Univerzita Karlova v Praze

Matematicko-fyzikální fakulta

# DIPLOMOVÁ PRÁCE

Bc. Jaroslav Kotil

# Goppa kódy a jejich aplikace

Katedra algebry

| | |
|---|---|
| Vedoucí diplomové práce: | prof. RNDr. Aleš Drápal, CSc., DSc. |
| Studijní program: | Matematika |
| Studijní obor: | Matematické metody informační bezpečnosti |

Praha 2013

Rád bych poděkoval prof. RNDr. Alešovi Drápalovi, CSc., DSc. za trpělivost, cenné připomínky a podněty, poskytnutí potřebné literatury a veškeré další pomoci při vzniku této diplomové práce. Dále bych chtěl také poděkovat svým rodičům, blízkým a kolegům za motivaci a podporu v době mých univerzitních studií a během psaní této práce.

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V Praze dne ..... Bc. Jaroslav Kotil

Název práce: Goppa kódy a jejich aplikace

Autor: Bc. Jaroslav Kotil

Katedra: Katedra algebry

Vedoucí diplomové práce: prof. RNDr. Aleš Drápal, CSc., DSc.

Abstrakt: V této diplomové práci představíme Goppa kódy, popíšeme jejich parametry a poté je zařadíme mezi Alternantní kódy, tedy residuální Reed–Solomonovy kódy, a Algebraicko–geometrické kódy. Dále předvedeme dekódování Goppa kódů a jejich variantu: Divoké Goppa kódy. V další části práce se budeme věnovat zástupci post–kvantové kryptografie: McEliecovu kryptosystému, proti kterému není znám žádný efektivní útok pomocí kvantových počítačů, a následně nastíníme jeho použití spolu s Goppa kódy. McEliecův kryptosystém bude také popsán z hlediska bezpečnosti a možných útoků, z nichž nejefektivnější jsou ty založené na principu dekódování s informační množinou.

Klíčová slova: Goppa kódy, Zobecněné Reed–Solomonovy kódy, Algebraicko–geometrické kódy, Post–kvantová kryptografie, McEliecův kryptosystém

Title: Goppa codes and their applications

Author: Bc. Jaroslav Kotil

Department: Department of algebra

Supervisor: prof. RNDr. Aleš Drápal, CSc., DSc.

Abstract: In this diploma paper we introduce Goppa codes, describe their parametres and inclusion in Alternant codes, which are residual Generalized Reed–Solomon codes, and Algebraic–geometry codes. Aftewards we demonstrate decoding of Goppa codes and introduce Wild Goppa codes. We also describe post–quantum cryptography member: McEliece cryptosystem for which no effective attacks with quantum computers are known. We outline a usage of this cryptosystem with Goppa codes and describe the security of the cryptosystem together with possible attacks of which the most effective ones are based on information–set decoding.

Keywords: Goppa codes, Generalized Reed–Solomon codes, Algebraic–geometry codes, Post–quantum cryptography, McEliece cryptosystem

# Obsah

**Úvod** **2**

**1 Základní poznatky** **4**
1.1 Složitost a třída $\mathcal{NP}$ . . . . . 4
1.2 Samoopravné kódy . . . . . 6
1.2.1 Základní definice a pojmy z teorie kódů . . . . . 6
1.2.2 Problém dekódování . . . . . 9
1.2.3 Zobecněné Reed–Solomonovy kódy . . . . . 12
1.2.4 Alternantní kódy . . . . . 14
1.2.5 Algebraicko–geometrické kódy . . . . . 15
1.3 Kryptografie . . . . . 21
1.3.1 Post–kvantová kryptografie . . . . . 22

**2 Goppa kódy** **24**
2.1 Klasická konstrukce . . . . . 24
2.2 Zařazení Goppa kódů . . . . . 29
2.2.1 Goppa kódy jako alternantní kódy . . . . . 29
2.2.2 Goppa kódy jako Algebraicko–geometrické kódy . . . . . 30
2.3 Parametry . . . . . 31
2.4 Divoké Goppa kódy . . . . . 33
2.5 Dekódování . . . . . 36

**3 McEliecův kryptosystém** **42**
3.1 Popis kryptosystému . . . . . 42
3.2 Niederreiterův kryptosystém . . . . . 44
3.3 Bezpečnost . . . . . 47
3.3.1 Dekódovací útoky . . . . . 48
3.3.2 Strukturální útoky . . . . . 50
3.3.3 Protokolové útoky . . . . . 53
3.4 Praktické aspekty kryptosystému . . . . . 54

**Závěr** **57**

**Literatura** **58**

# Úvod

V současné době došlo k obrovskému rozmachu informačních technologií. Příchod Internetu umožnil lidem přístup k informacím v takovém rozsahu jako nikdy před tím. K tomuto informačnímu kanálu je připojováno čím dál tím více zařízení. Kromě klasických počítačů a notebooků je možné se k Internetu připojit prakticky na každém kroku z mobilních telefonů přes Wi–Fi či $3G$ sítě. S rozvojem informačních systémů roste potřeba integrity dat. V informačních kanálech, přes které jsou zasílána data, může totiž docházet k chybovosti. Teorie samoopravných kódů studuje efektivní způsoby přenosu dat z jednoho místa na druhé. Tyto kódy umožňují detekci a opravy některých chyb vzniklých při přenosu. Kromě integrity je v praxi potřeba také zabezpečit soukromí. Pomocí kryptografie chráníme přenos zpráv tak, aby s výjimkou zúčastněných stran neměl k textu zprávy přístup nikdo jiný. Z tohoto důvodu jsou jak teorie samoopravných kódů, tak i kryptografie pro naši "informační dobu" velmi důležité.

V roce 1981 předvedl Valerij Denisovič Goppa na algebraickém semináři na Moskevské státní univerzitě způsob, jak lze propojit algebraickou geometrii a samoopravné kódy. Obsahem jeho vystoupení byla konstrukce určitých lineárních kódů odvozených z algebraických křivek nad konečným tělesem. Proto jsou v některých publikacích na jeho počest některé kódy vzniklé pomocí algebraické geometrie nazývány *geometrické Goppa kódy* nebo jen *Goppa kódy*. V dnešní době však používáme název *geometrické kódy*. Goppa kódy, v některých publikacích *klasické Goppa kódy*, pak značí určitou podtřídu alternantních kódů. Tuto podtřídu v následujícím textu budeme popisovat. Klasické jsou označovány proto, že odpovídají základní konstrukci geometrických kódů pro algebraická těleso rodu 0, které potom Goppa zobecnil pro variety vyššího rodu.

Abychom mohli Goppa kódy a jejich aplikace náležitě popsat, je důležité pochopit základní principy teorie složitosti, samoopravných kódů a kryptografie. První kapitola obsahuje úvod do těchto oblastí. Kromě obecných vlastností lineárních kódů popíšeme také problém dekódování kódů, představíme Zobecněné Reed–Solomonovy kódy, alternantní kódy a Algebraicko–geometrické kódy.

Druhá kapitola bude věnována Goppa kódům. Představíme jejich konstrukci pomocí lokalizace, popíšeme podobu prověrkové matice a ukážeme, že Goppa kódy lze také zkonstruovat pomocí evaluací určitých polynomů na množině. V další části budou pak Goppa kódy uvedeny v širších souvislostech. Zařadíme je mezi alternantní, respektive reziduální, kódy a ukážeme, že jsou součástí Algebraicko–geometrických kódů. Popíšeme jejich parametry a uvedeme zajímavé vlastnosti takzvaných Divokých Goppa kódů umožňující dekódování více chyb. V závěru ka-

pitoly popíšeme způsob rychlého algoritmu pro dekódování Goppa kódů pomocí Euklidova algoritmu.

V praxi používaná asymetrická kryptografie je založena na problémech z teorie čísel. Mezi nejoblíbenější patří problém faktorizace čísel a problém diskrétního logaritmu. Systémy založené na těchto problémech jsou výbornou volbou v mnoha aplikacích a jejich bezpečnost je dobře definována a chápána. Jednou z jejich hlavních nevýhod ale je, že budou zranitelné v případě dostupnosti kvantových počítačů patřičné kapacity. Velkou pozornost proto získává studium post–kvantové kryptografie popisující algoritmy použitelné na stávajících počítačích, proti nimž není znám žádný efektivní útok pomocí kvantových algoritmů.

Třetí kapitola přibližuje zástupce post–kvantové kryptografie: McEliecův kryptosystém. Tento asymetrický kryptosystém ve své konstrukci používá samoopravné kódy a při použití ireducibilních binárních Goppa kódů proti němu není znám efektivní útok. Jednou z hlavních nevýhod toto kryptosystému je velikost klíčů. Veřejný klíč totiž obsahuje zamaskovanou generující matici lineárního kódu dosahující pro bezpečné parametry stovek tisíců bytů v porovnání s 256 byty u 1024-bitového modulu RSA. V této kapitole proto zavedeme McEliecův kryptosystém, nastíníme jeho bezpečnost a popíšeme dva základní typy útoků - dekódovací útoky a strukturální útoky. Uvedeme také útoky na samotný protokol, tedy postup využívaný pro přenos samotné zprávy. V závěru popíšeme praktické aspekty kryptosystému, jako jsou expanze zprávy, velikost klíčů či volba bezpečných parametrů.

# Kapitola 1

# Základní poznatky

V této kapitole uvedeme některé informace z oblasti složitosti, teorie samoopravných kódů, algebraické geometrie a asymetrické kryptografie, které jsou potřebné pro formální zavedení Goppa kódů. To nám později rovněž umožní je zařadit do širších souvislostí. Pro pochopení následujícího textu budeme předpokládat základní znalosti algebry a kryptografie.

## 1.1 Složitost a třída $\mathcal{NP}$

Následující definice používají pojem algoritmu ve smyslu programu v nějakém základním programovacím jazyku (například C nebo Pascal). Počet kroků algoritmu je pak počet kroků v definovaném kódu programu. Tyto definice lze zavést pomocí pojmu Turringova stroje [1]. Je zřejmé, že realizace algoritmu daná programovacím jazykem je z hlediska teorie složitosti korektní jedině tehdy, pokud příkazy jazyka pracují se slovy (ve smyslu řetězců bitů), které mají ze shora omezenou délku (například 32 či 64 bitů). Délka vstupu užívaná v následujících definicích pak odkazuje na bitovou délku, případně na počet slov vstupu (kde slova mají délku právě 32 nebo 64 bitů).

Nechť funkce $f$ a $g$ jsou definované na kladných číslech. Řekneme, že

$$f \in \mathcal{O}(g),$$

pokud existuje kladná konstanta $k$ a $x_0 > 0$ takové, že

$$\forall x > x_0,\ f(x) \leq kg(x).$$

Řekneme, že algoritmus je *polynomiálně náročný*, respektive *má polynomiální složitost*, pokud ke svému ukončení potřebuje nejvýše $f(x)$ kroků, kde $f(x)$ je polynom určující počet kroků algoritmu na vstupních datech o velikosti $x$.

**Definice 1.1** (Třída $\mathcal{P}$)**.** *Třída $\mathcal{P}$ je množina výpočetních problémů řešitelných algoritmem v počtu kroků ohraničených polynomem závislým na délce vstupu. Třída $\mathcal{P}$ tedy odpovídá množině* polynomiálně složitých *algoritmů.*

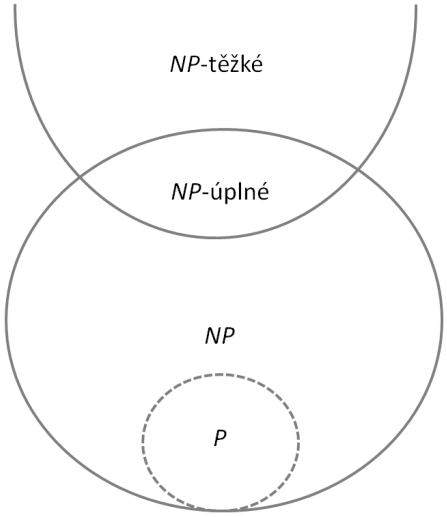
**Definice 1.2** (Třída $\mathcal{NP}$ problémů)**.** *Třída $\mathcal{NP}$ je množina výpočetních problémů, pro které existuje algoritmus schopný ověřit správnost výsledku pro daný vstup v polynomiálním čase.*

Alternativně lze říci, že třída $\mathcal{NP}$ je množina výpočetních problémů řešitelných nedeterministickým algoritmem v počtu kroků ohraničených polynomem závislým na délce vstupu. Nedeterministický algoritmus je takový algoritmus, který může v případě volby mezi dvěma alternativami pokračovat v obou zároveň. Takovéto větvení při volbě možnosti u deterministického algoritmu exponenciálně zvětšuje počet kroků algoritmu.

**Poznámka 1.1.** *Velkou neznámou teorie složitosti je odpověď na otázku, zda platí nebo neplatí $\mathcal{P} = \mathcal{NP}$.*

**Definice 1.3** (Třída $\mathcal{NP}$ těžkých problémů)**.** *Třída $\mathcal{NP}$ těžkých problémů je množina výpočetních problémů, které jsou alespoň tak náročné, jako je náročný nejtěžší problém v $\mathcal{NP}$.*

**Definice 1.4** (Třída $\mathcal{NP}$ úplných problémů)**.** *Třída $\mathcal{NP}$ úplných problémů je množina výpočetních problémů takových, že na každý takový problém lze zredukovat všechny ostatní $\mathcal{NP}$ problémy.*




Obrázek 1.1: Vztahy mezi vybranými třídami problémů.

**Příklad 1.1.** *Uvažme následující problém:*

*Je možné pro daný konečný graf $G = (V, E)$ obarvit vrcholy grafu třemi různými barvami tak, aby žádné dva sousední neměly stejnou barvu?*

*Tento problém se nachází ve třídě $\mathcal{NP}$ úplných problémů.*

Rozdělení složitosti na tyto třídy využijeme k porovnání obtížnosti problémů.

**Poznámka 1.2.** *Konkretní zadání (otázku) problému nazveme instancí. Problém potom můžeme chápat jako množinu všech jeho instancí.*

**Definice 1.5** (Redukce problémů)**.** *Řekneme, že problém $P_1$ je* polynomiálně převeditelný *(jinými slovy,* lze ho redukovat*) na problém $P_2$, pokud existuje polynomiální algoritmus $f$, který pro libovolnou instanci $x$ problému $P_1$ sestrojí instanci problému $f(x) \in P_2$ tak, že platí:*

$$x \in P_1 \Leftrightarrow f(x) \in P_2.$$

Základní pojmy v textu nedefinované lze nalézt v [1].

## 1.2 Samoopravné kódy

Teorie samoopravných kódů se zabývá efektivním a spolehlivým přenosem informací z jednoho místa na druhé. Mezi její aplikace patří například minimalizace šumu při přehrávání audio kompaktů či odstraňování chyb vzniklých při přenosu dat, a to na internetu, při digitálním televizním vysílání, při telefonním přenosu apod. Hlavní myšlenka spočívá v tom, že data převedeme do struktury s nadbytečnými informacemi, které nám potom umožní detekovat a opravit některé chyby. Pro převedení dat do takové struktury slouží *kódovací algoritmus* a k rekonstrukci původních dat pak *dekódovací algoritmus*.

V první části této sekce uvedeme základní fakta o lineárních samoopravných kódech a jejich vlastnostech. Další části zaměříme na Zobecněné Reed–Solomonovy kódy a z nich odvozené Alternantní kódy. Poté přejdeme k Algebraicko–geometrickým kódům. V poslední části nastíníme problematiku obecného dekódování lineárních kódů.

Rozšiřující znalosti o samoopravných kódech lze získat v [2].

### 1.2.1 Základní definice a pojmy z teorie kódů

V kapitole 1.2 budeme používat značení $q = p^m$, kde $p$ je prvočíslo a $m \in \mathbb{N}$.

**Definice 1.6.** *Nechť $\mathbb{F}_q$ je konečné těleso. Potom* lineární kód *$C$ délky $n$ a dimenze $k \geq 0$ je libovolný $k$–dimenzionální podprostor vektorového prostoru $\mathbb{F}_q^n$.*

Konečné těleso $\mathbb{F}_q$ kódu $C$ chápeme jako abecedu, nad kterou je kód vybudován. *Velikostí* kódu $C$ míníme počet jeho prvků, tedy $|C|$. Prvky $\mathbb{F}_q^n$ nazýváme *slova* a prvky kódu $C$ pak *kódová slova*.

**Definice 1.7.** Generující matice *lineárního kódu $C$ je taková $k \times n$ matice $G$, pro kterou platí*

$$C = \{xG : x \in \mathbb{F}_q^k\}.$$

*Prvek $x$ nazýváme zdrojové slovo. Řekneme, že generující matice $G$ je* ve standardním tvaru*, pokud*

$$G = (I_{k\times k} \,|\, A_{k\times(n-k)}),$$

*kde $I_{k\times k}$ značí $k \times k$ jednotkovou diagonální matici a $A_{k\times(n-k)}$ matici velikosti $k \times (n-k)$.*

Matici $G$ tedy můžeme ztotožnit se zobrazením $\mathbb{F}_q^k \to \mathbb{F}_q^n$, $x \mapsto xG$.

**Definice 1.8.** Prověrková (kontrolní) matice *$H$ je taková $(n-k) \times n$ matice, pro kterou platí*

$$C = \{c \in \mathbb{F}_q^n : Hc^\top = 0\}.$$

**Definice 1.9.** Hammingovu vzdálenost $\mathrm{d}(x, y)$ *slov* $x, y \in \mathbb{F}_q^n$ *definujeme jako počet souřadnic, v kterých se* $x$ *a* $y$ *liší, tedy*

$$\mathrm{d}(x, y) = |\{1 \leq i \leq n \,|\, x_i \neq y_i\}|.$$

Hammingova vzdálenost je metrika a je invariantní vůči posunutí [2]. To znamená, že platí

$$\mathrm{d}(x, y) = \mathrm{d}(x + z, y + z) \, \forall x, y, z \in C.$$

Speciálně platí $\mathrm{d}(x, y) = \mathrm{d}(x - y, y - y) = \mathrm{d}(x - y, 0)$.

**Příklad 1.2.** *Nechť* $x = (0, \alpha, 0, 1, \alpha, 0)$, $y = (1, \alpha, \alpha, 1, 1, 0) \in \mathbb{F}_8^6$ *jsou dvě slova. Potom Hammingova vzdálenost mezi* $x$ *a* $y$ *je rovna:*

$$\mathrm{d}((0, \alpha, 0, 1, \alpha, 0), (1, \alpha, \alpha, 1, 1, 0)) = 3$$

**Definice 1.10.** Hammingovu váhu $\mathrm{wt}(x)$ *slova* $x$ *definujeme jako počet nenulových souřadnic, tedy*

$$\mathrm{wt}(x) = \mathrm{d}(0, x),$$

*kde* $0$ *značí nulový vektor.*

**Definice 1.11.** Minimální vzdálenost *lineárního kódu* $C$ *je definována vztahem*

$$d = \min_{\substack{x, y \in C \\ y \neq x}} \mathrm{d}(x, y).$$

**Poznámka 1.3.** *Jelikož pro lineární kódy platí* $x - y \in C$, *kdykoliv* $x, y \in C$, *dostáváme*

$$d = \min_{\substack{x \in C \\ x \neq 0}} \mathrm{d}(x, 0) = \min_{\substack{x \in C \\ x \neq 0}} \mathrm{wt}(x).$$

Parametry lineárního kódu $C$ nad tělesem $\mathbb{F}_q$ značíme $[n, k, d]_q$, kde

- $n$ je jeho délka,
- $k$ je jeho dimenze,
- $d$ je jeho minimální vzdálenost.

Lze použít také zkrácený tvar $[n, k]_q$.

Kódy nad $\mathbb{F}_2$ nazýváme *binární*. Obecně se kód nad abecedou $\mathbb{F}_q$ nazývá $q$-ární kód. *Hustotou* kódu $C$ nazveme podíl $R = \frac{k}{n}$. Hustota kódu udává poměr délky zdrojového a kódového slova. Tudíž $1/R$ udává faktor, o který se zdrojové slovo prodlouží vlivem kódování.

**Příklad 1.3.** *Binární* $[4, 2, 3]$ *kód*

$$C = \{(0, 0, 0, 0), (1, 1, 0, 1), (0, 1, 1, 0), (1, 0, 1, 1)\}$$

*můžeme zkonstruovat pomocí generující matice*

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

*nebo pomocí prověrkové matice*

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

Při práci s lineárními kódy má zásadní význam *bodový součin* $u \cdot v = \sum_{i=1}^{n} u_i v_i$ pro vektory $u = (u_1, \ldots, u_n)$, $v = (v_1, \ldots, v_n) \in \mathbb{F}_q$. Je–li $C$ $[n,k]_q$ kód, pak $C^\perp = \{u \in \mathbb{F}_q^n \mid u \cdot c = 0, \forall c \in C\}$ je $[n, n-k]_q$ kód. $C^\perp$ nazýváme *duálním kódem* ke kódu $C$. Pro takové kódy platí vztahy $\dim C + \dim C^\perp = n$ a $(C^\perp)^\perp = C$. Následující tvrzení popisují vlastnosti duálních kódů.

**Tvrzení 1.12.** *Generující matice duálního kódu $C^\perp$ se shodují s prověrkovými maticemi kódu $C$.*

*Důkaz.* Důkaz lze nalézt v [2]. □

Řádkové vektory prověrkové matice jsou lineárně nezávislé. Z funkčního pohledu může kód $C$ definovat i matice $H'$, velikosti $l \times n$, generující stejný podprostor jako $H$ daný vztahem $Hc^\top = 0$, $\forall c \in C$, kde $l \geq n - k$. Řádky $H'$ již potom nejsou lineárně nezávislé. Matici $H'$ nazýváme *testovací maticí* kódu $C$.

Pokud je generující matice $G$ ve standardním tvaru, tedy

$$G = (I_{k\times k} \mid A_{k\times(n-k)}),$$

potom má prověrková matice tvar

$$H = (-A^\top_{(n-k)\times k} \mid I_{(n-k)\times(n-k)}).$$

Podobně lze přejít od matice prověrkové k matici generující. Pomocí tohoto speciálního tvaru generující a prověrkové matice je získání prověrkové matice, pokud známe tu generující a naopak, polynomiálně náročné [3].

**Definice 1.13.** *Nechť $\sigma$ je permutace množiny $\{1, \ldots, n\}$. Potom definujeme permutovaný kód $C^\sigma$ jako kód $C$ s pozicemi permutovanými podle $\sigma$, tedy*

$$C^\sigma = \{x \in \mathbb{F}_q^n \mid (x_{\sigma^{-1}(1)}, \ldots, x_{\sigma^{-1}(n)}) \in C\}.$$

Pro dvě permutace $\sigma$ a $\epsilon$ platí $(C^\sigma)^\epsilon = C^{\epsilon\sigma}$. Pokud $C'$ a $C$ jsou $[n,k]_q$ kódy takové, že $C' = C^\sigma$ pro nějakou permutaci $\sigma$, potom řekneme, že kódy $C'$ a $C$ jsou *permutačně ekvivalentní*. Pokud je $\sigma$ permutace taková, že $C^\sigma = C$, potom $\sigma$ patří do grupy automorfismů zobrazující $C$ na $C$.

**Tvrzení 1.14.** *Nechť $C$ je $[n,k]_q$ lineární kód. Potom $C$ je permutačně ekvivalentní kódu s generující maticí ve standardním tvaru.*

*Důkaz.* Protože má kód dimenzi $k$, můžeme pomocí elementárních operací upravit generující matici tak, aby na $k$ nenulových sloupcích měla vždy jen jednu 1 a 0 jinde. Poté stačí zvolit permutaci kódu tak, aby nová generující matice byla tvaru $(I_{k\times k} \mid A_{k\times(n-k)})$. □

Jeden z problémů teorie kódů je sestrojit nad abecedou $\mathbb{F}_q$ kódy, jejichž dimenze a minimální vzdálenosti jsou velké v porovnání s jejich délkou. Narážíme zde na jistá omezení. Jedno z nich prezentuje následující věta.

