

Konštrukcia Hadamardových matíc a ich aplikácie

Matej Černohorský



ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení:	Bc. Matej Černohorský
Osobní číslo:	A23424
Studijní program:	N0613A140022 Informační technologie
Specializace:	Softwarové inženýrství
Forma studia:	Kombinovaná
Téma práce:	Konstrukce Hadamardových matic a jejich aplikace
Téma práce anglicky:	Construction of Hadamard Matrices and Their Applications

Zásady pro vypracování

- Popíšte definici a základné vlastnosti Hadamardových matic, vrátane kritérií existencie.
- Analyzujte algebraické vlastnosti Hadamardových matic.
- Prezentujte rôzne metódy konštrukcie Hadamardových matic.
- Ilustratívne implementujte algoritmy konštrukcie Hadamardových matic.
- Uvedte prehľad praktických aplikácií Hadamardových matic v rôznych oblastiach.

Forma zpracování diplomové práce: **tištěná/elektronická**
Jazyk zpracování: **Slovenština**

Seznam doporučené literatury:

1. SEBERRY, Jennifer a Mieko YAMADA. *Hadamard Matrices: Constructions using Number Theory and Linear Algebra*. 1. vydání. Hoboken: Wiley, 2020. ISBN 978-1-119-52024-5.
2. HORADAM, K. J. *Hadamard Matrices and Their Applications*. Princeton, New Jersey: Princeton University Press, 2007. ISBN 978-0-691-12444-6.
3. COLBOURN, Charles J. a Jeffrey H. DINITZ, eds. *Handbook of Combinatorial Designs*. 2nd ed. Boca Raton: Chapman & Hall/CRC, 2007. ISBN 978-1-58488-506-1.
4. LANG, Serge. *Algebra*. 3. vydání. New York: Springer, 2002. Graduate Texts in Mathematics, sv. 211. ISBN 978-0-387-95385-4.
5. FRALEIGH, John B. *A First Course in Abstract Algebra*. 7. vydání. Boston: Addison-Wesley, 2003. ISBN 978-0-201-76390-5.

Vedoucí diplomové práce: **Mgr. Jan Krňávek, Ph.D.**
Ústav matematiky

Datum zadání diplomové práce: **27. října 2024**
Termín odevzdání diplomové práce: **2. června 2025**



L.S.

doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan

prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 17. listopadu 2024

PROHLÁŠENÍ AUTORA DIPLOMOVÉ PRÁCE

Beru na vědomí, že

- odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., v platném znění bez ohledu na výsledek obhajoby;
- diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému a bude dostupná k nahlédnutí;
- jedno vyhotovení diplomové práce v listinné podobě bude ponecháno Univerzitě Tomáše Bati ve Zlíně k uložení;
- na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- podle § 60 odst. 1 autorského zákona má Univerzita Tomáše Bati ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- podle § 60 odst. 2 a 3 mohu užít své dílo – diplomovou práci – nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tj. k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá; neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji, že

- jsem na diplomové práci pracoval(a) samostatně a použitou literaturu jsem řádně citoval(a); v případě publikace výsledků budu uveden(a) jako spoluautor;
- odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou obsahově totožné.

Ve Zlíně, dne

.....
podpis autora

ABSTRAKT

Táto práca analyzuje rôzne metódy generovania Hadamardových matíc pomocou matematických prístupov z oblasti teórie čísel, kombinatoriky a algebry. Hlavná časť práce je venovaná analýze a implementácii vybraných konštrukčných metód, pričom sú zhodnotené ich výhody, obmedzenia a možnosti využitia. Ďalej je analyzovaná Hadamardova domnienka, známe a neznáme rády a súčasný stav výskumu v tejto oblasti. Na demonštráciu teoretických poznatkov sú implementované vybrané algoritmy generovania Hadamardových matíc a ich výsledky sú vizualizované. Záverečná časť práce poskytuje prehľad využitia Hadamardových matíc v rôznych technických oblastiach.

Kľúčová slova: Hadamardova matica, Ortogonalita, Konštrukcia matíc, Hadamardova domnienka, Aplikácie Hadamardových matíc

ABSTRACT

This thesis analyzes various methods for generating Hadamard matrices using mathematical approaches from number theory, combinatorics, and algebra. The main part of the work is devoted to the analysis and implementation of selected construction methods with an evaluation of their advantages, limitations, and possible applications. The Hadamard conjecture is also examined, including known and unknown orders and the current state of research in this area. To demonstrate the theoretical findings, selected algorithms for generating Hadamard matrices are implemented and their results are visualized. The final part of the thesis provides an overview of the use of Hadamard matrices in various technical fields.

