Moderne kryptografi

Olav Geil
Institut for Matematiske Fag
Aalborg Universitet

Elektronik og IT-Gruppen
24. april 2008

Matematik og ingeniørvidenskab

- Uden ingeniørvidenskab var komplekse tal blot en kuriøsitet
- Uden computergrafik var kvaternioner blot en kuriøsitet
- Uden informationsteknologi var *endelige legemer* blot en kuriøsitet
- Matematik halter til tider bagefter ingeniørvidenskaben
- I moderne kryptografi er matematikken med hele vejen

Agenda

- Endelige legemer
- AES
- McEliece kryptosystem
- ElGamal kryptosystem
- Elliptisk kurve kryptosystem

Endelige legemer - type 1

$\mathbf{F}_5 = \{0, 1, 2, 3, 4\}$

Regneregel: $5 = 0$.

Eksempel:
$3 + 4 = 7 = 5 + 2 = 2$
$3 \cdot 4 = 12 = 5 + 5 + 2 = 2$

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	0
2	2	3	4	0	1
3	3	4	0	1	2
4	4	0	1	2	3

·	0	1	2	3	4
0	0	0	0	0	0
1	0	1	2	3	4
2	0	2	4	1	3
3	0	3	1	4	2
4	0	4	3	2	1

Endelige legemer - type 2

$\mathbf{F}_4 = \{0, 1, \alpha, \alpha + 1\}$

Regneregler: $2 = 0 \qquad \alpha^2 = \alpha + 1$

Eksempler:
$(\alpha + 1) + \alpha = 2\alpha + 1 = 1$

$(\alpha + 1)\alpha = \alpha^2 + \alpha = (\alpha + 1) + \alpha = 2\alpha + 1 = 1$

$\alpha\alpha = \alpha^2 = \alpha + 1$

Endelige legemer - type 2

$\mathbf{F}_4 = \{0, 1, \alpha, \alpha + 1\}$

Regneregler: $2 = 0 \qquad \alpha^2 = \alpha + 1$

+	0	1	α	$\alpha + 1$
0	0	1	α	$\alpha + 1$
1	1	0	$\alpha + 1$	α
α	α	$\alpha + 1$	0	1
$\alpha + 1$	$\alpha + 1$	α	1	0

$\cdot$	0	1	α	$\alpha + 1$
0	0	0	0	0
1	0	1	α	$\alpha + 1$
α	0	α	$\alpha + 1$	1
$\alpha + 1$	0	$\alpha + 1$	1	α

Endelige legemer - type 2

Repræsentation af $\mathbf{F}_4$ i computer

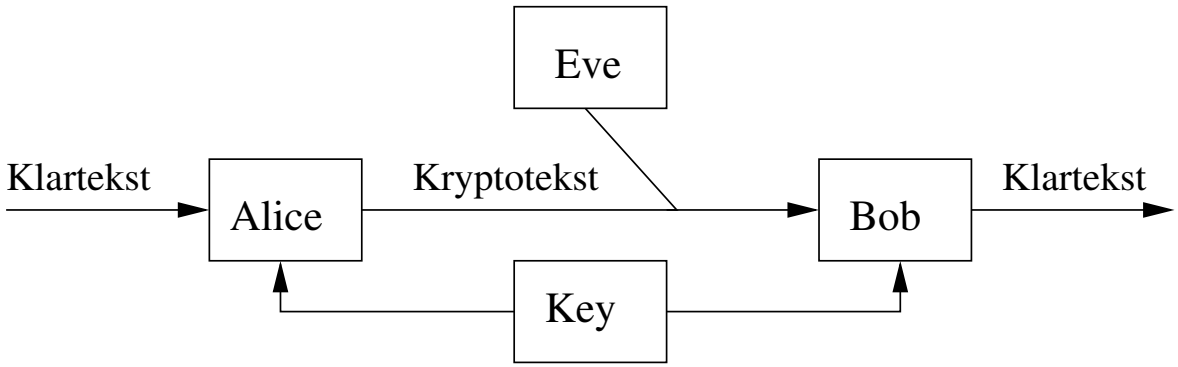
0	1	α	$\alpha + 1$
(00)	(10)	(01)	(11)

Kan konstruere LEGEMER $\mathbf{F}_{p^m}$,
for alle primtal p og positive heltal m.

Eksempel:
$\mathbf{F}_{256} = \mathbf{F}_{2^8} = \{a_7\alpha^7 + a_6\alpha^6 + \cdots + a_1\alpha + a_0 \mid a_7, \ldots, a_0 \in \{0, 1\}\}$
kan repræsenteres vha. bytes.