**Věta 1.15** (Singletonův odhad)**.** *Pro daný $[n,k,d]_q$ kód $C$ platí následující nerovnost:*

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

*Důkaz.* Uvažme lineární podprostor $E \subseteq \mathbb{F}_q^n$ daný

$$E = \{(a_1, \ldots, a_n) \in \mathbb{F}_q^n \mid a_i = 0, \forall i \geq d\}.$$

Potom každý prvek $a \in E$ má váhu $\leq d-1$, tedy $E \cap C = \emptyset$. Jelikož $\dim E = d-1$, dostáváme

$$\begin{aligned} k + (d-1) &= \dim C + \dim E \\ &= \dim(C+E) + \dim(C \cap E) = \dim(C+E) \leq n. \end{aligned}$$

□

Kódy s rovností $k + d = n + 1$ jsou v předchozím významu optimální a nazývají se MDS (Maximum Distance Separable).

**Tvrzení 1.16.** *Nechť $C$ je $[n, k, d]_q$ kód. Ať $G$ je jeho generující matice a nechť $H$ je jeho prověrková matice. Potom jsou následující tvrzení ekvivalentní:*

- *$C$ je MDS kód,*
- *každá množina $n - k$ sloupců $H$ je lineárně nezávislá,*
- *každá množina $k$ sloupců $G$ je lineárně nezávislá,*
- *$C^\perp$ je MDS kód.*

*Důkaz.* Důkaz lze nalézt v [2]. □

### 1.2.2 Problém dekódování

Budeme popisovat komunikační systém, který bude přenášet zprávy přes zařízení nazvané *kanál*. Na jednom konci kanálu je odesílatel zprávy

$$m = (m_1, \ldots, m_k) \in \mathbb{F}_q^k$$

a na druhém konci je příjemce této zprávy. Přes kanál budeme posílat prvky tělesa $\mathbb{F}_q$. Při přenosu těchto prvků může dojít k chybám – příjemce může obdržet jiný prvek, než který byl odeslán. Budeme předpokládat, že chyby přenášených prvků jsou na sobě nezávislé. Pro zlepšení kvality komunikace přes kanál budeme využívat samoopravné kódy. Zprávu $m$ nejprve zakódujeme a po přenosu dekódujeme. Využijeme vlastnosti samoopravných kódů, která nám umožňuje opravit některé vzniklé chyby. Schéma komunikačního kanálu zobrazuje obrázek 1.2.

**Definice 1.17.** *Pro $[n, k]_q$ kód $C$ definujeme* kódovací algoritmus *jako zobrazení*

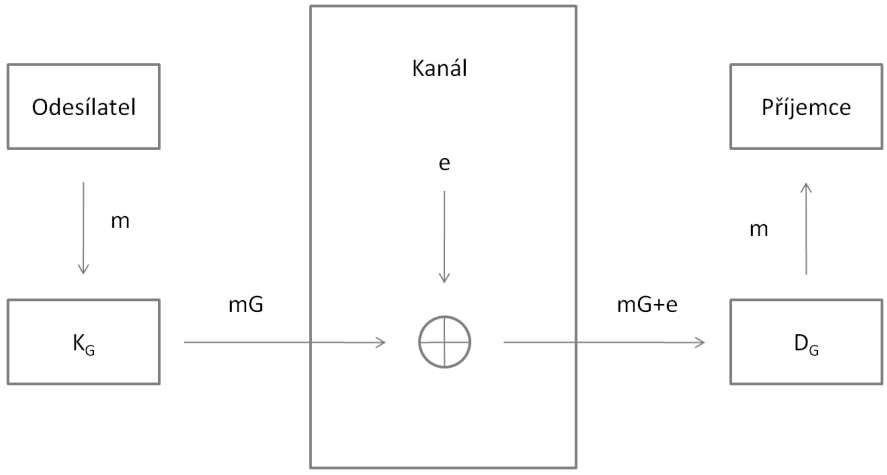
$$\begin{aligned} K_G : \quad & \mathbb{F}_q^k \to \mathbb{F}_q^n \\ & m \mapsto mG \end{aligned}$$

*a* dekódovací algoritmus *potom jako zobrazení*

$$D_C : \quad \mathbb{F}_q^n \to \mathbb{F}_q^k,$$

*pro které platí $mG \mapsto m$, $\forall m \in \mathbb{F}_q^k$. Vektor $e \in \mathbb{F}_q^n$ nazveme* chybový*, pokud je $m' = mG + e$ přijatá zpráva. Řekneme, že dekódovací algoritmus $D_C$ opravuje $t$ chyb právě tehdy, když navíc platí*

$$\forall e \in \mathbb{F}_q^n \text{ a } \forall m \in \mathbb{F}_q^k : \operatorname{wt}(e) \leq t \Rightarrow D_C(mG + e) = m.$$




Obrázek 1.2: Komunikace pomocí samoopravných kódů.

Nechť $C$ je lineární kód a $G$ jeho generující matice. Nechť $m'$ je slovo, pro které platí $m' = mG + e$, $m \in \mathbb{F}_q^k$, $e \in \mathbb{F}_q^n$ váhy $\mathrm{wt}(e) = t$. Čím větší je minimální vzdálenost kódu, tím více chyb může kód opravit. Předpokládejme, že kód má minimální vzdálenost $d$, potom můžeme opravit maximálně $t = \lfloor \frac{d-1}{2} \rfloor$ chyb. Nemůže jich být více, protože jinak by nebylo dekódování jednoznačné. Měli bychom totiž od slova $mG + e$ ve vzdálenosti alespoň $t \geq \frac{d}{2}$ dvě a více kódových slov. Přijaté slovo tedy vždy dekódujeme na nejbližší kódové slovo vzhledem k Hammingově vzdálenosti. Takovéto dekódování se nazývá *dekódování na nejpodobnější slovo* (Maximum Likelihood Decoding). Pro potřeby této práce budeme předpokládat, že kanál může způsobit maximálně $t$ chyb, kde $t$ je počet chyb, které použitý samoopravný kód je schopen opravit.

**Definice 1.18.** Syndrom *slova $y$ definujeme jako* $\mathrm{Syn}(y) = Hy^\top$.

Syndrom je sloupcový vektor délky $n - k$. Pokud je přijaté slovo $y = c + e$, kde $c \in C$, potom platí $\mathrm{Syn}(y) = H(c+e)^\top = He^\top$. Definujme zobrazení $\mathrm{S}_H$ vzhledem k prověrkové matici $H$:

$$\begin{aligned} \mathrm{S}_H : \quad & \mathbb{F}_q^k \to \mathbb{F}_q^{n-k} \\ & y \mapsto (Hy^\top)^\top \end{aligned}$$

a označme $\mathrm{S}_H^{-1}(s)$ množinu slov, které mají syndrom $s^\top$:

$$\mathrm{S}_H^{-1}(s) = \{y \in \mathbb{F}_q^n \,|\, (Hy^\top)^\top = s\}.$$

**Definice 1.19.** *Ať $C$ je $[n, k]_q$ kód a ať $a \in \mathbb{F}_q^n$. Rozkladovou třídou obsahující $a$ rozumíme množinu $a + C = \{a + c \,|\, c \in C\}$.*

**Lemma 1.20.** *Dvě slova $x$ a $y$ jsou ve stejné rozkladové třídě modulo $C$ právě tehdy, když mají stejný syndrom.*

*Důkaz.* Dvě slova $x$ a $y$ leží ve stejné rozkladové třídě, pokud existují kódová slova $c_1$ a $c_2$ taková, že platí $x = a + c_1$ a $y = a + c_2$. Pro $x$ a $y$ potom máme $Hx^\top = H(a+c_1)^\top = Ha^\top = H(a+c_2)^\top = Hy^\top$, takže $x$ a $y$ mají stejný syndrom. Naopak pokud mají dvě slova stejný syndrom, tedy $Hx^\top = Hy^\top$, potom platí $H(x-y)^\top = 0$. Tedy $x-y$ je kódové slovo, a proto $x$ a $y$ jsou ve stejné rozkladové třídě. $\square$

**Poznámka 1.4.** *Rozkladové třídy $a + C$, $a \in \mathbb{F}_q^n$, jsou afinními podprostory $\mathbb{F}_q^n$.*

**Důsledek 1.21.** *Pro všechna $s \in \mathbb{F}_q^{n-k}$ máme $\mathrm{S}_H^{-1}(s) = y + C = \{y + c \mid c \in C\}$, kde $y \in \mathbb{F}_q^n$ a $\mathrm{Syn}(y) = s^\top$.*

**Věta 1.22.** *Nechť $H$ je prověrková matice $[n, k]_q$ kódu a $s \in \mathbb{F}_q^{n-k}$ syndrom. Potom nalezení slova $y \in \mathbb{F}_q^n$ splňující rovnici $Hy^\top = s$ lze dosáhnout v polynomiálním čase.*

*Důkaz.* Bez újmy na obecnosti můžeme uvažovat prověrkovou matici $H_0$ tvaru $(I \mid X)$, kde $I$ je $(n-k) \times (n-k)$ jednotková matice a $X$ je $(n-k) \times k$ matice taková, že platí $H_0 = UH$, kde $U$ je $(n-k) \times (n-k)$ invertibilní matice. Matici $U$ lze nalézt polynomiálním algoritmem se složitostí $O((n-k)^3)$ pomocí invertování prvních $n-k$ sloupců $H$, [3]. Zvolme $y = (Us \mid 0) \in \mathbb{F}_q^n$, potom máme

$$Hy^\top = U^{-1}H_0y^\top = U^{-1}\begin{pmatrix} I & \Big| & X \end{pmatrix}\begin{pmatrix} Us \\ 0 \end{pmatrix} = s.$$

Jelikož násobení a sčítání jsou polynomiální operace, získáváme $y$ v polynomiálním čase. □

**Věta 1.23.** [Canteaut a Chabaud]
*Dekódování lineárního kódu na nejpodobnější slovo lze redukovat na nalezení kódového slova s minimální Hammingovou váhou.*

*Důkaz.* Nechť $c \in C$, $0 \neq e \in \mathbb{F}_q^n$, $\mathrm{wt}(e) < \frac{d}{2}$ a $y = c + e$ je slovo. Zkonstruujme lineární kód $C' = \langle C, y \rangle = y + C$ generovaný kódem $C$ a slovem $y$, tedy kód s generující maticí

$$\begin{pmatrix} G \\ y \end{pmatrix},$$

kde $G$ je generující matice kódu $C$. Platí $\mathrm{d}(y, c) = \mathrm{wt}(e) < \frac{d}{2}$. Proto z definice minimální vzdálenosti dostáváme $\forall c' \in C'$, $c' \neq 0$, $\mathrm{d}(y, c') \geq \frac{d}{2}$. Tedy $e = y - c$ je kódové slovo $C'$ nejmenší váhy. Jinými slovy pokud najdeme kódové slovo kódu $C'$ minimální váhy, potom můžeme dekódovat $y$. □

**Problém 1.1.** *Ať $C$ je lineární kód a $H$ jeho prověrková matice. Nechť je dáno slovo $y \in \mathbb{F}_q^n$ a jeho syndrom $s = Hy^\top \in \mathbb{F}_q^{n-k}$. Potom dekódování na nejpodobnější slovo odpovídá vyřešení jednoho z ekvivalentních problémů:*

1. *Nalezněte kódové slovo $x \in C$, kde Hammingova vzdálenost $x$ a $y$ je minimální.*
2. *Nalezněte chybový vektor $e \in y + C$ minimální Hammingovy váhy.*
3. *Nalezněte chybový vektor $e \in \mathrm{S}_H^{-1}(s^\top)$ minimální Hammingovy váhy.*

Problém dekódování lineárního kódu je studován již dlouho. Nejefektivnější algoritmy na řešení tohoto problému mají exponenciální složitost a jsou založeny na algoritmu *dekódování s informační množinou* (information set decoding). Více informací o těchto algoritmech lze nalézt v sekci 3.3.1. Pro daný chybový vektor $e$ je obtížné zjistit, zda je v dané rozkladové třídě minimální váhy, nebo zda je slovo $x$ opravdu nejblíže slovu $y$. Proto nevíme, zda tyto problémy jsou ve třídě $\mathcal{NP}$. Uvažme binární variantu tohoto problému nazvanou *computional syndrome decoding*:

**Problém 1.2** (Computional Syndrome Decoding (CSD)). *Pro danou binární* $(n-k) \times n$ *prověrkovou matici* $H$, *vektor* $s \in \mathbb{F}_q^{n-k}$ *a* $w > 0$ *nalezněte slovo* $e \in \mathrm{S}_H^{-1}(s)$ *takové, že* $\mathrm{wt}(e) \leq w$.

V tomto případě již jsme schopni rozhodnout, zda nalezené slovo $e$ splňuje dané požadavky, proto tento problém patří do třídy $\mathcal{NP}$. Berlekamp, McEliece a van Tilborg v roce 1978 v [4] ukázali, že pro binární kódy je rozhodovací verze tohoto problému, tedy problém rozhodnutí, zda takové slovo $e$ existuje, $\mathcal{NP}$ úplný problém. Proto je problém 1.2 v nejhorším případě $\mathcal{NP}$ těžký. Přestože příslušnost třídě $\mathcal{NP}$ nám neříká, jaká je složitost v průměrném případě, je zřejmé, že nedostaneme jednoduchý problém při použití parametru $w = t$, kde $t$ je počet chyb, které je schopný lineární kód opravit.

### 1.2.3 Zobecněné Reed–Solomonovy kódy

Zobecněné Reed–Solomonovy kódy (GRS - Generalized Reed–Solomon codes) představují nejznámější a zřejmě nejdůležitější třídu MDS kódů. Minimální vzdálenost $d$ je tedy maximální možná vzhledem k Singletonovu odhadu. GRS kódy jsou využívány například u CD disků ke korekci chyb vzniklých poškozením kompaktu.

**Definice 1.24.** *Nechť* $n \in \mathbb{N}$ *a ať*

$$B = (\beta_1, \ldots, \beta_n) \in (\mathbb{F}_q^*)^n,\ L = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q^n,$$

*kde* $\alpha_i \neq \alpha_j$, *je-li* $i \neq j$. *Definujme homomorfismus*

$$\mathrm{ev} = \mathrm{ev}(\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n),$$

$$\mathrm{ev} : \mathbb{F}_q[x] \to \mathbb{F}_q^n,$$

*následovně*

$$\mathrm{ev} : f \mapsto (\beta_1 f(\alpha_1), \ldots, \beta_n f(\alpha_n)).$$

*Nechť* $k$ *je celé takové, že* $1 \leq k \leq n$. Zobecněný Reed–Solomonův kód *(GRS kód) nad* $\mathbb{F}_q$ *asociovaný s trojicí* $(L, B, k)$ *definujeme jako*

$$\mathrm{GRS}_q(L, B, k) = \{\mathrm{ev}(f) \mid f \in \mathbb{F}_q[x]\,,\ \deg f < k\}.$$

Homomorfismus ev je injektivní, neboť nenulový polynom $f \in \mathbb{F}_q[x]$ stupně menšího než $k \leq n$ má méně než $k$ kořenů. Zřejmě $\mathrm{GRS}_q(L, B, k)$ je lineární $[n, k]_q$ kód. Váha kódového slova

$$0 \neq c = \mathrm{ev}(f) \in \mathrm{GRS}_q(L, B, k)$$

je dána vztahem

$$\mathrm{wt}(c) = n - |\{1 \leq i \leq n;\ f(\alpha_i) = 0\}| \geq n - \deg f \geq n - (k-1),$$

a tedy díky Singletonově odhadu dostáváme $d = n - k + 1$, kde $d$ je minimální vzdálenost GRS kódu. GRS kód je tedy MDS s parametry $[n, k, n-k+1]_q$.

Z definice vidíme, že generující matice $\mathrm{GRS}_q(L, B, k)$ kódu má tvar

$$G = \begin{pmatrix} \beta_1 & \beta_2 & \dots & \beta_n \\ \beta_1\alpha_1 & \beta_2\alpha_2 & \dots & \beta_n\alpha_n \\ \beta_1\alpha_1^2 & \beta_2\alpha_2^2 & \dots & \beta_n\alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \beta_1\alpha_1^{n-k-1} & \beta_2\alpha_2^{n-k-1} & \dots & \beta_n\alpha_n^{n-k-1} \end{pmatrix}$$

**Tvrzení 1.25.** *Duální kód ke* $\mathrm{GRS}_q(L, B, k)$ *kódu je roven* $\mathrm{GRS}_q(L, B', n-k)$ *pro nějaké* $B = (\beta_1', \dots, \beta_n') \in (\mathbb{F}_q^*)^n$.

*Důkaz.* Postupujme indukcí podle $k$. Případy $k = 0$ a $k = n$ jsou triviální. Dále ukážeme případ $k = n - 1$. Vyložíme, že ostatní případy se dají na tento případ redukovat.

Nechť tedy máme $k = n - 1$ a nechť $D$ je duální kód ke kódu $\mathrm{GRS}_q(L, B, n-1)$. Potom kód $D$ díky tvrzení 1.16 existuje a má parametry $[n, 1, n]_q$. Jelikož má dimenzi 1, skládá se ze všech skalárních násobků nějakého jedinečného vektoru $B' = (\beta_1', \dots, \beta_n')$, kde $\beta_i \in \mathbb{F}_q^*$, $1 \le i \le n$. Tento kód ovšem z definice odpovídá kódu $\mathrm{GRS}_q(L, B', 1)$, tedy $\mathrm{GRS}_q(L, B, k)^\perp = \mathrm{GRS}_q(L, B', 1)$. Speciálně tedy pro každý polynom $f \in \mathbb{F}_q[x]$, $\deg(f(x)) < n - 1$ platí

$$\beta_1'\beta_1 f(\alpha_1) + \dots + \beta_n'\beta_n f(\alpha_n) = 0, \tag{1.1}$$

neboť $(\beta_1 f(\alpha_1), \dots, \beta_n f(\alpha_n)) \in \mathrm{GRS}_q(L, B, k) = \mathrm{GRS}_q(L, B', 1)^\perp$.

Nyní pro $0 < k < n - 1$ dokážeme, že platí $\mathrm{GRS}_q(L, B, k)^\perp = \mathrm{GRS}_q(L, B', n - k)$. Kódová slova $\mathrm{GRS}_q(L, B, k)$ můžeme zapsat jako $(\beta_1 f(\alpha_1), \dots, \beta_n f(\alpha_n))$, kde $f(x) \in \mathbb{F}_q[x]$ stupně $\le k - 1$, a kódová slova $\mathrm{GRS}_q(L, B', n - k)$ jsou potom $(\beta_1' g(\alpha_1), \dots, \beta_n' g(\alpha_n))$, kde $g(x) \in \mathbb{F}_q[x]$ stupně $\le n - k - 1$. Jelikož

$$\deg(f(x)g(x)) \le n - 2 < n - 1,$$

tak z rovnice 1.1 máme

$$(\beta_1 f(\alpha_1), \dots, \beta_n f(\alpha_n)) \cdot (\beta_1' g(\alpha_1), \dots, \beta_n' g(\alpha_n)) = 0.$$

Tedy $\mathrm{GRS}_q(L, B', n - k) \subseteq \mathrm{GRS}_q(L, B, k)^\perp$. Jelikož mají oba kódy stejnou dimenzi, dostáváme rovnost. □

**Důsledek 1.26.** $\mathrm{GRS}_q(L, B, k)$ *kód má prověrkovou matici*

$$H = \begin{pmatrix} \beta_1' & \beta_2' & \dots & \beta_n' \\ \beta_1'\alpha_1 & \beta_2'\alpha_2 & \dots & \beta_n'\alpha_n \\ \beta_1'\alpha_1^2 & \beta_2'\alpha_2^2 & \dots & \beta_n'\alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \beta_1'\alpha_1^{n-k-1} & \beta_2'\alpha_2^{n-k-1} & \dots & \beta_n'\alpha_n^{n-k-1} \end{pmatrix} =$$

$$= \begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \dots & \alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{n-k-1} & \alpha_2^{n-k-1} & \dots & \alpha_n^{n-k-1} \end{pmatrix} \cdot \begin{pmatrix} \beta_1' & 0 & \dots & 0 \\ 0 & \beta_2' & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \beta_n' \end{pmatrix}.$$

*Důkaz.* Důsledek získáme z tvrzení 1.12 a toho, že řádkové vektory matice $H$ generují duální kód $\mathrm{GRS}_q(L, B', n-k)$ ke kódu $\mathrm{GRS}_q(L, B, k)$. □

### 1.2.4 Alternantní kódy

Pokud pro daný lineární kód $C$ budeme uvažovat pouze kódová slova tvořená prvky nějakého pevně vybraného podtělesa, dostaneme touto restrikcí na podtěleso kódy nazývané *reziduální*. Takové kódy mají vlastnosti odvozené z původního lineárního kódu. Významnou třídou kódů jsou *alternantní kódy*, které vznikají restrikcí GRS kódů. Mezi ně patří jak Goppa kódy, tak například BCH kódy. Více informací o různých třídách alternantních kódů a vztazích mezi nimi lze nalézt v [5].

Nechť $C'$ je $[n, k']_q$ kód, kde $q = p^m$, $n, k, m \in \mathbb{N}$ a $p$ je prvočíslo. Kód $C'$ je tedy lineární podprostor $(\mathbb{F}_q)^n$. $\mathbb{F}_q$ lze chápat jako vektorový prostor nad $\mathbb{F}_p$, $\mathbb{F}_p \subset \mathbb{F}_q$, a to dimenze $m$. V tomto pojetí je tedy $(\mathbb{F}_q)^n$ vektorový prostor nad $\mathbb{F}_p$ dimenze $mn$, kde $(\mathbb{F}_p)^n$ je jeho podprostorem. Z kódu $C'$ můžeme odvodit $p$-ární kód $C = C'|_{\mathbb{F}_p}$. Takovému kódu říkáme *reziduální* $p$-ární kód kódu $C'$.

**Tvrzení 1.27.** *Nechť $C'$ je $[N, K, D]_q$ kód, kde $p$ je prvočíslo a $m \in \mathbb{N}$. Reziduální kód $C = C'|_{\mathbb{F}_p^n}$ kódu $C'$ je $[n, k, d]_p$ kód, kde $n = N$, $k \geq mK - (m-1)N$ a $d \geq D$. Navíc, pokud platí $mK > (m-1)N$, pak jde o $[N, mK - (m-1)N, D]_p$ kód.*

*Důkaz.* Viz [6]. □

**Definice 1.28.** *Nechť máme kód $\mathrm{GRS}_q(L, B, k)$, potom alternantní $\mathcal{A}_p(L, B, k)$ kód nad $\mathbb{F}_p$ bude značit kód $\mathrm{GRS}_q(L, B, k)|_{\mathbb{F}_p}$.*

**Tvrzení 1.29.** *Alternantní kód $\mathcal{A}_p(L, B, k)$ má parametry $[n, k', d]_p$, kde*

$$mk - (m-1)n \leq k' \leq k \text{ a } d \geq n - k + 1.$$

*Důkaz.* $\mathrm{GRS}_q(L, B, k)$ má parametry $[n, k, n-k+1]$. $\mathcal{A}_p(L, B, k)$ má zřejmě délku $n$ a jako podprostor lineárního kódu splňuje $k' \leq k$. Zbytek plyne z tvrzení 1.27. □

**Poznámka 1.5.** *Z definice alternantních kódů a z podoby prověrkové matice* GRS *kódů plyne*

$$\mathcal{A}_p(L, B, k) = \{c \in \mathbb{F}_p^n \mid cH^\top = 0\},$$

*kde $H$ je prověrková matice příslušného* GRS *kódu.*

**Poznámka 1.6.** *Každý prvek $b \in \mathbb{F}_q$ můžeme jednoznačně zapsat jako $\sum_{i=0}^{m-1} b_i \alpha^i$, kde $\alpha$ je primitivní prvek $\mathbb{F}_q$ a $b_i \in \mathbb{F}_p$ pro $0 \leq i \leq m-1$. Proto můžeme každý prvek $b$ prověrkové matice $H$ nahradit sloupcovým vektorem $(b_0, \ldots, b_{m-1})^\top$. Získáme $(n-k)m \times n$ matici $\bar{H}$ s prvky z $\mathbb{F}_p$ tak, že platí*

$$\mathcal{A}_p(L, B, k) = \{c \in \mathbb{F}_p^n \mid c\bar{H}^\top = 0\}.$$

*Matice $\bar{H}$ je tedy testovací maticí $\mathcal{A}_p(L, B, k)$ kódu.*

### 1.2.5 Algebraicko–geometrické kódy

Zobecněné Reed–Solomonovy kódy jsou omezené velikostí abecedy – konečného tělesa $\mathbb{F}_q$. Polynom $f(x) \in \mathbb{F}_q[x]$ nemůže mít více jak $q$ kořenů. Pro větší velikosti bloků kódu jsou potřeba větší abecedy. Valerij Denisovič Goppa tento problém obešel konstrukcí kódů pomocí algebraické geometrie. Použil algebraické křivky, které nad konečným tělesem $\mathbb{F}$ mohou mít více jak $|\mathbb{F}|$ bodů, což umožňuje stavbu kódů s většími bloky, viz [7]. Algebraicko–geometrické kódy jsou v tomto smyslu zobecněním GRS kódů.

