correcció d'errors.

Codis: conjunts o famílies. En realitat, les demostracions no es fan usant codis de canal de la manera explicada fins ara, ja que la distribució de probabilitats de la variable Γ_n és massa complicada. En comptes d'això es treballa amb una versió més general del concepte de codi de canal, que considera els codis no com a subconjunts, sinó com a famílies (o tuples) de paraules de $\mathcal{X}^n$. D'aquesta manera la variable Γ_n pren valors en el conjunt $(\mathcal{X}^n)^M$ i té una distribució de probabilitats més senzilla: és un vector aleatori de M components independents, totes amb la distribució de X^n.

En tota la resta d'aquesta secció s'anomenarà "codi de canal" de tipus (n, M) a una M-tupla $\mathcal{C} = (\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_M) \subseteq (\mathcal{X}^n)^M$ amb components $\boldsymbol{c}_i \in \mathcal{X}^n$, que poden estar repetides, junt amb una aplicació de descodificació $\mathrm{ccdec}\colon \mathcal{Y}^n \to \{1, 2, \ldots, M\}$, que ara assigna a cada paraula $\boldsymbol{y} \in \mathcal{Y}^n$ un índex i, el qual indica en quina paraula codi $\boldsymbol{c}_i$ es descodifica la paraula $\boldsymbol{y}$ rebuda.

Valors de l'índex diferents $i \neq j$ poden correspondre a una mateixa paraula si $\boldsymbol{c}_i = \boldsymbol{c}_j$. Es considerarà un error de transmissió quan s'envia la paraula codi $\boldsymbol{c}_i$, d'índex i, i el descodificador produeix un índex $j \neq i$, fins i tot encara que si la paraula $\boldsymbol{c}_j$ sigui igual a la paraula $\boldsymbol{c}_i$.

Els mateixos esquemes de decisió considerats per a codis segueixen tenint sentit i es poden usar per definir aplicacions de descodificació.

Els codis com a subconjunts es poden veure com el cas particular de les tuples que tenen totes les components diferents. Amb aquesta generalització del concepte de codi en realitat no s'obté una versió afeblida del teorema de codificació de canal, sinó una versió totalment equivalent. En efecte, d'una banda, com que els subconjunts són un cas particular de les tuples, és clar que si el teorema és cert per a codis-subconjunt també ho és per a codis-tupla. De l'altra, el recíproc també es cert, tal i com es demostra a la següent:

Proposició 4.27. *Sigui $\mathcal{C}$ una M-tupla amb descodificació* $\mathrm{ccdec}\colon \mathcal{Y}^n \to \mathcal{C}$. *Aleshores existeix una M-tupla sense repeticions (un codi de veritat) $\mathcal{C}'$ i una aplicació de descodificació* $\mathrm{ccdec}'\colon \mathcal{Y}^n \to \mathcal{C}'$ *amb la mateixa probabilitat mitjana d'error $\mathscr{E}_{\mathrm{av}}(\mathcal{C}') = \mathscr{E}_{\mathrm{av}}(\mathcal{C})$.*

Prova: Naturalment, tot i que hi poden haver tuples de qualsevol nombre de components, se suposa que sempre s'agafen tuples de tipus (n, M) amb $M \leqslant |\mathcal{X}|^n$, de manera que hi ha prou paraules diferents a $\mathcal{X}^n$ per poder convertir-les en una altra tupla de la mateixa mida M amb totes les components diferents.

La demostració s'obté aplicant les vegades que calgui la construcció següent, que "separa" dues components repetides en una M-tupla donada.

Siguin $\boldsymbol{c}_r = \boldsymbol{c}_s$, amb $r \neq s$, dues components de la tupla $\mathcal{C}$ amb la mateixa paraula. Sigui $\boldsymbol{c} \in \mathcal{X}^n$ una paraula que no és cap de les $\boldsymbol{c}_i$ de la tupla $\mathcal{C}$ (aquí és on cal que $M \leqslant q^n = |\mathcal{X}^n|$).

Es considera la nova tupla $\mathcal{C}'$ amb components $\boldsymbol{c}'_i = \boldsymbol{c}_i$ per a $i \neq s$ i $\boldsymbol{c}'_s = \boldsymbol{c}$. O sigui, es canvia una de les paraules repetides per una que no surt a la tupla. En el cas de les tuples les aplicacions de descodificació donen l'índex de la component corresponent. Es defineix l'aplicació de descodificació per a $\mathcal{C}'$ posant $\mathrm{ccdec}'(\boldsymbol{y}) = \mathrm{ccdec}(\boldsymbol{y}) = i$ si $i \neq s$ i $\mathrm{ccdec}'(\boldsymbol{y}) = r$ si $\mathrm{ccdec}(\boldsymbol{y}) = s$. En particular, observi's que la nova aplicació de descodificació no pren mai el valor s, corresponent a la nova paraula $\boldsymbol{c}$, de manera que en transmetre aquesta paraula sempre es produirà un error.

Aleshores, per a cada índex $i \notin \{r, s\}$ es té $\mathscr{E}(\mathcal{C}|S = \boldsymbol{c}_i) = \mathscr{E}(\mathcal{C}'|S = \boldsymbol{c}_i)$ i, per a les altres dues paraules en les components r i s-èsimes es té

$$\begin{aligned}
&\mathscr{E}(\mathcal{C}'|S = \boldsymbol{c}'_r) + \mathscr{E}(\mathcal{C}'|S = \boldsymbol{c}'_s) \\
&\qquad = \sum_{\substack{\boldsymbol{y}\in\mathcal{Y}^n \\ \mathrm{ccdec}'(\boldsymbol{y})\neq r}} \Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}'_r) + \sum_{\substack{\boldsymbol{y}\in\mathcal{Y}^n \\ \mathrm{ccdec}'(\boldsymbol{y})\neq s}} \Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}'_s) \\
&\qquad\qquad = 1 - \sum_{\substack{\boldsymbol{y}\in\mathcal{Y}^n \\ \mathrm{ccdec}'(\boldsymbol{y})= r}} \Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}_r) + 1 \\
&\qquad = 1 - \sum_{\substack{\boldsymbol{y}\in\mathcal{Y}^n \\ \mathrm{ccdec}(\boldsymbol{y})= r}} \Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}_r) - \sum_{\substack{\boldsymbol{y}\in\mathcal{Y}^n \\ \mathrm{ccdec}(\boldsymbol{y})= s}} \Pr(Y^n = \boldsymbol{y}|S = \boldsymbol{c}_r) + 1 \\
&\qquad\qquad = \mathscr{E}(\mathcal{C}|S = \boldsymbol{c}_r) + \mathscr{E}(\mathcal{C}|S = \boldsymbol{c}_s),
\end{aligned}$$

ja que $\boldsymbol{c}_s = \boldsymbol{c}_r$. Per tant en calcular l'error mitjà es compensen l'un amb l'altre els errors condicionats a emetre les paraules r i s-èsimes de les dues tuples.

Observi's que en aquest argument se suposa que la probabilitat d'enviar una paraula de la tupla no depèn del seu índex. O sigui, que la variable S té distribució uniforme. □

Probabilitat d'error. Sigui S la variable aleatòria que diu amb quina freqüència s'ha d'enviar cada missatge. En aquest cas en què el codi és una tupla es pot considerar que S emet els índexs i de les paraules codi, és a dir, enters de l'interval $1 \leqslant i \leqslant M$.

Sigui i un índex fixat. Es consideren els codis de canal $\mathcal{C}$ que continguin la paraula $\boldsymbol{x}$ en la component i-èsima. Es denotarà $\mathscr{E}(i, \boldsymbol{x}, \boldsymbol{y})$ la probabilitat d'error en usar un d'aquests codis, enviar la paraula i-èsima $\boldsymbol{x}$ i rebre $\boldsymbol{y}$ a la sortida. És la probabilitat del conjunt de codis:

$$\big\{\mathcal{C} : \boldsymbol{c}_i = \boldsymbol{x}, \mathrm{ccdec}(\boldsymbol{y}) \neq i\big\} \subseteq (\mathcal{X}^n)^M.$$

Observi's que l'aplicació de descodificació ccdec depèn no només de la paraula $\boldsymbol{x} = \boldsymbol{c}_i$ enviada i la paraula $\boldsymbol{y}$ rebuda, sinó de totes les altres paraules del codi $\mathcal{C}$. De fet es podria usar la notació $\mathrm{ccdec}_{\mathcal{C}}$ per emfasitzar aquesta dependència, però s'ha optat per simplificar les notacions.

La probabilitat d'aquest conjunt, respecte la distribució de la variable Γ_n que genera tots els codis, és la següent:

$$\mathscr{E}(i, \boldsymbol{x}, \boldsymbol{y}) = \sum_{\substack{\mathcal{C}, \boldsymbol{c}_i = \boldsymbol{x} \\ \mathrm{ccdec}(\boldsymbol{y})\neq i}} \Pr(\Gamma_n = \mathcal{C}).$$

Per a qualssevol dos índexs i i j aquestes probabilitats coincideixen: $\mathscr{E}(i, \boldsymbol{x}, \boldsymbol{y}) = \mathscr{E}(j, \boldsymbol{x}, \boldsymbol{y})$. En efecte, la permutació $(\mathcal{X}^n)^M \to (\mathcal{X}^n)^M$ que intercanvia les paraules en les components indicades pels índexs és una bijecció, que conserva les probabilitats dels codis, i també el fet que en la descodificació es produeixi un error, i per tant la probabilitat d'error en tots dos casos, que és la suma de les probabilitats dels codis, és la mateixa.

Es denotarà $\mathscr{E}(\boldsymbol{x}, \boldsymbol{y})$ aquesta probabilitat comuna. És la probabilitat que, en enviar la paraula $\boldsymbol{x}$ i rebre la paraula $\boldsymbol{y}$, es produeixi un error, usant algun dels codis de canal que tingui $\boldsymbol{x}$ en alguna de les seves components. El lema següent redueix el càlcul de l'esperança de l'error mitjà sobre tots els codis, que s'ha de calcular per demostrar el teorema de codificació de canal, a l'esperança dels errors $\mathscr{E}(\boldsymbol{x}, \boldsymbol{y})$ en variar l'entrada i la sortida:

Lema 4.28. *Sigui $\mathscr{E}(\boldsymbol{x}, \boldsymbol{y})$ la probabilitat d'error en transmetre una paraula $\boldsymbol{x} \in \mathcal{X}^n$ i rebre la paraula $\boldsymbol{y} \in \mathcal{Y}^n$ usant un codi de canal de tipus (n, M). Aleshores*

$$\mathbb{E}\big[\mathscr{E}_{\text{av}}(\Gamma_n)\big] = \mathbb{E}\big[\mathscr{E}(X^n, Y^n)\big].$$

Prova: Es denotarà $p(i) = \Pr(S = i)$. Es calcula:

$$\begin{aligned}
\mathbb{E}\big[\mathscr{E}_{\text{av}}(\Gamma)\big] &= \sum_{\mathcal{C}} p(\mathcal{C})\mathscr{E}_{\text{av}}(\mathcal{C}) = \sum_{\mathcal{C}} p(\mathcal{C}) \sum_{i=1}^{M} p(i)\mathscr{E}(\mathcal{C}|S = \boldsymbol{c}_i) \\
&= \sum_{i=1}^{M} p(i) \sum_{\boldsymbol{x}\in\mathcal{X}^n} \sum_{\mathcal{C}} p(\boldsymbol{x})p(\mathcal{C}|\boldsymbol{c}_i = \boldsymbol{x}) \sum_{\substack{\boldsymbol{y}\in\mathcal{Y}^n \\ \text{ccdec}(\boldsymbol{y})\neq i}} p(\boldsymbol{y}|\boldsymbol{x}) \\
&= \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}^n\times\mathcal{Y}^n} p(\boldsymbol{x})p(\boldsymbol{y}|\boldsymbol{x}) \sum_{i=1}^{M} p(i) \sum_{\substack{\mathcal{C},\boldsymbol{c}_i=\boldsymbol{x} \\ \text{ccdec}(\boldsymbol{y})\neq i}} p(\mathcal{C}) \\
&= \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}^n\times\mathcal{Y}^n} p(\boldsymbol{x},\boldsymbol{y}) \sum_{i=1}^{M} p(i)\mathscr{E}(i, \boldsymbol{x}, \boldsymbol{y}) \\
&= \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}^n\times\mathcal{Y}^n} p(\boldsymbol{x},\boldsymbol{y})\mathscr{E}(\boldsymbol{x},\boldsymbol{y}) \sum_{i=1}^{M} p(i) = \mathbb{E}\big[\mathscr{E}(\boldsymbol{x},\boldsymbol{y})\big].
\end{aligned}$$

En els intercanvis d'ordre de sumació s'ha tingut en compte que les úniques variables dependents són X^n i Y^n, i que la resta de variables, tan la que dona l'índex i de la paraula transmesa, com la variable condicionada $\Gamma_n|\boldsymbol{c}_i = \boldsymbol{x}$, que genera les paraules del codi en les components diferents de la i-èsima, són independents amb les anteriors i entre elles. □

Demostracions del teorema de codificació de canal. Es donaran a continuació algunes demostracions del teorema de codificació de canal pel mètode probabilistic seguint l'argument explicat al principi de la secció. Totes usen el lema 4.28 i donen fites $\mathbb{E}\big[\mathscr{E}(X^n, Y^n)\big] \leqslant \epsilon_n$ que tendeixen a zero en comptes de fites per a l'esperança de la mitjana d'error.

L'estructura és la mateixa sempre: es considera un subconjunt $\mathcal{A} \subseteq \mathcal{X}^n \times \mathcal{Y}^n$, que es construeix aplicant la llei dels grans nombres a certes variables. Aquest subconjunt té les

característiques següent: la seva probabilitat en relació al parell (X^n, Y^n) és molt gran, tendint a 1 amb n (tot i que potser el nombre d'elements no ho és, comparat amb el total); per als parells $(\boldsymbol{x}, \boldsymbol{y})$ d'aquest conjunt s'aconsegueix trobar una fita de la probabilitat d'error $\mathscr{E}(\boldsymbol{x}, \boldsymbol{y})$ que tendeix a zero amb n; en canvi, per als parells $(\boldsymbol{x}, \boldsymbol{y})$ que no pertanyen al conjunt l'error no se sap controlar, i es dona per fet que la transmissió sempre és errònia, però això no és un problema ja que la probabilitat de trobar-se un parell com aquests és petita.

Primera demostració. La primera està basada en el preprint de Lomnitz i Feder que es pot trobar a *arXiv* i fa servir l'esquema de decisió per màxima versemblança:

PROVA: Es considera la descodificació per màxima versemblança. Sigui $\mathcal{C}$ un codi que té $\boldsymbol{x}$ a la paraula i-èsima i suposi's que en enviar aquesta paraula a través del canal es rep la paraula $\boldsymbol{y}$. La probabilitat d'error en aquesta transmissió, respecte tots els codis amb $\boldsymbol{c}_i = \boldsymbol{x}$, s'ha denotat $\mathscr{E}(i, \boldsymbol{x}, \boldsymbol{y})$, tot i que ja s'ha vist que no depèn de l'índex i.

Sigui $j \neq i$ i sigui $\boldsymbol{c}_j$ la paraula j-èsima del codi. La màxima versemblança podria descodificar $\boldsymbol{y}$ incorrectament, donant com a resultat l'índex j com al valor de $\operatorname{ccdec}(\boldsymbol{y})$, en comptes de l'índex correcte i, només si es compleix la desigualtat

$$p(\boldsymbol{y}|\boldsymbol{c}_j) \geqslant p(\boldsymbol{y}|\boldsymbol{x}).$$

Com que la paraula j-èsima del codi s'ha triat amb la distribució de X^n, l'esdeveniment que correspon a aquesta descodificació (possiblement) incorrecta, en relació a la distribució de probabilitat de X^n, és $\mathcal{A}_j = \{\boldsymbol{c}_j \in \mathcal{X}^n : p(\boldsymbol{y}|\boldsymbol{c}_j) \geqslant p(\boldsymbol{y}|\boldsymbol{x})\} \subseteq \mathcal{X}^n$.

Es recorda la desigualtat de Markov, que diu que per a tota variable aleatòria no negativa W la probabilitat d'un esdeveniment està fitada per $\Pr(W \geqslant a) \leqslant \frac{1}{a}\mathbb{E}[W]$. Aplicant-la a la variable $W = \Pr(Y^n = \boldsymbol{y}|X^n)$, funció de X^n a valors no negatius, i a la constant $a = p(\boldsymbol{y}|\boldsymbol{x})$, s'obté la desigualtat

$$\begin{aligned}\Pr(\mathcal{A}_j) &\leqslant \frac{\mathbb{E}\big[\Pr(Y^n = \boldsymbol{y}|X^n)\big]}{p(\boldsymbol{y}|\boldsymbol{x})} \\ &= \frac{\sum_{\boldsymbol{c}_j \in \mathcal{X}^n} \Pr(X^n = \boldsymbol{c}_j)\Pr(Y^n = \boldsymbol{y}|X^n = \boldsymbol{c}_j)}{p(\boldsymbol{y}|\boldsymbol{x})} = \frac{\Pr(Y^n = \boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{x})} = \frac{p(\boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{x})},\end{aligned}$$

que és una fita independent de j.

Es pot produir un error de descodificació quan tingui lloc l'esdeveniment $\mathcal{A}_j$ per a algun índex $j \neq i$. Altrament la descodificació serà correcta. Per tant la probabilitat d'error de descodificació respecte tots els codis $\mathcal{C}$ amb paraula i-èsima igual a $\boldsymbol{x}$, d'enviar aquesta paraula, rebre la paraula $\boldsymbol{y}$, i que es produeixi un error de descodificació, està fitada per la probabilitat de l'esdeveniment $\cup_{j\neq i}\mathcal{A}_j$ i es té:

$$\mathscr{E}(i, \boldsymbol{x}, \boldsymbol{y}) \leqslant \Pr\left(\bigcup_{j\neq i}\mathcal{A}_j\right) \leqslant \sum_{j\neq i}\Pr(\mathcal{A}_j) \leqslant \sum_{j\neq i}\frac{p(\boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{x})} = (M-1)\frac{p(\boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{x})} < 2^{nR}\frac{p(\boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{x})}.$$

Es considera el conjunt $\mathcal{A}_a^{(n)}$ del lema 4.20, format pels parells $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{X}^n \times \mathcal{Y}^n$ tals que

$$\log\frac{p(\boldsymbol{y}|\boldsymbol{x})}{p(\boldsymbol{y})} \geqslant na \quad \Leftrightarrow \quad 2^{nR}\frac{p(\boldsymbol{y})}{p(\boldsymbol{y}|\boldsymbol{x})} \leqslant 2^{n(R-a)}.$$

Es calcula:

$$\begin{aligned}\mathbb{E}\big[\mathscr{E}(\boldsymbol{x},\boldsymbol{y})\big] &= \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}^n\times\mathcal{Y}^n} p(\boldsymbol{x},\boldsymbol{y})\mathscr{E}(\boldsymbol{x},\boldsymbol{y}) \\ &= \sum_{(\boldsymbol{x},\boldsymbol{y})\notin\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y})\mathscr{E}(\boldsymbol{x},\boldsymbol{y}) \quad + \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y})\mathscr{E}(\boldsymbol{x},\boldsymbol{y}) \\ &\leqslant \sum_{(\boldsymbol{x},\boldsymbol{y})\notin\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y}) \quad + \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y})2^{n(R-a)} \\ &\leqslant 1-\Pr\left(\mathcal{A}_a^{(n)}\right)+2^{n(R-a)} := \epsilon_n,\end{aligned}$$

on s'ha fitat la probabilitat del conjunt $\mathcal{A}_a^{(n)}$ per 1.

Només falta veure que es pot fer tendir la fita a zero quan n tendeix a infinit. Agafant $a=\frac{1}{2}(R+C)$ es pot aplicar el lema 4.20, de manera que $1-\Pr\left(\mathcal{A}_a^{(n)}\right)$ tendeix a zero, i es té $a>R\Rightarrow R-a<0$, per tant $2^{n(R-a)}$ també tendeix a zero. □

Segona demostració. Es dona a continuació una demostració per al canal binari simètric, basada en la que es dona al text de Ball [2], en què es fa servir l'esquema de decisió per proximitat.

PROVA: Es considera el canal binari simètric amb probabilitat d'error p, que té capacitat $C=1-H(p)$ agafant a l'entrada la distribució uniforme. Sigui R un nombre amb $0<R<C$ i sigui $\epsilon>0$ tan petit com es vulgui.

La demostració és semblant a l'anterior. Es mantenen les notacions. Ara s'aplica descodificació per proximitat, per a la qual s'ha de suposar que la probabilitat d'error és $p<\frac{1}{2}$.

Sigui $\boldsymbol{c}_i=\boldsymbol{x}$ la paraula enviada i sigui $\boldsymbol{y}$ la paraula rebuda, a distància $d=d(\boldsymbol{y},\boldsymbol{x})$. El descodificador pot donar com a resultat la paraula errònia $\boldsymbol{c}_j$ amb índex $j\neq i$ en el cas que sigui $d(\boldsymbol{y},\boldsymbol{c}_j)\leqslant d$. Com que les paraules $\boldsymbol{c}_j$ han estat agafades uniformement en $\{\mathtt{0},\mathtt{1}\}^n$ la probabilitat (relativa a l'elecció de $\boldsymbol{c}_j$) que passi això és igual a $\frac{1}{2^n}|B_d(\boldsymbol{y})|=\frac{1}{2^n}|B_d|$ ja que el cardinal de les boles només depèn del radi però no del centre. Per tant la probabilitat d'error, tenint en compte totes les paraules del codi, queda fitada per

$$\mathscr{E}(\boldsymbol{x},\boldsymbol{y})\leqslant\frac{1}{2^n}\sum_{j\neq i}|B_d|=\frac{M-1}{2^n}|B_d|<\frac{2^{nR}}{2^n}|B_d|,\qquad d=d(\boldsymbol{y},\boldsymbol{x}).$$

Es fa servir ara la fita per al nombre de paraules d'una bola a $\{\mathtt{0},\mathtt{1}\}^n$ de radi $<\frac{n}{2}$ donada al problema **2.22**: per a tot $\lambda<\frac{1}{2}$ es té la fita

$$|B_{\lambda n}|=\sum_{i=0}^{\lfloor\lambda n\rfloor}\binom{n}{i}\leqslant 2^{nH(\lambda)}.$$

Es dedueix que per sempre que $d=d(\boldsymbol{x},\boldsymbol{y})<\frac{1}{2}n$ es compleix

$$\mathscr{E}(\boldsymbol{x},\boldsymbol{y})<\frac{2^{nR}}{2^n}2^{nH(\frac{d}{n})}=2^{n(R-1+H(\frac{d}{n}))}.$$

Per a un nombre a amb $0 \leqslant a < \frac{1}{2}$ es defineix el subconjunt

$$\mathcal{A}_a^{(n)} = \big\{(\boldsymbol{x},\boldsymbol{y}) \in \{0,1\}^n \times \{0,1\}^n : d(\boldsymbol{x},\boldsymbol{y}) \leqslant na\big\}.$$

Per als parells $(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{A}_a^{(n)}$ val la fita anterior, ja que $\frac{d(\boldsymbol{x},\boldsymbol{y})}{n} \leqslant a < \frac{1}{2}$. Aleshores:

$$\begin{aligned}
\mathbb{E}\big[\mathscr{E}(\boldsymbol{x},\boldsymbol{y})\big] &= \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{X}^n\times\mathcal{Y}^n} p(\boldsymbol{x},\boldsymbol{y})\mathscr{E}(\boldsymbol{x},\boldsymbol{y}) \\
&= \sum_{(\boldsymbol{x},\boldsymbol{y})\notin\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y})\mathscr{E}(\boldsymbol{x},\boldsymbol{y}) \quad + \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y})\mathscr{E}(\boldsymbol{x},\boldsymbol{y}) \\
&\leqslant \sum_{(\boldsymbol{x},\boldsymbol{y})\notin\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y}) \quad + \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{A}_a^{(n)}} p(\boldsymbol{x},\boldsymbol{y}) 2^{n(R-1+H(\frac{d}{n}))} \\
&\leqslant 1 - \Pr\left(\mathcal{A}_a^{(n)}\right) + 2^{n(R-1+H(\frac{d}{n}))} := \epsilon_n,
\end{aligned}$$

on s'ha fitat la probabilitat del conjunt $\mathcal{A}_a^{(n)}$ per 1.

La variable $d(X^n, Y^n)$ té distribució binomial $\mathrm{Binom}(n,p)$, amb esperança np. Per la llei dels grans nombres, per a tot nombre $a > p$ la probabilitat $\Pr\left(\mathcal{A}_a^{(n)}\right)$ tendeix a 1 amb n.

La capacitat d'un canal binari simètric és $C = 1 - H(p)$. La hipòtesi $R < C$ i la continuïtat de la funció H permeten agafar un nombre $a > p$ (que sigui també $< \frac{1}{2}$ ja que $p < \frac{1}{2}$) tal que se segueixi complint $R < 1 - H(a)$. Per a aquest valor de a, i tenint en compte que H es creixent, es té

$$\frac{d}{n} \leqslant a \quad \Rightarrow \quad H\left(\frac{d}{n}\right) \leqslant H(a) \quad \Rightarrow \quad R - 1 + H\left(\frac{d}{n}\right) \leqslant R - 1 + H(a) < 0.$$

Es dedueix que la potència de 2 en l'expressió per a l'error ϵ_n també tendeix a zero per aquest valor de a i les fites ϵ_n tendeixen a zero. Això demostra el teorema. □

Tercera demostració. A continuació es dona la demostració del teorema 4.19 que es basa en la descodificació per seqüències típiques conjuntes.

Aquest esquema de decisió, que no s'ha descrit prèviament, és el següent: es fixa un llindar $\varepsilon > 0$ i la paraula rebuda $\boldsymbol{y}$ es descodifica amb una paraula codi $\boldsymbol{c} \in \mathcal{C}$ tal que el parell $(\boldsymbol{c},\boldsymbol{y})$ sigui una seqüència ε-típica conjunta per al parell de variables (X^n, Y^n), si existeix. Es pot produir un error de descodificació en la situació següent: quan, en enviar la paraula $\boldsymbol{c}_i$, el parell $(\boldsymbol{c}_i,\boldsymbol{y})$ no sigui una seqüència típica conjunta; o bé quan sí que ho sigui, però hi hagi una altra component $\boldsymbol{c}_j$ amb $j \neq i$ tal que el parell $(\boldsymbol{c}_j,\boldsymbol{y})$ també és una seqüència típica conjunta.

Prova: S'agafa un $\varepsilon > 0$ que dona el llindar respecte el qual es considerin les seqüències típiques conjuntes en la descodificació. Sigui $\mathrm{JTS}_\varepsilon^{(n)}$ el conjunt corresponent.

En enviar $\boldsymbol{x} = \boldsymbol{c}_i$ i rebre $\boldsymbol{y}$ usant un codi $\mathcal{C}$ es pot produir un error perquè el parell $(\boldsymbol{x},\boldsymbol{y})$ no sigui una seqüència típica conjunta o perquè, tot i essent-ho, hi hagi una altra component $\boldsymbol{c}_j$ del codi tal que $(\boldsymbol{c}_j,\boldsymbol{y})$ sigui també una seqüència típica conjunta.

La primera possibilitat, que $(\boldsymbol{x}, \boldsymbol{y})$, que són un valor del parell de variables (X^n, Y^n), no sigui una seqüència típica conjunta, en la qual la descodificació sempre produeix un error: $\mathscr{E}(\boldsymbol{x}, \boldsymbol{y}) = 1$, està fitada per ε, segons l'apartat 1 de la proposició 2.43.

Suposi's que $(\boldsymbol{x}, \boldsymbol{y})$ sí que és seqüència típica conjunta. Tenint en compte que la variable que genera la component j-èsima del codi i la variable de sortida Y^n són independents, però segueixen tenint les mateixes distribucions de probabilitat de X^n i Y^n, l'apartat 3 de la proposició 2.43 dona la fita $2^{-n(I(X;Y)-3\varepsilon)}$ per a la probabilitat que $(\boldsymbol{c}_j, \boldsymbol{y})$ sigui seqüència típica conjunta. Per tant la probabilitat d'error per culpa que hi hagi una altra seqüència típica conjunta, amb primera component una altra paraula codi, està fitat per $(M-1)2^{-n(I(X;Y)-3\varepsilon)} < 2^{n(R-I(X;Y)+3\varepsilon)}$.

Aleshores es té la fita:

$$\begin{aligned}
\mathbb{E}\big[\mathscr{E}(\boldsymbol{x}, \boldsymbol{y})\big] &= \sum_{(\boldsymbol{x},\boldsymbol{y}) \in \mathcal{X}^n \times \mathcal{Y}^n} p(\boldsymbol{x}, \boldsymbol{y}) \mathscr{E}(\boldsymbol{x}, \boldsymbol{y}) \\
&< \sum_{(\boldsymbol{x},\boldsymbol{y}) \notin \mathrm{JTS}_\varepsilon^{(n)}} p(\boldsymbol{x}, \boldsymbol{y}) \cdot 1 \quad + \sum_{(\boldsymbol{x},\boldsymbol{y}) \in \mathrm{JTS}_\varepsilon^{(n)}} p(\boldsymbol{x}, \boldsymbol{y}) 2^{n(R-I(X;Y)+3\varepsilon)} \\
&\leqslant \varepsilon + 2^{n(R-I(X;Y)+3\varepsilon)} := \epsilon_n,
\end{aligned}$$

on s'ha fitat la probabilitat del conjunt $\mathrm{JTS}_\varepsilon^{(n)}$ per 1 i la del seu complementari per ε.

Agafant la distribució d'entrada que assoleixi la capacitat del canal aquestes fites són $\epsilon_n = \varepsilon + 2^{n(R-C+3\varepsilon)}$. Existeix un llindar ε tal que es poden fer tan petites com es vulgui d'un n endavant. En efecte, donat un $\epsilon > 0$ s'agafa $\varepsilon < \min(\frac{\epsilon}{2}, \frac{C-R}{6})$. Aleshores l'exponent $R - C + 3\varepsilon$ és negatiu, i per tant la potència de 2 tendeix a zero, de manera que es pot fer $2^{n(R-C+3\varepsilon)} < \frac{\epsilon}{2}$ per a n prou gran. Aleshores, usant que també $\varepsilon < \frac{\epsilon}{2}$, s'obté la fita $\epsilon_n < \epsilon$. □

4.5 Problemes Complementaris

4.10. *Producte de canals.* Donats dos canals $(\mathcal{X}_1, p(y_1|x_1), \mathcal{Y}_1)$ i $(\mathcal{X}_2, p(y_2|x_2), \mathcal{Y}_2)$ es considera el canal

$$(\mathcal{X}_1 \times \mathcal{X}_2, p(y_1, y_2|x_1, x_2), \mathcal{Y}_1 \times \mathcal{Y}_2)$$

amb probabilitats $p(y_1, y_2|x_1, x_2) = p(y_1|x_1)p(y_2|x_2)$. Calculeu la capacitat d'aquest canal producte en funció de les capacitats dels dos canals de què es parteix.

4.11. Considereu un canal amb alfabets d'entrada i sortida $\mathcal{X} = \mathcal{Y} = \{\mathtt{0}, \mathtt{1}\}^2$, que consisteixen en blocs de dos dígits binaris. Se suposa que el canal és sense soroll, però codifica permutant les lletres de la manera següent: $\mathtt{00} \mapsto \mathtt{01}$, $\mathtt{01} \mapsto \mathtt{10}$, $\mathtt{10} \mapsto \mathtt{11}$, $\mathtt{11} \mapsto \mathtt{00}$.

Les variables d'entrada i sortida es poden considerar com a parells $X = (X_1, X_2)$ i $Y = (Y_1, Y_2)$ on cadascuna de les quatre variables X_i i Y_j és binària. Calculeu

1. $I(X;Y)$ per a una distribució de probabilitats donada $X \sim (p_1, p_2, p_3, p_4)$;
2. la capacitat del canal;
3. $I(X_1;Y_1)$ i $I(X_2;Y_2)$.

4.12. Es considera un canal de comunicació amb alfabet d'entrada $\mathcal{X} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}, \mathtt{d}\}$, alfabet de sortida $\mathcal{Y} = \{\mathtt{0}, \mathtt{1}, \mathtt{2}\}$ i matriu de canal

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$$

Calculeu la capacitat d'aquest canal en bits/símbol transmès i digueu per a quina distribució de probabilitat de la variable d'entrada s'assoleix aquesta capacitat.

Feu el mateix si la matriu de canal és

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

4.13. Sigui $(\mathcal{X}, \boldsymbol{P}, \mathcal{X})$ un canal ternari amb alfabets d'entrada i sortida $\mathcal{X} = \mathbb{Z}_3 = \{0, 1, 2\}$. Es denoten X i Y les variables d'entrada i sortida.

1. Digueu quina pot ser la capacitat màxima; doneu una matriu $\boldsymbol{P}$ amb la qual el canal té aquesta capacitat i digueu per a quina distribució de X s'assoleix.
2. Digueu quina pot ser la capacitat mínima; doneu una matriu $\boldsymbol{P}$ amb la qual el canal té aquesta capacitat i digueu per a quina distribució de X s'assoleix.
3. Calculeu la capacitat del canal per a la matriu

$$\boldsymbol{P} = \begin{bmatrix} \frac{1}{2} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{2} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \end{bmatrix}$$

   i digueu per a quina distribució de X s'assoleix.
4. Calculeu la capacitat del canal per a la matriu

$$\boldsymbol{P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

   i digueu per a quina distribució de X s'assoleix.
5. Per al canal de l'apartat 4 doneu, per a cada longitud n, un codi d'aquesta longitud amb el qual es pot enviar informació amb ratio igual a la capacitat del canal i sense errors.
6. Es fa servir el canal de l'apartat 4 amb distribució d'entrada uniforme $X \sim (\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. Quina és la quantitat d'informació que es transmet per símbol enviat? Quina diferència hi ha amb la capacitat del canal?

5 Codis lineals

Un codi de bloc és un subconjunt $\mathcal{C} \subseteq \mathbb{A}^n$, format per algunes paraules de longitud n. Vist d'aquesta manera és un objecte amb poca estructura per poder treballar-hi de manera eficient: és difícil construir codis amb bons paràmetres i dissenyar algorismes eficients de codificació i descodificació.

Els codis que es fan servir a la pràctica són sempre *codis lineals*. S'agafa com a alfabet $\mathbb{A}$ un cos finit $\mathbb{F} = \mathbb{F}_q$. Les lletres són *escalars*, El conjunt $\mathbb{F}^n$ és un espai vectorial sobre el cos $\mathbb{F}$: les paraules són *vectors*, que es poden sumar entre ells i també multiplicar per escalars. Es treballa amb codis $\mathcal{C} \subseteq \mathbb{A}^n$ que siguin subespais vectorials. D'aquesta manera es disposa de tota la potència de l'*algebra lineal* per treballar amb aquests codis: *bases*, *matrius*, *aplicacions lineals*, *sistemes d'equacions lineals*, *producte escalar*, etc.

Definició 5.1 (Codi lineal)**.** *Un codi lineal de longitud n sobre el cos finit $\mathbb{F}$ és un subconjunt $\mathcal{C} \subseteq \mathbb{F}^n$ que sigui un subespai vectorial.*

Paràmetres d'un codi lineal. Com ja s'ha vist per a codis de bloc generals, els codis lineals tenen associats diversos paràmetres, que mesuren algunes de les seves característiques. En un codi lineal $\mathcal{C} \subseteq \mathbb{F}^n$ es consideren els paràmetres següents:

- el *cardinal* q del cos finit $\mathbb{F} = \mathbb{F}_q$ que fa d'alfabet: és una potència $q = p^e$ d'un nombre primer p, que és la característica del cos;
- la *longitud* n és el nombre de lletres que tenen les paraules; és la dimensió de l'espai ambient $\mathbb{F}^n$ que conté el codi; $\mathbb{F}^n$ és un espai de dimensió n que conté q^n paraules;
- la *dimensió* k és la dimensió del codi $\mathcal{C}$ com a subespai vectorial de $\mathbb{F}^n$ i satisfà la desigualtat $0 \leqslant k \leqslant n$; el codi conté $M = |\mathcal{C}| = q^k$ paraules;
- la *codimensió*, definida com la diferència entre la dimensió de l'espai ambient i la dimensió del codi, que es denotarà $m := n - k$;
- el *ratio d'informació*[3] és $R = \frac{k}{n}$ i el *ratio de redundància* és $1 - R = \frac{m}{n}$; aquests paràmetres signifiquen que, del total de n lletres que contenen les paraules codi, n'hi k que porten *informació* i unes altres $m = n - k$ que contenen *redundància*, gràcies a la qual es poden corregir errors;
- la *distància mínima* $d = d(\mathcal{C}) = \min\left\{d(\boldsymbol{c}, \boldsymbol{c}') : \boldsymbol{c}, \boldsymbol{c}' \in \mathcal{C}, \boldsymbol{c} \neq \boldsymbol{c}'\right\}$ és el mínim de les distancies de Hamming entre dues paraules codi diferents;
- el *radi de tangència* $\tau = \lfloor \frac{d-1}{2} \rfloor$ és el radi màxim tal que les boles d'aquest radi centrades en totes les paraules codi són disjuntes; coincideix amb la *capacitat correctora* del codi.

Un codi lineal amb aquests paràmetres es diu *de tipus* $[n, k, d]_q$, o simplement de tipus $[n, k, d]$ si el cardinal del cos finit se sobreentén. En la notació habitual per a codis de bloc més generals, no necessàriament lineals, és un codi de tipus $(n, M, d)_q$ amb $M = |\mathcal{C}| = q^k$.

[3] En l'enunciat del teorema de codificació de canal es fa servir el *ratio binari* $R = \frac{\log M}{n}$, amb el logaritme agafat en base 2, que és igual al ratio q-ari multiplicat per $\log q$.

5.1 Espai de Hamming

En els codis de bloc s'ha definit la distància de Hamming, que mesura com de diferents són dues paraules a partir del nombre de lletres diferents. Quan els codis són lineals aquesta distància prové d'una norma en l'espai, de manera anàloga al cas de $\mathbb{R}^n$, on moltes distàncies es defineixen a partir de normes: euclidiana, del suprem, del taxista, etc. De fet la distància i la norma de Hamming es poden definir en espais vectorials sobre qualsevol cos, també $\mathbb{R}^n$.

Definició 5.2 (Pes de Hamming)**.** *El pes de Hamming (o norma de Hamming) d'un element $\boldsymbol{x} \in \mathbb{F}^n$ es defineix com el nombre de coordenades no nu·les:*

$$w(\boldsymbol{x}) = \|\boldsymbol{x}\| = \big|\{i : \mathtt{x}_i \neq 0\}\big| \qquad si \qquad \boldsymbol{x} = (\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_n).$$

La relació entre distància de Hamming i pes de Hamming és

$$d(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|, \qquad \boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}^n.$$

Proposició 5.3. *El pes de Hamming compleix la desigualtat triangular*

$$\|\boldsymbol{x} + \boldsymbol{y}\| \leqslant \|\boldsymbol{x}\| + \|\boldsymbol{y}\|.$$

Prova: Si $\boldsymbol{x} = (\mathtt{x}_i)$ i $\boldsymbol{y} = (\mathtt{y}_i)$ aleshores $\mathtt{x}_i + \mathtt{y}_i \neq 0 \Rightarrow \mathtt{x}_i \neq 0$ o bé $\mathtt{y}_i \neq 0$.

També es pot veure usant la desigualtat triangular per a la distància de Hamming que ja s'ha demostrat abans. Com que $\|\boldsymbol{x}\| = d(\boldsymbol{x}, \boldsymbol{0})$ i $d(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|$, fent servir la desigualtat triangular per a la distància s'obté

$$\|\boldsymbol{x} + \boldsymbol{y}\| = d(\boldsymbol{x} + \boldsymbol{y}, \boldsymbol{0}) \leqslant d(\boldsymbol{x} + \boldsymbol{y}, \boldsymbol{y}) + d(\boldsymbol{y}, \boldsymbol{0}) = \big\|(\boldsymbol{x} + \boldsymbol{y}) - \boldsymbol{y}\big\| + \|\boldsymbol{y}\| = \|\boldsymbol{x}\| + \|\boldsymbol{y}\|. \quad \square$$

El pes de Hamming és una *norma* en el sentit habitual d'aquest concepte en matemàtiques: una aplicació $\|\cdot\| : \mathbb{F}^n \to \mathbb{R}$ amb valors no negatius, que val zero només en el vector $\boldsymbol{0} \in \mathbb{F}^n$, que satisfà la desigualtat triangular, i que el producte per escalars satisfà $\|\mathtt{a}\boldsymbol{x}\| = |\mathtt{a}|\|\boldsymbol{x}\|$, on aquí el "valor absolut" en el cos finit $\mathbb{F}$ és el valor absolut trivial, que es defineix posant $|\mathtt{a}| = 0$ o 1 segons si $\mathtt{a} = 0$ o $\mathtt{a} \neq 0$.

Observi's que a diferència de la norma euclidiana i altres normes habituals a $\mathbb{R}^n$, que prenen valors no fitats, la norma de Hamming a l'espai $\mathbb{F}^n$ està fitada per n.

Proposició 5.4 (Distància mínima d'un codi lineal)**.** *La distància mínima d'un codi lineal $\mathcal{C} \subseteq \mathbb{F}^n$ és el mínim dels pesos de Hamming de les seves paraules no nul·les:*

$$d(\mathcal{C}) = \min\big\{\|\boldsymbol{c}\| : \boldsymbol{c} \in \mathcal{C}, \boldsymbol{c} \neq \boldsymbol{0}\big\}.$$

Prova: Donats $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}$ amb $\boldsymbol{x} \neq \boldsymbol{y}$, aleshores $\boldsymbol{z} := \boldsymbol{x} - \boldsymbol{y} \neq \boldsymbol{0}$ és una paraula no nul·la del codi, de pes $\|\boldsymbol{z}\| = d(\boldsymbol{x}, \boldsymbol{y})$. Recíprocament, per a tota paraula $\boldsymbol{z} \in \mathcal{C}$ amb $\boldsymbol{z} \neq \boldsymbol{0}$ les paraules $\boldsymbol{z}$ i $\boldsymbol{0}$ són del codi, diferents, a distància $d(\boldsymbol{z}, \boldsymbol{0}) = \|\boldsymbol{z}\|$. Per tant,

$$\big\{d(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}, \boldsymbol{x} \neq \boldsymbol{y}\big\} = \big\{\|\boldsymbol{z}\| : \boldsymbol{z} \in \mathcal{C}, \boldsymbol{z} \neq \boldsymbol{0}\big\},$$

i la distància mínima és el mínim d'aquest conjunt. $\square$

Així, la relació entre distància i norma de Hamming simplifica el càlcul de la distància mínima dels codis lineals disminuint el nombre de comparacions que cal fer. En efecte, sigui $\mathcal{C} \subseteq \mathbb{A}^n$ un codi qualsevol, no necessàriament lineal, de $|\mathcal{C}| = M$ paraules. Per trobar la distància mínima $d(\mathcal{C})$ s'han de calcular les distàncies entre cada dues paraules diferents, i això requereix de l'ordre de M^2 càlculs. En canvi, si el codi és lineal n'hi ha prou amb calcular els pesos de les paraules no nul·les, que requereix només de l'ordre de M càlculs.

Problemes

5.1. Siguin $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_r \in \mathbb{F}_2^n$ paraules binàries de longitud n. Comproveu que

$$\|\boldsymbol{x}_1\| + \|\boldsymbol{x}_2\| + \cdots + \|\boldsymbol{x}_r\| \equiv \|\boldsymbol{x}_1 + \boldsymbol{x}_2 + \cdots + \boldsymbol{x}_r\| \pmod 2.$$

5.2. Siguin $\mathcal{C}_1$ i $\mathcal{C}_2$ dos codis lineals sobre el cos finit $\mathbb{F}_q$. Siguin n_1, n_2 les seves longituds, k_1, k_2 les seves dimensions i d_1, d_2 les seves distàncies mínimes. Digueu, en funció dels n_i, k_i, d_i i q quines poden ser les longituds, dimensions i distàncies mínimes dels codis següents:

1. $\mathcal{C} = \{\boldsymbol{c}_1 \| \boldsymbol{c}_2 : \boldsymbol{c}_1 \in \mathcal{C}_1, \boldsymbol{c}_2 \in \mathcal{C}_2\}$;
2. $\mathcal{C} = \{\boldsymbol{c} \| \boldsymbol{c} : \boldsymbol{c} \in \mathcal{C}_1\}$;
3. $\mathcal{C} = \{\boldsymbol{c} + \boldsymbol{c} : \boldsymbol{c} \in \mathcal{C}_1\}$;
4. $\mathcal{C} = \{\boldsymbol{c}_1 + \boldsymbol{c}_2 : \boldsymbol{c}_1 \in \mathcal{C}_1, \boldsymbol{c}_2 \in \mathcal{C}_2\}$, si $n_1 = n_2$.

5.3. Siguin $\mathcal{C}_1, \mathcal{C}_2 \subseteq \mathbb{F}_q^n$ dos codis lineals de la mateixa longitud, de dimensions k_i i distàncies mínimes d_i, per a $i = 1, 2$. Es defineix el codi

$$\mathcal{C} = \{\boldsymbol{c} = \boldsymbol{c}_1 \| \boldsymbol{c}_1 + \boldsymbol{c}_2 : \boldsymbol{c}_1 \in \mathcal{C}_1, \boldsymbol{c}_2 \in \mathcal{C}_2\} \subseteq \mathbb{F}_q^{2n}.$$

Comproveu que és lineal, que la seva dimensió és $k = k_1 + k_2$, i demostreu que la seva distància mínima és $d = \min\{2d_1, d_2\}$.