Advanced Encryption Standard

Symmetrisk kryptosystem



Enkryptering $\Leftrightarrow$ Dekryptering

DES (kan nu brydes)

AES= Rijndael Block Cipher (implementeres fra 2002)

AES

Klartekst:

128 bits $(m_1, m_2, \ldots, m_{128})$

16 bytes $(M_1, M_2, \ldots, M_{16})$

$$\begin{bmatrix} M_1 & M_2 & M_3 & M_4 \\ M_5 & M_6 & M_7 & M_8 \\ M_9 & M_{10} & M_{11} & M_{12} \\ M_{13} & M_{14} & M_{15} & M_{16} \end{bmatrix}$$

Kryptotekst:

$$\begin{bmatrix} M'_1 & M'_2 & M'_3 & M'_4 \\ M'_5 & M'_6 & M'_7 & M'_8 \\ M'_9 & M'_{10} & M'_{11} & M'_{12} \\ M'_{13} & M'_{14} & M'_{15} & M'_{16} \end{bmatrix}$$

16 bytes $(M'_1, M'_2, \ldots, M'_{16})$

128 bits $(m'_1, m'_2, \ldots, m'_{128})$

AES

$$\begin{bmatrix} M_1 & M_2 & M_3 & M_4 \\ M_5 & M_6 & M_7 & M_8 \\ M_9 & M_{10} & M_{11} & M_{12} \\ M_{13} & M_{14} & M_{15} & M_{16} \end{bmatrix} \overset{?}{\longleftrightarrow} \begin{bmatrix} M'_1 & M'_2 & M'_3 & M'_4 \\ M'_5 & M'_6 & M'_7 & M'_8 \\ M'_9 & M'_{10} & M'_{11} & M'_{12} \\ M'_{13} & M'_{14} & M'_{15} & M'_{16} \end{bmatrix}$$

Enkryptering og dekryptering kompliceret men hurtig.

AES

Regning med bytes mange muligheder...en er:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} (\text{mod} 2) = \begin{bmatrix} v_0 \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \\ v_7 \end{bmatrix}$$

Hvis matrix invertibel, så proces reversibel.

AES

Regning med bytes mange muligheder ... en anden er:
Identificer $(u_7, u_6, \ldots, u_1, u_0)$ med
$u_7\alpha^7 + u_6\alpha^6 + \cdots u_1\alpha + u_0$

Regneregler: $2 = 0$
$\alpha^8 = \alpha^4 + \alpha^3 + \alpha + 1$

$$
\begin{aligned}
& (\alpha^2+1)(\alpha^7+\alpha^5+\alpha^4+\alpha^3+1) \\
= \;& \alpha^9+2\alpha^7+\alpha^6+2\alpha^5+\alpha^4+\alpha^3+\alpha^2+1 \\
= \;& \alpha^9+\alpha^6+\alpha^4+\alpha^3+\alpha^2+1 \\
= \;& \alpha\alpha^8+\alpha^6+\alpha^4+\alpha^3+\alpha^2+1 \\
= \;& \alpha(\alpha^4+\alpha^3+\alpha+1)+\alpha^6+\alpha^4+\alpha^3+\alpha^2+1 \\
= \;& \alpha^6+\alpha^5+2\alpha^4+\alpha^3+2\alpha^2+\alpha+1 \\
= \;& \alpha^6+\alpha^5+\alpha^3+\alpha+1
\end{aligned}
$$

$$
\begin{aligned}
& (\alpha^2+1)+(\alpha^4+\alpha^3+\alpha^2+\alpha) \\
= \;& \alpha^4+\alpha^3+2\alpha^2+\alpha+1 \\
= \;& \alpha^4+\alpha^3+\alpha+1
\end{aligned}
$$

F_{256}

Kan trække fra og dividere:

$-(\alpha^7 + \alpha^5 + \alpha^2 + 1) = \alpha^7 + \alpha^5 + \alpha^2 + 1$ fordi
$(\alpha^7 + \alpha^5 + \alpha^2 + 1) + (\alpha^7 + \alpha^5 + \alpha^2 + 1) = 0$

For alle $a_7\alpha^7 + \cdots + a_0$ findes $b_7\alpha^7 + \cdots + b_0$ så
$(a_7\alpha^7 + \cdots + a_0)(b_7\alpha^7 + \cdots + b_0) = 1$.
Vi skriver $\frac{1}{a_7\alpha^7+\cdots+a_0} = b_7\alpha^7 + \cdots + b_0$, så vi kan også dividere.

AES

Indkodning i 10 runder:

- ByteSub
- ShiftRows
- MixColumns
- AddRoundKey (en hemmelig nøgle genererer 10 nøglematricer $K_1, \ldots, K_{10}$)

Alt kan inverteres.

ByteSub

Først $a_7\alpha^7 + \cdots + a_0 \mapsto \frac{1}{a_7\alpha^7+\cdots+a_0}$.

Så

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} u_0 \\ u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{bmatrix} \pmod 2$$

AES runder

ShiftRows:
Bytes i matrix skifter plads.