Než přistoupíme k definici Algebraicko–geometrických kódů, uvedeme některé základní poznatky z algebraické geometrie. Více informací je možné nalézt například v [8] a [9].

#### Některé základní poznatky algebraické geometrie

**Definice 1.30.** Algebraickým funkčním tělesem $F/K$ *se rozumí rozšíření* $K \subseteq F$ *komutativních těles takové, že* $\exists x \in F$, $x$ *je transcendentní nad* $K$ *a pro stupeň rozšíření platí* $[F : K(x)] < \infty$. *V případě* $F = K(x)$ *nazýváme* $F/K$ racionálním funkčním tělesem.

**Definice 1.31.** *Ať* $F/K$ *je algebraické funkční těleso, potom* $\nu : F \to \mathbb{Z} \cup \{\infty\}$ *nazveme* diskrétní valuací, *pokud splňuje*

1. $\nu(xy) = \nu(x) + \nu(y)$, $\forall x, y \in F$,
2. $\nu(x + y) \geq \min\{\nu(x), \nu(y)\}$, $\forall x, y \in F$,
3. $\nu(x) = \infty \Leftrightarrow x = 0$,
4. $\nu(a) = 0$, $\forall\, 0 \neq a \in K$,
5. $\exists z \in F$, $\nu(z) = 1$.

**Definice 1.32.** *Ať* $F/K$ *je algebraické funkční těleso. Okruh* $\mathcal{O} \subsetneq F$ *nazveme* valuační, *je–li* $K \subsetneq \mathcal{O} \subsetneq F$ *a* $\forall a \in F$ *je buď* $a \in \mathcal{O}$, *nebo* $a^{-1} \in \mathcal{O}$.

Uvažme $P = \mathcal{O} \setminus \mathcal{O}^* \subseteq \mathcal{O}$, potom $P$ je neprázdná množina, neboť $\mathcal{O} \subsetneq F$. Dále pro $x \in P$ a $z \in \mathcal{O}$ je $xz \in P$. Navíc pro každé $x, y \in P$ je $\frac{x}{y}$ nebo $\frac{y}{x}$ v $\mathcal{O}$. Bez újmy na obecnosti předpokládejme $\frac{x}{y} \in \mathcal{O}$. Potom

$$1 \in \mathcal{O} \Rightarrow \frac{x}{y} + 1 \in \mathcal{O} \Rightarrow x + y = y\Big(\frac{x}{y} + 1\Big) \in P.$$

Tedy $P$ je ideál $\mathcal{O}$. Protože vlastní ideál nemůže obsahovat 1, je $P$ jednoznačně určený maximální ideál. $P$ naopak jednoznačně určuje okruh, v kterém je maximálním ideálem, a to

$$\mathcal{O}_P = \{z \in F;\, z^{-1} \notin P\}.$$

Jelikož je $P$ maximální ideál, je $\mathcal{O}_P/P$ těleso. Proto $K$ můžeme ztotožnit s

$$(K + P)/P \subseteq \mathcal{O}_P/P.$$

**Definice 1.33.** *Ideál $P = \mathcal{O} \setminus \mathcal{O}^* \subseteq \mathcal{O}$ nazýváme* místo, *$\mathcal{O}_P$ potom* valuační okruh *místa $P$. Množinu všech míst $F/K$ budeme značit $\mathbb{P}_F$.* Stupeň místa *$P$ definujeme jako* $\deg P = [\mathcal{O}_P : K]$, *kde $\mathcal{O}_P$ je valuační okruh s maximálním ideálem $P$.*

**Tvrzení 1.34.** *Valuační okruh $\mathcal{O}$ je obor hlavních ideálů. Speciálně tedy místo $P$ je hlavní ideál $\mathcal{O}$. Navíc, pokud $P = t\mathcal{O}$, potom $\forall\, z \in F^*$ má jednoznačnou reprezentaci $z = t^n u$, $u \in \mathcal{O}^*$.*

*Důkaz.* Viz [10]. □

**Definice 1.35.** *Každý prvek $t \in P$ takový, že $P = t\mathcal{O}$, nazveme* uniformizujícím *prvkem místa $P$. S místem $P \in \mathbb{P}_F$ asociujeme diskrétní valuaci $\nu_P$ definovanou následovně: Zvolme uniformizující prvek $t$ místa $P$. Potom $\forall\, z \in F^*$ definujme $\nu_P(z) = n$, $\nu_P(0) = \infty$, kde $z = t^n u$, $u \in \mathcal{O}_P^*$, $n \in \mathbb{Z}$.*

Pro místo $P \in \mathbb{P}_F$ s diskrétní valuací $\nu_P$ máme

$$\mathcal{O}_P = \{z \in F \mid \nu_P(z) \geq 0\},$$

$$\mathcal{O}_P^* = \{z \in F \mid \nu_P(z) = 0\},$$

$$P = \{z \in F \mid \nu_P(z) > 0\}.$$

Každý prvek $x \in F$ chápeme jako zobrazení $F \to \mathcal{O}_P/P$, kde $x(P) = x + P$, pokud $x \in \mathcal{O}_P$, a $x(P) = \infty$ jinak.

Ať $P \in \mathbb{P}_F$ a $z \in F$. Řekneme, že $P$ je *nulou* prvku $z$ násobnosti $m$, pokud $\nu_P(z) = m > 0$. Řekneme, že $P$ je *pól* prvku $z$ násobnosti $m$, pokud naopak platí $\nu_P(z) = -m < 0$. Speciálně $z \in K$ nemá žádné nuly ani póly.

### Racionální funkční těleso

Vlastnosti racionálního funkčního tělesa, které nyní uvedeme, lze snadno ověřit.

Nechť $x$ je transcendentní nad $K$. Potom $K(x)/K$ je racionální funkční těleso. Pro ireducibilní polynom $p(x) \in K[x]$ máme valuační okruh

$$\mathcal{O}_{p(x)} = \left\{ \frac{f(x)}{g(x)} \,\middle|\, f(x),\, g(x) \in K[x],\, g(x) \neq 0,\, p(x) \nmid g(x) \right\}.$$

Maximální ideál $\mathcal{O}_{p(x)}$ je

$$P_{p(x)} = \left\{ \frac{f(x)}{g(x)} \in \mathcal{O}_{p(x)} \,\middle|\, p(x) \mid f(x) \right\}.$$

Navíc $p(x)$ je uniformizující prvek $P_{p(x)}$ a příslušná valuace $P_{p(x)}$:

$$\nu_{P_{p(x)}}(z) = k \Leftrightarrow k = \max\{i \in \mathbb{Z} \cup \{0\},\, p^i(x) \mid z\}.$$

$\mathcal{O}_{P_{p(x)}}/P_{p(x)}$ je izomorfní $K[x]/(p(x))$. Tento izomorfismus je dán vztahem:

$$\phi = \begin{cases} K[x]/(p(x)) \to \mathcal{O}_{P_{p(x)}}/P_{p(x)} \\ f(x) \mod p(x) \mapsto f(x)(P_{p(x)}). \end{cases}$$

Máme $\deg P = \deg p(x)$.

Ve speciálním případě $p(x) = x - \alpha$, $\alpha \in K$ je $\deg(P_{p(x)}) = 1$ a dále máme $z(P) = z(\alpha)$, $z \in K(x)$, kde $z(\alpha)$ je ve tvaru

$$z(\alpha) = \begin{cases} \frac{f(\alpha)}{g(\alpha)} \text{ pokud } g(\alpha) \neq 0, \\ \infty \text{ pokud } g(\alpha) = 0. \end{cases}$$

Kromě $\mathcal{O}_{p(x)}$ je také $\mathcal{O}_\infty$ valuační okruh, kde

$$\mathcal{O}_\infty = \left\{ \frac{f(x)}{g(x)} \,\middle|\, f(x),\, g(x) \in K[x],\, g(x) \neq 0,\, \deg(f(x)) \leq \deg(g(x)) \right\}.$$

Maximální ideál $\mathcal{O}_\infty$ je

$$P_\infty = \left\{ \frac{f(x)}{g(x)} \in \mathcal{O}_\infty \,\middle|\, \deg(f(x)) < \deg(g(x)) \right\}.$$

Uniformizujícím prvkem je $t = 1/x$ a příslušná valuace potom

$$\nu_\infty\left(\frac{f(x)}{g(x)}\right) = \deg(g(x)) - \deg(f(x)).$$

$\mathcal{O}_{P_\infty}/P_\infty$ je určené pomocí $z(P_\infty) = z(\infty)$, pro $z = \frac{a_n x^n + \ldots + a_0}{b_m x^m + \ldots + b_0} \in K(x)$, $a_n, b_m \neq 0$, kde $z(\infty)$ je definované jako

$$z(P_\infty) = \begin{cases} \frac{a_n}{b_m} \text{ pokud } n = m, \\ 0 \text{ pokud } n < m, \\ \infty \text{ pokud } n > m. \end{cases}$$

**Tvrzení 1.36.** *Místa racionálního funkčního tělesa $K(x)/K$ jsou pouze $P_{p(x)}$ a $P_\infty$.*

*Důkaz.* Viz [9]. □

**Důsledek 1.37.** *Místa racionálního funkčního tělesa $K(x)/K$ stupně 1 odpovídají jedna ku jedné $K \cup \{\infty\}$.*

**Poznámka 1.7.** *V algebraické geometrii je $K \cup \{\infty\}$ často interpretováno jako* projektivní přímka $\mathbb{P}^1$ *nad $K$. Místa $K(x)/K$ stupně 1 potom odpovídají jedna k jedné bodům $\mathbb{P}^1$. Místa stupně 1 potom označujeme racionální body.*

### Body a divisory

Grupa divisorů $\mathrm{Div}(F) = \mathrm{Div}(F/K)$ algebraického funkčního tělesa $F/K$ je volná abelovská grupa generovaná místy $F/K$. Prvky $\mathcal{G} \in \mathrm{Div}(F)$ mají proto tvar

$$\mathcal{G} = \sum_{P \in \mathbb{P}_F} a_P P,$$

kde $a_P$ jsou celá a jsou až na konečný počet nulová. Nosičem $\mathcal{G} = \sum_{P \in \mathbb{P}_F} a_P P$ nazveme konečnou množinu

$$\mathrm{supp}(\mathcal{G}) = \{P \in \mathbb{P}_F \mid a_P \neq 0\}.$$

Definujme částečné uspořádání na $\mathrm{Div}(F)$:

$$\mathcal{D} = \sum_{P \in \mathbb{P}_F} d_P P \geq \mathcal{E} = \sum_{P \in \mathbb{P}_F} e_P P \Leftrightarrow \forall P \in \mathbb{P}_F,\, d_P \geq e_P.$$

Divisor $\mathcal{D} = \sum_{P \in \mathbb{P}_F} d_P P$ nazveme pozitivní, pokud $\mathcal{D} \geq 0$. Alternativně máme $\forall P \in \mathbb{P}_F : d_P \geq 0$. Každý divisor $\mathcal{D} = \sum_{P \in \mathbb{P}_F} d_P P$ asociujeme s jeho stupněm $\deg(\mathcal{D}) \in \mathbb{Z}$ definovaným jako $\deg(\mathcal{D}) = \sum_{P \in \mathbb{P}_F} d_P \deg P$. Tato definice má smysl, neboť $d_P$ jsou nulová až na konečný počet z nich. V tomto smyslu můžeme deg chápat jako homomorfismus z grupy divisorů do grupy celých čísel.

S $0 \neq f \in F$ asociujeme divisor

$$(f) = \sum_{P \in \mathbb{P}_F} \nu_P(f) P.$$

Tato suma je konečná, neboť počet nul a pólů prvku $f$ je konečný. Divisor $(f)$ nazýváme *hlavní*. Kladnou část hlavního divisoru značíme

$$(f)_0 = \sum_{P \in \mathbb{P}_F,\, \nu_P(f) > 0} \nu_P(f) P.$$

Stupeň hlavního divisoru je roven nule, viz [10].

### Riemann-Rochův prostor

*Rod* $g$ algebraického funkčního tělesa $F/K$ je definován jako

$$g = \max_{\mathcal{A} \in \mathrm{Div}(F)} \{\deg \mathcal{A} - \dim(\mathcal{L}(\mathcal{A})) + 1\}.$$

**Poznámka 1.8.** *Rod racionálního funkčního tělesa je roven nule.*

Pro divisor $\mathcal{G} \in \mathrm{Div}(F)$ definujme *Riemman–Rochův* prostor $\mathcal{L}(\mathcal{G})$ následovně:

$$\mathcal{L}(\mathcal{G}) = \{0 \neq f \in F \mid (f) \geq -\mathcal{G}\} \cup \{0\}.$$

Tento prostor je konečně dimenzionální a jeho dimenze je zdola omezená Riemann–Rochovou větou:

$$\dim(\mathcal{L}(\mathcal{G})) \geq \deg(\mathcal{G}) + 1 - g.$$

Důsledkem je, že je–li $\deg(\mathcal{G}) > 2g - 2$, potom u Riemann–Rochovy věty nastává rovnost. Pro divisor $\deg(\mathcal{G}) < 0$ máme totiž $\deg \mathcal{L}(\mathcal{G}) = 0$, neboť pro $0 \neq x \in \mathcal{L}(\mathcal{G})$ by platilo $(x) \geq -\mathcal{G} > 0$, což není možné, neboť stupeň $(f)$ je roven nule.

### Algebraicko–geometrické kódy

Nyní již můžeme přistoupit k samotné definici Algebraicko–geometrických kódů.

**Definice 1.38.** *Nechť $F/\mathbb{F}_q$ je algebraické funkční těleso rodu $g$. Ať $\mathcal{G}$ je divisor $F/\mathbb{F}_q$ a nechť $P_1, \ldots, P_n$, $n \in \mathbb{N}$ jsou po dvou různá místa $F_q(x)/\mathbb{F}_q$ stupně 1 taková, že pro divisor $\mathcal{D} = P_1 + \ldots + P_n$ máme* $\mathrm{supp}(\mathcal{G}) \cap \mathrm{supp}(\mathcal{D}) = \emptyset$. *Potom* Algebraicko–geometrický kód *(AG kód) $\mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{G})$ definujeme jako obraz zobrazení*

$$ev_{\mathcal{D}} : \mathcal{L}(\mathcal{G}) \to \mathbb{F}_q^n,$$

*kde*

$$ev_{\mathcal{D}} : f \to (f(P_1), \ldots, f(P_n)).$$

Tedy

$$\mathcal{C}_{\mathcal{L}}(\mathcal{D},\mathcal{G}) = \{(f(P_1),\ldots,f(P_n)) \mid f \in \mathcal{L}(\mathcal{G})\} \subseteq \mathbb{F}_q^n.$$

**Poznámka 1.9.** *Pro $f \in \mathcal{L}(\mathcal{G})$ máme $v_{P_i}(f) \geq 0$, $1 \leq i \leq n$, neboť platí*

$$\operatorname{supp}\mathcal{G} \cap \operatorname{supp}\mathcal{D} = \emptyset.$$

*Navíc $f(P_i) \in \mathcal{O}_{P_i}/P_i$. Jelikož $\deg P_i = 1$ máme $f(P_i) \in \mathbb{F}_q$.*

Parametry Algebraicko–geometrických kódů vyplývají z Riemann-Rochovy věty.

**Tvrzení 1.39.** *Nechť máme $\deg(\mathcal{G}) < n$ a $\mathcal{D} = P_1 + \ldots + P_n$. Potom Algebraicko–geometrický kód $\mathcal{C}_{\mathcal{L}}(\mathcal{D},\mathcal{G})$ má parametry $[n,k,d]_q$, kde $k \geq \deg(\mathcal{G}) + 1 - g$ a $d \geq n - \deg(\mathcal{G})$. Navíc pokud $2g - 2 < \deg(\mathcal{G})$, potom $k = \deg(\mathcal{G}) + 1 - g$.*

*Důkaz.* Z předpokladu $\deg(\mathcal{G}) < n$ máme $\deg(\mathcal{G} - \mathcal{D}) = \deg\mathcal{G} - n < 0$. Tedy $\mathcal{L}(\mathcal{G} - \mathcal{D}) = 0$. Lineární zobrazení $\mathrm{ev} : \mathcal{L}(\mathcal{G}) \to \mathcal{C}_{\mathcal{L}}(\mathcal{D},\mathcal{G})$ je na. Jádrem ev je

$$\operatorname{Ker}(\mathrm{ev}) = \{f \in \mathcal{L} \mid v_{P_i}(f) > 0,\ 1 \leq i \leq n\} = \mathcal{L}(\mathcal{G} - \mathcal{D}) = \{0\}.$$

Tedy ev je prosté. Proto máme $k = \dim\mathcal{C}_{\mathcal{L}}(\mathcal{D},\mathcal{G}) = \dim\mathcal{L}(\mathcal{G})$. Z Riemann–Rochovy věty dostáváme $k = \dim\mathcal{L}(\mathcal{G}) \geq \deg(\mathcal{G}) + 1 - g$. Je–li $2g - 2 < \deg(G)$, potom dostáváme rovnost. Pro odhad minimální vzdálenosti $d$ předpokládejme, že $\mathcal{C}_{\mathcal{L}}(\mathcal{D},\mathcal{G}) \neq \{0\}$. Zvolme $l \in \mathcal{L}(\mathcal{G})$ tak, že $\mathrm{wt}(\mathrm{ev}(l)) = d$. Potom právě pro $n-d$ míst $P_{j_1},\ldots,P_{j_{n-d}} \in \operatorname{supp}(\mathcal{D})$ je nulou $l$, tedy

$$0 \neq l \in \mathcal{L}(\mathcal{G} - (P_{j_1} + \ldots + P_{j_{n-d}})).$$

Z definice máme

$$(l) + \mathcal{G} - (P_{j_1} + \ldots + P_{j_{n-d}}) \geq 0.$$

Proto také

$$\deg((l)) + \deg(\mathcal{G}) - \deg(P_{j_1} + \ldots + P_{j_{n-d}}) \geq 0.$$

Jelikož $\deg((l)) = 0$, dostáváme

$$\deg(\mathcal{G}) - (n - d) \geq 0.$$

□

**Poznámka 1.10.** *Polynom $f \in \mathbb{F}_q[x]$ stupně $t$ můžeme pokládat za racionální funkci s jediným pólem $P_\infty$ násobnosti $t$. Máme totiž*

$$\deg(f) \leq t \iff (f) \geq -tP_\infty \iff f \in \mathcal{L}(tP_\infty).$$

**Lemma 1.40.** *Nechť máme monický polynom $f \in \mathbb{F}_q[x]$, $\deg(f) < n$ a*

$$L = (\alpha_1,\ldots,\alpha_n) \in \mathbb{F}_q^n,$$

*potom pomocí Lagrangeovy interpolace můžeme polynom $f(x)$ vyjádřit jako*

$$f(x) = \sum_{i=1}^{n} L_i(x) f(\alpha_i),$$

*kde*

$$L_i(x) = \prod_{\substack{j=1\\ j\neq i}}^{n} \frac{x - \alpha_j}{\alpha_i - \alpha_j}.$$

*Důkaz.* Dosazením hodnot $\alpha_1, \ldots, \alpha_n$ za proměnou $x$ dostáváme $L_i(\alpha_j) = 0$ pro $j \neq i$ a $L_i(\alpha_i) = 1$. Polynom

$$f(x) = L_1 f(\alpha_1) + L_2 f(\alpha_2) + \ldots + L_n f(\alpha_n) = \sum_{i=1}^{n} L_i(x) f(\alpha_i)$$

má proto v každém z bodů $\alpha_i$ pro $1 \leq i \leq n$ předepsanou hodnotu $f(\alpha_i)$. Jelikož je stupně menšího jak $n$, je těmito hodnotami určen. □

**Tvrzení 1.41.** *Nechť $n$ je přirozené a ať*

$$B = (\beta_1, \ldots, \beta_n) \in (\mathbb{F}_q^*)^n,\ L = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_q^n,$$

*kde $\alpha_i \neq \alpha_j$, je-li $i \neq j$. Nechť $k$ je celé takové, že $1 \leq k \leq n$. Nechť $u \in \mathbb{F}_q[x]$ takový, že $u(\alpha_i) = \beta_i$. Potom každý $\mathrm{GRS}_q(L, B, k)$ můžeme reprezentovat jako Algebraicko–geometrický kód.*

*Důkaz.* Díky Lagrangeovy interpolaci víme, že polynom $u$ existuje. Můžeme ho získat z lemma 1.40. Potom tedy $\deg u(x) \leq n - 1$. Uvažme racionální funkční těleso $\mathbb{F}_q(x)/\mathbb{F}_q$. Označme $P_i$ nulu $x - \alpha_i$, pro $1 \leq i \leq n$, a $P_\infty$ pól $x$. Potom $u(P_i) = \beta_i$ pro $1 \leq i \leq n$. Definujme divisory

$$\mathcal{D} = P_1 + \ldots + P_n \text{ a } \mathcal{G} = (k-1)P_\infty - (u).$$

Jelikož je stupeň hlavního divisoru $(u)$ roven 0, máme $\deg \mathcal{G} = k - 1 \geq 0$. Dle poznámky 1.10 máme $(u) \geq -(n-1)P_\infty$. Nechť máme polynom $f \in \mathbb{F}_q[x]$ takový, že $\deg f < k$. Z poznámky 1.10 máme $(f) \geq -(k-1)P_\infty$ a

$$(uf) = (u) + (f) \geq (k-1)P_\infty - \mathcal{G} - (k-1)P_\infty = -\mathcal{G}.$$

Máme $(uf) \in \mathcal{L}(\mathcal{G})$ a navíc dostáváme

$$u(\alpha_i) f(\alpha_i) = \beta_i f(\alpha_i).$$

Proto získáváme

$$\mathrm{GRS}_q(L, B, k) \subseteq \mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{G}).$$

Rovnost dokážeme tím, že ukážeme rovnost dimenzí kódů. Dimenze $\mathrm{GRS}_q(L, B, k)$ je $k$. Jelikož je rod $g$ racionálního funkčního tělesa roven 0 a $-2 < \deg \mathcal{G}$, dostáváme z tvrzení 1.39 rovnost

$$\dim \mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{G}) = \deg(\mathcal{G}) + 1 - g = k,$$

čímž jsme toto tvrzení dokázali.

