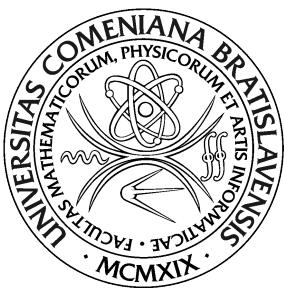


UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



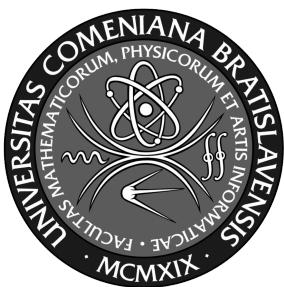
GRUPY AUTOMORFIZMOV LINEÁRNYCH KÓDOV

Diplomová práca

2022

Bc. Branislav Boráň

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



GRUPY AUTOMORFIZMOV LINEÁRNYCH KÓDOV

Diplomová práca

- Študijný program: Aplikovaná informatika
Študijný odbor: 2511 Aplikovaná informatika
Školiace pracovisko: Katedra algebry a geometrie
Školiteľ: doc. RNDr. Róbert Jajcay, DrSc.

Bratislava, 2022

Bc. Branislav Boráň



ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Branislav Boráň

Študijný program: aplikovaná informatika (Jednoodborové štúdium,
magisterský II. st., denná forma)

Študijný odbor: informatika

Typ záverečnej práce: diplomová

Jazyk záverečnej práce: anglický

Sekundárny jazyk: slovenský

Názov: Automorphism groups of linear codes and linear codes with prescribed automorphism groups

Grupy automorfizmov lineárnych kódov a lineárne kódy s predpisanou grupou automorfizmov

Anotácia: Lineárne kódy sú podpriestory konečnorozmerných vektorových priestorov nad konečnými poľami. Majú preto bohaté grupy automorfizmov, ktoré zároveň obsahujú množstvo informácií o uvažovanom kóde. Určenie úplnej grupy automorfizmov kódu je výpočtovo náročná úloha. Namiesto určenia grupy automorfizmov pre daný kód sa preto uvažuje obrátená úloha zostrojenia kódu s predpisanou grupou automorfizmov. Cieľom práce je preskúmať oba smery tejto interakcie.

Cieľ: Cieľom navrhanej problematiky je poskytnúť študentovi výpočtovo zložitý problém vyžadujúci dôkladné porozumenie štruktúry uvažovaných objektov ako aj programátorské a organizačné schopnosti.

Literatúra: R. Hill, A first course in coding theory, Oxford University Press, 1993
S. Roman, Coding and information theory, Springer, 1992
R. Jajcay, P. Potocnik and Stephen E. Wilson, Half-cyclic, dihedral and half-dihedral codes,
J. of Applied Mathematics and Computing 64 (2020), 691-708.

Kľúčové

slová: lineárny kód, grúpa automorfizmov, konečné pole

Vedúci: doc. RNDr. Róbert Jajcay, DrSc.

Katedra: FMFI.KAG - Katedra algebry a geometrie

Vedúci katedry: doc. RNDr. Pavel Chalmovianský, PhD.

Dátum zadania: 09.12.2020

Dátum schválenia: 10.12.2020

prof. RNDr. Roman Ďuríkovič, PhD.
garant študijného programu

študent

vedúci práce

Čestne prehlasujem, že túto diplomovú prácu som vypracoval samostatne len s použitím uvedenej literatúry a za pomoci konzultácií u môjho školiteľa.

.....
Bratislava, 2022

Bc. Branislav Boráň

Pod'akovanie

Chcel by som sa v prvom rade pod'akovať môjmu školiteľovi doc. RNDr. Róbertovi Jajcayovi, DrSc. za odbornú pomoc a usmernenia pri písaní tejto práce, za materiály, cenné rady, ktoré mi veľmi pomohli pri riešení tejto diplomovej práce. V neposlednom rade chcem tiež pod'akovať všetkým mojím kamarátom a celej mojej rodine za podporu počas môjho štúdia.

Abstrakt

Táto práca sa venuje problematike skúmaniau grúp automorfizmov lineárnych kódov ako aj lineárnym kódom s predpísanou grupou automorfizmov. V našej práci sa zameriavame na LDPC kódy.

Kľúčové slová: automorfizmus grúp, LDPC, klietky

Abstract

This thesis deals with the problem of examining groups of automorphisms of linear codes as well as linear codes with a prescribed group of automorphisms. In our work we focus on LDPC codes.

Keywords: Automorphism groups, LDPC, cages

Obsah

1	Úvod	1
2	Motivácia	2
3	Analýza problému	3
3.1	Lineárny kód $C(n,r)$	3
3.1.1	Generujúca matica lineárneho kódu G	3
3.1.2	Kontrolná matica lineárneho kódu H	4
3.1.3	Dĺžka lineárneho kódu n	4
3.1.4	Maximálny počet kódových slov m	4
3.1.5	Minimálna Hammingová vzdialenosť d	5
3.1.6	Kódová váha R	5
3.1.7	Perfektné lineárne kódy	5
3.1.8	LDPC kódy	6
3.2	Grafová reprezentácia LDPC kódov	6
3.2.1	Obvod grafu g	7
3.2.2	Automorfizmus grafu	7
3.2.3	Konštrukcia LDPC kódov pomocou klietok	8
4	Návrh riešenia	12
4.1	Generovanie Cage(k,g) alebo Rec(k,g)	13

4.1.1	Sage grafy ako vstupné dátá	13
4.1.2	Zoznám susedností ako vstupné dátá	13
4.1.3	Nejednotný prístup bez vstupných dát	14
4.2	Validácia vygenerovaných Cage(k,g) alebo Rec(k,g)	15
4.3	Získavanie údajov z Cage(k,g) alebo Rec(k,g)	15
4.3.1	Cykly formujúce Cage(k,g) alebo Rec(k,g)	15
4.3.2	Vrcholy, hrany a počet automorfizmov	16
4.3.3	Incidenčné matice	16
4.4	$C(n,m,d)$ a G z incidenčnej matice klietky	16
4.4.1	$d, m, Aut(Gr)$	17
4.4.2	Porovnanie s perfektnými kódmi	17
4.5	Validácia lineárneho kódu	17
5	Výsledky	18
5.1	Generovanie Cage(k,g), Rec(k,g) a G	18
5.1.1	Cage(3,5) - Petersenov graf	18
5.1.2	Cage(3,6) - Heawoodov graf	20
5.1.3	Cage(3,7) - McGeeho graf	21
5.1.4	Cage(3,8) - Tutteho-Coxeterov graf	23
5.1.5	Cage(3,10) - Balabanov graf	25
5.1.6	Cage(3,11) - Balabanov graf	26
5.1.7	Cage(3,14)	27
5.1.8	Rec(3,16)	27
5.1.9	Rec(3,17)	28
5.1.10	Rec(3,18)	28
5.1.11	Cage(3,20)	28
5.1.12	Rec(3,23)	29
5.1.13	Rec(3,25)	29

<i>OBSAH</i>	x
--------------	---

5.1.14 Cage(4,5) - Robertsonov graf	30
5.1.15 Cage(4,7)	32
5.1.16 Cage(4,9)	33
5.1.17 Cage(4,10)	33
5.1.18 Cage(5,10)	34
5.1.19 Cage(6,4)	34
5.1.20 Cage(7,5)	36
5.1.21 Cage(7,7)	36
5.1.22 Cage(7,8)	37
5.1.23 Rec(10,5)	37
5.1.24 Rec(11,5)	38
5.1.25 Rec(12,5)	38
5.1.26 Rec(13,5)	39
5.2 Vyhodnotenie výsledkov	39
5.2.1 Porovnanie výsledkov	39
5.2.2 Zhrnutie výsledkov	41