MixColumns:

$$\begin{bmatrix} \alpha & \alpha+1 & 1 & 1 \\ 1 & \alpha & \alpha+1 & 1 \\ 1 & 1 & \alpha & \alpha+1 \\ \alpha+1 & 1 & 1 & \alpha \end{bmatrix} \begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} \\ S_{21} & S_{22} & S_{23} & S_{24} \\ S_{31} & S_{32} & S_{33} & S_{34} \\ S_1 & S_{42} & S_{43} & S_{44} \end{bmatrix}$$

AddRoundKey:

$$\begin{bmatrix} S_{11} & S_{12} & S_{13} & S_{14} \\ S_{21} & S_{22} & S_{23} & S_{24} \\ S_{31} & S_{32} & S_{33} & S_{34} \\ S_1 & S_{42} & S_{43} & S_{44} \end{bmatrix} + K_i$$

McEliece kryptosystem

Gør brug af fejlkorrigerende koder.

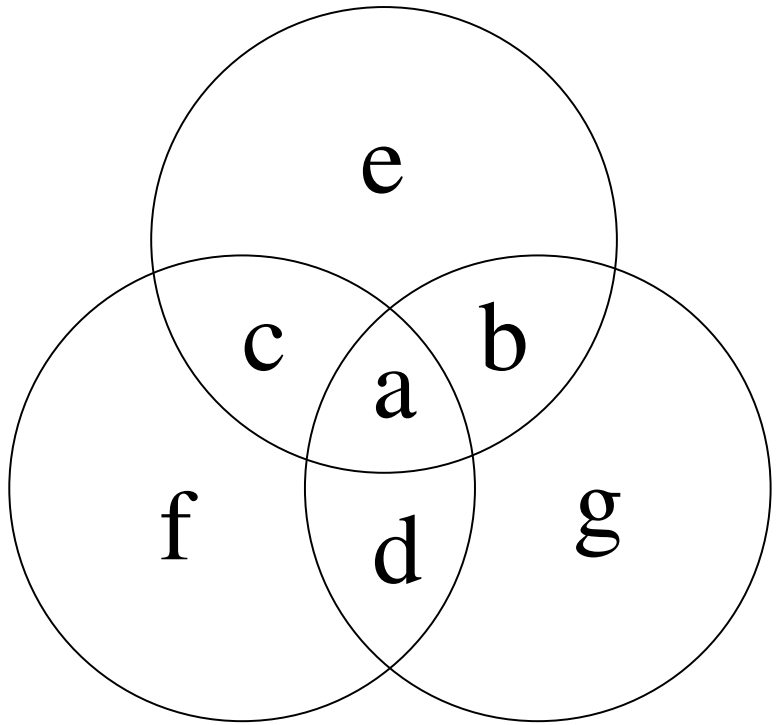
Fejlkorrigerende koder beskytter data mod støj.

En slags elektronisk indpakning.

Fejlkorrigerende koder benyttes i CD-afspiller, DVD-afspiller, digital radio og tv, rumfart osv. osv. ... og snart i chips i forbrugsvarer.

Fejlkorrigerende koder

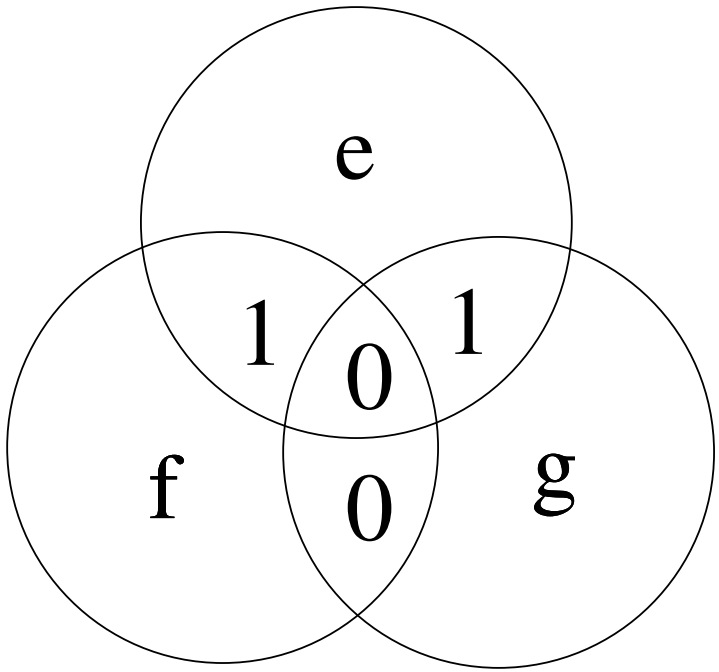
Indkodning: (a, b, c, d) indkodes til (a, b, c, d, e, f, g)

