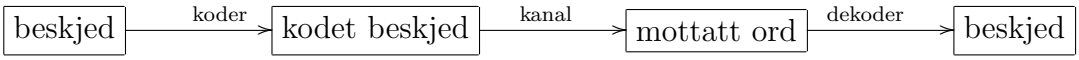
Forelesningsnotater SIF 5021 Algebra og tallteori
V-02. Et kort innføring med eksempler fra kodeteori

Sverre O. Smalø

I forbindelse med elektronisk digital kommunikasjon vil kommunikasjonskanalen av og til forandre på noen av informasjonssymbolene eller informasjonsordene på grunn av forstyrrelser som kan være menneskelig feil, lynpåvirkning, elektrisk interferens, kosmisk stråling eller på grunn av svakheter i lese-, skrive- eller opptaksenhetene. Derfor blir ofte ekstra symboler lagt til slik at en kan oppdage at en feil har oppstått, og av og til legges flere symboler til slik at en ikke bare kan oppdage feil, men også rette dem opp. Dette kan illustreres med følgende figur:



Et eksempel på feiloppdagende koding er: The American Standard Code for Information Interchange, ASCII.

Her er informasjonsordene alle 7-bits binære sekvenser:

$$\{(a_1, a_2, ...a_7); a_i \in \{0, 1\}\}$$

For eksempel svarer bokstaven A (stor A) til sekvensen $(1, 0, 0, 0, 0, 0, 1)$ og bokstaven B (stor B) svarer til sekvensen $(1, 0, 0, 0, 0, 1, 0)$ og bokstaven C svarer til sekvensen $(1, 0, 0, 0, 0, 1, 1)$.

Til de 7-bits binære sekvensene legger en så til et ekstra bit i følge regelen at i den nye 8-bit sekvensen skal antall 1-ere alltid være et jamnt tall (partall), og derfor også antall 0-ere et jamnt tall (partall).

Etter denne endringen svarer da bokstaven A til den 8-bit sekvensen $(1, 0, 0, 0, 0, 0, 1, 0)$, B til sekvensen $(1, 0, 0, 0, 0, 1, 0, 0)$ og bokstaven C til sekvensen $(1, 0, 0, 0, 0, 0, 1, 1, 1)$.

Vi sier her at koden består av alle 8-bit sekvenser med jamn paritet. Så, når en elektronisk enhet mottar et symbol i ASCII-kode vil den alltid sjekke om paritetsbetingelsen er tilfredsstilt før symbolet blir behandlet. Dersom paritetsbetingelsen ikke er tilfredsstilt, vet enheten at en feil har oppstått og spør om at siste symbol blir sendt en gang til. En sier da at ASCII-koden har evne til å oppdage en feil og bruker en redundans.

Andre slike feildetekterende systemer er i bruk, for eksempel enhver bok er utstyrt med et ISBN-nummer. Dette "International Standard Book Numberin System" tilordner et 10-sifret tall til hver bok, samt at tallet er delt i

fire ved hjelp av tre bindestrek (eller opperom), der den siste bindestreken (opperommet) gir ingen informasjon og er faktisk overflødig.

Så til oppbyggingen: Tallet før den første bindestreken angir språk, 0- er engelsk, 2- er fransk, 3- er tysk, 91-er svensk og 82- er norsk. De neste sifrene angir forlag, de neste sifrene angir bok nummer på forlaget og etter den siste bindestreken er det bare et siffer som kan være et av sifrene 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 og X. Her representerer X desimal-tallet 10 og vi skal komme tilbake hvorfor vi har behov for dette tallet. Den betingelsen som skal være oppfylt for at et ti-sifret tall representert med sifrene $a_1a_2a_3a_4a_5a_6a_7a_8a_9a_{10}$ skal være et ISBN-nummer er at tallet 11 går opp i $1a_1 + 2a_2 + 3a_3 + 4a_4 + 5a_5 + 6a_6 + 7a_7 + 8a_8 + 9a_9 + 10a_{10}$.