6 Záver	43
----------------	-----------

Zoznam skratiek

|Aut(Gr)| Počet automorfizmov. 9–11, 15–17, 19, 21, 22, 24–29, 31–42

|cols| Počet stĺpcov matice. 4, 5

|edg| Počet hrán grafu. 15, 16, 19, 21, 22, 24–29, 31–39

|rows| Počet riadkov matice. 4, 5

|ver| Počet vrcholov grafu. 8–11, 13–16, 19, 21, 22, 24–29, 31–39, 42

Aut(Gr) Grupa automorfizmov. ix, 8, 12, 16, 17

C(n,m,d) Lineárny kód. ix, 3–6, 12, 15–18, 39, 42

C(n,r) Lineárny kód. viii, 3, 4

Cage(k,g) Klietka. viii, ix, 8, 10, 13–16, 18, 40–42

d Minimálna Hammingová vzdialenosť. viii, ix, 3, 5, 16, 17, 19, 21, 22, 24–29, 31–39, 42

dim(|rows|,|cols|) Dimenzia matice. 4

G Generujúca matica lineárneho kódu. viii, ix, 3–5, 16–22, 24–29, 31–42, 46,

g Obvod grafu. viii, 7, 8, 10, 14

H Kontrolná matica lineárneho kódu. viii, 4, 6, 7, 16, 17, 19–42, 46, 47

k Stupeň vrcholov v klietke. 8, 10, 14

LDPC Low density parity check codes - Lineárne kódy s malou hustotou matice H. viii, 6–8

m Maximálny počet kódových slov. viii, ix, 3–5, 16, 17, 19, 21, 22, 24–29, 31–39, 42

M(k,g) Moorové ohraničenie pre klietku. 8, 10, 14, 15, 19, 21, 22, 24–29, 31–41

n Dĺžka lineárneho kódu. viii, 3–5

p(n,m,d) Parameter perfektného kódu. 17, 19, 21, 22, 24–29, 31–42

QC-MDPC Kvázicyklické lineárne kódy s väčšou hustotou matice H ako LDPC. 6

R Kódová váha. viii, 5

r Počet nezávislých vektorov. 3, 4

Rec(k,g) Rekordný graf. viii, ix, 10, 13–16, 18, 41, 42

t(G) Čas generovania generujúcej matice lineárneho kódu. 19, 21, 22, 24–29, 31–39, 42

t(H) Čas generovania kontrolnej matice lineárneho kódu. 19, 21, 22, 24–29, 31–39, 42

Kapitola 1

Úvod

xxxxxx

Kapitola 2

Motivácia

XXXXXX

Kapitola 3

Analýza problému

3.1 Lineárny kód $C(n,r)$

Lineárny kód $C(n,r)$ je r -rozmerný lineárny podpriestor priestoru F_n^2 . F_n^2 je priestor n -rozmerných vektorov, kde koordináty berieme z poľa F^2 . r -rozmerný lineárny podpriestor obsahuje práve r lineárne nezávislých vektorov. Ak by sme zobrali r takých vektorov, potom tieto vektory generujú daný r -rozmerný podpriestor a hovoríme, že tvoria bázu podpriestoru.[MTSB13] [HP03] Ak je splnená vlastnosť, ktorá hovorí, že súčet 2 kódových slov *mod* 2 je kódové slovo, tak uvažovaný binárny kód je lineárny a vieme nájsť Generujúcu maticu lineárneho kódu G .[Mal07][Hil91] Binárny $[n,r,d]$ kód môže byť chápaný ako $(n,2^r,d)$ kód[Hil91], takýto kód budeme nazývať Lineárny kód $C(n,m,d)$, kde m je maximálny počet slov a d je minimálna kódová vzdialenosť.

3.1.1 Generujúca matica lineárneho kódu G

Generujúca matica lineárneho kódu G je zostrojená z bázy lineárneho kódu $C(n,r)$. Riadky matice predstavujú prvky bázy a zároveň predstavujú line-

árne nezávislé vektory dĺžky n. [MTSB13] [HP03] Uvažujme \vec{s} ako vstup (nekódované slovo), \vec{v} je výstup (kódované slovo), potom platí:

$$C(n, r) = \{\vec{s} \times G : \vec{s} \in F_2^r\}, \quad \vec{v} = \vec{s} \times G \quad (3.1)$$

3.1.2 Kontrolná matica lineárneho kódu H

V r-rozmernom linearnom kóde C(n,r) v F_n^2 potom existuje $n - r$ lineárne nezávislých vektorov \vec{v} takých, že každé kódové slovo je kolmé na všetky tieto vektory. Keď týchto $n - r$ vektorov zoberieme ako riadky matice, dostaneme kontrolnú maticu lineárneho kódu H. [MTSB13] [HP03] Ľubovoľný vektor \vec{v} je kódovým slovom práve vtedy, ak platí:

$$C(n, r) = \{\vec{v} \in F_2^n : H \times \vec{v}^T = 0\} \quad (3.2)$$

3.1.3 Dĺžka lineárneho kódu n

Dĺžka lineárneho kódu n špecifikuje dimenzie $\dim(|\text{rows}|, |\text{cols}|)$ kontrolnej matice H. $|\text{rows}|$ predstavuje počet riadkov matice a $|\text{cols}|$ je počet stĺpcov. V Generujúcej matici G predstavuje n počet stĺpcov.[MTSB13]

3.1.4 Maximálny počet kódových slov m

Maximálny počet kódových slov m je maximálny počet takých slov, ktoré je možné v lineárnom kóde C(n,m,d) zakódovať. Ak zoberieme všetky riadky Generujúcej matice G o ktorých vieme, že sú nezávislé ako parameter r[Hil91], potom platí:

$$m = 2^r \quad (3.3)$$

3.1.5 Minimálna Hammingová vzdialenosť d

Uvažujme vektory \vec{u} a \vec{v} ako kódové slová. Minimálna Hammingová vzdialenosť d 2 vektorov $d(\vec{u}, \vec{v})$, pre ktoré platí, že $\vec{u} \in F_n^2$ a $\vec{v} \in F_n^2$ je počet koordinátov, na ktorých sa vektory \vec{u} a \vec{v} líšia.[MTSB13] [HP03] d v kóde predstavuje najmenší počet koordinátov v generujúcej matici G[Hil91].

3.1.6 Kódová váha R

Kódová váha R predstavuje počet informácií alebo bitov nad celkovým počtom prenesených bitov. R je vyjadrená vzťahom [MTSB13]:

$$R = \frac{|cols| - |rows|}{|cols|} \quad (3.4)$$

3.1.7 Perfektné lineárne kódy

Sphere-packing ohraničenie

Predpokladajme, že Lineárny kód $C(n,m,d)$ je binárny $(n, m, 2t + 1)$ -kód. Nech t je parameter, pre ktorý platí, že každé 2 sféry polomeru t centrovane na odlišné kódové slová nemajú spoločné vektory. Pre parameter t platí:

$$d = 2t + 1 \quad (3.5)$$

Sphere-packing ohraničenie je dané vzťahom:

$$m \left\{ \binom{n}{0} + \binom{n}{1} + \dots + \binom{n}{t} \right\} \leq 2^n \quad (3.6)$$

Počet všetkých vektorov v m sférach s polomerom t centrovanych na m kódových slov je dané ľavou stranou rovnice. O Sphere-packing ohraničení hovoríme, ak je ľavá strana rovnice menšia nanajvýš rovná 2^n . 2^n vyjadruje

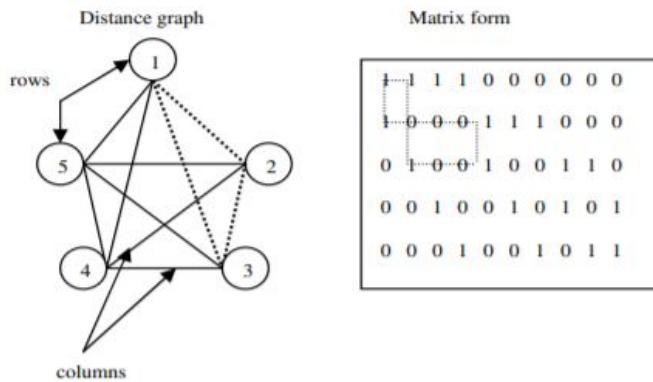
celkový počet vektorov v F_2^n . V prípade, že sú obe strany rovnice v rovnosti, tak hovoríme, že Lineárny kód C(n,m,d) je **perfektný** [Hil91]