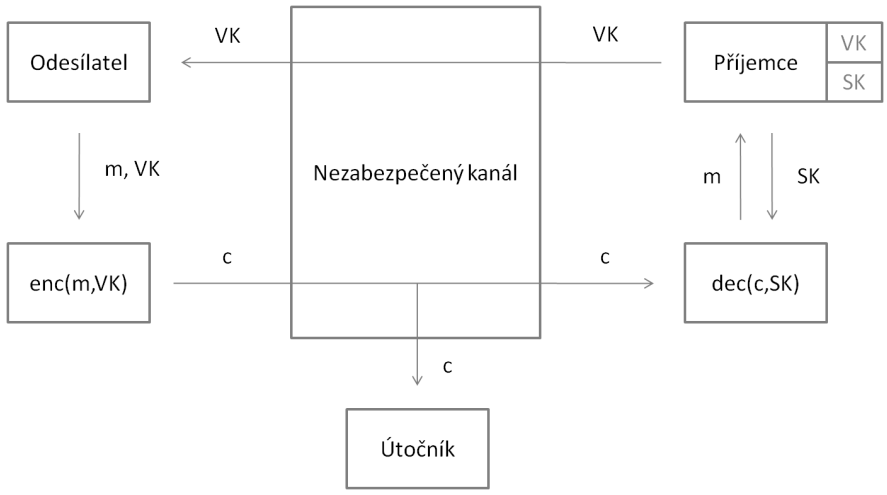
□

**Poznámka 1.11.** *Algebraicko–geometrické kódy můžeme dát do vztahu s eliptickými křivkami. Místa stupně 1 racionálního funkčního tělesa totiž odpovídají racionálním bodům hladké projektivní absolutně ireducibilní křivky nad konečným tělesem $\mathbb{F}_q$, viz [9].*

## 1.3 Kryptografie

V této sekci nastíníme úvod do kryptografie. Představíme asymetrickou kryptografii neboli kryptografii s veřejným klíčem a post–kvantovou kryptografii, jejímž zástupcům se budeme v druhé části této práce věnovat.

Kryptografii budeme chápat jako vědu o metodách zabezpečení přenosu zpráv mezi dvěma nebo více osobami. Pro zabezpečený přenos zpráv dělíme kryptografii na *symetrickou* a *asymetrickou*. Řekněme, že uživatel Alice chce poslat tajnou zprávu Bobovi přes nezabezpečený kanál. Pro takovýto přenos je potřeba domluvit tajný klíč, pomocí kterého budou Alice a Bob komunikovat. V symetrickém případě je klíč Boba a Alice shodný. U asymetrické kryptografie potom existují dva typy klíčů – klíč sloužící k šifrování nazývaný *veřejný klíč*, zkráceně $VK$, a dešifrovací *soukromý klíč*, $SK$. Přenos veřejného klíče nemusí být zabezpečený, proto asymetrická kryptografie nevyžaduje nutnost zabezpečené výměny klíčů. Takovýto přenos by měl pouze disponovat autentizací pro ověření, že máme veřejný klíč správného uživatele. Nechť tedy Alice zvolí zprávu $m$, zašifruje ji Bobovým $VK$ a odešle. Bob potom jako jediný může dešifrovat zprávu $m$, neboť jako jediný vlastní $SK$. V asymetrické kryptografii potřebujeme funkci, jejíž hodnoty lze lehce nalézt (každý může zašifrovat zprávu) a již je výpočetně obtížné invertovat, pokud nemáme nějakou dodatečnou informaci. Takovéto funkce nazýváme *jednosměrné funkce s padacími vrátky*. Ke konstrukci těchto funkcí často využíváme $\mathcal{NP}$ problémy. Počátek asymetrické kryptografie datujeme k roku 1976, kdy Diffie a Hellman [11] předvedli první zabezpečenou výměnu zprávy pomocí dvojice $VK$ a $SK$.




Obrázek 1.3: Obecné schéma asymetrické kryptografie.

Obrázek 1.3 ilustruje obecné schéma asymetrické kryptografie. Schéma obsahuje

- generátor klíčů, který vytvoří z náhodných vstupních dat dvojici ($VK$, $SK$), kde $VK$ je veřejný a $SK$ soukromý klíč,
- šifrovací funkce $\text{enc}(m, VK) = c$, kde enc závisí jen na $VK$ a zprávě $m$,

- dešifrovací funkce $\mathrm{dec}(c, SK) = m$, kde dec závisí jen na $SK$ a šifrové zprávě $c$.

Prostor klíčů a otevřených i šifrových zpráv lze brát jako množinu $\mathbb{F}_q^n$. Asymetrický kryptosystém musí splňovat podmínku

$$\mathrm{dec}(\mathrm{enc}(m, VK), SK)) = m$$

pro všechny páry $(VK,\ SK)$ prostoru klíčů a pro všechny zprávy z prostoru zpráv. Dále požadujeme, aby výpočty funkcí enc i dec probíhaly v polynomiálním čase, a nakonec, aby z $VK$ a šifrového textu $c$ bylo výpočetně náročné získat $m$.

Asymetrickou kryptografii v praxi používáme pouze k bezpečnému přenosu klíče, neboť samotný proces šifrování a dešifrování je pro větší zprávy často výpočetně zdlouhavý. Po přenosu klíče přejdeme k *symetrické kryptografii*, která je obecně rychlejší, avšak požaduje bezpečný přenos klíče.

Hlavními zástupci asymetrické kryptografie jsou kryptosystémy založené na teorii čísel, latinských čtvercích a samoopravných kódech. V praxi populární jsou ty, které jsou založené na teorii čísel – problému faktorizace a diskrétního logaritmu.

- **Problém faktorizace:** Pro dané $n = pq$, kde $p$ a $q$ jsou dvě různá prvočísla, nalezněte $p$ a $q$. Jedná se o těžký problém. Nejrychlejší známé algoritmy mají exponenciální složitost.

- **Problém diskrétního logaritmu:** Nechť $a$ je celé kladné a $p$ je prvočíslo. Pro $b = a^c \mod p$ nalezněte $c$. Pro velká čísla jde o těžký problém. Nejrychlejší známé algoritmy mají exponenciální složitost.

### 1.3.1 Post–kvantová kryptografie

Přestože v praxi používáme silná kryptografická schémata s veřejným klíčem založená na problému faktorizace čísel či na problému diskrétního logaritmu, a přestože proti nim neexistují efektivní útoky, existují názory vyžadující alternativní schémata, a to ze dvou důvodů:

Za prvé, v případě příchodu výkonných kvantových počítačů budou tato schémata nepoužitelná, jmenovitě například RSA, DSA, ECC, ECDSA. Problém faktorizace a diskrétního algoritmu bude totiž možné řešit v polynomiálním čase pomocí Shorova algoritmu [12] nebo jeho modifikovaných verzích [13] [14] pro eliptické křivky.

Za druhé standardní schémata s dostatečně bezpečnými parametry jsou často příliš výpočetně náročná pro použití v malých zařízeních, jako jsou *RFID čipy*, nebo *senzorové sítě*.

Budeme popisovat zástupce *post–kvantové kryptografie*, tedy kryptosystémy s veřejným klíčem,

- pro které nejsou známé žádné útoky pomocí kvantových počítačů a

- které můžeme implementovat i na klasických počítačích.

Zástupci takové kryptografie jsou některé systémy založené na **samoopravných kódech**, hashích, mřížích či multivarietách, [15].

Hang Dinh a další v [16] ukázali, že právě takovým post–kvantovým kryptosystémem je také McEliecův kryptosystém, který budeme podrobněji popisovat v kapitole 3. Tento kryptosystém je založen na samoopravných kódech a je výpočetně nenáročný.

# Kapitola 2

# Goppa kódy

V první části této kapitoly uvedeme definici Goppa kódů a její prověrkové a generující matice. Dále ukážeme možnost konstrukce Goppa kódů přes evaluace polynomů a zařadíme Goppa kódy do třídy Algebraicky–geometrických a Zobecněných Reed–Solomonových kódů. Uvedeme parametry Goppa kódů a zajímavou vlastnost takzvaných *Divokých Goppa kódů*. V poslední sekci se budeme věnovat dekódování, které používá jako klíčový prvek Euklidův algoritmus. U dekódovacího algoritmu v sekci 2.5 přejdeme od syndromového vektoru k chybově lokačnímu a chybově evaluačnímu vektoru a využijeme zmíněný Euklidův algoritmus pro dekódování slova.

## 2.1 Klasická konstrukce

Než přistoupíme k vlastní konstrukci, je potřeba zavést pojem lokalizace.

### Lokalizace

Nechť okruh $R$ je eukleidovským oborem. Ať $g \in R$ je nenulový a nechť $S \subset R$ je multiplikativní množina obsahující jedničku, neobsahující nulu, složená ze všech prvků nesoudělných s $g$. Potom je $P = R \setminus S$ ideál. Speciálně $(g) \subseteq P$. Sestrojme lokalizaci $R$ v $\mathcal{P}$, značíme $S^{-1}R$, jako

$$S^{-1}R = \left\{ \frac{a}{b} \,\middle|\, b \in S,\, a \in R \right\}$$

podmnožinu podílového tělesa

$$T = \left\{ \frac{a}{b} \,\middle|\, b \neq 0,\, a, b \in R \right\}.$$

**Tvrzení 2.1.** *Pro každý ideál $\mathcal{N}$ okruhu $R$ platí, že $S^{-1}\mathcal{N}$ je ideál okruhu $S^{-1}R$. Přitom je vlastním ideálem právě tehdy, když $\mathcal{N} \cap S = \emptyset$. Naopak, každý ideál $\mathcal{J}$ okruhu $S^{-1}R$ je roven ideálu $S^{-1}\gamma^{-1}(\mathcal{J})$, kde $\gamma : R \to S^{-1}R$ je přirozený homomorfismus. Přitom*

$$\gamma^{-1}(\mathcal{J}) = \left\{ a \in R \,\middle|\, \frac{a}{s} \in \mathcal{J} \text{ pro některé } s \in S \right\} = \left\{ a \in R \,\middle|\, \frac{a}{1} \in \mathcal{J} \right\}.$$

*Důkaz.* Viz [17]. □

Označme $\mathcal{I}$ ideál $S^{-1}R$ tvořený zlomky $\frac{u}{v}$, kde $v \in S$ a $u$ je násobek $g$. Tedy $u \in (g)$. Ať $u, w \in R$, potom ideál $\mathcal{I}$ má následující vlastnosti:

$$u \equiv w \mod \mathcal{I} \Leftrightarrow \frac{u}{v} \equiv \frac{w}{v} \mod \mathcal{I} \Leftrightarrow vu \equiv vw \mod \mathcal{I}, \tag{2.1}$$

$$u \equiv w \mod \mathcal{I} \Leftrightarrow u \equiv w \mod g. \tag{2.2}$$

**Vlastní konstrukce**

Uvažme okruh $R = \mathbb{F}_{p^m}[x]$. Nechť $n$ je přirozené číslo, $n \leq p^m$, a ať

$$L = \{\alpha_1, \ldots, \alpha_n\} \in \mathbb{F}_{p^m},$$

kde $\alpha_i \neq \alpha_j$ pro $i \neq j$. Dále nechť máme celé $t$ takové, že $0 < t < n$. Potom monický polynom

$$g(x) = \sum_{j=0}^{t} g_j x^j \in \mathbb{F}_{p^m}[x]$$

stupně $t$ nazýváme *Goppovým polynomem* pokud

$$g(\alpha_i) \neq 0, \text{ je-li } 1 \leq i \leq n.$$

Nechť $S$ je multiplikativní množina složená ze všech polynomů nesoudělných s polynomem $g(x) \in \mathbb{F}_{p^m}[x]$. Uvažme ideál $\mathcal{I}$ generovaný polynomem $g(x)$. Potom v $S^{-1}R$ definujme kód

$$C_{p^m}(L, g) = \left\{ c = (c_1, \ldots, c_n) \in \mathbb{F}_{p^m}^n : \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \bmod g(x) \right\}.$$

*Goppa kód* $\Gamma_p(L, g)$ definujeme jako kód reziduální kódu $C_{p^m}(L, g)$. Tedy

$$\Gamma = \Gamma_p(L, g) = \left\{ c = (c_1, \ldots, c_n) \in \mathbb{F}_p^n : \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \bmod g(x) \right\}.$$

Jak ukážeme v tvrzení 2.7, jedná se o lineární kód s parametry

$$[n, k, d]_p,$$

kde $k \geq n - mt, d \geq t + 1$. Goppa kód se nazývá *ireducibilní*, pokud je polynom $g$ ireducibilní.

Při popisu Goppa kódů budeme často využívat vlastností kódu $C_{p^m}(L, g)$, který, jak později dokážeme, je Zobecněným Reed–Solomonůvým kódem. Ukážeme, že Goppa kód oproti tomuto kódu umožňuje lepší možnosti dekódování a bezpečnost v kryptografii založené na samoopravných kódech.

**Prověrková matice**

Je zřejmé, že Goppa kód $\Gamma$ je lineární. Z kongruence

$$\sum_{i=1}^{n} \frac{c_i}{x-\alpha_i} \equiv 0 \bmod g(x)$$

lze zkonstruovat prověrkovou matici. V okruhu $\mathbb{F}_{p^m}[x]/(g)$ má polynom $x-\alpha_i$ inverzní prvek díky nesoudělnosti s $g(x)$. Můžeme ho získat Euklidovým algoritmem, a to například ve tvaru

$$-\frac{g(x)-g(\alpha_i)}{x-\alpha_i} g(\alpha_i)^{-1}.$$

Skutečně platí

$$(x-\alpha_i)\Big(-\frac{(g(x)-g(\alpha_i))}{x-\alpha_i} g(\alpha_i)^{-1}\Big) \equiv 1 \mod g(x).$$

Tedy $c \in \Gamma_p(L,g)$ právě tehdy, když $c \in \mathbb{F}_p^n$ a platí

$$\sum_{i=1}^{n} c_i \frac{g(x)-g(\alpha_i)}{x-\alpha_i} g(\alpha_i)^{-1} \equiv 0 \bmod g(x).$$

Jelikož

$$\deg\Big(\frac{g(x)-g(\alpha_i)}{x-\alpha_i}\Big) < \deg(g) = t,$$

dostáváme

$$c \in \Gamma \Leftrightarrow c \in \mathbb{F}_p^n \text{ a } \sum_{i=1}^{n} c_i \frac{g(x)-g(\alpha_i)}{x-\alpha_i} g(\alpha_i)^{-1} = 0 \text{ v } \mathbb{F}_{p^m}[x]. \tag{2.3}$$

Pro Goppův polynom

$$g(x) = \sum_{j=0}^{t} g_j x^j$$

s koeficienty $g_j \in \mathbb{F}_{p^m}$ a $g_t \neq 0$, potom máme

$$\frac{g(x)-g(\alpha_i)}{x-\alpha_i} = \frac{\sum_{j=0}^{t} g_j (x^j - \alpha_i^j)}{x-\alpha_i} = \sum_{j=0}^{t} g_j \frac{x^j-\alpha_i^j}{x-\alpha_i} =$$

$$= g_t(x^{t-1} + x^{t-2}\alpha_i + \ldots + \alpha_i^{t-1}) + g_{t-1}(x^{t-2} + \ldots + \alpha_i^{t-2}) + \ldots + g_2(x+\alpha_i) + g_1 =$$

$$= \sum_{j=1}^{t} x^{t-j} \Big(\sum_{k=1}^{j} g_{t-k+1} \alpha_i^{j-1}\Big).$$

Pokud dosadíme do (2.3) a porovnáme $x$-ové koeficienty na levé straně rovnosti s nulovými koeficienty nulového polynomu na straně pravé, dostaneme

$$c \in \Gamma_p(L,g) \Leftrightarrow Hc^\top = 0 \text{ a } c \in \mathbb{F}_p^n,$$

kde

$$H = \begin{pmatrix} g_t g(\alpha_1)^{-1} & \dots & g_t g(\alpha_n)^{-1} \\ (g_{t-1} + \alpha_1 g_t) g(\alpha_1)^{-1} & \dots & (g_{t-1} + \alpha_1 g_t) g(\alpha_n)^{-1} \\ \vdots & \ddots & \vdots \\ (g_1 + \alpha_1 g_2 + \ldots + \alpha_1^{t-1} g_t) g(\alpha_1)^{-1} & \dots & (g_1 + \alpha_n g_2 + \ldots + \alpha_n^{t-1} g_t) g(\alpha_n)^{-1} \end{pmatrix}.$$

Tedy $H$ je prověrková $t \times n$ matice kódu $\Gamma$ a zároveň vidíme, že kód $C_{p^m}(L, g)$ má parametry $[n, n-t]_{p^m}$.

U Goppa kódů nám stačí kontrolovat rovnost $Hc^\top = 0$ nad tělesem $\mathbb{F}_{p^m}$, neboť $\mathbb{F}_p$ je jeho podtělesem. Označme

$$Z = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 \\ g_{t-1} & 1 & 0 & \dots & 0 \\ g_{t-2} & g_{t-1} & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ g_1 & g_2 & g_3 & \dots & 1 \end{pmatrix}, X = \begin{pmatrix} 1 & 1 & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \dots & \alpha_n^2 \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_1^{t-1} & \alpha_2^{t-1} & \dots & \alpha_n^{t-1} \end{pmatrix}$$

$$Y = \begin{pmatrix} g(\alpha_1)^{-1} & 0 & \dots & 0 \\ 0 & g(\alpha_2)^{-1} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & g(\alpha_n)^{-1} \end{pmatrix}.$$

Potom $H = ZXY$ a $Z$ je $t \times t$ matice, $X$ je $t \times n$ Vandermondova matice a $Y$ je $n \times n$ diagonální matice. Jelikož je matice $Z$ invertibilní, je prověrkovou maticí také následující jednodušší matice

$$H' = XY = \begin{pmatrix} \frac{1}{g(\alpha_1)} & \frac{1}{g(\alpha_2)} & \dots & \frac{1}{g(\alpha_n)} \\ \frac{\alpha_1}{g(\alpha_1)} & \frac{\alpha_2}{g(\alpha_2)} & \dots & \frac{\alpha_n}{g(\alpha_n)} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\alpha_1^{t-1}}{g(\alpha_1)} & \frac{\alpha_2^{t-1}}{g(\alpha_2)} & \dots & \frac{\alpha_n^{t-1}}{g(\alpha_n)} \end{pmatrix}.$$

Testovací matici s prvky tělesa $\mathbb{F}_p$ potom získáme, pokud dle poznámky 1.6 každý prvek prověrkové matice $H$, respektive $H'$, nahradíme jemu odpovídajícím sloupcovým vektorem délky $m$ nad $\mathbb{F}_p$.

Z porovnání prověrkové matice kódu $C_{p^m}(L, g)$ a struktury prověrkové matice GRS kódu je vidět, že je tento kód GRS. Tedy, že Goppa kód je alternantní. Přímý důkaz tohoto faktu uvedeme v tvrzení 2.5.

Pokud lze převést prověrkovou matici do speciálního tvaru $H = (I_k|P)$, kde $I_j$ je jednotková diagonální matice příslušné velikosti $j$, potom můžeme získat generující matici Goppa kódu podle následujícího vztahu:

$$G = (-P^\top | I_{n-k})|_{\mathbb{F}_p}.$$

### Polynomiálně-evaluační pohled na Goppa kódy

Goppa kódy můžeme definovat více způsoby. Jedním ze způsobů je *polynomiálně–evaluační*. Goppa kódy definujeme jako evaluace určitých polynomů na souřadnicích vektoru $L$.

Něchť $L$, $g$ jsou definovány stejně jako v předchozí sekci. Položme

$$h(x) = \prod_{i=1}^{n}(x - \alpha_i) \in \mathbb{F}_{p^m}[x].$$

Pak $\mathrm{NSD}(g(x), h(x)) = 1$, neboť prvky $\alpha_i$ nejsou kořeny polynomu $g(x)$. Máme

$$h'(x) = \sum_{i=1}^{n} \prod_{\substack{j=1 \\ j \neq i}}^{n} (x - \alpha_j),$$

kde $h'$ značí derivaci $h$. Platí

$$h'(\alpha_i) = \prod_{\substack{j=1 \\ j \neq i}}^{n} (\alpha_i - \alpha_j),\ \forall i \in \{1, \ldots, n\}.$$

**Definice 2.2.** *Pro každý monický polynom $f \in g\mathbb{F}_{p^m}[x]$ definujme*

$$\mathrm{ev}_{h'}(f) = \Big(\frac{f(\alpha_1)}{h'(\alpha_1)}, \frac{f(\alpha_2)}{h'(\alpha_2)}, \ldots, \frac{f(\alpha_n)}{h'(\alpha_n)}\Big) \in \mathbb{F}_{p^m}^n.$$

**Lemma 2.3.** *Pokud má monický polynom $f \in g\mathbb{F}_{p^m}[x]$ stupeň menší než $n$, potom je určen souřadnicemi $\mathrm{ev}_{h'}(f)$ a platí*

$$f(x) = \sum_{i=1}^{n} \frac{f(\alpha_i)h(x)}{h'(\alpha_i)(x - \alpha_i)}.$$

*Důkaz.* Z lemma 1.40 plyne určení souřadnicemi $\mathrm{ev}_{h'}(f)$. Upravme

$$f(x) = \sum_{i=1}^{n} \prod_{\substack{j=1 \\ j \neq i}}^{n} f(\alpha_i) \frac{x - \alpha_j}{\alpha_i - \alpha_j} = \sum_{i=1}^{n} \frac{f(\alpha_i)h(x)}{(x - \alpha_i) \prod_{\substack{j=1 \\ j \neq i}} (\alpha_i - \alpha_j)} = \sum_{i=1}^{n} \frac{f(\alpha_i)h(x)}{h'(\alpha_i)(x - \alpha_i)}.$$

□

**Tvrzení 2.4.** *Nechť máme $C_{p^m}(L, g)$ a ať $h(x) = \prod_{i=1}^{n}(x - \alpha_i) \in \mathbb{F}_{p^m}[x]$, potom platí*

$$C_{p^m}(L, g) = \{\mathrm{ev}_{h'}(f),\ f \in g\mathbb{F}_{p^m}[x],\ \deg(f) < n,\ f \text{ monický}\} =$$
$$= \Big\{\Big(\frac{f(\alpha_1)}{h'(\alpha_1)}, \ldots, \frac{f(\alpha_n)}{h'(\alpha_n)}\Big),\ f \in g\mathbb{F}_{p^m}[x],\ \deg(f) < n,\ f \text{ monický}\Big\}.$$

*Důkaz.* Z vlastnosti (2.1) a nesoudělnosti $g(x)$ a $h(x)$ máme

$$\sum_{i=1}^{n} c_i \frac{1}{x - \alpha_i} \equiv 0 \mod g(x) \Leftrightarrow \sum_{i=1}^{n} c_i \frac{h(x)}{x - \alpha_i} \equiv 0 \mod g(x).$$

Zapišme tedy kód $C_{p^m}(L,g)$ následovně

$$C_{p^m}(L,g) = \Big\{ c \in \mathbb{F}_{p^m}^n \ : \ \sum_{i=1}^{n} c_i \frac{h(x)}{x-\alpha_i} \equiv 0 \mod g(x) \Big\}.$$

Nechť máme monický polynom $f \in g\mathbb{F}_{p^m}[x]$, kde $f(x) = p(x)g(x)$ pro nějaké $p \in \mathbb{F}_{p^m}[x]$ stupně $< n-t$. Potom vektor

$$\Big( \frac{p(\alpha_1)g(\alpha_1)}{h'(\alpha_1)}, \dots, \frac{p(\alpha_n)g(\alpha_n)}{h'(\alpha_n)} \Big) \in C_{p^m}(L,g),$$

neboť z lemma 2.3 dostáváme

$$\sum_{i=1}^{n} \frac{p(\alpha_i)g(\alpha_i)}{h'(\alpha_i)} \frac{h(x)}{(x-\alpha_i)} = p(x)g(x) \equiv 0 \mod g(x).$$

Naopak každý prvek $C$ lze zapsat jako vektor tohoto tvaru. Pokud totiž pro polynom $d(x)$ platí

$$d(x) = \sum_{i=1}^{n} c_i \frac{h(x)}{x-\alpha_i} \equiv 0 \mod g(x),$$

pak je tento polynom násobek $g(x)$, řekněme $d(x) = p(x)g(x)$, a máme

$$p(\alpha_i)g(\alpha_i) = c_i h'(\alpha_i),$$

tedy

$$c = \Big( \frac{p(\alpha_1)g(\alpha_1)}{h'(\alpha_1)}, \dots, \frac{p(\alpha_n)g(\alpha_n)}{h'(\alpha_n)} \Big).$$

□

Goppa kód proto jako residuální kód kódu $C_{p^m}(L,g)$ definují evaluace polynomů na bodech $L$.