Her er ISBN-tallet til en bok jeg var med på å skrive: 0-521-41134-3. Språket er altså engelsk, 521 er tallet som "Cambride University Press" har fått tildelt som forlegger, 41134 er boknummeret forlaget har gitt boken og kontrollsiferet er altså 3. Siden dette er et ISBN-nummer skal 11 gå opp i $1 \cdot 0 + 2 \cdot 5 + 3 \cdot 2 + 4 \cdot 1 + 5 \cdot 4 + 6 \cdot 1 + 7 \cdot 1 + 8 \cdot 3 + 9 \cdot 4 + 10 \cdot 3 = 10 + 6 + 4 + 20 + 6 + 7 + 24 + 36 + 30 = 11 + 5 + 22 + 2 + 11 + 2 + 22 + 2 + 33 + 3 + 22 + 8 = 11 + 22 + 11 + 22 + 11 + 33 + 22 + 11 = 13 \cdot 11$, som altså stemmer.

Det ekstra sifferet her gjør oss i stand til a oppdage en feil i et siffer. Det gjør en også i stand til å oppdage om to siffer har byttet plass, en feil som ofte oppstår i forbindelse med skriving av et tall ved hjelp av et tastatur. Det har forresten ingen betydning hvilke tall som byttes om, det vil i alle tilfeller bli oppdaget at en feil har forekommet. Dette skyldes at alle tallene som er brukt som "vekter" er forskjellige, og at tallet 11 er et primtall. Merk at dersom vi hadde brukt delelighet med 10 istedet for 11 ville vi ikke ha oppdaget en feil på størrelse 2 i femte siffer, og ingen feil på størrelse 5 på sifrene i posisjon 2, 4, 6, 8 og 10.

De to eksemplene jeg har tatt er systemer som er konstruert for å oppdage feil, men vi kan konstuere systemer som ikke bare påpeker feil, men som også er i stand til å rette opp feil dersom det ikke blir for mange av dem. For eksempel ASCII-koden vil akseptere en symbol dersom et partall antall feil er begått i overføringen.

Ved å legge til flere siffer er det mulig å rette opp feil. Den enkleste måten dette kan gjøres på er å repetere samme siffer tre ganger. (1,1,0,0) kodes da til (1,1,1,1,1,1,0,0,0,0,0,0), overføres ved en kanal til, la oss si (1,1,0,1,1,1,0,1,0,0,0,0). Mottaker kan da resonere som følger: dersom sansynlighet for feil i hvert bit er liten, er det mest sansynlig at det første sifferet er 1, at det neste er 1, at det tredje er 0, og at det siste også er null, og mottager kommer fram til (1,1,0,0) som er det opprinnelige ordet. En

kan selvsagt aldri være sikker her, men en vil minimalisere sansynligheten for å gjøre feil så mye som mulig.

Spørsmålet blir nå om det finnes bedre måter å gjøre dette på enn a repetere samme siffer tre ganger, eventuelt 5, 7 elle 2n+1 ganger?

Før vi starter med eksempler, la oss innføre noen begreper og presisjoner. Igjen, for ikke å bli for generell, la oss se på 7-bits binære sekvenser: $a = (a_1, a_2, a_3, a_4, a_5, a_6, a_7); a_i \in \{0, 1\}$. Dersom vi betrakter to slike sekvenser $a = (a_1, a_2, a_3, a_4, a_5, a_6, a_7)$ og $b = (b_1, b_2, b_3, b_4, b_5, b_6, b_7)$ så er det kanskje naturlig å bruke antall siffer der de er forskjellige som et mål på distansen mellom de to sekvensene. Dette gjør vi formelt ved a innføre en avstand $d(a, b) =$ antall i der $a_i \neq b_i$. For eksempel blir da $d((0, 0, 0, 0, 0, 0, 0), (1, 1, 0, 1, 1, 1, 1)) = 6$. Vanligvis brukes symbolet 0 også for sekvensen med bare nuller. En innfører da begrepet vekt til en sekvens a ved at $w(a) = d(0, a) =$ antall i med $a_i \neq 0$. For eksempel $w((1, 1, 0, 0, 0, 1, 1)) = 4$. Avstanden d vil nå ha de samme egenskapene som vanlig avstand i plan eller rom. Den oppfyller nemlig følgende betingelser:

$$d(a, b) = d(b, a); \ \forall a, b$$

$$d(a, b) = 0 \iff a = b$$

$$d(a, b) \leq d(a, c) + d(c, b); \ \forall a, b, c$$

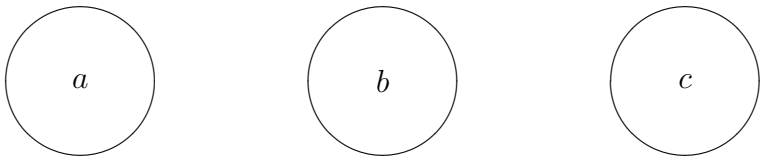
Den siste betingelsen blir kalt trekantulikheten.

Nå, la oss anta at vi har valgt en delmengde

$$U \subseteq \{(a_1, a_2, a_3, a_4, a_5, a_6, a_7); a_i \in \{0, 1\}\}$$

slik at $d(a, b) \geq 3 \ \forall a, b \in U$ med $a \neq b$, og at vi har en bijeksjon mellom mengden av beskjeder og U.

Vi kan nå illustrere egenskapen at avstanden mellom to elementer i U er minst 3 i planet ved at vi tar elementene i U, og alle 7-bits sekvenser med avstand mindre enn eller lik 1 fra disse elementene i U.



Vi får da disjunkte delmengder av alle 7-bits sekvenser som vi kaller ballene om elementene i U av radius 1. Dersom en sekvens fra U er overført via en kanal, og høyst en feil har oppstått, vil nå den mottatte sekvensen befinne seg i nøyaktig en av de disjunkte ballene, og hver ball vil inneholde nøyaktig ett element fra U. Vi dekoder da den mottatte sekvensen til dette entydige elmentet i U.

Spørsmålet blir nå: Finnes det gode valg for U? Her er svaret ja. Neste spørsmål blir da: Hvordan finne disse gode valg for U?

Konstruksjon av gode koder som også har en enkel innkoding og dekoding avhenger av kunnskap om endelige algebraiske strukturer, kjent som endelige kropper (eller galoiskropper) og lineær algebra over disse. Eksempler på slike som ofte brukes er $\{0,1\}$ med binær addisjon og multiplikasjon, og $\{0,1,2,3,4,5,6,7,8,9,X\}$ med addisjon og multiplikasjon redusert ved multiplum av 11.

La meg gi en slik U, den såkalte Hamming (7,4)-koden. 7 står for lengden av kodeordene, 4 står for lengden av beskjedordene og alfabetet er $\{0,1\}$.

$$\{(a_1,a_2,a_3,a_4); a_i \in \{0,1\}\} \to U \subseteq \{(a_1,a_2,a_3,a_4,a_5,a_6,a_7)\}$$

Her er det naturlig å introdusere binær matrisemultiplikasjon.

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

Denne matrisemultiplikasjon minner om vanlig matrisemultiplikasjon av reelle eller komplekse matriser, og vil faktisk tilfredsstille alle regler dere kjenner for matrisemultiplikasjon når dere erstatter vanlig addisjon og multiplikasjon med henholdsvis binær addisjon og binær multiplikasjon.

Jeg har foretatt alle multiplikasjonene av alle 4-bit sekvenser med den gitte 4×7-matrisen M gitt ovenfor, og jeg har fått følgende resultat, der nest siste kolonne gir alle elementene i U og siste kolonne angir vekten av hvert av elementene i U:

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

$(0\,0\,0\,0)$	$(0\,0\,0\,0\,0\,0\,0)$	0
$(0\,0\,0\,1)$	$(0\,0\,0\,1\,1\,1\,1)$	4
$(0\,0\,1\,0)$	$(0\,0\,1\,0\,1\,1\,0)$	3
$(0\,0\,1\,1)$	$(0\,0\,1\,1\,0\,0\,1)$	3
$(0\,1\,0\,0)$	$(0\,1\,0\,0\,1\,0\,1)$	3
$(0\,1\,0\,1)$	$(0\,1\,0\,1\,0\,1\,0)$	3
$(0\,1\,1\,0)$	$(0\,1\,1\,0\,0\,1\,1)$	4
$(0\,1\,1\,1)$	$(0\,1\,1\,1\,1\,0\,0)$	4
$(1\,0\,0\,0)$	$(1\,0\,0\,0\,0\,1\,1)$	3
$(1\,0\,0\,1)$	$(1\,0\,0\,1\,1\,0\,0)$	3
$(1\,0\,1\,0)$	$(1\,0\,1\,0\,1\,0\,1)$	4
$(1\,0\,1\,1)$	$(1\,0\,1\,1\,0\,1\,0)$	4
$(1\,1\,0\,0)$	$(1\,1\,0\,0\,1\,1\,0)$	4
$(1\,1\,0\,1)$	$(1\,1\,0\,1\,0\,0\,1)$	4
$(1\,1\,1\,0)$	$(1\,1\,1\,0\,0\,0\,0)$	3
$(1\,1\,1\,1)$	$(1\,1\,1\,1\,1\,1\,1)$	7

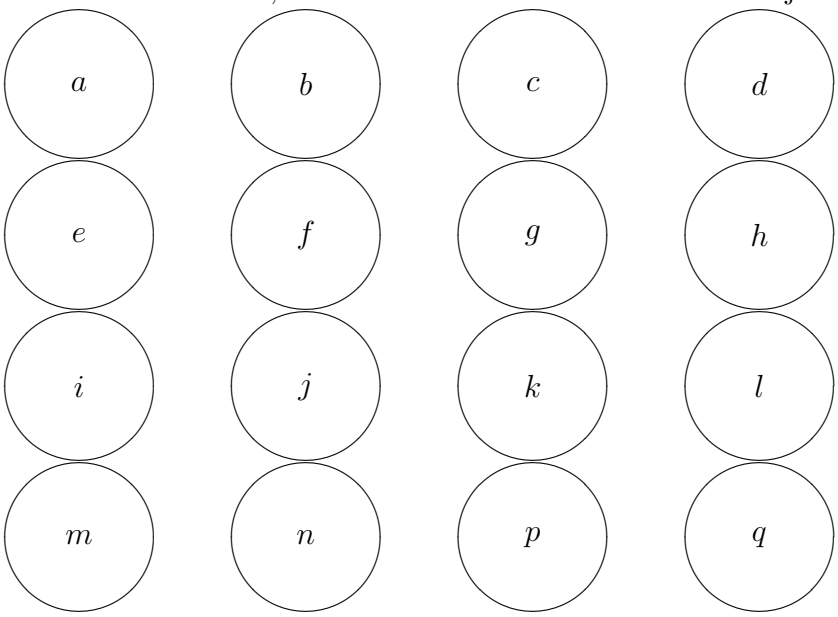
Matrisen M blir kalt generatormatrisen for koden og samlingen av alle sekvenser i U blir kalt koden. Fra lineær algebra vet dere at et underrom av det reelle vektorrommet $\mathbb{R}^7$ har essensielt to forskjellige beskrivelser, enten gitt ved en basis eller gitt som løsningsrommet til et ligningsett. Beskrivelsen ovenfor svarer til å gi en basis for underrommet. Vi har nemlig at alle elementene i U er binære lineærkombinasjoner av radene i matrisa M, som da på sett og vis er en basis for U. En kan også beskrive U som de 7-bits binære sekvenser a som er slik at når jeg ganger med følgende 7×3-matrise P fra høyre ved binær matrisemultiplikasjon, så får vi null. Matrisen P er gitt som følger:

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

Den litt observante leser oppdager at radene i matrisa P er gitt ved den binære representasjonen av tallene fra 1 til 7.

Jeg påstår nå at $d(a,b) \geq 3 \ \ \forall a, b \in U$ med $a \neq b$ der U er som ovenfor. Vi kunne ha fastslått dette ved å verifisere at $d(a,b) \geq 3$ for alle mulige valg av a og b som fører til at vi må foreta $(16 \times 15)/2$ sammenligninger. Men det er lett å se at dersom $xM = a \in U$ og $yM = b \in U$ så er $(x-y)M = a - b \in U$ og derfor har vi at dersom $a \neq b$ og begge er med i U, så er $d(a,b) = w(a-b) \geq 3$.