3.1.8 LDPC kódy

LDPC kódy sú lineárne samoopravné kódy, ktoré umožňujú prenos dát rýchlosťou blízkou kapacite kanálu a zároveň pre ne existujú vysoko účinné dekódovacie algoritmy. Kódy majú veľmi riedku maticu H, pomocou ktorej sa dajú opraviť chyby v kódových slovách. Obsahuje menej ako 1% jednotiek.[MTSB13] Hlavnou nevýhodou väčšiny LDPC kódov je vysoká časová náročnosť ich kódovacieho algoritmu. Výhodou je paralelizmus pri dekódovaní a jednoduché výpočtové operácie. Dekódovacie výpočty sú rozdelené do 2 množín uzlov a to do kontrolných uzlov a premenných uzlov. Uzol na jednej strane je spojený s uzlom na druhej strane, čo umožňuje paralelné výpočty na každej strane.[Mal07] Téme LDPC kódov som sa okrajovo venoval aj vo svojej bakalárskej práci, v ktorej som skúmal hostotu inverzií riedkych cyklických matíc. Zameral som sa však na QC-MDPC McElieceov kryptosystém. Rozdiel medzi QC-MDPC kódmi a LDPC je v hustote kontrolnej matice, ktorá môže byť o niečo hustejšia ako pri LDPC kódoch (obsahuje menej ako 2% jednotiek).[Bor18]

3.2 Grafová reprezentácia LDPC kódov

Kontrolná matica lineárneho kódu H môže byť reprezentovaná grafom vzdialosti, v ktorom riadky matice predstavujú vrcholy a stĺpce matice reprezentujú hrany grafu. Stĺpec je potom množina hrán formujúca kompletnejší graf medzi vrcholmi spojenými v stĺpci. Nasledujúci obrázok ilustruje grafovú reprezentáciu matice LDPC kódu odvodenu z grafu vzdialenosťi:[Mal07]



Obr. 3.1: Vzťah medzi grafom a kontrolnou maticou [Mal07]

Graf vzdialenosť je formovaný cestami hrán alebo vrcholov. Obvod grafu g korešponduje s obvodom dĺžky $2g$ v kontrolnej matici H .

3.2.1 Obvod grafu g

Obvod grafu g ovplyvňuje dekódovanie LDPC kódu. V grafovej reprezentácii LDPC kódu sa jedná o najmenší cyklus v grafe. Jeho dĺžku vypočítame buď pomocou vrcholov alebo pomocou hrán. V kontrolnej matici H kódu je dĺžka obvodu $2g$, pretože cyklus alternuje medzi riadkami a stĺpcami z čoho vyplýva, že cyklus grafu reprezentuje iba polovicu maticového kódu.[Mal07]

3.2.2 Automorfizmus grafu

Automorfizmus grafu je permutácia ϕ všetkých vrcholov grafu, ktorá zachováva jeho štruktúru takým spôsobom, že akékoľvek 2 vrcholy U a V susedia iba vtedy a len vtedy ak platí, že $\phi(U)$ susedí s $\phi(V)$.[EJ13] Zjednodušene môžeme povedať, že sa jedná o bijektívne zobrazenie, pri ktorom sa každý vrchol grafu a každá hrana zobrazí na iný vrchol a hranu. Hovoríme tiež, že ide o jeho obraz. Množina všetkých automorfizmov grafu Gr tvorí **grupu**

automorfizmov $\text{Aut}(\text{Gr})$.[EJ13]

3.2.3 Konštrukcia LDPC kódov pomocou klietok

Na konštrukciu LDPC kódov môžme využiť grafy vzdialenosťi. Tieto grafy delíme na regulárne s vrcholmi rovnakého stupňa k (napr. Moorové grafy) a neregulárne s vrcholmi rôznych stupňov k .[Mal07] V našej práci sa budeme venovať regulárnym grafom vzdialenosťi. **Klietka Cage(k,g)** je k -regulárny graf obvodu g s počtom vrcholov $|ver|$ väčším nanajvýš rovným ako je dolné ohraničenie počtu vrcholov, tiež známym ako **Moorové ohraničenie pre klietku $M(k,g)$** .[Mal07] Výpočet Moorovho ohraničenia sa líši podľa toho, či je jej obvod g párny alebo nepárny:

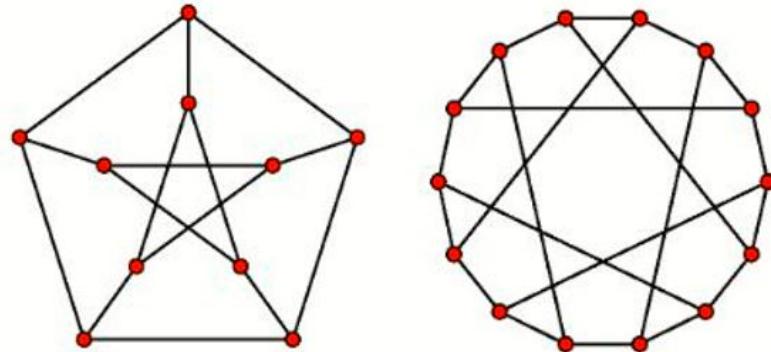
nepárny obvod:

$$M(k, g) \leq 1 + \sum_{i=0}^{(g-3)/2} k(k-1)^i = \frac{k(k-1)^{(g-1)/2} - 2}{k-2} \quad (3.7)$$

párny obvod:

$$M(k, g) \leq 2 \sum_{i=0}^{(g-2)/2} k(k-1)^i = \frac{2(k-1)^{g/2} - 2}{k-2} \quad (3.8)$$

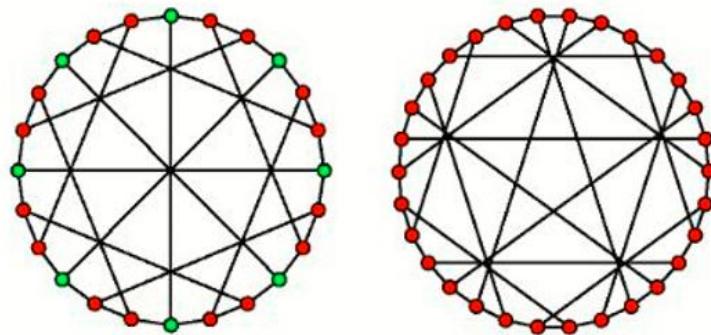
Aj keď neexistuje jednotná konštrukcia klietok, existuje niekoľko známych klietok pre stupeň vrchola k a obvod g . Ukážeme si niektoré z nich:



Obr. 3.2: Petersenov graf (vľavo) a Heawoodov graf (vpravo) [EJ13]

Petersenov graf predstavuje klietku $cage(3, 5)$, $|ver|= 10$ a je vrcholovo tranzitívny.

Heawoodov graf predstavuje klietku $cage(3, 6)$, $|ver|= 14$, $|\text{Aut}(\text{Gr})|= 336$ a je vrcholovo tranzitívny. [EJ13]



Obr. 3.3: McGeeho graf (vľavo) a Tutteho-Coxeterov graf (vpravo)[EJ13]

McGeeho graf predstavuje klietku $cage(3, 7)$, $|ver|= 24$, $|\text{Aut}(\text{Gr})|= 32$ a nie je vrcholovo tranzitívny.

Tutteho-Coxeterov graf predstavuje klietku $cage(3, 8)$, $|ver|= 30$, $|\text{Aut}(\text{Gr})|= 1440$ a je vrcholovo tranzitívny.[EJ13]

Balabanov graf predstavuje klietku $cage(3, 11)$, $|ver|= 112$, $|\text{Aut}(Gr)|= 64$ a nie je vrcholovo tranzitívny.