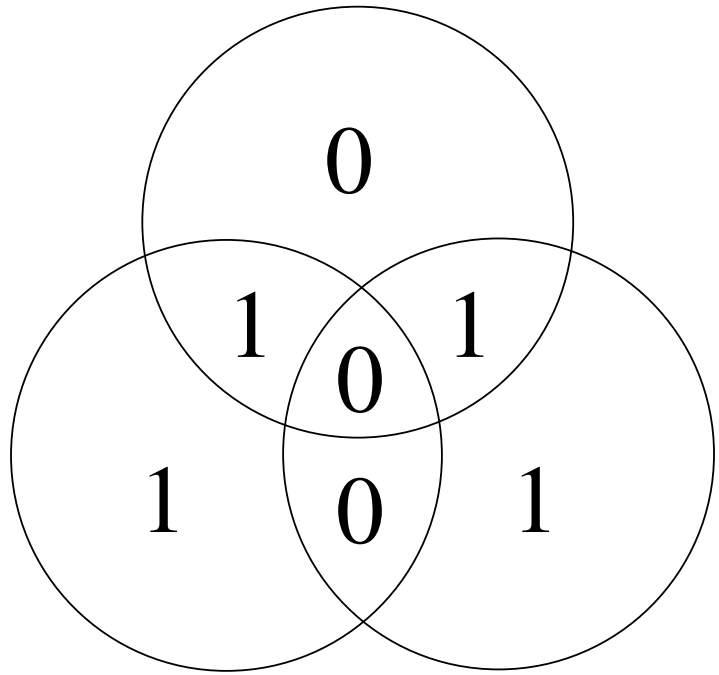


Regneregel: $2 = 0$
Indkodningsregler: $a + b + c + e = 0, \quad a + c + d + f = 0,$
$a + b + d + g = 0$

Fejlkorrigerende koder

Indkodning: $(0,1,1,0)$ indkodes til $(0,1,1,0,0,1,1)$



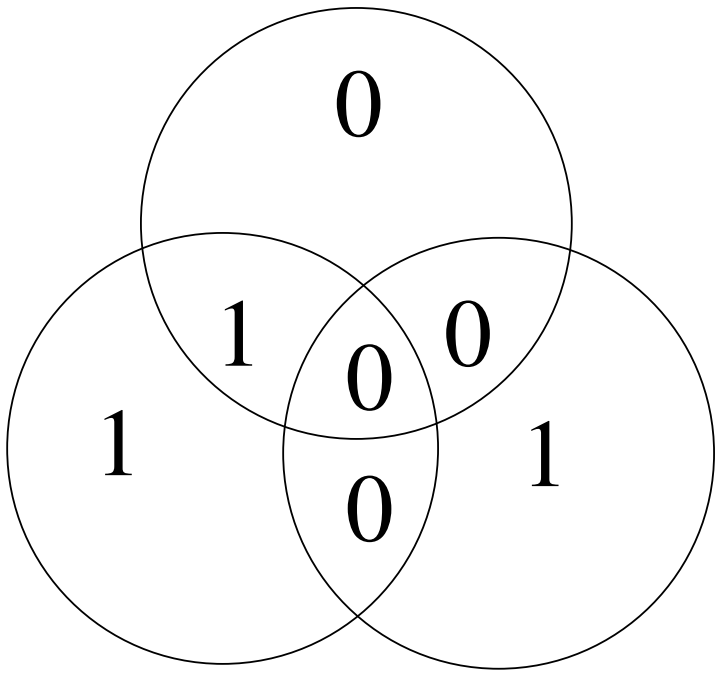


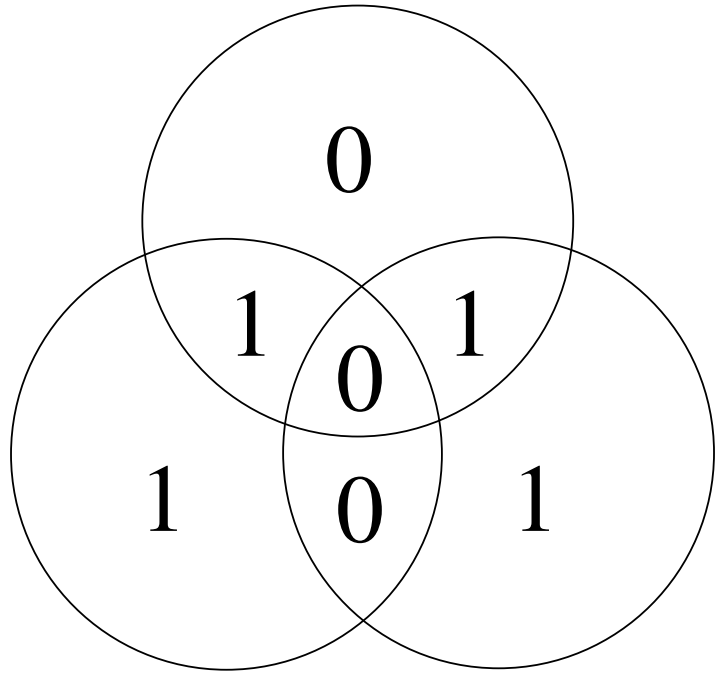
Regneregel: $2 = 0$

Indkodningsregler: $a + b + c + e = 0$, $a + c + d + f = 0$, $a + b + d + g = 0$

Fejlkorrigerende koder

Dekodning (fejlretning): $(0,0,1,0,0,1,1)$ afkodes til $(0,1,1,0,0,1,1)$ som svarer til besked $(0,1,1,0)$.