5.4. Sigui $\mathcal{C}$ un codi lineal de tipus $[n, k, d]_2$. Es consideren els conjunts següents:

1. les paraules parells del codi $\mathcal{C}_{\text{ev}} = \{\boldsymbol{x} = (\mathtt{x}_i) \in \mathcal{C} : \sum \mathtt{x}_i = \mathtt{0}\}$;
2. les paraules senars del codi $\mathcal{C}_{\text{odd}} = \{\boldsymbol{x} = (\mathtt{x}_i) \in \mathcal{C} : \sum \mathtt{x}_i = \mathtt{1}\}$;
3. les paraules complementaries de les del codi $\mathcal{C}_{\text{comp}} = \{\overline{\boldsymbol{x}} = (1 - \mathtt{x}_i) : \boldsymbol{x} = (\mathtt{x}_i) \in \mathcal{C}\}$.

Discutiu, per a cadascun d'aquests tres conjunts, si són o no codis lineals, i quan ho siguin doneu els seus paràmetres en funció dels de $\mathcal{C}$.

5.5. Sigui $\mathcal{C} \subseteq \mathbb{F}^n$ un codi lineal q-ari de longitud n, dimensió k i distància mínima $d > 1$. Per a cada índex $i = 1, \ldots, n$ i cada element $\mathtt{a} \in \mathbb{F}$ (lletra de l'alfabet) sigui

$$\mathcal{C}_{i,\mathtt{a}} = \big\{\boldsymbol{x}' = (\mathtt{x}_1, \ldots, \mathtt{x}_{i-1}, \mathtt{x}_{i+1}, \ldots, \mathtt{x}_n) : \boldsymbol{x} = (\mathtt{x}_1, \ldots, \mathtt{x}_{n-1}, \mathtt{a}, \mathtt{x}_{n+1}, \ldots, \mathtt{x}_n) \in \mathcal{C}\big\}$$

el codi de longitud $n - 1$ que s'obté retallant la lletra i-èsima de totes les paraules de $\mathcal{C}$ que tenen una $\mathtt{a}$ en aquesta posició. Se suposa que $|\mathcal{C}_{i,\mathtt{0}}| \neq |\mathcal{C}|$; és a dir, que no totes les paraules codi tenen un zero en una posició donada.

1. Per a quins elements $\mathtt{a} \in \mathbb{F}$ el conjunt $\mathcal{C}_{i,\mathtt{a}}$ és un codi lineal?
2. Demostreu que tots els codis $\mathcal{C}_{i,\mathtt{a}}$ contenen el mateix nombre de paraules.
   INDICACIÓ: Doneu una bijecció $\mathcal{C}_{i,0} \to \mathcal{C}_{i,\mathtt{a}}$ per a cada $\mathtt{a} \in \mathbb{F}$.
3. Quina és la dimensió del codi $\mathcal{C}_{i,0}$?
4. Compareu la distància mínima dels codis $\mathcal{C}_{i,\mathtt{a}}$ amb la de $\mathcal{C}$.
5. Demostreu que si $\mathcal{C}$ és un codi MDS aleshores els codis $\mathcal{C}_{i,\mathtt{a}}$ també ho són.

5.2 Matriu generadora

Com que un codi lineal $\mathcal{C}$ és un subespai vectorial de l'espai vectorial $\mathbb{F}^n$, per a donar un codi lineal no cal donar la llista de totes les seves paraules sinó que n'hi ha prou a donar-ne una base $\boldsymbol{g}_1, \boldsymbol{g}_2, \ldots, \boldsymbol{g}_k$ formada per k vectors, amb $k = \dim \mathcal{C}$. Fixada una base $(\boldsymbol{g}_i)_{1 \leqslant i \leqslant k}$ totes les paraules $\boldsymbol{c} \in \mathcal{C}$ es poden escriure de manera única com a combinació lineal dels vectors d'aquesta base

$$\boldsymbol{c} = \mathtt{m}_1 \boldsymbol{g}_1 + \mathtt{m}_2 \boldsymbol{g}_2 + \cdots + \mathtt{m}_k \boldsymbol{g}_k, \qquad \mathtt{m}_i \in \mathbb{F}.$$

Els coeficients $\mathtt{m}_i \in \mathbb{F}$ són les *coordenades* del vector $\boldsymbol{c}$ en la base $\boldsymbol{g}_i$. Les q^k paraules del codi $\mathcal{C}$ s'obtenen en fer variar aquestes coordenades $\mathtt{m}_i$ entre tots els valors possibles de $\mathbb{F}$.

Definició 5.5 (Matriu generadora)**.** *Una matriu generadora del codi lineal $\mathcal{C} \subseteq \mathbb{F}^n$ es una matriu*

$$\mathbf{G} = \begin{bmatrix} \mathtt{g}_{11} & \mathtt{g}_{12} & \cdots & \mathtt{g}_{1n} \\ \mathtt{g}_{21} & \mathtt{g}_{22} & \cdots & \mathtt{g}_{2n} \\ \vdots & \vdots & & \vdots \\ \mathtt{g}_{k1} & \mathtt{g}_{k2} & \cdots & \mathtt{g}_{kn} \end{bmatrix} \in \mathrm{Mat}_{k \times n}(\mathbb{F}),$$

que té per files les coordenades dels vectors d'una base de $\mathcal{C}$:

$$\boldsymbol{g}_1 = (\mathtt{g}_{11}, \mathtt{g}_{12}, \ldots, \mathtt{g}_{1n}), \quad \boldsymbol{g}_2 = (\mathtt{g}_{21}, \mathtt{g}_{22}, \ldots, \mathtt{g}_{2n}), \quad \ldots \quad \boldsymbol{g}_k = (\mathtt{g}_{k1}, \mathtt{g}_{k2} \cdots \mathtt{g}_{kn}).$$

El nombre de files de $\mathbf{G}$ és la dimensió k i el nombre de columnes és la longitud n. Un codi admet moltes matrius generadores, tantes com bases diferents tingui com a subespai de $\mathbb{F}^n$ (veure problema **5.24**). Les transformacions elementals de files passen d'una matriu generadora a una altra matriu generadora del mateix codi. Sempre es pot passar d'una matriu generadora a una altra qualsevol fent transformacions elementals de files: dues matrius són generadores d'un mateix codi lineal si, i només si, són equivalents per files.

En particular en tot codi es pot agafar una matriu generadora que estigui en forma esglaonada reduïda per files. Aquesta matriu és única.

Una matriu $\mathbf{G} \in \mathrm{Mat}_{k \times n}(\mathbb{F})$ amb $k \leqslant n$ és la matriu generadora d'algun codi si, i només si, té rang màxim igual a k; és a dir, si les seves k files són vectors linealment independents de $\mathbb{F}^n$. En aquest cas el codi és simplement el codi generat per les files de la matriu.

Exemples 5.6. *Alguns exemples de codis lineals i matrius generadores són:*

1. *El codi total $\mathcal{C} = \mathbb{F}^n$ format per totes les paraules de longitud n. És de tipus $[n, n, 1]_q$ i admet com a matriu generadora la matriu identitat $\mathbf{G} = \mathbf{I}_n \in \mathrm{Mat}_{n \times n}(\mathbb{F})$.*

2. *El codi de repetició* $\mathrm{Rep}_q(n)$ *es lineal de tipus* $[n,1,n]_q$ *i admet com a matriu generadora la matriu*

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & \cdots & 1 \end{bmatrix}.$$

3. *El codi binari parell* $\mathrm{Par}_2(n)$ *és lineal de tipus* $[n, n-1, 2]_2$ *i admet com a matriu generadora la matriu*

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 1 \\ 0 & 1 & \cdots & 0 & 1 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 1 \end{bmatrix}.$$

4. *El codi binari* $\mathcal{C} = \{00000, 11100, 00111, 11011\}$ *es lineal de tipus* $[5,2,3]_2$ *i admet com a matriu generadora*

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

5. *El codi binari format per les vuit paraules següents:*

$$\mathcal{C} = \{000000, 001011, 010101, 100110, 011110, 101101, 110011, 111000\} \subset \mathbb{F}_2^6$$

és un codi lineal de tipus $[6,3,3]_2$ *i admet matriu generadora*

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

Codificació. Amb un codi lineal de tipus $[n,k,d]_q$ es poden codificar els elements d'un alfabet de missatges $\mathcal{M}$ amb $|\mathcal{M}| = q^k$. S'agafa com a alfabet de missatges el conjunt de vectors k-dimensionals: $\mathcal{M} \approx \mathbb{F}^k$. D'aquesta manera es pot fer servir àlgebra lineal per a la codificació i descodificació. En els codis lineals s'agafen sempre aplicacions de codificació que siguin aplicacions lineals:

Definició 5.7 (Codificació d'un codi lineal)**.** *Una codificació d'un codi lineal* $\mathcal{C}$ *de tipus* $[n,k,d]_q$ *és una aplicació* $\mathrm{enc}\colon \mathbb{F}_q^k \to \mathbb{F}_q^n$ *que sigui lineal i tingui per imatge el subespai* $\mathcal{C} \subseteq \mathbb{F}_q^n$.

Tenint en compte la fórmula de les dimensions $\dim U = \dim \mathrm{Ker}\, f + \dim \mathrm{Im}\, f$ que satisfà tota aplicació lineal $f\colon U \to V$ entre espais vectorials es veu que la dimensió de Ker(enc) és zero, que equival a dir que la codificació enc és injectiva.

A partir d'una matriu generadora $\mathbf{G}$ del codi es pot definir una aplicació de codificació $\mathrm{enc}\colon \mathbb{F}^k \to \mathcal{C} \subseteq \mathbb{F}^n$ que consisteix simplement en multiplicar els elements $\boldsymbol{m} \in \mathbb{F}^k$, vistos com a vectors-fila, per la matriu generadora a la dreta. El resultat és la paraula codi corresponent $\boldsymbol{c} \in \mathcal{C}$, en forma de vector-fila:

$$\boldsymbol{c} = \mathrm{enc}(\boldsymbol{m}) = \boldsymbol{m} \cdot \mathbf{G}, \qquad \begin{bmatrix} \mathtt{c}_1 & \mathtt{c}_2 & \cdots & \mathtt{c}_n \end{bmatrix} = \begin{bmatrix} \mathtt{m}_1 & \cdots & \mathtt{m}_k \end{bmatrix} \cdot \begin{bmatrix} \mathtt{g}_{11} & \mathtt{g}_{12} & \cdots & \mathtt{g}_{1n} \\ \vdots & \vdots & & \vdots \\ \mathtt{g}_{k1} & \mathtt{g}_{k2} & \cdots & \mathtt{g}_{kn}. \end{bmatrix}.$$

Aquesta aplicació de codificació depèn, és clar, de la matriu generadora escollida. Si convé explicitar-ho es pot denotar $\mathrm{enc}_{\mathbf{G}}$.

Recíprocament, donada una aplicació de codificació d'un codi lineal $\mathrm{enc}\colon \mathbb{F}^k \to \mathcal{C} \subseteq \mathbb{F}^n$, la seva matriu $\mathrm{Mat}(\mathrm{enc}) \in \mathrm{Mat}_{n\times k}$ en les bases canòniques té per columnes les paraules codi $\boldsymbol{g}_i = \mathrm{enc}(\boldsymbol{e}_i)$ per a $i = 1, \ldots, k$, que són una base del codi. Per tant la seva transposada és una matriu generadora del codi.

Aquesta relació entre bases, matrius generadores i aplicacions de codificació es resumeix en el:

Lema 5.8. *Sigui $\mathcal{C} \subseteq \mathbb{F}^n$ un codi lineal. Hi ha correspondències bijectives entre:*

1. *bases de $\mathcal{C}$ com a $\mathbb{F}$-espai vectorial;*
2. *matrius generadores de $\mathcal{C}$;*
3. *aplicacions lineals $\mathrm{enc}\colon \mathbb{F}^k \to \mathbb{F}^n$ amb imatge $\mathcal{C}$.*

PROVA: La correspondència entre bases $\boldsymbol{g}_1, \ldots, \boldsymbol{g}_n$ i matrius generadores $\mathbf{G}$ ve de la definició mateixa: la matriu generadora té per files les components dels vectors d'una base. La correspondència entre matrius generadores $\mathbf{G}$ i aplicacions lineals enc és simplement a través de la matriu de l'aplicació lineal en les bases canòniques $\mathrm{Mat}(\mathrm{enc}) = \mathbf{G}$: les imatges dels vectors $\boldsymbol{e}_i$ de la base canònica de l'espai $\mathbb{F}^k$ són una base de l'espai imatge $\mathcal{C}$, i són les columnes de la matriu $\mathbf{G}$. □

En àlgebra lineal la matriu d'una aplicació lineal, en unes bases fixades, és la que té per columnes les coordenades (en la base d'arribada) de les imatges dels vectors de la base de sortida. Aquesta matriu permet calcular la imatge d'un vector (columna) multiplicant-lo per la matriu a l'esquerra. En teoria de codis el costum es treballar amb la versió per files d'això mateix: la matriu generadora $\mathbf{G}$ és la matriu de l'aplicació lineal $\mathrm{enc}\colon \mathbb{F}^k \to \mathbb{F}^m$ en les bases canòniques, que té per *files* les imatges $\mathrm{enc}(\boldsymbol{e}_i) = \boldsymbol{g}_i$ dels vectors de la base canònica de l'espai $\mathbb{F}^k$. La imatge d'un vector $\mathbf{m} \in \mathbb{F}^k$ (ara un vector fila) es calcula multiplicant-lo per la dreta per la matriu $\mathbf{G}$.

Descodificació. La descodificació $\mathrm{dec}\colon \mathcal{C} \to \mathbb{F}^k$ és l'aplicació inversa $\mathrm{dec} = \mathrm{enc}^{-1}$, que recupera el missatge a partir de la paraula codi corresponent.

El càlcul de $\mathrm{dec}(\boldsymbol{c})$ equival a resoldre el sistema lineal $\mathbf{X} \cdot \mathbf{G} = \boldsymbol{c}$. En la notació habitual dels sistemes lineals és el sistema $\mathbf{G}^\mathsf{T} \cdot \mathbf{X}^\mathsf{T} = \boldsymbol{c}^\mathsf{T}$ de n equacions i k incògnites amb matriu del sistema la transposada $\mathbf{G}^\mathsf{T}$. Com que $k \leqslant n$, aquest és un sistema sobredeterminat, que per a valors generals del terme constant no té solució. De fet, el sistema $\mathbf{G}^\mathsf{T} \cdot \mathbf{X}^\mathsf{T} = \boldsymbol{x}^\mathsf{T}$ amb $\boldsymbol{x} \in \mathbb{F}^n$ té solució si, i només si, $\boldsymbol{x} = \boldsymbol{c} \in \mathcal{C}$ és una paraula codi. En cas que en tingui, el fet que la matriu $\mathbf{G}$ té rang màxim igual k implica que la solució és única i serà precisament el missatge $\boldsymbol{m} = \mathrm{dec}(\boldsymbol{c}) \in \mathbb{F}^k$.

Com que el sistema d'equacions conté equacions redundants es podria resoldre amb menys de les n equacions. De fet es pot resoldre a partir de qualsevol subconjunt d'equacions que en tingui k que són independents. Aquesta observació dona lloc a algorismes de correcció d'errors probabilístics que intenten recuperar el missatge a partir d'un subconjunt d'equacions

en el qual els termes independents, que són lletres de la paraula rebuda $\boldsymbol{x}$, no continguin cap error. Per exemple, una descodificació incompleta nndec: $\mathbb{F}^n \to \mathcal{C} \sqcup \{*\}$ és:

Algorisme 5.9 (Correcció d'errors per *information set decoding*)**.** *Sigui $\mathcal{C}$ un codi lineal de dimensió k i capacitat correctora τ amb matriu generadora $\mathbf{G}$. Es fixa una fita B per al nombre d'intents de descodificació. Donada una paraula $\boldsymbol{x} \in \mathbb{F}^n$,*

SUBCONJUNT I: *S'agafa aleatòriament un conjunt d'índexs $I \subseteq \{1, 2, \ldots, n\}$ de $|I| = k$ elements. Sigui $\mathbf{G}_I$ la submatriu de $\mathbf{G}$ formada per les columnes amb els índexs de I. Si $\mathbf{G}_I$ no és invertible es torna al començament i s'agafa un altre subconjunt I.*

RECUPERAR MISSATGE: *Sigui $\boldsymbol{x}_I$ el vector format per les components de $\boldsymbol{x}$ en les posicions determinades pels índexs de I. Es calcula $\boldsymbol{m} = \boldsymbol{x}_I \mathbf{G}_I^{-1}$.*

COMPROVAR DISTÀNCIA *Sigui $\boldsymbol{c} = \boldsymbol{m} \cdot \mathbf{G}$. Si $d(\boldsymbol{c}, \boldsymbol{x}) \leqslant \tau$ es retorna* $\mathrm{nndec}(\boldsymbol{x}) = \boldsymbol{c}$*; altrament es torna al primer pas i s'agafa un altre subconjunt I, excepte si s'han fet ja B intents, en què es retorna* $\mathrm{nndec}(\boldsymbol{x}) = *$.

Es pot modificar aquest algorisme de diverses maneres per tenir algorismes de descodificació completa.

Definició 5.10 (Codis linealment equivalents)**.** *Dos codis lineals $\mathcal{C}, \mathcal{C}' \subseteq \mathbb{F}^n$ es diuen linealment equivalents (notació $\mathcal{C} \sim \mathcal{C}'$) si existeixen una permutació σ del conjunt dels índexs $\{1, 2, \ldots, n\}$ i n constants no nu·les $\mathtt{a}_1, \ldots, \mathtt{a}_n \in \mathbb{F}$ tals que*

$$\boldsymbol{c} = (\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_n) \in \mathcal{C} \quad \Leftrightarrow \quad \boldsymbol{c}' = \left(\mathtt{a}_1 \mathtt{x}_{\sigma(1)}, \mathtt{a}_2 \mathtt{x}_{\sigma(2)}, \ldots, \mathtt{a}_n \mathtt{x}_{\sigma(n)}\right) \in \mathcal{C}'.$$

Això és la versió per a codis lineals d'un concepte més general d'equivalència de codis que es pot definir per a codis de bloc qualssevol (veure problema **1.26**).

Lema 5.11. *Dos codis linealment equivalents són del mateix tipus $[n, k, d]_q$.*

PROVA: S'ha de veure que tenen la mateixa longitud, dimensió i distància mínima. Per definició tenen la mateixa longitud.

La transformació $\boldsymbol{c} \mapsto \boldsymbol{c}'$ és una bijecció entre tots dos codis, que té per inversa la transformació anàloga amb coeficients $\mathtt{a}_i^{-1}$ (existeixen per ser $\mathtt{a}_i \neq 0$) i amb permutació σ^{-1}, la inversa de sigma. Per tant tenen la mateixa dimensió.

La transformació $\boldsymbol{c} \mapsto \boldsymbol{c}'$ conserva el pes de les paraules: $\|\boldsymbol{c}'\| = \|\boldsymbol{c}\|$ ja que la coordenada $\mathtt{a}_i \mathtt{x}_{\sigma(i)}$ és diferent de zero si, i només si, ho és la coordenada $\mathtt{x}_{\sigma(i)}$, i per tant el nombre de coordenades no nul·les es manté. Això implica que tots dos codis tenen la mateixa distància mínima. □

En termes de matrius generadores, dos codis lineals són equivalents quan es pot passar d'una matriu generadora de l'un a una de l'altre fent transformacions elementals de files de tots tres tipus (que no canvien el codi) i també transformacions elementals de columnes de dos tipus: permutar-les o multiplicar-les per un escalar no nul. Les transformacions de columnes que consisteixen a sumar-li a una un múltiple d'una altra no es poden fer ja que no

mantenen en general tots els paràmetres del codi: la longitud i la dimensió sí que es conserven però la distància mínima pot canviar.

Des del punt de vista de la seva aplicació a la detecció i la correcció d'errors dos codis linealment equivalents serveixen igual. Per tant és habitual canviar un codi per un de linealment equivalent quan convingui.

Definició 5.12 (codificació sistemàtica)**.** *La matriu generadora* $\mathbf{G} \in \mathrm{Mat}_{k\times n}(\mathbb{F})$ *d'un codi es diu sistemàtica (a l'esquerra) si és de la forma següent:*

$$\mathbf{G} = \left[\begin{array}{cccc|ccc} 1 & 0 & \cdots & 0 & \mathtt{g}_{1,k+1} & \cdots & \mathtt{g}_{1,n} \\ 0 & 1 & \cdots & 0 & \mathtt{g}_{2,k+1} & \cdots & \mathtt{g}_{2,n} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 & \mathtt{g}_{k,k+1} & \cdots & \mathtt{g}_{k,n} \end{array}\right] = \left[\mathbf{I}_k \,\middle|\, \mathbf{G}'\right], \quad \textit{amb} \quad \mathbf{G}' \in \mathrm{Mat}_{k\times m}(\mathbb{F}).$$

És a dir, una matriu generadora sistemàtica descompon horitzontalment en dos blocs: el de l'esquerra és un bloc quadrat de mida k on hi ha la matriu identitat $\mathbf{I}_k \in \mathrm{Mat}_k(\mathbb{F})$ i el de la dreta és un bloc de k files i $m = n - k$ columnes que pot ser una matriu qualsevol.

La codificació a partir d'una matriu generadora sistemàtica s'anomena *codificació sistemàtica*: cada paraula $\boldsymbol{m} = (\mathtt{m}_1, \mathtt{m}_2, \ldots, \mathtt{m}_k) \in \mathbb{F}^k$ es codifica amb la paraula codi

$$\boldsymbol{c} = \mathrm{enc}(\boldsymbol{m}) = \boldsymbol{m}\cdot\mathbf{G} = (\mathtt{m}_1, \mathtt{m}_2, \ldots, \mathtt{m}_k, \mathtt{c}_{k+1}, \ldots, \mathtt{c}_n) \in \mathcal{C}, \quad \mathtt{c}_j = \sum_{i=1}^{k} \mathtt{m}_i \mathtt{g}_{i,j}, \quad k+1 \leqslant j \leqslant n,$$

de manera que les k primeres coordenades de la paraula codi $\boldsymbol{c} = \mathrm{enc}(\boldsymbol{m})$ reprodueixen exactament el missatge $\boldsymbol{m}$, o sigui, *la informació*, i les últimes m coordenades contenen *la redundància*: les lletres que s'han afegit a $\boldsymbol{m}$ que permeten que es pugui fer detecció i/o correcció d'errors. Aquestes últimes m lletres $\mathtt{c}_j$ de la paraula $\boldsymbol{c}$ es coneixen també pel nom de *símbols de control*, o *símbols de control de paritat*. Quan s'ha fet la codificació amb una matriu sistemàtica la descodificació és molt senzilla: només cal eliminar les m últimes lletres de la paraula codi quedant-se amb les k primeres, que són el missatge.

No tot codi $\mathcal{C}$ admet codificació sistemàtica. Més concretament,

Proposició 5.13. *Sigui* $\mathcal{C} \subseteq \mathbb{F}^n$ *un codi lineal i sigui* $\mathbf{G}$ *una matriu generadora qualsevol.*

1. *$\mathcal{C}$ admet codificació sistemàtica si, i només si, la forma esglaonada reduïda de la matriu $\mathbf{G}$ és una matriu generadora sistemàtica: de la forma $\left[\mathbf{I}_k \,\middle|\, \mathbf{G}'\right]$.*
2. *$\mathcal{C}$ té codis equivalents que admeten codificació sistemàtica.*

Prova: Les matrius $\left[\mathbf{I}_k \,\middle|\, \mathbf{G}'\right]$ són esglaonades reduïdes. Com que tota matriu té una única forma esglaonada reduïda el codi admet codificació sistemàtica si, i només si, aquesta forma esglaonada reduïda és de la forma $\left[\mathbf{I}_k \,\middle|\, \mathbf{G}'\right]$.

Permutant les columnes de tal manera que la forma esglaonada reduïda de la matriu tingui les columnes dels pivots en les primeres k posicions s'obté un codi equivalent que admet codificació sistemàtica. □

Com que canviar un codi per un de linealment equivalent no canvia el seu tipus ni, per tant, la seva capacitat de detectar i corregir errors, es poden agafar codis que admetin codificació sistemàtica sempre que convingui.

Donat un codi $\mathcal{C}$ la manera de trobar una matriu generadora sistemàtica d'un codi equivalent $\mathcal{C}'$ és partir d'una matriu generadora qualsevol de $\mathcal{C}$ i aplicar reducció de Gauss-Jordan a les files de la matriu, passant a la forma esglaonada reduïda. Si aquesta matriu ja és sistemàtica no cal fer res més: també és matriu generadora del codi $\mathcal{C}$. Altrament, el codi $\mathcal{C}$ no té matriu generadora sistemàtica, però permutant columnes de manera que les primeres continguin els vectors de la base canònica de $\mathbb{F}^k$, que corresponen a les columnes de la forma esglaonada reduïda que contenen els pivots, es té la matriu sistemàtica d'un altre codi $\mathcal{C}'$ linealment equivalent.

L'existència de codificació sistemàtica permet caracteritzar els codis MDS:

Lema 5.14. *Un codi és MDS si, i només si, tots els seus codis linealment equivalents tenen una codificació sistemàtica.*

Prova: La propietat de ser MDS depèn només dels paràmetres del codi. Per tant és un invariant de la classe d'equivalència lineal.

Sigui $\mathcal{C}$ un codi MDS. Sigui $\mathbf{G}$ la seva matriu generadora esglaonada reduïda, que es pot obtenir a partir d'una matriu generadora qualsevol aplicant Gauss-Jordan. Sigui $\mathtt{g}_{kj} = 1$ el pivot de l'última fila de la matriu, que ha de tenir $j \geqslant k$. Si fos $j > k$ aleshores aquesta paraula tindria les primeres k coordenades nul·les i per tant tindria pes de Hamming $w \leqslant n - k < d$, que contradiu la propietat de ser MDS. Per tant ha de ser $j = k$ i això significa que la matriu és sistemàtica a l'esquerra: el codi és sistemàtic.

Suposi's que tots els codis linealment equivalents al codi donat tenen codificació sistemàtica. Sigui $\boldsymbol{c}$ una paraula codi no nul·la de pes $w = w(\boldsymbol{c})$. Canviant el codi per un de linealment equivalent es pot suposar que totes coordenades no nul·les de $\boldsymbol{c}$ són les w últimes, i per tant que les primeres $n - w$ coordenades són zeros. Suposi's que el pes de la paraula fos $w < n - k + 1$. Aleshores el nombre de zeros inicials és $n - w > k - 1 \Rightarrow n - w \geqslant k$. Això contradiu que el codi sigui sistemàtic ja que amb aquest tipus de codificació l'única paraula codi que comença amb k zeros és la paraula zero. □

Problemes

5.6. Sigui $\mathcal{C}$ el codi amb matriu generadora

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

1. Quins són els paràmetres d'aquest codi?
2. Digueu si és sistemàtic a l'esquerra i a la dreta;
3. Codifiqueu els missatges `101` ,`100` i `000` amb la codificació corresponent a la matriu generadora donada a l'enunciat i amb codificació sistemàtica.

4. Quantes matrius generadores diferents té aquest codi?

5.7. Es considera el codi corrector d'errors definit per:

$$\mathcal{C} = \left\{(\mathtt{x}_1, \mathtt{x}_2, \mathtt{x}_3, \mathtt{x}_4) \in \mathbb{F}_2^4 : \mathtt{x}_1 + \mathtt{x}_3 = \mathtt{x}_1 + \mathtt{x}_2 + \mathtt{x}_4 = \mathtt{0}\right\}$$

1. Trobeu totes les paraules codi i digueu quina és la distància mínima. Quina és la capacitat detectora i correctora?
2. Doneu una matriu generadora sistemàtica a l'esquerra i codifiqueu les paraules `00` i `11` usant aquesta matriu.
3. Doneu també una matriu generadora sistemàtica a la dreta i codifiqueu les mateixes paraules.
4. Descodifiqueu per proximitat les paraules $\boldsymbol{x}_1 = \mathtt{1110}$, $\boldsymbol{x}_2 = \mathtt{0111}$ i $\boldsymbol{x}_3 = \mathtt{1111}$.

5.8. Es considera el codi lineal sobre el cos $\mathbb{F}_8 = \mathbb{F}_2[X]_{X^3+X+1}$ amb matriu generadora

$$\mathbf{G} = \begin{bmatrix} \mathtt{1} & \mathtt{x} & \mathtt{0} & \mathtt{0} & \mathtt{1}+\mathtt{x} & \mathtt{0} \\ \mathtt{0} & \mathtt{0} & \mathtt{x} & \mathtt{1} & \mathtt{0} & \mathtt{x} \\ \mathtt{0} & \mathtt{1} & \mathtt{0} & \mathtt{x} & \mathtt{1} & \mathtt{1} \end{bmatrix}.$$

1. Feu una taula amb els elements del cos $\mathbb{F}_8$ vistos com a polinomis $\mathtt{b}_0 + \mathtt{b}_2\mathtt{x} + \mathtt{b}_2\mathtt{x}^2$ amb $\mathtt{b}_i \in \{\mathtt{0}, \mathtt{1}\}$ i la seva correspondència amb potències de $\mathtt{x}$.
2. Trobeu una matriu generadora sistemàtica a l'esquerra.
3. Calculeu la distància mínima.
4. Codifiqueu les paraules $\boldsymbol{m}_1 = (\mathtt{x}, \mathtt{x}^2, \mathtt{x})$ i $\boldsymbol{m}_2 = (\mathtt{1}, \mathtt{1}+\mathtt{x}, \mathtt{x})$ de $\mathbb{F}_8^3$ amb la matriu generadora donada a l'enunciat i també amb la matriu generadora sistemàtica trobada al segon apartat.

5.9. Sigui $\mathcal{C}$ el codi sobre $\mathbb{F}_3$ amb matriu generadora

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 0 & 1 & 2 & 2 \\ 1 & 2 & 2 & 1 & 0 & 2 \\ 0 & 1 & 1 & 2 & 2 & 2 \end{bmatrix}$$

Resolent sistemes lineals $X \cdot \mathbf{G} = \boldsymbol{c}$ digueu si les paraules següents són o no del codi i, en cas que ho siguin, digueu quin missatge $\boldsymbol{m}$ codifiquen, amb la codificació $\mathrm{enc}\colon \mathbb{F}_3^3 \to \mathbb{F}_3^6$ determinat per la matriu $\mathbf{G}$:

$$\boldsymbol{x}_1 = 222212, \quad \boldsymbol{x}_2 = 211001, \quad \boldsymbol{x}_3 = 011100, \quad \boldsymbol{x}_4 = 211201, \boldsymbol{x}_5 = 111222$$

5.10. Doneu una nova demostració de la fita de Singleton $d \leqslant n - k + 1$ per a codis lineals fent servir que tot codi té una matriu generadora en forma esglaonada reduïda.

5.3 Matriu de control

Es pot donar un subespai vectorial de dimensió k dins de $\mathbb{F}^n$ donant-ne una base, formada per k vectors, cadascun amb n coordenades, tal i com s'ha fet a la secció anterior. Això equival a donar una matriu generadora del codi corresponent.

Hi ha una altra manera natural de donar un subespai k-dimensional de $\mathbb{F}^n$: com el conjunt de solucions $\boldsymbol{c} = (\mathtt{c}_1, \mathtt{c}_2, \ldots, \mathtt{c}_n) \in \mathbb{F}^n$ d'un sistema lineal homogeni de $m = n - k$ equacions independents en n incògnites:

$$\begin{cases} \mathtt{h}_{1,1}X_1 & + & \mathtt{h}_{1,2}X_2 & + & \cdots & + & \mathtt{h}_{1,n}X_n & = & 0, \\ \mathtt{h}_{2,1}X_1 & + & \mathtt{h}_{2,2}X_2 & + & \cdots & + & \mathtt{h}_{2,n}X_n & = & 0, \\ \quad\cdots & & \quad\cdots & & & & \quad\cdots & & \\ \mathtt{h}_{m,1}X_1 & + & \mathtt{h}_{m,2}X_2 & + & \cdots & + & \mathtt{h}_{m,n}X_n & = & 0. \end{cases}$$

En forma matricial aquest sistema s'escriu com:

$$\mathbf{H} \cdot \mathbf{X} = \mathbf{0}, \qquad \mathbf{H} = \begin{bmatrix} \mathtt{h}_{1,1} & \mathtt{h}_{1,2} & \cdots & \mathtt{h}_{1,n} \\ \mathtt{h}_{2,1} & \mathtt{h}_{2,2} & \cdots & \mathtt{h}_{2,n} \\ \vdots & \vdots & & \vdots \\ \mathtt{h}_{m,1} & \mathtt{h}_{m,2} & \cdots & \mathtt{h}_{m,n} \end{bmatrix} \in \mathrm{Mat}_{m\times n}(\mathbb{F}),$$

on $\mathbf{X}$ denota el vector columna de les n incògnites i $\mathbf{0}$ és el vector columna de k zeros.

En aquesta situació, la matriu $\mathbf{H}$ es coneix com a:

Definició 5.15 (Matriu de control)**.** *Una matriu de control (o de control de paritat) d'un codi lineal $\mathcal{C}$ és una matriu d'un sistema homogeni d'equacions lineals independents que tingui com a solucions les paraules del codi:*

$$\mathcal{C} = \left\{ \boldsymbol{x} \in \mathbb{F}^n : \mathbf{H} \cdot \boldsymbol{x} = \mathbf{0} \right\} \quad i \quad \mathrm{rank}(\mathbf{H}) = m.$$

És a dir, les matrius de control d'un codi lineal $\mathcal{C}$ de longitud n i codimensió m són les matrius $\mathbf{H} \in \mathrm{Mat}_{m\times n}(\mathbb{F})$ tals que les paraules del codi queden caracteritzades per la identitat:

$$\boldsymbol{x} = (\mathtt{x}_1, \ldots, \mathtt{x}_n) \in \mathcal{C} \quad \Leftrightarrow \quad \begin{bmatrix} \mathtt{h}_{1,1} & \mathtt{h}_{1,2} & \cdots & \mathtt{h}_{1,n} \\ \vdots & \vdots & & \vdots \\ \mathtt{h}_{m,1} & \mathtt{h}_{m,2} & \cdots & \mathtt{h}_{m,n} \end{bmatrix} \begin{bmatrix} \mathtt{x}_1 \\ \mathtt{x}_2 \\ \vdots \\ \mathtt{x}_n \end{bmatrix} = \begin{bmatrix} \mathtt{0} \\ \vdots \\ \mathtt{0} \end{bmatrix}.$$

Les equacions lineals de la forma

$$\mathtt{h}_1 X_1 + \mathtt{h}_2 X_2 + \cdots + \mathtt{h}_n X_n = 0$$

que satisfan les paraules codi s'anomenen equacions de control de paritat, o simplement controls de paritat. Les files de la matriu $\mathbf{H}$ són coeficients d'equacions de control de paritat i també ho són les combinacions lineals d'aquestes files.

La condició $\operatorname{rank}(\mathbf{H}) = m$ equival a dir que les equacions del sistema lineal $\mathbf{H} \cdot \mathbf{X} = \mathbf{0}$ són independents: el conjunt de les seves solucions no es pot donar també com el de les solucions d'un sistema amb menys equacions.

De vegades aquesta condició es relaxa i s'accepten matrius de control amb rang arbitrari; per exemple això és habitual en estudiar els codis LDPC (secció 5.7). En aquest cas la codimensió del codi només es pot assegurar que és $m \leqslant \operatorname{rank}(\mathbf{H})$ i, per tant, que la dimensió és $k \geqslant n - \operatorname{rank}(\mathbf{H})$. La majoria de resultats i tècniques relacionats amb la matriu de control valen igual en aquesta situació més general. En particular el càlcul de la distància mínima amb la proposició 5.28 i la descodificacions per síndrome tal com s'explica a la secció 5.4.

En discutir les possibles matrius de control d'un codi la situació és anàloga a la de les matrius generadores. Un mateix codi admet moltes matrius de control diferents: totes les que tenen el codi com a conjunt de solucions del sistema homogeni corresponent. Fent transformacions elementals de files es manté la propietat de ser matriu de control d'un mateix codi. Si es fan transformacions elementals de columnes de tipus permutació o producte per un escalar no nul es passa de la matriu de control d'un codi a la matriu de control d'un altre codi linealment equivalent. En particular en tot codi es pot agafar una matriu de control de paritat que estigui en forma esglaonada reduïda, i aquesta matriu és única.

La condició que una matriu $\mathbf{H} \in \operatorname{Mat}_{m\times n}(\mathbb{F})$ sigui la matriu de control d'algun codi lineal de dimensió $k = n - m$ és que tingui rang màxim igual a m; és a dir, que les equacions del sistema lineal corresponent siguin independents.

Proposició 5.16 (Relació entre matriu generadora i de control)**.** *Sigui $n = k + m$ i siguin $\mathbf{G} \in \operatorname{Mat}_{k\times n}(\mathbb{F})$ i $\mathbf{H} \in \operatorname{Mat}_{m\times n}(\mathbb{F})$ dues matrius de rang màxim (k i m, respectivament).*

$\mathbf{G}$ i $\mathbf{H}$ són una matriu generadora i una de control d'un mateix codi $\mathcal{C}$ si, i només si, $\mathbf{H} \cdot \mathbf{G}^{\mathsf{T}} = \mathbf{0}$ és la matriu zero de mida $m \times k$.

Prova: Sigui $\mathcal{C}$ el subespai de $\mathbb{F}^n$ format pels vectors que són solució del sistema lineal $\mathbf{H} \cdot \mathbf{X} = \mathbf{0}$: els que satisfan les m equacions

$$\mathtt{h}_{i,1}X_1 + \mathtt{h}_{i,2}X_2 + \cdots + \mathtt{h}_{i,n}X_n = 0, \quad i = 1, \ldots, m.$$

Sigui $\mathcal{C}'$ el subespai de $\mathbb{F}^n$ que té per base els k vectors $\boldsymbol{g}_j = (\mathtt{g}_{j,1}, \mathtt{g}_{j,2}, \ldots, \mathtt{g}_{j,n})$ de les files de $\mathbf{G}$, per a $j = 1, \ldots, k$. Aquests subespais tenen la mateixa dimensió: $k = n - m$, per tant són iguals si, i només si, un està contingut en l'altre.

La condició que el vector $\boldsymbol{g}_j$ satisfaci l'equació i-èsima equival a què el coeficient (i, j)-èsim de la matriu $\mathbf{H} \cdot \mathbf{G}^{\mathsf{T}}$ sigui igual a zero. Per tant, el producte de les matrius és zero.

Per tant $\mathcal{C}' = \mathcal{C} \Leftrightarrow \mathcal{C}' \subseteq \mathcal{C}$ si, i només si, tots els vectors $\boldsymbol{g}_j$ satisfan totes les equacions, que equival al fet que el producte de matrius $\mathbf{H} \cdot \mathbf{G}^{\mathsf{T}}$ sigui la matriu zero. □

Naturalment, la condició de la proposició és equivalent a dir que la matriu transposada $\left(\mathbf{H} \cdot \mathbf{G}^{\mathsf{T}}\right)^{\mathsf{T}} = \mathbf{G} \cdot \mathbf{H}^{\mathsf{T}}$ sigui igual a zero: ara la matriu zero de mides $k \times m$.

Passar d'una matriu generadora d'un codi a una matriu de control, o al revés, correspon a canviar entre donar un subespai vectorial $\mathcal{C} \subseteq \mathbb{F}^n$ donant-ne generadors o equacions. Aquest és un tipus de transformació que se sol fer com a exercici en tots els cursos d'àlgebra lineal elemental. Els càlculs necessaris consisteixen a resoldre un sistema lineal.

Corol·lari 5.17 (Control amb generadora sistemàtica)**.** *Sigui $\mathcal{C} \in \mathbb{F}^n$ un codi lineal de dimensió k. Donada una matriu generadora sistemàtica $\mathbf{G} = \left[\mathbf{I}_k \,\middle|\, \mathbf{G}'\right]$ de $\mathcal{C}$ aleshores la matriu $\mathbf{H} = \left[\mathbf{H}'\middle|\mathbf{I}_m\right]$ amb $\mathbf{H}' = -\mathbf{G}'^{\mathsf{T}}$ és una matriu de control per a aquest codi.*

PROVA: La matriu generadora sistemàtica és de la forma

$$\mathbf{G} = \left[\begin{array}{cccc|ccc} 1 & 0 & \cdots & 0 & \mathsf{g}_{1,k+1} & \cdots & \mathsf{g}_{1,n} \\ 0 & 1 & \cdots & 0 & \mathsf{g}_{2,k+1} & \cdots & \mathsf{g}_{2,n} \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & 1 & \mathsf{g}_{k,k+1} & \cdots & \mathsf{g}_{k,n} \end{array}\right] = \left[\mathbf{I}_k \,\middle|\, \mathbf{G}'\right], \quad \text{amb} \quad \mathbf{G}' \in \mathrm{Mat}_{k\times m}(\mathbb{F}).$$

La matriu $\mathbf{H}$ de l'enunciat és la matriu de la forma:

$$\mathbf{H} = \left[\mathbf{H}'\middle|\mathbf{I}_m\right] = \left[-\mathbf{G}'^{\mathsf{T}} \,\middle|\, \mathbf{I}_m\right] = \left[\begin{array}{ccc|cccc} -\mathsf{g}_{1,k+1} & \cdots & -\mathsf{g}_{k,k+1} & 1 & 0 & \cdots & 0 \\ -\mathsf{g}_{1,k+2} & \cdots & -\mathsf{g}_{k,k+2} & 0 & 1 & \cdots & 0 \\ \vdots & & \vdots & \vdots & \vdots & & \vdots \\ -\mathsf{g}_{1,n} & \cdots & -\mathsf{g}_{k,n} & 0 & 0 & \cdots & 1 \end{array}\right].$$

Totes dues matrius tenen rang màxim k i m, respectivament. El producte $\mathbf{H}\cdot\mathbf{G}^{\mathsf{T}}$ és

$$\mathbf{H}\cdot\mathbf{G}^{\mathsf{T}} = \mathbf{0} = \begin{bmatrix}\mathbf{H}' & \mathbf{I}_m\end{bmatrix}\begin{bmatrix}\mathbf{I}_k \\ \mathbf{G}'^{\mathsf{T}}\end{bmatrix} = -\mathbf{G}'^{\mathsf{T}}\mathbf{I}_k + \mathbf{I}_m\mathbf{G}'^{\mathsf{T}} = -\mathbf{G}'^{\mathsf{T}} + \mathbf{G}'^{\mathsf{T}} = \mathbf{0}.$$

Per tant aplicant la proposició 5.16 es dedueix que $\mathbf{H}$ és matriu de control del codi $\mathcal{C}$. □

Exemples 5.18. *Per als codis lineals dels exemples 5.6 es poden agafar les matrius de control següents:*

1. *Les paraules del codi total no satisfan cap equació; la "matriu" és buida: no té cap fila.*
2. *El codi de repetició* $\mathrm{Rep}_q(n)$ *admet matriu de control:*

$$\mathbf{H} = \begin{bmatrix} 1 & 0 & \cdots & 0 & -1 \\ 0 & 1 & \cdots & 0 & -1 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -1 \end{bmatrix}.$$

3. *El codi parell* $\mathrm{Par}_q(n)$ *admet matriu de control:*

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

4. *El codi de l'apartat 4 admet matriu de control:*

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

5. *El codi de l'apartat 5 admet matriu de control:*

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

Codi dual. A l'espai vectorial $\mathbb{F}^n$ es defineix un producte bilineal posant:

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \mathtt{x}_1\mathtt{y}_1 + \mathtt{x}_2\mathtt{y}_2 + \cdots + \mathtt{x}_n\mathtt{y}_n \in \mathbb{F}, \qquad \boldsymbol{x} = (\mathtt{x}_i)_{1 \leqslant i \leqslant n}, \quad \boldsymbol{y} = (\mathtt{y}_i)_{1 \leqslant i \leqslant n}.$$

Dos vectors es diuen ortogonals si el seu producte bilineal és zero. Es denota $\boldsymbol{x} \perp \boldsymbol{y}$.

Definició 5.19 (Codi dual)**.** *Sigui $\mathcal{C} \subseteq \mathbb{F}^n$ un codi lineal. Es defineix el* codi ortogonal*, o codi dual, $\mathcal{C}^\perp \subseteq \mathbb{F}^n$ com el conjunt de tots els vectors ortogonals a tots els de $\mathcal{C}$:*

$$\mathcal{C}^\perp := \{\boldsymbol{x} \in \mathbb{F}^n : \langle \boldsymbol{x}, \boldsymbol{c} \rangle = 0 \ \forall \boldsymbol{c} \in \mathcal{C}\}.$$

Lema 5.20. *L'ortogonal $\mathcal{C}^\perp$ d'un codi $\mathcal{C}$ de tipus $[n,k]_q$ és un codi de tipus $[n, n-k]_q$. El paper de les matrius generadores i de control de tots dos codis s'intercanvien.*

Prova: Que és lineal és immediat, els paràmetres també, i la relació entre matrius generadores i de control és conseqüència de la proposició 5.16 que relaciona matrius generadores i de control. □

Per exemple, el codi de repetició $\mathrm{Rep}_q(n)$ i el codi parell $\mathrm{Par}_q(n)$ són codis duals.

Problemes

5.11. Comproveu que els codis binaris dels apartats 5 i 6 dels exemples 1.20 són lineals i calculeu els seus paràmetres. Doneu matrius generadores dels codis que siguin no sistemàtiques i unes altres (si existeixen) que siguin sistemàtiques a l'esquerra. Si no existeixen matrius generadores sistemàtiques trobeu un codi equivalent que sí que en tingui. Doneu matrius de control dels codis.