Bensonov graf predstavuje klietku $cage(3, 12)$, $|ver|= 126$, $|\text{Aut}(Gr)|= 12096$ a je vrcholovo tranzitívny. [EJ13]

Ďalšie grafy:

Robertsonov graf predstavuje klietku $cages(4, 5)$

Exoo, McKay a Nadonov graf predstavujú klietku $cage(4, 7)$

Grafy, ktorých počty automorfizmov 20, 30 a 120 majú klietku $cages(5, 5)$

Hoffmanov-Singletonov graf predstavuje klietku $cages(7, 5)$

O'Keefe a Wongov graf predstavujú klietku $cages(7, 6)$ [EJ13]

Okrem klietok rozlišujeme takzvané **Rekordné grafy Rec(k,g)**. Rekordný graf Rec(k,g) je podobný typ grafu ako Cage(k,g). Jedná sa o k-regulárny graf s počtom vrcholov $|ver|$ väčším nanajvýš rovným ako je $M(k,g)$. Rozdiel je však v obvode g. Rec(k,g) nemá všetky obvody g rovnakej dĺžky ale môže obsahovať aj obvody g dĺžky väčšej ako g. O rekordných grafoch budeme hovoriť v prípade, že klietka Cage(k,g) pre uvažované parametre nie je známa ale je pre ne známy Rec(k,g). Ukážeme si niektoré z nich [EJ13]:

Rec(3,13): $|ver|= 272$

Rec(3,14): $|ver|= 384$

Rec(3,15): $|ver|= 620$, $|\text{Aut}(Gr)|= 14880$

Rec(3,16): $|ver|= 960$, $|\text{Aut}(Gr)|= 96$

Rec(3,17): $|ver|= 2176$, $|\text{Aut}(Gr)|= 544$

Rec(3,18): $|ver|= 2560$, $|\text{Aut}(Gr)|= 640$

Rec(3,19): $|ver|= 4324$, $|\text{Aut}(Gr)|= 51888$

Rec(3,20): $|ver|= 5376$, $|\text{Aut}(Gr)|= 2688$

Rec(3,21): $|ver|= 16028$

Rec(3,22): $|ver| = 16206$

Rec(3,23): $|ver| = 49326$

Rec(3,24): $|ver| = 49608$

Rec(3,25): $|ver| = 108906$

Rec(8,5): $|ver| = 80$

Rec(9,5): $|ver| = 96$

Rec(10,5): $|ver| = 124$

Rec(11,5): $|ver| = 154$

Rec(12,5): $|ver| = 203$, $|\text{Aut}(\text{Gr})| = 203$

Rec(13,5): $|ver| = 230$

Kapitola 4

Návrh riešenia

Problematiku riešime v programe **Sage** [The], ktorý je založený na programovacom jazyku **Python**. Ponúka veľké množstvo vopred naimplementovaných funkcií pre prácu s grafmi, linearnymi kódmi $C(n,m,d)$, maticami a grupami automorfizmov $\text{Aut}(\text{Gr})$. Využili sme online aplikáciu **CoCalc** [CoC], ktorá nám umožňuje vytvárať Sage projekty priamo na internete ako aj aplikáciu **SageMath 9.2**, ktorá je voľne dostupná na stiahnutie a inštaláciu. CoCalc prevádzkuje prostredie Ubuntu Linux, s ktorým je možné komunikovať cez terminál.



Collaborative Calculation in the Cloud

Obr. 4.1: Sage a Cocalc[The][CoC]

4.1 Generovanie Cage(k,g) alebo Rec(k,g)

Existuje viacero možností ako vygenerovať klietku Cage(k,g) alebo rekordný graf Rec(k,g) v programe Sage. V našej práci uvažujeme 3 spôsoby generovania Cage(k,g) alebo Rec(k,g). Prvý spôsob je **využietie grafov**, ktoré Sage ponúka ako vopred naimplementované grafy. Druhý spôsob sa opiera o naprogramovanie vlastného parsera, ktorý dokáže spracovať vstupné textové súbory **zoznamu susednosti** a z nich vytvoriť Cage(k,g) alebo Rec(k,g). Tretí spôsob spočíva vo **využití známych údajov**, ktoré sú pre danú klietku Cage(k,g) všeobecne známe (počet vrcholov $|ver|$, typ grafu a podobne) a navrhnutie vlastného postupu, ktorý bude viest' k vygenerovaniu Cage(k,g) alebo Rec(k,g) .

4.1.1 Sage grafy ako vstupné dátá

Program Sage ponúka zopár vopred naimplementovaných grafov, ktoré môžme využiť ako klietky Cage(k,g). Jedná sa o **Petersenov graf** - $cage(3, 5)$, **Heawoodov graf** - $cage(3, 6)$, **McGeeho graf** - $cage(3, 7)$, **Tutteho-Coxeterov graf** - $cage(3, 8)$, **Balabanov graf** - $cage(3, 10)$, **Robertsonov graf** - $cage(4, 5)$, **Hoffmanov-Singletonov graf** - $cage(7, 5)$

4.1.2 Zoznám susednosti ako vstupné dátá

Uvažujeme existujúce vstupné textové súbory, ktoré nesú informácie o susednostiach všetkých uvažovaných vrcholov jednotlivo pre 1 klietku Cage(k,g) alebo rekordný graf Rec(k,g). Riadok predstavuje zoznam vrcholov, s ktorými susedí konkrétny vrchol. Tieto vstupné súbory je možné získať na webovej stránke [Exo]. Navrhнемe si vlastný parser, ktorý nám rozdelí tieto dátá na zoznamy vrcholov. Z nich bude možné vytvoriť hrany. Tieto hrany bude

možné pridať prázdnemu grafu (pomocou metódy `EmptyGraph()`[The]) a tým z neho vytvoriť `Cage(k,g)` alebo `Rec(k,g)`.

4.1.3 Nejednotný prístup bez vstupných dát

Generovanie klietok bez vstupných dát nemá jednotný prístup, je potrebné k nim pristupovať jednotlivo alebo preskúmať skupiny, ktoré sa generujú podobným spôsobom.[EJ13]

Generovanie cage(6,4)

Navrhнемe vlastný spôsob generovania klietky `Cage(k,g)` na základe parametrov k a g bez ďalších vstupných parametrov. Budúcu klietku vygenerujeme ako bipartitný graf. V Sage použijeme metódu `DegreeSequenceBipartite(s,s)`[The], ktorá bude mať 2 rovnaké parametre s , pričom každý predstavuje zoznam vrcholov. Najskôr je potrebné si vypočítať Moorové ohraničenie pre klietku $M(k,g)$ pre minimálny počet vrcholov. $Cage(6,4)$ je taký typ klietky, ktorej počet vrcholov $|ver|$ je rovnaký ako minimálny počet vrcholov z $M(k,g)$.[Mal07] Tieto vrcholy rozdelíme na 2 zoznamy takým spôsobom, že každý zoznam bude obsahovať $\frac{m}{2}$ vrcholov stupňa k a teda platí:

$$s = [k, k, k, k, k, k] \quad len(s) = \frac{m}{2} \quad (4.1)$$

Budúcej klietke `Cage(k,g)` nastavíme vrcholy pomocou metódy `set_vertex()`[The]. Výsledný graf bude `cage(6,4)`

4.2 Validácia vygenerovaných Cage(k,g) alebo Rec(k,g)

Na validáciu existencie klietok Cage(k,g) alebo rekordných grafov Rec(k,g) bude potrebné zistieť **Moorové ohraničenie pre klietku M(k,g)** na základe ktorého vieme otestovať počet vrcholov $|ver|$ ako aj existenciu. Následne vieme otestovať aj počet hrán $|edg|$, minimálny rozmer incidenčnej matice. Na validáciu ako aj výpočet bude potrebné zostrojiť samostatné metódy.

4.3 Získavanie údajov z Cage(k,g) alebo Rec(k,g)

Sage ponúka zopár vopred naimplementovaných metód na získanie údajov z klietky Cage(k,g) alebo rekordného grafu Rec(k,g), niektoré však bude potrebné naprogramovať. Údaje ktoré budeme pri Cage(k,g) alebo Rec(k,g) sledovať budú **cykly formujúce Cage(k,g) alebo Rec(k,g)**, **vrcholy a hrany** ako aj **Počet automorfizmov $|\text{Aut}(\text{Gr})|$** zistený priamo z Cage(k,g) alebo Rec(k,g). Špeciálny údaj bude **incidenčná matica**, ktorú budeme v ďalšej časti spracovávať v spojitosti s lineárnym kódom C(n,m,d).

4.3.1 Cykly formujúce Cage(k,g) alebo Rec(k,g)

Navrhнемe vlastný spôsob zistenia cyklov formujúcich klietku Cage(k,g) alebo rekordný graf Rec(k,g). Získame utriedený zoznam vrcholov v cykloch (pomocou *graph.minimum_cycle_basis()* [The]) Následne z Cage(k,g) alebo Rec(k,g) vytvoríme podgrafy obsahujúce tieto vrcholy (pomocou metódy *subgraph()* [The], zoznam vrcholov bude parameter). Takýto podgraf obsahuje len 1 cyklus (získame ho pomocou metódy *cycle_basis()* [The]), ktorý pridáme do nášho zoznamu cyklov, ktoré sledujeme. Výsledný zoznam obsa-

huje cykly formujúce $\text{Cage}(k,g)$ alebo $\text{Rec}(k,g)$.