Lineære koder over $\mathbf{F}_q$

Lineær $[n, k]$-kode.
Beskeden $\vec{m}$ indkodes til kodeordet $\vec{c}$.

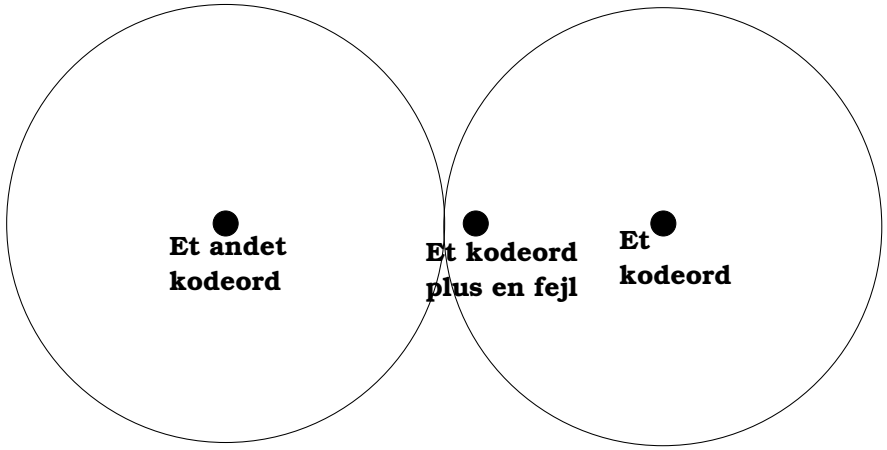
$$[m_1, \ldots, m_k] \begin{bmatrix} g_{1,1} & \cdots & g_{1,n} \\ \vdots & \ddots & \vdots \\ g_{k,1} & \cdots & g_{k,n} \end{bmatrix} = [c_1, \ldots, c_n]$$

Alternativ tjek for, at $\vec{c}$ kodeord er

$$\begin{bmatrix} h_{1,1} & \cdots & h_{1,n} \\ \vdots & \ddots & \vdots \\ h_{n-k,1} & \cdots & h_{n-k,n} \end{bmatrix} \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_n \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}$$

G kaldes generatormatrix. H kaldes paritetstjekmatrix.

Dekodningsproblemet

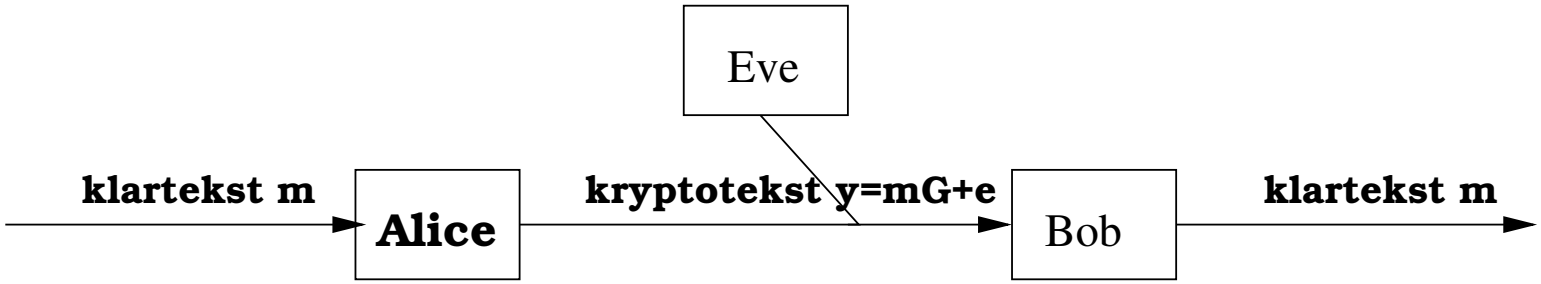


$\text{dist}(\vec{c}_i, \vec{c}_j) = \#$ positioner, hvor $\vec{c}_i$ afviger fra $\vec{c}_j$.
$\text{dist}\big((1,0,0,0,1),(1,1,0,0,0)\big) = 2$

$d = \min\{\text{dist}(\vec{c}_i, \vec{c}_j) \mid \vec{c}_i, \vec{c}_j \text{ er forskellige ord i koden}\}$
$t = \lfloor \frac{d-1}{2} \rfloor$ fejl kan med garanti rettes.