5.12. Es considera el codi lineal sobre el cos $\mathbb{F}_5 = \mathbb{Z}_5$ amb matriu generadora

$$\mathbf{G} = \begin{bmatrix} 1 & 4 & 1 & 2 & 4 & 0 & 0 \\ 3 & 4 & 0 & 0 & 1 & 1 & 4 \\ 0 & 3 & 3 & 4 & 2 & 2 & 3 \end{bmatrix}.$$

1. Comproveu que aquest codi no és sistemàtic a l'esquerra.
2. Doneu un codi equivalent que sí que ho sigui.
3. Calculeu una matriu de control per al codi.
4. Calculeu la seva distància mínima.

5.13. Es consideren els codis binaris $\mathcal{C}_1$ i $\mathcal{C}_2$ definits, respectivament, per les matrius generadora i de control següents:

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}, \qquad \mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 \end{bmatrix}.$$

1. Calculeu la matriu generadora sistemàtica a la dreta de $\mathcal{C}_1$.

2. Calculeu la matriu de control sistemàtica a l'esquerra de $\mathcal{C}_2$.
3. Comproveu que $\mathcal{C}_1 = \mathcal{C}_2$.
4. Calculeu els seus paràmetres.
5. Feu una llista completa de totes les paraules codi.
6. Quina propietat indesitjable té aquest codi? com es milloraria?

5.14. Es consideren els codis binaris de longitud n següents:

- de repetició: $\mathrm{Rep}_2(n) = \{\boldsymbol{x}_0 = \mathtt{00}\cdots\mathtt{0}, \boldsymbol{x}_1 = \mathtt{11}\cdots\mathtt{1}\}$;
- de paritat parell: $\mathrm{Par}_2(n) = \left\{\boldsymbol{x} = (\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_n) \in \{\mathtt{0},\mathtt{1}\}^n : \mathtt{x}_1{+}\mathtt{x}_2{+}\cdots{+}\mathtt{x}_n = \mathtt{0}\right\}$.

1. comproveu que són codis lineals i doneu la seva dimensió i distància mínima;
2. quins són els seus ratios d'informació i de redundància?
3. doneu matrius generadores i matrius de control de paritat de tots dos codis;
4. quants errors poden detectar i corregir cadascun?
5. comproveu que l'un és el dual de l'altre.

Discutiu els codis anàlegs sobre un alfabet que és un cos finit $\mathbb{F}_q$ de q elements.

5.15. *Extensió parell.* Sigui $\mathcal{C}$ un codi binari lineal de tipus $[n,k,d]_2$. S'anomena *extensió parell* de $\mathcal{C}$ el codi $\mathcal{C}^{\mathrm{ev}}$ que s'obté afegint a cada paraula del codi $\boldsymbol{x} = (\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_n) \in \mathcal{C}$ un bit de paritat $\mathtt{x}_{n+1} = \sum_{i=1}^{n} \mathtt{x}_i$.

1. Comproveu que aquesta extensió és un codi lineal i discutiu els seus paràmetres en funció dels de $\mathcal{C}$.
2. Generalitzeu la construcció per a codis sobre un cos finit qualsevol i vegeu el comportament dels paràmetres en aquest cas.
3. Construïu matrius generadores i de control del codi $\mathcal{C}^{\mathrm{ev}}$ a partir de matrius per al codi original.

5.16. Doneu una nova demostració de la fita de Singleton $d \leqslant n-k+1$ per a codis lineals usant la caracterització de la distància mínima en termes de la independència de columnes d'una matriu de control de la proposició 5.28

5.4 Descodificació per síndrome

En els codis lineals es pot usar un mètode per descodificar per proximitat, anomenat *descodificació per síndromes*, que no requereix comparar la paraula rebuda amb totes les paraules codi:

Definició 5.21 (Síndrome)**.** *Sigui $\mathcal{C} \subseteq \mathbb{F}^n$ un codi lineal de codimensió m i sigui $\mathbf{H}$ una matriu de control. Per a cada vector $\boldsymbol{x} \in \mathbb{F}^n$ es defineix la seva síndrome com el vector*

$$\boldsymbol{s} = \mathrm{syn}(\boldsymbol{x}) := \mathbf{H} \cdot \boldsymbol{x} \in \mathbb{F}^m.$$

L'aplicació $\boldsymbol{x} \mapsto \mathrm{syn}(\boldsymbol{x})\colon \mathbb{F}^n \to \mathbb{F}^m$ és lineal exhaustiva amb nucli $\mathcal{C}$.

La síndrome $\boldsymbol{s} = \operatorname{syn}(\boldsymbol{x})$ d'una paraula depèn de la matriu de control que s'estigui fent servir per calcular-la. Les paraules codi $\boldsymbol{c} \in \mathcal{C}$ són les que tenen síndrome zero $\operatorname{syn}(\boldsymbol{c}) = \mathbf{0}$ sigui quina sigui la matriu $\mathbf{H}$.

El fet que l'aplicació lineal $\operatorname{syn}\colon \mathbb{F}^n \to \mathbb{F}^m$ tingui nucli $\mathcal{C}$ és per definició de matriu de control; el fet que sigui exhaustiva és conseqüència de la fórmula de les dimensions: $n = \dim \mathbb{F}^n = \dim \operatorname{Ker}(\operatorname{syn}) + \dim \operatorname{Im}(\operatorname{syn}) = k + \dim \operatorname{Im}(\operatorname{syn}) \Rightarrow \dim \operatorname{Im}(\operatorname{syn}) = n - k = m$.

De manera anàloga a la relació entre matrius generadores i aplicacions de codificació vista al lema 5.8 també hi ha una relació entre matrius de control i aplicacions de síndrome:

Lema 5.22. *Sigui $\mathcal{C}$ un codi lineal. Hi ha una correspondència bijectiva entre:*

1. *sistemes lineals homogenis de m equacions i n incògnites amb conjunt de solucions $\mathcal{C}$;*
2. *matrius de control $\mathbf{H}$ del codi;*
3. *aplicacions lineals $\operatorname{syn}\colon \mathbb{F}^n \to \mathbb{F}^m$ amb nucli $\mathcal{C}$.*

Prova: La correspondència entre sistemes d'equacions i matrius de control és la definició de matriu de control. La correspondència aplicacions de síndrome i matrius de control és la que hi ha entre aplicacions lineals i les seves matrius en bases fixades; en aquest cas les bases canòniques dels espais $\mathbb{F}^n$ i $\mathbb{F}^m$. □

Definició 5.23 (Classe lateral)**.** *Per a cada paraula $\boldsymbol{x} \in \mathbb{F}^n$ la seva classe lateral*[4] *(respecte del codi $\mathcal{C}$) és el conjunt de totes les paraules que s'obtenen en sumar-li paraules codi:*

$$\boldsymbol{x} + \mathcal{C} = \{\boldsymbol{x} + \boldsymbol{c} : \boldsymbol{c} \in \mathcal{C}\} \subseteq \mathbb{F}^n.$$

La classe de qualsevol paraula que sigui del codi és el codi mateix: $\boldsymbol{c} + \mathcal{C} = \mathcal{C}$. Les classes laterals són una *partició* de l'espai $\mathbb{F}^n$ en subconjunts disjunts: entre totes recobreixen l'espai, ja que tot $\boldsymbol{x} \in \mathbb{F}^n$ pertany a la classe $\boldsymbol{x} + \mathcal{C}$, i dues classes són o disjuntes o iguals. Sigui

$$\mathbb{F}^n/\mathcal{C} = \{\boldsymbol{x} + \mathcal{C} : \boldsymbol{x} \in \mathbb{F}^n\}$$

el conjunt de totes les classes laterals.

Geomètricament les classes són els *subespais afins* en la direcció de $\mathcal{C}$: la classe del zero és l'únic subespai afí que és també un subespai lineal, que és el mateix $\mathcal{C}$, i les altres classes són els subespais afins paral·lels a aquest, que s'obtenen en sumar-li un element $\boldsymbol{x} \in \mathbb{F}^n$.

Els elements d'una mateixa classe lateral es caracteritzen pel fet de tenir la mateixa síndrome:

Lema 5.24. *Dues paraules $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}^n$ pertanyen a la mateixa classe lateral si, i només si, tenen la mateixa síndrome:* $\operatorname{syn}(\boldsymbol{x}) = \operatorname{syn}(\boldsymbol{y})$.

La classe lateral d'una paraula $\boldsymbol{x}$ de síndrome $\boldsymbol{s} = \operatorname{syn}(\boldsymbol{x})$ és $\boldsymbol{x} + \mathcal{C} = \operatorname{syn}^{-1}(\boldsymbol{s})$.

[4] Aquest és un concepte general d'àlgebra lineal relacionat amb l'*espai quocient*.

PROVA: Que dues paraules pertanyin a la mateixa classe vol dir que difereixen en una paraula codi: $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{c}$ per a alguna $\boldsymbol{c} \in \mathcal{C}$. Si això es compleix, per linealitat de syn es té

$$\operatorname{syn}(\boldsymbol{y}) = \operatorname{syn}(\boldsymbol{x} + \boldsymbol{c}) = \operatorname{syn}(\boldsymbol{x}) + \operatorname{syn}(\boldsymbol{c}) = \operatorname{syn}(\boldsymbol{x}) + \mathbf{0} = \operatorname{syn}(\boldsymbol{x}).$$

Recíprocament, si totes dues paraules tenen la mateixa síndrome aleshores la paraula $\boldsymbol{y} - \boldsymbol{x}$ és del codi ja que té síndrome zero: $\operatorname{syn}(\boldsymbol{y} - \boldsymbol{x}) = \operatorname{syn}(\boldsymbol{y}) - \operatorname{syn}(\boldsymbol{x}) = \mathbf{0}$. Dient $\boldsymbol{c}$ a aquesta paraula codi es té $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{c}$ i totes dues paraules pertanyen a la mateixa classe lateral.

Com a conseqüència es té que totes les paraules de síndrome $\boldsymbol{s}$ un vector de $\mathbb{F}^m$ donat són l'antiimatge $\operatorname{syn}^{-1}(\boldsymbol{s})$ d'aquest vector per l'aplicació síndrome. □

Per tant hi ha una bijecció entre el conjunt $\mathbb{F}^n/\mathcal{C}$ de les classes laterals i l'espai $\mathbb{F}^m$ de totes les síndromes:

$$\boldsymbol{x} + \mathcal{C} \leftrightarrow \boldsymbol{s} = \operatorname{syn}(\boldsymbol{x}).$$

En cada classe lateral es tria un element de pes mínim, que s'anomena *líder* de la classe:

Definició 5.25 (Líder)**.** *Un* líder *de la classe lateral $\boldsymbol{x} + \mathcal{C}$ és una paraula de la classe que tingui pes mínim:*

$$\operatorname{lead}(\boldsymbol{x} + \mathcal{C}) := \big\{ \boldsymbol{e} \in \boldsymbol{x} + \mathcal{C} : \|\boldsymbol{e}\| \leqslant \|\boldsymbol{y}\| \ \forall\ \boldsymbol{y} \in \boldsymbol{x} + \mathcal{C} \big\} = \operatorname*{argmin}_{\boldsymbol{e} \in \boldsymbol{x} + \mathcal{C}} \|\boldsymbol{e}\|.$$

Naturalment, el líder (únic) de la classe del codi $\mathcal{C}$ és la paraula zero. En general, però, una classe pot tenir més d'un líder diferent.

Com que les classes laterals es corresponen amb les síndromes a través de la bijecció entre $\mathbb{F}^n/\mathcal{C}$ i $\mathbb{F}^m$ es pot parlar dels líders de les síndromes, de manera que lead es veu com una aplicació

$$\operatorname{lead}\colon \mathbb{F}^m \to \mathbb{F}^n, \qquad \operatorname{lead}(\boldsymbol{s}) = \operatorname{lead}(\boldsymbol{x} + \mathcal{C}) \ \text{ per a } \ \boldsymbol{x} \in \mathbb{F}^n \ \text{ amb } \ \operatorname{syn}(\boldsymbol{x}) = \boldsymbol{s}.$$

Descodificació per proximitat usant síndromes. Suposi's que es transmet informació a través d'un canal de comunicacions usant el codi $\mathcal{C}$. En enviar una paraula codi $\boldsymbol{c}$ es rep una paraula $\boldsymbol{x} \in \mathbb{F}^n$, que pot contenir errors. Es pot escriure $\boldsymbol{x} = \boldsymbol{c} + \boldsymbol{e}$ on $\boldsymbol{e} = \boldsymbol{x} - \boldsymbol{c}$ s'anomena *paraula d'error*. És una paraula que té lletres no nul·les exactament en les posicions en què el canal ha introduït errors. El seu pes de Hamming $\|\boldsymbol{e}\| = d(\boldsymbol{x}, \boldsymbol{c})$ és el nombre d'errors que s'han produït.

La paraula rebuda $\boldsymbol{x}$ i la paraula d'error $\boldsymbol{e}$ pertanyen a la mateixa classe lateral: tenen la mateixa síndrome.

Proposició 5.26 (Descodificació per síndrome)**.** *L'aplicació que a cada paraula $\boldsymbol{x} \in \mathbb{F}^n$ li resta un líder de la seva classe és una descodificació per proximitat:*

$$\operatorname{nndec}(\boldsymbol{x}) = \boldsymbol{x} - \operatorname{lead}(\boldsymbol{x} + \mathcal{C}) = \boldsymbol{x} - \operatorname{lead}\big(\operatorname{syn}(\boldsymbol{x})\big).$$

PROVA: Hi ha una bijecció entre el codi $\mathcal{C}$ i la classe lateral $\boldsymbol{x} + \mathcal{C}$ donada per $\boldsymbol{c} \leftrightarrow \boldsymbol{y} = \boldsymbol{x} + \boldsymbol{c}$. Per tant totes les paraules codi es poden escriure de la forma $\boldsymbol{c} = \boldsymbol{y} - \boldsymbol{x}$ per a $\boldsymbol{y} \in \boldsymbol{x} + \mathcal{C}$. La distància $d(\boldsymbol{x}, \boldsymbol{c})$ és el pes $\|\boldsymbol{y}\|$. Per tant la paraula codi que està a distància mínima de

$\boldsymbol{x}$ és aquella que correspon a un vector $\boldsymbol{y} \in \boldsymbol{x} + \mathcal{C}$ de la classe lateral que tingui pes mínim, que és, per definició, el líder de la classe. És a dir, la paraula $\boldsymbol{c} \in \mathcal{C}$ amb $d(\boldsymbol{x}, \boldsymbol{c})$ mínima és $\boldsymbol{x} - \boldsymbol{e}$ amb $\boldsymbol{e} = \operatorname{lead}\big(\operatorname{syn}(\boldsymbol{x})\big)$. □

L'algorisme següent permet fer la descodificació incompleta $\operatorname{nndec}\colon \mathbb{F}^n \to \mathcal{C} \sqcup \{*\}$ calculant prèviament una taula de síndromes i líders.

Algorisme 5.27 (Correcció d'errors per taula de síndromes)**.** *Sigui $\mathcal{C}$ un codi lineal amb capacitat correctora τ. Sigui* $\operatorname{syn}\colon \mathbb{F}^n \to \mathbb{F}^m$ *una aplicació de síndrome, corresponent a alguna matriu de control del codi.*

PRECOMPUTACIÓ: *Es crea una taula amb les síndromes $\boldsymbol{s} = \operatorname{syn}(\boldsymbol{e})$ de totes les paraules $\boldsymbol{e} \in \mathbb{F}^n$ de pes $\|\boldsymbol{e}\| \leqslant \tau$.*

DESCODIFICACIÓ: *Sigui $\boldsymbol{x} \in \mathbb{F}^n$. Es calcula $\boldsymbol{s} = \operatorname{syn}(\boldsymbol{x})$ i es busca a la taula. Si $\boldsymbol{s} = \boldsymbol{s}(\boldsymbol{e})$ aleshores es descodifica* $\operatorname{nndec}(\boldsymbol{x}) = \boldsymbol{x} - \boldsymbol{e}$*; altrament, si $\boldsymbol{s}$ no apareix a la taula de síndromes creada en el primer pas, es descodifica* $\operatorname{nndec}(\boldsymbol{x}) = *$.

També es pot fer una descodificació completa, de la manera següent. En acabar amb totes les paraules $\boldsymbol{e}$ de pes $\leqslant \tau$ se segueix amb paraules de pes més gran $\tau + 1, \tau + 2, \ldots$ allargant la taula amb les síndromes que no havien aparegut fins que la taula contingui totes les síndromes $\boldsymbol{s} \in \mathbb{F}^m$. Aleshores el segon pas de l'algorisme donarà com a resultat una paraula codi per a tota paraula $\boldsymbol{x}$.

La diferència entre tots mètodes és que en el primer cas el fet que $\|\boldsymbol{e}\| \leqslant \tau$ garanteix que $\boldsymbol{e}$ és l'únic líder de la classe corresponent. En canvi, quan $\|\boldsymbol{e}\| > \tau$ la classe pot tenir més d'un líder, que correspon al fet que la paraula $\boldsymbol{x}$ tingui més d'una paraula codi a distància mínima.

Tot i que aquests algorismes són molt simples i eficients, per a la majoria de codis d'interès pràctic el nombre de classes laterals és molt gran i la construcció d'una taula de síndromes/líders no és practicable.

Exemple. Es considera el codi lineal de tipus $[6, 3, 3]_2$ amb matriu de control de paritat donada a l'apartat 5 dels exemples 5.18. Hi ha $2^m = 2^3 = 8$ síndromes possibles. La taula següent conté les classes laterals corresponents i agafa un líder per a cadascuna:

$\boldsymbol{s}$	$\{\boldsymbol{x} \in \mathbb{F}^n : \operatorname{syn}(\boldsymbol{x}) = \boldsymbol{s}\}$	$\operatorname{lead}(\boldsymbol{s})$
000	000000,100110,010101,110011,001011,101101,011110,111000	000000
001	000001,100111,010100,110010,001010,101100,011111,111001	000001
010	000010,100100,010111,110001,001001,101111,011100,111010	000010
011	001000,101110,011101,111011,000011,100101,010110,110000	001000
100	000100,100010,010001,110111,001111,101001,011010,111100	000100
101	010000,110110,000101,100011,011011,111101,001110,101000	010000
110	100000,000110,110101,010011,101011,001101,111110,011000	100000
111	001100,101010,011001,111111,000111,100001,010010,110100	001100

En la taula les paraules de cada classe s'han escrit posant primer el líder i després sumant-li les paraules codi en el mateix ordre en què estan escrites a la primera fila. En cada fila cal

triar un líder que generi un seguit de 2^k vectors que ho hagin aparegut en les files anteriors. Aquesta mena de taules es coneixen amb el nom de *standard array*. Es poden fer servir directament per descodificar: es busca la paraula rebuda a la taula i es descodifica com la paraula codi que té damunt seu a la primera fila.

La primera fila és la de síndrome zero, que té per classe lateral tot el codi i per líder la paraula zero. Aquesta situació és general per a tots els codis.

Després hi ha sis files amb síndromes que tenen un líder de pes 1. En tots els casos aquest és l'únic líder de la classe. En canvi la síndrome `111` correspon a una classe on hi ha tres paraules diferents de pes mínim igual a 2. Qualsevol de les tres es podria haver agafat com a líder de la classe.

Usant aquesta taula es poden fer descodificacions per proximitat com a l'algorisme 5.27:

- incompleta: si la paraula rebuda té síndrome $\mathrm{syn}(\boldsymbol{x}) \neq \mathtt{111}$ es descodifica restant-li el líder corresponent; si la síndrome és `111` es reconeix un error no corregible;
- completa: es tria alguna de les tres paraules de pes 2 com a líder de la classe de síndrome `111`, per exemple la que hi ha a la taula, i es descodifica sempre restant el líder de la síndrome.

La situació és coherent amb el fet que la capacitat correctora d'aquest codi és igual a $1 = \lfloor \frac{3-1}{2} \rfloor$. Sempre que es produeix com a màxim un error es corregirà correctament usant la taula. Quan es produeixen dos errors no es pot corregir.

Problemes

5.17. Construïu una taula de classes laterals amb síndromes per al codi de l'apartat 4 dels exemples 5.18.

5.18. Considereu el codi de tipus $[8, 3, 5]$ sobre el cos $\mathbb{F}_5$ generat pels tres vectors `14101430`, `20133131` i `14122021`. Calculeu una matriu de control de paritat i descodifiqueu per síndrome les paraules següents:

$$\boldsymbol{x} = \mathtt{14114403}, \qquad \boldsymbol{y} = \mathtt{12021340}.$$

5.19. Sigui $\mathcal{C} \subseteq \{\mathtt{0}, 1\}^7$ el codi lineal format pels vectors que són solució del sistema lineal

$$X_2 + X_4 + X_6 = X_1 + X_2 + X_4 = X_5 + X_6 + X_7 = 0.$$

Calculeu la seva dimensió i distància mínima i calculeu també la dimensió i la distància mínima del seu codi dual $\mathcal{C}^\perp$.

5.20. Considereu el codi de tipus $[8, 3, 5]$ sobre el cos $\mathbb{F}_5$ generat pels tres vectors `14101430`, `20133131` i `14122021`. Calculeu una matriu de control de paritat i descodifiqueu per síndrome les paraules següents:

$$\boldsymbol{x} = \mathtt{14114403}, \qquad \boldsymbol{y} = \mathtt{12021340}.$$

5.5 Codis de Hamming

Els primers codis correctors d'errors van ser proposats per *R.W. Hamming* l'any 1950 a [21] amb l'objectiu de corregir els errors que es produïen en els lectors de targetes perforades usats per introduir els programes i les dades als primers ordenadors. Són codis binaris correctors d'un error.

La construcció d'aquests codis es basa en la propietat següent, que caracteritza la distància mínima d'un codi lineal en termes de columnes dependents/independents en una matriu de control:

Proposició 5.28. *Sigui $\mathcal{C}$ un codi lineal amb matriu de control $\mathbf{H}$. La distància mínima $d(\mathcal{C})$ és d si, i només si,*

1. *$d-1$ columnes de $\mathbf{H}$ sempre són linealment independents, i*
2. *existeixen d columnes de $\mathbf{H}$ que són linealment dependents.*

És a dir, $d(\mathcal{C})$ és el menor $d \geqslant 1$ tal que la matriu $\mathbf{H}$ té d columnes linealment dependents.

Prova: Per a cada paraula codi $\boldsymbol{c} = (\mathtt{c}_1, \mathtt{c}_2, \ldots, \mathtt{c}_n) \in \mathcal{C}$ la relació $\mathbf{H} \cdot \boldsymbol{c} = \mathbf{0}$ es pot escriure en termes de les columnes $\mathbf{h}_i$ de $\mathbf{H}$ com la identitat

$$\mathtt{c}_1\mathbf{h}_1 + \mathtt{c}_2\mathbf{h}_2 + \cdots + \mathtt{c}_n\mathbf{h}_n = \mathbf{0}.$$

Si la paraula $\boldsymbol{c}$ té pes de Hamming $\|\boldsymbol{c}\| = w$ la identitat anterior és una relació de dependència lineal entre w columnes diferents de la matriu de control.

Per tant, el codi conté alguna paraula de pes w si, i només si, hi ha w columnes de $\mathbf{H}$ que són linealment dependents.

Que el codi tingui distància mínima d vol dir que no hi ha cap paraula de pes $1 \leqslant w < d$ i, en canvi, hi ha alguna paraula de pes d.

La primera condició equival a què qualsevol família de $< d$ columnes és independent, que clarament equival a què ho sigui qualsevol família de $d-1$ columnes; la segona a què existeixi una família de d columnes dependents. □

Corol·lari 5.29. *Sigui $\mathbf{H}$ una matriu de control per a un codi lineal $\mathcal{C} \subseteq \mathbb{F}^n$.*

1. *si totes les columnes de $\mathbf{H}$ són $\neq \mathbf{0}$ aleshores $d(\mathcal{C}) \geqslant 2$;*
2. *si cap columna de $\mathbf{H}$ és múltiple d'una altra aleshores $d(\mathcal{C}) \geqslant 3$;*
3. *en el cas binari $q = 2$ si les columnes de $\mathbf{H}$ són totes diferents i $\neq \mathbf{0}$ aleshores $d(\mathcal{C}) \geqslant 3$.*

Prova: És una aplicació immediata de la proposició 5.28. En un espai vectorial la família que conté un únic vector és independent si, i només si, el vector és no nul. Una família formada per dos vectors és independent si, i només si, cap és múltiple d'un altre. En el cas binari aquesta condició equival a dir que els vectors són no nuls i diferents. □

Definició 5.30 (Codi de Hamming)**.** *Per a cada enter $m \geqslant 2$ un codi de Hamming binari $\mathrm{Ham}_2(m)$ és un codi amb matriu de control de paritat que tingui per columnes tots els vectors no nuls de $\mathbb{F}_2^m$.*

Per exemple, es poden construir codis de Hamming $\mathrm{Ham}_2(3)$ i $\mathrm{Ham}_2(4)$ agafant les matrius de control següents:

$$\begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \quad \text{i} \quad \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}.$$

Canviar l'ordre de les columnes equival a canviar el codi per un de linealment equivalent, i això conserva les seves propietats. Per tant la construcció dels codis de Hamming binaris els determina només llevat d'equivalència lineal (de fet, llevat de permutar les lletres de cada paraula). En les matrius anteriors s'han posat les columnes en ordre creixent dels enters representats. Es poden trobar matrius de control sistemàtiques posant a les últimes m columnes els vectors binaris que tenen només un 1.

Per exemple, el codi $\mathrm{Ham}_2(4)$ admet la matriu de control sistemàtica

$$\mathbf{H} = \left[\begin{array}{ccccccccccc|cccc} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right]$$

a la qual li correspon la matriu generadora sistemàtica següent:

$$\mathbf{G} = \left[\begin{array}{ccccccccccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{array}\right].$$

Usant aquesta matriu generadora cada $\boldsymbol{m} = (\mathtt{m}_1, \mathtt{m}_2, \mathtt{m}_3, \mathtt{m}_4, \mathtt{m}_5, \mathtt{m}_6, \mathtt{m}_7, \mathtt{m}_8, \mathtt{m}_9, \mathtt{m}_{10}, \mathtt{m}_{11}) \in \mathbb{F}_2^{11}$ es codifica amb la paraula $\boldsymbol{c} = \mathrm{enc}(\boldsymbol{m}) = \boldsymbol{m} \cdot \mathbf{G} \in \mathbb{F}_2^{15}$ obtinguda afegint als 11 bits d'informació $\mathtt{m}_i$ els quatre bits de paritat següents:

$$\begin{aligned} \mathtt{c}_{12} &= \mathtt{m}_5 + \mathtt{m}_6 + \mathtt{m}_7 + \mathtt{m}_8 + \mathtt{m}_9 + \mathtt{m}_{10} + \mathtt{m}_{11}, \\ \mathtt{c}_{13} &= \mathtt{m}_2 + \mathtt{m}_3 + \mathtt{m}_4 + \mathtt{m}_8 + \mathtt{m}_9 + \mathtt{m}_{10} + \mathtt{m}_{11}, \\ \mathtt{c}_{14} &= \mathtt{m}_1 + \mathtt{m}_3 + \mathtt{m}_4 + \mathtt{m}_6 + \mathtt{m}_7 + \mathtt{m}_{10} + \mathtt{m}_{11}, \\ \mathtt{c}_{15} &= \mathtt{m}_1 + \mathtt{m}_2 + \mathtt{m}_4 + \mathtt{m}_5 + \mathtt{m}_7 + \mathtt{m}_9 + \mathtt{m}_{11}. \end{aligned}$$

Proposició 5.31. *Els codis de Hamming* $\mathrm{Ham}_2(m)$ *són codis perfectes de tipus*

$$[2^m - 1, 2^m - m - 1, 3]_2.$$

PROVA: La seva longitud és el nombre de columnes de la matriu de control $\mathbf{H}$ a partir de la qual s'ha definit, que és $n = 2^m - 1$: el nombre de vectors no nuls a l'espai $\mathbb{F}^m$.

El rang d'aquesta matriu és màxim igual a m. En efecte, conté submatrius $m \times m$ que contenen en les seves columnes qualssevol m vectors no nuls, en algun ordre. En particular moltes d'aquestes submatrius tenen determinant no nul. Per tant la seva codimensió és m i la dimensió és $k = n - m = 2^m - m - 1$.

El càlcul de la distància mínima a partir de la matriu de control es pot fer amb la proposició 5.28. En aquest cas, com que cap columna de $\mathbf{H}$ és zero, una columna sempre és linealment independent. Com que dues columnes són sempre diferents, l'única combinació lineal seva que dona zero és la trivial i per tant dues columnes són sempre independents. En canvi existeixen tres columnes que són linealment dependents: agafant dues columnes diferents qualssevol la seva suma és una altra columna diferent de totes dues. Aleshores la suma de totes tres (amb coeficient 1) dona zero i per tant aquestes tres columnes són linealment dependents. Així, el mínim nombre de columnes linealment dependents és 3 i segons la proposició 5.28 això és la distància mínima del codi.

Per comprovar que són perfectes es veu que assoleixen la igualtat en la fita de Hamming

$$M = |\mathcal{C}| \leqslant \frac{2^n}{\sum_{i=0}^{\tau} \binom{n}{i}}.$$

En aquest cas es té $\tau = 1$ i, per tant, el denominador és $\binom{n}{0} + \binom{n}{1} = 1 + n$. Es té

$$(1+n)M = (1 + 2^m - 1)2^k = 2^m 2^{n-m} = 2^n.$$

Per tant, se satisfà la igualtat i el codi es perfecte. □

Aquests codis tenen capacitat correctora $\tau = \lfloor \frac{d-1}{2} \rfloor = 1$, de manera que corregeixen tots els errors que afectin només un dels 2^m bits de la paraula transmesa.

En augmentar el paràmetre m són cada vegada més eficients: el seu ratio de redundància $R = \frac{m}{2^m - m - 1}$ tendeix a zero. En canvi, com que les paraules són cada vegada més llargues, és més probable que es vegin afectades per més d'un error, que el codi no sigui capaç de corregir. Per a valors petits de m la taula següent dona els paràmetres dels codis $\mathrm{Ham}(m)$:

m	n	k
2	3	1
3	7	4
4	15	11
5	31	26
6	63	57
7	127	120
8	255	247
⋮	⋮	⋮

Observi's que el codi $\mathrm{Ham}_2(2)$ és de tipus $[3, 1, 3]$ i és el codi de repetició de longitud 3.

Descodificació de codis de Hamming. Sigui $\mathbf{H}$ una matriu de control d'un codi de Hamming $\mathrm{Ham}(m)$. Sigui $\boldsymbol{e}_i = (\mathtt{0}, \ldots, \mathtt{0}, \mathtt{1}, \mathtt{0}, \ldots, \mathtt{0}) \in \mathbb{F}_2^n$ el vector i-èsim de la base canònica, que té un $\mathtt{1}$ a la coordenada i-èsima i un zero a les demés coordenades. Aquests són els vectors d'error dels errors que afecten un únic bit. La síndrome $\mathrm{syn}(\boldsymbol{e}_i) = \mathbf{H} \cdot \boldsymbol{e}_i \in \mathbb{F}^m$ és la columna i-èsima de la matriu $\mathbf{H}$. Com que les columnes de $\mathbf{H}$ contenen tots els vectors no nuls de $\mathbb{F}_2^m$ les síndromes dels vectors $\boldsymbol{e}_i$ són totes les síndromes no nul·les possibles.

Si s'envia la paraula del codi $\boldsymbol{c}$ i es rep la paraula $\boldsymbol{x} = \boldsymbol{c} + \boldsymbol{e}$ corrompuda amb un error $\boldsymbol{e}$, per descodificar n'hi ha prou a calcular la síndrome $\boldsymbol{s} = \mathrm{syn}(\boldsymbol{x})$. Si la síndrome és zero no hi ha error: la paraula rebuda és del codi i coincideix amb l'enviada. Si la síndrome és un vector no nul de $\mathbb{F}_2^m$ aleshores aquest vector és una de les columnes de la matriu $\mathbf{H}$; si és la columna i-èsima aleshores $\boldsymbol{s} = \boldsymbol{s}(\boldsymbol{e}_i)$ i es dedueix que el vector d'error és l'i-èsim vector de la base canònica. És a dir, que l'error de transmissió s'ha produït en el bit i-èsim de la paraula. Per corregir-lo simplement es canvia aquest bit en la paraula $\boldsymbol{x}$.

En particular, si la matriu de control té a les columnes els dígits binaris dels enters entre 1 i $2^m - 1$ ordenats en ordre creixent, aleshores la síndrome $\boldsymbol{s} = \mathrm{syn}(\boldsymbol{x})$ és la representació binària d'un nombre enter entre 0 i $2^m - 1$. Si aquesta síndrome és diferent de zero indica la posició del bit on s'ha produït l'error, que s'ha de canviar per corregir-lo.

Extensió amb bit de paritat. Com que la seva distància mínima és senar, afegint-los un bit de paritat s'aconsegueix augmentar-la fins a quatre. A la pràctica aquests són els que se solen usar, ja que a més de corregir un error poden detectar-ne tres, i tenen longitud 2^m potència de 2, que sol anar bé en aplicacions digitals. Així,

Els codis de Hamming estesos $\mathrm{Ham}_2^{\mathrm{ev}}(m)$ són codis de tipus $[2^m, 2^m - m - 1, 4]_2$.

Hamming no binaris. Es poden construir també codis de Hamming sobre altres cossos finits $\mathbb{F}_q$. Per a cada $m \geqslant 1$ s'agafen matrius de control de paritat que tinguin a les columnes el màxim nombre possible de paraules de $\mathbb{F}_q^m$ de tal manera que cap d'elles sigui múltiple d'una altra. En el problema **5.22** es calculen els seus paràmetres i es comprova que són codis perfectes: són els únics que es coneixen quan $q > 3$ apart del codi total.

Problemes

5.21. Sigui $\mathrm{Ham}_2^{\mathrm{ev}}(m)$ l'extensió parell del codi de Hamming $\mathrm{Ham}_2(m)$. Digueu com es construeix una matriu de control per a aquest codi i doneu un algorisme de detecció/correcció per síndrome que corregeixi tots els errors d'un bit i detecti tots els errors de fins a tres bits.

5.22. *Codis de Hamming no binaris.* Sigui $\mathbb{F}_q$ un cos finit de q elements. Per a cada enter $m \geqslant 2$ es defineix un codi de Hamming q-ari de codimensió m com un codi amb matriu de control que tingui a les seves columnes generadors de tots els subespais de dimensió 1 de $\mathbb{F}_q^m$: totes les rectes que passen per l'origen. Es denotarà $\mathrm{Ham}_q(m)$.

1. Doneu els paràmetres d'aquest codi: longitud, dimensió i distància mínima.
2. Comproveu que tots els aquests codis de Hamming són codis perfectes.
3. Trobeu una matriu generadora de $\mathrm{Ham}_3(2)$.

5.6 Codis de Reed-Muller

Referència: Ball [2, Chapter 9], Brunat-Ventura [3, Capítol 12]

Els *codis de Reed-Muller* es construeixen avaluant polinomis de varies variables en cossos finits: amb funcions $\mathbb{F}^m \to \mathbb{F}$ definides per polinomis en m variables.

Aquí es considerarà només el cas de codis binaris, amb $\mathbb{F} = \mathbb{F}_2 = \{\mathtt{0}, \mathtt{1}\}$. En el cas binari, com que $\mathtt{x}^n = \mathtt{x}$ per a tot $\mathtt{x} \in \mathbb{F}_2$ i tot exponent $n \geqslant 1$, només s'han de considerar polinomis en què tots els monomis són producte de variables diferents. En aquesta secció es treballarà amb la notació següent: m és el nombre de variables; $\boldsymbol{X}$ són totes les variables $X_1, \ldots, X_m$; per a cada subconjunt $I \subseteq \{1, \ldots, m\}$ es denota $\boldsymbol{X}^I = \prod_{i \in I} X_i$ el monomi producte de les variables amb índexs en I. El nombre de monomis de grau $d = |I|$ és $\binom{m}{d}$ i el nombre total de monomis de tots els graus és el de subconjunts: $2^m = \sum_{d=0}^{m} \binom{m}{d} = (1+1)^m$.

Definició 5.32. *S'anomenen polinomis booleans en m variables els polinomis de la forma*

$$f(\boldsymbol{X}) = \sum_{I \subseteq \{1,\ldots,m\}} \mathtt{f}_I \boldsymbol{X}^I, \quad \mathtt{f}_I \in \mathbb{F}_2.$$

Es denotarà $\mathcal{B}[\boldsymbol{X}]$ el conjunt d'aquests polinomis.
És un $\mathbb{F}_2$-espai vectorial de dimensió 2^m, amb base els monomis $\boldsymbol{X}^I$.

Es diu grau d'un polinomi booleà al nombre màxim de variables $|I|$ d'un monomi $\boldsymbol{X}^I$ que porti coeficient no nul: $\mathtt{f}_I = \mathtt{1}$. Tot polinomi booleà en m variables té grau com a màxim igual a m, a diferència dels polinomis ordinaris, que n'hi ha de grau arbitràriament gran.

Es denotarà $\mathcal{B}[\boldsymbol{X}]_r$ el subespai dels polinomis booleans de grau $\deg f \leqslant r$. Aquest subespai té dimensió $\dim \mathcal{B}[\boldsymbol{X}]_r = \sum_{d=0}^{r} \binom{m}{d}$.

Funcions booleanes. S'anomenen *funcions booleanes* les aplicacions $\{\mathtt{0}, \mathtt{1}\}^m \to \{\mathtt{0}, \mathtt{1}\}$. Identificant $\{\mathtt{0}, \mathtt{1}\}$ amb el cos finit $\mathbb{F}_2$ de dos elements, el conjunt $\mathrm{Map}(\mathbb{F}_2^m, \mathbb{F}_2)$ de les funcions booleanes és un $\mathbb{F}_2$-espai vectorial de dimensió $|\mathbb{F}_2^m| = 2^m$. Els elements $\boldsymbol{x}$ d'aquest espai es denotaran com a 2^m-tuples indexades en el conjunt $\mathbb{F}_2^m$ de la manera següent: l'aplicació $\boldsymbol{x} \colon \mathbb{F}_2^m \to \mathbb{F}_2$ s'identifica amb la 2^m-tupla $(\mathtt{x}_{\boldsymbol{a}})_{\boldsymbol{a} \in \mathbb{F}_2^m}$ amb components $\mathtt{x}_{\boldsymbol{a}} = \boldsymbol{x}(\boldsymbol{a})$ les imatges per $\boldsymbol{x}$ dels elements $\boldsymbol{a} \in \mathbb{F}_2^m$. D'aquesta manera es té una identificació $\mathrm{Map}(\mathbb{F}_2^m, \mathbb{F}_2) \approx \mathbb{F}_2^{2^m}$.

Cada polinomi booleà indueix una aplicació $\boldsymbol{a} \mapsto f(\boldsymbol{a}) \colon \mathbb{F}_2^m \to \mathbb{F}_2$ en avaluar-lo sobre els elements $\boldsymbol{a} \in \mathbb{F}_2^m$, que es denotarà $\mathrm{av}(f)$. De fet, es poden identificar els polinomis amb les aplicacions gràcies al següent:

Lema 5.33. *L'aplicació* $\mathrm{av} \colon \mathcal{B}[\mathcal{X}] \to \mathrm{Map}(\mathbb{F}_2^m, \mathbb{F}_2)$ *que envia cada polinomi booleà a l'aplicació que indueix en avaluar-lo en els elements de $\mathbb{F}_2^m$ és un isomorfisme de $\mathbb{F}_2$-espais vectorials.*

Prova: L'aplicació av és clarament lineal. Com que tots dos espais tenen la mateixa dimensió n'hi ha prou a veure que és injectiva: el nucli és zero. O sigui, que l'únic polinomi booleà $f(\boldsymbol{X})$ que indueix l'aplicació zero és el polinomi zero: el que no té cap monomi $\boldsymbol{X}^I$ amb coeficient $\mathtt{f}_I = \mathtt{1} \in \mathbb{F}_2$.

Es farà per inducció sobre el nombre de variables m.

Per a $m = 1$ hi ha quatre polinomis booleans. $\mathtt{0}, \mathtt{1}, X, \mathtt{1}+X$, L'únic dels quatre que indueix l'aplicació zero és el primer: $\mathtt{1}$ no s'anul·la enlloc; X no s'anul·la en $\mathtt{1}$ i $\mathtt{1}+X$ no s'anul·la en $\mathtt{0}$.

Suposi's demostrat per a un m. Tot polinomi booleà en $m+1$ variables s'escriu com $f(\boldsymbol{X}_{m+1}) = f_1(\boldsymbol{X}_m) + f_2(\boldsymbol{X}_m)X_{m+1}$, amb f_1 i f_2 polinomis booleans en m variables, on el subíndex a $\boldsymbol{X}$ indica el nombre de variables. Suposi's que aquest polinomi indueix l'aplicació zero. Aleshores en avaluar-la en elements $\boldsymbol{a} \in \mathbb{F}_2^{m+1}$ amb última coordenada $\mathtt{0}$ es veu que la funció f_1 s'anul·la en tot $\mathbb{F}_2^m$, i per hipòtesi d'inducció ha de ser el polinomi booleà zero: $f_1(\boldsymbol{X}_m) = 0$. Per tant $f(\boldsymbol{X}_{m+1}) = f_2(\boldsymbol{X}_m)X_{m+1}$. En avaluar f en elements amb última coordenada $\mathtt{1}$ es veu que f_2 s'anul·la en tot $\mathbb{F}_2^m$ i per tant també aquest polinomi ha de ser el zero. Per tant, $f(\boldsymbol{X}_{m+1}) = 0$, com s'havia de demostrar. □

Al conjunt $\mathrm{Map}(\mathbb{F}_2^m, \mathbb{F}_2)$, a més de la suma i producte per escalars, hi ha també un producte d'aplicacions, que s'obté multiplicant les imatges[5]. Aquest producte es tradueix en un producte en l'espai dels polinomis booleans: el producte de polinomis que reflecteix el de les aplicacions induïdes. És el producte que s'obté estenent per distributivitat el producte de monomis següent: el producte de dos monomis d'exponents I i J és el monomi producte de totes les variables que estan en l'un o en l'altre. O sigui, $\boldsymbol{X}^I\boldsymbol{X}^J = \boldsymbol{X}^{I\cup J}$.

En el conjunt $\mathrm{Map}(\mathbb{F}_2^m, \mathbb{F}_2)$ es pot considerar la base "canònica" formada per les 2^m aplicacions que envien un element $\boldsymbol{e} \in \mathbb{F}_2^m$ fixat a l'$\mathtt{1}$ i tots els demés a $\mathtt{0}$. Si $\boldsymbol{e} = (\mathtt{e}_1, \ldots, \mathtt{e}_m)$ són les components, aquesta aplicació és la induïda pel polinomi booleà

$$f_{\boldsymbol{e}}(\boldsymbol{X}) = \prod_{i=1}^{m}(X_i + \mathtt{e}_i + \mathtt{1})$$

ja que, en avaluar aquest polinomi en un element qualsevol $\boldsymbol{a} = (\mathtt{a}_1, \ldots, \mathtt{a}_m) \in \mathbb{F}_2^m$ el resultat és $f_{\boldsymbol{e}}(\boldsymbol{a}) = \prod_{i=1}^m(\mathtt{a}_i + \mathtt{e}_i + \mathtt{1})$, que és igual a $\mathtt{1}$ quan ho són tots els factors, i això passa quan sigui $\mathtt{a}_i = \mathtt{e}_i$ per a tot $i = 1, \ldots, m$; és a dir, si, i només si, $\boldsymbol{a} = \boldsymbol{e}$.

Més endavant es consideraran també polinomis que generalitzen els anteriors:

$$f_{K,\boldsymbol{u}}(\boldsymbol{X}) = \prod_{i\in K}(X_i + \mathtt{u}_i + 1), \qquad K \subseteq \{1, \ldots, m\}, \quad \boldsymbol{u} \in \mathbb{F}_2^{|K|},$$

on les components de $\boldsymbol{u} = (\mathtt{u}_i)_{i\in K}$ estan indexades per índexs del conjunt K. En avaluar un polinomi com aquest en un element $\boldsymbol{a} \in \mathbb{F}_2^m$ el resultat és $\mathtt{1}$ quan les components $\mathtt{a}_i$ per als índexs i que pertanyen a K coincideixen amb els $\mathtt{u}_i$ i zero altrament.

Ordre dels elements de les bases. Per treballar amb codis de Reed-Muller, en particular per escriure les matrius generadores i de control, s'han de fixar bases en els espais dels espais de polinomis booleans i de les aplicacions $\mathbb{F}_2^m \mapsto \mathbb{F}_2$.

Com a base dels polinomis booleans s'agafen els monomis. Aquí es consideraran ordenats primer per grau i després en ordre lexicogràfic, que és l'ordre que s'agafa habitualment. O sigui, ordenats com

$$1, X_1, \ldots, X_m, X_1X_2, \ldots, X_1X_m, X_2X_3, \quad \ldots, \quad X_1X_2X_3, X_1X_2X_4, \quad \ldots, \quad X_1X_2\cdots X_n.$$

[5]Per a tot cos $\mathbb{K}$ i conjunt A el conjunt $\mathrm{Map}(A, \mathbb{K})$ té una estructura natural de $\mathbb{K}$-*àlgebra* amb suma, producte per escalars i producte tots tres definits sobre les imatges de les aplicacions.