4.3.2 Vrcholy, hrany a počet automorfizmov

Z vygenerovanej klietky $\text{Cage}(k,g)$ alebo rekordného grafu $\text{Rec}(k,g)$ vieme získať zoznam všetkých vrcholov získame z vygenerovanej $\text{Cage}(k,g)$ alebo $\text{Rec}(k,g)$ pomocou metódy $\text{vertices}()$ [The], zoznam všetkých hrán (pomocou metódy $\text{edges}()$ [The]) a zoznam všetkých automorfizmov(pomocou $\text{automorphism_group}()$ [The]). Pre zistenie Počet vrcholov grafu $|ver|$, Počet hrán grafu $|edg|$ a Počet automorfizmov $|\text{Aut}(\text{Gr})|$ nám stačí iba zistiť veľkosť ich zoznamov.

4.3.3 Incidenčné matice

Z klietky $\text{Cage}(k,g)$ alebo rekordného grafu $\text{Rec}(k,g)$ vieme získať incidenčnú maticu (pomocou metódy $\text{incidence_matrix}()$ [The]). Incidenčná matica predstavuje zároveň kontrolnú maticu lineárneho kódu H .

4.4 $C(n,m,d)$ a G z incidenčnej matice klietky

Z incidenčnej matice H je možné získať Lineárny kód $C(n,m,d)$ pomocou existujúcej metódy v Sage $\text{codes.from_parity_check_matrix}(H)$ [The] a následne z neho Generujúcu maticu lineárneho kódu G pomocou metódy $\text{systematic_generator_matrix}()$. Ďalej nás bude zaujímať **Minimálna Hammingová vzdialenosť d**, **Maximálny počet kódových slov m**, **Grupa automorfizmov Aut(Gr)** a porovnanie s perfektnými lineárnymi kódmi

4.4.1 d , m , $\text{Aut}(\text{Gr})$

Minimálnu vzdialenosť v kóde d je možné zistiť z Lineárneho kódu pomocou metódy *minimum_distance()*[The]

Maximálny počet kódových slov m je možné zistiť z Generujúcej matice G výpočtom.

Grupu automorfizmov $\text{Aut}(\text{Gr})$ je možné zistiť z Lineárneho kódu $C(n,m,d)$ pomocou metódy *permutation_automorphism_group()* a **Počet automorfizmov $|\text{Aut}(\text{Gr})|$** je možné zistiť z Lineárneho kódu $C(n,m,d)$ pomocou metódy *order()*[The]

4.4.2 Porovnanie s perfektnými kódmi

Zadefinujeme si parameter $p(n,m,d)$ na zaklade definície perfektného kódu:

$$p(n, m, d) = \frac{m \left\{ \binom{n}{0} + \binom{n}{1} + \dots + \binom{n}{t} \right\}}{2^n} \quad (4.2)$$

Tento parameter bude v intervale $<0,1>$ a bude znázorňovať akým spôsobom sa líšime od perfektného kódu. ($p(n,m,d) = 1$ by znamenalo, že uvažujeme perfektný kód)

4.5 Validácia lineárneho kódu

Na validáciu využijeme vzťah z definície lineárneho kódu $C(n,m,d)$ a jeho kontrolnej matice H :

$$G \times H^T = 0 \quad (4.3)$$

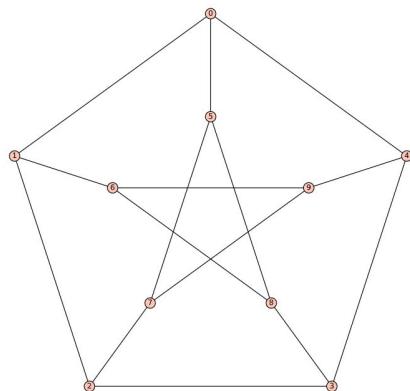
Kapitola 5

Výsledky

5.1 Generovanie Cage(k,g), Rec(k,g) a G

V tejto kapitole spracovávame výsledky všetkých typov uvažovaných klietok Cage(k,g) a rekordných grafov Rec(k,g) od ich vygenerovania až po získanie lineárneho kódu C(n,m,d) spolu s generujúcou maticou G.

5.1.1 Cage(3,5) - Petersenov graf



Obr. 5.1: Cage(3,5) [The]

[1 1 1 0 0 0 0 0 0 0 0 0 0 0 0]	
[1 0 0 1 1 0 0 0 0 0 0 0 0 0 0]	
[0 0 0 1 0 1 1 0 0 0 0 0 0 0 0]	
[0 0 0 0 0 1 0 1 1 0 0 0 0 0 0]	
[0 1 0 0 0 0 0 1 0 1 0 0 0 0 0]	
[0 0 1 0 0 0 0 0 0 1 1 0 0 0]	
[0 0 0 0 1 0 0 0 0 0 0 1 1 0]	
[0 0 0 0 0 0 1 0 0 0 1 0 0 0 1]	
[0 0 0 0 0 0 0 1 0 0 1 1 0 0]	
[0 0 0 0 0 0 0 0 1 0 0 0 1 1]	

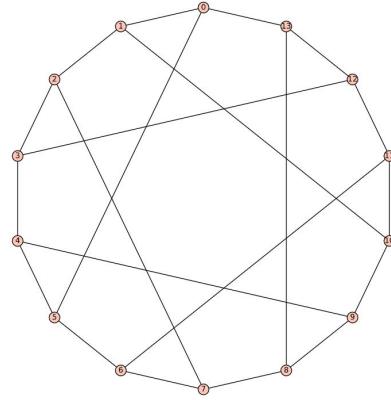
	[1 0 1 0 1 0 0 0 0 0 0 0 1 1 0 0]
	[0 1 1 0 0 0 0 0 0 0 1 0 1 1 1 0]
	[0 0 0 1 1 0 1 0 0 0 0 0 0 1 1]
	[0 0 0 0 0 1 1 0 1 0 0 0 1 1 1]
	[0 0 0 0 0 0 1 1 1 0 0 0 1 1 0]
	[0 0 0 0 0 0 0 1 1 1 0 0 1 1 0]
	[0 0 0 0 0 0 0 0 0 1 1 1 1 1 1]

Obr. 5.2: H (vľavo) a G (vpravo)

Cage(3,5)		C(15,6,5)	
ver	10	d	5
edg	15	m	64
rozmer H	10×15	rozmer G	6×15
t(H)	0,417s	t(G)	0,005s
 Aut(Gr) 	120	 Aut(Gr) 	120
M(k,g)	10	p(n,m,d)	0,236328

Tabuľka 5.1: Tabuľka Cage(3,5)

5.1.2 Cage(3,6) - Heawoodov graf



Obr. 5.3: Cage(3,6) [The]

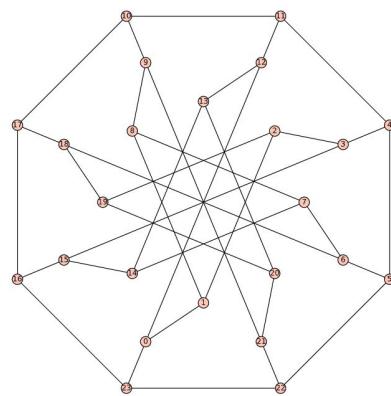
```
[1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 1 0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 1 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1]
```

Obr. 5.4: H (vľavo) a G (vpravo)

Cage(3,6)		C(21,8,6)	
ver	14	d	6
edg	21	m	256
rozmer H	14×21	rozmer G	8×21
t(H)	0,007s	t(G)	0,007s
Aut(Gr)	336	Aut(Gr)	336
M(k,g)	14	p(n,m,d)	0,028320

Tabuľka 5.2: Tabuľka Cage(3,6)

5.1.3 Cage(3,7) - McGeeho graf



Obr. 5.5: Cage(3,7)[The]