Dekodningsproblemet for generel lineær kode er NP-fuldstændigt.

McEliece kryptosystem



Offentlig nøgle: Generatormatrix G og t (fejlretningsevnen)

Hemmelig nøgle: Dekodningsalgoritme for koden med generatormatrix G.

Strategi: Alice indkoder besked $\vec{m}$ til kodeord $\vec{c} = \vec{m}G$ og tilføjer tilfældigt op til t fejl (fejlvektor $\vec{e}$).

Bob dekoder modtaget ord $\vec{y}$ til kodeord $\vec{c}$ og genskaber besked $\vec{m}$.

Hvorfor McEliece kryptosystemet virker

$G = SG'P$

S er $k \times k$ matrix af fuld rang og P er permutationsmatrix.

G' er generatormatrix for Goppa kode efter Bobs valg.

Ud fra G kan man ikke gætte S, G' og P

Dekodning en integreret proces, hvor S^{-1}, P^{-1} og dekodningsalgoritme for valgt Goppa kode benyttes.

Dekodning er hurtig. Benytter Euklids udvidede algoritme.

Binære Goppakoder

Goppapolynomiet: $g(X) = g_0 + g_1 X + \cdots + g_t X^t$, hvor $g_i \in \mathbf{F}_{2^m}$

$\gamma_0, \gamma_1, \ldots, \gamma_{n-1} \in \mathbf{F}_{2^m}$ vælges så
$g(\gamma_0) \neq 0, g(\gamma_1) \neq 0, \ldots, g(\gamma_{n-1}) \neq 0.$

Paritetstjekmatrix $H = XYZ$, hvor

$$X = \begin{bmatrix} g_t & 0 & 0 & \cdots & 0 \\ g_{t-1} & g_t & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_1 & g_2 & g_3 & \cdots & g_t \end{bmatrix} \quad Y = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ \gamma_0 & \gamma_1 & \cdots & \gamma_{n-1} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_0^{t-1} & \gamma_1^{t-1} & \cdots & \gamma_{n-1}^{t-1} \end{bmatrix}$$

$$Z = \begin{bmatrix} \frac{1}{g(\gamma_0)} & & & \\ & \frac{1}{g(\gamma_1)} & & \\ & & \ddots & \\ & & & \frac{1}{g(\gamma_{n-1})} \end{bmatrix}$$

Binære Goppakoder - fortsat

Koden er BINÆR selvom paritetstjekmatricen H lever over $\mathbf{F}_{2^m}$ (subfield-subcode).

En slags genereliseret genereliseret Reed-Solomon kode.

Tilhørende generatormatrix G' kan findes.

McEliece kryptosystem

Oprindeligt kodningskryptosystem foreslået af McEliece i 1978

Oprindeligt system benytter Goppakoder.

Undgå visse uheldige valg af Goppapolynomier.

Endnu ej brudt.

Andre forslag af koder er brudt.

Det diskret logaritme problem

$\mathbf{F}_{11} = \{0, 1, 2, \ldots, 10\}$. Regneregel: $11 = 0$.
$\mathbf{F}_{11}^* = \{1, 2, \ldots, 10\}$.

$2^1 = 2$, $2^2 = 4$, $2^3 = 8$, $2^4 = 5$, $2^5 = 10$, $2^6 = 9$, $2^7 = 7$, $2^8 = 3$, $2^9 = 6$, $2^{10} = 1$.

$3^1 = 3$, $3^2 = 9$, $3^3 = 5$, $3^4 = 4$, $3^5 = 1$, $3^6 = 3$, $3^7 = 9$, $3^8 = 5$, $3^9 = 4$, $3^{10} = 1$.

2 kaldes primitivt element fordi det genererer hele $\mathbf{F}_{11}^*$.

Det diskrete logaritme problem

$\mathbf{F}_{11} = \{0, 1, 2, \ldots, 10\}$. Regneregel: $11 = 0$.

$2^1 = 2$, $2^2 = 4$, $2^3 = 5$, $2^4 = 5$, $2^5 = 10$, $2^6 = 9$, $2^7 = 7$, $2^8 = 3$, $2^9 = 6$, $2^{10} = 1$.

Spørgsmål: Det oplyses, at 2 er primitivt element i $\mathbf{F}_{11}$. Givet $x \in \mathbf{F}_{11}$, hvilket b opfylder $2^b = x$?

For store legemer et svært problem.