En l'espai de les aplicacions s'agafen bases canòniques $(\boldsymbol{x}_{\boldsymbol{e}})$ indexades en elements $\boldsymbol{e} \in \mathbb{F}_2^m$, formades per aplicacions

$$\boldsymbol{x}_{\boldsymbol{e}}(\boldsymbol{a}) = \begin{cases} 1, & \boldsymbol{a} = \boldsymbol{e}, \\ 0, & \boldsymbol{a} \neq \boldsymbol{e}. \end{cases}$$

En aquest cas hi ha dues opcions principals per ordenar els elements $(\mathtt{a}_i)_{1 \leqslant i \leqslant m} \in \mathbb{F}_2^m$: segons l'ordre dels enters que representen en binari:

$$\mathtt{a}_1 2^{m-1} + \mathtt{a}_2 2^{m-2} + \cdots + \mathtt{a}_{m-2} 4 + \mathtt{a}_{m-1} 2 + \mathtt{a}_m \in \mathbb{N},$$

o el dels polinomis a coeficients en $\mathbb{F}_2$:

$$\mathtt{a}_1 + \mathtt{a}_2 X + \mathtt{a}_3 X^2 + \cdots + \mathtt{a}_{m-1} X^{m-2} + \mathtt{a}_m X^{m-1} \in \mathbb{F}_2[X].$$

En totes dues opcions, els elements de $\mathbb{F}_2^m$, ordenats de la manera corresponent, s'obtenen recursivament a partir dels de $\mathbb{F}_2^{m-1}$ afegint-los primer a tots un `0` i després a tots un `1`:

- al començament, en el cas dels enters;
- al final, en el cas dels polinomis.

En considerar ordres diferents s'obtenen codis linealment equivalents. Ull! els textos de Ball [2] i Brunat-Ventura [3] ordenen com a enters. Aquí es farà servir l'ordre com a polinomis, que és el que considera Sage.

Definició 5.34. *(Codi de Reed-Muller) Sigui $0 \leqslant r \leqslant m$. El codi de Reed-Muller binari $\mathrm{RM}(r, m)$ és el conjunt de vectors binaris de longitud 2^m obtinguts avaluant els polinomis booleans $\boldsymbol{m}$ de grau $\leqslant r$ en tots els elements de $\mathbb{F}_2^m$:*

$$\mathrm{RM}(r, m) = \left\{ \left(\boldsymbol{m}(\boldsymbol{a}_1), \boldsymbol{m}(\boldsymbol{a}_2), \ldots, \boldsymbol{m}(\boldsymbol{a}_{2^m}) \right) : \boldsymbol{a}_i \in \mathbb{F}_2^m, \boldsymbol{m}(\boldsymbol{X}) \in \mathcal{B}[\boldsymbol{X}]_r \right\}.$$

És a dir, el conjunt de missatges $\mathcal{M}$ s'identifica amb l'espai $\mathcal{B}[\boldsymbol{X}]_r$, i cada element $\boldsymbol{m}$ d'aquest espai es codifica amb la paraula codi $\boldsymbol{c} = \mathrm{enc}(\boldsymbol{m}) \in \mathbb{F}_2^n$, de longitud $n = 2^m$, que té components $\mathtt{c}_{\boldsymbol{a}} = \boldsymbol{m}(\boldsymbol{a})$, les imatges dels índexs corresponents per l'aplicació induïda pel polinomi $\boldsymbol{m}$. La codificació és doncs l'aplicació d'avaluació de polinomis booleans.

Per exemple, el codi $\mathrm{RM}(1, 4)$ s'obté avaluant polinomis de grau $\leqslant 1$. Una base d'aquests polinomis són $1, X_1, X_2, X_3, X_4$. Tenint en compte el conveni sobre l'ordre dels elements de les bases la matriu generadora que conté a les files les avaluacions d'aquests monomis és:

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

La codificació $\mathrm{enc}(\boldsymbol{m})$ d'una paraula $\boldsymbol{m} = (\mathtt{m}_0, \mathtt{m}_0, \mathtt{m}_2, \mathtt{m}_3, \mathtt{m}_4) \in \mathbb{F}_2^5$ usant aquesta matriu generadora és la paraula codi obtinguda avaluant el polinomi booleà $\boldsymbol{m}(\boldsymbol{X}) = \mathtt{m}_0 + \mathtt{m}_1 X_1 + \mathtt{m}_2 X_2 + \mathtt{m}_3 X_3 + \mathtt{m}_4 X_4 \in \mathcal{B}[\boldsymbol{X}]_1$ en els elements de $\mathbb{F}_2^4$.

Observi's que es tenen inclusions $\mathrm{RM}(r, m) \subseteq \mathrm{RM}(r', m)$ sempre que $r \leqslant r'$ ja que $\mathcal{B}[\boldsymbol{X}]_r \subseteq \mathcal{B}[\boldsymbol{X}]_{r'}$. per als casos extrems del paràmetre r els codis són ben coneguts:

- $\mathrm{RM}(0,m) = \mathrm{Rep}_2(2^m)$: els únics polinomis booleans de grau $\leqslant 0$ són els constants;
- $\mathrm{RM}(m,m) = \{\mathtt{0},\mathtt{1}\}^{2^m}$: tota aplicació $\mathbb{F}_2^m \to \mathbb{F}_2$ s'obté amb un polinomi booleà.

De fet, totes les matrius generadores dels codis $\mathrm{RM}(r,m)$, per a un m fixat, s'obtenen agafant les primeres k files de la matriu de l'aplicació del lema 5.33, on $k = \sum_{i=0}^{r} \binom{m}{i}$ és la dimensió del codi.

De vegades s'accepta també el valor $r = -1$ en els paràmetres de Reed-Muller, entenent que $\mathrm{RM}(-1,m)$ és el codi trivial, que només conté la paraula zero, la qual s'obté avaluant l'únic polinomi de grau $-\infty \leqslant -1$: el polinomi zero.

Lema 5.35. *Els codis de Reed-Muller satisfan la recursivitat següent:*

$$\mathrm{RM}(r,m+1) = \big\{\boldsymbol{c} = \boldsymbol{c}_1 \| (\boldsymbol{c}_1 + \boldsymbol{c}_2) : \boldsymbol{c}_1 \in \mathrm{RM}(r,m), \boldsymbol{c}_2 \in \mathrm{RM}(r-1,m)\big\}.$$

Prova: Els polinomis en $m+1$ variables de grau $\leqslant r$ es poden escriure de la forma $\boldsymbol{m}(\boldsymbol{X}_{m+1}) = \boldsymbol{m}_1(\boldsymbol{X}_m) + \boldsymbol{m}_2(\boldsymbol{X}_m)X_{m+1}$, on $\boldsymbol{m}_1$ i $\boldsymbol{m}_2$ són polinomis en les m primeres variables que tenen graus $\leqslant k+1$ i $\leqslant k$, respectivament.

Es consideren els elements de $\mathbb{F}_2^{m+1}$ ordenats posant primer els que tenen última coordenada zero i després els que tenen última coordenada diferent de zero. Dels dos ordres considerats en el text això correspon a l'ordre que s'obté agafant els polinomis que representen els elements de $\mathbb{F}_2^{m+1}$: primer els de grau $\leqslant m-1$ (coeficient de X^m igual a zero) i després els de grau m (coeficient de X^m igual a $\mathtt{1}$). Aleshores en avaluar $\boldsymbol{m}$ en els primers, de la forma $\boldsymbol{a}\|\mathtt{0}$, s'obtenen els valors $\boldsymbol{m}_1(\boldsymbol{a})$, i en avaluar en els segons, de la forma $\boldsymbol{a}\|\mathtt{1}$, s'obtenen les sumes $\boldsymbol{m}_1(\boldsymbol{a}) + \boldsymbol{m}_2(\boldsymbol{a})$.

Per tant les paraules codi $\boldsymbol{c}$ de $\mathrm{RM}(r,m+1)$ són la concatenació d'una paraula $\boldsymbol{c}_1$ de $\mathrm{RM}(r,m)$ amb la suma d'aquesta paraula i una paraula $\boldsymbol{c}_2$ de $\mathrm{RM}(r-1,m)$, de totes les maneres possibles. □

Teorema 5.36 (Paràmetres dels codis RM)**.** *Els codis binaris de Reed-Muller* $\mathrm{RM}(r,m)$ *són codis lineals de tipus* $[n,k,d]$ *amb:*

$$n = 2^m, \qquad k = \sum_{i=0}^{r} \binom{m}{i}, \qquad d = 2^{m-r}.$$

Prova: Per construcció les paraules codi tenen 2^m components: tantes com elements $\boldsymbol{a} \in \mathbb{F}_2^m$ en què s'avaluen les funcions booleanes $\boldsymbol{m}(\boldsymbol{X}) \in \mathcal{B}[\boldsymbol{X}]_r$.

Com que l'avaluació és una bijecció del conjunt de polinomis booleans amb el d'aplicacions la dimensió és la de l'espai $\mathcal{B}[\boldsymbol{X}]_r$ de les funcions booleanes de grau $\leqslant r$, que és la donada a l'enunciat.

Per veure la distància mínima es fa inducció, primer sobre $r \geqslant 0$ i després sobre $m \geqslant r$. Per a $m = r$ el codi $\mathrm{RM}(m,m)$ és el codi total, que té distància mínima $1 = 2^{m-m}$. Suposant-ho cert per als r menors que el donat i per a un $m \geqslant r$ es considera el codi $\mathrm{RM}(r,m+1)$.

Usant el lema 5.35 les paraules codi són concatenacions $\boldsymbol{c} = \boldsymbol{c}_1\|(\boldsymbol{c}_1 + \boldsymbol{c}_2)$ amb $\boldsymbol{c}_1 \in \mathrm{RM}(r,m)$ i $\boldsymbol{c}_2 \in \mathrm{RM}(r-1,m)$. Observi's que, com que $\mathrm{RM}(r-1,m) \subseteq \mathrm{RM}(r,m)$ la suma també és $\boldsymbol{c}_1 + \boldsymbol{c}_2 \in \mathrm{RM}(r,m)$. Per hipòtesi d'inducció els pesos d'aquestes paraules satisfan $\|\boldsymbol{c}_1\| \geqslant 2^{m-r}$ i $\|\boldsymbol{c}_2\| \geqslant 2^{m-(r-1)} = 2^{m-r+1}$. Aleshores:

- si $\boldsymbol{c}_1 = \boldsymbol{0}$ aleshores $\|\boldsymbol{c}\| = \|\boldsymbol{c}_2\| \geqslant 2^{m-r+1}$;
- si $\boldsymbol{c}_1 \neq \boldsymbol{0}$ però $\boldsymbol{c}_1 + \boldsymbol{c}_2 = \boldsymbol{0}$, aleshores $\boldsymbol{c}_1 = \boldsymbol{c}_2 \in \mathrm{RM}(r-1, m)$ i $\|\boldsymbol{c}\| = \|\boldsymbol{c}_1\| \geqslant 2^{m-r+1}$;
- si cap dels dos és zero aleshores $\boldsymbol{c}_1, \boldsymbol{c}_1 + \boldsymbol{c}_2 \in \mathrm{RM}(r, m)$ són no nuls i també s'obté $\|\boldsymbol{c}\| = \|\boldsymbol{c}_1\| + \|\boldsymbol{c}_1 + \boldsymbol{c}_2\| \geqslant 2 \cdot 2^{m-r} = 2^{1+m-r}$.

De manera que el pes de tota paraula no nul·la del codi és sempre com a mínim $2^{(m+1)-r}$. □

La taula següent mostra el tipus de codi per a nombre de variables $2 \leqslant m \leqslant 5$:

	$m=2$	3	4	5
$r=0$	$[4,1,4]$	$[8,1,8]$	$[16,1,16]$	$[32,1,32]$
1	$[4,3,2]$	$[8,4,4]$	$[16,5,8]$	$[32,6,16]$
2	$[4,4,1]$	$[8,7,2]$	$[16,11,4]$	$[32,16,8]$
3		$[8,8,1]$	$[16,15,2]$	$[32,26,4]$
4			$[16,16,1]$	$[32,31,2]$
5				$[32,32,1]$

Producte bilineal. En l'espai $\mathrm{Map}(\mathbb{F}_2^m, \mathbb{F}_2)$ hi ha el producte bilineal ordinari que és la suma dels productes de components:

$$\langle \boldsymbol{x}, \boldsymbol{y} \rangle = \sum_{\boldsymbol{a} \in \mathbb{F}_2^m} \mathtt{x}_{\boldsymbol{a}} \mathtt{y}_{\boldsymbol{a}}, \qquad \boldsymbol{x} = (\mathtt{x}_{\boldsymbol{a}})_{\boldsymbol{a} \in \mathbb{F}_2^m}, \quad \boldsymbol{y} = (\mathtt{y}_{\boldsymbol{a}})_{\boldsymbol{a} \in \mathbb{F}_2^m}.$$

A través de l'avaluació de polinomis booleans aquest producte indueix un producte en l'espai dels polinomis booleans:

$$\big\langle f(\boldsymbol{X}), g(\boldsymbol{X}) \big\rangle = \big\langle \mathrm{av}(f), \mathrm{av}(g) \big\rangle = \sum_{\boldsymbol{a} \in \mathbb{F}_2^m} f(\boldsymbol{a}) g(\boldsymbol{a}) = \sum_{\boldsymbol{a} \in \mathbb{F}_2^m} fg(\boldsymbol{a}).$$

Observi's que la bilinealitat redueix el càlcul de productes bilineals de polinomis booleans a productes de monomis:

$$\Big\langle \sum \mathtt{f}_I \boldsymbol{X}^I, \sum \mathtt{g}_J \boldsymbol{X}^J \Big\rangle = \sum_{I,J} \mathtt{f}_I \mathtt{g}_J \langle \boldsymbol{X}^I, \boldsymbol{X}_J \rangle,$$

els quals es determinen en el següent:

Lema 5.37. *Per a tot $I, J \subseteq \{1, \ldots, m\}$ es té $\langle \boldsymbol{X}^I, \boldsymbol{X}^J \rangle = \mathtt{1} \Leftrightarrow I \cup J = \{1, \ldots, m\}$.*

Prova: El producte $\langle \boldsymbol{X}^I, \boldsymbol{X}^J \rangle$ és la suma d'avaluar $\boldsymbol{X}^{I \cup J}$ en tots els elements de $\mathbb{F}_2^m$.

Primer es veu que per a tot subconjunt $K \subseteq \{1, \ldots, m\}$ diferent del total la suma dels valors del monomi $\boldsymbol{X}^K$ en els elements de $\mathbb{F}_2^m$ és igual a zero. En efecte, si $i \notin K$ aleshores

$$\sum_{\boldsymbol{a} \in \mathbb{F}_2^m} \boldsymbol{a}^K = \sum_{\substack{\boldsymbol{a} \in \mathbb{F}_2^m \\ \mathtt{a}_i = 0}} \boldsymbol{a}^K + \sum_{\substack{\boldsymbol{a} \in \mathbb{F}_2^m \\ \mathtt{a}_i = 1}} \boldsymbol{a}^K = \mathtt{0},$$

ja que tots dos sumatoris contenen els mateixos sumands: com que la variable X_i no apareix en el monomi $\boldsymbol{X}^K$ el valor de $\mathtt{a}_i$ no afecta el resultat de l'avaluació $\boldsymbol{a}^K$.

En canvi, per al subconjunt $K = \{1, \ldots, m\}$ la suma val $\mathtt{1}$ ja que en avaluar el monomi $\boldsymbol{X}^K = X_1 X_2 \cdots X_m$ s'obté sempre $\mathtt{0}$ excepte en l'element $\boldsymbol{a} = \mathtt{11}\cdots\mathtt{1} \in \mathbb{F}_2^m$ en què val $\mathtt{1}$. □

Els resultats tècnics següents són la base de l'algorisme de descodificació de codis de Reed-Muller:

Lema 5.38. *Fixats un índex $\boldsymbol{e} \in \mathbb{F}_2^m$ i un subconjunt $J \subseteq \{1, \ldots, m\}$ existeix un únic element $\boldsymbol{u} \in \mathbb{F}_2^{|J|}$ tal que $\langle f_{\boldsymbol{e}}(\boldsymbol{X}), f_{J,\boldsymbol{u}}(\boldsymbol{X})\rangle = \mathtt{1}$.*

Prova: Cada producte $f_{\boldsymbol{e}}(\boldsymbol{a}) f_{J,\boldsymbol{u}}(\boldsymbol{a})$ és diferent de zero quan tots dos factors ho siguin. Això només passa quan el primer és no nul, que equival a què $\boldsymbol{a} = \boldsymbol{e}$, i el segon també és no nul, que equival a què $\boldsymbol{a}$ i $\boldsymbol{u}$ tenen les mateixes components per als índexs $i \in J$. Totes dues condicions es donen només quan $\boldsymbol{u} \in \mathbb{F}_2^{|J|}$ és l'element que té totes les components iguals a les de $\boldsymbol{e}$ per als índexs $i \in J$: $\mathtt{u}_i = \mathtt{e}_i$ per a tot $i \in J$ □

Lema 5.39. *Sigui $\boldsymbol{m}(\boldsymbol{X}) = \sum \mathtt{m}_I \boldsymbol{X}^I \in \mathcal{B}[\boldsymbol{X}]_r$. Per a tot $J \subseteq \{1, \ldots, m\}$ amb $|J| = r$ es té*

$$\langle \boldsymbol{m}(\boldsymbol{X}), f_{J^c,\boldsymbol{u}}(\boldsymbol{X})\rangle = \mathtt{m}_J, \qquad \forall \boldsymbol{u} \in \mathbb{F}_2^{|J^c|}.$$

Prova: El polinomi booleà $f_{J^c,\boldsymbol{u}}(\boldsymbol{X}) = \prod_{i\in J^c}(X_i + \mathtt{u}_i + 1)$ té un únic monomi de grau $|J^c| = m - r$: el monomi $\boldsymbol{X}^{J^c}$ producte de totes les variables amb índexs $i \in J^c$, i tots els demés monomis tenen grau estrictament inferior.

Per tant en el producte $\boldsymbol{m}(\boldsymbol{X}) f_{J^c,\boldsymbol{u}}(\boldsymbol{X}) = \sum \mathtt{m}_I \boldsymbol{X}^I f_{J^c,\boldsymbol{u}}(\boldsymbol{X})$ els monomis de tots els sumands amb $I \neq J$ tenen tots grau estrictament menor que m, i pel lema 5.37 el producte bilineal dona zero. En canvi, en el sumand amb $I = J$ apareix una única vegada el monomi $\boldsymbol{X}^{I\cup J^c} = X_1 \cdots X_m$, i novament pel lema 5.37 el producte és $\mathtt{1}$. En multiplicar pel coeficient $\mathtt{m}_I$ dona com a resultat aquest valor, tal com diu l'enunciat. □

Descodificació per majoria. El mètode de descodificació dels codis de Reed-Muller per majoria és el següent: Donada una paraula $\boldsymbol{x} = (\mathtt{x}_{\boldsymbol{a}})_{\boldsymbol{a}\in\mathbb{F}_2^m}$ de longitud 2^m, per a cada $J \subseteq \{1, \ldots, m\}$ amb $|J| = r$ es defineix la 2^{m-r}-tupla:

$$S_J(\boldsymbol{x}) = \left(\langle \boldsymbol{x}, P_{J^c,\boldsymbol{u}}\rangle = \sum_{\boldsymbol{a}\in\mathbb{F}_2^m} \mathtt{x}_{\boldsymbol{a}} P_{J^c,\boldsymbol{u}}(\boldsymbol{a}) \right)_{\boldsymbol{u}\in\mathbb{F}_2^{|J^c|}} \in \mathbb{F}_2^{|J^c|}.$$

És una $2^{|J^c|}$ tupla de dígits binaris que en certa manera fa el paper de síndrome en la descodificació dels codis de Reed-Solomon. L'algorisme de descodificació és el següent: sigui $\boldsymbol{c} = \mathrm{enc}(\boldsymbol{m})$ amb $\boldsymbol{m} = \sum_{|I|\leqslant r} \mathtt{m}_I \mathtt{X}^I$ la paraula codi enviada.

Inicialització: Es posa en un vector la paraula rebuda $\boldsymbol{x} = (\mathtt{x}_{\boldsymbol{a}})_{\boldsymbol{a}\in\mathbb{F}_2^m}$ i s'inicialitza un comptador amb $k = r$, el grau màxim dels monomis en la paraula codi.

Repetir: Mentre sigui $k \geqslant 0$,

Càlcul dels coeficients de grau k: Per a cada subconjunt $J \subseteq \{1, \dots, m\}$ amb $|J| = k$ es calcula el conjunt $S_J(\boldsymbol{x})$ i s'agafa el coeficient $\mathtt{m}'_J$ com el dígit binari majoritari en aquest conjunt.

Actualització de $\boldsymbol{x}$ i de k: Es resta a la paraula $\boldsymbol{x}$ el vector corresponent a l'avaluació del polinomi booleà $\sum_{|J|=k} \mathtt{m}'_J \boldsymbol{X}^J$ i es decrementa el comptador k.

Sortida: Es retorna el polinomi booleà $\sum_{|I|\leqslant r} \mathtt{m}'_I \boldsymbol{X}^I \in \mathcal{B}[\boldsymbol{X}]_r$.

Proposició 5.40. *L'algorisme tot just descrit corregeix els errors de transmissió sempre que n'hi hagi com a màxim*

$$\tau = \left\lfloor \frac{2^{m-r}-1}{2} \right\rfloor = 2^{m-r-1} - 1.$$

Prova: Si no hi ha errors de transmissió: $\boldsymbol{x} = \boldsymbol{c}$, aleshores tots els vectors $S_J(\boldsymbol{c})$ contenen sempre el coeficient $\mathtt{m}_J$ repetit $2^{|J^c|}$ vegades i la descodificació per majoria dona $\mathtt{m}'_J = \mathtt{m}_J$.

Els errors es poden entendre com que la paraula rebuda és la paraula codi a la qual s'han sumat uns quants vectors de la base canònica de $\mathbb{F}_2^m$, tants com errors:

$$\boldsymbol{x} = \boldsymbol{c} + \sum_{i=1}^{e} \mathrm{av}\big(f_{\boldsymbol{e}_i}(\boldsymbol{X})\big).$$

Els vectors $S_J(\boldsymbol{x})$ són clarament lineals respecte de $\boldsymbol{x}$. Pel lema 5.38 cada error que se suma modifica una única component en $S_J(\boldsymbol{x})$.

Si hi ha com a molt $\tau < 2^{m-r-1} - 1$ errors, com que aquests conjunts tenen $\geqslant 2^{m-r}$ components, menys de la meitat s'hauran vist afectades, i per tant la majoria de components seguirà contenint el coeficient: $\mathtt{m}'_J = \mathtt{m}_J$ per a tot J. □

5.7 Codis de Gallager (LDPC)

Referències: Ball [2, Chapter 8]

L'any 1960, a la seva tesi doctoral del M.I.T. (publicada el 1963 [9]), *Robert G. Gallager* fa una proposta de codis correctors d'errors lineals que permeten construir successions de codis asimptòticament bons: amb distància mínima relativa i ratio fitats inferiorment per valors positius. Aquests codis permeten apropar-se a la fita teòrica del teorema de codificació de canal de Shannon.

Es tracta de codis que es defineixen com el conjunt de solucions d'un sistema d'equacions lineals homogeni

$$\mathcal{C} = \{\boldsymbol{c} \in \mathbb{F}_2^n : \mathbf{H}\boldsymbol{c} = \mathbf{0}\}, \qquad \mathbf{H} \in \mathrm{Mat}_{m\times n}(\mathbb{F}_2)$$

amb matriu $\mathbf{H}$ que tingui la majoria d'entrades iguals a zero: una matriu *esparsa*, o de baixa densitat. Per aquesta raó es coneixen també com a *low-density parity-check codes*, o *codis LDPC*.

No es demana que les matrius $\mathbf{H}$ tinguin rang màxim, i per tant no sempre són matrius de control de paritat del codi, tot i que es poden convertir en matrius de control eliminant files corresponents a equacions dependents. La seva dimensió satisfà la desigualtat $k \geqslant n - m$.

Gallager proposa diferents construccions per a aquest tipus de codis, que tenen alguna component d'aleatorietat, inspirant-se en la demostració de Shannon del teorema de codificació de canal. També dona algorismes de descodificació, alguns d'ells especialment indicats per a la descodificació tova. Als anys 60 aquests codis no van rebre prou atenció ja que la tecnologia de què es disposava feia impensable implementar-los a la pràctica amb paràmetres raonables.

Van ser redescoberts a principis dels anys 90 per *MacKay* i *Neal*, que van veure que amb codis LDPC s'obtenien prestacions, en termes de capacitat correctora en relació amb la complexitat de la seva implementació, que superaven en molts casos les d'altres codis que s'estaven fent servir. Des d'aleshores s'han anat incorporant en diferents tecnologies, en alguns casos substituint altres classes de codis que es feien servir prèviament: *televisió via satèl·lit*, xarxes *Wi-Fi*, *ethernet*, etc.

Construcció de codis asimptòticament òptims. Es veurà a continuació una construcció de codis asimptòticament òptims proposada per Sipser i Spielman que prové de la teoria de grafs, concretament dels anomenats grafs bipartits expansius. Veure Ball [2, Chapter 8].

Per a aquests codis es poden considerar també les síndromes d'elements $\boldsymbol{x} \in \mathbb{F}_2^n$, definides com $\boldsymbol{s}(\boldsymbol{x}) = \mathbf{H}\boldsymbol{x}$, que es faran servir en l'algorisme de descodificació. Les paraules codi es caracteritzen per tenir síndrome zero i la síndrome de la paraula rebuda coincideix amb la de la paraula d'error.

Es denotaran $\mathbf{h}_j \in \mathbb{F}_2^m$ les columnes de $\mathbf{H}$. Es consideraran només matrius $\mathbf{H}$ tals que totes les seves columnes tinguin el mateix nombre w d'uns: $\|\mathbf{h}_j\| = w$ per a tot $j = 1, \ldots, m$. Per a cada subconjunt $J \subseteq \{1, \ldots, n\}$ es denotarà $\mathbf{H}_J$ la submatriu de mides $m \times |J|$ formada per les columnes indexades per elements de J. Es denotarà $\mathrm{nz}(J)$ el nombre de files no nul·les de la submatriu $\mathbf{H}_J$.

Definició 5.41. *Donats enters positius n, m i w i un nombre $\delta \in (0,1)$ es defineix un codi expansiu* $\mathrm{Expan}(n, m, w, \delta)$ *com un codi format per les solucions d'un sistema lineal homogeni de matriu $\mathbf{H} \in \mathrm{Mat}_{m\times n}$ amb totes les columnes de pes w i tal que*

$$\mathrm{nz}(J) > \frac{3}{4} w|J| \qquad \textit{per a tot } J \textit{ amb} \quad 1 \leqslant |J| \leqslant \delta n.$$

Aquesta darrera condició en la definició es pot entendre com que els uns de les columnes de la matriu $\mathbf{H}$ estiguin "ben repartits" entre les files.

Lema 5.42. *Els codis* $\mathrm{Expan}(n, m, w, \delta)$ *tenen distància mínima relativa $> \delta$.*

Prova: Equival a dir que la seva distància mínima és $d(\mathcal{C}) > \delta n$. Suposi's que aquesta distància fos $\leqslant \delta n$. Això equival a que existeixi una paraula codi de pes $1 \leqslant \|\boldsymbol{c}\| \leqslant \delta n$. Sigui J el conjunt de columnes de $\mathbf{H}$ corresponents a les posicions en què aquesta paraula té un 1, que conté $1 \leqslant |J| \leqslant \delta n$ elements. La condició que $\boldsymbol{c}$ és una paraula codi equival a que la suma d'aquestes columnes és el vector zero: $\sum_{j\in J} \mathbf{h}_j = \mathbf{0}$.

La submatriu $\mathbf{H}_J$ té $\mathrm{nz}(j) > \frac{3}{4}w|J|$ files no nul·les. El nombre total d'uns en aquesta submatriu és $w|J|$. Suposi's que totes les files no nul·les tinguessin almenys dos uns. Aleshores

el nombre total d'uns en la submatriu seria $w|J| \geqslant 2\mathrm{nz}(J) > 2\frac{3}{4}w|J| \Rightarrow 1 > \frac{3}{2}$, que és fals. Per tant hi ha d'haver files a $\mathbf{H}_J$ que contenen només un 1.

Però això contradiu el fet que la suma de les seves columnes hagi de ser zero, ja que en els índexs corresponents a files amb un únic 1 la suma és 1. Per tant no hi ha cap paraula codi no nul·la de pes $\leqslant \delta n$ i es dedueix que $d(\mathcal{C}) > \delta n$. □

Teorema 5.43 (Existència de codis)**.** *Siguin $w \geqslant 5$ i $R \in (0,1)$. Existeix un nombre $\delta \in (0,1)$, que depèn de w i de R, tal que per a tots els n suficientment grans existeix algun codi* $\mathrm{Expan}(n, m, w, \delta)$ *amb $m = \lfloor (1-R)n \rfloor$.*

Prova: És el lema 8.2 de Ball [2], on està enunciat i demostrat en termes de grafs bipartits expansius, i la demostració interpretada com una aplicació del mètode probabilista.

Es considera el conjunt $\mathcal{H}$ de totes les matrius $\mathbf{H} \in \mathrm{Mat}_{m\times n}$ que tenen totes les columnes de pes w. Per tal que hi hagi matrius com aquestes ha de ser $m = \lfloor (1-R)n \rfloor \geqslant w$. Això es compleix si $w \leqslant (1-R)n \Leftrightarrow n \geqslant \frac{w}{1-R}$. Això dona una primera condició que han de satisfer les longituds n. El nombre d'elements d'aquest conjunt és $|\mathcal{H}| = \binom{m}{w}^n$.

Es fixa un enter ν tal que $1 - R - \frac{1}{\nu} > 0$, de manera que per a tot $n \geqslant \nu$ es tingui la desigualtat $m > (1-R)n - 1 = n(1 - R - \frac{1}{n}) \geqslant n(1 - R - \frac{1}{\nu})$. Aquesta desigualtat, que es farà servir més endavant, imposa una nova condició sobre els enters n per als quals existeixen codis: que sigui $n \geqslant \nu$.

Sigui $\delta \in (0,1)$. Es vol veure que el conjunt $\mathcal{H}$ conté matrius que puguin complir la propietat de la definició 5.41. Per a això cal que el nombre de files sigui suficient per satisfer la condició que existeix alguna matriu en aquest conjunt que té $\mathrm{nz}(J) > \frac{3}{4}w|J|$ per a tot subconjunt J amb $1 \leqslant |J| \leqslant n\delta$. Això imposa una nova condició en el nombre de files: ha de ser $m \geqslant \mathrm{nz}(J) > \frac{3}{4}w|J|$ per als J dins de l'interval indicat. Aquesta condició es compleix sempre que $m > n(1 - R - \frac{1}{\nu}) \geqslant \frac{3}{4}wn\delta \geqslant \frac{3}{4}w|J|$, que equival a la condició

$$\delta \leqslant \frac{4(1 - R - \frac{1}{\nu})}{3w}.$$

Se suposarà en endavant que δ satisfà aquesta condició.

Ara s'ha de veure que el subconjunt $\mathcal{A} \subseteq \mathcal{H}$ de les matrius que tenen la propietat de la definició 5.41 és no buit. De manera equivalent, que el subconjunt $\mathcal{A}^c$ és diferent del total.

Les matrius de $\mathcal{A}^c$ són aquelles per a les quals existeix un subconjunt $J \subseteq \{1, \ldots, n\}$ amb $1 \leqslant |J| \leqslant \delta n$ elements tal que $\mathrm{nz}(J) \leqslant \frac{3}{4}w|J| \Leftrightarrow \mathrm{nz}(J) \leqslant \left\lfloor \frac{3}{4}w|J| \right\rfloor$. Això vol dir que tots els uns de la matriu $\mathbf{H}_J$ estan en les files amb índexs en un subconjunt d'índexs $I \subseteq \{1, \ldots, m\}$ que conté $|I| = \left\lfloor \frac{3}{4}w|J| \right\rfloor$ elements. Sigui $\mathcal{A}_{I,J} \subseteq \mathcal{H}$ el subconjunt de les matrius que tenen aquesta propietat per a subconjunts $J \subseteq \{1, \ldots, n\}$ i $I \subseteq \{1, \ldots, m\}$ fixats, amb nombres d'elements $1 \leqslant |J| \leqslant \delta n$ i $|I| \leqslant \left\lfloor \frac{3}{4}w|J| \right\rfloor$. El seu nombre d'elements és

$$|\mathcal{A}_{I,J}| = \binom{m}{w}^{n-|J|} \binom{|I|}{w}^{|J|} = \binom{m}{w}^n \left(\frac{\binom{|I|}{w}}{\binom{m}{w}} \right)^{|J|} \leqslant \binom{m}{w}^n \left(\frac{|I|}{m} \right)^{w|J|}.$$

La primera igualtat s'obté comptant: en les $n - |J|$ columnes amb índex fora de J els w uns es poden repartir arbitràriament en les m files disponibles; en les $|J|$ columnes amb índex en

J s'han de posar obligatòriament en files amb índexs en I. La darrera desigualtat és general per a quocients de coeficients binomials: per a enters $0 \leqslant c \leqslant b \leqslant a$ qualssevol es té

$$\frac{\binom{b}{c}}{\binom{a}{c}} = \frac{b(b-1)\cdots(b-c+1)}{a(a-1)\cdots(a-k+1)} \leqslant \left(\frac{b}{a}\right)^c.$$

Aleshores, com que $\mathcal{A}^c \subseteq \cup_{I,J}\mathcal{A}_{I,J}$, es té

$$|\mathcal{A}^c| \leqslant \sum_{1\leqslant|J|\leqslant\delta n} \sum_{|I|=\left\lfloor\frac{3}{4}w|J|\right\rfloor} |\mathcal{A}_{I,J}| \leqslant \binom{m}{w}^n \sum_{|J|=1}^{\lfloor\delta n\rfloor} \binom{n}{|J|}\binom{m}{|I|}\left(\frac{|I|}{m}\right)^{w|J|}.$$

A partir de la sèrie de potències per a la funció exponencial es troba la fita següent:

$$e^k = \sum_{i=0}^{\infty}\frac{k^i}{i!} > \frac{k^k}{k!} \quad \Rightarrow \quad \frac{1}{k!} < \left(\frac{e}{k}\right)^k, \qquad \forall k \geqslant 1,$$

de la qual es dedueix la fita per als coeficients binomials següent:

$$\binom{n}{k} = \frac{n(n-1)\cdots(n-k+1)}{k!} \leqslant \frac{n^k}{k!} < \left(\frac{ne}{k}\right)^k, \qquad \forall n \geqslant k \geqslant 1.$$

Aplicant-la als dos coeficients biniomials del sumatori de l'expressió anterior, es té

$$\begin{aligned}|\mathcal{A}^c| &< \binom{m}{w}^n \sum_{|J|=1}^{\lfloor\delta n\rfloor}\left(\frac{ne}{|J|}\right)^{|J|}\left(\frac{me}{|I|}\right)^{\frac{3}{4}w|J|}\left(\frac{|I|}{m}\right)^{w|J|} \\ &= \binom{m}{w}^n \sum_{|J|=1}^{\lfloor\delta n\rfloor}\left(e^{1+\frac{3}{4}w}\right)^{|J|}\left(\frac{n}{|J|}\right)^{|J|}\left(\frac{|I|}{m}\right)^{\frac{1}{4}w|J|}.\end{aligned}$$

on a la desigualtat, en el quocient del mig s'ha fet servir que la funció exponencial de base $\frac{me}{|I|} > 1$ és creixent, junt amb la desigualtat $|I| = \left\lfloor\frac{3}{4}w|J|\right\rfloor \leqslant \frac{3}{4}w|J|$. Usant la desigualtat $m > n(1-R-\frac{1}{\nu})$ en el factor del mig i $|I| \leqslant \frac{3}{4}w|J|$ en el de la dreta, queda

$$\begin{aligned}|\mathcal{A}^c| &< \binom{m}{w}^n \sum_{|J|=1}^{\lfloor\delta n\rfloor}\left(e^{1+\frac{3}{4}w}\right)^{|J|}\left(\frac{m}{(1-R-\frac{1}{\nu})|J|}\right)^{|J|}\left(\frac{\frac{3}{4}w|J|}{m}\right)^{\frac{1}{4}w|J|} \\ &= \binom{m}{w}^n \sum_{|J|=1}^{\lfloor\delta n\rfloor}\left(e^{1+\frac{3}{4}w}\right)^{|J|}\left(\frac{1}{1-R-\frac{1}{\nu}}\right)^{|J|}\left(\frac{3}{4}w\right)^{\frac{1}{4}w|J|}\left(\frac{|J|}{m}\right)^{(\frac{1}{4}w-1)|J|}.\end{aligned}$$

Tenint en compte que l'exponent $(\frac{1}{4}w-1)|J|$ és positiu gràcies a la hipòtesi $w \geqslant 5$, a partir de les desigualtats $|J| \leqslant \delta n$ i $m > n(1-R-\frac{1}{\nu})$, s'obté

$$|\mathcal{A}^c| < \binom{m}{w}^n \sum_{|J|=1}^{\lfloor \delta n \rfloor} \left(e^{1+\frac{3}{4}w}\right)^{|J|} \left(\frac{1}{1-R-\frac{1}{\nu}}\right)^{\frac{1}{4}w|J|} \left(\frac{3}{4}w\right)^{\frac{1}{4}w|J|} \delta^{(\frac{1}{4}w-1)|J|}$$

$$= \binom{m}{w}^n \sum_{|J|=1}^{\lfloor \delta n \rfloor} \left(C(w,R)\delta^{\frac{1}{4}w-1}\right)^{|J|} < \binom{m}{w}^n \sum_{|J|=1}^{\infty} \left(C(w,R)\delta^{\frac{1}{4}w-1}\right)^{|J|}$$

$$= \binom{m}{w}^n \frac{x}{1-x},$$

amb $C(w,R) = \left(e^{1+\frac{3}{4}w}\right)\left(\frac{1}{1-R-\frac{1}{\nu}}\right)^{\frac{1}{4}w}\left(\frac{3}{4}w\right)^{\frac{1}{4}w} > 0$ un nombre positiu que només depèn de w i R, i on s'ha posat $x = C(w,R)\delta^{\frac{1}{4}w-1}$. Agafant $\delta \in (0,1)$ prou petit es pot aconseguir fer $x < \frac{1}{2}$. Per a aquest δ es té $\frac{x}{1-x} < 1$ i

$$|\mathcal{A}^c| < \binom{m}{w}^n = |\mathcal{H}| \quad \Rightarrow \quad \mathcal{A} \neq \varnothing$$

i existeixen codis $\mathrm{Expan}(n,m,w,\delta)$ amb la propietat de la definició 5.41 i amb $m = \lfloor (1-R)n \rfloor$ per a totes les longituds n suficientment grans. □

Corol·lari 5.44 (Codis asimptòticament bons)**.** *Per a tot $R \in (0,1)$ existeix un $\delta \in (0,1)$ tal que existeixen famílies de codis $\mathcal{C}_n = \mathrm{Expan}(n,m,w,\delta)$ per a tot n suficientment gran, amb ratio $R(\mathcal{C}_n) \geqslant R > 0$ i distància mínima relativa $\delta(\mathcal{C}_n) > \delta > 0$.*

Prova: Donat un ratio $R \in (0,1)$ es pot agafar $w \geqslant 5$ qualsevol i aplicar el teorema 5.43. Com que $m = \lfloor (1-R)n \rfloor \leqslant (1-R)n$, el ratio d'aquests codis és

$$\frac{k}{n} \geqslant \frac{n-m}{n} \geqslant \frac{n-(1-R)n}{n} = R.$$

Al lema 5.42 s'ha vist que la distància mínima d'aquests codis és $> n\delta$ i, per tant, la distància mínima relativa és $> \delta$. □

Algorisme de descodificació. A continuació es donarà un algorisme de descodificació eficient per a aquest tipus de codis. Donat un codi $\mathcal{C} = \mathrm{Expan}(n,m,w,\delta)$ definit per la matriu $\mathbf{H} \in \mathrm{Mat}_{m\times n}(\mathbb{F}_2)$ s'anomenarà síndrome d'una paraula $\boldsymbol{x} \in \mathbb{F}_2^n$ el vector

$$\mathrm{syn}(\boldsymbol{x}) = \mathbf{s} = \mathbf{H}\cdot\boldsymbol{x} \in \mathbb{F}_2^m,$$

com si la matriu $\mathbf{H}$ fos una matriu de control de paritat, tot i que ara pot no ser-ho. Naturalment, les paraules codi $\boldsymbol{c} \in \mathcal{C}$ es caracteritzen per tenir síndrome zero. L'única diferència entre que $\mathbf{H}$ sigui o no matriu de control és que la imatge de l'aplicació lineal $\mathrm{syn}\colon \mathbb{F}^n \to \mathbb{F}^m$ sigui o no l'espai total: que totes les paraules $\boldsymbol{s} \in \mathbb{F}^m$ siguin síndrome d'alguna paraula $\boldsymbol{x} \in \mathbb{F}^n$ o no. Es denotaran com és habitual $\boldsymbol{e}_i$ els elements de la base canònica de $\mathbb{F}_2^m$, que són les paraules de pes 1.

Lema 5.45. *Sigui $\boldsymbol{x} \in \mathbb{F}_2^n$ una paraula a distància $1 \leqslant d(\boldsymbol{x}, \mathcal{C}) \leqslant \delta n$. Aleshores,*

1. *$wd(\boldsymbol{x}, \mathcal{C}) \geqslant \|\operatorname{syn}(\boldsymbol{x})\| > \frac{1}{2} wd(\boldsymbol{x}, \mathcal{C})$;*
2. *$\|\operatorname{syn}(\boldsymbol{x} + \boldsymbol{e}_j)\| < \|\operatorname{syn}(\boldsymbol{x})\|$ per a algun índex $1 \leqslant j \leqslant n$.*

PROVA: Sigui $\boldsymbol{c}$ una paraula codi a distància $d(\boldsymbol{x}, \boldsymbol{c}) = d(x, \mathcal{C}) \leqslant \delta n$. Sigui J el conjunt de columnes de la matriu $\mathbf{H}$ corresponents a les posicions en què $\boldsymbol{x}$ i $\boldsymbol{c}$ difereixen. Per hipòtesi té nombre d'elements $1 \leqslant |J| = d(\boldsymbol{x}, \boldsymbol{c}) \leqslant \delta n$.

Es considera la matriu $\mathbf{H}_J$. La síndrome $\operatorname{syn}(\boldsymbol{x})$ és la suma de les seves columnes. La propietat de la definició 5.41 assegura que aquesta matriu té $\operatorname{nz}(J) > \frac{3}{4} w|J|$ files no nul·les. Siguin s i p el nombre de files no nul·les de paritat senar i parell. Es té $s + p = \operatorname{nz}(J) > \frac{3}{4} w|J|$. El nombre d'uns en la matriu $\mathbf{H}_J$ és $w|J| \geqslant s + 2p$. D'aquesta desigualtat es dedueix que $s \leqslant s + 2p \leqslant w|J| = wd(\boldsymbol{x}, \mathcal{C})$. A partir de totes dues desigualtats es dedueix:

$$w|J| \geqslant p + p + s > p + \frac{3}{4} w|J| \quad \Rightarrow \quad p < \frac{1}{4} w|J| \quad \Rightarrow \quad s > \frac{1}{2} w|J| = \frac{1}{2} wd(\boldsymbol{x}, \mathcal{C}).$$

Però el nombre s de files de pes senar és el pes de la síndrome $\operatorname{syn}(\boldsymbol{x})$. Això demostra el primer apartat.

La síndrome és la suma de les síndromes dels errors:

$$\operatorname{syn}(\boldsymbol{x}) = \operatorname{syn}(\boldsymbol{x} + \boldsymbol{c}) = \sum_{j \in J} \operatorname{syn}(\boldsymbol{e}_j).$$

Per a tota component de $\operatorname{syn}(\boldsymbol{x})$ que sigui igual a `1` existeix almenys alguna $\operatorname{syn}(\boldsymbol{e}_j)$ que té un `1` en aquesta component. Totes les síndromes $\operatorname{syn}(\boldsymbol{e}_j)$ tenen pes w. Suposi's que, per a tot $j \in J$, la síndrome $\operatorname{syn}(\boldsymbol{e}_j)$ té $\leqslant \frac{1}{2} w$ uns en comú amb $\operatorname{syn}(\boldsymbol{x})$. Aleshores el nombre d'uns de $\operatorname{syn}(\boldsymbol{x})$ seria $\|\operatorname{syn}(\boldsymbol{x})\| \leqslant \frac{1}{2} w|J| = \frac{1}{2} wd(\boldsymbol{x}, \mathcal{C})$, que contradiu el primer apartat.

Per tant, existeix alguna j tal que $\operatorname{syn}(\boldsymbol{e}_j)$ té $\nu > \frac{1}{2} w$ uns en comú amb $\operatorname{syn}(\boldsymbol{x})$. En sumar les dues paraules $\operatorname{syn}(\boldsymbol{x}) + \operatorname{syn}(\boldsymbol{e}_j)$ hi haurà ν uns de $\operatorname{syn}(\boldsymbol{x})$ que passaran a ser en zeros i $w - \nu$ zeros que passaran a ser uns. Per tant,

$$\|\operatorname{syn}(\boldsymbol{x}+\boldsymbol{e}_j)\| = \|\operatorname{syn}(\boldsymbol{x})+\operatorname{syn}(\boldsymbol{e}_j)\| = \|\operatorname{syn}(\boldsymbol{x})\| - \nu + w - \nu = \|\operatorname{syn}(\boldsymbol{x})\| + w - 2\nu < \|\operatorname{syn}(\boldsymbol{x})\|.$$

Per tant aquest vector $\boldsymbol{e}_j$ satisfà la condició del segon apartat. □

Aquest lema justifica l'algorisme de descodificació per reducció del pes de les síndromes següent:

INICIALITZACIÓ: S'inicialitza un vector $\boldsymbol{x}$ amb la paraula rebuda.