Keywords: Hadamard matrix, Orthogonality, Matrix construction, Hadamard conjecture, Applications of Hadamard matrices

*Ďakujem vedúcemu mojej diplomovej práce **Mgr. Janovi Krňávkovi, Ph.D.** za odborné vedenie, cenné pripomienky a podporu počas celého procesu vypracovania práce.*

OBSAH

ÚVOD	11
1 HADAMARDOVE MATICE	12
1.1 HISTORICKÝ VÝVOJ.....	12
1.2 ZÁKLADNÉ DEFINÍCIE.....	13
1.3 ORTOGONALITA	13
1.3.1 Vlastnosti ortogonálnych matíc	13
1.4 HADAMARDOVA MATICA	14
1.5 DETERMINANTY A VLASTNÉ ČÍSLA HADAMARDOVÝCH MATÍC.....	16
1.6 HADAMARDOVA DOMNIENKA.....	16
2 ZÁKLADNÉ ALGEBRAICKÉ ŠTRUKTÚRY	18
2.1 MNOŽINY A BINÁRNE OPERÁCIE.....	18
2.2 GRUPY, PODGRUPY A OKRUHY	21
2.3 POLIA A ICH VLASTNOSTI.....	24
2.4 KVADRATICKÉ ZVÝŠKY V KONEČNÝCH POLIACH.....	25
2.5 CÍRKULAČNÉ MATICE	28
2.6 TENZOROVÝ SÚČIN MATÍC	29
3 KONŠTRUKCIA HADAMARDOVÝCH MATÍC	30
3.1 SYLVESTEROVA REKURZÍVNA KONŠTRUKCIA.....	31
3.1.1 Tenzorový súčin ako všeobecný princíp	32
3.2 PALEYHO KONŠTRUKCIE	33
3.3 WILLIAMSONOVA KONŠTRUKCIA	37
3.4 TURYNÖVE METÓDY	42
3.5 KOCYKlickÉ METÓDY	47
4 SÚČASNÝ STAV VÝSKUMU HADAMARDOVEJ DOMNIENKY	55
4.1 NEDÁVNO SKONŠTRUOVANÉ RÁDY	58
4.2 STRATÉGIE PREHL'ADÁVANIA	58
5 APLIKÁCIE HADAMARDOVÝCH MATÍC.....	60

5.1	KÓDOVANIE A KOMUNIKAČNÉ SYSTÉMY	60
5.2	HADAMARDOVE KÓDY	60
5.3	WALSHOVE MATICE A ICH VYUŽITIE V CDMA	63
5.4	EXPERIMENTÁLNY DIZAJN	64
5.5	KVANTOVÉ VÝPOČTY.....	66
5.5.1	Hadamardova brána: Kvantová minca.....	67
5.6	STRUČNÝ PREHL'AD ĎALŠÍCH OBLASTÍ.....	68
ZÁVER.....		69
ZOZNAM POUŽITEJ LITERATÚRY		71
ZOZNAM OBRÁZKOV		74
ZOZNAM TABULIEK		75
ZOZNAM POUŽITÝCH SYMBOLOV A SKRATEK		76
ZOZNAM PRÍLOH		77

ÚVOD

Hadamardove matice, štvorcové matice s prvkami $+1$ alebo -1 , ktorých riadky sú navzájom ortogonálne, predstavujú fascinujúcu oblasť na priesečníku algebry, kombinatoriky a teórie čísel. Vďaka svojim unikátnym vlastnostiam nachádzajú široké uplatnenie v rôznych oblastiach, akými sú teória kódovania, spracovanie signálov, experimentálny dizajn či dokonca kvantové výpočty.

Ústredným problémom v teórii Hadamardových matíc je tzv. *Hadamardova domnienka*, ktorá postuluje existenciu Hadamardovej matice pre každý rád n , ktorý je násobkom štyroch. Hoci bola táto domnienka potvrdená pre mnohé hodnoty n , jej všeobecná platnosť zostáva jedným z najstarších otvorených problémov modernej matematiky. Súčasné konštrukčné metódy, hoci pokročilé, narážajú na limity a nedokážu pokryť všetky teoreticky možné prípady, čo motivuje ďalší výskum.

Hlavným cieľom tejto diplomovej práce je poskytnúť analýzu rôznych metód generovania Hadamardových matíc. Práca sa venuje teoretickému rozboru známych konštrukčných prístupov vrátane Sylvesterovej rekurzívnej konštrukcie, Paleyho konštrukcií založených na vlastnostiach konečných polí, Williamsonovej metódy, Turynových postupov využívajúcich špeciálne sekvencie a kocklických metód. V práci sú analyzované výhody, obmedzenia a vzájomných vzťahov týchto metód. Pre lepšiu demonštráciu teoretických poznatkov sú vybrané algoritmy implementované v jazyku Python a ich výsledky sú vizualizované.