En annen egenskap som denne U-en tilfredsstiller er følgende: Dersom vi tar de 16 sekvensene i U og tar ballene av radius 1 om disse i rommet av alle 7-bit binære sekvenser, så har vi altså at disse ballene er disjunkte.



Videre vil hver slik ball inneholde 8 sekvenser da hver sekvens i rommet av 7-bits binære sekvenser har 7 "naboer" med avstand 1, og som da inklusive seg selv gir at hver ball med radius 1 inneholder 8 elementer. Vi får da $16 \times 8 = 2^7$ som er antall 7-bits binære sekvenser. At alle elementene blir med i nøyaktig en ball er ikke oppfylt av så mange koder, og slike koder har derfor fått sitt eget navn og blir kalt for perfekte koder. Det er kjent en familie av slike koder der antall symboler er q, en primtallspotens, lengden av sekvensene er gitt ved $n = (q^r - 1)/q - 1$, lengden av beskjedene er gitt ved q^{n-r} og avstanden er 3. I vårt eksempel var $q = 2,\ r = 3,\ n = 7,\ n - r = 4$.

I eksemplet jeg gikk gjennom brukte vi det binære alfabetet $\{0, 1\}$. La oss nå se på et annet alfabet og for enkelhets skyld ta alfabetet med 7 symboler, $\{0, 1, 2, 3, 4, 5, 6\}$. La oss se på sekvenser av lengde 8 med elementer hentet fra alfabetet med 7 symboler som over. Vi må først lære oss å operere med addisjon og multiplikasjon redusert med mulitiplum av 7. Det vi gjør er altså å foreta vanlig addisjon og multiplikasjon og til slutt trekker fra et passe multiplum av 7 slik at vi havner blant tallene 0,1,2,3,4,5,6. Matriseoperasjoner går da også greit.

Betrakt nå 6×8-matrisen M med koefisienter hentet fra vårt alfabet 0,1,2,3,4,5,6:

$$\begin{pmatrix} 1&0&0&0&0&0&1&6 \\ 0&1&0&0&0&0&1&5 \\ 0&0&1&0&0&0&2&4 \\ 0&0&0&1&0&0&3&3 \\ 0&0&0&0&1&0&4&2 \\ 0&0&0&0&0&1&5&1 \end{pmatrix}$$

Vi kan nå ta en sekvens av lengde 6 fra vårt alfabet, multiplisere fra høyre med matrisen M, redusere med et multiplum av 7 og få en sekvens av lengde 8 fra vårt alfabet. Mengden en får fram på denne måten er vår nye mengde U. Dersom vi bruker samme avstandsbegrep som for binære sekvenser på vår nye mengde U, vil avstanden mellom to forskjellige sekvenser i U være minst 3. Koden U vil her ha $7^6 = (50 - 1)^3 = 50^3 - 3 \times 50^2 + 3 \times 50 - 1 = 125000 - 7500 + 150 - 1 = 117649$ kodeord.

Paritetssjekkmatrisen P er her:

$$\begin{pmatrix} 0&1 \\ 1&0 \\ 1&1 \\ 1&2 \\ 1&3 \\ 1&4 \\ 1&5 \\ 1&6 \end{pmatrix}$$

Ved å bruke denne paritetssjekmatrisen blir dekodingen forholdsvis enkel:

La oss si at kodesekvensen a blir sendt og at sekvensen b mottas og at høyst en feil har oppstått. Feilen blir da $b-a$ siden $b-(b-a)=a$. I og med at det har oppstått høyst en feil, så må vi finne plassen der feilen har oppstått og hvor stor feilen er. La $e=b-a$. Vi har da at $eP=(b-a)P=bP-aP=bP-0=bP$ som vi kjenner. Dette er et tallpar (c,d) i vårt alfabet 0,1,2,3,4,5,6. Dersom vi ser på e så har denne formen $(0,...0,z,0,...0)$ der z er et tall blant 0,1,2,3,4,5,6, og den er plasert på plass nr. i. La $eP=(c,d)$. Dersom $(c,d)=(0,0)$ så er $b=a$ og det er ingenting å gjøre. Dersom $(c,d)\neq(0,0)$ og $c=0$ er $i=1$ og $z=d$ og vi kan finne $a=b-(d,0,0,0,0,0,0,0)$. Dersom $c\neq 0$ så er $z=c$ og i er bestemt av følgende formel: $i=2+dc^{-1}$ der $1^{-1}=1,\ 2^{-1}=4,\ 3^{-1}=5,\ 4^{-1}=2,\ 5^{-1}=3,\ 6^{-1}=6$, og det siste produktet er regnet ut og redusert modulo 7.