REPETIR: Mentre $\operatorname{syn}(\boldsymbol{x}) > 0$:

- PER A $j = 1, \ldots, n$: Si $\|\operatorname{syn}(\boldsymbol{x} + \boldsymbol{e}_j)\| < \|\operatorname{syn}(\boldsymbol{x})\|$ es canvia $\boldsymbol{x}$ per $\boldsymbol{x} + \boldsymbol{e}_j$ i es torna al pas de REPETIR.
- ERROR: Si s'ha passat per tots els índexs j sense trobar cap $\boldsymbol{e}_j$ que rebaixi la síndrome s'acaba l'algorisme declarant que s'ha produït un error que no es pot corregir.

SORTIDA: Es retorna la paraula $\boldsymbol{x}$.

És clar que aquest algorisme acaba sempre en un nombre finit de passos: cada vegada que es troba un vector $\boldsymbol{e}_j$ amb $\|\operatorname{syn}(\boldsymbol{x}+\boldsymbol{e}_j)\| < \|\operatorname{syn}(\boldsymbol{x})\|$ es rebaixa estrictament el pes de la síndrome. Per tant, o bé en algun pas no es trobarà cap $\boldsymbol{e}_j$ així, i es retorna error, o bé l'algorisme acaba retornant una paraula de síndrome zero: una paraula codi.

En la recerca del vector $\boldsymbol{e}_j$ que rebaixa la síndrome es pot optar per recórrer tots aquests vectors i sumar a $\boldsymbol{x}$ aquell que rebaixi més la síndrome, sempre que n'hi hagi algun.

Proposició 5.46. *Aquest algorisme corregeix sempre correctament fins a* $\frac{1}{2}\delta n$ *errors.*

PROVA: Teorema 8.7 de Ball [2]; l'argument en aquesta referència no m'acaba de convèncer. Sigui $\boldsymbol{c}$ la paraula codi enviada i sigui $\boldsymbol{y} \in \mathbb{F}_2^n$ la paraula rebuda. Es denotarà $\boldsymbol{x}$ la paraula que s'inicialitza com a $\boldsymbol{y}$ es va modificant durant l'execució. La hipòtesi és que $d(\boldsymbol{y},\boldsymbol{c}) \leqslant \frac{1}{2}\delta n$; és a dir, que s'han produït com a màxim $\frac{1}{2}\delta n$ errors.

La distància $d(\boldsymbol{y},\boldsymbol{c})$ també satisfà $d(\boldsymbol{y},\boldsymbol{c}) \leqslant \frac{1}{2}\lfloor \delta n \rfloor$. En efecte, en ser un enter, compleix $d(\boldsymbol{y},\boldsymbol{c}) \leqslant \lfloor \frac{1}{2}\delta n \rfloor$. Es comprova fàcilment que, per a tot nombre real $\alpha \geqslant 0$, es té $\lfloor \frac{1}{2}\alpha \rfloor \leqslant \frac{1}{2}\lfloor \alpha \rfloor$.

Es considera el procés de descodificació donat a l'algorisme. La síndrome de la paraula inicial satisfà $\|\operatorname{syn}(\boldsymbol{y})\| \leqslant w d(\boldsymbol{y},\boldsymbol{c}) \leqslant w\frac{1}{2}\lfloor \delta n \rfloor$ i en cada pas disminueix. Per tant $\|\operatorname{syn}(\boldsymbol{x})\| \leqslant w\frac{1}{2}\lfloor \delta n \rfloor$ per la paraula $\boldsymbol{x}$ en tots els passos.

Ara s'ha de controlar la distància $d(\boldsymbol{x},\boldsymbol{c})$ de totes les paraules $\boldsymbol{x}$ que es van calculant durant l'algorisme. Aquesta distància, de fet, pot ser que no disminueixi en cada pas sinó que de vegades augmenti. En efecte, en la demostració del lema 5.45 s'agafa un $\boldsymbol{e}_j$ que correspon a un error, de manera que $d(\boldsymbol{x}+\boldsymbol{e}_j,\boldsymbol{c}) = d(\boldsymbol{x},\boldsymbol{c}) - 1$. Això és gràcies a què en aquest lema es parteix del coneixement de la paraula $\boldsymbol{c}$, o sigui, el conjunt $|J|$ de les posicions on hi ha errors, però ara aquesta informació no es té. Podria haver-hi un $\boldsymbol{e}_k$ per a un $k \notin J$ que també rebaixi el pes de la síndrome. Si s'agafa aquest k la distància a la paraula $\boldsymbol{c}$ augmenta, ja que no correspon a una de les posicions on hi ha hagut error. De fet, això pot passar: que s'introdueixin nous errors durant l'execució de l'algorisme. Si és el cas aquests errors es corregeixen en algun pas posterior ja que, tal com es veurà a continuació, l'algorisme sempre acaba en la paraula $\boldsymbol{c}$.

Durant tot l'algorisme la distància $d(\boldsymbol{x},\boldsymbol{c})$ es manté sempre $< \lfloor \delta n \rfloor$. En el primer pas gràcies a la hipòtesi $d(\boldsymbol{y},\boldsymbol{c}) \leqslant \lfloor \frac{1}{2}\delta n \rfloor < \lfloor \delta n \rfloor$. Com que en cada pas la distància disminueix o augmenta en una unitat, per poder arribar ser $\geqslant \lfloor \delta n \rfloor$ en algun pas hauria de ser igual. Si durant l'algorisme en algun moment arribés a ser $d(\boldsymbol{x},\boldsymbol{c}) = \lfloor \delta n \rfloor$ el lema 5.45 es podria seguir aplicant i es tindria

$$w\frac{1}{2}\lfloor \delta n \rfloor \geqslant \|\operatorname{syn}(\boldsymbol{x})\| > \frac{1}{2} w d(\boldsymbol{x},\boldsymbol{c}) = \frac{1}{2} w \lfloor \delta n \rfloor,$$

que és una contradicció amb el que s'ha vist abans que $\|\operatorname{syn}(\boldsymbol{x})\| \leqslant w\frac{1}{2}\lfloor \delta n \rfloor$ durant tot l'algorisme.

Per tant tota l'estona es té $d(\boldsymbol{x},\boldsymbol{c}) < \lfloor \delta n \rfloor \leqslant \delta n$. Amb això s'asseguren dues coses. Primera: que es pot aplicar el lema 5.45 i en tots els passos de l'algorisme es trobarà un vector $\boldsymbol{e}_j$ de pes 1 tal que, en sumar-lo a $\boldsymbol{x}$, rebaixa la síndrome; per tant l'algorisme no acaba mai amb error i retorna una paraula codi. Segona: la paraula retornada ha de ser

necessàriament $\boldsymbol{c}$ ja que, tot i que durant l'execució la $\boldsymbol{x}$ se'n pugui allunyar, sempre es manté a distància $d(\boldsymbol{x}, \boldsymbol{c}) < \delta n < d(\mathcal{C})$ i per tant és impossible que $\boldsymbol{x}$ acabi sent una altra paraula codi diferent de $\boldsymbol{c}$. □

5.8 Problemes Complementaris

5.23. Perquè si la matriu generadora d'un codi lineal té una columna igual al vector zero $\mathbf{0} \in \mathbb{F}^k$ aquest codi no és eficient?

Sigui $\mathcal{C}$ un codi de tipus $[n, k, d]_q$ amb matriu generadora sense columnes zero. Demostreu que la suma dels pesos de Hamming de totes les paraules és

$$\sum_{\boldsymbol{c} \in \mathcal{C}} w(\boldsymbol{c}) = nq^{k-1}(q-1).$$

5.24. Quantes matrius generadores diferents (i, per tant, aplicacions de codificació) té un codi lineal de tipus $[n, k, d]_q$?

5.25. De les afirmacions següents digueu quines són certes i quines falses, justificant la resposta:

1. Les matrius següents són la matriu generadora i la matriu de control de paritat d'un mateix codi binari:
$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{bmatrix}, \qquad \mathbf{H} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$
2. El codi binari lineal amb matriu de control de paritat
$$\mathbf{H} = \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$
té dimensió 3 i distància mínima 4.
3. No existeix cap codi binari de longitud $n = 16$, dimensió 11 i distància mínima 4.

5.26. Es vol transmetre una seqüència binària per un canal binari simètric. A fi de reduir errors en el receptor, es decideix que cada quatre símbols transmesos $\mathtt{x}_1, \mathtt{x}_2, \mathtt{x}_3, \mathtt{x}_4 \in \{\mathtt{0}, \mathtt{1}\}$ aniran seguits d'uns altres quatre que es calculen com a combinació lineal d'aquests:

$$\begin{aligned} \mathtt{p}_1 &= \mathtt{x}_1 + \mathtt{x}_2 + \mathtt{x}_4 \\ \mathtt{p}_2 &= \mathtt{x}_1 + \mathtt{x}_3 + \mathtt{x}_4 \\ \mathtt{p}_3 &= \mathtt{x}_1 + \mathtt{x}_2 + \mathtt{x}_3 \\ \mathtt{p}_4 &= \mathtt{x}_2 + \mathtt{x}_3 + \mathtt{x}_4. \end{aligned}$$

Es demana:

1. Demostreu que la paraula rebuda $\boldsymbol{x} = \mathtt{01000101}$ conté errors i corregiu-la per proximitat.
2. Es pot assegurar que la correcció dona com a resultat la paraula transmesa?
3. Quants errors es poden corregir com a màxim en una paraula rebuda fent servir aquest codi?
4. El codi, es perfecte? Raoneu-ho. Quin és l'avantatge dels codis perfectes?
5. Doneu l'expressió d'una fita de la probabilitat de detectar que s'ha rebut una paraula amb error però no poder corregir-la, quan la transmissió es fa en un canal binari simètric de probabilitat d'error p.
6. Un codi més fàcil d'implementar és aquell en el que cada quatre símbols emesos es repeteixen. Quants errors pot corregir aquest codi?

5.27. Comproveu que els duals de codis de Reed-Muller també ho són:

$$\mathrm{RM}(r,m)^{\perp} = \mathrm{RM}(m-r-1,m).$$

5.28. Comproveu que per a $0 \leqslant r \leqslant m$, les matrius $\mathbf{G}(r,m)$ definides de manera recurrent com:

$$\begin{aligned}
\mathbf{G}(0,m) &= \begin{pmatrix} \mathtt{1} & \mathtt{1} & \cdots & \mathtt{1} \end{pmatrix} \in \mathrm{Mat}_{1\times 2^m}(\mathbb{F}_2), \quad m \geqslant 0,\\
\mathbf{G}(m,m) &= \left[\begin{array}{c} \mathbf{G}(m-1,m) \\ \hline \mathtt{0}\ \mathtt{0}\ \cdots\ \mathtt{0}\ \mathtt{1} \end{array}\right], \quad m \geqslant 1,\\
\mathbf{G}(r,m) &= \left[\begin{array}{c|c} \mathbf{G}(r,m-1) & \mathbf{G}(r,m-1) \\ \hline \mathbf{0} & \mathbf{G}(r-1,m-1) \end{array}\right], \quad 0 < r < m,
\end{aligned}$$

són matrius generadores dels codis de Reed-Muller.

5.29. La matriu generadora següent defineix un codi binari $\mathcal{C}$:

$$\mathbf{G} = \begin{bmatrix} \mathtt{1} & \mathtt{0} & \mathtt{1} & \mathtt{1} & \mathtt{0} \\ \mathtt{0} & \mathtt{1} & \mathtt{1} & \mathtt{0} & \mathtt{1} \end{bmatrix}.$$

1. Quines són les paraules codi?
2. Trobeu una matriu de control de paritat.
3. Quants vectors d'error pot detectar? Quants en pot corregir?

6 Codis polinomials

En els codis lineals $\mathcal{C} \subseteq \mathbb{F}^n$ l'estructura d'espai vectorial proporciona eines tant per a la construcció i estudi dels codis mateixos com per a la codificació i descodificació.

Es pot afegir fàcilment més estructura a l'espai $\mathbb{F}^n$ i els seus subespais interpretant les coordenades dels seus elements com els coeficients d'un polinomi. És a dir, identificant l'espai n-dimensional $\mathbb{F}^n$ amb l'espai $\mathbb{F}[X]_n$ dels polinomis amb coeficients en $\mathbb{F}$ de grau $< n$.

Quan es considera aquesta estructura addicional, interpretant les paraules com a polinomis, se sol parlar de *codis polinomials* o *codis algebraics*. Ara es poden usar les operacions, tècniques i propietats de la teoria de polinomis: producte, divisió euclidiana, màxim comú divisor, algorisme d'Euclides, identitat de Bézout, arrels, avaluació, interpolació, etc.

Espai de polinomis. Es recorden aquí algunes definicions i notacions dels polinomis. Sigui $\mathbb{F} = \mathbb{F}_q$ un cos finit de q elements. Es denota $\mathbb{F}[X]$ l'*anell de polinomis* amb coeficients en $\mathbb{F}$ en la variable X:

$$f(X) = \mathtt{a}_0 + \mathtt{a}_1 X + \cdots \mathtt{a}_r X^r, \qquad \mathtt{a}_i \in \mathbb{F}, \quad r \geqslant 0.$$

Per a cada enter $n \geqslant 1$ es denota $\mathbb{F}[X]_n$ el subconjunt de $\mathbb{F}[X]$ format pels polinomis de grau més petit que n. La suma de polinomis i el producte per escalars (elements de $\mathbb{F}$) donen a $\mathbb{F}[X]_n$ estructura d'espai vectorial. És un espai de dimensió n on s'agafen com a base canònica els monomis $1, X, X^2, \ldots, X^{n-1}$ de grau $< n$.

Els elements de $\mathbb{F}[X]_n$ s'identifiquen amb els vectors de $\mathbb{F}^n$, i per tant amb les paraules de longitud n sobre l'alfabet $\mathbb{F}$, de la manera natural:

$$\boldsymbol{u}(X) = \mathtt{u}_0 + \mathtt{u}_1 X + \cdots + \mathtt{u}_{n-1} X^{n-1} \in \mathbb{F}[X]_n \quad \approx \quad \boldsymbol{u} = (\mathtt{u}_0, \mathtt{u}_1, \ldots, \mathtt{u}_{n-1}) \in \mathbb{F}^n.$$

Aquesta correspondència entre polinomis i vectors conserva les estructures d'espai vectorial en tots dos conjunts. Transforma la base dels monomis de $\mathbb{F}[X]_n$ en la base canònica de $\mathbb{F}^n$: per a tot índex $i = 1, \ldots, n$ fa correspondre el monomi X^{i-1} amb el vector $\boldsymbol{e}_i$ que té totes les coordenades zero excepte la i-èsima, que és igual a 1.

Considerant només l'estructura d'espai vectorial, amb tots dos conjunts $\mathbb{F}^n$ i $\mathbb{F}[X]_n$ es poden fer les mateixes coses. En canvi, a l'espai dels polinomis el producte afegeix estructura que es pot aprofitar. D'entrada es poden multiplicar dos polinomis $\boldsymbol{u}(X), \boldsymbol{v}(X) \in \mathbb{F}[X]_n$ obtenint un resultat $\boldsymbol{u}(X)\boldsymbol{v}(X) \in \mathbb{F}[X]_n$ sempre que la suma dels graus sigui $\deg \boldsymbol{u} + \deg \boldsymbol{v} < n$. Aquesta restricció es pot evitar agafant un polinomi mòdul $N(X)$ de grau $\deg N = n$ i veient $\mathbb{F}[X]_n$ com els elements de l'anell $\mathbb{F}[X]_{N(X)}$ de les classes de congruència mòdul $N(X)$. Aquí el producte de dos polinomis ja està sempre definit: quan el grau de $\boldsymbol{u}(X)\boldsymbol{v}(X)$ és $\geqslant n$ s'agafa com a resultat el reste de dividir aquest producte pel polinomi $N(X)$ que fa de mòdul.

Es tenen, per tant, identificacions del conjunt de les paraules de longitud n amb coeficients en el cos $\mathbb{F}$ amb els dos conjunts següents:

$$\mathbb{F}^n \quad \approx \quad \mathbb{F}[X]_n \quad \approx \quad \mathbb{F}[X]_{N(X)}.$$

A la secció 6.1 s'estudien els codis de Reed-Solomon en la seva versió original, tal com van ser introduïts per Reed i Solomon, en què les paraules codi s'obtenen avaluant polinomis-missatge en elements del cos; més endavant, a la secció 6.6 es veurà com molts d'aquests codis s'obtenen també com a codis BCH, que és la manera com se solen presentar aquests codis en les aplicacions. A la secció 6.2 es consideraran els codis polinomials en general, formats per les paraules que són múltiples d'un polinomi generador, per al qual és suficient treballar en l'espai $\mathbb{F}[X]_n$. A la secció 6.3 es veuran els codis cíclics, que són un cas particular de codis polinomials, on es treballa a l'espai $\mathbb{F}[X]_{N(X)}$ i s'agafa com a mòdul el polinomi $N(X) = X^n - 1$.

6.1 Codis de Reed-Solomon

L'any 1960 a l'article [26] Reed i Solomon construeixen famílies de codis lineals MDS de longitud i dimensió arbitràries, que des d'aleshores es coneixen pel nom de *codis de Reed-Solomon* en honor seu. Això sí, la construcció requereix que el cos finit tingui almenys tants elements com la longitud del codi. Per tant els codis no poden ser binaris i si es volen longituds grans s'han d'agafar cossos cada vegada més grans.

En aquest article també proposen algorismes de descodificació, que no són, però, prou eficients per ser usats en la pràctica. Més endavant s'han trobat algorismes simples i eficients per a la descodificació. Aquí es veurà un algorisme eficient que només requereix la solució d'un sistema d'equacions lineals.

Poc després de la proposta original de Reed i Solomon es va veure que en molts casos aquests codis també es poden obtenir d'una altra manera com a codis polinomials BCH. Aquesta altra versió dels codis de Reed-Solomon es veurà a la secció 6.6. Per als codis BCH es coneixen diversos algorismes de descodificació eficients, que es descriuen a la secció 6.7, els quals es poden aplicar també als codis de Reed-Solomon quan es presenten d'aquesta manera.

Els codis de Reed-Solomon estan entre els més usats en la pràctica. A principis dels anys 80 es van convertir en els primers codis correctors d'errors en ser incorporats a un producte de consum massiu: el *compact disc*. Altres tecnologies d'emmagatzematge òptic usen també codis de Reed-Solomon: *CD-ROM*, *DVD*, etc. També es fan servir en molts formats de *codis de barres* 2-dimensionals: els *codis QR*.

Codis de Reed-Solomon avaluant polinomis. En la seva *presentació original* per Reed i Solomon els codis es construeixen avaluant polinomis de grau $< k$ en un conjunt d'elements d'un cos finit:

Definició 6.1. *Siguin $\alpha_1, \alpha_2, \ldots, \alpha_n$ elements diferents d'un cos finit $\mathbb{F} = \mathbb{F}_q$. Sigui k un enter amb $1 \leqslant k \leqslant n$. Es defineix el codi de Reed-Solomon* $\mathrm{RS}\left(k, (\alpha_i)_{1 \leqslant i \leqslant n}\right)$ *com*

$$\mathcal{C} = \left\{ \left(\boldsymbol{m}(\alpha_1), \boldsymbol{m}(\alpha_2), \ldots, \boldsymbol{m}(\alpha_n) \right) \in \mathbb{F}^n : \boldsymbol{m}(X) \in \mathbb{F}[X]_k \right\}.$$

Com que l'avaluació de polinomis en elements d'un cos es comporta linealment respecte dels polinomis, aquesta definició proporciona codis lineals. La definició ja porta implícita una aplicació lineal de codificació: els missatges són elements de l'espai vectorial $\mathbb{F}[X]_k$ dels

polinomis de grau $< k$ i la codificació $\mathrm{enc}\colon \mathbb{F}[X]_k \to \mathbb{F}^n$ envia cada missatge $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$ a la n-tupla $\boldsymbol{c} \operatorname{enc}\big(\boldsymbol{m}(X)\big)$ que té components els valors $\boldsymbol{m}(\alpha_i)$ en els α_i. La matriu generadora corresponent a aquesta aplicació és la matriu

$$\mathbf{G} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ \alpha_1 & \alpha_2 & \alpha_3 & \cdots & \alpha_n \\ \vdots & \vdots & \vdots & & \vdots \\ \alpha_1^{k-1} & \alpha_2^{k-1} & \alpha_3^{k-1} & \cdots & \alpha_n^{k-1} \end{bmatrix} \in \mathrm{Mat}_{k\times n}(\mathbb{F}).$$

Una de les propietats més importants dels codis de Reed-Solomon és que assoleixen la fita de Singleton. És a dir, són codis MDS:

Lema 6.2. *Els codis* $\mathrm{RS}\big(k, (\alpha_i)\big)$ *són codis lineals de tipus* $[n, k, d]_q$ *amb* $d = n - k + 1$.

Prova: Per definició, les paraules tenen longitud n.

L'aplicació de codificació corresponent $\mathrm{enc}\colon \mathbb{F}[X]_k \to \mathbb{F}^n$ és lineal. És injectiva, ja que un polinomi $\boldsymbol{m}(X)$ de grau $\deg \boldsymbol{m} < k \leqslant n$ queda unívocament determinat pels seus valors en els n elements diferents α_i. O, de manera equivalent, perquè el seu nucli és trivial: si un polinomi $\boldsymbol{m}(X)$ té imatge zero aleshores té almenys n arrels diferents en els elements α_i, i com que el seu grau és $< n$ ha de ser el polinomi zero.

Per tant el codi, que per definició és la seva imatge $\mathcal{C} = \mathrm{Im}(\mathrm{enc})$, és un subespai vectorial de $\mathbb{F}^n$ de dimensió k.

Per a tot polinomi $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$ diferent de zero, de grau $\deg \boldsymbol{m} \leqslant k-1$, la paraula $\boldsymbol{c} = \mathrm{enc}(\boldsymbol{m})$ pot tenir com a màxim $k-1$ coordenades iguals a zero, ja que cada coordenada $\mathtt{c}_i = \boldsymbol{m}(\alpha_i)$ igual a zero correspon a una arrel del polinomi. Per tant $\boldsymbol{c}$ té com a mínim $n - (k-1) = n - k + 1$ coordenades diferents de zero:

$$\|\boldsymbol{c}\| = \big\|\operatorname{enc}(\boldsymbol{m})\big\| \geqslant n - k + 1,\ \forall\ \boldsymbol{c} \neq \mathbf{0} \qquad \Rightarrow \qquad d(\mathcal{C}) \geqslant n - k + 1.$$

Naturalment, la distància mínima del codi ha de ser per força $d(\mathcal{C}) = n-k+1$, ja que aquest nombre és sempre una fita superior: la fita de Singleton. □

Descodificació i correcció d'errors per interpolació. Per descodificar una paraula codi $\boldsymbol{c} = (\mathtt{c}_1, \ldots, \mathtt{c}_n) \in \mathrm{RS}\big(k, (\alpha_i)\big)$ simplement es calcula el polinomi interpolador: $\boldsymbol{m}(X)$ és l'únic polinomi de grau $< m$ tal que $\boldsymbol{m}(\alpha_i) = \mathtt{c}_i$ per a tot $i = 1, \ldots, n$. De fet, aquest polinomi ja queda unívocament determinat només per k identitats $\boldsymbol{m}(\alpha_i) = \mathtt{c}_i$ per a k índexs i qualsevol.

Per tant si s'envia una paraula codi $\boldsymbol{c} \in \mathcal{C}$ i es rep una paraula $\boldsymbol{x} \in \mathbb{F}^n$ amb errors en algunes coordenades, a partir de k coordenades qualssevol de $\boldsymbol{x}$ que no continguin cap error es pot recuperar el missatge.

En el seu article [18] Reed i Solomon es basen en aquesta observació per proposar l'algorisme de descodificació següent: suposi's que es rep a la sortida del canal la paraula $\boldsymbol{x} = (\mathtt{x}_1, \mathtt{x}_2, \cdots, \mathtt{x}_n) \in \mathbb{F}^n$. Per a cada subconjunt $I \subseteq \{1, \ldots, n\}$ de $|I| = k$ elements s'agafen les k coordenades $\mathtt{x}_i$ de la paraula rebuda per als índexs $i \in I$ en aquest subconjunt

i es calcula el polinomi de grau $< k$ que pren els valors $\mathtt{x}_i$ en els elements α_i: el *polinomi interpolador* determinat pels seus valors en aquests k elements diferents:

$$\boldsymbol{m}_I(X) \in \mathbb{F}[X]_k \qquad \text{amb} \quad \boldsymbol{m}_I(\alpha_i) = \mathtt{x}_i \quad \forall\, i \in I.$$

Aleshores es descodifica suposant que el missatge enviat és el polinomi de $\mathbb{F}[X]_k$ que hagi aparegut més vegades com a polinomi interpolador $\boldsymbol{m}_I(X)$ per a més subconjunts I.

Aquest algorisme és clarament ineficient degut a la gran quantitat de polinomis interpoladors que s'haurien de calcular i no es pot usar en la pràctica.

Es pot modificar per transformar-lo en un algorisme probabilistic, que és essencialment l'algorisme de correcció d'errors per subconjunt d'informació, on en aquest cas es pot agafar com a subconjunt d'informació qualsevol subconjunt de k posicions de la paraula rebuda, ja que k columnes diferents de la matriu generadora són sempre independents.

Síndrome, localitzador i avaluador. En molts mètodes de correcció d'errors apareixen polinomis de síndrome calculats a partir del polinomi interpolador de la paraula rebuda, i també dos polinomis que es coneixen pel nom de polinomi interpolador i polinomi avaluador; el primer té com a arrels els α_i corresponents a les posicions on s'ha produït un error i el segon permet determinar les magnituds d'aquests errors a partir del seu valor en α_i.

Sigui $\boldsymbol{C} = \mathrm{RS}\big(k, (\alpha_i)\big)$ un codi de Reed-Solomon. Sigui $\boldsymbol{c} = (\mathtt{c}_i) = \mathrm{enc}\big(\boldsymbol{m}(X)\big)$ una paraula codi. Siguin $\boldsymbol{v} = (\mathtt{v}_i) \in \mathbb{F}^n$ i $\boldsymbol{e} = (\mathtt{e}_i) = \boldsymbol{v} - \boldsymbol{c}$, que representen la paraula rebuda en enviar $\boldsymbol{c}$ i la paraula d'error corresponent.

Per a cada vector $\boldsymbol{x} = (\mathtt{x}_i) \in \mathbb{F}^n$ es denotarà $P_{\boldsymbol{x}} \in \mathbb{F}[X]_n$ el polinomi interpolador amb valors $P_{\boldsymbol{x}}(\alpha_i) = \mathtt{x}_i$ per a tot $i = 1, \ldots, n$. En particular es té $P_{\boldsymbol{c}}(X) = \boldsymbol{m}(X)$ com a polinomis de $\mathbb{F}[X]_n$.

Es defineix l'aplicació de síndrome $\mathrm{syn}\colon \mathbb{F}^n \to \mathbb{F}[X]_m$ posant

$$\mathrm{syn}(\boldsymbol{v}) = \boldsymbol{s}(X) \in \mathbb{F}[X]_m \quad \text{si} \quad P_{\boldsymbol{v}}(X) = \boldsymbol{s}_0(X) + X^k \boldsymbol{s}(X), \quad \text{amb} \quad \boldsymbol{s}_0(X) \in \mathbb{F}[X]_k.$$

O sigui, la síndrome és la part de grau $\geqslant k$ del polinomi interpolador del vector $\boldsymbol{v}$. És clar que les paraules codi es caracteritzen per la propietat de tenir síndrome zero: són les que el seu polinomi interpolador té grau $< k$, i per tant la part de grau $\geqslant k$ és el polinomi zero.

Es defineix el polinomi localitzador d'errors com

$$L(X) = \prod_{\mathtt{e}_i \neq 0} (X - \alpha_i), \quad \text{de grau} \quad t = \deg L = w(\boldsymbol{e}).$$

Posant $N(X) = \prod_{i=1}^{n} (X - \alpha_i)$ i $L^c(X) = \prod_{\mathtt{e}_i = 0} (X - \alpha_i)$ es té $N(X) = L(X)L^c(X)$. Es defineix el polinomi avaluador d'errors com

$$A(X) \in \mathbb{F}[X]_t \quad \text{amb} \quad P_{\boldsymbol{e}}(X) = A(X)L^c(X).$$

Està ben definit ja que el polinomi $L^c(X)$ s'anul·la en tots els α_i en els quals $P_{\boldsymbol{e}}(\alpha_i) = 0$ i per tant aquest polinomi divideix l'interpolador $P_{\boldsymbol{e}}(X)$. Pel que fa al grau, com que $\deg L^c(X) = n - t$ es té $\deg A = \deg P_{\boldsymbol{e}} - \deg L^c < t$.

Aquests polinomis satisfan la identitat

$$P_{\boldsymbol{v}}(X) = \boldsymbol{s}_0(X) + X^k \boldsymbol{s}(X) = P_{\boldsymbol{c}}(X) + P_{\boldsymbol{e}}(X) = \boldsymbol{m}(X) + A(X)L^c(X).$$

Multiplicant per $L(X)$ es dedueix la identitat

$$N(X)\big(-A(X)\big) + P_{\boldsymbol{v}}(X)L(X) = L(X)\boldsymbol{m}(X).$$

Algorisme de Berlekamp-Welch. A continuació es veu un algorisme de descodificació eficient, que es coneix com a *algorisme de Berlekamp-Welch*, en què la descodificació per proximitat es redueix a trobar una solució no trivial d'un sistema lineal homogeni de n equacions en $n+1$ incògnites.

Proposició 6.3. *Sigui $\mathcal{C} = \mathrm{RS}\big(k, (\alpha_j)\big)$, de tipus $[n,k,d]$. Sigui $\tau = \lfloor \frac{d-1}{2} \rfloor$ la seva capacitat correctora, i sigui $\mu = n - \tau - 1$. Sigui $\boldsymbol{x} \in \mathbb{F}^n$ un vector qualsevol.*

1. *Existeixen polinomis $P(X), Q(X) \in \mathbb{F}[X]$ de graus $\deg P \leqslant \mu$ i $0 \leqslant \deg Q \leqslant \tau$ tals que*

$$P(\alpha_i) = \mathtt{x}_i Q(\alpha_i), \qquad \text{per a tot} \quad i = 1, \ldots, n.$$

2. *$d(\boldsymbol{x}, \mathcal{C}) \leqslant \tau$ si, i només si, Q divideix P i $\deg(P/Q) < k$. En aquest cas, el missatge $\boldsymbol{m}(X)$ tal que $\boldsymbol{c} = \mathrm{enc}\big(\boldsymbol{m}(X)\big) \in \mathcal{C}$ satisfà $d(\boldsymbol{x}, \boldsymbol{c}) \leqslant \tau$ és el quocient:*

$$\boldsymbol{m}(X) = \frac{P(X)}{Q(X)}.$$

PROVA: Primer es veu que $\deg(\boldsymbol{m}Q) \leqslant \mu$ per a tot missatge $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$. Aquesta desigualtat es farà servir en el segon apartat. Tenint en compte que el codi és MDS, o sigui que $d = n - k + 1$, la capacitat correctora satisfà la desigualtat $\tau = \lfloor \frac{n-k}{2} \rfloor \leqslant \frac{n-k}{2}$. Per tant, $2\tau \leqslant n - k$ i es dedueix que

$$\deg(\boldsymbol{m}Q) \leqslant k - 1 + \tau \leqslant k - 1 + 2\tau - \tau \leqslant k - 1 + n - k - \tau = n - \tau - 1 = \mu.$$

1. Naturalment, els polinomis $P = Q = 0$ satisfan les identitats $P(\alpha_j) = Q(\alpha_j)\mathtt{x}_j$, però en demanar que el grau de Q sigui $\geqslant 0$ s'està dient que aquest polinomi és no nul, de manera que s'ha de provar que existeixen polinomis que satisfan les identitats amb $Q \neq 0$. S'agafen polinomis genèrics

$$P(X) = p_0 + p_1 X + \cdots + p_\mu X^\mu, \quad Q(X) = q_0 + q_1 X + \cdots + q_\tau X^\tau.$$

   Considerant els seus $\mu + 1 + \tau + 1 = n + 1$ coeficients p_r i q_s com a incògnites. Cada igualtat $P(\alpha_i) - \mathtt{x}_i Q(\alpha_i) = 0$ és una equació lineal homogènia en aquests coeficients.

   Aquestes identitats corresponen a un sistema homogeni de n equacions en $n+1$ incògnites. Per Rouché-Frobenius el sistema té solució no trivial i per tant existeixen polinomis $P(X)$ i $Q(X)$ que satisfan totes les identitats i no són tots dos zero. En aquest cas Q ha de ser diferent de zero, ja que si aquest fos zero el polinomi P, de grau $\deg P \leqslant \mu = n - \tau - 1 < n$, s'anul·laria en tots els n elements α_j, i per tant també seria el polinomi zero. Amb això queda demostrat el primer apartat.

2. Siguin P i Q com al primer apartat, de manera que $P(\alpha_i) - \mathtt{x}_i Q(\alpha_i) = 0$ per a tot α_i. Donats $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$ un missatge qualsevol i $\boldsymbol{c} = \operatorname{enc}\big(\boldsymbol{m}(X)\big) = (\mathtt{c}_1, \ldots, \mathtt{c}_n)$ la paraula codi corresponent es considera el polinomi $R_{\boldsymbol{m}}(X) = P(X) - \boldsymbol{m}(X)Q(X)$. Per la desigualtat que s'ha vist al començament de la prova aquest polinomi té grau $\leqslant \mu$.

   Per a cada element α_i es té

$$R_{\boldsymbol{m}}(\alpha_i) = P(\alpha_i) - \boldsymbol{m}(\alpha_i)Q(\alpha_i) = P(\alpha_i) - \mathtt{c}_i Q(\alpha_i) = 0 \quad \Leftrightarrow \quad (\mathtt{c}_i - \mathtt{x}_i)Q(\alpha_i) = 0.$$

   Si $\boldsymbol{c}$ és una paraula codi amb $d(\boldsymbol{x}, \boldsymbol{c}) \leqslant \tau$ aleshores $\mathtt{c}_i = \mathtt{x}_i$ per almenys $n - \tau = \mu + 1$ elements α_i diferents. Això implica que el polinomi $R_{\boldsymbol{m}}(X)$ té més arrels que el seu grau i per tant és el polinomi zero. Es dedueix que Q divideix P i que el seu quocient és el polinomi $\boldsymbol{m}$, de grau $< k$.

   Recíprocament, suposi's que Q divideix P i que el quocient té grau $< k$. Sigui $\boldsymbol{m}(X)$ aquest quocient i sigui $\boldsymbol{c} = \operatorname{enc}\big(\boldsymbol{m}(X)\big)$ la paraula codi corresponent. Aleshores es té $(\boldsymbol{c}_i - \mathtt{x}_i)Q(\alpha_i) = (\boldsymbol{m}(\alpha_i) - \mathtt{x}_i)Q(\alpha_i) = 0$ per a tot α_i. Com que $\deg Q \leqslant \tau$ el polinomi Q té com a màxim τ arrels. Per tant hi pot haver com a màxim k índexs i amb $\boldsymbol{c}_i \neq \boldsymbol{x}_i$ i això vol dir que $d(\boldsymbol{c}, \boldsymbol{x}) \leqslant \tau$ i per tant $d(\boldsymbol{x}, \mathcal{C}) \leqslant \tau$ i la paraula $\boldsymbol{c}$ a distància mínima és la codificació del polinomi quocient $m = P/Q$. □

Aquesta proposició permet dissenyar un algorisme eficient per corregir errors:

Algorisme 6.4 (Algorisme de Berlekamp-Welch)**.** *Sigui* $\mathcal{C} = \operatorname{RS}\big(k, (\alpha_i)_{1 \leqslant i \leqslant n}\big)$. *Sigui* τ *la seva capacitat correctora i* $\mu = n - \tau - 1$.

ENTRADA: *Una paraula* $\boldsymbol{x} = (\mathtt{x}_1, \ldots, \mathtt{x}_n) \in \mathbb{F}^n$.

SISTEMA: *Es resol el sistema lineal homogeni amb* n *equacions*

$$\sum_{i=0}^{\mu} \alpha_j^i p_i + \sum_{i=0}^{\tau} -\mathtt{x}_j \alpha_j^i q_i = 0, \qquad j = 1, \ldots, n,$$

en les en $n + 1$ *incògnites* $p_0, \ldots, p_\mu$ *i* $q_0, \ldots, q_\tau$.

POLINOMIS: *Siguin* $P(X) := \sum_{i=0}^{\mu} p_i X^i$ *i* $Q(X) := \sum_{i=0}^{\tau} q_i X^i$ *polinomis corresponents a una solució no trivial qualsevol* $(p_0, \ldots, p_\mu, q_0, \ldots, q_\tau)$ *d'aquest sistema.*

SORTIDA: *Si* $Q(X) \mid P(X)$ *i* $\deg \frac{P(X)}{Q(X)} < k$ *es retorna* $\boldsymbol{m}(X) = \frac{P(X)}{Q(X)}$. *Altrament es declara que* $d(\boldsymbol{x}, \mathcal{C}) > \tau$, *de manera que l'error no està dins de la capacitat correctora.*

Algorisme de Gao. Un altre algorisme eficient de descodificació, proposat per Gao a [31], corregeix els errors amb l'algorisme d'Euclides estès parcial. Es basa en la següent:

Proposició 6.5. *En les hipòtesi de la proposició 6.3, sigui* $\boldsymbol{r}_0(X) = \prod_{i=1}^{n}(X - \alpha_i) \in \mathbb{F}[X]$ *i sigui* $\boldsymbol{r}_1(X) \in \mathbb{F}[X]_n$ *el polinomi interpolador amb* $\boldsymbol{r}_1(\alpha_i) = \mathtt{x}_i$.

S'aplica l'algorisme d'Euclides estès als dos polinomis $\boldsymbol{r}_0$ *i* $\boldsymbol{r}_1$ *obtenint-se identitats*

$$\boldsymbol{r}_0(X)\boldsymbol{u}_i(X) + \boldsymbol{r}_1(X)\boldsymbol{v}_i(X) = \boldsymbol{r}_i(X), \qquad i = 0, 1, 2, \ldots$$

Si $\boldsymbol{r}_i(X)$ *és el primer reste de grau* $\boldsymbol{r}_i(X) \leqslant \mu$ *aleshores els polinomis* $P(X) = \boldsymbol{r}_i(X)$ *i* $Q(X) = \boldsymbol{v}_i(X)$ *satisfan les condicions del primer apartat de la proposició 6.3.*

PROVA: Com que $\boldsymbol{r}_0(\alpha_i) = 0$ i $\boldsymbol{r}_1(\alpha_i) = \mathtt{x}_i$, avaluant les identitats polinomials de l'algorisme en els elements α_i es veu que en tots els passos de l'algorisme els polinomis $P(X) = \boldsymbol{r}_j(X)$ i $Q(X) = \boldsymbol{v}_j(X)$ satisfan $P(\alpha_i) = \mathtt{x}_i Q(\alpha_i)$ per a tot i.

Observi's que en avançar l'algorisme els graus dels polinomis $\boldsymbol{r}_i$ van decreixent amb i els dels $\boldsymbol{v}_i$ van augmentant. Per demostrar l'afirmació de l'enunciat s'ha de veure que quan s'agafa el primer reste de grau $\leqslant \mu$ el polinomi $\boldsymbol{v}$ corresponent té grau $\leqslant \tau$.

Com que $\deg \boldsymbol{r}_0 = n > \mu$ és clar que existeix un primer índex $i \geqslant 1$ tal que $\boldsymbol{r}_i(X) \leqslant \mu$, i que en aquest cas el reste anterior té grau $\boldsymbol{r}_{i-1} > \mu$ i, per tant, que $\boldsymbol{r}_{i-1} \geqslant \mu + 1 = n - \tau$.

Siguin $\boldsymbol{q}_1, \boldsymbol{q}_2, \ldots$ els quocients de $\boldsymbol{r}_{i-1}(X) = \boldsymbol{r}_i(X)\boldsymbol{q}_i(X) + \boldsymbol{r}_{r+1}(X)$. D'aquestes expressions (i tenint en compte que $\deg \boldsymbol{r}_0 = n \geqslant \deg \boldsymbol{r}_1$) es dedueix que $\deg \boldsymbol{r}_i = n - \deg \prod_{j=1}^{i} \boldsymbol{q}_j$ per a tot $i \geqslant 0$, i de les expressions $\boldsymbol{v}_{i+1} = \boldsymbol{v}_{i-1} - \boldsymbol{q}_i \boldsymbol{v}_i$ es dedueix que $\deg \boldsymbol{v}_i = \deg \prod_{j=0}^{i-1} \boldsymbol{q}_j = n - \deg \boldsymbol{r}_{i-1}$ per a tot $i \geqslant 1$.

Per tant, agafant els polinomis P i Q corresponents a l'índex i amb la condició de l'enunciat es té $\deg Q = \deg \boldsymbol{v}_i = n - \deg \boldsymbol{r}_{i-1} \leqslant n - (n - \tau) = \tau$. □

Algorisme 6.6 (Algorisme de Gao)**.** *Sigui* $\mathcal{C} = \mathrm{RS}_q\big(k, (\alpha_i)_{1 \leqslant i \leqslant n}\big)$.

ENTRADA: *Una paraula* $\boldsymbol{x} = (\mathtt{x}_1, \ldots, \mathtt{x}_n) \in \mathbb{F}^n$.

INICIALITZACIÓ: *S'inicialitzen dos polinomis* $\boldsymbol{r}_0(X) = \prod_{i=1}^{n}(X - \alpha_i)$ *i* $\boldsymbol{r}_1(X)$ *el polinomi interpolador amb valors* $\boldsymbol{r}_1(\alpha_i) = \mathtt{x}_i$*, i uns altres dos polinomis* $\boldsymbol{v}_0(X) = 0$ *i* $\boldsymbol{v}_1(X) = 1$.

REPETIR: *Mentre* $\deg \boldsymbol{r}_1 > \mu$,

> DIVISIÓ EUCLIDIANA: *Es fa la divisió* $\boldsymbol{r}_0(X) = \boldsymbol{r}_1(X)\boldsymbol{q}(X) + \boldsymbol{r}(X)$.
>
> ACTUALITZACIÓ: *S'actualitzen els quatre polinomis:*
>
> - $\boldsymbol{r}_0, \boldsymbol{r}_1 \leftarrow \boldsymbol{r}_1, \boldsymbol{r}$*;*
> - $\boldsymbol{v}_0, \boldsymbol{v}_1 \leftarrow \boldsymbol{v}_1, \boldsymbol{v}_0 - \boldsymbol{q}\boldsymbol{v}_1$.

SORTIDA: *Si* $\boldsymbol{v}_1$ *divideix* $\boldsymbol{r}_1$ *i* $\deg \frac{\boldsymbol{r}_1}{\boldsymbol{v}_1} < k$ *es retorna* $\boldsymbol{m}(X) = \frac{\boldsymbol{r}_1(X)}{\boldsymbol{v}_1(X)}$*; altrament es retorna "error no corregible".*

Quan l'algorisme retorna un polinomi $\boldsymbol{m}(X)$, aquest és l'únic missatge tal que la paraula codi corresponent $\boldsymbol{c} = \mathrm{enc}\big(\boldsymbol{m}(X)\big) \in \mathcal{C}$ està a distància $d(\boldsymbol{x}, \boldsymbol{c}) \leqslant \tau$; quan retorna error no corregible significa $d(\boldsymbol{x}, \mathcal{C}) > \tau$.

Codis MDS. S'ha vist que els codis de Reed-Solomon $\mathrm{RS}\big(n, (\alpha_i)\big)$ són codis MDS. Com que per construir-los calen n elements α_i diferents del cos $\mathbb{F}_q$, en augmentar la longitud n s'ha de treballar en cossos finits $\mathbb{F}_q$ de cardinal cada vegada més gran. Concretament ha de ser $q \geqslant n$. Per tant la idea no es pot fer servir per construir codis binaris. Hi ha una manera d'aconseguir codis MDS sobre el cos $\mathbb{F}_q$ una mica més llargs: amb longitud fins a $q + 1$, que consisteix essencialment en la mateixa construcció agafant un altre "element" α que no pertany al cos i que es pot interpretar com el punt de l'infinit de la recta projectiva (veure problema **6.1**).

Es pot veure que no hi ha altres codis MDS lineals que evitin aquestes restriccions sobre el cardinal del cos: excepte els casos trivials de codis de dimensions $k = 0, 1, n - 1$ i n, que

corresponen essencialment als codis trivial, de repetició, de paritat i total, i que poden ser de qualsevol longitud independentment del cos $\mathbb{F}_q$, no existeixen altres codis MDS lineals de longitud $n > q + 1$. Veure[2, Secció 6.3] per a més detalls.

Problemes

6.1. Per a polinomis $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$ de grau $< k$ es defineix $f(\infty) = \mathtt{m}_{k-1}$, el coeficient de grau $k-1$ del polinomi. Comproveu que en la construcció dels codis de Reed-Solomon es pot agafar també l'element $\alpha = \infty$ i que d'aquesta manera es poden construir codis MDS q-aris de dimensió k i longitud $k \leqslant n \leqslant q+1$ avaluant els polinomis de $\mathbb{F}[X]_k$ en n elements diferents del conjunt $\mathbb{F} \sqcup \{\infty\}$, que conté $q+1$ elements.