Det diskrete logaritme problem

$\mathbf{F}_8 = \{0, 1, \alpha, \alpha + 1, \alpha^2, \alpha^2 + 1, \alpha^2 + \alpha, \alpha^2 + \alpha + 1\}$.
Regneregler: $2 = 0$, $\alpha^3 = \alpha^2 + 1$

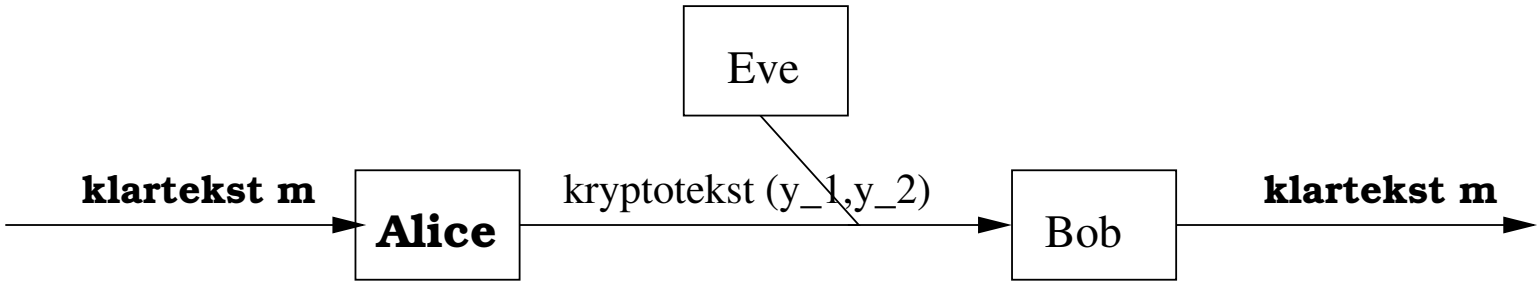
$\mathbf{F}_8^* = \{1, \alpha, \alpha + 1, \alpha^2, \alpha^2 + 1, \alpha^2 + \alpha, \alpha^2 + \alpha + 1\}$.

$\alpha^1 = \alpha$, $\alpha^2 = \alpha^2$, $\alpha^3 = \alpha^2 + 1$, $\alpha^4 = \alpha^2 + \alpha + 1$,
$\alpha^5 = \alpha + 1$,$\alpha^6 = \alpha^2 + 1$, $\alpha^7 = 1$.

Spørgsmål: Det oplyses, at α er et primitivt element i $\mathbf{F}_8$. Givet $x \in \mathbf{F}_8$, hvilket b opfylder $\alpha^b = x$?

Eksempel: $x = \alpha^2 + \alpha + 1$, så $b = 4$

ElGamal kryptosystem



Offentlig nøgle: $\{\mathbf{F}_q$, primitivt element $\alpha, \beta = \alpha^a\}$

Hemmelig nøgle: $\{a\}$, så $1 \leq a < q - 1$.

Enkryptering af besked m: Alice vælger tilfældigt k, så $k \in \mathbf{F}_q^*$.
Udregner $y_1 = \alpha^k$ og $y_2 = m\beta^k$.

Dekryptering: Bob udregner $m = y_2 \left(y_1^a\right)^{-1}$.

ElGamal kryptosystem

Kryptosystemet virker fordi diskret logaritme problem er svært og fordi:

$$\begin{aligned} y_2 (y_1^a)^{-1} &= (m\beta^k)\left((\alpha^k)^a\right)^{-1} \\ &= m\beta^k\left((\alpha^a)^k\right)^{-1} \\ &= m\beta^k(\beta^k)^{-1} = m. \end{aligned}$$

Eksponentiering er hurtigt (smart algoritme haves)

ElGamal - et eksempel

Offentlig nøgle: $\{\mathbf{F}_{11}, \alpha = 2, \beta = 2^4 = 5\}$.

Hemmelig nøgle: $a = 4$.

Alice enkrypterer besked $m = 7$ som følger:
Vælger tilfældigt $k = 8$. Udregner $y_1 = \alpha^k = 2^8 = 3$.
$y_2 = m\beta^k = 7 \cdot 5^8 = 6$.

Bob dekrypterer:
$m = y_2 (y_1)^{-1} = 6(3^4)^{-1} = 6 \cdot 4^{-1} = 6 \cdot 3 = 18 = 7$.

Genereliseringer af ElGamal

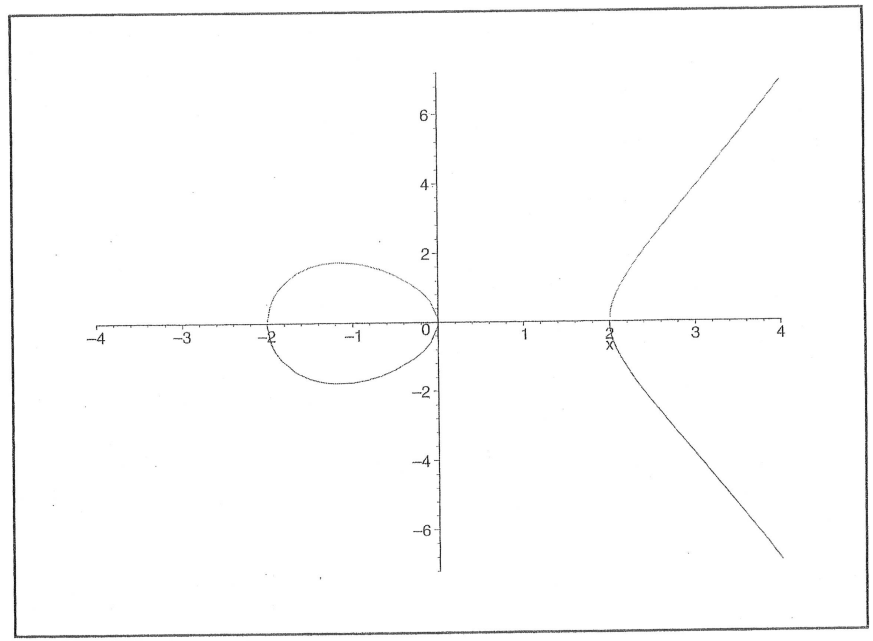
ElGamal:
Gruppe (struktur) er $(\mathbf{F}_q^*, \cdot)$