Dere vil kanskje ha oppdaget at tallet 7 ikke spiller så stor rolle, men at det er et primtall er viktig. Dere kan nå ta å erstatte 7 med et vilkårlig primtall p, produsere en $p-1\times p+1$-matrise M i følge systemet:

$$\begin{pmatrix}
1&0&0&0&0&0&\cdots&0&0&1&p{-}1\\
0&1&0&0&0&0&\cdots&0&0&1&p{-}2\\
0&0&1&0&0&0&\cdots&0&0&2&p{-}3\\
\vdots&\vdots&\vdots&\vdots&\vdots&\vdots&&\vdots&\vdots&\vdots&\\
0&0&0&0&0&0&\cdots&0&0&p{-}4&3\\
0&0&0&0&0&0&\cdots&1&0&p{-}3&2\\
0&0&0&0&0&0&\cdots&0&1&p{-}2&1
\end{pmatrix}$$

Dette blir innkodingsmatrisen til en $(p+1,p-1,3)$-kode med alfabet $\{0,1,\cdots p-1\}$ som altså retter en feil og har tilsammen p^{p-1} kodeord. For eksempel ved a bruke primtallet 11 får vi da en enkel kode med 11^{10} kodeord som overskrider antall mennesker på jorden, og ved å gi hvert menneske et 12-sifret telefonnr kan vi nå den vi ville ringe selv om vi slo ett siffer feil. Dette ville kanskje redusere antall feilringinger betraktelig, men teleselskapene vil vel tape på dette.

Paritetssjekkmatrisen til dette systemet er gitt ved følgende matrise:

$$\begin{pmatrix}
0&1\\
1&0\\
1&1\\
1&2\\
1&3\\
1&4\\
\vdots&\vdots\\
1&p{-}2\\
1&p{-}1
\end{pmatrix}$$

Til slutt vil jeg bare nevne at personnummersystemet i Norge er basert på en feiloppdagende kode med 2 kontrollsiffer. Det er bygd opp ved fødselsdag,

måned og år, så kommer tre nesten vilkårlige siffer der det tredje angir kjønn ved at kvinner har et partall som sitt 9-ende siffer og menn har et odde tall. Til slutt er det lagt til to kontrollsiffer. Alfabetet som er brukt er $\{0,1,2,3,4,5,6,7,8,9\}$ og all aritmetikk er modulo 11. Denne koden er konstruert for å oppdage ikke bare tilfeldige feil, men også en del vanlige feil som at folk blander måned og dag, tar hensyn til håndstillingsfeil på tastaturet og en del andre typiske feil. En kontrollmatrise her kan for eksempel være

$$\begin{pmatrix} 3 & 5 \\ 7 & 4 \\ 6 & 3 \\ 1 & 2 \\ 8 & 7 \\ 9 & 6 \\ 4 & 5 \\ 5 & 4 \\ 2 & 3 \\ 1 & 2 \\ 0 & 1 \end{pmatrix}$$

Det vil si, tar en et gyldig personnummer og oppfatter det som en sekvens av lengde 11 og ganger med matrisen ovenfor fra høyre, så vil resultatet bli et tallpar der hvert av tallene er delelig med 11.

Kilder:

R. Hill, *A First Course in Coding Theory,* Clarendon Press, Oxford 1986.

E. S. Selmer: *Personnummerering i Norge: Litt anvendt tallteori og psykologi*, Nordisk Matematisk Tidsskript, Bind 12 (1964), 36-44.

Trondheim 21/1-02