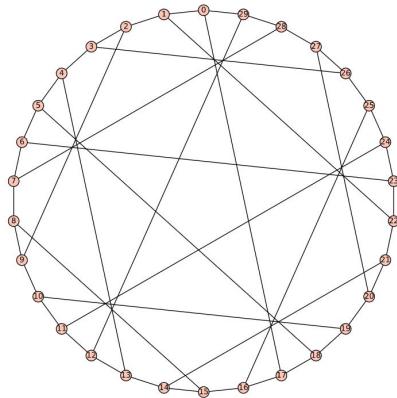
Obr. 5.6: H

Obr. 5.7: G

Cage(3,7)		C(36,13,7)	
ver	24	d	7
edg	36	m	8192
rozmer H	24×36	rozmer G	13×36
t(H)	0,006s	t(G)	0,012s
Aut(Gr)	32	Aut(Gr)	32
M(k,g)	22	p(n,m,d)	0,000931

Tabuľka 5.3: Tabuľka Cage(3,7)

5.1.4 Cage(3,8) - Tutteho-Coxeterov graf



Obr. 5.8: Cage(3,8) [The]

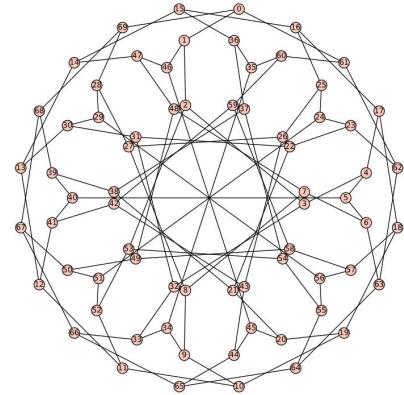
Obr. 5.9: H

Obr. 5.10: G

Cage(3,8)		C(45,16,8)	
ver	30	d	8
edg	45	m	65536
rozmer H	30×45	rozmer G	16×45
t(H)	0,009s	t(G)	0,021s
Aut(Gr)	1440	Aut(Gr)	1440
M(k,g)	30	p(n,m,d)	0,000028

Tabuľka 5.4: Tabuľka Cage(3,8)

5.1.5 Cage(3,10) - Balabanov graf

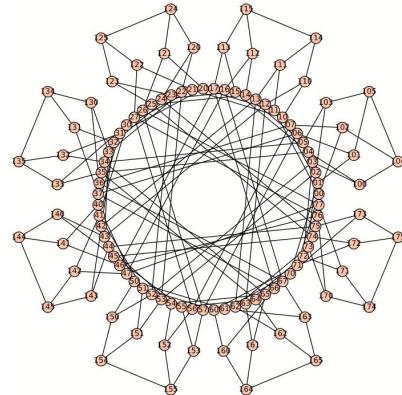


Obr. 5.11: Cage(3,10) [The]

Cage(3,10)		C(105,36,10)	
ver	70	d	10
edg	105	m	68719×10^6
rozměr H	70×105	rozměr G	36×105
t(H)	0,039s	t(G)	0,091s
 Aut(Gr) 	80	 Aut(Gr) 	80
M(k,g)	62	p(n,m,d)	$8,425 \times 10^{-15}$

Tabuľka 5.5: Tabuľka Cage(3,10)

5.1.6 Cage(3,11) - Balabanov graf

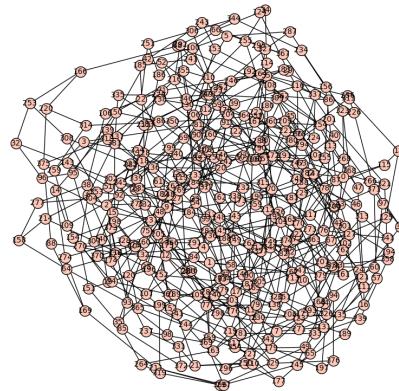


Obr. 5.12: Cage(3,11) [The]

Cage(3,11)		C(168,57,11)	
ver	112	d	11
edg	168	m	14412×10^{13}
rozmer H	112×168	rozmer G	57×168
t(H)	0,095s	t(G)	0,22s
Aut(Gr)	64	Aut(Gr)	64
M(k,g)	94	p(n,m,d)	$4,172 \times 10^{-25}$

Tabuľka 5.6: Tabuľka Cage(3,11)

5.1.7 Cage(3,14)



Obr. 5.13: Rec(3,14)

Rec(3,14)		C(576,193,14)	
ver	384	d	14
edg	576	m	12554×10^{54}
rozmer H	384×586	rozmer G	193×576
t(H)	1,26s	t(G)	2,627s
Aut(Gr)	96	Aut(Gr)	96
M(k,g)	254	p(n,m,d)	$2,535 \times 10^{-102}$

Tabuľka 5.7: Tabuľka Rec(3,14)

5.1.8 Rec(3,16)

Rec(3,16)		C(1440,481,16)	
ver	960	d	16
edg	1440	m	62435×10^{140}
rozmer H	960×1440	rozmer G	481×1440
t(H)	7,389s	t(G)	17,603s
Aut(Gr)	96	Aut(Gr)	96
M(k,g)	510	p(n,m,d)	$5,178 \times 10^{-271}$

Tabuľka 5.8: Tabuľka Rec(3,16)

5.1.9 Rec(3,17)

Rec(3,17)		C(3264,1089,17)	
ver	2176	d	17
edg	3264	m	66323×10^{323}
rozmer H	2176×3264	rozmer G	1089×3264
t(H)	37,479s	t(G)	87,334s
Aut(Gr)	544	Aut(Gr)	544
M(k,g)	766	p(n,m,d)	?

Tabuľka 5.9: Tabuľka Rec(3,17)

5.1.10 Rec(3,18)

Rec(3,18)		C(3840,1281,18)	
ver	2560	d	18
edg	3840	m	41632×10^{381}
rozmer H	2560×3840	rozmer G	1281×3840
t(H)	57,305s	t(G)	121,156s
Aut(Gr)	640	Aut(Gr)	640
M(k,g)	1022	p(n,m,d)	?

Tabuľka 5.10: Tabuľka Rec(3,20)

5.1.11 Cage(3,20)

Rec(3,20)		C(?, ?, 20)	
ver	5376	d	20
edg	8064	m	?
rozmer H	5376×8064	rozmer G	?
t(H)	240,707s	t(G)	?s
Aut(Gr)	2688	Aut(Gr)	2688
M(k,g)	2046	p(n,m,d)	?

Tabuľka 5.11: Tabuľka Rec(3,20)

5.1.12 Rec(3,23)

Rec(3,23)		C(?, ?, 23)	
ver	49326	d	23
edg	73989	m	?
rozmer H	49326 × 73987	rozmer G G	?
t(H)	?s	t(G)	?s
Aut(Gr)	1	Aut(Gr)	1
M(k,g)	6142	p(n,m,d)	?

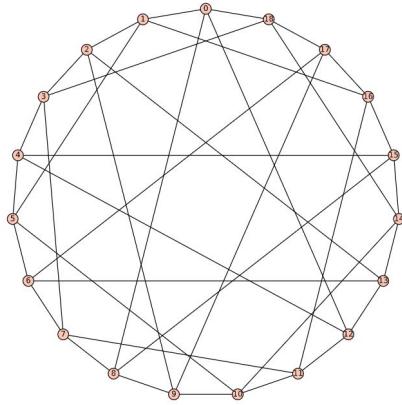
Tabuľka 5.12: Tabuľka Rec(3,23)

5.1.13 Rec(3,25)

Rec(3,25)		C(?, ?, 25)	
ver	108906	d	25
edg	163359	m	?
rozmer H	108906 × 163359	rozmer G	?
t(H)	?s	t(G)	?s
Aut(Gr)	?	Aut(Gr)	?
M(k,g)	12286	p(n,m,d)	?