Elliptisk kurve kryptografi:
Gruppe (struktur) er $(E, +)$, hvor E.....

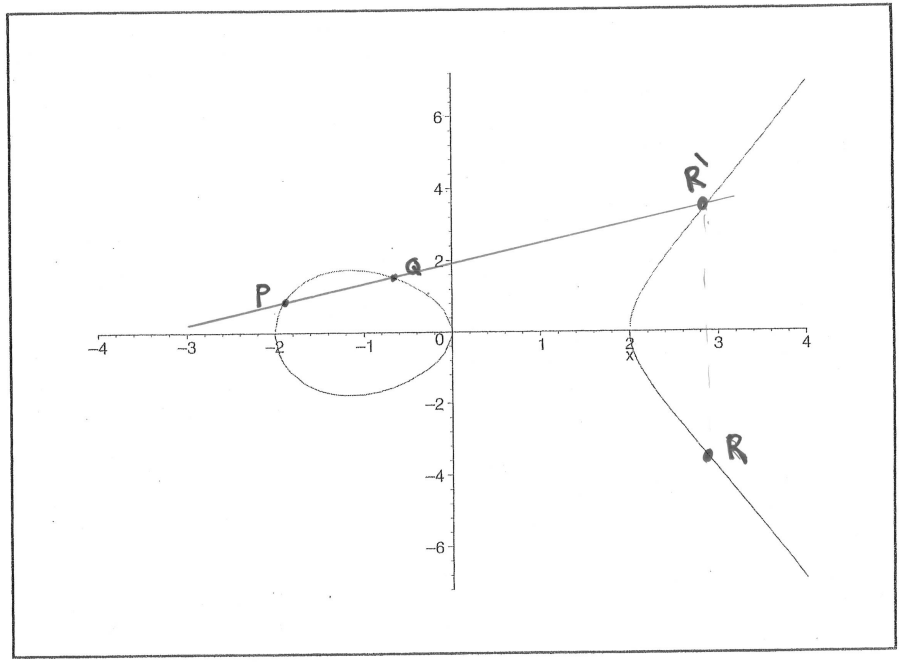
Motiverende eksempel

Ligning: $Y^2 = X^3 - 4X$.

Den elliptiske kurve E består af $(x, y) \in \mathbf{R} \times \mathbf{R}$ som løsning til ligning. Herforuden indeholder E et (imaginært) element kaldet $\mathcal{O}$.



Motiverende eksempel - fortsat



- $P + \mathcal{O} = P = \mathcal{O} + P$
- $(x, y) + (x, -y) = \mathcal{O}$
- $P + Q = R$ som på figur

Elliptisk kurve over $\mathbf{F}_q$

Specielt nemt for $\mathbf{F}_p$ med $p > 2$.
Ligning: $Y^2 = X^3 + aX + b$.

Samme "formler som før":

- $P + \mathcal{O} = P = \mathcal{O} + P$
- $P = (x_1, y_1)$, $Q = (x_2, y_2)$. $P + Q = (x_3, y_3)$ hvor
  $x_3 = \lambda^2 - x_1 - x_2$
  $y_3 = \lambda(x_1 - x_3) - y_1$
  $$\lambda = \begin{cases} (y_2 - y_1)(x_2 - x_1)^{-1} & \text{hvis } P \neq Q \\ (3x_1^2 + a)(2y_1)^{-1} & \text{hvis } P = Q \end{cases}$$

Elliptisk kurve kryptografi

Ikke sikkert, at E har primitivt element, men så har delgruppe af E.

Kryptosystem a'la ElGamal.

Næste generation af systemer benytter hyperelliptiske kurver

Kvantecomputere ej lavet endnu.

McEliece kryptosystem og elliptisk kurve kryptosystem synes MINDST USIKRE overfor kvantecomputerangreb.

Sikker kommunikation mulig ved brug af kvantekryptografi. Maskine bygget af NEC, Japan. Benytter optisk kabel (16 km).