## 2.2 Zařazení Goppa kódů

V předchozí sekci jsme nahlédli, že Goppa kódy jsou alternantní. V této sekci pro lepší orientaci v souvislostech mezi kódy uvedeme přímý důkaz téhož a ukážeme, že Goppa kódy jsou také Algebraicko–geometrické.

### 2.2.1 Goppa kódy jako alternantní kódy

**Tvrzení 2.5.** *Goppa kód $\Gamma_p(L,g)$ je alternantní kód kódu $\mathrm{GRS}_{p^m}(L,B,n-t)$ pro $L = (\alpha_1,\dots,\alpha_n) \in \mathbb{F}_{p^m}^n$, kde $\forall k$ platí $g(\alpha_k) \neq 0$, $\alpha_i \neq \alpha_j$ pro $i \neq j$ a kde $B = (\beta_1,\dots,\beta_n)$ máme definované následovně*

$$\beta_i = \frac{g(\alpha_i)}{\prod_{\substack{j=1 \\ j\neq i}}^{n} (\alpha_i - \alpha_j)}, \ 1 \le i \le n.$$

*Důkaz.* Nechť máme $u \in \mathrm{GRS}_{p^m}(L, B, n-t)$. Potom

$$u_i = \beta_i f(\alpha_i) = \frac{f(\alpha_i)g(\alpha_i)}{\prod_{\substack{j=1 \\ j\neq i}}^{n}(\alpha_i - \alpha_j)},$$

kde $f(x)$ je polynom stupně $< n-t$, tedy

$$\sum_{i=1}^{n} \frac{u_i}{x-\alpha_i} = \frac{1}{\prod_{i=1}^{n}(x-\alpha_i)} \sum_{i=1}^{n} f(\alpha_i)g(\alpha_i) \prod_{\substack{j=1 \\ j\neq i}}^{n} \frac{x-\alpha_j}{\alpha_i - \alpha_j}.$$

Označme

$$l(x) = \sum_{i=1}^{n} f(\alpha_i)g(\alpha_i) \prod_{\substack{j=1 \\ j\neq i}}^{n} \frac{x-\alpha_j}{\alpha_i - \alpha_j}.$$

Potom $l(\alpha_i) = f(\alpha_i)g(\alpha_i)$ pro $i = 1, \ldots, n$. Dále pro stupně polynomů platí $\deg l(x) \leq n-1$ a $\deg f(x)g(x) \leq n-1$. Jelikož polynom $l(x) - f(x)g(x)$ je určený hodnotami na $n$ bodech $\alpha_i$, máme z lemma 1.40 $l(x) = f(x)g(x)$. Tedy

$$\sum_{i=1}^{n} \frac{u_i}{x-\alpha_i} = \frac{f(x)g(x)}{\prod_{i=1}^{n}(x-\alpha_i)},$$

a proto $u \in C_{p^m}(L, g)$. Tedy

$$C_{p^m}(L, g) \supseteq \mathrm{GRS}_{p^m}(L, B, n-t),$$

a jelikož mají oba kódy dimenzi $n-t$, dostáváme rovnost. Proto

$$\Gamma_p(L, g) = C_{p^m}(L, g)|_{\mathbb{F}_p} = \mathrm{GRS}_{p^m}(L, B, n-r)|_{\mathbb{F}_p} = \mathcal{A}_p(L, B, n-t).$$

□

### 2.2.2 Goppa kódy jako Algebraicko–geometrické kódy

V tvrzení 1.41 jsme dokázali, že GRS kódy jsou Algebraicko–geometrické. Díky tvrzení 2.5 jsou tedy Goppa kódy také zástupci Algebraicko–geometrických kódů. V této sekci uvedeme přímý popis divisorů sloužící ke konstrukci Goppa kódu $\Gamma_p(L, g)$ jako reziduálního Algebraicko–geometrického kódu.

Nechť $\mathbb{F}_{p^m}(x)/\mathbb{F}_{p^m}$ je racionální funkční těleso. Nechť $L = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_{p^m}^n$, kde $\alpha_i \neq \alpha_j$ pro $i \neq j$. Označme $P_i$ nulu $x - \alpha_i$ pro $1 \leq i \leq n$ a $P_\infty$ pól $x$. Označme $\mathcal{D}$ divisor

$$\mathcal{D} = P_1 + \ldots + P_n$$

a dále opět položme

$$h(x) = \prod_{i=1}^{n}(x - \alpha_i) \in \mathbb{F}_{p^m}[x].$$

**Tvrzení 2.6.** *Ať $g \in \mathbb{F}_{p^m}[x]$ je polynom stupně $t$, kde $0 < t < n$. Potom*

$$\Gamma_p(L, g) = \mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{A} - \mathcal{E})|_{\mathbb{F}_p},$$

*kde $\mathcal{A}$ a $\mathcal{E}$ jsou kladné divisory definované jako $\mathcal{A} = (h') + (n-1)P_\infty$, kde $h'$ značí derivaci $h$, a $\mathcal{E} = (g)_0$.*

*Důkaz.* Nejprve dokážeme, že $\mathcal{A}$ je skutečně kladný divisor. V definici $h$ vidíme rozklad na lineární činitele v $\mathbb{F}_{p^m}[x]$ a také, že polynom $h$ neobsahuje kořen násobnosti $p$. Tudíž je derivace $h$ nenulová. Navíc má $h'$ stupeň $\leq n-1$. Z předchozí poznámky 1.10 vyplývá $(h') \geq -(n-1)P_\infty$, neboť $\deg(h') \leq (n-1)$, a tedy $\mathcal{A} = (h') + (n-1)P_\infty \geq 0$. Z tvrzení 2.5 o vztahu mezi Goppa kódy a GRS kódy nám stačí dokázat pouze

$$\mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{A} - \mathcal{E}) = \mathrm{GRS}_{p^m}(L, B, n-t),$$

kde $B = (\beta_1, \ldots, \beta_n)$ je definované následovně

$$\beta_i = \frac{g(\alpha_i)}{\prod_{\substack{j=1 \\ j \neq i}}^{n} (\alpha_i - \alpha_j)}, \; 1 \leq i \leq n.$$

Nechť máme polynom $f \in \mathbb{F}_{p^m}[x]$ takový, že $\deg f < n-t$. Díky poznámce 1.10 máme $(f) \geq -(n-t-1)P_\infty$ a

$$\left(\frac{gf}{h'}\right) = (g) + (f) - (h') \geq (\mathcal{E} - tP_\infty) - (n-t-1)P_\infty - \mathcal{A} + (n-1)P_\infty \geq -(\mathcal{A} - \mathcal{E}).$$

Máme $\frac{gf}{h'} \in \mathcal{L}(\mathcal{A} - \mathcal{E})$ a pomocí definice $\beta_i$ a $h'(\alpha_i)$ dostáváme

$$\frac{g(\alpha_i) f(\alpha_i)}{h'(\alpha_i)} = \beta_i f(\alpha_i).$$

Získáváme

$$\mathrm{GRS}_{p^m}(L, B, n-t) \subseteq \mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{A} - \mathcal{E}).$$

Rovnost dokážeme tím, že ukážeme rovnost dimenzí kódů. Máme, dimenze kódu $\mathrm{GRS}_{p^m}(L, B, n-t)$ je $n-t$. Pro zjištění dimenze $\mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{A} - \mathcal{E})$ nejprve spočteme $\deg(\mathcal{A} - \mathcal{E})$. Z definice vyplývá $\deg(\mathcal{A}) = \deg((h')) + n - 1 = n - 1$, neboť stupeň hlavního divisoru je roven nule. Stupeň $\mathcal{E}$ je stejný jako stupeň polynomu $g$, tedy $t$. Dostáváme $\deg(\mathcal{A} - \mathcal{E}) = n - 1 - t$. Jelikož $t$ je z intervalu $0 < t < n$ a protože rod $g$ racionálního funkčního tělesa je roven nule, platí

$$2g - 2 = -2 < \deg(\mathcal{A} - \mathcal{E}) < n.$$

Díky tvrzení 1.39 získáváme rovnost

$$\dim \mathcal{C}_{\mathcal{L}}(\mathcal{D}, \mathcal{A} - \mathcal{E}) = \deg(\mathcal{A} - \mathcal{E}) + 1 - g = n - t,$$

čímž jsme toto tvrzení dokázali. □

## 2.3 Parametry

**Tvrzení 2.7.** *Nechť $\Gamma_p(L, g)$ je Goppa kód s $L = (\alpha_1, \ldots, \alpha_n)$ a $\deg g = t$. Potom má parametry $[n, k, d]_p$, kde $k \geq n - mt$ a $d \geq t + 1$*

*Důkaz.* Goppa kód $\Gamma_p(L, g)$ je alternantní GRS $[n, n-t, t+1]_{p^m}$ kódu. Parametry potom plynou z vlastností alternantních kódů v tvrzení 1.29. □

U binárních kódů lze zpřesnit odhad minimální vzdálenosti pro některé Goppovy polynomy.

**Tvrzení 2.8.** *Nechť $\Gamma_2(L, g)$ je Goppa kód s bezčtvercovým polynomem $g(x)$ stupně $t$. Potom minimální vzdálenost tohoto kódu je alespoň $2t + 1$.*

*Důkaz.* Nechť máme $L = (\alpha_1, \ldots, \alpha_n)$ a ať $c = (c_1, \ldots, c_n)$ je kódové slovo váhy $w \geq 0$. Označme

$$f(x) = \prod_{i=1}^{n} (x - \alpha_i)^{c_i}.$$

Potom $\deg f(x) = w$ a derivace $f'(x) = \sum_{i=1}^{n} \prod_{j \neq i} (x - \alpha_j)^{c_j}$. Platí

$$\frac{f'(x)}{f(x)} = \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i}.$$

Jelikož $\alpha_i$ jsou různá, jsou $f(x)$ a $f'(x)$ nesoudělné. Navíc $g(\alpha_i) \neq 0$, $1 \leq i \leq n$, a proto je také $f(x)$ nesoudělné s $g(x)$. Dostáváme

$$\frac{f'(x)}{f(x)} \equiv 0 \mod g(x) \Leftrightarrow g(x) \mid f'(x).$$

Zapišme $f(x)$ ve tvaru $f(x) = a(x)^2 + xb(x)^2$ pro nějaká $a(x)$, $b(x) \in \mathbb{F}_{2^m}[x]$. Máme $b(x) \neq 0$, neboť z konstrukce je $f(x)$ bezčtvercový polynom. Jelikož pracujeme v tělese charakteristiky 2, dostáváme $f'(x) = b(x)^2$, tedy $g(x) \mid b(x)^2$. Protože je $g(x)$ také bezčtvercový, máme $g(x) \mid b(x)$. Dostáváme

$$c \in \Gamma_2(L, g) \Leftrightarrow \frac{f'(x)}{f(x)} \equiv 0 \mod g(x) \Leftrightarrow g(x) \mid b(x).$$

Pokud $c \neq 0$, potom $\deg b(x) \geq \deg g(x) = t$. Tedy

$$\deg f(x) \geq 2 \deg g(x) + 1 = 2t + 1$$

a minimální vzdálenost pro binární Goppa kód s bezčtvercovým polynomem je $d \geq 2 \deg g(x) + 1 = 2t + 1$. □

Pokud je polynom $g(x)$ bezčtvercový, pak $\Gamma_2(L, g)$ nazýváme *separabilním Goppa kódem.*

**Tvrzení 2.9.** *Nechť $g(x) \in \mathbb{F}_{2^m}[x]$ je ireducibilní polynom stupně $t$ a nechť $\Gamma_2(L, g)$ je jemu odpovídající Goppa kód délky $n$ a dimenze $k$. Pokud $t < 2^{m/2-1}$, pak $k = n - mt$.*

*Důkaz.* Viz [18]. □

**Příklad 2.1.** *Zvolme*

$$p = 2,\ m = 3,\ n = p^m = 8,\ L = (0, 1, \alpha, \ldots, \alpha^6),$$

*kde $\alpha$ je primitivní prvek $\mathbb{F}_8$ a $\mathbb{F}_8 \simeq \mathbb{F}_2[\alpha]/(\alpha^3 + \alpha + 1)$. Zvolme ireducibilní polynom*

$$g(x) = x^2 + \alpha x + 1 \in \mathbb{F}_8[x]$$

*a zkonstruujme ireducibilní Goppa kód* $\Gamma = \Gamma_2(L, g)$ *délky* $n = |L| = 8$ *a dimenze* $k \geq 8 - 2 \cdot 3 = 2$. *Minimální vzdálenost je potom* $d \geq 2 \deg g(x) + 1 = 5$. *Prověrková matice* $H$ *nad* $\mathbb{F}_8$ *má tvar*

$$H = \begin{pmatrix} \frac{1}{g(0)} & \frac{1}{g(1)} & \frac{1}{g(\alpha)} & \cdots & \frac{1}{g(\alpha^6)} \\ \frac{0}{g(0)} & \frac{1}{g(1)} & \frac{\alpha}{g(\alpha)} & \cdots & \frac{\alpha^6}{g(\alpha^6)} \end{pmatrix} = \begin{pmatrix} 1 & \alpha^6 & 1 & \alpha^5 & \alpha^6 & \alpha^5 & \alpha^2 & \alpha^2 \\ 0 & \alpha^6 & \alpha & 1 & \alpha^2 & \alpha^2 & 1 & \alpha \end{pmatrix}.$$

*Prověrková* $6 \times 8$ *matice* $H'$ *nad* $\mathbb{F}_2$ *je potom*

$$H' = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

*Z matice* $H'$ *můžeme jednoduše získat generující* $2 \times 8$ *matici* $G$*:*

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

*Dostáváme tedy* $[8, 2, 2]_2$ *Goppa kód*

$$\Gamma_2(L, g) = \left\{ \begin{array}{l} (0,0,0,0,0,0,0,0), \\ (1,0,0,0,1,1,1,1), \\ (0,1,1,1,1,1,0,1), \\ (1,1,1,1,0,0,1,0) \end{array} \right\}$$

*schopný opravit 2 chyby.*

## 2.4 Divoké Goppa kódy

Zajímavé vlastnosti mají Goppa kódy tvaru $\Gamma_p(L, g^{p-1})$, kde $g \in \mathbb{F}_{p^m}[x]$ je monický bezčtvercový polynom stupně $t$. Takové Goppa kódy nazýváme *divoké Goppa kódy*. Veškeré vlastnosti vycházejí z následující věty:

**Věta 2.10.** [Sugiyama–Kasahara–Hirasawa–Namekawa 1976]
*Pro každý monický bezčtvercový polynom* $g \in \mathbb{F}_{p^m}[x]$ *stupně* $t$ *platí*

$$\Gamma_p(L, g^{p-1}) = \Gamma_p(L, g^p).$$

*Důkaz.* Nechť $S$ je multiplikativní množina tvořená polynomy $\mathbb{F}_{p^m}[x]$ nesoudělnými s $g$. Potom v $S^{-1}\mathbb{F}_{p^m}[x]$ platí

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod g^p \Rightarrow \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod g^{p-1}.$$

Tedy máme

$$c \in \Gamma_p(L, g^p) \Rightarrow c \in \Gamma_p(L, g^{p-1}).$$

Naopak uvažme kódové slovo $c = (c_1, \ldots, c_n) \in \Gamma_p(L, g^{p-1})$ a rozkladové nadtěleso $K$ tělesa $\mathbb{F}_{p^m}$ určené polynomem $g$. Potom z

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod g^{p-1}$$

vyplývá

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod (x - r_j)^{p-1} \tag{2.4}$$

pro každý kořenový činitel $(x - r_j)$, $1 \leq j \leq d$, polynomu $g$ a pracujeme v příslušném lokalizovaném okruhu $S^{-1}K[x]$, případně $S_j^{-1}K[x]$, kde $S_j$ je multiplikativní množina tvořená polynomy nesoudělnými s $x - r_j$. Zafixujme jeden kořen $r$ polynomu $g$ a uvažme polynom

$$s(x) = -\frac{1}{\alpha_i - r} - \frac{x - r}{(\alpha_i - r)^2} - \frac{(x - r)^2}{(\alpha_i - r)^3} - \ldots - \frac{(x - r)^{p-1}}{(\alpha_i - r)^p}.$$

Polynom $s(x)$ můžeme pomocí vzorce pro sčítání geometrické posloupnosti vyjádřit jako

$$s(x) = -\frac{1}{\alpha_i - r} \frac{(\frac{x-r}{\alpha_i - r})^p - 1}{\frac{x-r}{\alpha_i - r} - 1}.$$

Upravme

$$s(x) = -\frac{1}{\alpha_i - r} \frac{(\frac{x-r}{\alpha_i - r})^p - 1}{\frac{x-r}{\alpha_i - r} - 1} = -\frac{1}{\alpha_i - r} \frac{((x - r)^p - (\alpha_i - r)^p)(\alpha_i - r)}{(x - \alpha_i)(\alpha_i - r)^p} =$$

$$= \frac{1}{x - \alpha_i}\left(\left(\frac{\alpha_i - r}{\alpha_i - r}\right)^p - \left(\frac{x - r}{\alpha_i - r}\right)^p\right) = \frac{1}{x - \alpha_i}\left(1 - \left(\frac{x - r}{\alpha_i - r}\right)^p\right)$$

a dostaneme

$$s(x) \equiv \frac{1}{x - \alpha_i} \mod (x - r)^p,\ s(x) \equiv \frac{1}{x - \alpha_i} \mod (x - r)^{p-1}.$$

Po dosazení do rovnice (2.4) získáme

$$\sum_{i=1}^{n} \frac{c_i}{\alpha_i - r} + (x - r)\sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^2} + \ldots + (x - r)^{p-2}\sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^{p-1}} \equiv$$

$$\equiv 0 \mod (x - r)^{p-1}, \tag{2.5}$$

neboť $c \in \Gamma_p(L, g^{p-1})$. Porovnejme koeficienty polynomu na levé straně rovnice s nulovým polynomem na pravé straně a dostaneme

$$\sum_{i=1}^{n} \frac{c_i}{\alpha_i - r} = 0, \sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^2} = 0, \ldots, \sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^{p-1}} = 0.$$

Navíc platí

$$\sum_{i=1}^{n} \frac{c_i}{\alpha_i - r} = 0 \Rightarrow \sum_{i=1}^{n} \frac{c_i^p}{(\alpha_i - r)^p} = \sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^p} = 0,$$

neboť $c_i^p = c_i \in \mathbb{F}_p$, a dostáváme tedy i

$$(x-r)^{p-1} \sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^p} \equiv 0 \mod (x-r)^p.$$

Rovnici (2.5) proto můžeme zapsat jako

$$\sum_{i=1}^{n} \frac{c_i}{\alpha_i - r} + \ldots + (x-r)^{p-2} \sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^{p-1}} + (x-r)^{p-1} \sum_{i=1}^{n} \frac{c_i}{(\alpha_i - r)^p} = 0$$

a vidíme, že

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod (x-r)^p.$$

Získáme tedy soustavu rovnic

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod (x-r_1)^p$$

$$\vdots$$

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod (x-r_d)^p,$$

a za předpokladu, že $g$ má rozklad na lineární činitele $(x-r_j)$, dospějeme k závěru, že $g^p$ je součinem různých lineárních faktorů $(x-r_j)^p$. Pomocí zobecněné Čínské věty o zbytcích dostaneme

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod g^p$$

v $S^{-1}K[x]$, a tedy také

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \mod g^p$$

v $S^{-1}\mathbb{F}_{p^m}[x]$. □

### Parametry a výhody divokých Goppa kódů

Přestože jsou kódy $\Gamma_p(L, g^{p-1})$ a $\Gamma_p(L, g^p)$ z předchozí věty stejné, jsou alternantními kódy dvou různých GRS kódů. GRS kód asociovaný s Goppovým polynomem $g^{p-1}$ má vetší dimenzi, ale zároveň menší minimální vzdálenost než ten, který je asociovaný s $g^p$. Proto můžeme získat odhad na minimální vzdálenost. Dolní odhad pro minimální vzdálenost $d$ kódu $\Gamma_p(L, g^p)$ je $d \geq pt + 1$.

Výhodou divokých Goppa kódů je možnost dekódování $\lfloor pt/2 \rfloor$ chyb. Pro prvočíslo $p \in \{3, 5 \ldots\}$ dovolují Divoké kódy opravu více chyb než Goppa kód s ireducibilním polynomem stejného stupně $(p-1)t$, který umožňuje opravu $\lfloor (p-1)t/2 \rfloor$ chyb. Navíc mají tyto kódy stejnou dimenzi. Pro Goppův polynom tvaru $g^{p-1}$ má Goppa kód dimenzi alespoň $n - m(p-1)t$, tedy stejně jako pro ireducibilní

polynom stupně $(p-1)t$.

Divoký Goppa kód má parametry $[n, \geq n - m(p-1)t, \geq pt+1]_p$.

Více o divokých Goppa kódech lze nalézt v [19].

## 2.5 Dekódování

Goppa kódy a Reed–Solomonovy kódy mohou být efektivně dekódovány podobnou technikou využívající Euklidův algoritmus. Pro dekódování přijatého slova potřebujeme vědět souřadnice chyb a jejich hodnoty. V následující sekci předvedeme postup dekódování Goppa kódů využívající stejný princip, jakým je možné dekódovat kód $C_{p^m}(L, g)$.

Ať $\Gamma_p(L, g)$ je Goppa kód s $\deg(g(x)) = t$ a minimální vzdáleností $d \geq t+1$. Nechť $c$ je kódové slovo a ať slovo $r = (r_1, \ldots, r_n) = c + e$ vznikne přidáním chybového vektoru $e = (e_1, \ldots, e_n)$, $\mathrm{wt}(e) \leq \lfloor \frac{t}{2} \rfloor$ ke kódovému slovu $c = (c_1, \ldots, c_n)$. Potom máme

$$\sum_{i=1}^{n} \frac{r_i}{x - \alpha_i} = \sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} + \sum_{i=1}^{n} \frac{e_i}{x - \alpha_i}.$$

Jelikož $c$ je kódové slovo, dostáváme

$$\sum_{i=1}^{n} \frac{r_i}{x - \alpha_i} \equiv \sum_{i=1}^{n} \frac{e_i}{x - \alpha_i} \mod g(x).$$

Definujme *syndromový polynom* $S(x)$ slova $r$ jako

$$S(x) = \sum_{i=1}^{n} \frac{r_i}{x - \alpha_i}.$$

Platí

$$S(x) \equiv \sum_{i=1}^{n} \frac{e_i}{x - \alpha_i} \mod g(x).$$

Syndromový polynom je nulový právě tehdy, když $r \in \Gamma_p(L, g)$.