Tabuľka 5.13: Tabuľka Rec(3,25)

5.1.14 Cage(4,5) - Robertsonov graf



Obr. 5.14: Cage(4,5) [The]

Obr. 5.15: H

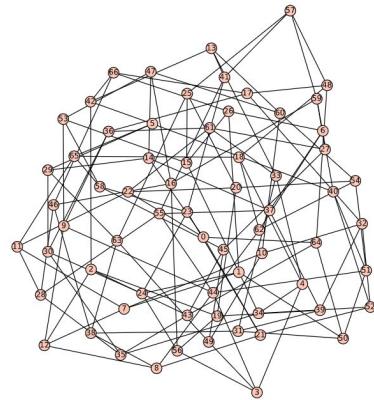
```
[1 0 0 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 1 1 0 0 0 0 0]
[0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 1 1]
[0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0]
[0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0]
[0 0 0 0 0 1 1 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0]
[0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 0 0]
[0 0 0 0 0 0 0 1 0 1 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 0 1 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 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 1 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 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 1 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 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 1 0 1 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 0 1 0 0 0 0]
[0 0 0 0 0 0 0 1 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 0 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 1 1 1 1]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 1 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 1 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0]
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 1 0 0]
```

Obr. 5.16: G

Cage(4,5)		C(38,20,5)	
ver	19	d	5
edg	38	m	1048576
rozmer H	19×38	rozmer G	20×38
t(H)	1,385s	t(G)	0,015s
Aut(Gr)	24	Aut(Gr)	24
M(k,g)	17	p(n,m,d)	0,002831

Tabuľka 5.14: Tabuľka Cage(4,5)

5.1.15 Cage(4,7)

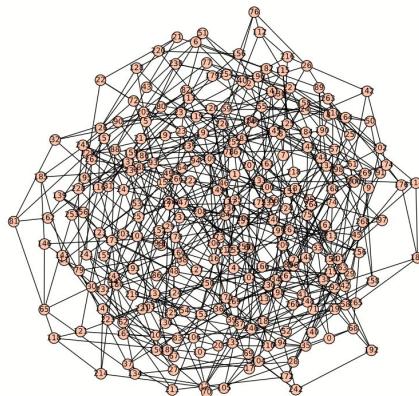


Obr. 5.17: Cage(4,7)

Cage(4,7)		C(134,68,7)	
ver	67	d	7
edg	134	m	29515×10^{16}
rozmer H	67×134	rozmer G	68×134
t(H)	0,046s	t(G)	0,144s
Aut(Gr)	4	Aut(Gr)	4
M(k,g)	53	p(n,m,d)	$5,436 \times 10^{-15}$

Tabuľka 5.15: Tabuľka Cage(4,7)

5.1.16 Cage(4,9)



Obr. 5.18: Cage(4,9)

Cage(4,9)		C(540,271,9)	
ver	270	d	9
edg	540	m	37943×10^{77}
rozmer H	270×540	rozmer G	271×540
t(H)	0,827s	t(G)	2,257s
Aut(Gr)	90	Aut(Gr)	90
M(k,g)	161	p(n,m,d)	$3,721 \times 10^{-72}$

Tabuľka 5.16: Tabuľka Cage(4,9)

5.1.17 Cage(4,10)

Cage(4,10)		C(768,385,10)	
ver	384	d	10
edg	768	m	78804×10^{111}
rozmer H	384×768	rozmer G	385×768
t(H)	1,535s	t(G)	4,707s
Aut(Gr)	768	Aut(Gr)	768
M(k,g)	242	p(n,m,d)	$7,339 \times 10^{-106}$

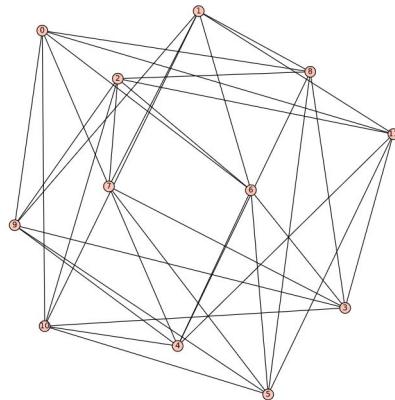
Tabuľka 5.17: Tabuľka Cage(4,10)

5.1.18 Cage(5,10)

Cage(5,10)		C(?, ?, 10)	
ver	1296	d	10
edg	3240	m	31867×10^{581}
rozmer H	1296×3240	rozmer G	?
t(H)	22,806s	t(G)	83,87s
Aut(Gr)	3888	Aut(Gr)	3888
M(k,g)	682	p(n,m,d)	$6,728 \times 10^{-378}$

Tabuľka 5.18: Tabuľka Cage(5,10)

5.1.19 Cage(6,4)



Obr. 5.19: Cage(6,4)

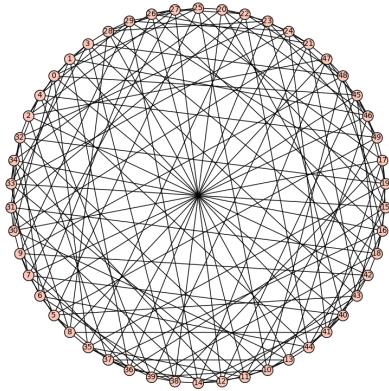
Obr. 5.20: H

Obr. 5.21: G

Cage(6,4)		C(36,25,4)	
ver	12	d	4
edg	36	m	33554432
rozmer H	12×36	rozmer G	25×36
t(H)	0,052s	t(G)	0,013s
Aut(Gr)	1036800	Aut(Gr)	1036800
M(k,g)	12	p(n,m,d)	0,018066

Tabul'ka 5.19: Tabul'ka Cage(6,4)

5.1.20 Cage(7,5)



Obr. 5.22: Cage(7,5)

Cage(7,5)		C(175,126,5)	
ver	50	d	5
edg	175	m	85071×10^{33}
rozmer H	50×175	rozmer G	126×175
t(H)	0,078s	t(G)	0,22s
Aut(Gr)	252000	Aut(Gr)	252000
M(k,g)	50	p(n,m,d)	$2,736 \times 10^{-11}$

Tabuľka 5.20: Tabuľka Cage(7,5)

5.1.21 Cage(7,7)

Cage(7,7)		C(2240,1601,7)	
ver	640	d	7
edg	2240	m	88925×10^{476}
rozmer H	640×2240	rozmer G	1601×2240
t(H)	7,476s	t(G)	39,238s
Aut(Gr)	320	Aut(Gr)	320
M(k,g)	302	p(n,m,d)	$8,212 \times 10^{-184}$

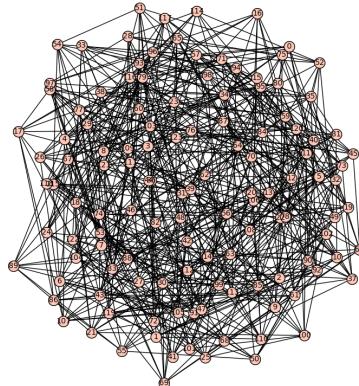
Tabuľka 5.21: Tabuľka Cage(7,7)

5.1.22 Cage(7,8)

Cage(7,8)		C(2352,1681,8)	
ver	672	d	8
edg	2352	m	10750×10^{502}
rozmer H	672×2352	rozmer G	1681×2352
t(H)	8,513s	t(G)	43,11s
Aut(Gr)	14112	Aut(Gr)	14112
M(k,g)	518	p(n,m,d)	$2,213 \times 10^{-193}$

Tabuľka 5.22: Tabuľka Cage(7,8)

5.1.23 Rec(10,5)

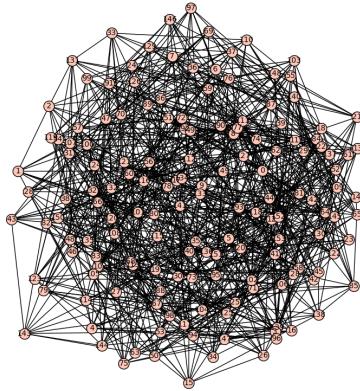


Obr. 5.23: Rec(10,5)

Rec(10,5)		C(620,497,5)	
ver	124	d	5
edg	620	m	40918×10^{144}
rozmer H	124×620	rozmer G	497×620
t(H)	0,44s	t(G)	2,879s
Aut(Gr)	1	Aut(Gr)	1
M(k,g)	101	p(n,m,d)	$1,810 \times 10^{-32}$

Tabuľka 5.23: Tabuľka Rec(10,5)

5.1.24 Rec(11,5)



Obr. 5.24: Rec(11,5)

Rec(11,5)		C(847,694,5)	
ver	154	d	5
edg	847	m	82190×10^{204}
rozmer H	154×847	rozmer G	694×847
t(H)	0,778s	t(G)	5,221s
Aut(Gr)	1	Aut(Gr)	1
M(k,g)	122	p(n,m,d)	$3,145 \times 10^{-41}$

Tabuľka 5.24: Tabuľka Rec(11,5)

5.1.25 Rec(12,5)

Rec(12,5)		C(1218,1016,5)	
ver	203	d	5
edg	1218	m	70222×10^{301}
rozmer H	203×1218	rozmer G	1016×1218
t(H)	1,351s	t(G)	10,986s
Aut(Gr)	203	Aut(Gr)	203
M(k,g)	145	p(n,m,d)	$1,155 \times 10^{-55}$

Tabuľka 5.25: Tabuľka Rec(12,5)

5.1.26 Rec(13,5)

Rec(13,5)		C(1495,1266,5)	
ver	230	d	5
edg	1495	m	12705×10^{377}
rozmer H	230×1495	rozmer G	1266×1495
t(H)	1,927s	t(G)	16,658s
Aut(Gr)	1	Aut(Gr)	1
M(k,g)	170	p(n,m,d)	$1, 296 \times 10^{-63}$

Tabuľka 5.26: Tabuľka Rec(13,5)

5.2 Vyhodnotenie výsledkov

5.2.1 Porovnanie výsledkov

V nasledovnej tabuľke si zobrazíme dosiahnuté výsledky a porovnáme ich s vypočítaným Moorovým ohraničením $M(k,g)$ pre povolený počet vrcholov $|ver|$ ako aj získané lineárne kódy $C(n,m,d)$ s perfektnými pomocou zadaného parametra $p(n,m,d)$.