6.2. Estudieu tots els codis de Reed-Solomon que es poden construir sobre els cossos finits de cardinal $q = 2$, 3 i 4.

6.3. Es consideren els polinomis $N(X)$ i $P_v(X)$ del text, que són l'entrada de l'algorisme de Gao. Sigui $\boldsymbol{r}_i$ el primer reste de l'algorisme d'Euclides de grau $\leqslant \mu$ i siguin $\boldsymbol{u}_i, \boldsymbol{v}_i$ els coeficients corresponents, de manera que $N\boldsymbol{u}_i + P_{\boldsymbol{v}}\boldsymbol{v}_i = \boldsymbol{r}_i$. Es multipliquen tots tres polinomis per una constant no nul·la per tal que $\boldsymbol{v}_i$ sigui mònic. Demostreu que:

1. $\boldsymbol{v}_i(X) = L(X)$ és el polinomi localitzador;
2. $\boldsymbol{u}_i(X) = -A(X)$ és menys el polinomi avaluador;
3. $\boldsymbol{r}_i(X) = L(X)\boldsymbol{m}(X)$ és el producte del polinomi localitzador pel missatge.

6.4. Sigui $N(X) = N_0(X) + X^k M(X)$, amb $\deg N_0 < k$ i $\deg M < m$. Es fa l'algorisme d'Euclides estès amb els dos polinomis $M(X)$ i $\mathrm{syn}(X)$. Sigui $\boldsymbol{r}_i$ el primer reste de l'algorisme d'Euclides de grau $\leqslant \tau$ i siguin $\boldsymbol{u}_i, \boldsymbol{v}_i$ els coeficients corresponents, de manera que $M\boldsymbol{u}_i + \boldsymbol{s}\boldsymbol{v}_i = \boldsymbol{r}_i$. Es multipliquen tots tres polinomis per una constant no nul·la per tal que $\boldsymbol{v}_i$ sigui mònic. Demostreu que:

1. $\boldsymbol{v}_i(X) = L(X)$ és el polinomi localitzador;
2. $\boldsymbol{u}_i(X) = -A(X)$ és menys el polinomi avaluador.

6.2 Codi generat per un polinomi

Partint d'un polinomi $\boldsymbol{g}(X) \in \mathbb{F}[X]$ es poden construir codis lineals agafant les paraules que, com a polinomis, són múltiples de $\boldsymbol{g}(X)$:

Definició 6.7 (Codi generat per un polinomi)**.** *Sigui $\boldsymbol{g}(X) \in \mathbb{F}[X]$ un polinomi de grau $m = \deg \boldsymbol{g}$. Sigui $n \geqslant m$ un enter. El codi $\mathrm{Pol}(n, \boldsymbol{g})$ de longitud n generat per $\boldsymbol{g}$ és el subconjunt $\mathcal{C} \subseteq \mathbb{F}[X]_n$ format per tots els polinomis de grau $< n$ que són divisibles per $\boldsymbol{g}$.*

Lema 6.8. *El codi $\mathrm{Pol}(n, \boldsymbol{g})$ és un codi lineal de dimensió $k = n - m$ que admet com a base els polinomis de $\mathbb{F}[X]_n$ següents:*

$$\boldsymbol{g}(X),\ X\boldsymbol{g}(X),\ X^2\boldsymbol{g}(X),\ \ldots,\ X^{k-1}\boldsymbol{g}(X). \tag{8}$$

PROVA: Com que la divisibilitat per un polinomi es conserva per suma i producte per escalars és un codi lineal. Sigui $k := n - m$.

Que un polinomi $\boldsymbol{c}(X) \in \mathbb{F}[X]_n$ sigui divisible per $\boldsymbol{g}(X)$ vol dir que existeix un polinomi quocient $\boldsymbol{q}(X)$ amb $\boldsymbol{c}(X) = \boldsymbol{g}(X)\boldsymbol{q}(X)$. Aquest polinomi quocient $\boldsymbol{q}$ ha de tenir grau $\deg \boldsymbol{q} = \deg \boldsymbol{c} - \deg \boldsymbol{g} < n - m = k$. Per tant $\boldsymbol{q}(X) \in \mathbb{F}[X]_k$. Recíprocament, per a cada polinomi $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$ el producte $\boldsymbol{c}(X) = \boldsymbol{m}(X)\boldsymbol{g}(X)$ és un polinomi del codi.

L'aplicació $\boldsymbol{m}(X) \mapsto \boldsymbol{m}(X)\boldsymbol{g}(X)\colon \mathbb{F}[X]_k \to \mathbb{F}[X]_n$ és una aplicació lineal injectiva que té imatge $\mathcal{C}$: una aplicació de codificació. Per tant, el codi té dimensió k. Aquesta aplicació envia els monomis $1, X, \ldots, X^{k-1}$ de la base canònica de $\mathbb{F}[X]_k$ a les paraules codi de (8). Per tant, aquestes paraules són una base del codi. □

El codi no canvia si es multiplica el polinomi generador per una constant no nul·la. Per tant, multiplicant per l'escalar convenient (l'invers del coeficient líder) es pot suposar, sense perdre generalitat, que el polinomi generador és mònic (té coeficient líder igual a 1).

Se suposarà sempre també, si no es diu el contrari, que el polinomi generador té terme constant diferent de zero, o sigui que $\boldsymbol{g}(0) \neq 0$. Això equival a què $\boldsymbol{g}(X)$ no sigui divisible per X. Altrament el codi seria degenerat: la primera component de totes les paraules codi seria igual a zero.

Exemples 6.9. *Sigui $\mathbb{F} = \mathbb{F}_q$ el cos base de q elements.*

1. *Sigui $\boldsymbol{g}(X) = X - 1$, de grau $m = 1$. Per a cada $n \geqslant 1$ el codi polinomial* $\mathrm{Pol}(n, \boldsymbol{g})$ *és el codi parell* $\mathrm{Par}_q(n)$.
2. *Sigui $\boldsymbol{g}(X) = 1 + X + X^2 + \cdots + X^m$. Per a $n = m + 1$ el codi polinomial* $\mathrm{Pol}(n, \boldsymbol{g})$ *és el codi de repetició* $\mathrm{Rep}_q(n)$.

PROVA:

1. La dimensió és $k = n-1$. Els polinomis $\boldsymbol{c}(X) = \mathtt{c}_0 + \mathtt{c}_1 X + \cdots + \mathtt{c}_{n-1}X^{n-1}$ de grau $< n$ divisibles per $\boldsymbol{g}(X)$ són els que tenen l'1 com a arrel: els que $\boldsymbol{c}(1) = \mathtt{c}_0 + \mathtt{c}_1 + \cdots + \mathtt{c}_{n-1} = 0$, que és la condició per tal que el vector $\boldsymbol{c} = (\mathtt{c}_0, \ldots, \mathtt{c}_{n-1})$ pertanyi a $\mathrm{Par}_q(n)$.
2. La dimensió és $k = n - m = 1$. Els polinomis $\boldsymbol{c}(X)$ de grau $< n$ (o sigui $\leqslant m$) són els múltiples de $\boldsymbol{g}(X)$. Aquí els missatges han de ser constants $\boldsymbol{m}(X) = \mathtt{m} \in \mathbb{F}[X]_k = \mathbb{F}$ i el producte $\boldsymbol{m}(X)\boldsymbol{g}(X)$ és $\mathtt{m}\boldsymbol{g}(X) = \mathtt{m} + \mathtt{m}X + \cdots \mathtt{m}X^m$, que correspon a la paraula codi $(\mathtt{m}, \mathtt{m}, \ldots, \mathtt{m}) \in \mathbb{F}^n$. Per tant el codi és el de repetició. □

Aplicacions de codificació. Usant el polinomi generador es poden definir diverses aplicacions de codificació: aplicacions lineals $\mathrm{enc}\colon \mathbb{F}[X]_k \to \mathbb{F}[X]_n$ amb imatge $\mathcal{C}$, on s'agafa el conjunt $\mathbb{F}[X]_k$ dels polinomis de grau $< k$ com el de missatges a codificar.

- La codificació més senzilla consisteix simplement a multiplicar pel polinomi generador:

$$\boldsymbol{c}(X) = \mathrm{enc}\big(\boldsymbol{m}(X)\big) = \boldsymbol{g}(X)\boldsymbol{m}(X).$$

La matriu generadora del codi corresponent a aquesta aplicació lineal de codificació és la matriu

$$\mathbf{G} = \begin{bmatrix} \mathsf{g}_0 & \mathsf{g}_1 & \cdots & \mathsf{g}_m & 0 & \cdots & 0 \\ 0 & \mathsf{g}_0 & \cdots & \mathsf{g}_{m-1} & \mathsf{g}_m & \cdots & 0 \\ \vdots & \ddots & \ddots & & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \mathsf{g}_0 & \cdots & \mathsf{g}_{m-1} & \mathsf{g}_m \end{bmatrix} \in \mathrm{Mat}_{k\times n}(\mathbb{F}), \tag{9}$$

on els g_i són els coeficients del polinomi generador:

$$\boldsymbol{g}(X) = \mathsf{g}_0 + \mathsf{g}_1 X + \mathsf{g}_2 X^2 + \cdots + \mathsf{g}_m X^m.$$

Aquesta matriu generadora (9) conté a les files els elements de la base (8).

- Per a cada $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$ sigui $X^m\boldsymbol{m}(X) = \boldsymbol{g}(X)\boldsymbol{q}(X) + \boldsymbol{r}(X)$ la divisió euclidiana. L'aplicació $\mathrm{enc}\colon \mathbb{F}[X]_k \to \mathbb{F}[X]_n$ definida posant

$$\boldsymbol{c}(X) = \mathrm{enc}\big(\boldsymbol{m}(X)\big) = -\boldsymbol{r}(X) + X^m\boldsymbol{m}(X) \tag{10}$$

és una codificació lineal sistemàtica a la dreta.

- Per a cada $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$ sigui $(-X^k)\boldsymbol{a}(X) + \boldsymbol{g}(X)\boldsymbol{b}(X) = \boldsymbol{m}(X)$ la solució de la identitat de Bézout amb $\deg(\boldsymbol{a}) < m$. L'aplicació $\mathrm{enc}\colon \mathbb{F}[X]_k \to \mathbb{F}[X]_n$ definida posant

$$\boldsymbol{c}(X) = \mathrm{enc}\big(\boldsymbol{m}(X)\big) = \boldsymbol{m}(X) + X^k\boldsymbol{a}(X)$$

és una codificació lineal sistemàtica a l'esquerra (aquí cal suposar que $\boldsymbol{g}(0) \neq 0$).

Aplicació de síndrome. A partir del polinomi generador es pot definir també una aplicació de síndrome $\mathrm{syn}\colon \mathbb{F}[X]_n \to \mathbb{F}[X]_m$ prou natural. Es tracta simplement d'agafar el reste de la divisió euclidiana pel polinomi generador:

$$\mathrm{syn}\big(\boldsymbol{u}(X)\big) = \boldsymbol{r}(X) \quad \text{si} \quad \boldsymbol{u}(X) = \boldsymbol{g}(X)\boldsymbol{q}(X) + \boldsymbol{r}(X), \quad \deg \boldsymbol{r} < m.$$

La matriu d'aquesta aplicació en les bases canòniques és una matriu de control de la forma $\mathbf{H} = \big[\mathbf{I}_m \big| \mathbf{H}'\big]$. Si $\mathbf{G}$ és la matriu generadora sistemàtica a la dreta corresponent a la codificació (10), que és de la forma $\mathbf{G} = \big[\mathbf{G}' \big| \mathbf{I}_k\big]$, aleshores es té $\mathbf{H}' = -\mathbf{G}'^{\mathsf{T}}$.

Detecció de ràfegues. Els codis polinomials poden detectar errors del tipus conegut com a *ràfega d'errors*: errors que poden afectar només símbols que estan dins d'un segment de la paraula codi $\boldsymbol{c}(X)$. La longitud de la ràfega és la longitud d'aquest segment.

Proposició 6.10. *Un codi polinomial* $\mathrm{Pol}(n, \boldsymbol{g})$ *amb polinomi generador de grau* $\deg g = m$ *detecta totes les ràfegues d'errors de longitud* $\leqslant m$ *en una paraula codi.*

Prova: Sigui $\boldsymbol{c}(X)$ la paraula transmesa, que és múltiple de $\boldsymbol{g}(X)$.

Una ràfega d'errors que afecti un màxim de m dígits consecutius es pot representar com un polinomi d'error de la forma $X^i\boldsymbol{e}(X)$ amb $\deg \boldsymbol{e} < m$ i $0 \leqslant i \leqslant \deg \boldsymbol{c} - \deg \boldsymbol{e}$.

Quan es produeix aquest error es rep la paraula $\boldsymbol{u}(X) = \boldsymbol{c}(X) + X^i\boldsymbol{e}(X)$. Es detecta l'error quan aquesta paraula no sigui una paraula codi; o sigui, quan $\boldsymbol{u}(X)$ no sigui divisible pel polinomi generador $\boldsymbol{g}(X)$.

Suposi's que l'error no es detecta: $\boldsymbol{u}(X)$ és divisible per $\boldsymbol{g}(X)$. Aleshores $\boldsymbol{g}(X)$ dividirà $X^i\boldsymbol{e}(X)$. La hipòtesi que $\boldsymbol{g}(0) \neq 0$ equival a què $\boldsymbol{g}(X)$ és relativament primer amb el polinomi X. Per tant, $\boldsymbol{g}(X)$ divideix $\boldsymbol{e}(X)$. Com que $\deg \boldsymbol{e} < m = \deg \boldsymbol{g}$ això només pot passar si $\boldsymbol{e}(X) = 0$ i per tant no hi ha hagut cap error. □

El codi parell $\mathrm{Par}_q(n)$ és el codi generat pel polinomi $\boldsymbol{g}(X) = X - 1$ de grau $m = 1$. Aquest codi detecta tots els errors en un dígit, que són les ràfegues de longitud $\leqslant 1$.

El codi de repetició $\mathrm{Rep}_q(n)$ és el codi generat pel polinomi $\boldsymbol{g}(X) = 1 + X + \cdots + X^{n-1}$ de grau $m = n - 1$. Aquest codi detecta fins a $n - 1$ errors i totes les ràfegues de longitud $\leqslant m$ corresponen a errors com aquests. En aquest cas, com que el codi és cíclic (secció 6.3), de fet les ràfegues d'errors es poden distribuir cíclicament sobre tota la paraula i en realitat una ràfega d'errors de longitud $\leqslant m$ vol dir errors que no afectin totes les lletres de la paraula, de manera que amb aquest codi es detectarà qualsevol error sempre que en la paraula rebuda hi hagi algun símbol correcte.

Escurçat de codis polinomials. Un mateix polinomi generador $\boldsymbol{g}(X)$ de grau m es pot fer servir per generar codis de qualsevol longitud $n \geqslant m$. Tots aquests codis tenen la mateixa codimensió m i la seva dimensió $k = n - m$ augmenta amb la longitud. La relació entre els codis $\mathrm{Pol}(n_1, \boldsymbol{g})$ i $\mathrm{Pol}(n_2, \boldsymbol{g})$ de longituds $n_1 < n_2$ generats per un mateix polinomi $\boldsymbol{g}(X)$ és que el primer s'obté a partir del segon escurçant-lo en les últimes $n_2 - n_1$ posicions.

Factors del polinomi generador i arrels. El polinomi generador, que es pot suposar mònic, descompon en producte de polinomis primers:

$$\boldsymbol{g}(X) = \prod_{i=1}^{r} \boldsymbol{p}_i(X)^{\mu_i}, \qquad \boldsymbol{p}_i(X) \in \mathbb{F}[X] \quad \text{primer.}$$

La divisibilitat per $\boldsymbol{g}(X)$ es tradueix en divisibilitat per les diferents potències de primer: per a tot polinomi $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$ es té

$$\boldsymbol{g}(X) \mid \boldsymbol{u}(X) \quad \Leftrightarrow \quad \boldsymbol{p}_i(X)^{\mu_i} \mid \boldsymbol{u}(X) \quad \forall\, i = 1, \ldots, r.$$

Se sol treballar gairebé sempre amb codis generats per polinomis que no tenen factors de multiplicitat més gran que 1: *polinomis lliures de quadrats*. Quan el polinomi generador té aquesta propietat la condició de pertànyer al codi és equivalent a la divisibilitat pels seus factors primers.

Per a polinomis primers de grau 1, de la forma $\boldsymbol{p}(X) = X - \alpha$, la divisibilitat es tradueix en el fet de tenir una arrel:

$$(X - \alpha) \mid \boldsymbol{u}(X) \quad \Leftrightarrow \quad \boldsymbol{u}(\alpha) = 0.$$

Per a polinomis primers de grau més gran $\deg \boldsymbol{p} = \nu > 1$ la divisibilitat també es pot caracteritzar de la mateixa manera, només que ara l'arrel α s'ha d'agafar en una extensió

del cos $\mathbb{F}$ dels coeficients del polinomi. Sigui $\mathbb{E} = \mathbb{F}[X]_{\boldsymbol{p}(X)}$, que és un cos de q^ν elements si $\mathbb{F} = \mathbb{F}_q$. Es té $\mathbb{F} \subseteq \mathbb{E}$ identificant els elements de $\mathbb{F}$ amb els polinomis constants de $\mathbb{F}[X]_{\boldsymbol{p}(X)}$. Aleshores la divisibilitat d'un polinomi $\boldsymbol{u}(X) \in \mathbb{F}[X]$ pel polinomi $\boldsymbol{p}(X)$ equival a què l'element $\alpha \in \mathbb{E}$ representat pel polinomi $X \in \mathbb{F}[X]_{\boldsymbol{p}(X)}$ sigui una arrel de $\boldsymbol{u}(X)$.

Aquesta manera de veure els codis polinomials, en què les paraules codi es caracteritzen pel fet d'anul·lar-se en elements de $\mathbb{F}$ o d'una extensió seva, és la que es fa servir en la construcció els codis BCH, que es veuran a la secció 6.6.

Exemple: els CRC. Es coneix amb el nom de *CRC* (cyclic redundancy check) una tècnica usada sovint en dispositius de memòria i sistemes de comunicacions digitals per detectar errors en la recuperació de les dades emmagatzemades o en la seva transmissió. Consisteix a codificar la informació amb un codi binari generat per un polinomi $\boldsymbol{g}(X) \in \mathbb{F}_2[X]$ de grau m, que se sol denotar amb el nom CRC-m.

Les dades que es volen protegir són paraules binàries de longitud k, identificades amb polinomis $\boldsymbol{m}(X) \in \mathbb{F}_2[X]_k$. Per codificar-les es poden simplement multiplicar pel polinomi $\boldsymbol{g}(X)$, el qual les converteix en la seqüència binària de longitud $n = k + m$ corresponent al polinomi $\boldsymbol{c}(X) = \boldsymbol{m}(X)\boldsymbol{g}(X) \in \mathbb{F}[X]_n$. Hi ha diverses alternatives a aquest tipus de codificació, entre altres les codificacions sistemàtiques descrites per als codis polinomials.

Aquestes dades es poden corrompre en haver estat emmagatzemades o transmeses convertint-se en una seqüència binària $\boldsymbol{u} \in \mathbb{F}^n$ de la mateixa longitud que $\boldsymbol{c}$. Per detectar possibles errors s'identifica aquesta seqüència amb un polinomi $\boldsymbol{u}(X) \in \mathbb{F}_2[X]_n$ i es divideix pel polinomi generador $\boldsymbol{g}(X)$. Si el reste de la divisió és zero es dona per fet que $\boldsymbol{u} = \boldsymbol{c}$ i que les dades són correctes. Altrament, si el reste no és zero, es conclou que s'han produït errors en la transmissió, ja que el polinomi enviat era múltiple del polinomi generador i el polinomi rebut no ho és.

A canvi de l'increment en la quantitat de dades a transmetre l'ús de CRC permet detectar molts errors. En particular, tal com s'ha vist a la proposició 6.10 es detecten totes les *ràfegues d'errors* errors que afectin fins a un màxim de m bits consecutius de $\boldsymbol{c}$.

Els polinomis usats en els CRC s'agafen de manera que puguin detectar el màxim nombre possible d'errors, segons criteris i necessitats diferents per a cada aplicació. A l'*enllaç* següent es poden trobar dotzenes de polinomis i quines aplicacions els fan servir. A l'*enllaç* es dona informació sobre criteris per determinar els millors polinomis.

Problemes

6.5. Estudieu els tres tipus de codificació considerats per als codis polinomials en el cas particular dels codis de repetició i codis parells, vistos com a codis polinomials.

6.6. Siguin $\mathcal{C} = \mathrm{Pol}_2(n, \boldsymbol{g})$ un codi binari polinomial amb polinomi generador $\boldsymbol{g}(X) \in \mathbb{F}_2[X]$. Comproveu que totes les paraules de $\mathcal{C}$ tenen pes parell si, i només si, $\boldsymbol{g}(X)$ té pes parell.

6.7. Siguin $\mathcal{C}_1 = \mathrm{Pol}(n, \boldsymbol{g}_1)$ i $\mathcal{C}_2 = \mathrm{Pol}(n, \boldsymbol{g}_2)$ dos codis polinomials de la mateixa longitud. Comproveu que la intersecció $\mathcal{C}_1 \cap \mathcal{C}_2$ és un codi polinomial, i digueu quin és el seu polinomi generador.

6.8. Doneu matrius generadores sistemàtiques a la dreta i a l'esquerra per als tres codis polinomials següents:

1. $\mathrm{Pol}_2(6, 1 + X + X^3)$;
2. $\mathrm{Pol}_2(10, 1 + X + X^2 + X^3 + X^4)$;
3. $\mathrm{Pol}_5(7, 1 + 2X + 3X^2 + 4X^3)$.

En tots tres casos calculeu aquestes matrius a partir de la matriu generadora de la forma (9) fent transformacions elementals i també a partir de les aplicacions de codificació corresponents, usant divisió euclidiana i identitat de Bézout.

6.9. REPASSAR!!! Siguin $\alpha_1, \ldots, \alpha_m$ elements diferents d'un cos finit $\mathbb{F}$. Sigui $n \geqslant m$. Es defineix

$$\mathcal{C} = \big\{\boldsymbol{c}(X) \in \mathbb{F}[X]_n : \boldsymbol{c}(\alpha_i) = 0, 1 \leqslant i \leqslant m\big\}.$$

1. Vegeu que $\mathcal{C}$ és un codi lineal.
2. Vegeu que $\mathcal{C}$ és polinomial i digueu quin és el seu polinomi generador $\boldsymbol{g}(X)$.
3. Quins codis polinomials es poden construir d'aquesta manera?
4. Quins codis binaris no degenerats es poden construir d'aquesta manera?
5. Comproveu que l'aplicació $\mathrm{syn}_1 \colon \mathbb{F}[X]_n \to \mathbb{F}^m$ definida posant

$$\mathrm{syn}\,\big(\boldsymbol{u}(X)\big) = \big(\boldsymbol{u}(\alpha_1), \ldots, \boldsymbol{u}(\alpha_m)\big)$$

   és una aplicació de síndrome i calculeu la matriu de control corresponent.
6. Comproveu que l'aplicació $\mathrm{syn}_2 \colon \mathbb{F}[X]_n \to \mathbb{F}[X]_m$ definida posant $\mathrm{syn}_2\big(\boldsymbol{v}(X)\big) = \boldsymbol{r}(X)$ si $\boldsymbol{v}(x) = \boldsymbol{g}(X)\boldsymbol{q}(X) + \boldsymbol{r}(X)$ és la divisió euclidiana és una aplicació lineal de síndrome i calculeu la matriu de control corresponent.
7. Compareu l'aplicació de síndrome de l'apartat anterior amb la que dona el reste de dividir per $\boldsymbol{g}(X)$, i digueu quina relació hi ha entre les dues matrius de control.

6.3 Codis cíclics

El concepte de codi cíclic es pot definir en general per a codis de bloc qualssevol. Donada una paraula $\boldsymbol{a} = (\mathtt{a}_1, \mathtt{a}_2, \ldots, \mathtt{a}_n) \in \mathbb{A}^n$ es denotarà $\gamma(\boldsymbol{a}) \in \mathbb{A}^n$ la paraula que s'obté en fer una permutació cíclica de les lletres cap a la dreta:

$$\gamma(\boldsymbol{a}) = (\mathtt{a}_n, \mathtt{a}_1, \mathtt{a}_2 \ldots, \mathtt{a}_{n-1}),$$

Per a tot enter $r \in \mathbb{Z}$ es denota $\gamma^r(\boldsymbol{a})$ la permutació corresponent, que mou les lletres cíclicament $|r|$ posicions a la dreta o esquerra segons el signe de r.

Definició 6.11 (Codi cíclic)**.** *Un codi $\mathcal{C} \subseteq \mathbb{A}^n$ és un codi cíclic si és tancat per les permutacions cícliques de les seves paraules: per a tota $\boldsymbol{c} \in \mathcal{C}$ també $\gamma(\boldsymbol{c}) \in \mathcal{C}$.*

Aquí es consideraran codis cíclics que siguin lineals. Com que γ és una aplicació lineal $\mathbb{F}^n \to \mathbb{F}^n$, per veure que un codi lineal és cíclic és suficient comprovar la condició de la definició en les paraules d'una base del codi. De fet, un codi lineal és cíclic si, i només si, totes les matrius obtingudes a partir d'una matriu generadora (resp. de control) fent permutacions cícliques de les columnes són també matrius generadores (resp. de control) del mateix codi.

Exemples 6.12. *Codis cíclics i codis no cíclics.*

1. *El codi de repetició* $\mathrm{Rep}_q(n)$ *sobre qualsevol alfabet q-ari* $\mathbb{A}$ *és cíclic.*
2. *El codi parell* $\mathrm{Par}_q(n)$ *sobre qualsevol alfabet* $\mathbb{Z}_q$ *o* $\mathbb{F}_q$ *es cíclic.*
3. *Es consideren els codis lineals binaris amb matrius generadores*

$$\mathbf{G}_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \qquad \mathbf{G}_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}.$$

Tot i que són linealment equivalents el primer és cíclic però el segon no ho és.

PROVA:

1. Les paraules codi són de la forma $(\mathtt{a}, \mathtt{a}, \ldots, \mathtt{a})$ i qualsevol permutació cíclica és la identitat. O sigui $\gamma(\boldsymbol{c}) = \boldsymbol{c}$ per a tota paraula codi. De fet el codi de repetició (i els seus subconjunts) és l'únic amb aquesta propietat: que totes les paraules codi queden fixes per permutació cíclica.
2. La propietat que caracteritza les paraules codi $\boldsymbol{c} = (\mathtt{c}_1, \mathtt{c}_2 \ldots, \mathtt{c}_n) \in \mathrm{Par}_q(n)$ és que tinguin suma $\mathtt{c}_1 + \mathtt{c}_2 + \cdots + \mathtt{c}_n = 0$. Però en aquest cas la paraula $\gamma(\boldsymbol{c})$ també satisfà aquesta propietat: $\mathtt{c}_n + \mathtt{c}_0 + \cdots + \mathtt{c}_{n-1} = 0$ ja que la suma és commutativa i és igual l'ordre en què se sumin els elements.
3. Tots dos codis són de tipus $[7, 4, 3]_2$ i són linealment equivalents a qualsevol codi de Hamming $\mathrm{Ham}_2(3)$: les columnes de les seves matrius de control contenen tots els vectors binaris de longitud 3 excepte el zero.

   Per comprovar que el primer és cíclic n'hi ha prou a veure que la permutació cíclica de cada fila és una paraula codi. Aquestes permutacions cícliques són `0100011`, que és la segona fila; `1010001`, que és la suma de la primera i la tercera; `1001011`, que és la suma de la primera i la quarta; i finalment `1000110`, que és la primera.

   En canvi el segon no ho és: cap permutació cíclica d'una fila és combinació lineal de files (n'hi hauria prou que n'hi hagués una que no ho fos). □

L'anell $\mathbb{F}[X]_{X^n-1}$. Per treballar amb codis cíclics és convenient identificar l'espai vectorial $\mathbb{F}^n \approx \mathbb{F}[X]_n$ amb l'anell $\mathbb{F}[X]_{X^n-1}$ de les classes de congruència de polinomis amb mòdul $N(X) = X^n - 1$. D'aquesta manera està definida la multiplicació d'un element $\boldsymbol{u}(X) \in$

$\mathbb{F}[X]_{X^n-1}$ per un polinomi $f(X) \in \mathbb{F}[X]$ qualsevol i la permutació cíclica d'un polinomi $\boldsymbol{u}(X) = \mathtt{u}_0 + \mathtt{u}_1 X + \cdots \mathtt{u}_{n-1} X^{n-1}$, que és el polinomi

$$\gamma\big(\boldsymbol{u}(X)\big) = \mathtt{u}_{n-1} + \mathtt{u}_0 X + \mathtt{u}_1 X^2 + \cdots + \mathtt{u}_{n-2} X^{n-1}$$

i s'obté fent el producte per X mòdul el polinomi $X^n - 1$:

$$\begin{aligned} X\boldsymbol{u}(X) &= \mathtt{u}_0 X + \mathtt{u}_1 X^2 + \cdots + \mathtt{u}_{n-2} X^{n-1} + \mathtt{u}_{n-1} X^n \\ &= (X^n - 1)\mathtt{u}_{n-1} + \mathtt{u}_{n-1} + \mathtt{u}_0 X + \cdots + \mathtt{u}_{n-2} X^{n-1} \equiv \gamma\big(\boldsymbol{u}(X)\big) \pmod{X^n - 1}. \end{aligned}$$

De forma més general es pot escriure el producte $X^r \boldsymbol{u}(X)$ com:

$$\begin{aligned} X^r \boldsymbol{u}(X) &= \mathtt{u}_0 X^r + \mathtt{u}_1 X^{r+1} + \cdots + \mathtt{u}_{n-2} X^{n+r-1} + \mathtt{u}_{n-1} X^{n+r} \\ &= \mathtt{u}_{n-r} + \mathtt{u}_{n-r+1} X + \cdots + \mathtt{u}_{n-1} X^{r-1} + \mathtt{u}_0 X^r + \cdots + \mathtt{u}_{n-r-1} X^{n-1} \\ &\quad + (X^n - 1)\mathtt{u}_{n-r} + (X^n - 1)\mathtt{u}_{n-r+1} X + \cdots + (X^n - 1)\mathtt{u}_{n-1} X^{r-1} \\ &\equiv \gamma^r\big(\boldsymbol{u}(X)\big) \pmod{X^n - 1} \end{aligned} \tag{11}$$

Això permet donar la caracterització següent:

Lema 6.13. *Un codi lineal $\mathcal{C} \subseteq \mathbb{F}[X]_{X^n-1}$ és cíclic si, i només si, és tancat pel producte per X: per a tot $\boldsymbol{c}(X) \in \mathcal{C}$ també $X\boldsymbol{c}(X) \in \mathcal{C}$.*

De fet, com que la multiplicació d'un polinomi $\boldsymbol{u}(X)$ per un polinomi qualsevol es pot obtenir multiplicant-lo primer varies vegades per X, multiplicant cadascun d'aquests productes per un escalar, i finalment sumant tots els resultats, gràcies a la linealitat del codi $\mathcal{C}$ la condició del lema 6.13 equival a què el codi sigui tancat pel producte per polinomis qualssevol: el producte $f(X)\boldsymbol{c}(X)$, vist com a element de $\mathbb{F}[X]_{X^n-1}$, pertany al codi $\mathcal{C}$ per a tota paraula codi $\boldsymbol{c}(X) \in \mathcal{C}$ i tot polinomi $f(X) \in \mathbb{F}[X]$.

En terminologia matemàtica de *teoria d'anells* els subespais $\mathbb{F}[X]_{N(X)}$ amb aquesta propietat s'anomenen *ideals*. En aquests termes, els codis cíclics són els subespais vectorials de $\mathbb{F}[X]_{X^n-1}$ que són ideals d'aquest anell.

Codis cíclics com a codis polinomials. En un codi polinomial $\mathrm{Pol}(n, \boldsymbol{g})$, si s'agafa la base $X^i \boldsymbol{g}(X)$ amb $i = 0, \ldots, k-1$, les permutacions cícliques $\gamma\big(X^i \boldsymbol{g}(X)\big) = X^{i+1}\boldsymbol{g}(X)$ són un altre element de la base per a $i = 0, \ldots, k-2$. Per tant, l'únic que falta per assegurar que aquest codi és cíclic és que $X^k \boldsymbol{g}(X) = \gamma\big(X^{k-1}\boldsymbol{g}(X)\big)$ també pertanyi al codi. A partir d'aquesta observació s'obté la caracterització dels codis cíclics següent:

Teorema 6.14 (Generadors de codis cíclics)**.** *Un codi lineal de longitud n és cíclic si, i només si, és un codi polinomial amb polinomi generador $\boldsymbol{g}(X)$ un divisor del polinomi $X^n - 1$.*

Prova: Sigui $\mathcal{C} \subseteq \mathbb{F}[X]_n$ un codi cíclic. Primer es vol veure que $\mathcal{C}$ és un codi generat per un polinomi. S'agafa el polinomi $\boldsymbol{g}(X) = \mathtt{g}_0 + \mathtt{g}_1 X + \cdots + \mathtt{g}_m X^m$ amb $m \leqslant n-1$ que correspongui a una paraula codi no nul·la. Es veurà que $\mathcal{C}$ està generat per aquest polinomi.

La condició que el codi sigui cíclic assegura que tots els polinomis $X^r \boldsymbol{g}(X)$, vistos mòdul $X^n - 1$, pertanyen al codi per a tot $r \geqslant 0$ (el lema 6.13 ens ho garanteix) i per tant tots els

múltiples de $\boldsymbol{g}(X)$ pertanyen al codi. Sigui $\boldsymbol{c}(X) \in \mathcal{C} \subseteq \mathbb{F}[X]_n$ una paraula codi. Es fa la divisió euclidiana per $\boldsymbol{g}(X)$ i s'obté:

$$\boldsymbol{c}(X) = \boldsymbol{g}(X)\boldsymbol{q}(X) + \boldsymbol{r}(X), \quad \deg \boldsymbol{r} < m.$$

Com que $\boldsymbol{c}(X)$ pertany al codi per hipòtesi i $\boldsymbol{g}(X)\boldsymbol{q}(X)$ també hi pertany per ser múltiple de $\boldsymbol{g}(X)$ es dedueix que el reste $\boldsymbol{r}(X)$ pertany al codi per linealitat. La hipòtesi que $\boldsymbol{g}(X)$ s'havia agafat com un element no nul de grau m, implica que $\boldsymbol{r}(X)$ és el polinomi zero (altrament tindria grau més gran o igual a m) i, per tant, $\boldsymbol{g}(X)$ divideix el polinomi $\boldsymbol{c}(X)$.

S'ha vist doncs que el codi consisteix en els polinomis de grau $\leqslant n-1$ que són múltiples de $\boldsymbol{g}(X)$, i és per tant el codi $\mathrm{Pol}(n, \boldsymbol{g})$. Es pot suposar, sense perdre generalitat, que $\boldsymbol{g}(X)$ és mònic. Falta veure que divideix $X^n - 1$.

Tal com ens diu el lema 6.13, el polinomi $X^{n-m}\boldsymbol{g}(X)$ és un polinomi del codi. Si s'escriu com:

$$\begin{aligned} X^{n-m}\boldsymbol{g}(X) &= X^{n-m}(g_0 + g_1 X + \cdots + g_{m-1}X^{m-1} + X^m) \\ &= X^n - 1 + (1 + g_0 X^{n-m} + \cdots + g_{m-1}X^{n-1}) \end{aligned}$$

el terme entre parèntesi és el reste $\boldsymbol{r}(X)$, amb $\deg \boldsymbol{r} < n$, obtingut de dividir mòdul $X^n - 1$ i per tant pertany al codi (veieu l'equació (11)), així que és divisible per $\boldsymbol{g}(X)$. Es dedueix que $X^n - 1$ també és divisible per $\boldsymbol{g}(X)$.

Recíprocament, sigui ara $\mathcal{C} = \mathrm{Pol}(n, \boldsymbol{g})$ un codi generat per un polinomi $\boldsymbol{g}(X)$ mònic que divideix $X^n - 1$. Sigui $\boldsymbol{h}(X)$ el quocient, de manera que $\boldsymbol{g}(X)\boldsymbol{h}(X) = X^n - 1$ i $\boldsymbol{h}$ també és mònic. Siguin $m = \deg \boldsymbol{g}$ i $k = \deg \boldsymbol{h}$, amb $m + k = n$.

Els polinomis $\boldsymbol{g}(X), X\boldsymbol{g}(X), \ldots, X^{k-1}\boldsymbol{g}(X)$ són una base del codi $\mathcal{C}$. Per veure que $\mathcal{C}$ és cíclic s'ha de veure que $X^k\boldsymbol{g}(X)$, vist mòdul $X^n - 1$, també pertany al codi. La identitat

$$\boldsymbol{g}(X)\boldsymbol{h}(X) = X^k\boldsymbol{g}(X) + \mathtt{h}_{k-1}X^{k-1}\boldsymbol{g}(X) + \cdots + \mathtt{h}_1 X\boldsymbol{g}(X) + \mathtt{h}_0\boldsymbol{g}(X) = X^n - 1$$

considerada mòdul $X^n - 1$ assegura que $X^k\boldsymbol{g}(X)$ és combinació lineal dels elements de la base i, per tant, pertany al codi. □

La caracterització dels codis cíclics lineals del teorema 6.14 permet trobar tots els codis cíclics de longitud prefixada n a partir de la descomposició en primers del polinomi $X^n - 1$ a l'anell de polinomis $\mathbb{F}[X]$:

Corol·lari 6.15. *Sigui $X^n - 1 = \prod \boldsymbol{p}_i(X)^{a_i}$ la descomposició en producte de polinomis primers a $\mathbb{F}[X]$. Hi ha exactament $\prod(a_i + 1)$ codis cíclics diferents de longitud n, corresponents als diferents divisors mònics del polinomi $X^n - 1$.*

Prova: En efecte, si un polinomi descompon en factors primers com $f(X) = \prod \boldsymbol{p}_i(X)^{a_i}$ els seus divisors s'obtenen posant-hi cada primer $p_i(X)$ un nombre de vegades entre 0 i a_i, el qual dona $a_i + 1$ possibilitats.

Si $p \nmid n$ aleshores el polinomi $X^n - 1$ no pot tenir factors primers de multiplicitat > 1 ja que el màxim comú divisor amb la seva derivada és $\gcd(X^n - 1, nX^{n-1}) = 1$.

En general, si es posa $n = p^e n_0$ amb $p \nmid n_0$ aleshores es té $X^n - 1 = (X^{n_0} - 1)^{p^e}$ i tots els seus factors tenen la mateixa multiplicitat p^e. Per tant el nombre que diu l'enunciat és sempre de la forma $(p^e + 1)^r$ amb r el nombre de factors primers diferents. □

Exemple 6.16. *Al cos finit $\mathbb{F}_2$ es té la descomposició en primers:*

$$X^7 - 1 = (X+1)(X^3+X+1)(X^3+X^2+1).$$

Per tant hi ha 8 codis binaris cíclics de longitud 7:

1. *El factor trivial constant igual a 1 correspon al codi total $\mathcal{C} = \mathbb{F}_2[X]_7$ i el factor total X^7-1 correspon al codi trivial $\mathcal{C} = \{\mathbf{0}\} = \{\mathtt{0000000}\}$.*
2. *El factor $X+1$ correspon al codi binari parell.*
3. *El factor de grau sis $(X^3+X+1)(X^3+X^2+1) = (X^7-1)/(X+1) = X^6+X^5+X^4+X^3+X^2+X+1$ correspon al codi de repetició.*
4. *Els factors de grau tres X^3+X+1 i X^3+X^2+1 generen dos codis de tipus $[7,4,3]$, tots dos linealment equivalents al codi de Hamming $\mathrm{Ham}_2(3)$.*
5. *Els factors de grau quatre $X^4+X^3+X^2+1 = (X^3+X+1)(X+1)$ i $X^4+X^2+X+1 = (X^3+X^2+1)(X+1)$ generen codis de tipus $[7,3,4]$, linealment equivalents entre ells i duals dels codis de l'apartat anterior.*

En els codis generats per polinomis s'ha vist com construir aplicacions de codificació i matrius generadores a partir del polinomi generador. Per a codis cíclics es poden fer construccions anàlogues amb el:

Definició 6.17 (Polinomi de control)**.** *Sigui $\mathcal{C}$ un codi cíclic de longitud n amb polinomi generador $\boldsymbol{g}(X)$. El polinomi $\boldsymbol{h}(X) = (X^n-1)/\boldsymbol{g}(X)$ es diu* polinomi de control *del codi.*

El polinomi de control té grau $\deg \boldsymbol{h} = n - m = k$, la dimensió del codi.

Proposició 6.18. *Sigui $\mathcal{C}$ un codi cíclic amb polinomi de control $\boldsymbol{h}(X)$. Aleshores*

1. $\mathcal{C} = \left\{\boldsymbol{u}(X) \in \mathbb{F}[X]_n : \boldsymbol{u}(X)\boldsymbol{h}(X) \equiv \mathbf{0} \pmod{X^n-1}\right\}$;
2. *Una matriu de control per al codi $\mathcal{C}$ és*

$$\mathbf{H} = \begin{bmatrix} \mathtt{h}_k & \mathtt{h}_{k-1} & \cdots & \mathtt{h}_0 & \mathtt{0} & \cdots & \mathtt{0} \\ \mathtt{0} & \mathtt{h}_k & \cdots & \mathtt{h}_1 & \mathtt{h}_0 & \cdots & \mathtt{0} \\ \vdots & \ddots & \ddots & & \ddots & \ddots & \vdots \\ \mathtt{0} & \cdots & \mathtt{0} & \mathtt{h}_k & \cdots & \mathtt{h}_1 & \mathtt{h}_0 \end{bmatrix}, \tag{12}$$

on $\boldsymbol{h}(X) = \mathtt{h}_0 + \mathtt{h}_1 X + \cdots + \mathtt{h}_{k-1}X^{k-1}$.

Prova: Per veure el primer apartat, n'hi ha prou a observar que un element $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$ pertany al codi $\mathcal{C}$ si, i només si, és de la forma $\boldsymbol{u}(X) = \boldsymbol{m}(X)\boldsymbol{g}(X)$ per a un polinomi $\boldsymbol{m}(X) \in \mathbb{F}[X]_k$. Multiplicant per $\boldsymbol{h}(X)$ això equival a què el polinomi $\boldsymbol{u}(X)\boldsymbol{h}(X)$ sigui de la forma $\boldsymbol{m}(X)\boldsymbol{g}(X)\boldsymbol{h}(X) = \boldsymbol{m}(X)(X^n-1)$ per a algun polinomi $\boldsymbol{m}$, que equival a què el producte $\boldsymbol{u}(X)\boldsymbol{h}(X)$ sigui divisible per X^n-1.

Ara es veurà el segon apartat. Per fer-ho s'introdueix la notació següent: per a un element $\boldsymbol{v}(X) = \mathtt{v}_0 + \mathtt{v}_1 X + \cdots + \mathtt{v}_{n-1}X^{n-1} \in \mathbb{F}[X]_n$ es denotarà $\overline{\boldsymbol{v}}(X) = \mathtt{v}_{n-1} + \mathtt{v}_{n-2}X + \cdots + \mathtt{v}_0 X^{n-1}$ el polinomi recíproc $X^{n-1}\boldsymbol{v}(\frac{1}{X})$.

Aleshores el fet que el producte de dos polinomis $\boldsymbol{u}(X), \boldsymbol{v}(X) \in \mathbb{F}[X]_n$ sigui divisible per $X^n - 1$ es pot caracteritzar de la manera següent:

$$X^n - 1 \mid \boldsymbol{u}(X)\boldsymbol{v}(X) \;\Leftrightarrow\; \left\langle \gamma^i(\boldsymbol{u}), \gamma^j(\overline{\boldsymbol{v}}) \right\rangle = \mathtt{0}\ \forall i, j \in \mathbb{Z} \;\Leftrightarrow\; \left\langle \boldsymbol{u}, \gamma^k(\overline{\boldsymbol{v}}) \right\rangle = \mathtt{0}\ \forall k \in \mathbb{Z}.$$

En efecte, el sumatori $\boldsymbol{u}(X)\boldsymbol{v}(X) = \sum_{i,j=0}^{n-1} \mathtt{u}_i \mathtt{v}_j X^{i+j}$ es pot escriure com:

$$\boldsymbol{u}(X)\boldsymbol{v}(X) = \sum_{i+j<n} \mathtt{u}_i \mathtt{v}_j X^{i+j} + \sum_{i+j\geqslant n} \mathtt{u}_i \mathtt{v}_j X^{i+j-n} + \left(\sum_{i+j\geqslant n} \mathtt{u}_i \mathtt{v}_j X^{i+j-n} \right) (X^n - 1).$$

Aleshores agafant els dos primers sumatoris es té una congruència mòdul $X^n - 1$:

$$\begin{aligned}
\boldsymbol{u}(X)\boldsymbol{v}(X) \equiv \quad & (\mathtt{u}_0\mathtt{v}_0 + \mathtt{u}_1\mathtt{v}_{n-1} + \mathtt{u}_2\mathtt{v}_{n-2} + \cdots + \mathtt{u}_{n-1}\mathtt{v}_1) \\
& + (\mathtt{u}_0\mathtt{v}_1 + \mathtt{u}_1\mathtt{v}_0 + \mathtt{u}_2\mathtt{v}_{n-1} + \cdots + \mathtt{u}_{n-1}\mathtt{v}_2)X \\
& \cdots \\
& + (\mathtt{u}_0\mathtt{v}_{n-2} + \mathtt{u}_1\mathtt{v}_{n-3} + \mathtt{u}_2\mathtt{v}_{n-4} + \cdots + \mathtt{u}_{n-1}\mathtt{v}_{n-1})X^{n-2} + \\
& + (\mathtt{u}_0\mathtt{v}_{n-1} + \mathtt{u}_1\mathtt{v}_{n-2} + \mathtt{u}_2\mathtt{v}_{n-3} + \cdots + \mathtt{u}_{n-1}\mathtt{v}_0)X^{n-1} \\
= \; & \left\langle \boldsymbol{u}, \gamma(\overline{\boldsymbol{v}}) \right\rangle + \left\langle \boldsymbol{u}, \gamma^2(\overline{\boldsymbol{v}}) \right\rangle X + \cdots \left\langle \boldsymbol{u}, \gamma^{n-1}(\overline{\boldsymbol{v}}) \right\rangle X^{n-2} + \left\langle \boldsymbol{u}, \overline{\boldsymbol{v}} \right\rangle X^{n-1}
\end{aligned}$$

i es dedueix que el producte és divisible per $X^n - 1$ si, i només si, tots els coeficients $\left\langle \boldsymbol{u}, \gamma^k(\overline{\boldsymbol{v}}) \right\rangle$ d'aquest darrer polinomi són iguals a zero per a $k = 0, \ldots, n-1$.