Nechť $E$ je množina všech indexů nenulových pozic vektoru $e = (e_1, \ldots, e_n)$. Definujme *lokační chybový polynom* $\sigma(x)$ a *chybově–evaluační polynom* $w(x)$ následovně:

$$\sigma(x) = \prod_{i \in E} (x - \alpha_i)$$

a

$$w(x) = \sum_{i \in E} e_i \prod_{j \in E \setminus \{i\}} (x - \alpha_j).$$

Pokud je chybový vektor nenulový, jsou také dané polynomy nenulové a máme $\deg w(x) < \deg \sigma(x) \leq t$ a $\mathrm{NSD}(\sigma(x), w(x)) = 1$, neboť pro všechny faktory $\sigma(x)$ platí $x - \alpha_i \nmid w(x)$. Jelikož

$$\sigma(x) S(x) = \prod_{i \in E} (x - \alpha_i) \sum_{i=1}^{n} \frac{r_i}{x - \alpha_i} \equiv$$

$$\equiv \prod_{i\in E}(x-\alpha_i)\sum_{i=1}^{n}\frac{e_i}{x-\alpha_i} = \sum_{i\in E} e_i \prod_{j\in E\setminus\{i\}}(x-\alpha_j) = w(x) \mod g(x),$$

dostáváme kongruenci

$$\sigma(x)S(x) \equiv w(x) \mod g(x). \tag{2.6}$$

Řešení této rovnice, a tedy i příslušných polynomů, určitě existuje, neboť předpokládáme, že počet chyb je menší než $\lfloor \frac{t}{2} \rfloor$, tedy menší než polovina minimální vzdálenosti kódu. Použitím derivace lokačně chybového polynomu

$$\sigma'(x) = \sum_{i\in E}\prod_{j\in E\setminus\{i\}}(x-\alpha_i)$$

získáme rovnici

$$w(\alpha_i) = e_i \prod_{j\in E\setminus\{i\}}(\alpha_i-\alpha_j) = e_i\sigma'(\alpha_i), \forall i \in E.$$

Dostáváme $e_i = \frac{w(\alpha_i)}{\sigma'(\alpha_i)}$. Z polynomů $\sigma(x)$ a $S(x)$ nejmenšího stupně splňující ekvivalenci (2.6) získáme chybový vektor a pomocí něj i původní kódové slovo. Souřadnice chybového vektoru $e$ jsou

$$e_i = \begin{cases} 0, & \text{pokud } \sigma(\alpha_i) \neq 0 \\ \frac{w(\alpha_i)}{\sigma'(\alpha_i)}, & \text{pokud } \sigma(\alpha_i) = 0. \end{cases}$$

Řešení kongruence (2.6) můžeme získat pomocí rozšířeného Euklidova algoritmu pro $g(x)$ a $S(x)$.

### Euklidův algoritmus

Než přistoupíme k samotnému dekódovacímu algoritmu, uvedeme Euklidův algoritmus a některé důležité vlastnosti. Všechny polynomy v této sekci, pokud nebude řečeno jinak, budeme chápat nad $\mathbb{F}_{p^m}$.

**Algoritmus 1** (Euklidův algoritmus).

`VSTUP:` *Polynomy* $a(x)$, $b(x) \in \mathbb{F}_{p^m}[x]$, $\deg a(x) \geq \deg b(x)$

`VÝSTUP:` $\mathrm{NSD}(a(x), b(x)) = s_{l-1}(x) = u_{l-1}(x)a(x) + v_{l-1}(x)b(x)$

*1.* $s_0(x) = a(x)$, $s_1(x) = b(x)$
$u_0(x) = 1$, $u_1(x) = 0$
$u_0(x) = 0$, $u_1(x) = 1$
$i = 1$

*2.* *Dokud* $s_i(x) \neq 0$ *dělej*
$i = i + 1$
*Spočti* $q_i(x)$, $s_i(x)$ *tak, že*
$s_{i-2}(x) = q_i(x)s_{i-1}(x) + s_i(x)$, $\deg s_i(x) < \deg s_{i-1}(x)$
*Spočti* $u_i(x)$, $v_i(x)$ *tak, že*
$u_{i-2}(x) = q_i(x)u_{i-1}(x) + u_i(x)$, $v_{i-2}(x) = q_i(x)v_{i-1}(x) + v_i(x)$

*3.* $l{=}i$

*VRAŤ* $s_{l-1}(x)$, $u_{l-1}(x)$, $v_{l-1}(x)$

Ve zbylé části sekce budeme používat hodnoty definované v algoritmu 1.

**Lemma 2.11.** *Sekvence polynomů použitých v Euklidově algoritmu splňují následující rovnice:*

1. $(-1)^{i+1}a(x) = v_i(x)s_{i-1}(x) - v_{i-1}(x)s_i(x)$, $1 \leq i \leq l-1$
2. $(-1)^i b(x) = u_i(x)s_{i-1}(x) - u_{i-1}(x)s_i(x)$, $1 \leq i \leq l-1$
3. $(-1)^i = u_i(x)v_{i-1}(x) - u_{i-1}(x)v_i(x)$, $1 \leq i \leq l-1$
4. $\deg u_i(x) + \deg s_{i-1}(x) = \deg b(x)$, $2 \leq i \leq l-1$
5. $\deg v_i(x) + \deg s_{i-1}(x) = \deg a(x)$, $1 \leq i \leq l-1$
6. $s_i(x) = u_i(x)a(x) + v_i(x)b(x)$, $0 \leq i \leq l$

*Důkaz.* Triviálně přímou indukcí podle $i$. □

U dekódovacího algoritmu nás především zajímá sekvence $v_i(x)$.

**Lemma 2.12.** *Ať $k, h \geq 0$ splňují rovnice*

$$k + h = \deg a(x) - 1,$$

$$h \geq \deg \mathrm{NSD}(a(x),\, b(x)).$$

*Potom existuje právě jedno $i$, $1 \leq i \leq l-1$, takové, že*

$$\deg v_i(x) \leq k, \tag{2.7}$$

$$\deg s_i(x) \leq h. \tag{2.8}$$

*Důkaz.* Stupně polynomů $s_i(x)$ s rostoucím $i$ ryze klesají, proto můžeme najít $i$, pro které platí

$$\deg s_i(x) \leq h \leq \deg s_{i-1}(x) - 1. \tag{2.9}$$

Z 5 rovnice lemma 2.11 máme

$$\deg v_i(x) \leq k \leq \deg v_{i+1}(x) - 1. \tag{2.10}$$

Tyto dvě rovnice implikují jak existenci, tak jednoznačnost takového $i$, neboť z rovnice (2.9) není možná menší hodnota $i$ a z rovnice (2.10) zase hodnota větší. □

Rovnice 6 z lemma 2.11 poskytuje vztah

$$v_i(x)b(x) \equiv s_i(x) \mod a(x).$$

Rovnice 5 společně s klesajícími stupni polynomů $s_i(x)$ implikuje

$$\deg v_i(x) + \deg s_i(x) < \deg a(x).$$

Pro dva polynomy $v(x), s(x) \in \mathbb{F}_q[x]$ pro které tyto dva vztahy platí, ukazuje následující lemma jednoznačnost až na nějaký společný faktor.

**Lemma 2.13.** *Ať $s(x)$ a $v(x)$ jsou dva polynomy splňující*

$$v(x)b(x) \equiv s(x) \mod a(x), \tag{2.11}$$

$$\deg v(x) + \deg s(x) < \deg a(x). \tag{2.12}$$

*Potom existuje $i$, $1 \leq i \leq l$, a polynom $\lambda(x)$ takové, že*

$$v(x) = \lambda(x)v_i(x), \tag{2.13}$$

$$s(x) = \lambda(x)s_i(x). \tag{2.14}$$

*Důkaz.* Ať $h = \deg s(x)$ a $k = \deg a(x) - \deg s(x) - 1$. Potom z rovnice (2.11) máme $\mathrm{NSD}(a(x), b(x)) \mid s(x)$, tedy $h \geq \deg \mathrm{NSD}(a(x), b(x))$. Ať $i$ je takové, že splňuje lemma 2.12 pro hodnoty $k$ a $h$. Potom z (2.9) vyplývá

$$\deg(s_{i-1}(x)) \geq h + 1 \geq \deg s_i(x) + 1.$$

Podobně (2.10) implikuje $\deg(v_{i+1}(x)) \geq k + 1 \geq \deg v_i(x) + 1$. Pokud tedy existuje $i$ splňující rovnice (2.13) a (2.14), potom je jedinečné a rovné $i$ z lemma 2.12. Rovnici (2.11) můžeme zapsat ve tvaru

$$s(x) = u(x)a(x) + v(x)b(x) \tag{2.15}$$

pro nějaké $u(x)$. Z lemma 2.11 rovnice 6 dostaneme

$$s_i(x) = u_i(x)a(x) + v_i(x)b(x),\ 0 \leq i \leq l. \tag{2.16}$$

Kombinací (2.15) a (2.16) získáme rovnici

$$s_i(x)v(x) \equiv s(x)v_i(x) \mod a(x),$$

kde $s_i(x)$ a $s(x)$ mají stupeň nejvýše $h$ a $v(x)$ spolu s $v_i(x)$ mají stupeň nejvýše $k$. Jelikož $k + h < \deg a(x)$, dostáváme

$$s_i(x)v(x) = s(x)v_i(x).$$

Dosazením do rovnic (2.15) a (2.16) získáme

$$u_i(x)v(x) = u(x)v_i(x).$$

Jelikož jsou z rovnice 3 lemma 2.11 polynomy $u_i(x)$ a $v_i(x)$ nesoudělné, máme $u(x) = \lambda(x)u_i(x)$ a $v(x) = \lambda(x)v_i(x)$ pro nějaký polynom $\lambda(x) \in \mathbb{F}_{p^m}[x]$. Dosazením do (2.15) a (2.16) dostáváme $s(x) = \lambda(x)s_i(x)$. □

### Dekódovací algoritmus

**Tvrzení 2.14.** *Nechť $\Gamma_p(L, g)$ je Goppa kód s $g(x)$ stupně $t$. Ať $\sigma(x)$ je lokační chybový polynom a $w(x)$ chybově–evaluační polynom. Nechť $e$ je chybový polynom váhy $\mathrm{wt}(e) \leq \lfloor \frac{t}{2} \rfloor$ a $S(x)$ je syndromový polynom. Potom*

$$\sigma(x) = \lambda v_i(x), \tag{2.17}$$

$$w(x) = \lambda s_i(x), \tag{2.18}$$

*kde $\lambda \in \mathbb{F}_{p^m}$ je konstanta taková, že polynom $\lambda v_i(x)$ je monický, kde $s_i(x)$ a $v_i(x)$ získáme z iterace Euklidova algoritmu pro $a(x) = g(x)$, $b(x) = S(x)$, kdy poprvé dostaneme $\deg s_i(x) < \lfloor \frac{t}{2} \rfloor$.*

*Důkaz.* Z lemma 2.13 pro polynomy $\sigma(x)$ a $w(x)$ dostáváme $\sigma(x) = \lambda(x)v_i(x)$ a $w(x) = \lambda(x)s_i(x)$. Jelikož jsou polynomy $\sigma(x)$ a $w(x)$ nesoudělné a $\sigma$ je monický polynom, musí být $\lambda(x) = \lambda \in \mathbb{F}_{p^m}$ takto definovaná konstanta. □

Nyní můžeme zformulovat dekódovací algoritmus pro Goppa kódy.

Nechť $\Gamma_p(L, g)$ je Goppa kód s Goppovým polynomem stupně $\deg g(x) = t$ a $L = (\alpha_1, \ldots, \alpha_n) \in \mathbb{F}_{p^m}^n$. Ať $r = c+e$ je slovo, kde $c$ je kódové slovo a $\mathrm{wt}(e) \geq \lfloor \frac{t}{2} \rfloor$. Potom pomocí tvrzení 2.14 následující algoritmus dekóduje slovo $r$.

**Algoritmus 2** (Dekódovací algoritmus pro Goppa kódy)**.**

`VSTUP:` $\Gamma_p(L, g)$, $\deg(g) = t$ *a* $r$ *jako výše*

`VÝSTUP:` $c \in \Gamma_p(L, g)$

`1.` *Spočti syndromový polynom* $S(x) \equiv \sum_{i=1}^{n} \frac{r_i}{x-\alpha_i} \mod g(x)$

`2.` *Iteruj Euklidův algoritmus pro* $a(x) = g(x)$ *a* $b(x) = S(x)$*, dokud není* $\deg s_i(x) < \lfloor \frac{t}{2} \rfloor$*. Ať* $\nu$ *je vedoucí koeficient* $v_i(x)$*. Polož* $\sigma(x) = \frac{v_i(x)}{\nu}$ *a* $w(x) = \frac{s_i(x)}{\nu}$*.*

`3.` *Nalezni množinu* $E = \{\alpha \mid \sigma(\alpha) = 0\}$*.*

`4.` *Urči hodnoty chybové vektoru* $e_\alpha = \frac{w(\alpha_i)}{\sigma'(\alpha_i)}$, $\forall \alpha_i$, $1 \leq i \leq n$*.*

`5.` *Urči* $e = (e_1, \ldots, e_n)$*, kde* $e_i = e_\alpha \Leftrightarrow \alpha \in E$ *a* $\alpha = \alpha_i$*, a jinak* $e_i = 0$*.*

`VRAŤ` $c = r - e$*.*

**Poznámka 2.1.** *Dělení* $\nu$ *ve druhém kroku není v algoritmu potřeba, neboť kořeny polynomu* $\sigma$ *zůstávají nezměněny a při hledání chybového polynomu potom ve čtvrtém kroku dojde k vykrácení* $\nu$*.*

### Dekódování binárních Goppa kódů

Nechť $\Gamma_2(L, g)$ je binární Goppa kód s ireducibilním polynomem $g(x) \in \mathbb{F}_{2^m}$ stupně $t$. Potom dle tvrzeni 2.8 máme minimální vzdálenost $d \geq 2t + 1$, avšak dekódovací algoritmus 2 dokáže opravit maximálně $\lfloor \frac{t}{2} \rfloor$ chyb. V binárním případě platí

$$w(x) = \sum_{i \in E} \prod_{j \in E \setminus \{i\}} (x - \alpha_j) = \sigma'(x)$$

a klíčová rovnice (2.6) má tvar

$$\sigma(x)S(x) \equiv \sigma'(x) \mod g(x). \tag{2.19}$$

Zapišme $\sigma(x)$ ve tvaru

$$\sigma(x) = \alpha^2(x) + x\beta^2(x), \tag{2.20}$$

kde

$$\deg \alpha(x) \geq \frac{t}{2} \text{ a } \deg \beta(x) \geq \frac{t-1}{2}. \tag{2.21}$$

Jelikož pracujeme v charakteristice 2, dostáváme

$$\sigma'(x) = \beta^2(x). \tag{2.22}$$

Pro $g(x)$ bezčtvercový můžeme pomocí Euklidova algoritmu nalézt inverz $T(x)$ polynomu $S(x)$ modulo $g(x)$, tedy $S(x)T(x) \equiv 1 \mod g(x)$. Dosazením do rovnic (2.19), (2.20) a (2.22) dostáváme

$$(T(x) + x)\beta^2(x) \equiv \alpha^2(x) \mod g(x). \tag{2.23}$$

Pokud je polynom $T(x)$ tvaru $T(x) = x$, dostáváme $\alpha(x) = 0$, a máme proto $\sigma(x) = x\beta^2(x)$. Jelikož $\sigma(x)$ je bezčtvercový, dostáváme $\beta(x) = 1$, a tedy také $\sigma(x) = x$. Pokud $T(x) \neq x$, označme polynom $R(x) \in \mathbb{F}_{2^m}[x]$, pro který platí $R^2(x) \equiv T(x) + x \mod g(x)$. Tento polynom je jednoznačně daný, neboť zobrazení $p(x) \mod g(x) \to p^2(x) \mod g(x)$ je v tělese charakteristiky 2 lineární a inverzní zobrazení posílá $T(x) + x$ na $R(x)$. Dosaďme do rovnice (2.23). Dostáváme

$$R^2(x)\beta^2(x) \equiv \alpha^2(x) \mod g(x),$$

a tedy

$$R(x)\beta(x) \equiv \alpha(x) \mod g(x). \tag{2.24}$$

Dekódovací algoritmus pro binární kódy postupuje stejně jako algoritmus 2. Euklidův algoritmus pro $g(x)$ a $R(x)$ nalezne $\alpha(x)$ a $\beta(x)$ splňující rovnice (2.21) a (2.24). Pro získané $\sigma(x) = \alpha^2(x) + x\beta^2(x)$ můžeme pokračovat algoritmem 2. Stačí nám kroky $3, 5$ a 6, neboť $e_\alpha = 1$, $\forall \alpha \in E$.

### Shrnutí

Předvedený způsob dekódování Goppa kódů nazýváme *Pattersonovou dekódovací metodou*, viz [20]. Pro Goppa kód $\Gamma_p(L, g)$, kde $\deg(g) = t$, jsme předvedli algoritmus schopný opravit $\lfloor \frac{t}{2} \rfloor$ chyb. V binárním případě, pokud je $g$ bezčtvercový, dokonce $t$ chyb. Pro binární Goppa kódy existuje řada dalších efektivnějších algoritmů. Přehled algoritmů pro binární Goppa kódy a počet chyb, které jsou schopny opravit, je možné nalézt v následující tabulce.

| Generic list–decoding [21] | Bernstein [22] | Binary list-decoding [21] |
|---|---|---|
| $\lfloor n - \sqrt{n(n-2t}\rfloor - 1$ | $n - \sqrt{n(n-2t-2)}$ | $\lfloor \frac{1}{2}(n - \sqrt{n(n-4t-2)}\rfloor - 1$ |

Tabulka 2.1: Porovnání dekódovacích algoritmů pro binární Goppa kody. Údaje v tabulce značí počty chyb, které daný algoritmus umožňuje opravit.

# Kapitola 3

# McEliecův kryptosystém

*McEliecův kryptosystém* je asymetrický kryptosystém prezentovaný v roce 1978 CalTechským matematikem Robertem McEliece. Tento zástupce post–kvantové kryptografie je založený na problému dekódování lineárních kódů, používající ve své původní verzi námi popsané Goppa kódy.

## 3.1 Popis kryptosystému

Uvedeme popis McEliecova kryptosystému bez explicitně použitého samoopravného kódu. Pro naše potřeby budeme považovat generující matici $G$ lineárního kódu nad $\mathbb{F}_q$ jako zobrazení $\mathbb{F}_q^k \to \mathbb{F}_q^n$, které zobrazuje zprávu $m$ délky $k$ na prvek $\mathbb{F}_q^n$. Hlavní myšlenka McEliecova kryptosystému spočívá v použití samoopravného kódu s efektivním dekódovacím algoritmem a jeho následném zamaskování jako lineárního kódu bez viditelné struktury.

**Poznámka 3.1.** *Původní návrh McEliecova kryptosystému [23] využívá binární ireducibilní Goppa kód* $\Gamma_2(L, g)$ *s parametry* $[1024, 524, 101]$*, pro který dle sekce 2.5 existuje rychlý dekódovací algoritmus.*

**Parametr kryptosystému:**

Lineární $[n, k, d]_q$ kód $C$ schopný opravit $t$ chyb, pro který existuje rychlý dekódovací algoritmus $D_C$.

**Tvorba klíčů:**

Nechť

- $S$ je $k \times k$ náhodná binární regulární matice,
- $G'$, $k \times n$ generující matice kódu $C$,
- $P$, $n \times n$ náhodná binární permutační matice,
- $G$, $k \times n$ matice taková, že $G = SG'P$.

Potom definujeme:

Veřejný klíč: $(G, t)$.
Soukromý klíč: $(S, D_C, P)$.

Permutační matice je taková matice, která má na každém řádku a v každém sloupci pouze jednu jedničku a zbytek jsou nuly. Matice $S$ nemá dle Canteauta a Chabaudeho [24] žádnou kryptografickou funkci. Slouží pouze k zamaskování generující matice kódu, aby nedošlo k prozrazení většiny bitů otevřeného textu. Matice $P$ je naopak nezbytná k zamaskování algebraické struktury kódu. Často bývá v soukromém klíči místo $S$ používána matice $S^{-1}$, neboť tuto inverzní matici, kterou potřebujeme u dešifrování, můžeme dopředu před-počítat.

**Šifrování a dešifrování:**

Nyní popíšeme jak probíhá šifrování a dešifrování u McEliecova kryptosystému. Pro jednoduchost budeme předpokládat, že otevřený text, zpráva $m$, má délku $k$. Pokud by byla velikost zprávy větší, je možné zprávu rozdělit na bloky odpovídající délky $k$ a postupovat stejně.

**Algoritmus 3** (McEliecův kryptosystém - šifrování)**.**
`VSTUP:` *$m \in \mathbb{F}_q^k$ zpráva, veřejný klíč $(G, t)$.*
`VÝSTUP:` *Šifrový text $c \in \mathbb{F}_q^n$.*
`1.` *Zakóduj $m$, tedy spočti $mG$.*
`2.` *Přidej ke kódovému slovu náhodný chybový vektor $e$ váhy $t$ a délky $n$.*
`VRAŤ` $c = mG + t$.

**Poznámka 3.2.** *U šifrovacího algoritmu dochází k zajímavé situaci, kdy se pojmy šifrování a kódování shodují. Rozdíl oproti kódování je ten, že struktura matice $G$ není všem známá a přenášené slovo dokáže dekódovat pouze příjemce, který je majitel soukromého klíče.*

Dešifrovací algoritmus k dekódování zprávy $y$ potřebuje znát chybový vektor $t$. Legitimní příjemce zprávy $c$ využije znalosti svého soukromého klíče a díky rychlému dekódovacímu algoritmu a původním parametrům $S$ a $P$ získá přístup k zprávě $m$. Debatu o možnostech útočníka necháme do další sekce.

**Algoritmus 4** (McEliecův kryptosystém - dešifrování)**.**
`VSTUP:` *Šifrový text $c = mG + e \in \mathbb{F}_q^n$, soukromý klíč $(S, D_C, P)$.*
`VÝSTUP:` *Zpráva $m$.*
`1.` *Spočti $cP^{-1} = mSG + eP^{-1}$.*
`2.` *Použij dekódovací algoritmus $D_C$ k nalezení $mS$.*
`3.` *Spočti $mSS^{-1}$.*
`VRAŤ` *$m$.*

**Tvrzení 3.1.** *Algoritmus 4 funguje.*

*Důkaz.* Máme

$$c' = cP^{-1} = mGP^{-1} + eP^{-1} = mSG' + eP^{-1}.$$

Je-li $P$ permutační matice, potom $eP^{-1}$ má váhu $t$. Kód $C$ dokáže opravit $t$ chyb, tedy

$$m' = D_C(c') = mS$$

a zprávu získáme snadno jako $m = m'S^{-1} = mSS^{-1}$.

□

## 3.2 Niederreiterův kryptosystém

Harald Niederreiter [25] v roce 1986 zveřejnil obdobu McEliecova kryptosystému, využívající místo generující matice matici prověrkovou. *Niederreiterův kryptosystém* v původním článku používá místo Goppa kódů zobecněné Reed–Solomonovy kódy. Bezpečnost kryptosystému s těmito kódy však byla prokázána Sidelnikovem a Shestakovem v roce 1992 jako nevyhovující. Útok nahlédneme v sekci 3.3.2. Niederreitova varianta při použití binárních Goppa kódů je však stejně bezpečná jako McEliecův kryptosystém [26].

**Parametry kryptosystému:**

Lineární $[n, k, d]_q$ kód $C$ schopný opravit $t$ chyb, pro který existuje rychlý dekódovací algoritmus $D_C$.

**Tvorba klíčů:**

Nechť

- $S$ je $(n-k) \times (n-k)$ náhodná binární regulární matice,
- $H'$, $(n-k) \times n$ prověrková matice matice kódu $C$,
- $P$, $n \times n$ náhodná binární permutační matice,
- $H$, $(n-k) \times n$ matice taková, že $H = SH'P$.

Potom definujeme:

Veřejný klíč: $(H, t)$.
Soukromý klíč: $(S, D_C, H, P)$.