Cage(k,g)	M(k,g)	rozm. H	Aut(Gr)	rozm. G	p(n,m,d)
Cage(3,5)	10	10×15	120	6×15	0,236328
Cage(3,6)	14	14×21	336	8×21	0,028320
Cage(3,7)	22	24×36	32	13×36	0,000931
Cage(3,8)	30	30×45	1440	16×45	0,000028
Cage(3,10)	62	70×105	80	36×105	$8,425 \times 10^{-15}$
Cage(3,11)	94	112×168	64	57×168	$4,172 \times 10^{-25}$
Rec(3,14)	254	384×576	96	193×576	$2,535 \times 10^{-102}$
Rec(3,16)	510	960×1440	96	481×1440	$5,178 \times 10^{-271}$
Rec(3,17)	766	2176×3264	544	1089×3264	?
Rec(3,18)	1022	2560×3840	640	1281×3840	?
Rec(3,20)	2046	5376×8064	2688	?	
Rec(3,23)	6142	49326×73989	1	?	?
Rec(3,25)	12286	108906×163359	-	?	?

Tabuľka 5.27: Tabuľka overenia dosiahnutých výsledkov č.1

Cage(k,g)	M(k,g)	rozm. H	Aut(Gr)	rozm. G	p(n,m,d)
Cage(4,5)	17	19×38	24	20×38	$0,002831$
Cage(4,7)	53	67×134	4	68×134	$5,436 \times 10^{-15}$
Cage(4,9)	161	270×540	90	271×540	$3,721 \times 10^{-72}$
Cage(4,10)	242	384×768	768	385×768	$7,339 \times 10^{-106}$
Cage(5,10)	682	1296×3240	3888	?	$6,728 \times 10^{-378}$
Cage(6,4)	12	12×36	1036800	25×36	0,018066
Cage(7,5)	50	50×175	252000	126×175	$2,736 \times 10^{-11}$
Cage(7,7)	302	640×2240	320	1601×2240	$8,212 \times 10^{-184}$
Cage(7,8)	518	672×2352	14112	1681×2352	$2,213 \times 10^{-193}$
Rec(10,5)	101	124×620	1	497×620	$1,810 \times 10^{-32}$
Rec(11,5)	122	154×847	1	694×847	$3,145 \times 10^{-41}$
Rec(12,5)	145	203×1218	203	1016×1218	$1,155 \times 10^{-55}$
Rec(13,5)	170	230×1495	1	1266×1495	$1,296 \times 10^{-63}$

Tabuľka 5.28: Tabuľka overenia dosiahnutých výsledkov č.2

5.2.2 Zhrnutie výsledkov

Vo všetkých prípadoch bola splnená podmienka Moorovho ohraničenia $M(k,g)$ zaručujúca existenciu uvažovaných klietok Cage(k,g) alebo rekordných grafov Rec(k,g). Vo všetkých prípadoch sa nám podarilo zistiť rozmer kontrolnej (incidenčnej) matice H, počet automorfizmov $|\text{Aut}(G)|$ až na rekordný graf rec(3,25), kedy program nedokázal vypočítať počet automor-

fizmov $|\text{Aut}(\text{Gr})|$. Správnosť $|\text{Aut}(\text{Gr})|$ sme overili aj pomocou teórie, kde je tento údaj pre niektoré uvažované klietky $\text{Cage}(k,g)$ alebo rekordné grafy $\text{Rec}(k,g)$ známy ako aj údaj o počte vrcholov $|\text{ver}|$. Na základe incidenčnej matice klietky $\text{Cage}(k,g)$ sme boli schopní vygenerovať lineárny kód $C(n,m,d)$ ako aj jeho generujúcu maticu G až na $cage(5, 10)$, $cage(3, 20)$, $cage(3, 23)$ a $cage(3, 25)$, kedy už bolo výpočtovo náročné generujúcu maticu G vygenerovať. Z generujúcich matíc G bolo možné zistiť maximálny počet kódových slov m v kóde a minimálnu kódovú vzdialenosť d , ktorá ako sa zdá, je zhodná s obvodom klietky. Túto hypotézu bude potrebné ešte overiť. Na základe uvedených parametrov bolo možné vypočítať parameter perfektného kódu $p(n,m,d)$, ktorý vyjadruje vzdialenosť nášho kódu od perfektného kódu (v intervale $<0,1>$, kedy 1 znamená prefektný kód). Sledujeme aj čas generovania kontrolných $t(H)$ a generujúcich matíc $t(G)$. Lineárny kód $C(n,m,d)$ sme overili výpočtom, ktorý vyplýva z definície lineárneho kódu a jeho kontrolnej matice H . Vo všetkých prípadoch sme dostali očakávanú nulovú maticu, čo potvrdzuje správnosť vygenerovaného lineárneho kódu $C(n,m,d)$ a jeho generujúcej matice G .

Kapitola 6

Záver

XXXXXX

Literatúra

- [Bor18] Branislav Boráň. Hustota inverzií riedkych cyklických matíc, 2018.
- [CoC] Cocalc. Available at <https://cocalc.com>.
- [EJ13] Geoffrey Exoo and Robert Jajcay. Dynamic cage survey. *The Electronic Journal of Combinatorics*, 2013.
- [Exo] Geoffrey Exoo. Regular graphs of given degree and girth. Available at <http://cs.indstate.edu/ge/CAGES/index.html>.
- [Hil91] R. Hill. *A First Course in Coding Theory*. Oxford applied mathematics and computing science series. Clarendon Press, 1991.
- [HP03] W Cary Huffman and Vera Pless. *Fundamentals of error-correcting codes*. Cambridge Univ. Press, Cambridge, 2003.
- [Mal07] Gabofestwe Alafang Malema. *Low-Density Parity-Check Codes: Construction and Implementation*. PhD thesis, 2007.
- [MTSB13] Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. Mdpc-mceliece: New mceliece variants from moderate density parity-check codes. pages 2069–2073. IEEE, 2013.

[The] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version x.y.z)*.

Zoznam obrázkov

3.1	Vzťah medzi grafom a kontrolnou maticou [Mal07]	7
3.2	Petersenov graf (vľavo) a Heawoodov graf (vpravo) [EJ13] . .	9
3.3	McGeeho graf (vľavo) a Tutteho-Coxeterov graf (vpravo)[EJ13]	9
4.1	Sage a Cocalc[The][CoC]	12
5.1	Cage(3,5) [The]	18
5.2	H (vľavo) a G (vpravo)	19
5.3	Cage(3,6) [The]	20
5.4	H (vľavo) a G (vpravo)	20
5.5	Cage(3,7)[The]	21
5.6	H	22
5.7	G	22
5.8	Cage(3,8) [The]	23
5.9	H	23
5.10	G	24
5.11	Cage(3,10) [The]	25
5.12	Cage(3,11) [The]	26
5.13	Rec(3,14)	27
5.14	Cage(4,5) [The]	30
5.15	H	30

5.16 G	31
5.17 Cage(4,7)	32
5.18 Cage(4,9)	33
5.19 Cage(6,4)	34
5.20 H	35
5.21 G	35
5.22 Cage(7,5)	36
5.23 Rec(10,5)	37
5.24 Rec(11,5)	38