Ara es veu primer que la matriu $\mathbf{H}$ té les dimensions adequades: $m \times n$. A més clarament té rang màxim igual a m. Es considera la matriu generadora $\mathbf{G}$ associada a $\boldsymbol{g}(X)$. Les files de $\mathbf{G}$ corresponen a les paraules $\gamma^i(\boldsymbol{g})$ per a exponents $0 \leqslant i \leqslant k-1$. Les files de $\mathbf{H}$ corresponen a les paraules $\gamma^j(\overline{\boldsymbol{h}})$ per a exponents $0 \leqslant j \leqslant m-1$. Tenint en compte l'observació anterior tots els productes escalars d'una fila de l'una per una de l'altra són zero, i per tant el producte $\mathbf{G} \cdot \mathbf{H}^{\mathsf{T}} = \mathbf{0}$ és la matriu zero. Això demostra que $\mathbf{H}$ és matriu de control del codi. □

Codis cíclics escurçats. Molts codis polinomials es poden obtenir escurçant codis cíclics. Per veure com es fa això cal un lema tècnic sobre polinomis a coeficients en un cos finit:

Lema 6.19. *Tot polinomi primer $\boldsymbol{p}(X) \in \mathbb{F}_q[X]$ de grau r divideix el polinomi $X^{q^r} - X$.*

Excepte per al polinomi X, de grau $r = 1$, que divideix $X^{q^r} - X = X(X^{q-1} - 1)$, d'acord amb l'enunciat del lema, però no divideix cap polinomi de la forma $X^n - 1$, en tots els altres casos el polinomi primer $\boldsymbol{p}(X)$ és relativament primer amb X i per tant com que divideix $X^{q^r} - X = X(X^{q^r-1} - 1)$ ha de dividir el segon factor, de la forma $X^n - 1$ amb $n = q^r - 1$.

Corol·lari 6.20. *Tot polinomi $P(X)$ amb $P(\mathtt{0}) \neq \mathtt{0}$ i que sigui producte de primers diferents genera un codi cíclic de longitud* $\operatorname{lcm}\{q^r - 1\}$ *on r són els graus dels factors primers de P.*

Prova: Primer s'observa que tot polinomi $X^n - 1$ divideix els polinomis $X^m - 1$ per a tots els exponents m múltiples de n. En efecte, si $m = nt$ es té

$$X^m - 1 = (X^n - 1)(1 + X^n + X^{2n} + \cdots + X^{(t-1)n}).$$

Sigui $P(X) = \prod \boldsymbol{p}_i(X)$ la descomposició en polinomis primers diferents i sigui r_i el grau de $\boldsymbol{p}_i$. Cada $\boldsymbol{p}_i(X)$, que no pot ser el polinomi X per la hipòtesi $P(0) \neq 0$, divideix el polinomi $X^{q^{r_i}-1} - 1$ i per tant divideix $X^m - 1$ per a tot enter m múltiple de $q^{r_i} - 1$. Agafant $m = \operatorname{lcm}\{q^{r_i} - 1\}$ tots els polinomis $\boldsymbol{p}_i(X)$ divideixen $X^m - 1$ i, com que són primers entre ells, el seu producte $P(X)$ també divideix aquest polinomi. □

Per tant, tot codi $\mathrm{Pol}(n, \boldsymbol{g})$ amb polinomi generador $\boldsymbol{g}$ que satisfaci les condicions del lema (no divisible per X i lliure de quadrats) és un codi cíclic escurçat, que s'obté escurçant un codi cíclic amb el mateix polinomi generador i de longitud m l'enter del lema.

Problemes

6.10. Trobeu tots els codis binaris cíclics de longitud 5 i digueu el tipus $[n, k, d]$ de cadascun.

6.11. Sigui $\mathcal{C}$ un codi cíclic binari de longitud senar. Demostreu que $\mathcal{C}$ conté alguna paraula de pes senar si, i només si, conté la paraula $\mathtt{111\cdots1}$.

6.12. Comproveu que el polinomi $1 + X + X^2 + X^3$ genera un codi binari cíclic de longitud 8. Codifiqueu la paraula `10101` amb codificació sistemàtica.

6.13. Calculeu la distància mínima del codi binari cíclic de longitud 6 generat pel polinomi $1 + X + X^3 + X^4$.

6.4 Codis de Golay

Els codis de Golay són dos codis concrets, un de binari i un de ternari, introduïts per Golay el 1949 a [20]. Són els únics codis lineals perfectes que existeixen, apart dels codis totals, els de repetició binaris de longitud senar, i els de Hamming. Tots dos són codis cíclics.

Definició 6.21 (Codi de Golay binari)**.** *El codi de Golay binari* Gol_2 *és el codi binari de longitud* 23 *generat pel polinomi*

$$\boldsymbol{g}(X) = 1 + X^2 + X^4 + X^5 + X^6 + X^{10} + X^{11} \in \mathbb{F}_2[X].$$

Lema 6.22. Gol_2 *és un codi cíclic perfecte de tipus* $[23, 12, 7]_2$*. El codi estès* $\mathrm{Gol}_2^{\mathrm{ev}}$ *és un codi de tipus* $[24, 12, 8]_2$*.*

En l'article [20] Golay introdueix aquest codi donant la matriu de control sistemàtica

$\mathbf{H} = \left[\mathbf{I}_{11}\middle|\mathbf{H}'\right]$ amb

$$\mathbf{H}' = \left[\begin{array}{cccccccccccc} 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{array}\right] \in \mathrm{Mat}_{11\times 12}(\mathbb{F}_2).$$

És fàcil comprovar que és el codi cíclic amb polinomi de control $\boldsymbol{h}(X) = 1 + X + X^2 + X^3 + X^4 + X^5 + X^6 + X^7 + X^8 + X^9 + X^{10}$ veient que aquesta matriu s'obté a partir de la matriu de control construïda a partir d'aquest polinomi i aplicant reducció de Gauss-Jordan.
PROVA: La longitud i la dimensió són clares a partir de la definició. Es pot calcular la distància mínima per força bruta: calculant totes les $2^{11} = 2048$ paraules codi i el seu pes. La comprovació que es perfecte s'ha fet a l'exercici **1.14**. □

Definició 6.23 (Codi de Golay ternari)**.** *El codi de Golay ternari* Gol_3 *és el codi ternari de longitud* 11 *generat pel polinomi*

$$\boldsymbol{g}(X) = 2 + X^2 + 2X^3 + X^4 + X^5 \in \mathbb{F}_3[X].$$

Lema 6.24. Gol_3 *és un codi cíclic perfecte de tipus* $[11, 6, 5]_3$*. El codi estès* $\mathrm{Gol}_3^{\mathrm{ev}}$ *és de tipus* $[12, 6, 6]_3$*.*

PROVA: La longitud i la dimensió són clares a partir de la definició. Es pot calcular la distància mínima per força bruta: calculant totes les $3^6 = 729$ paraules codi i el seu pes. Per veure que el codi estès té distància mínima 6 n'hi ha prou a veure que totes les paraules de pes 5 del codi Gol_3 tenen suma de dígits diferent de zero. La comprovació que es perfecte s'ha fet a l'exercici **1.14**. □

En l'article [20] Golay introdueix aquest codi donant la matriu de control:

$$\mathbf{H} = \left[\begin{array}{ccccccccccc} 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 2 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 2 & 1 & 0 & 2 \\ 0 & 0 & 1 & 0 & 0 & 1 & 2 & 1 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 & 0 & 1 & 2 & 0 & 1 & 2 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 & 2 & 2 & 1 & 1 \end{array}\right] \in \mathrm{Mat}_{5\times 11}(\mathbb{F}_3).$$

És fàcil comprovar que és el codi cíclic amb polinomi de control $\boldsymbol{h}(X) = 1 + X + 2X^3 + 2X^4 + X^5 + X^6$ veient que aquesta matriu s'obté a partir de la matriu de control construïda a partir d'aquest polinomi i aplicant reducció de Gauss-Jordan.

Codis perfectes. Per a alfabets amb nombre d'elements $q = p^e$ una potència de primer, que inclou el cas dels codis lineals, se sap que no existeixen més codis perfectes que els coneguts. Això es demostra estudiant l'equació

$$M \sum_{i=0}^{\tau} \binom{n}{i} (q-1)^i = q^n$$

que ha de complir un codi perfecte de tipus $(n, M, d)_q$, on $\tau = \lfloor \frac{d-1}{2} \rfloor$ és la capacitat correctora. Quan $q = p^e$ és potència de primer es veu que aquesta equació no té cap altra solució que les ja conegudes i, a més, la solució corresponent a la terna $(90, 2^{78}, 5)_2$, però no és difícil comprovar que no hi ha cap codi binari amb aquests paràmetres.

El cas general, d'alfabets amb nombre d'elements no potència de primer, no està resolt en general. Veure [3, Secció 11.5].

6.5 Reed-Solomon en versió BCH

Els anys 1959 i 1960 Hocquenghem [22] i Bose i Ray-Chaudhuri [19] introdueixen una classe de codis cíclics en què el polinomi generador es defineix a partir de les seves arrels en un cos finit $\mathbb{F}_{2^e}$ de característica 2. Un any més tard Gorenstein i Zierler generalitzen aquesta construcció a altres cossos i observen que molts codis de Reed-Solomon es poden obtenir també d'aquesta manera. En la versió BCH les paraules dels codis de Reed-Solomon, vistes com a polinomis, es caracteritzen per la condició que s'anul·lin en totes les potències d'un element del cos $\mathbb{F}$.

El fet que hi ha molts algorismes eficients de correcció d'errors per als codis BCH, i que aquests són codis cíclics, fa que moltes vegades els codis de Reed-Solomon es tracten com a cas particular dels codis BCH. Tot i això s'han anat trobant també algorismes eficients específics per a corregir errors en els codis de Reed-Solomon donats en la seva versió original, on les paraules s'obtenen avaluant polinomis, els quals no sempre són cíclics.

En aquesta secció s'introdueixen els codis BCH que donen lloc a codis Reed-Solomon i en la següent 6.6 es veurà el cas general.

Codis polinomials determinats per arrels a $\mathbb{F}$. Siguin $\beta_1, \ldots, \beta_m \in \mathbb{F}$ elements diferents i sigui $n \geqslant m$. Es considera el codi lineal format pels polinomis de $\mathbb{F}[X]_n$ que s'anul·len en aquests elements

$$\mathcal{C} = \big\{ \boldsymbol{u}(X) \in \mathbb{F}[X]_n : \boldsymbol{u}(\beta_i) = 0 \ \forall i = 1, \ldots, m \big\} \subseteq \mathbb{F}[X]_n. \tag{13}$$

La condició $\boldsymbol{u}(\beta_i) = 0$ equival a què el polinomi $X - \beta_i$ divideixi $\boldsymbol{u}$. Per tant, les paraules codi són els polinomis de $\mathbb{F}[X]_n$ que es divideixen pel polinomi

$$\boldsymbol{g}(X) = (X - \beta_1)(X - \beta_2) \cdots (X - \beta_m)$$

i el codi $\mathcal{C}$ és el codi polinomial $\mathrm{Pol}(n, \boldsymbol{g})$ amb polinomi generador $\boldsymbol{g}(X)$.

Els codis polinomials que es poden construir d'aquesta manera són els que el polinomi generador descompon en producte de polinomis de grau 1: té tantes arrels diferents al cos $\mathbb{F}$ com el seu grau.

Amb aquesta construcció de codis polinomials es veu que la matriu següent és una matriu de control:

$$\mathbf{H} = \begin{bmatrix} 1 & \beta_1 & \beta_1^2 & \cdots & \beta_1^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \beta_m & \beta_m^2 & \cdots & \beta_m^{n-1} \end{bmatrix} \in \mathrm{Mat}_{m\times n}(\mathbb{F}).$$

En efecte, donat un polinomi $\boldsymbol{u}(X) = \mathtt{u}_0 + \mathtt{u}_1 X + \cdots + \mathtt{u}_{n-1}X^{n-1} \in \mathbb{F}[X]_n$ es té

$$\boldsymbol{u}(X) \in \mathcal{C} \;\Leftrightarrow\; \boldsymbol{u}(\beta_i) = \mathtt{u}_0 + \mathtt{u}_1\beta_i + \cdots + \mathtt{u}_{n-1}\beta_i^{n-1} = 0 \;\forall i \;\Leftrightarrow\; \mathbf{H}\cdot\mathbf{u} = \mathbf{0},$$

on $\mathbf{H}\cdot\mathbf{u}$ és el producte de la matriu $\mathbf{H}$ pel vector columna $\mathbf{u}$ dels coeficients $\mathtt{u}_i$ del polinomi.

Aquesta matriu de control és la que correspon a l'aplicació de síndrome $\mathrm{syn}\colon \mathbb{F}[X]_n \to \mathbb{F}^m$ que envia cada polinomi de grau $< n$ als valors d'aquest polinomi en els elements β_i:

$$\mathrm{syn}\big(\boldsymbol{u}(X)\big) = \big(\boldsymbol{u}(\beta_1), \boldsymbol{u}(\beta_2), \ldots, \boldsymbol{u}(\beta_m)\big).$$

Codis BCH. Els codis BCH són els que s'obtenen d'aquesta manera agafant com a arrels β_i totes les potències consecutives diferents d'un element $\beta \in \mathbb{F}$:

Definició 6.25. *Sigui $\beta \in \mathbb{F}$ un element no nul. Sigui $n = \mathrm{ord}(\beta)$ el seu ordre. Per a cada enter d amb $1 \leqslant d \leqslant n+1$ es defineix el codi:*

$$\mathrm{BCH}(\beta, d) = \big\{\boldsymbol{c}(X) \in \mathbb{F}[X]_n : \boldsymbol{c}(\beta) = \boldsymbol{c}(\beta^2) = \cdots = \boldsymbol{c}(\beta^{d-1})\big\}.$$

Recordi's que l'ordre de tot element $\beta \in \mathbb{F}^*$ és un divisor de $q-1$, i per tant la longitud dels codis BCH té aquesta propietat. Quan s'agafa β un element primitiu del cos, d'ordre $n = q-1$, els codis BCH que s'obtenen se solen anomenar *codis BCH primitius*.

Proposició 6.26. *El codi $\mathcal{C} = \mathrm{BCH}(\beta, d)$ és un codi cíclic MDS de dimensió $k = n - d + 1$ amb distància mínima d.*

Prova: $\mathcal{C}$ és el codi polinomial determinat per les arrels $\beta, \beta^2, \ldots, \beta^{d-1}$, amb polinomi generador

$$\boldsymbol{g}(X) = (X-\beta)(X-\beta^2)\cdots(X-\beta^{d-1})$$

de grau $m = d-1$. Per tant la seva codimensió és $m = d-1$ i la seva dimensió és $k = n - m = n - d + 1$.

Com que $(\beta^r)^n = 1$ per a tot exponent r totes les arrels de $\boldsymbol{g}(X)$ són arrels del polinomi $X^n - 1$ i, per tant, $\boldsymbol{g}(X) \mid (X^n - 1)$. El teorema 6.14 diu que $\mathcal{C}$ és un codi cíclic.

Només falta veure que la seva distància mínima és igual a d. La demostració d'això es basa en les propietats de les matrius de Vandermonde. La matriu (15) corresponent a aquestes arrels β^i és

$$\mathbf{H} = \begin{bmatrix} 1 & \beta & \beta^2 & \cdots & \beta^{n-1} \\ 1 & \beta^2 & \beta^4 & \cdots & \beta^{2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \beta^{d-1} & \beta^{(d-1)2} & \cdots & \beta^{(d-1)(n-1)} \end{bmatrix} \in \mathrm{Mat}_{(d-1)\times n}(\mathbb{E}).$$

Tota submatriu quadrada formada per $d-1$ columnes és de la forma

$$\begin{bmatrix} \beta^{j_1} & \beta^{j_2} & \dots & \beta^{j_{d-1}} \\ \beta^{2j_1} & \beta^{2j_2} & \dots & \beta^{2j_{d-1}} \\ \vdots & \vdots & & \vdots \\ \beta^{(d-1)j_1} & \beta^{(d-1)j_2} & \dots & \beta^{(d-1)j_{d-1}} \end{bmatrix}.$$

per a alguns índexs $j_1, \dots, j_{d-1}$ amb $0 \leqslant j_1 < \cdots < j_{d-1} \leqslant n-1$. És una matriu de Vandermonde que, com que els β^{j_i} són tots no nuls i diferents entre ells, té determinant $\neq 0$. Per tant $d-1$ columnes qualssevol de $\mathbf{H}$ són sempre independents. Existeixen d columnes dependents; de fet, com que la matriu té $d-1$ files, d columnes qualssevol són sempre dependents. De la proposició 5.28 es dedueix que la distància mínima és d. □

Els codis $\mathrm{BCH}(\beta, d)$ són una altra manera de veure els codis de Reed-Solomon que s'obtenen en avaluar polinomis en les potències successives de l'element β^{-1}:

Teorema 6.27. *Sigui $\beta \in \mathbb{F}$ un element d'ordre n. Sigui d un enter amb $1 \leqslant d \leqslant n+1$. Sigui $k = n - d + 1$. Aleshores,*

$$\mathrm{BCH}(\beta, d) = \mathrm{RS}\left(k, \{\beta^i\}_{0 \leqslant i \leqslant n-1}\right).$$

Prova: Com que les potències β^i són totes diferents el codi RS està ben definit. Tots dos són codis de tipus $[n, k, d]_q$. Sigui $m = n - k = d - 1$ la seva codimensió. Els codis són:

$$\begin{aligned} \mathcal{C}_1 &= \left\{\boldsymbol{u}(X) \in \mathbb{F}[X]_n : \boldsymbol{u}(\beta) = \boldsymbol{u}(\beta^2) = \cdots = \boldsymbol{u}(\beta^m) = 0\right\}, \\ \mathcal{C}_2 &= \left\{ \mathrm{enc}\left(\boldsymbol{m}(X)\right) = \left(\boldsymbol{m}(\mathtt{1}), \boldsymbol{m}(\beta), \boldsymbol{m}(\beta^2), \dots, \boldsymbol{m}(\beta^{n-1})\right) \in \mathbb{F}^n : \boldsymbol{m}(X) \in \mathbb{F}[X]_k \right\}. \end{aligned}$$

La demostració fa servir la transformada de Fourier discreta corresponent als elements β i β^{-1}. Siguin $\mathcal{F}(\beta) = \left[\beta^{-ij}\right]_{0 \leqslant i,j < n} \in \mathrm{Mat}_n(\mathbb{F})$ i $\mathcal{F}(\beta^{-1}) = \left[\beta^{ij}\right]_{0 \leqslant i,j < n}$ les matrius corresponents. El seu producte és $\mathcal{F}(\beta)\mathcal{F}(\beta^{-1}) = n\,\mathbf{I}_n$. Com que $n = \mathrm{ord}(\beta)$ és un divisor de $q-1$, no és divisible per la característica de $\mathbb{F}$ i $n \neq 0$ a $\mathbb{F}$. Per tant les transformades de Fourier discretes són invertibles i una és essencialment l'inversa de l'altra.

Multiplicar $\mathcal{F}(\beta^{-1})$ pel vector columna $\mathbf{u}$ dels coeficients d'un polinomi $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$ dona com a resultat els valors d'aquest polinomi en les potències $\beta^0, \beta, \beta^2, \dots, \beta^{n-1}$:

$$\mathcal{F}(\beta) \cdot \mathbf{u} = \left(\boldsymbol{u}(1), \boldsymbol{u}(\beta), \boldsymbol{u}(\beta^2), \dots, \boldsymbol{u}(\beta^{n-1})\right).$$

Un element $\boldsymbol{c} = \mathrm{enc}\left(\boldsymbol{m}(X)\right) \in \mathcal{C}_2$ s'obté amb la multiplicació matricial $\mathbf{c} = \mathcal{F}(\beta^{-1}) \cdot \mathbf{m}^{\mathrm{ext}}$, on $\mathbf{m}^{\mathrm{ext}}$ és el vector columna dels coeficients del polinomi $\boldsymbol{m}(X) = \sum_{i=0}^{k-1} \mathtt{m}_i X^i$ vist com a polinomi de $\mathbb{F}[X]_n$, amb la resta de coeficients fins a $n-1$ iguals a zero:

$$\mathbf{m}^{\mathrm{ext}} = (\mathtt{m}_0, \mathtt{m}_1, \dots, \mathtt{m}_{k-1}, \mathtt{0}, \dots, \mathtt{0}) \in \mathbb{F}^n.$$

Identificant $\boldsymbol{c}$ amb un polinomi $\boldsymbol{c}(X) \in \mathbb{F}[X]_n$ de la manera habitual, el producte $\mathcal{F}(\beta) \cdot \mathbf{c}$ té coordenades $\boldsymbol{c}(1), \boldsymbol{c}(\beta^{-1}), \boldsymbol{c}(\beta^{-2}), \dots, \boldsymbol{c}(\beta^{-(n-1)})$. Tenint en compte la relació entre les transformades de Fourier aquest producte és $n\,\mathbf{m}^{\mathrm{ext}}$. Per tant les seves últimes $m = n - k$

coordenades, amb exponents $-(n-i) = i-n$ per a $i = 1, \ldots, m$, són zero. Com que $\beta^n = 1$ això vol dir que $\boldsymbol{c}(\beta) = \cdots = \boldsymbol{c}(\beta^m) = 0$.

Això demostra que tota paraula $\boldsymbol{c} = \operatorname{enc}\big(\boldsymbol{m}(X)\big)$ del codi $\mathcal{C}_2$, vista com a polinomi de $\mathbb{F}[X]_n$, s'anul·la en les m primeres potències de β, i per tant pertany al codi $\mathcal{C}_1$. Com que tots dos codis tenen la mateixa dimensió, han de ser iguals. □

Usant aquest teorema 6.27 es poden obtenir com a codis BCH els codis RS $\big(k, \{\alpha_i\}\big)$ amb elements α_i totes les potències d'algun element $\beta \in \mathbb{F}^*$. Del teorema es dedueix que aquests codis RS $\big(k, \{\beta^i\}_{0 \leqslant i < \operatorname{ord}(\beta)}\big)$ són codis cíclics.

En particular, agafant com a β un element primitiu del cos s'obtenen els codis de Reed-Solomon de longitud $n = q-1$ que consisteixen en avaluar polinomis de grau $< k$ en tots els elements no nuls de $\mathbb{F}$. Moltes vegades a la literatura s'anomenen *codis de Reed-Solomon* de tipus BCH només aquests codis, obtinguts avaluant polinomis de $\mathbb{F}[X]_k$ en totes les potències d'un element primitiu del cos, que tenen per tant longitud $q-1$, i s'anomenen *codis de Reed-Solomon escurçats* els codis d'altres longituds $n < q-1$ que s'obtenen en escurçar-los en algunes de les darreres posicions, i que consisteixen a avaluar els polinomis de $\mathbb{F}[X]_k$ només en les primeres potències $\beta^0, \ldots, \beta^{n-1}$.

Problemes

6.6 Codis BCH

En aquesta secció es veuen els codis BCH més generals. Són codis polinomials cíclics en què les paraules codi, vistes com a polinomis $\boldsymbol{c}(X) \in \mathbb{F}[X]_n$ amb coeficients en un cos finit $\mathbb{F}$, es caracteritzen per la condició que s'anul·lin en les potències d'un element que pertany a una extensió $\mathbb{E}$ del cos $\mathbb{F}$ dels coeficients.

Aquesta construcció permet garantir que tenen una distància mínima almenys igual a un valor prefixat, que s'anomena *distància de disseny del codi*. En general no són codis MDS; només es pot garantir que ho són si s'agafa l'extensió trivial $\mathbb{E} = \mathbb{F}$, i en aquest són una construcció alternativa de codis de Reed-Solomon que s'ha vist a la secció 6.5.

Extensions de cossos finits. En l'estudi dels codis BCH es fan servir els conceptes d'extensió de cossos i polinomi mínim. Convé repassar-los a les seccions 0.3 i 0.4.

Sigui $\mathbb{F} = \mathbb{F}_q$ un cos finit de q elements. En teoria de cossos quan $\mathbb{F} \subseteq \mathbb{E}$ és un subcos d'un cos $\mathbb{E}$ es diu que $\mathbb{E}$ és una *extensió* de $\mathbb{F}$ i se sol fer servir la notació $\mathbb{E}/\mathbb{F}$ per indicar-ho.

Les extensions finites de cossos finits s'obtenen de la manera següent: tots els cossos finits $\mathbb{E}$ que contenen $\mathbb{F}$ com a subcos són els que tenen cardinal q^e, una potència de q. Es poden construir com $\mathbb{E} = \mathbb{F}[\mathtt{z}]_{P(\mathtt{z})}$ amb $P(\mathtt{z}) \in \mathbb{F}[\mathtt{z}]$ un polinomi primer de grau e. N'hi ha un per a cada enter $e \geqslant 1$, que s'anomena grau de l'extensió $\mathbb{E}/\mathbb{F}$.

Per exemple, tots els cossos finits $\mathbb{F}_{2^e}$ de nombre d'elements potència de 2 (que són els cossos finits de característica 2) són tots extensions del cos binari $\mathbb{F}_2 = \{\mathtt{0}, \mathtt{1}\}$. Anàlogament, els cossos de característica un primer p són les extensions del cos $\mathbb{F}_p = \mathbb{Z}_p$.

Més en general, les extensions han de ser sempre entre cossos finits de la mateixa característica i requereixen que els exponents en les potències corresponents divideixin l'un a l'altre. Per exemple, $\mathbb{F}_{256}$, amb $256 = 2^8$, és una extensió de $\mathbb{F}_{16}$, $\mathbb{F}_4$ i $\mathbb{F}_2$, que corresponen

als divisors $4, 2$ i 1 de 8, però no és una extensió de $\mathbb{F}_8$, ja que $8 = 2^3$ i 3 no divideix 8, ni de cap altre cos diferent dels tres esmentats i d'ell mateix; el cos $\mathbb{F}_{512}$, amb $512 = 2^9$, només és extensió de $\mathbb{F}_8$, de $\mathbb{F}_2$ i d'ell mateix; el cos $\mathbb{F}_{2048}$, amb $2048 = 2^{11}$, només és extensió de $\mathbb{F}_2$ i d'ell mateix.

En els articles de Hocquenghem [22] i Bose i Ray-Chaudhuri [19] on s'introdueixen aquests codis per primera vegada només es considera el cas dels codis binaris, amb cos de coeficients $\mathbb{F} = \mathbb{F}_2$ i $\mathbb{E} = \mathbb{F}_{2^e}$ una extensió qualsevol. Més endavant la seva construcció es va generalitzar a extensions $\mathbb{E}/\mathbb{F}$ qualssevol.

Codis polinomials determinats per arrels. Sigui $\mathbb{E}/\mathbb{F}$ una extensió de cossos finits. Siguin $\alpha_1, \ldots, \alpha_\mu \in \mathbb{E}$ elements diferents del cos extensió. Donat un enter n es considera el codi lineal format pels polinomis de $\mathbb{F}[X]_n$ a coeficients en el cos base $\mathbb{F}$ de grau $< n$ que s'anul·len en tots els α_i:

$$\mathcal{C} = \big\{ \boldsymbol{u}(X) \in \mathbb{F}[X]_n : \boldsymbol{u}(\alpha_1) = \boldsymbol{u}(\alpha_2) = \cdots = \boldsymbol{u}(\alpha_\mu) = 0 \big\}. \tag{14}$$

Sigui $P_i(X)$ el polinomi minimal de cada element α_i. Aleshores $\boldsymbol{u}(\alpha_i) = 0 \Leftrightarrow P_i(X) \mid \boldsymbol{u}(X)$ i els polinomis de $\mathcal{C}$ són els divisibles pel polinomi

$$\boldsymbol{g}(X) = \operatorname{lcm} \big\{ P_i(X) : 1 \leqslant i \leqslant \mu \big\}.$$

Per tant $\mathcal{C}$ és el codi polinomial $\mathrm{Pol}(n, \boldsymbol{g})$ generat per aquest polinomi $\boldsymbol{g}(X)$.

Elements α_i diferents poden tenir el mateix polinomi mínim. El mínim comú múltiple $\boldsymbol{g}(X)$ és el producte de tots els $P_i(X)$ que siguin diferents. Cada $P_i(X)$ té grau un divisor de e. Per tant, la codimensió del codi $\mathcal{C}$, que és el grau $m = \deg \boldsymbol{g}$, és la suma dels graus dels P_i que siguin diferents, que com a màxim pot arribar a ser $e\mu$ en el cas que tots siguin diferents i que tots tinguin grau e.

Es considera la matriu $\widetilde{\mathbf{H}}$ següent, amb coeficients en el cos $\mathbb{E}$:

$$\widetilde{\mathbf{H}} = \begin{bmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdots & \alpha_1^{n-1} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \alpha_\mu & \alpha_\mu^2 & \cdots & \alpha_\mu^{n-1} \end{bmatrix} \in \mathrm{Mat}_{\mu \times n}(\mathbb{E}). \tag{15}$$

Donat un polinomi $\boldsymbol{u}(X) = \mathtt{u}_0 + \mathtt{u}_1 X + \cdots + \mathtt{u}_{n-1} X^{n-1} \in \mathbb{F}[X]_n$ es té

$$\boldsymbol{u}(X) \in \mathcal{C} \;\Leftrightarrow\; \boldsymbol{u}(\alpha_i) = \mathtt{u}_0 + \mathtt{u}_1 \alpha_i + \cdots + \mathtt{u}_{n-1} \alpha_i^{n-1} = 0 \;\forall i \;\Leftrightarrow\; \widetilde{\mathbf{H}} \cdot \mathbf{u} = \mathbf{0}.$$

La matriu $\widetilde{\mathbf{H}}$ es comporta com una matriu de control del codi $\mathcal{C}$ en el sentit que les paraules codi $\boldsymbol{c} \in \mathcal{C}$ es caracteritzen per la condició $\widetilde{\mathbf{H}} \cdot \mathbf{c} = \mathbf{0}$, però no és exactament una matriu de control, ja que té coeficients en un cos $\mathbb{E}$ que en general no és el cos $\mathbb{F}$ on es considera el codi. De fet, $\widetilde{\mathbf{H}}$ és la matriu d'una aplicació $\mathbb{E}$-lineal $\mathbb{E}^n \to \mathbb{E}^\mu$ que té com a nucli un $\mathbb{E}$-subespai vectorial de $\mathbb{E}^n$. El codi $\mathcal{C}$ és el subconjunt d'aquest nucli format pels vectors que tenen components en el subcos $\mathbb{F} \subseteq \mathbb{E}$, que és un $\mathbb{F}$-espai vectorial però no és tancat per la multiplicació per escalars de $\mathbb{E}$.

Tot i així, la condició de la proposició 5.28 que permet determinar la distància mínima del codi se segueix complint per a la matriu $\widetilde{\mathbf{H}}$: si $\boldsymbol{c} \in \mathcal{C}$ és una paraula codi de pes $\|\boldsymbol{c}\| = w$,

aleshores la igualtat $\widetilde{\mathbf{H}} \cdot \mathbf{c} = \mathbf{0}$ és una relació de dependència lineal entre w columnes de $\widetilde{\mathbf{H}}$, i per tant si $d-1$ columnes de $\widetilde{\mathbf{H}}$ són sempre independents aleshores $d(\mathcal{C}) \geqslant d$.

Partint de la matriu $\widetilde{\mathbf{H}}$ es pot calcular una matriu de control per al codi, de la manera següent: donada una equació lineal de la forma

$$\eta_1 X_1 + \eta_2 X_2 + \cdots + \eta_n X_n = 0, \qquad \eta_j \in \mathbb{E},$$

s'han de trobar només les solucions $(\mathtt{x}_1, \mathtt{x}_2, \ldots, \mathtt{x}_n)$ d'aquesta equació amb components $\mathtt{x}_j \in \mathbb{F}$, i no pas totes les solucions amb components $\mathtt{x}_j \in \mathbb{E}$. Per a cada $j = 1, \ldots, n$ sigui

$$\eta_j = \mathtt{h}_{0,j} + \mathtt{h}_{1,j}\mathtt{z} + \cdots + \mathtt{h}_{e-1,j}\mathtt{z}^{e-1} \in \mathbb{E}, \qquad \mathtt{h}_{i,j} \in \mathbb{F},$$

la representació de cada $\eta_j \in \widetilde{\mathbf{H}}$ com a polinomi en $\mathtt{z}$ de grau $< e$ amb coeficients en $\mathbb{F}$.

Substituint aquestes expressions per als η_j en l'equació lineal s'obté

$$\sum_{j=1}^{n} \eta_j X_j = \sum_{j=1}^{n} \left(\sum_{i=0}^{e-1} \mathtt{h}_{i,j}\mathtt{z}^i \right) X_j = \sum_{i=0}^{e-1} \left(\sum_{j=1}^{n} \mathtt{h}_{i,j} X_j \right) \mathtt{z}^i = 0.$$

Com que els $\mathtt{z}$ són una $\mathbb{F}$-base de $\mathbb{E}$, aquesta identitat equival a què tots els coeficients $\sum \mathtt{h}_{ij} X_j \in \mathbb{F}$ siguin zero, que correspon al sistema lineal homogeni de matriu:

$$\begin{bmatrix} \mathtt{h}_{0,1} & \mathtt{h}_{0,2} & \cdots & \mathtt{h}_{0,n} \\ \mathtt{h}_{1,1} & \mathtt{h}_{1,2} & \cdots & \mathtt{h}_{1,n} \\ \vdots & \vdots & & \vdots \\ \mathtt{h}_{e-1,1} & \mathtt{h}_{e-1,2} & \cdots & \mathtt{h}_{e-1,n} \end{bmatrix} \in \mathrm{Mat}_{e,n}(\mathbb{F}).$$

Fent això amb cadascuna de les files de la matriu $\widetilde{\mathbf{H}}$, que són equacions lineals amb coeficients en $\mathbb{E}$, s'obtenen les solucions de $\widetilde{\mathbf{H}} \cdot \mathbf{c} = \mathbf{0}$ amb $\boldsymbol{c} \in \mathbb{F}^n$ com les d'un sistema lineal $\mathbf{H} \cdot \mathbf{c} = \mathbf{0}$ per a una matriu $\mathbf{H}$ de $e\mu$ files i n columnes: cada equació amb coeficients en $\mathbb{E}$, que correspon a una fila de $\widetilde{\mathbf{H}}$, s'ha convertit en e equacions amb coeficients en $\mathbb{F}$, que corresponen a e files de la matriu $\mathbf{H}$.

La matriu $\mathbf{H}$ construïda d'aquesta manera té la propietat $\boldsymbol{c} \in \mathcal{C} \Leftrightarrow \mathbf{H} \cdot \mathbf{c} = \mathbf{0}$, però pot ser que contingui equacions redundants: que no sigui de rang màxim. De fet, la matriu $\widetilde{\mathbf{H}}$, que no conté equacions redundants si es consideren solucions a $\mathbb{E}^n$, pot contenir equacions redundants si es consideren solucions a $\mathbb{F}^n$: si els polinomis irreductibles de α_i i de α_j són iguals aleshores $\boldsymbol{u}(\alpha_i) = 0 \Leftrightarrow \boldsymbol{u}(\alpha_j) = 0$ per a tot polinomi $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$.

Per obtenir una matriu de control de $\mathcal{C}$ simplement s'eliminen totes les files de $\mathbf{H}$ corresponents a equacions redundants, de manera que quedi una matriu de rang màxim m. Com que $\mathbf{H}$ té $e\mu$ files aquesta construcció d'una matriu de control proporciona una nova justificació de la fita $m \leqslant e\mu$ per a la codimensió del codi.

Codis BCH. Els codis BCH són els que es construeixen d'aquesta manera agafant com a arrels α_i totes les potències d'un element del cos $\mathbb{E}$:

Definició 6.28 (Codis BCH). *Sigui $\beta \in \mathbb{E}$ un element d'ordre n. Sigui δ un enter amb $1 \leqslant \delta \leqslant n+1$. Es defineix el codi:*

$$\mathrm{BCH}(\beta, \delta) = \{\boldsymbol{c}(X) \in \mathbb{F}[X]_n : \boldsymbol{c}(\beta) = \boldsymbol{c}(\beta^2) = \cdots = \boldsymbol{c}(\beta^{\delta-1}) = 0\}.$$

El nombre δ en la definició s'anomena distància de disseny *o* distància prescrita *del codi.*

Recordi's que l'ordre d'un element $\beta \in \mathbb{E}$ és sempre un divisor de $q^e - 1$, i per tant la longitud dels codis BCH ha de ser un eneter amb aquesta propietat. Quan s'agafa β que sigui un element primitiu del cos i la longitud n que sigui igual al seu ordre $q^e - 1$ els codis BCH que s'obtenen se solen anomenar *codis BCH primitius*.

El fet d'haver agafar les arrels α_i d'aquesta manera permet assegurar que la distància de disseny δ és una fita inferior per a la distància mínima del codi:

Proposició 6.29. *El codi $\mathcal{C} = \mathrm{BCH}(\beta, \delta)$ és un codi cíclic de codimensió $m \leqslant e(\delta - 1)$ i distància mínima $d(\mathcal{C}) \geqslant \delta$.*

Prova: El codi és un codi determinat per les $\delta - 1$ arrels $\alpha_i = \beta^i \in \mathbb{E}$ per a $i = 1, \ldots, \delta - 1$. Com ja s'ha vist abans per a arrels diferents qualssevol, és un codi polinomial amb polinomi generador el producte dels polinomis mínims diferents de les potències β^i. Cadascun dels polinomis té grau $\leqslant e$ i per tant el grau del polinomi generador és $\leqslant e(\delta - 1)$.

Com que β i totes les seves potències satisfan $\beta^n = 1$ es dedueix que els polinomis mínims de totes les potències β^i divideixen el polinomi $X^n - 1$ i, per tant, el codi $\mathrm{BCH}(\beta, \delta)$ és cíclic.

Per veure la fita sobre la distància mínima es fan servir matrius de Vandermonde. La matriu (15) corresponent a aquestes α_i és

$$\widetilde{\mathbf{H}} = \begin{bmatrix} 1 & \beta & \beta^2 & \cdots & \beta^{n-1} \\ 1 & \beta^2 & \beta^4 & \cdots & \beta^{2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \beta^{\delta-1} & \beta^{(\delta-1)2} & \cdots & \beta^{(\delta-1)(n-1)} \end{bmatrix} \in \mathrm{Mat}_{(\delta-1)\times n}(\mathbb{E}).$$

Tota submatriu quadrada formada per $\delta - 1$ columnes és de la forma

$$\begin{bmatrix} \beta^{j_1} & \beta^{j_2} & \cdots & \beta^{j_{\delta-1}} \\ \beta^{2j_1} & \beta^{2j_2} & \cdots & \beta^{2j_{\delta-1}} \\ \vdots & \vdots & & \vdots \\ \beta^{(\delta-1)j_1} & \beta^{(\delta-1)j_2} & \cdots & \beta^{(\delta-1)j_{\delta-1}} \end{bmatrix}.$$

per a alguns índexs $j_1, \ldots, j_{\delta-1}$ amb $0 \leqslant j_1 < \cdots < j_{\delta-1} \leqslant n-1$. És una matriu de Vandermonde que, com que els β^{j_i} són tots no nuls i diferents entre ells, té determinant $\neq 0$. Per tant $\delta - 1$ columnes de $\widetilde{\mathbf{H}}$ són sempre independents. Això vol dir que el codi $\mathcal{C}$ no pot tenir cap paraula $\boldsymbol{c} \neq \mathbf{0}$ de pes $1 \leqslant \|\boldsymbol{c}\| \leqslant \delta - 1$, ja que la identitat $\widetilde{\mathbf{H}} \cdot \mathbf{c} = \mathbf{0}$ donaria una relació de dependència lineal entre $\delta - 1$ columnes de la matriu. Per tant $d(\mathcal{C}) \geqslant \delta$. □

Codis BCH binaris. La primera vegada que es van estudiar els codis BCH va ser en el cas binari, amb cos base $\mathbb{F} = \{\mathtt{0}, \mathtt{1}\}$. En cossos $\mathbb{E} = \mathbb{F}_{2^e}$ de característica 2 se satisfà la igualtat $(\alpha + \beta)^2 = x^2 + y^2$. Per a tot element $\mathtt{u} \in \mathbb{F} = \{\mathtt{0}, \mathtt{1}\}$ es compleix que $\mathtt{u}^2 = \mathtt{u}$. D'aquests dos fets es dedueix que per a tot polinomi $P(X) \in \mathbb{F}[X]$ i tot element $\alpha \in \mathbb{E}$ es compleix $P(\alpha^2) = P(\alpha)^2$ i, per tant, $P(\alpha^2) = 0 \Leftrightarrow P(\alpha) = 0$.

Per tant, el codi BCH binaris amb distància de disseny parell $2t$ és el mateix que el de distància de disseny senar $\delta = 2t + 1$. Tots dos venen donats per:

$$\mathrm{BCH}(\beta, 2t) = \mathrm{BCH}(\beta, 2t+1) = \left\{\boldsymbol{u}(X) \in \mathbb{F}[X]_n : \boldsymbol{u}(\beta) = \boldsymbol{u}(\beta^3) = \cdots = \boldsymbol{u}(\beta^{2t-1})\right\}.$$

En efecte, si el polinomi $\boldsymbol{u}(X)$ s'anul·la en aquestes potències senars, aleshores també s'anul·la en totes les potències parells $\beta^2, \ldots, \beta^{2t}$, ja que cadascuna és el quadrat d'una de les anteriors.

Per tant en aquests codis es pot assegurar que la codimensió és $m \leqslant et$ i la distància mínima $d(\mathcal{C}) \geqslant 2t + 1$.

Exemple: codis BCH binaris de longitud 15. Vegeu [3, Exemple 15.3.2] o els exemples de la *wikipèdia*. Es considera el cos $\mathbb{E} = \mathbb{F}_{16}$ construït com $\mathbb{F}[\mathtt{z}]_{P(\mathtt{z})}$ amb el polinomi primitiu $1 + \mathtt{z} + \mathtt{z}^4$. Sigui $\beta = [\mathtt{z}]$, que és un element primitiu: té ordre 15, amb polinomi minim $P_\beta(X) = 1 + X + X^4$. A la taula següent es donen tots els elements $\alpha \in \mathbb{E}$ en termes de potències de β, els seus coeficients com a polinomis binaris $\alpha = \mathtt{a}_0 + \mathtt{a}_1\mathtt{z} + \mathtt{a}_2\mathtt{z}^2 + \mathtt{a}_3\mathtt{z}^3$, i el seu polinomi mínim $P_\alpha(X)$ sobre el cos $\mathbb{F}$ de dos elements:

α	$\mathtt{a}_0\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3$	$P_\alpha(X)$	α	$\mathtt{a}_0\mathtt{a}_1\mathtt{a}_2\mathtt{a}_3$	$P_\alpha(X)$
0	0000	X	β^8	1010	$1 + X + X^4$
β	0100	$1 + X + X^4$	β^9	0101	$1 + X + X^2 + X^3 + X^4$
β^2	0010	$1 + X + X^4$	β^{10}	1110	$1 + X + X^2$
β^3	0001	$1 + X + X^2 + X^3 + X^4$	β^{11}	0111	$1 + X^3 + X^4$
β^4	1100	$1 + X + X^4$	β^{12}	1111	$1 + X + X^2 + X^3 + X^4$
β^5	0110	$1 + X + X^2$	β^{13}	1011	$1 + X^3 + X^4$
β^6	0011	$1 + X + X^2 + X^3 + X^4$	β^{14}	1001	$1 + X^3 + X^4$
β^7	1101	$1 + X^3 + X^4$	β^{15}	1000	$1 + X$

Es discuteixen a continuació els codis $\mathrm{BCH}(15, \delta)$ per a distàncies de disseny $1 \leqslant \delta \leqslant 15$.