**Šifrování a dešifrování:**

Nyní popíšeme, jak probíhá šifrování a dešifrování u tohoto alternativního kryptosystému. Na rozdíl od McEliecova kryptosystému budeme potřebovat, aby otevřený text $m$ byl předem zakódován do vektoru $e \in \mathbb{F}_q^n$ váhy $t$, tedy do chybového vektoru nulového slova. Místo do kódového slova potom následující algoritmus zašifruje otevřený text $e$ do syndromu za pomoci veřejné prověrkové matice $H$.

**Algoritmus 5** (Niederreiterův kryptosystém - šifrování).
`VSTUP:` *Veřejný klíč* $(H, t)$ *a* $m \in \mathbb{F}_q^l$.
`VÝSTUP:` *Šifrový text* $c \in \mathbb{F}_q^{n-k}$.
`1.` *Zakóduj zprávu* $m$ *do vektoru* $e$ *velikosti* $n$ *a váhy* $t$.
`2.` *Spočti šifrový text, syndrom* $c = He^\top$.
`VRAŤ` $c$.

Pro dešifrování zprávy musíme najít vektor $e$ váhy $t$ vedoucí na syndrom $c$ vzhledem k prověrkové matici $H$. Stejně jako u McEliecova systému využijeme soukromého klíče a rychlého dekódovacího algoritmu.

**Algoritmus 6** (Niederreiterův kryptosystém - dešifrování).
VSTUP: *Syndrom* $c = He^\top \in \mathbb{F}_q^{n-k}$, *soukromý klíč* $(S, D_C, H, P)$
VÝSTUP: *Zpráva* $m$.
1. *Spočti* $c' = S^{-1}c$.
2. *Použij dekódovací algoritmus* $D_C$ *k nalezení* $e' = Pe^\top$.
3. *Spočti* $P^{-1}e'$.
4. *Dekóduj* $e$ *na zprávu* $m$.
VRAŤ $m$.

**Tvrzení 3.2.** *Algoritmus 6 funguje.*

*Důkaz.* Máme $c' = S^{-1}c = S^{-1}He^\top = S^{-1}SH'Pe^\top = H'Pe^\top$. Jelikož $P$ neovlivní váhu $m$, získáme pomocí $D_C$ slovo $Pe^\top$. Potom stačí pouze vynásobit $P^{-1}$ a dostáváme $m^\top$. □

Nevýhodou Niederreiterova systému je nutnost kódování otevřeného textu na chybový vektor délky $n$. V binárním případě můžeme využít například následující funkce: $\phi_{n,t} : \mathbb{F}_2^l \to W_{2,n,t}$, kde $l = \lfloor \log_2 \binom{n}{t} \rfloor$ a $W_{2,n,t} \subset \mathbb{F}_2^n$ značí množinu slov váhy $t$.

**Algoritmus 7** (Kódování $\phi_{n,t} \to W_{2,n,t}$).
VSTUP: *Otevřený text* $m \in \mathbb{F}_2^l$, $n$ *a* $t$.
VÝSTUP: *Slovo* $e = (e_1, \ldots, e_n)$ *váhy* $t$ *a délky* $n$.
1. $c \leftarrow \binom{n}{t}$, $c' \leftarrow 0$, $j \leftarrow n$, $t' \leftarrow t$, $i \leftarrow m$ *v decimálním základu.*
2. *Dokud* $j > 0$ *dělej*
   $c' \leftarrow c\,\frac{j-t'}{j}$,
   *pokud* $i \leq c'$, *potom* $e_j \leftarrow 0$ *a* $c \leftarrow c'$,
   *jinak* $e_j \leftarrow 1$, $i \leftarrow i - c'$, $c \leftarrow c\,\frac{t'}{j}$ *a* $t' \leftarrow t' - 1$,
   $j \leftarrow j - 1$.
VRAŤ $m$.

Inverzní prvek $\phi_{n,t}^{-1}(e)$ lze získat jako

$$\phi_{n,t}^{-1}(e) = \sum_{i=1}^{n} \left( e_i \binom{i-1}{\sum_{j=1}^{i} e_j} \right) + 1,$$

kde

$$\begin{aligned} \binom{n}{k} &= \tfrac{n!}{k!(n-k)!} \text{ pro } n \geq k, \\ &= 0 \text{ jinak.} \end{aligned}$$

Algoritmus 7 má polynomiální složitost $\mathcal{O}(n^2 \log_2 n)$, viz [3].

Výhodou Niederreiterova systému je, že pro stejné parametry jako pro McElicův systém můžeme použít matici $H$ veřejného klíče ve formě $(I \mid J)$ a uložit pouze část $J$. Dojde tak k redukci velikost klíče při stejných bezpečnostních nárocích.

Niederreiterův veřejný klíč je tedy $n/(n-k)$ krát menší než McEliceho verze, jelikož Niederreiterova matice má velikost $(n-k) \times k$ a McEliece $n \times k$. Systematická forma matice $H$ a malá váha vektoru $m$ také značně snižuje výpočetní složitost šifrování.

**Příklad 3.1.** *Pokračování příkladu 2.1.*

*Máme* $[8,2,2]_2$ *Goppa kód s generující maticí*

$$G' = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}.$$

### Tvorba klíčů

*Zvolme:*

$$S = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}, \; P = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

*a spočtěme*

$$G = SG'P = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

*Veřejný klíč je tedy* $(G, 2)$ *a soukromý klíč* $(S, D_C, P)$.

### Šifrování

*Zvolme zprávu* $m = (1,0)$*, kterou chceme zašifrovat. Zvolme chybový vektor*

$$e = (0,1,0,0,0,0,0,1)$$

*váhy* 2 *a zašifrujme zprávu* $m$*:*

$$c = mG + e = (1,0)\begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} + (0,1,0,0,0,0,0,1)$$
$$= (1,0,1,1,1,1,0,1).$$

### Dešifrování

*Spočtěme*

$$cP^{-1} = (1,0,1,1,1,1,0,1)\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix} = (1,1,0,1,1,0,1,1),$$

*a dekódujme, například pomocí algoritmu 2 slovo $cP^{-1}$. Získáme $mS = (1,1)$. Nyní již spočtěme*

$$mSS^{-1} = (1,1)\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} = (1,0)$$

*a dostaneme danou zprávu $m$.*

## 3.3 Bezpečnost

V této sekci přiblížíme bezpečnost McEliecova kryptosystému. Uvedeme předpoklady bezpečnosti a poté další část zaměříme na útoky pomocí dekódování a rozpoznávání struktury kódu. V závěru uvedeme útoky na protokol použitý u McEliecova kryptosystému.

Prolomení bezpečnosti McEliecova kryptosystému odpovídá vyřešení následujícího problému:

**Problém 3.1.** *[McEliecův problém]*
*Nechť $C$ je lineární $[n,k]$ kód schopný opravit $t$ chyb s generující maticí $G$. Pro daný veřejný klíč $(G,t)$ a šifrový text $c$ nalezni zprávu $m$ takovou, že platí* $\mathrm{wt}(mG - c) = t$.

Bezpečnost potom určují dva hlavní předpoklady: obtížnost dekódování lineárního kódu a problém rekonstrukce kódu.

#### Obtížnost dekódování lineárního kódu

Jak jsme nahlédli v sekci 1.2.2, dekódování na nejpodobnější slovo u obecného lineárního kódu $C$, tedy kódu bez viditelné struktury, je obtížný problém. U binárních kódů je tento problém dokonce $\mathcal{NP}-$těžký. Bohužel tyto výsledky nám nedávají příliš informací o bezpečnosti McEliecově kryptosystému, neboť:

1. Kód v McEliecově kryptosystému není náhodný lineární kód, ale je permutačně ekvivalentní se zástupcem specifické třídy lineárních kódů, například Goppa kódů.

2. Problém 1.1 dekódování na nejpodobnější slovo není potřeba vyřešit v obecném případě, ale pouze pro použitou specifickou třídu kódů, a navíc stačí dekódovat pouze $t$ chyb.

3. V kryptografických aplikacích je správným ukazatelem složitost útoků udávána v průměrném případě. Složitost dekódování na nejpodobnější slovo u binárních kódů jako $\mathcal{NP}-$těžkého problému nám proto nedává příliš informací o složitosti útoku v průměrném případě.

Krom obecných předpokladů složitosti lze studovat bezpečnost i vzhledem k nejlepším známým útokům. Útoky, kde je snahou dekódovat šifrový text $c$ v kódu bez viditelné struktury, nazýváme *dekódovací útoky*. Jak jsme uvedli v sekci 1.2.2, nejlepší dekódovací útoky jsou ty, které jsou založené na dekódování s informační množinou.

### Problém rekonstrukce kódu

Druhým předpokladem je obtížnost získání původní struktury kódu z veřejného klíče, tedy například získání generující matice $G'$ z $G$. Útokům zaměřujícím se na strukturu kódu říkáme *strukturální útoky*. Obtížnost těchto útoků závisí na třídě použitých lineárních kódů. Pro některé třídy, například GRS kódy, jsou známé efektivní útoky, zatímco pro jiné, například binární Goppa kódy, známé nejsou.

## 3.3.1 Dekódovací útoky

Nechť je dán McEliecův kryptosystém s veřejným klíčem $(G, t)$. Budeme chtít opravit $t$ chyb u šifrového slova z McEliecova kryptosystému, kde známe pouze generující matici $G$ (nebo prověrkovou matici). Nejefektivnější známé algoritmy proti McEliecovu a Niederreiterovu kryptosystému jsou založené na *dekódování s informační množinou*. Hlavní myšlenka tohoto algoritmu byla navržena McEliecem [23] a existuje k ní mnoho variant. Ostatní algoritmy s menším úspěchem využívají například iterativní dekódování nebo statistické dekódování [3].

**Definice 3.3.** *Nechť $C$ je $[n, k, d]_q$ kód. Ať $G$ je generující maticí $C$. Označme $s_1, \ldots, s_n$ sloupce matice $G$, $I$ neprázdnou $k$ prvkovou podmnožinu $\{1, \ldots, n\}$ a $G_I$ matici tvořenou postupně sloupci $s_i$, $i \in I$. Množinu $I$ nazveme* informační množinou*, pokud je $G_I$ invertibilní matice.*

Pro každý vektor $y \in \mathbb{F}_q^n$ označme $y_I$ vektor tvořený souřadnicemi $y$ s indexy z $I$. Ať $y = c + e$, kde $c$ je kódové slovo $C$ a $e \in \mathbb{F}_q^n$, $\mathrm{wt}(e) = t$. V následující sekci budeme používat stejné značení.

### Princip dekódování s informační množinou

Nechť $G$ je ve standardním tvaru. Uvažme kódové slovo $c = mG$, $m \in \mathbb{F}_q^k$. Jelikož tvoří prvních $k$ sloupců generující matice $G$ jednotkovou matici, odpovídá prvních $k$ souřadnic slova $c$ zprávě $m$.

Nechť není nyní $G$ generující matici ve standardním tvaru. Ať $I$ je $k$ prvková informační množina. Pokud je $k \times k$ matice $G_I$ invertibilní, pak $G_I^{-1}G$ a $G$ generují stejný kód, neboť $G_I^{-1}G$ odpovídá matici $G$ po Gaussově eliminaci na sloupce s indexy z $I$. Navíc platí $m = v_I$, kde $v = mG_I^{-1}G$.

Dekódování pomocí informační množiny v *klasické formě* má na vstupu vektor $y = c + e$, $c \in C$, $e \in \mathbb{F}_q^n$, $\mathrm{wt}(e) = t$. Předpokládejme, že platí $y_I = c_I$. Jinými slovy na souřadnicích z $I$ nejsou chyby. Potom $y_I G_I^{-1}$ odpovídá vzoru $c$ podle lineárního zobrazení $K_G$ daného

$$\begin{aligned} K_G : \quad & \mathbb{F}_q^k \to \mathbb{F}_q^n \\ & m \mapsto mG. \end{aligned}$$

Dostáváme tedy $c = y_I G_I^{-1} G$.

Pokud $e_I \neq 0$, potom můžeme podle Lee a Brickella [28] stále dekódovat $y$ za podmínky, že připustíme malý počet chyb. Nechť máme $\mathrm{wt}(e_I) = r$, $0 \leq r \leq t$.

Algoritmus 8 ukazuje hlavní myšlenku tohoto principu. Pokud $e_I \neq 0$, potom dekódování nazývá *zobecněné dekódování s informační množinou.* Leon [29] navrhl zlepšení pomocí hledání kódových slov s nulami na $s$ po sobě jdoucích souřadnicích. Stern [30] poté navrhl rozdělit informační množinu do dvou částí pro zrychlené hledání kódových slov s nulami na $s$ po sobě jdoucích souřadnicích pomocí *narozeninového útoku.* Obrázek 3.1 zobrazuje dané varianty pro porovnání.

**Algoritmus 8** (Dekódování s informační množinou (pro parametr $r$)).

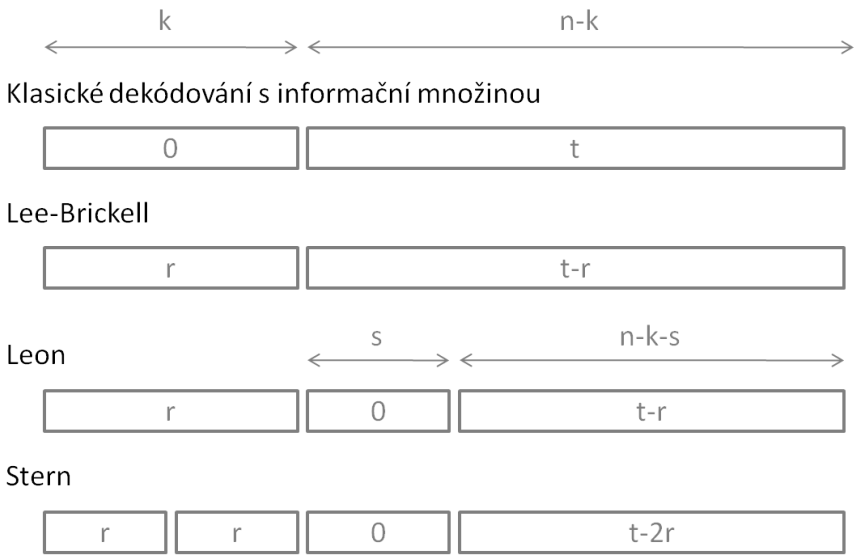
`VSTUP:` $[n,k]_q$ *kód schopný opravit* $t$ *chyb s* $k \times n$ *generující maticí* $G$, *přijaté slovo* $y = mG + e, \mathrm{wt}(e) \leq t$ *a parametr* $0 \leq r \leq t$.

`VÝSTUP:` *Zpráva* $m$.

`1.` *Zvol informační množinu* $I$ *a spočti* $E = c - c_I G_I^{-1} G$.

`2.` *Zvol* $e' \in \mathbb{F}_q^{|I|}$ *váhy* $\leq r$ *a spočti* $\epsilon = E - e' G_I^{-1} G$.
*Pokud* $\mathrm{wt}(\epsilon) \leq t$, *potom vrať* $\epsilon$ *a skonči, jinak běž na 2.*

`3.` *Běž na 1.*




Obrázek 3.1: Rozmístění nenulových prvků v chybovém vektoru (číslo v obdélníku značí Hammingovu váhu dané části).

Na obrázku 3.1 jsme uvedli různé přístupy pro rozložení váhy chybového vektoru. Každé volbě parametrů odpovídá pravděpodobnost, s jakou odpovídá naše rozložení skutečnému rozložení váhy chybového vektoru. Pravděpodobnost úspěchu dané iterace algoritmu je

$$P_{\text{klasické dekódování}} = \frac{\binom{n-k}{t}}{\binom{n}{t}},\ P_{\text{Lee-Brickell}} = \frac{\binom{k}{r}\binom{n-k}{t-r}}{\binom{n}{t}},$$

$$P_{\text{Leon}} = \frac{\binom{k}{r}\binom{n-k-s}{t-r}}{\binom{n}{t}},\ P_{\text{Stern}} = \frac{\binom{k/2}{r}^2\binom{n-k-s}{t-2r}}{\binom{n}{t}}$$

Celková složitost každého algoritmu odpovídá složitosti jedné iterace algoritmu vydělené příslušnou pravděpodobností úspěchu, respektive vynásobenou očekávaným počtem iterací algoritmu. Všechny útoky mají exponenciální složitost v závislosti na vstupních parametrech. Klasickému dekódování s informační množinou

dominuje inverze matic. Algoritmus má proto složitost

$$\mathcal{O}\Big(k^3 \frac{\binom{n}{t}}{\binom{n-k}{t}}\Big).$$

V případě Sternova útoku v základní verzi odpovídá složitost útoku

$$\mathcal{O}\Big(\frac{4^{r+1} srk^2(n-k)(n-\frac{k+1}{2})\binom{k/2}{p}^3\binom{n-k}{s}\binom{n}{k}}{2^{s+1}\binom{2r}{r}\binom{t}{2r}\binom{k-t}{k-2r}\binom{n-k-t+2r}{t}}\Big), \text{ viz [31].}$$

Tabulka 3.1 dokumentuje zlepšování dekódovacích útoků s informační množinou proti McEliecovu kryptosystému s původními parametry. Všechny útoky mají nicméně stále exponenciální složitost. Obranou tedy je nepoužít původní parametry, ale parametry vyšší.

| Rok | Útok | Složitost útoku |
|---|---|---|
| 1978 | Klasické dekódování s informační množinou | $\mathcal{O}(2^{80,7})$ |
| 1989 | Stern | $\mathcal{O}(2^{60,8})$ |
| 2008 | Bernstein, Lange a Peters [32] | $\mathcal{O}(2^{60,5})$ |
| 2009 | Finiasz a Sendrier [33] | $\mathcal{O}(2^{59,9})$ |
| 2013 | Becker, Joux, May a Meurer [31] | $\mathcal{O}(2^{52,50})$ |

Tabulka 3.1: Zlepšování útoků s informační množinou vůči McEliecově kryptosystému s binárním ireducibilním $[1024, 524]$ kódem schopným opravit 50 chyb.

### 3.3.2 Strukturální útoky

Strukturální útoky závisejí na třídě použitých kódů v McEliecově kryptosystému. V této sekci budeme popisovat strukturální útoky proti McEliecově kryptosystému. Při použití ireducibilních binárních Goppa kódů není znám žádný efektivní algoritmus pro rozložení $G$ na $(S, G', P)$. Nastíníme, proč takové rozložení není jednoduché. V závěru uvedeme strukturální útok na Niederreiterův kryptosystém s GRS kódy.

Binární Goppa kód $\Gamma_2(L, g)$ délky $n = 2^m$, který je schopný opravit $t$ chyb, obsahuje:

- generátor, monický ireducibilní polynom $g(x)$ stupně $t$ nad $\mathbb{F}_{2^m}$
- vektor $L$ tvořený $n$ po dvou různých prvků $\mathbb{F}_{2^m}$. V tomto případě vektor $L$ odpovídá permutaci prvků $\mathbb{F}_{2^m} = \mathbb{F}_n$.

Pokud známe jednu z těchto komponent, můžeme získat tu druhou v polynomiálním čase následovně:

1. Nechť známe množinu $L$, potom můžeme získat polynom $g(x)$ z rovnice

$$\sum_{i=1}^{n} \frac{c_i}{x - \alpha_i} \equiv 0 \bmod g(x)$$

   Euklidovým algoritmem, neboť kódová slova pro použití v rovnici můžeme získat z generující matice $G$.

2. Nechť známe polynom $g(x)$, potom můžeme sestrojit generující matici $G'$ Goppa kódu $\Gamma_2(L', g)$, kde $L' = (\alpha_1, \ldots, \alpha_n)$ je zvoleno náhodně, a použít algoritmus [34] na zjištění $L$ se složitostí $\mathcal{O}(n^3)$. Tento algoritmus využívá faktu, že $\Gamma_2(L, g)$ odpovídá permutovanému kódu $\Gamma_2(L', g)$.

Při použití obou způsobů získáváme náročný útok na McEliecův kryptosystém. Buď musíme procházet všechny permutace souřadnic $L$, pro který potom získáme polynom $g$ [34], nebo musíme procházet všechny polynomy stupně $t$ a pro ně potom získat $L$ [35]. Pro dané parametry Goppa kódu máme přibližně

$$2^{tm}/t = n^t/t$$

ireducibilních polynomů stupně $t$ nad $\mathbb{F}_{2^m}$ a $n! = \mathcal{O}(\sqrt{n}(\frac{n}{e})^n)$ permutací. Proto je druhý útok efektivnější. Díky dalším modifikacím uvedeným v [3] lze získat složitost tohoto strukturálního útoku

$$\mathcal{O}\Big(\frac{n^t}{t \log n}\Big) = \mathcal{O}\Big(\frac{2^{tm}}{tm}\Big).$$

**Strukturální útok na Niederreiterův kryptosystém s GRS kódy**

Použití GRS kódů v Niederreiterově kryptosystému není bezpečné. Sidelnikov a Shestakov v roce 1992 předvedli útok na tento kryptosystém se složitostí

$$\mathcal{O}(s^4 + sn).$$

V následujícím textu nastíníme princip útoku. Podrobný popis útoku je možné nalézt v [27].

Zapišme prověrkovou $(n-k) \times n$ matici $H$ kódu $\mathrm{GRS}_q(L, B, k)$ ve tvaru

$$\bar{H}^T = \begin{pmatrix} \beta_1'\alpha_1^0 & \beta_1'\alpha_1^1 & \ldots & \beta_1'\alpha_1^s \\ \beta_2'\alpha_1^0 & \beta_2'\alpha_2^1 & \ldots & \beta_2'\alpha_2^s \\ \vdots & \vdots & \ddots & \vdots \\ \beta_n'\alpha_n^0 & \beta_n'\alpha_n^1 & \ldots & \beta_n'\alpha_n^s \end{pmatrix},$$

kde $s = n-k-1$. Veřejný klíč Niederreiterova kryptosystému má tvar $H' = S\bar{H}P$, kde $S$ je regulární a $P$ permutační matice. Matice $P$ nemění strukturu $\bar{H}^T$, proto jí prozatím neuvažujme. Na prvky matice $H'$ lze nahlížet jako na $\beta_j'$ vynásobené hodnotami polynomů $S_i(x)$, $1 \le i \le s$, $1 \le j \le n$, jejíž koeficienty reprezentuje $i$-tý sloupec matice $S$. Proto

$$H'^T = \begin{pmatrix} \beta_1'S_1(\alpha_1) & \beta_1'S_2(\alpha_1) & \ldots & \beta_1'S_s(\alpha_1) \\ \beta_2'S_1(\alpha_2) & \beta_2'S_2(\alpha_2) & \ldots & \beta_2'S_s(\alpha_2) \\ \vdots & \vdots & \ddots & \vdots \\ \beta_n'S_1(\alpha_n) & \beta_n'S_2(\alpha_n) & \ldots & \beta_n'S_s(\alpha_n) \end{pmatrix},$$

kde $S_i(x) = \sum_{j=0}^{s} S_{ji}x^j$. Sidelnikov a Shestakov ukázali, že každý řádek matice $H'^T$ lze reprezentovat polynomem o jedné neznámé $\alpha_i$. Díky tomu zkonstruovali systém polynomiálních rovnic, který vedl na privátní klíč.

Pro útok je potřeba uvažovat matice $\bar{H}$, $S$, $H'$ nad $\mathbb{F} = \mathbb{F}_q \cup \infty$, kde máme $\frac{1}{\infty} = 0$, $\frac{1}{0} = \infty$. Pro každý polynom $f(x) = \sum_{j=0}^{s} f_j x^j \in F_q[x]$ stupně $\leq s$ položme $f(\infty) = f_s$. Potom Sidelnikov a Shestakov ukázali, že pro každý $\mathbb{F}-$automorfismus

$$\psi(x) = \begin{cases} \frac{a}{c} \text{ pokud } c \neq 0,\ x = \infty, \\ \frac{ax+b}{cx+d} \text{ jinak,} \end{cases} \quad \text{kde } a, b, c, d \in \mathbb{F}_q,\ ad - bc \neq 0$$

existují $\hat{\beta}_1, \ldots, \hat{\beta}_n$ a matice $S'$ taková, že

$$H' = S(S')^{-1} \begin{pmatrix} \hat{\beta}_1\psi(\alpha_1)^0 & \hat{\beta}_1\psi(\alpha_1)^1 & \ldots & \hat{\beta}_1\psi(\alpha_1)^s \\ \hat{\beta}_2\psi(\alpha_2)^0 & \hat{\beta}_2\psi(\alpha_2)^1 & \ldots & \hat{\beta}_2\psi(\alpha_2)^s \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\beta}_n\psi(\alpha_n)^0 & \hat{\beta}_n\psi(\alpha_n)^1 & \ldots & \hat{\beta}_n\psi(\alpha_n)^s \end{pmatrix}^T.$$

Navíc pro každé $\alpha_1, \alpha_2, \alpha_3 \in \mathbb{F}_q$ je možné nalézt takový automorfismus $\psi(x)$, že

$$\begin{array}{lcl} \psi(\alpha_1) = & 1 & = x_1 \\ \psi(\alpha_2) = & 0 & = x_2 \\ \psi(\alpha_3) = & \infty & = x_2 \\ \psi(\alpha_2) & & = x_j,\ j \notin \{1,\ 2,\ 3\} \end{array}$$

Potom ukázali, že $H'$ definuje kód s kódovými slovy

$$(\hat{\beta}_1 f(1), \hat{\beta}_2 f(0), \hat{\beta}_3 f(\infty), \hat{\beta}_4 f(\alpha_4), \ldots, \hat{\beta}_n f(\alpha_n)),$$

kde $f$ je polynom nad $F_q$ stupně $\geq s$. Připomeňme $x_3 = \infty$, tudíž $x_i \neq \infty$ pro $i \neq 3$.

Myšlenka útoku je tedy následující: Pokud vezmeme dvě kódová slova, které mají na $s-1$ pozicích nuly, potom jím odpovídající polynomy $\phi_1$ a $\phi_2$, které tyto kódová slova definují, mají $s-1$ společných faktorů a oba mají stupeň $\geq s$. Navíc můžeme předpokládat, že $\phi_1(0) = 1 = \phi_2(1)$ a $\phi_1(1) = 0 = \phi_2(0)$, což vede na rovnici

$$\frac{\phi_1(x_j)}{\phi_2(x_j)} = \frac{\phi_1(\infty)}{\phi_2(\infty)} \frac{(x_j - 1)}{x_j} = \frac{\phi_1(\alpha_3)}{\phi_2(\alpha_3)} \frac{(x_j - 1)}{x_j}.$$

Řešením rovnice jsme schopni zjistit $\alpha_j$, která nejsou kořeny $\phi_1$ ani $\phi_2$. Tento postup můžeme opakovat pro další kódová slova, čímž získáme všechna $\alpha_i$, kde $1 \leq i \leq n$. Potom při použití takového automorfismu $\psi$, pro který $\psi(\alpha_i)$, kde $1 \leq i \leq n$ není žádný prvek nekonečnem, jsou schopni získat $z_i$, $1 \leq i \leq n$ pomocí Gaussovy eliminace a předpokladu $\beta'_1 = 1$.

**Poznámka 3.3.** *Útok na Niederreiterův kryptosystém s* GRS *kódy nelze aplikovat na McEliecův, nebo Niederreiterův kryptosystém s Goppa kódy. Přestože pro každý Goppa kód existuje prověrková matice $\bar{H}$ nějakého* GRS *kódu, nelze pro ně analogicky vyjádřit matici $H'$. Při použití Goppa kódů můžeme totiž na prověrkovou matici nahlížet jako na matici pouze nad $\mathbb{F}_2$. Máme proto různé matice $S$. Pro* GRS *kód máme $(s+1) \times (s+1)$ matici nad $\mathbb{F}_{2^m}$, zatímco pro Goppa kódy máme $S$ $m(s+1) \times m(s+1)$ matici. Matice $H'$ proto nemá žádnou viditelnou strukturu, pokud je matice $S$ neznámá.*

### 3.3.3 Protokolové útoky

Tuto sekci zaměříme na útoky proti McEliecovu protokolu. Útoky budou vyžadovat dodatečnou informaci, například znalost části otevřeného textu nebo orákulum, které dokáže dešifrovat jakýkoliv šifrový text kromě cílového.

#### Útok se znalostí části otevřeného textu

Částečná znalost otevřeného textu snižuje výpočetní složitost útoků na McEliecův kryptosystém. Tento útok nalézá využití u standardizovaných emailů či zpráv, které mají daný formát nebo známé části textu.

Nechť $m_l$ označuje prvních $k_l$ souřadnic $m$ a $m_r$ potom $k_r$ souřadnic, kde máme $k = k_l + k_r$ a $m = (m_l \parallel m_r)$. Pokud útočník zná $m_r$, potom máme

$$c = mG + e = (m_l G_l \parallel 0) + (0 \parallel m_r + G_r) + e,$$

kde $G_l$ je matice tvořená horními $k_l$ řádky a $G_r$ je matice tvořený zbývajícími $k_r$ řádky $G$. Dostáváme

$$c - (0 \parallel m_r + G_r) = (m_l G_l \parallel 0) + e,$$

tedy problém dešifrování $c$ redukujeme na dešifrování $\hat{c} = (m_l G_l + e_{l'})$, kde $e_{l'}$ je příslušná část vektoru $e$. Podobně můžeme postupovat pro jakoukoliv část zprávy $m$.

Útoky na McEliecův kryptosystém s $[n, k]$ kódem a se znalostí $l$ souřadnic otevřeného textu $m$ lze tedy redukovat na útoky proti McEliecova kryptosystému při použití kódu s parametry $[n, k - l]$. Více je možné nalézt v [36].

#### Útok na podobné zprávy

Nechť tentokrát používá McEliecův kryptosystém binární kód. Nechť $m_1$ a $m_2$ jsou dvě zprávy a $c_1 = m_1 G + e$, $c_2 = m_2 G + \hat{e}$, $(e_1, \ldots, e_n) = e \neq \hat{e} = (\hat{e}_1, \ldots, \hat{e}_n)$. Předpokládejme, že útočník zná $\delta m = m_1 + m_2$. Získáme

$$\delta m G + c_1 + c_2 = (m_1 + m_2)G + (m_1 G + e) + (m_2 G + \hat{e}) = e + \hat{e}.$$

Útočník poté zvolí $k$ nulových souřadnic $\delta m G + c_1 + c_2$ a použije algoritmus dekódování s informační množinou, například ze sekce 3.3.1, na $c_1$, nebo $c_2$. S velkou pravděpodobností získá $m_1$, respektive $m_2$, neboť váha chybových vektorů je malá oproti délce $n$.

Tento útok lze použít i na opakované zašifrování stejné zprávy $m$. Rozdíl mezi $e_1$ a $e_2$ je totiž roven $c_1 + c_2$. Více informací o těchto útocích lze nalézt v [37].

#### Reakční útok

Útočník při tomto útoku mění jednu nebo malý počet souřadnic šifrového textu $c$. Označme $\hat{c}$ změněný šifrový text. Útočník posílá $\hat{c}$ příjemci, který má soukromý klíč, a sleduje jeho reakci:

- Reakce A: Příjemce požádá o znovu–zaslání šifrového textu z důvodu neopravitelné chyby nebo nesmyslného textu.
- Reakce B: Příjemce zašle odpověď, nebo neudělá nic, neboť získal příslušný otevřený text $m$.

Reakce B nastane, pokud zpráva $\hat{c}$ obsahuje $t$ či méně chyb. V opačném případě nastane reakce A. Opakováním tohoto procesu se změnou dalších souřadnic získáme v čase $\mathcal{O}(n)$ chybový vektor $e$ a tedy i zprávu $m$.

#### Útok s adaptivní volbou šifrového textu (CCA2)

**Definice 3.4.** *Nechť útočník zná šifrový text $c$ a chce nalézt otevřený text $m$ takový, že $c = mG + e$. Nechť má útočník přístup k dešifrovacímu orákulu, což mu dává možnost získat všechny dvojice šifrový text–otevřený text kromě daného textu $c$ a $m$. Řekneme, že kryptosystém je odolný vůči* adaptivnímu útoku s šifrovým textem (CCA2–bezpečný)*, pokud útočník ani s tímto orákulem nemá žádnou výhodu k dešifrování daného textu $c$.*

U McEliecova kryptosystému může útočník vytvořit nový šifrový text následovně:

$$c' = c + m'G = (m + m')G + e,$$

kde $m'$ je jím zvolená zpráva. Poté použije orákulum, které mu vrátí $c' = m + m'$, a z něj poté získá zprávu $m = c' - m'$. Vidíme tedy, že McEliecův kryptosystém není CCA2–bezpečný.

#### Modifikace protokolu - CCA2 bezpečná varianta

Existuje několik návrhů jak upravit McEliecův systém, aby byl CCA2–bezpečný a odolný vůči ostatním protokolovým útokům. Takové návrhy s důkazem, že tyto úpravy nenavyšují výrazně složitost algoritmu ani velikost klíčů, můžeme najít například v [38].

## 3.4 Praktické aspekty kryptosystému

Pro úplnost popisu McEliecova kryptosystému uvedeme shrnutí praktických vlastností tohoto systému. Všechna uvedená tvrzení lze buď snadno ověřit, nebo je lze nalézt v uvedených publikacích. McElieceho kryptosystém má nevýhodu velkého veřejného klíče a výhodu rychlého šifrování a dešifrování.

Nechť $C$ je binární lineární $[n, k, d]$ kód schopný opravit $t$ chyb.

#### Expanze zprávy

Oproti klasickým zástupcům asymetrické kryptografie dochází u McEliecova kryptosystému při šifrování k expanzi zprávy. Šifrová zpráva je $1/R = \frac{n}{k}$ krát vetší jak otevřená, kde $R$ je hustota použitého kódu $C$. U Niederreiterova kryptosystému je expanzní faktor roven počtu opravitelných chyb $t$ dělených $(1-R)n|\mathbb{F}_2| = 2(n-k)$.

Niederreiterova verze tedy netrpí takovým problémem expanze zprávy jako McEliecova. Bohužel kódy, pro které se expanzní faktor blíží jedné, a pro které existuje vhodný dekódovací algoritmus - jako například pro $LDPC$ kódy, nejsou bezpečné pro použití v McEliecově kryptosystému [39].

### Rychlost generování klíče, šifrování a dešifrování

Operace u McEliecova kryptosystému využívají pouze základní operace a jsou proto velmi rychlé. Generování klíčů má složitost

$$\mathcal{O}(k^2n + n^2 + t^3(n-k) + (n-k)^3).$$

Většina času obvykle zabírá inverze matice $S$ a násobení matic. Šifrování má složitost

$$\mathcal{O}\Big(\frac{kn}{2} + t\Big)$$

a je algoritmicky jednoduché - používá pouze násobení vektoru s maticí a generování náhodného vektoru. Dešifrování využívá daný dešifrovací algoritmus kódu $C$. Pro ireducibilní binární Goppa kódy lze využít Pattersonův algoritmus, kde dešifrování potom má složitost

$$\mathcal{O}(ntm^2).$$

Podrobné vyložení uvedených složitostí je možné nalézt v z [38].

### Velikost klíčů

Hlavním důvodem, proč není McEliecův kryptosystém používaný v praxi je velikost klíčů. Veřejnému klíči McEliecova kryptosystému dominuje generující matice, která má velikost $nk$ bitů. Použití generující matice v základním tvaru $G = (I \mid A)$, by vedlo ke snížení velikosti klíčů, neboť by stačilo ukládat pouze část $A$. Jak jsme ukázali v sekci 3.3.1, tento přístup vede k prozrazeni zprávy $m$. Pro $CCA2$ variantu tento postup však využít lze [38]. Můžeme proto pro veřejný klíč použít pouze matici $A$ o velikosti $k(n-k)$ bitů. Přesto je ovšem klíč pro bezpečné parametry stále veliký. Velikost čísla $t$ je z hlediska rozměru generující matice zanedbatelná.

Nejlepší způsoby redukce velikosti veřejného klíče, a tedy i generující matice, využívají konstrukce generující matice s nenáhodnou strukturou dovolující kompresi. Takovýto přístup je ovšem v rozporu s hlavním požadavkem na bezpečnost: neodlišitelností generující matice od zcela náhodné. Proto je potřeba v takovém případě zvláštního důkazu o bezpečnosti daného systému, neboť bezpečnost řady pokusů o redukci velikosti matice byla nedostatečná [3]. První schéma s důkazem bezpečnosti vychází z takzvaných kvazi–dyadických Goppa kódů [40].

Pro úplnost - velikost privátního klíče odpovídá

$$(n-k)n + (n-k+1+2\log_2 n) + k^2 + n\log_2 n$$

bitům [38].

### Volba parametrů

Parametry kryptosystému musí odpovídat dostupným útokům na daný systém, tak aby bylo výpočetně složité daný útok realizovat v rozumném čase. Pro měření bezpečnosti využíváme počet binárních operací, které musí algoritmus útoku provést. Tabulka 3.2 dokumentuje některé parametry ireducibilních Goppa kódu. Tabulka 3.3 potom pro porovnání referuje v praxi používané klíče RSA.

| $\Gamma_2(L, g)$ | $t$ | Velikost VK v bytech | | Útok [31] |
|---|---|---|---|---|
| | | McEliece | CCA2 varianta | |
| $[1024, 524]$ | 50 | 67 072 | 32 750 | $2^{52,5}$ |
| $[2048, 1696]$ | 32 | 434 176 | 74 624 | $2^{75,8}$ |
| $[4096, 3844]$ | 21 | 1 968 128 | 121 086 | $2^{74,3}$ |
| $[4096, 3616]$ | 40 | 1 851 392 | 216 960 | $2^{114,6}$ |
| $[8192, 7945]$ | 19 | 8 135 680 | 245 302 | $2^{82,6}$ |

Tabulka 3.2: Porovnání velikosti veřejných klíčů (generujících matic) pro různé binární ireducibilní Goppa kódy. Útokem je označen algoritmus dekódování informační množinou s vylepšeními Beckera, Jouxe, Maye a Meurera. Útok je uveden v počtu binárních operací.

| Modulo v bitech | Velikost VK v bytech | Útok |
|---|---|---|
| 1024 | 256 | $2^{79}$ |
| 2048 | 512 | $2^{95}$ |
| 4096 | 1024 | $2^{115}$ |

Tabulka 3.3: Parametry asymetrického kryptosystému RSA. Veřejný klíč obsahuje modulo a exponent, který je často stejné délky. Útok pomocí NFS (celočíselnému sítu) je uveden v počtu binárních operací. Uvedené údaje jsou převzaté z [41].

# Závěr

V této práci jsme prezentovali Goppa kódy a jejich aplikaci v kryptografii. Předvedli jsme formálně správnou konstrukci, která v dostupné literatuře buď chybí, nebo není matematicky plně vyjádřena. Podali jsme ucelený popis těchto kódů a jejich vlastností. Dále jsme je zasadili do souvislosti se Zobecněnými Reed–Solomonovy, Algebraicko–geometrickými kódy a vyložili jsme algoritmus pro dekódování Goppa kódů i jeho verzi pro binární Goppa kódy. Vysvětlen byl také důkaz Sugiyama–Kasahara–Hirasawa–Namekawaovy věty pro Divoké Goppa kódy, který nebyl dosud z dostupné literatury zcela pochopitelný.

Jedna z kapitol také uvedla popis McEliecova kryptosystému využívajícího samoopravné kódy. V ní jsme popsali bezpečnost a možné útoky na tento kryptosystém, včetně nejlepšího typu útoku pomocí dekódování s informační množinou. Nahlédli jsme, proč některé kódy, například Zobecněné Reed–Solomonovy kódy, nejsou pro použití bezpečné a proč nejsou dostupné jednoduché útoky na McEliecův kryptosystém s binárními ireducibilními Goppa kódy. V závěru jsme se dotkli praktické výhody (rychlosti šifrování a dešifrování) a nevýhody (velkých veřejných klíčů).

V průběhu výzkumu jsme narazili na řadu otázek, jejíchž zodpovězení by dalece přesáhlo stanovený rozsah této práce. Z toho důvodu je ponecháváme jako náměty k dalšímu studiu. Nezodpovězena zůstala například otázka cykličnosti části Goppa kódů, vztahu s BCH a s ostatními alternantními kódy, nebo otázka asymptotiky. Spolu se zvyšujícími pokroky v oblasti kvantových počítačů vidíme také velký potenciál v post–kvantové kryptografii a studiu způsobů zmenšení klíčů u McEliecova kryptosystému s Goppa kódy, a to ať již pomocí kvazi-diadické struktury generující matice Goppa kódů, nebo jiných způsobů.

# Literatura


[1] Holub Š.: *Složitost pro kryptografii*, Online, 2004
http://www.karlin.mff.cuni.cz/ holub/skripta/slozitost.pdf

[2] Kaiser T.: Samoopravné kódy, Online, 2006
http://home.zcu.cz/ kaisert/kody/kody.pdf

[3] Bernstein D. J., Buchmann J., Dahmen E.: *Post-Quantum Cryptography*, Springer, 2008, ISBN 978-3-540-88701-0

[4] Berlekamp E. R., McEliece R. J., van Tilborg H.: *On the Inherent intractibility of Certain Coding Problems*, IEEE Transactions on Information Theory, IT-24, 1978, 384-386

[5] MacWilliams F. J., Sloane, N. J. A.: *The Theory of Error-Correctiong Codes*, North-Holland, Amsterdam, 1977, ISBN 0-444-85193-3

[6] Ling S., Xing C.: *Coding Theory A First Course*, Cambridge University Press, 2004, ISBN 978-0-521-52923-5

[7] Goppa V. D.: *Codes associated with divisors*, Problems of Information Transmission, 1977, 13:1, 22–27

[8] Hartshorne R.: *Algebraic geometry*, Springer, 1977, ISBN 978-0-387-90244-9

[9] Silverman J. H.: *The Arithmetic of Elliptic Curves*, Springer, 2008, ISBN 978-0-387-09493-6

[10] Stichtenot H.: *Algebraic Function Field and Codes*, Springer, 1993, ISBN 978-3-540-76877-7

[11] Diffie W., Hellman M. E.: *New directions in cryptography*, IEEE Transactions on Information Theory, 1976, IT-22(6), 644–654

[12] Shor P.W.: *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*, SIAM Journal on Computing, Vol. 26, No. 5., 1997, 1484-1509

[13] Eicher J, Opoku Y.: *Using the Quantum Computer to Break Elliptic Curve Cryptosystems*, Online, 1997
http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.38.1840

[14] Proos J., Zalka C.: *Shor's Discrete Logarithm Quantum Algorithm for Elliptic Curves*, Quantum Information and Computing 3, 2003, 317–344

[15] *Post-quantum cryptography website*, Online, 2013
http://pqcrypto.org/

[16] Dinh H., Moore C., Russell A.: *The McEliece Cryptosystem Resists Quantum Fourier Sampling Attacks*, Lecture Notes in Computer Science, Vol. 6841, Springer, 2011, 761-779

[17] Drápal A.: *Scripta ke komutativním okruhům*, Online, 2006
http://www.karlin.mff.cuni.cz/∼zemlicka/11-12/komalg.pdf

[18] Van der Vlugt M.: *The True Dimension of Certain Binary Goppa Codes*, IEEE Transactions on Information Theory, 1990, IT-36(2), 397-398

[19] Bernstein D. J., Lange T., Peters C.: *Wild McEliece*, Proceedings of Selected Areas in Cryptography, SAC, Springer, 2010, 143-158

[20] Patterson N. J.: *The algebraic decoding of Goppa codes*, IEEE Transactions on Information Theory, 1975, IT-21(2), 203–207

[21] Augot D., Barbier M., Couvreur A.: *List-decoding of binary Goppa codes up to the binary Johnson bound*, Information Theory Workshop (ITW), 2011, 229-233

[22] Bernstein D. J.:. *List decoding for binary Goppa codes*, Coding and Cryptology, 2008, Third International Workshop, IWCC 2011, ISBN 978-3-642-20901-7

[23] McEliece R. J.: *A public-key cryptosystem based on algebraic coding theory*, DSN progress report, Vol. 42, No. 44., 1978, 114-116

[24] Canteaut A., Chabaud F.: *Improvements of the Attacks on Cryptosystems Based on Error-Correcting Codes* Rapport interne du Departement Mathematiques et Informatique, LIENS:95 21, 1995

[25] Niederreiter H.: *Knapsack-type cryptosystems and algebraic coding theory*, Problems of Control and Information Theory 15, 1986, 159-166

[26] Xing Li Y., Deng R. H., Wang X. M.: *On the equivalence of McEliece's and Niederreiter's public-key cryptosystems*, IEEE Transactions on Information Theory, 40(1), 1994, 271-273

[27] Sidelnikov V. M., Shestakov S. O.: *On insecurity of cryptosystems based on generalized Reed–Solomon codes*, Discrete Mathematics and Applications 2, 1992, 439-444

[28] Lee P. J., Brickell E. F.: *An observation on the security of McEliece's public-key cryptosystem*, Lecture Notes in Computer Science 330, Springer, 1988, ISBN 3-540-50251-3

[29] Leon J. S.: *A probabilistic algorithm for computing minimum weights of large error-correcting codes*, IEEE Transactions on Information Theory, 34(5), 1988, 1354-1359

[30] Stern J.: *A method for finding codewords of small weight*, Lecture Notes in Computer Science 388, Springer, 1988, ISBN 0-387-51643-3

[31] Hamdaoui Y., Sendrier N.: *A Non Asymptotic Analysis of Information Set Decoding*, Cryptology ePrint Archive, Report 2013/162, 2013

[32] Bernstein D. J., Lange T.: *Attacking and defending the McEliece cryptosystem*, PQCrypto '08 Proceedings of the 2nd International Workshop on Post-Quantum Cryptography, Springer, 2008, 31-46

[33] Finiasz M., Sendrier N.: *Security Bounds for the Design of Code-based Cryptosystems*, Advances in Cryptology, Asiacrypt 2009, Lecture Notes in Computer Science 5912, Springer, 88-105

[34] Gibson J. K.: *Equivalent Goppa codes and trapdoors to McEliece's public key cryptosystem*, Lecture Notes in Computer Science 547, Springer, 1991, ISBN 3-540-54620-0

[35] Loidreau P., Sendrier N.: *Some weak keys in McEliece public-key cryptosystem*, IEEE International symposium on Information Theory, 1998, 382

[36] Kobara K., Imai H.: *Semantically Secure McEliece Public-Key Cryptosystems–Conversions for McEliece PKC*, Proceedings of 4th International Workshop on Practice and Theory in Public Key Cryptosystems (PKC 01), 2001, 19–35

[37] Berson T. A.: *Failure of the McEliece public-key cryptosystem under message-resend and related-message attack*, CRYPTO 1294, Lecture Notes in Computer Science, Springer, 1997, 213–220

[38] Engelbert D., Overbeck R., Schmidt A.: *A summary of McEliece-type cryptosystems and their security*, Journal of Mathematical Cryptology 1, 2008, 151-199

[39] Minder L.: *Cryptography based on error correcting codes*, Ph.D. thesis, École Polytechnique Fédérale de Lausanne, 2007

[40] Misoczki R., Barreto S. L. M. P.: *Compact McEliece Keys from Goppa Codes*, Lecture Notes in Computer Science 5867, Cryptology ePrint Archive, Report 2009/187, 376-392

[41] Overbeck R.: *Public key cryptography based on coding theory*, Ph.D. thesis, Darmstadt, 2007