- $\delta = 1$. El codi $\mathrm{BCH}(15, 1)$ és el codi amb polinomi generador $\boldsymbol{g}(X) = \mathtt{1}$ constant. És el codi total, de tipus $[15, 15, 1]_2$.
- $\delta = 2, 3$. Tots dos codis $\mathrm{BCH}(15, 2)$ i $\mathrm{BCH}(15, 3)$ són el mateix: el codi de tipus $[15, 11, 3]_2$ amb polinomi generador

  $$\boldsymbol{g}(X) = P_\beta(X) = 1 + X + X^4.$$

  Aquest codi és el codi el codi de Hamming $\mathrm{Ham}_2(3)$.
- $\delta = 4, 5$. Tots dos codis $\mathrm{BCH}(15, 4)$ i $\mathrm{BCH}(15, 5)$ són el mateix: el codi de tipus $[15, 7, 5]_2$ amb polinomi generador

  $$\begin{aligned}\boldsymbol{g}(X) &= P_\beta(X)P_{\beta^3}(X) = (1 + X + X^4)(1 + X + X^2 + X^3 + X^4)\\ &= 1 + X^4 + X^6 + X^7 + X^8.\end{aligned}$$

- $\delta = 6, 7$. Tots dos codis BCH$(15, 6)$ i BCH$(15, 7)$ són el mateix: el codi de tipus $[15, 5, 7]_2$ amb polinomi generador

$$\begin{aligned}\boldsymbol{g}(X) &= P_\beta(X)P_{\beta^3}(X)P_{\beta^5}(X)\\ &= (1+X+X^4)(1+X+X^2+X^3+X^4)(1+X+X^2)\\ &= 1+X+X^2+X^4+X^5+X^8+X^{10}.\end{aligned}$$

- $8 \leqslant \delta \leqslant 15$. Tots aquests codis BCH$(\beta, \delta)$ són el mateix codi: el codi de tipus $[15, 1, 14]_2$ generat pel polinomi

$$\begin{aligned}\boldsymbol{g}(X) &= P_\beta(X)P_{\beta^3}(X)P_{\beta^5}(X)P_{\beta^7}(X)\\ &= (1+X+X^4)(1+X+X^2+X^3+X^4)(1+X+X^2)(1+X^3+X^4)\\ &= \textstyle\sum_{i=0}^{14} X^i.\end{aligned}$$

  Aquest codi és el codi de repetició $\mathrm{Rep}_2(15)$.

Problemes

6.14. Digueu quins subcossos conté cadascun dels cossos $\mathbb{F}_{256}$, $\mathbb{F}_{512}$, $\mathbb{F}_{1024}$ i $\mathbb{F}_{2048}$.

Entre els cossos amb nombre d'elements $q = 3, 8, 27, 32, 81, 64, 125, 15625, 19683, 65536$ digueu quins són extensions d'un altre.

6.15. *Generalització amb múltiples de potències de β.* Comproveu que si en la definició dels codis BCH es consideren els polinomis $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$ que s'anul·len en $\delta - 1$ elements diferents de la forma

$$\alpha\beta, \alpha\beta^2, \ldots, \alpha\beta^{\delta-1},$$

on $\alpha \in \mathbb{E}$ és un element no nul qualsevol, s'obtenen codis amb les mateixes propietats que els codis BCH(n, δ) definits a 6.28, que corresponen a agafar $\alpha = 1$.

Els codis amb $\alpha = 1$ de vegades s'anomenen codis BCH *en sentit estricte*.

6.16. Considereu els codis BCH de longitud 15 de l'exemple de la pag. 233. Per a cadascun d'ells:

1. calculeu la matriu de control $\widetilde{\mathbf{H}}$ sobre el cos $\mathbb{E}$ i digueu quin és el seu rang;
2. calculeu la matriu $\mathbf{H}$ sobre el cos $\mathbb{F}$ que li correspon i elimineu les files redundants obtenint així una matriu de control del codi;
3. calculeu el polinomi de control del codi, la matriu de control corresponent, i compareu-la amb la matriu obtinguda en l'apartat anterior.

6.7 Descodificació de codis BCH

Per als codis BCH es disposa d'un ampli ventall d'algorismes de correcció d'errors eficients. Tots ells fan servir tècniques d'àlgebra lineal i aritmètica de polinomis sobre el cos $\mathbb{E}$. i corregeixen fins a $\lfloor \frac{\delta-1}{2} \rfloor$ errors, on δ és la distància de disseny del codi. Quan la distància de disseny és inferior a la distància mínima, o sigui quan $\delta < d = d(\mathcal{C})$, no poden corregir els errors $\boldsymbol{e}$ de pes $\lfloor \frac{\delta-1}{2} \rfloor < \|\boldsymbol{e}\| \leqslant \tau = \lfloor \frac{d-1}{2} \rfloor$, tot i que aquests errors estiguin dins de la capacitat correctora.

Com que els algorismes funcionen també en codis escurçats en les seves últimes posicions, que corresponen a agafar polinomis de longituds menor que $\mathrm{ord}(\beta)$, en tota aquesta secció es treballarà amb codis BCH construïts de la manera següent:

- el cos base és $\mathbb{F} = \mathbb{F}_q$, el cos de q elements;
- $\mathbb{E}$ és l'extensió de $\mathbb{F}$ que té q^e elements;
- $\beta \in \mathbb{E}^*$ és un element no nul;
- la longitud n del codi és un enter $n \leqslant \mathrm{ord}(\beta)$ ($\mathrm{ord}(\beta)$ és un divisor de $q^e - 1$);
- la distància de disseny δ satisfà $\delta \leqslant n + 1 \Leftrightarrow \delta - 1 \leqslant n$.

El codi $\mathcal{C} = \mathrm{BCH}(\beta, \delta)$ està format pels polinomis de $\mathbb{F}[X]_n$ que s'anul·len en les $\delta - 1$ primeres potències de β:

$$\mathcal{C} = \big\{ \boldsymbol{u}(X) \in \mathbb{F}[X]_n : \boldsymbol{u}(\beta) = \boldsymbol{u}(\beta^2) = \cdots = \boldsymbol{u}(\beta^{\delta-1}) \big\}.$$

És el codi polinomial que té com a polinomi generador el producte dels polinomis mínims diferents de les potències β^i per a $i = 0, \ldots, \delta - 1$.

Síndrome sobre $\mathbb{E}$. Es treballa amb l'aplicació lineal de síndrome $\mathrm{syn}\colon \mathbb{E}[X]_n \to \mathbb{E}^{\delta-1}$ definida per a les paraules (polinomis) amb coeficients en el cos $\mathbb{E}$ i que pren com a valors vectors amb coeficients en aquest mateix cos:

$$\mathrm{syn}\big(\widetilde{\boldsymbol{u}}(X)\big) = \big(\widetilde{\boldsymbol{u}}(\beta), \widetilde{\boldsymbol{u}}(\beta^2), \ldots, \widetilde{\boldsymbol{u}}(\beta^{\delta-1})\big), \quad \widetilde{\boldsymbol{u}}(X) \in \mathbb{E}[X]_n.$$

La matriu d'aquesta aplicació $\mathbb{E}$-lineal és la matriu

$$\widetilde{\mathbf{H}} = \begin{bmatrix} 1 & \beta & \beta^2 & \cdots & \beta^{n-1} \\ 1 & \beta^2 & \beta^4 & \cdots & \beta^{2(n-1)} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & \beta^{\delta-1} & \beta^{(\delta-1)2} & \cdots & \beta^{(\delta-1)(n-1)} \end{bmatrix} \in \mathrm{Mat}_{(\delta-1)\times n}(\mathbb{E}).$$

De vegades convé veure aquesta síndrome com un polinomi a través de la identificació $\mathbb{E}^{\delta-1} \approx \mathbb{E}[X]_{\delta-1}$. És a dir, la síndrome és

$$\mathrm{syn}\big(\widetilde{\boldsymbol{u}}(X)\big) = \sum_{i=1}^{\delta-1} \widetilde{\boldsymbol{u}}(\beta^i) X^{i-1} \in \mathbb{E}[X]_{\delta-1}.$$

El codi $\mathcal{C}$ és el subconjunt de $\mathbb{E}[X]_n$ format pels polinomis amb síndrome zero i que tenen coeficients en el cos $\mathbb{F}$. És a dir, la intersecció $\mathcal{C} = \ker(\text{syn}) \cap \mathbb{F}[X]_n$.

Observi's que $\mathcal{C}$ és un subespai vectorial de l'$\mathbb{F}$-espai vectorial $\mathbb{F}[X]_n$, però no és un subespai vectorial de l'$\mathbb{E}$-espai vectorial $\mathbb{E}[X]_n$, ja que no té perquè ser tancat pel producte per escalars d'aquest cos.

Per a cada polinomi $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$ sigui $\boldsymbol{s}(X) = \text{syn}\big(\boldsymbol{u}(X)\big) = \sum_{i=1}^{\delta-1} \boldsymbol{u}(\beta^i)X^{i-1}$ el seu polinomi síndrome. Tot i que $\boldsymbol{u}(X)$ té coeficients en $\mathbb{F}$ el seu polinomi síndrome $\boldsymbol{s}(X)$ té en general coeficients en $\mathbb{E}$.

Com que syn és $\mathbb{E}$-lineal, en particular també és $\mathbb{F}$-lineal. Identificant els polinomis $\boldsymbol{u}(X)$ i $\boldsymbol{s}(X)$ amb els vectors $\boldsymbol{u} = (\mathtt{u}_0, \ldots, \mathtt{u}_{n-1})$ i $\boldsymbol{s} = (\mathtt{s}_1, \ldots, \mathtt{s}_{\delta-1})$ que tenen per components els seus coeficients, la síndrome és simplement el producte matricial $\boldsymbol{s} = \widetilde{\mathbf{H}} \cdot \boldsymbol{u}$.

Polinomis localitzador i avaluador d'errors. Es fa servir el codi $\mathcal{C}$ per transmetre informació a través d'un canal de comunicacions.

Sigui $\boldsymbol{c}(X) = \sum_{i=1}^{n-1} \mathtt{c}_i X^i \in \mathcal{C}$ la paraula codi enviada. Sigui $\boldsymbol{u}(X) = \sum_{i=1}^{n-1} \mathtt{u}_i X^i \in \mathbb{F}[X]_n$ la paraula rebuda i sigui $\boldsymbol{e}(X) = \sum_{i=1}^{n-1} \mathtt{e}_i X^i = \boldsymbol{u}(X) - \boldsymbol{c}(X) \in \mathbb{F}[X]_n$ la paraula d'error corresponent. Sigui $w = \|\boldsymbol{e}(X)\| = \#\{i : \mathtt{e}_i \neq 0\}$ el seu pes de Hamming, que és el nombre d'errors de transmissió.

Com passa sempre per a codis lineals, la linealitat de l'aplicació syn permet assegurar que la síndrome de la paraula rebuda és la mateixa que la de la paraula d'error:

$$\boldsymbol{s} = \text{syn}\big(\boldsymbol{u}(X)\big) = \text{syn}\big(\boldsymbol{e}(X)\big).$$

Definició 6.30 (Polinomis localitzador i avaluador)**.** *Donada una paraula d'error $\boldsymbol{e}(X)$ de pes $\|\boldsymbol{e}(X)\| = w$ es defineixen el seu* polinomi localitzador d'errors Λ *com*

$$\Lambda(X) = \Lambda_{\boldsymbol{e}}(X) = \prod_{\mathtt{e}_i \neq 0} (1 - \beta^i X) = 1 + \lambda_1 X + \cdots + \lambda_w X^w,$$

i el seu polinomi avaluador d'errors Ω *com*

$$\Omega(X) = \Omega_{\boldsymbol{e}}(X) = \sum_{\mathtt{e}_i \neq 0} \mathtt{e}_i \beta^i \Lambda_i(X), \qquad \Lambda_i(X) = \frac{\Lambda(X)}{1 - \beta^i X} = \prod_{\substack{j \neq i \\ \mathtt{e}_j \neq 0}} (1 - \beta^j X).$$

El polinomi localitzador té grau $\deg \Lambda = w$ igual al nombre d'errors: és el pes de la paraula d'error $\boldsymbol{e}(X)$. Com que $n \geqslant \text{ord}(\beta)$ les potències $\beta^0, \beta, \cdots, \beta^{n-1}$ són totes diferents. Per tant les arrels del polinomi localitzador són les potències β^{-i} per als índexs i amb $\mathtt{e}_i \neq 0$, que són els que corresponen als errors de transmissió. Coneixent aquest polinomi es poden saber les posicions dels errors: hi ha un error en la posició i-èsima si, i només si, $\Lambda(\beta^{-i}) = 0$.

Com que tots els polinomis $\Lambda_i(X)$ tenen grau $w-1$ (o $-\infty$ si $w = 0$) el polinomi avaluador té grau $\deg \Omega \leqslant \deg \Lambda - 1 = w - 1$. Coneixent tots dos polinomis Λ i Ω es poden calcular les magnituds dels errors amb la fórmula de la

Proposició 6.31. *Siguin $\Lambda(X)$ i $\Omega(X)$ els polinomis localitzador i avaluador corresponents a una paraula d'error $\boldsymbol{e}(X) = \sum_{i=0}^{n-1} \mathsf{e}_i X^i \in \mathbb{F}[X]_n$. Aleshores els coeficients $\mathsf{e}_r \neq 0$ són:*

$$\Lambda(\beta^{-r}) = 0 \quad \Rightarrow \quad \mathsf{e}_r = -\frac{\Omega(\beta^{-r})}{\Lambda'(\beta^{-r})}, \qquad r = 0, \ldots, n-1.$$

Prova: Aplicant la fórmula de derivar productes, es calcula la derivada del polinomi localitzador $\Lambda(X) = \prod_{\mathsf{e}_i \neq 0}(1 - \beta^i X)\Lambda_i(X)$ i s'obté:

$$\Lambda'(X) = \sum_{\mathsf{e}_i \neq 0} -\beta^i \Lambda_i(X).$$

Sigui r un índex amb $\Lambda(\beta^{-r}) = 0 \Leftrightarrow \mathsf{e}_r \neq 0$. Per a cada índex i amb $\mathsf{e}_i \neq 0$ es té:

$$\Lambda_i(\beta^{-r}) = \begin{cases} \prod_{\substack{j \neq i \\ \mathsf{e}_j \neq 0}} (1 - \beta^{j-r}) \neq 0, & i = r, \\ 0, & i \neq r. \end{cases}$$

Per tant

$$\Lambda'(\beta^{-r}) = -\beta^r \Lambda_r(\beta^{-r}),$$
$$\Omega(\beta^{-r}) = \mathsf{e}_r \beta^r \Lambda(\beta^{-r}),$$

i dividint s'obté l'expressió per a e_r de l'enunciat. □

Així, per tal de corregir els errors en una paraula rebuda n'hi ha prou a calcular els polinomis avaluador i localitzador corresponents. Els algorismes eficients que es veuran més endavant fan precisament això: calculen aquests dos polinomis partint de la síndrome de la paraula rebuda.

Els polinomis localitzador i avaluador d'una paraula d'error satisfan la congruència fonamental següent:

Teorema 6.32 (Identitat fonamental)**.** *Siguin $\Lambda(X)$ i $\Omega(X)$ els polinomis localitzador i avaluador d'una paraula error $\boldsymbol{e}(X)$ de síndrome $\boldsymbol{s}(X)$. Suposi's que $\|\boldsymbol{e}(X)\| \leqslant \lfloor \frac{\delta-1}{2} \rfloor$. Aleshores se satisfà la congruència següent:*

$$\boldsymbol{s}(X)\Lambda(X) \equiv \Omega(X) \pmod{X^{\delta-1}}.$$

Prova: A l'anell de sèries de potències formals $\mathbb{F}[\![X]\!]$ es té $(1 - \beta^i X)^{-1} = \sum_{t=0}^{\infty} \beta^{it} X^t$ per a cada índex i. Aleshores, tenint en compte que $\Lambda_i(X) = \Lambda(X)(1 - \beta^i X)^{-1}$ es té:

$$\Omega(X) = \sum_{\mathsf{e}_i \neq 0} \mathsf{e}_i \beta^i \Lambda_i(X) = \Lambda(X) \sum_{\mathsf{e}_i \neq 0} \mathsf{e}_i \beta^i (1 - \beta^{i_j} X)^{-1} = \Lambda(X) \sum_{\mathsf{e}_i \neq 0} \mathsf{e}_i \beta^i \sum_{t=0}^{\infty} \beta^{it} X^t$$
$$= \Lambda(X) \sum_{t=0}^{\infty} \left(\sum_{\mathsf{e}_i \neq 0} \mathsf{e}_i \beta^{i(1+t)} \right) X^t = \Lambda(X) \sum_{t=0}^{\infty} \left(\sum_{i=0}^{\delta-1} \mathsf{e}_i \big(\beta^{(1+t)}\big)^i \right) X^t = \Lambda(X) \sum_{t=0}^{\infty} \boldsymbol{e}(\beta^{1+t}) X^t.$$

Els primers $\delta - 2$ coeficients de la sèrie de la dreta són $\boldsymbol{e}(\delta)$, …, $\boldsymbol{e}(\beta^{\delta-1})$: els coeficients del polinomi $\boldsymbol{s}(X)$. Per tant, totes dues expressions coincideixen en tots els coeficients de grau $\leqslant \delta - 2$, el qual és equivalent a la congruència mòdul la potència $X^{\delta-1}$. □

La identitat fonamental permet calcular el polinomi avaluador a partir del polinomi síndrome i el polinomi localitzador: es calcula el producte $\boldsymbol{s}(X)\Lambda(X)$ i s'eliminen tots els monomis de grau $\geqslant \delta - 1$; el polinomi que queda és el polinomi localitzador sempre que aquest polinomi sigui de grau $\leqslant \delta - 2$.

Recíprocament, donats dos polinomis Λ i Ω dels graus adequats que satisfacin la identitat fonamental la fórmula de la proposició 6.31 permet recuperar la paraula d'error:

Lema 6.33. *Donat un polinomi $\boldsymbol{s}(X) \in \mathbb{E}[X]_{\delta-1}$, siguin $\Lambda(X), \Omega(X) \in \mathbb{E}[X]$ polinomis que satisfacin les condicions següents:*

- $\deg \Omega < \deg \Lambda = w \leqslant \lfloor \frac{\delta-1}{2} \rfloor$;
- $\gcd(\Lambda, \Omega) = 1$;
- $\Lambda(X)$ *descompon completament a* $\mathbb{E}[X]$ *en factors de grau* 1 *i* $\Lambda(0) \neq 0$;
- $\boldsymbol{s}(X)\Lambda(X) \equiv \Omega(X) \pmod{X^{\delta-1}}$.

Aleshores existeix una paraula $\boldsymbol{e}(X) \in \mathbb{F}[X]_n$ de pes $\|\boldsymbol{e}(X)\| = w$ amb $\operatorname{syn}\big(\boldsymbol{e}(X)\big) = \boldsymbol{s}(X)$.

Prova: Multiplicant Λ i Ω per una constant no nul·la es pot suposar que $\Lambda(1) = 1$.

Es defineix el polinomi $\boldsymbol{e}(X) = \sum_{i=0}^{n-1} \mathtt{e}_i X^i$ posant-li coeficients $\mathtt{e}_i = 0$ si $\Lambda(\beta^{-i}) \neq 0$ i els coeficients $\mathtt{e}_r$ de la proposició 6.31 quan $\Lambda(\beta^{-r}) = 0$. Per construcció aquesta paraula té pes $w = \deg \Lambda$. A més els seus polinomis localitzador i avaluador són $\Lambda_{\boldsymbol{e}}(X) = \Lambda(X)$ i $\Omega_{\boldsymbol{e}}(X) = \Omega(X)$. La identitat fonamental assegura que $\operatorname{syn}\big(\boldsymbol{e}(X)\big)\Lambda(X) \equiv \boldsymbol{s}(X)\Lambda(X) \pmod{X^{\delta-1}}$. Es dedueix que la diferència $\operatorname{syn}\big(\boldsymbol{e}(X)\big) - \boldsymbol{s}(X)$, multiplicada pel polinomi $\Lambda(X)$, té el zero com a arrel de multiplicitat almenys $\delta - 1$. Com que $\Lambda(X) \neq 0$ la diferència $\operatorname{syn}\big(\boldsymbol{e}(X)\big) - \boldsymbol{s}(X)$ és divisible per $X^{\delta-1}$ i, tenint en compte els graus, això equival a què és el polinomi zero i per tant $\operatorname{syn}\big(\boldsymbol{e}(X)\big) = \boldsymbol{s}(X)$. □

Descodificació basada en l'algorisme d'Euclides estès. Es dona a continuació un primer mètode de descodificació eficient, basat en l'algorisme d'Euclides estès, que de vegades s'anomena *algorisme de Sugiyama*, un dels autors de l'article on es va proposar.

Aquest algorisme calcula simultàniament els polinomis Λ i Ω:

Algorisme 6.34. *Correcció d'errors en codis BCH usant l'algorisme d'Euclides estès.*

Entrada: *El polinomi de síndromes $\boldsymbol{s}(X) \in \mathbb{E}[X]_{\delta-1}$ d'una paraula $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$.*

Inicialització: *S'inicialitzen dos polinomis $\boldsymbol{r}_0(X) = X^m$ i $\boldsymbol{r}_1(X) = \boldsymbol{s}(X)$ i uns altres dos polinomis $\boldsymbol{b}_0(X) = \mathtt{0}$ i $\boldsymbol{b}_1(X) = \mathtt{1}$. S'inicialitza un comptador $i = 1$.*

Repetir: *Mentre* $\deg \boldsymbol{r}_i \geqslant \lfloor \frac{\delta-1}{2} \rfloor$,

Càlcul de $\boldsymbol{r}_{i+1}(X)$ i $\boldsymbol{b}_{i+1}(X)$: *es calculen dos nous polinomis posant*

$$\begin{aligned} \boldsymbol{r}_{i-1}(X) &= \boldsymbol{r}_i(X)\boldsymbol{q}(X) + \boldsymbol{r}_{i+1}(X) \qquad \textit{(divisió euclidiana)} \\ \boldsymbol{b}_{i+1}(X) &= \boldsymbol{b}_{i-1}(X) - \boldsymbol{q}(X)\boldsymbol{b}_i(X). \end{aligned}$$

Comptador: *S'incrementa el comptador $i \leftarrow i + 1$.*

SORTIDA: *Es retornen els dos últims polinomis* $\boldsymbol{r}_i(X)$ *i* $\boldsymbol{b}_i(X)$ *calculats.*

Teorema 6.35. *Siguin* $\Lambda(X) = \boldsymbol{b}_i(X)$ *i* $\Omega(X) = \boldsymbol{r}_i(X)$ *els polinomis que retorna l'algorisme anterior amb entrada la síndrome d'una paraula* $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$*. Aleshores* $d\big(\boldsymbol{u}(X), \mathcal{C}\big) \leqslant \tau$ *si, i només si, aquests dos polinomis satisfan les condicions del lema 6.33.*

En aquest cas els polinomis són el localitzador i avaluador de la paraula d'error $\boldsymbol{e}(X)$ *de pes* $\|\boldsymbol{e}(X)\| = \deg \Lambda$ *tal que* $\boldsymbol{u}(X) = \boldsymbol{c}(X) + \boldsymbol{e}(X)$ *amb* $\boldsymbol{c}(X) \in \mathcal{C}$*.*

Descodificació usant recurrències lineals. Diversos algorismes es descodificació usen la teoria de les successions que satisfan una *recurrència lineal*, basant-se en el fet que el vector de síndromes d'una paraula d'error té aquesta propietat, on el polinomi característic de la recurrència lineal és el polinomi localitzador d'errors:

Proposició 6.36. *Les components del vector de síndromes* $\boldsymbol{s} = \operatorname{syn}\big(\boldsymbol{e}(X)\big) = (\mathtt{s}_1, \mathtt{s}_2, \ldots, \mathtt{s}_m)$ *d'una paraula d'error* $\boldsymbol{e}(X)$ *de pes* $w = \|\boldsymbol{e}(X)\|$ *satisfan la recurrència lineal:*

$$\mathtt{s}_i = -(\lambda_1 \mathtt{s}_{i-1} + \lambda_2 \mathtt{s}_{i-2} + \cdots + \lambda_w \mathtt{s}_{i-w}) = -\sum_{r=1}^{w} \lambda_r \mathtt{s}_{r-i}, \qquad i = w+1, w+2, \ldots, \delta-1,$$

de polinomi característic el polinomi localitzador $\Lambda_{\boldsymbol{e}}(X) = 1 + \sum_{r=1}^{w} \lambda_r X^r$*.*

PROVA: Les components del vector de síndromes s'obtenen avaluant el polinomi d'error en potències de β: són els valors $\mathtt{s}_i = \boldsymbol{e}(\beta^i) = \sum_{j=0}^{n-1} \mathtt{e}_j \beta^{ij}$ per a $i = 1, \ldots, \delta-1$.

Sigui $\boldsymbol{e}(X) = \sum_{j=0}^{n-1} \mathtt{e}_j X^j$. Per a cada $i = 1, \ldots, \delta-1$ i cada $j = 0, \ldots, n-1$, en multiplicar el polinomi localitzador per $\mathtt{e}_j \beta^{ij}$ i substituir la variable per β^{-j} s'obté

$$\mathtt{e}_j \beta^{ij} \Lambda_{\boldsymbol{e}}(\beta^{-j}) = \sum_{r=0}^{w} \mathtt{e}_j \beta^{ij} \lambda_r \beta^{-jr} = \mathtt{e}_j \beta^{ij} + \lambda_1 \mathtt{e}_j \beta^{(i-1)j} + \cdots + \lambda_w \mathtt{e}_j \beta^{(i-w)j} = 0.$$

En efecte, si $\mathtt{e}_j = 0$ el producte val clarament zero, i si $\mathtt{e}_j \neq 0$, per definició de polinomi localitzador, la potència β^{-j} és una arrel de $\Lambda_{\boldsymbol{e}}(X)$ i per tant el producte també és zero.

Sumant aquestes expressions per a tots els índexs j s'obté

$$\sum_{j=0}^{n-1} \mathtt{e}_j \beta^{ij} + \lambda_1 \sum_{j=0}^{n-1} \mathtt{e}_j \beta^{(i-1)j} + \cdots + \lambda_w \sum_{j=0}^{n-1} \mathtt{e}_j \beta^{(i-w)j} = \mathtt{s}_i + \lambda_1 \mathtt{s}_{i-1} + \cdots + \lambda_w \mathtt{s}_{i-w} = 0,$$

que és la relació de recurrència de l'enunciat.

Observi's que en realitat aquesta relació de recurrència la satisfan tots els elements de la successió $\mathtt{s}_i = \boldsymbol{e}(\beta^i)$ per a tots els índexs $i \in \mathbb{Z}$ amb $i \geqslant w+1$. □

En notació matricial aquesta recurrència lineal s'escriu com

$$\begin{bmatrix} \mathtt{s}_1 & \mathtt{s}_2 & \cdots & \mathtt{s}_w \\ \mathtt{s}_2 & \mathtt{s}_3 & \cdots & \mathtt{s}_{w+1} \\ \vdots & \vdots & & \vdots \\ \mathtt{s}_w & \mathtt{s}_{w+1} & \cdots & \mathtt{s}_{2w-1} \end{bmatrix} \begin{bmatrix} \lambda_w \\ \lambda_{w-1} \\ \vdots \\ \lambda_1 \end{bmatrix} = - \begin{bmatrix} \mathtt{s}_{w+1} \\ \mathtt{s}_{w+2} \\ \vdots \\ \mathtt{s}_{2w} \end{bmatrix}. \tag{16}$$

Un primer mètode basat en la recurrència de la proposició 6.36 es coneix com a *algorisme de Peterson-Gorenstein-Zierler* i consisteix en el següent:

Algorisme 6.37. *Correcció d'errors en codis BCH invertint la matriu de síndromes.*

ENTRADA: *El vector de síndromes* $\boldsymbol{s} = (\mathtt{s}_1, \ldots, \mathtt{s}_{\delta-1}) \in \mathbb{E}^{\delta-1}$ *d'una paraula* $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$.

INICIALITZACIÓ: *S'inicialitza un comptador* w*, que representa el pes de la paraula d'error, amb el valor màxim* $w = \lfloor \frac{\delta-1}{2} \rfloor$ *dels errors corregibles.*

REPETIR: *Mentre* $w \geqslant 0$*,*

ERROR DE PES w?: *es calcula el determinant de la matriu* $\mathbf{S} \in \mathrm{Mat}_w(\mathbb{E})$ *de* (16)*.*

SI: $\det \mathbf{S} \neq 0$ *es resol el sistema* (16) *i s'acaba l'algorisme retornant el polinomi localitzador* $\Lambda(X) = 1 + \sum_{i=1}^{w} \lambda_i X^i$ *corresponent a la solució.*

COMPTADOR: *Es decrementa el comptador* $w \leftarrow w - 1$*.*

SORTIDA: *Si el comptador arriba a zero s'acaba l'algorisme declarant un error no corregible.*

Un altre mètode, basat també en la recurrència que satisfà el vector de síndromes, fa servir l'*algorisme de Berlekamp-Massey* que serveix en general per calcular el polinomi característic d'una successió recurrent lineal. De fet, la motivació original d'aquest algorisme, que té múltiples aplicacions en àmbits diversos, va ser justament el problema de descodificar codis BCH. L'algorisme és el següent:

Algorisme 6.38. *Correcció d'errors en codis BCH usant Berlekamp-Massey.*

ENTRADA: *El vector de síndromes* $\boldsymbol{s} = (\mathtt{s}_1, \ldots, \mathtt{s}_{\delta-1}) \in \mathbb{E}^{\delta-1}$ *d'una paraula* $\boldsymbol{u}(X) \in \mathbb{F}[X]_n$.

INICIALITZACIÓ: *S'inicialitza un comptador amb el valor* $w = 0$*. S'inicialitzen dues* $\Lambda(X)$ *i* $B(X)$ *que contindran polinomis amb els polinomis constants* $\Lambda(X) = B(X) = 1$*. S'inicialitza una variable amb el valor* $L = 0$*.*

REPETIR: *Mentre sigui* $w \leqslant \lfloor \frac{\delta-1}{2} \rfloor$*,*

CÀLCUL DE Δ: *Es calcula*

$$\Delta = \sum_{i=0}^{w-1} \lambda_i^{(w-1)} \mathtt{s}_{w-i}, \quad \textit{on} \quad \Lambda(X) = \sum_{i=0}^{w} \lambda_i X^i$$

ACTUALITZACIÓ DE $\Lambda(X)$, $B(X)$ I L: *S'actualitzen*

$$\Lambda(X) \leftarrow \Lambda(X) - \Delta X B(X),$$

$$B(X) \leftarrow \begin{cases} \Delta^{-1}\Lambda(X), & \Delta \neq 0 \quad i \quad 2L \leqslant w-1, \\ XB(X), & \textit{altrament}, \end{cases}$$

$$L \leftarrow \begin{cases} w - L, & \Delta \neq 0 \quad i \quad 2L \leqslant w-1, \\ L, & \textit{altrament}, \end{cases}$$

COMPTADOR: *S'incrementa el comptador* $w \leftarrow w + 1$*.*

SORTIDA: *Es retorna el polinomi* $\Lambda(X)$*.*

Per a tots dos algorismes val el resultat següent, que assegura que corregeixen fins a τ errors:

Teorema 6.39. *Suposi's que en un dels algorismes 6.37 o 6.38 el resultat és un polinomi $\Lambda(X)$. Sigui $\Omega(X)$ el polinomi corresponent, calculat a partir de la identitat fonamental. Aleshores $d\big(\boldsymbol{u}(X), \mathcal{C}\big) \leqslant \lfloor \frac{\delta-1}{2} \rfloor$ si, i només si, aquests dos polinomis satisfan les condicions del lema 6.33.*

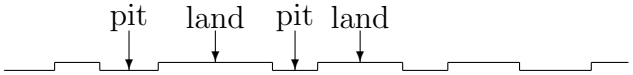
En aquest cas els polinomis són el localitzador i avaluador de la paraula d'error $\boldsymbol{e}(X)$ de pes $\|\boldsymbol{e}(X)\| = \deg \Lambda$ tal que $\boldsymbol{u}(X) = \boldsymbol{c}(X) + \boldsymbol{e}(X)$ amb $\boldsymbol{c}(X) \in \mathcal{C}$.

6.8 Exemple: el CD

El sistema *Compact Disc Digital Audio* per emmagatzemar audio en format digital en un dispositiu de lectura òptica va ser desenvolupat per Philips i Sony i adoptat com a estàndard pels principals fabricants l'any 80.

El suport és un disc de 12 cm de diàmetre i 1.2 mm de gruix. El disc és una superfície de plàstic metal·litzat on es grava la informació, i que es recobreix amb plàstic transparent per protegir-la permetent però la lectura. La informació està gravada en una pista en forma espiral, de l'interior cap a l'exterior, d'uns 5 Km de llarg.

La gravació consisteix en incisions (pits) d'una profunditat de $0.11\mu m$, amplada de $0.6\mu m$, i llargada variant entre $0.9\mu m$ i $3.3\mu m$ aproximadament. Les incisions estan separades per espais (lands) de les mateixes longituds. Pits i lands representen seqüències d'entre 2 i 10 zeros i la transició entre l'un i l'altre representa un 1. Un tall transversal de la pista de gravació tindria l'aspecte següent:



00010010001000000010010000010010000010000100

Per tant, la informació binària que es grava a un CD té unes característiques molt especials. Es tracta de cadenes de zeros i uns amb les condicions següents:

- els uns estan sempre aïllats entre dos zeros,
- els zeros van en grups d'entre 2 i 10.

Una part important de la feina de codificació i la que infla més el volum d'informació és la que s'encarrega de transformar cadenes binaries, que en principi poden ser qualsevol, en cadenes que compleixin aquestes condicions.

Transformació Analògic/Digital. El so és un fenomen que es pot representar, de manera analògica, com una funció que dona l'amplitud a cada instant de temps. El procés de transformar aquesta informació en informació digital té dos passos:

Mostreig: A intervals regulars de temps es mesura el valor de la funció amplitud. En el cas del CD el mostreig es fa a 44.1 kHz; és a dir, es prenen 44 100 mostres per segon.

Quantització: Els valors obtinguts en el mostreig són nombres reals. El procés de quantització els substitueix per aproximacions que es puguin emmagatzemar amb un nombre fixat de bits. En el cas del CD la quantització es fa amb paraules de 16 bits; o sigui, es considera que l'amplitud pot prendre només 65536 valors diferents.

Com que el CD conté so estereo, s'han de processar dos canals alhora. La quantitat d'informació que cal gravar per segon és:

$$2 \text{ canals} \quad \times \quad 44\,100 \text{ mostres} \quad \times \quad 16 \text{ bits} \quad = \quad 1.4112 \text{ Megabits/segon.}$$

Correcció d'errors: codificació. Per fer la codificació, les mostres de so s'agrupen en blocs que contenen 6 mostres consecutives per cada canal. Aquests blocs s'anomenen *frames* i formen paraules de $2 \times 6 \times 16 = 192$ bits, que s'interpreten com a paraules $\boldsymbol{m}$ de 24 bytes. Identificant els bytes amb elements del cos finit $\mathbb{F}$ de 256 elements, es té $\boldsymbol{m} \in \mathbb{F}^{24}$.

Les paraules $\boldsymbol{m}$ es codifiquen amb un codi de Reed-Solomon $\mathcal{C}_1$ de dimensió 24 i longitud 28; per tant, amb distancia mínima $d = n - k + 1 = 5$. Així, cada paraula $\boldsymbol{m}$ es converteix en una paraula $\boldsymbol{c} = \text{enc}(\boldsymbol{m}) \in \mathbb{F}^{28}$ de longitud 28.

La seqüència de les paraules $\boldsymbol{c}$ es transforma amb un sistema d'entrellaçament amb retard de 4 posicions. D'aquesta manera les paraules $\boldsymbol{c}$ donen lloc a paraules $\boldsymbol{c}'$, també de longitud 28, que consisteixen simplement en les lletres de les $\boldsymbol{c}$ "entrellaçades".

Les paraules $\boldsymbol{c}'$ es codifiquen amb un altre codi de Reed-Solomon $\mathcal{C}_2$ de dimensió 28 i longitud 32; per tant, també amb distancia mínima $d = n + k - 1 = 5$. Així, cada paraula $\boldsymbol{c}'$ es converteix en una paraula $\boldsymbol{c}''$ de longitud 32.

Per tant el procés de codificació amb els codis correctors d'errors $\mathcal{C}_1$ i $\mathcal{C}_2$ converteix cada frame de 24 bytes en una paraula de 32 bytes.

Correcció d'errors: descodificació. El descodificador del CD funciona de la manera següent: A l'entrada del descodificador es reben paraules de longitud 32 que, si no hi ha errors, pertanyen al codi $\mathcal{C}_2$.

Ara s'aplica la tècnica mixta de correcció/detecció que corregeix un error i en detecta 3: si la paraula rebuda no conté errors o conté només un error (error aïllat) es fa servir el codi $\mathcal{C}_2$ per corregir-lo i es descodifica amb la paraula de longitud 28 corresponent. Tot i que el codi $\mathcal{C}_2$ pot corregir fins a 2 errors només es fa servir per corregir-ne un. En canvi, si es detecta més d'un error és probable que s'hagi produït un error de ràfega i el descodificador marca totes les 28 posicions que correspondrien a la paraula descodificada com a esborralls.

Un cop fet aquest procés es desfà l'entrellaçat amb retard obtenint-se així una seqüència de paraules de 28 lletres, amb esborralls procedents de les paraules que el codi $\mathcal{C}_2$ no ha sabut descodificar. El codi $\mathcal{C}_1$ es fa servir aleshores per descodificar aquesta informació aplicant correcció d'esborralls o d'errors en funció que la paraula contingui o no esborralls.

Degut a la codificació usant intercalat amb retard el sistema pot arribar a corregir ràfegues de fins a 3584 bits consecutius, que ocupen uns 3 mm. de la pista gravada en el CD.

Byte de control i display. Després de codificar els frames, a cada paraula de longitud 32 se li afegeix un byte més per *control* i *display*. Aquests bytes contenen informació sobre el

contingut del disc: autor, títols, durada, temps que falta fins al final, etc. que l'aparell lector pot proporcionar en un visor i utilitza també per saltar d'una cançó a una altra, etc.

Així cada frame inicial s'ha convertit en una paraula de 33 bytes.

Modulació. La *modulació* és el procés que converteix les paraules de 33 bytes amb la informació ja codificada en el format apte per ser escrit a la superfície del disc. En primer lloc cada byte es converteix en una paraula de 14 bits donada per una taula (com que només hi ha 256 bytes, la taula no ocupa molt espai). Aquest procés s'anomena modulació ETF (eight to fourteen). Com a exemple, aquesta és una part de la taula de modulació ETF:

byte	codi
01110010	10010010000010
01110011	00100000100010
01110100	01000010000010
01110101	00000010000010
01110110	00010001000010
01110111	00100001000010
01111000	01001000000010
01111001	00001001001000
01111010	10010000000010
01111011	10001000000010
01111100	01000000000010

Finalment, i per tal que tota la seqüència final tingui les característiques adequades, cada dues paraules de longitud 14 s'enganxen amb una paraula de 3 bits: si les seqüències de zeros a banda i banda són massa curtes s'escullen tres zeros; si són massa llargues s'hi posa un 1 al lloc adequat. D'aquesta manera cada octet del frame es converteix en 17 bits. Per tant el frame ha quedat convertit en una paraula de $33 \times 17 = 561$ bits que ja es pot gravar al disc.

Sincronització. A cadascuna d'aquestes paraules de 561 bits se li afegeix la paraula de 24 bits següent:

100000000001000000000010

i tres bits més per tal que enganxi bé amb la paraula que vingui després. Aquesta paraula de 24 bits no es genera mai amb la modulació ETF de la informació, i el lector la reconeix per tal de sincronitzar-se i trencar bé els frames. Per tant cada frame ha quedat convertit en 588 bits que, per fi!, són els que es graven físicament al disc.

Resumint, cada frame conté

tipus de bits	num.
bits d'informació (audio)	192
redundància codis correctors	64
byte per control i display	8
afegits per la modulació ETF	297
paraula de sincronització	27
total bits de canal	588

O sigui, el nombre de bits que es graven físicament sobre la superfície del disc (bits de canal) és $588/192 = 3.0625$ vegades el nombre de bits d'informació que corresponen efectivament al so. Observi's que dels bits que cal afegir la major part són per culpa de les característiques físiques del sistema de lectura, especialment de la modulació ETF, i només una part petita prové de la redundància dels codis correctors.

En total, els 1.4112 Megabits/segon d'informació audio es converteixen en 4.3218 Megabits/segon d'informació de canal.

Lectura de la informació. La gravació i la lectura del disc es fan a velocitat lineal constant, que pot variar entre 1.2 i 1.4 m/s. segons el dispositiu de gravació. La velocitat de rotació varia entre 500 rpm quan es llegeix prop del centre fins a 200 rpm quan es llegeix prop de la vora.

Com que el nombre de bits per segon és fix, les diferents velocitats lineals corresponen a longituds diferents de pits i lands. D'aquesta manera, amb una longitud de pista fixada, les diferents velocitats de gravació-lectura corresponen a diferents durades del disc. Amb la màxima velocitat que admeten els gravadors-lectors més precisos s'arriben a gravar en un disc fins a uns 80 minuts d'audio. No cal que el reproductor conegui la velocitat a que s'ha gravat el disc: s'adapta automàticament per tal de llegir 4.3218 Megabits d'informació per segon.

El quadre següent resumeix el funcionament del CD:

Senyal analògica (sò)
$\Downarrow$
Mostreig: 44100 mostres/segon per canal
Cada mostra = 16 + 16 bits = 4 bytes
$\Downarrow$
frame = 6 mostres = 192 bits = 24 bytes
byte = element de $\mathbb{F}_{256}$
$\Downarrow$
afegir 4 bytes de redundància
codi Reed-Solomon [24,28]
$\Downarrow$
entrellaçat creuat amb retard de 4 posicions
$\Downarrow$
afegir 4 bytes de redundància
codi Reed-Solomon [28,32]
$\Downarrow$
modulació EFM, sincronització, display
frame = 571 bits
$\Downarrow$
gravació

Bibliografia bàsica

[1] Ash, Robert B., *Information theory*, Dover, 1965.

[2] Ball, Simeon, *A Course in Algebraic Error-Correcting Codes*, Birkhäuser, 2020.

[3] Brunat, Josep M.; Ventura, Enric, *Informació i codis*, Edicions UPC, 2001.

[4] Cover, Tom M.; Thomas, Joy A., *Elements of information theory*, 2nd. Ed. Wiley-Interscience, 2006. *pdf*.

Bibliografia complementaria

[5] Bell, C.; Cleary, J.C.; Witten, I.H. *Text Compression.* Prentice Hall. 1990.

[6] Berlekamp, E.R. *Algebraic coding theory.* Mac-Graw Hill. 1968.

[7] Berstel, J; Perrin, D; Reutenauer, *Codes and automata.* Cambridge University Press 2009. *pdf*.

[8] Robert G. Gallager *Variations on a theme by Huffman.* IEEE Transactions on Information Theory, vol. IT-24, no. 6, 1978.

[9] Robert G. Gallager *Low-Density Parity-Check Codes.* Expanded and revised version of 1960 M.I.T. Ph.D. thesis. 1963.

[10] Mac Williams, F.J.; Sloane, N.J.A. *The Theory of Codes.* Academic Press. 1985.

[11] MacKay, David J.C. *Information theory, inference and learning algorithms*, Cambridge University Press, 2005. *pdf*.

[12] Khinchin, A.I. *Mathematical foundations of information theory.* Dover. 1957.

[13] Pohlmann K.C. *Principles of Digital Audio.* (3rd Edition) McGraw-Hill. 1995.

[14] Roman, S. *Coding and Information Theory.* Springer-Verlag Graduate Texts in Mathematics. 1992.

[15] Roman, S. *Introduction to Coding and Information Theory.* Springer-Verlag Undergraduate Texts in Mathematics. 1997.

[16] Salomon, D., Motta, G. *Handbook of data compression.* 5th Ed. Springer. 2010.

[17] Sayood, K. *Introduction to data compression.* 5th Ed. Elsevier. 2017.

[18] Wicker, S.B.; Bhargava, V.K. *Reed-Solomon codes and their applications.* IEEE Press. 1994.

Articles fundacionals

[19] *Bose, Raj Chandra*; *Ray-Chaudhuri, Dijen K.*, *On a class of error correcting binary group codes*. Information and Control **3**, no. 1 (1960), pp. 68–79.

[20] *Golay, M. J. E.*, *Notes on digital coding*. Proceedings of the IRE **37** (June 1949), pag. 657.

[21] *Hamming, Richard W.*, *Error detecting and error correcting codes*. Bell Sys. Tech. J. **29** (1950), pp. 147–160.

[22] *Hocquenghem, Alexis*, *Codes correcteurs d'erreurs*, Chiffres 2 (Paris, september 1959), pp. 147–156. Aquest article sembla introbable.

[23] *Huffman, David A.*, *A method for the construction of minimum redundancy codes*. Proc. IRE, **40** (1952), pp. 1098–1101.

[24] *Lempel, Abraham*; *Ziv, Jacob*, *A universal algorithm for sequential data compression*. IEEE Trans. Information Theory, **23**, no. 3 (1977), pp. 337–343. *pdf*.

[25] *Lempel, Abraham*; *Ziv, Jacob*, *Compression of individual sequences via variable-rate code*. IEEE Trans. Information Theory, **24**, no. 5 (1978), pp. 530–536. *pdf*.

[26] *Reed, Irving S.*; *Solomon, Gustave*, *Polynomial codes over certain finite fields*. J. Soc. Indust. Appl. Math. Proc. **8** no. 2 (1960), pp. 300–304.

[27] *Shannon, Claude E.*, *A mathematical theory of communication*. Bell Sys. Tech. J. **27** (July 1948), pp. 379–423, and **27** (Oct. 1948) pp. 623–656.

[28] Shannon, Claude E., *Communication in the presence of noise*. Proc. IRE, **37** (1949), pp. 10–21.

[29] Shannon, Claude E., *Communication theory of secrecy systems*. Bell Sys. Tech. J. **28** (Oct. 1949), pp. 656–715.

[30] Shannon, Claude E., *Prediction and entropy of printed english*. Bell Sys. Tech. J. **30** (Jan. 1951), pp. 50–64.

Altres articles

[31] Gao, Shuhong, *A new algorithm for decoding Reed-Solomon codes*. In Bhargava et al (eds). Communications, information and network security. Chapter 5 (2003).