Práca sa taktiež zameriava na súčasný stav Hadamardovej domnienky, sumarizuje známe a neznáme rády Hadamardových matíc a diskutuje algebraické a výpočtové prekážky, ktoré bránia konštrukcii matíc pre niektoré hodnoty. V neposlednom rade ponúka prehľad aplikácií Hadamardových matíc v rôznych technických a vedeckých disciplínach.

Štruktúra práce je nasledovná: Prvá kapitola predstavuje Hadamardove matice, ich historický vývoj, základné definície, vlastnosti a formuluje Hadamardovu domnienku. Druhá kapitola poskytuje prehľad základných algebraických štruktúr, ktoré sú nevyhnutné na pochopenie niektorých konštrukčných metód. Tretia kapitola analyzuje a implementuje vybrané konštrukčné metódy. Štvrtá kapitola sa venuje súčasnému stavu výskumu Hadamardovej domnienky, známym a neznámym rádom a stratégiám prehľadávania. Piata kapitola uzatvára teoretickú časť prehľadom aplikácií Hadamardových matíc v rôznych oblastiach.

1 HADAMARDOVE MATICE

1.1 Historický vývoj

Hadamardove matice, pomenované po francúzskom matematikovi Jacquesovi Hadamardovi, majú korene v druhej polovici 19. storočia. Prvým známym objavom tejto štruktúry bola *Sylvesterova rekurzívna konštrukcia* [1] z roku 1867, ktorá umožňuje vytvárať matice rádov 2^n pomocou tenzorového súčinu s maticou

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Hadamard [2] v roku 1893 ukázal, že ak H je $n \times n$ matica s prvkami ± 1 a jej riadky sú navzájom ortogonálne, potom determinant H dosahuje maximum $n^{n/2}$. Tento výsledok viedol k formulácii tzv. *Hadamardovej domnienky*, ktorá tvrdí, že Hadamardove matice existujú pre všetky rozmery n , ktoré sú násobkom štyroch. Hoci táto hypotéza zostáva otvorená, matice boli konštruované pre veľké množstvo konkrétnych hodnôt n .

V 30. rokoch Paley [3] rozšíril možnosti konštrukcie pomocou konečných polí. Jeho metódy, známe ako *Paleyho konštrukcie typu I a II*, využívajú kvadratické zvyšky a vlastnosti Legendreovho symbolu v prvočíselných moduloch. Tieto konštrukcie pokrývajú množstvo nových rádov, najmä tam, kde algebra poskytuje pevné štrukturálne záruky.

V priebehu 20. storočia sa výskum Hadamardových matíc výrazne rozšíril, pričom boli objavené nové konštrukčné metódy a aplikácie. Kombinatorické metódy, ako sú blokové návrhy, latinské štvorce či Hadamardove rozdielové množiny, umožnili alternatívne konštrukcie ortogonálnych matíc bez nutnosti algebraického základu. Tieto konštrukcie viedli k hlbšiemu prepojeniu medzi Hadamardovými maticami a oblasťou teórie návrhov.

V druhej polovici 20. storočia sa vývoj v oblasti Hadamardových matíc postupne presunul od čisto algebraických a kombinatorických konštrukcií k numerickým a výpočtovým prístupom. Výskumníci začali využívať počítačové simulácie, heuristiky a stochastické metódy na prehľadávanie veľkých priestorov riešení pre špecifické rozmery n , kde klasické konštrukcie už narážali na výpočtové alebo teoretické obmedzenia. Tieto prístupy významne rozšírili množinu známych Hadamardových matíc a otvorili nové smery výskumu v kombinatorickej optimalizácii.

V súčasnosti existujú rozsiahle databázy známych Hadamardových matíc, ako napríklad databáza Catia a Pasechnika [4], ktoré obsahujú tisíce rôznych rozmerov a štruktúr, pre ktoré

bola existencia týchto matíc preukázaná. Tieto databázy vznikli kombináciou klasických algebraických konštrukcií, kombinatorických návrhov, ako aj moderných heuristických a výpočtových prístupov. Napriek významnému pokroku v generovaní matíc pre vysoké rády zostáva Hadamardova domnienka stále otvoreným a zároveň jedným z najstarších nevyriešených problémov modernej matematiky.

1.2 Základné definície

1.3 Ortogonalita

V najjednoduchšom prípade sa ortogonalita interpretuje ako vzájomná kolmosť vektorov, no jej význam siaha oveľa ďalej. V lineárnej algebre sa ortogonálne objekty využívajú pri konštrukcii ortonormálnych báz, čo umožňuje efektívnejšiu reprezentáciu vektorových priestorov. Ortogonálne matice majú výnimočné vlastnosti, ako je zachovanie dĺžky vektorov a jednoduchá inverzia, čo ich robí užitočnými v numerických výpočtoch a aplikáciách v informatike.

Definícia 1.1 (Ortogonalita vektorov). Nech $u, v \in \mathbb{R}^n$ sú dva vektory. Povieme, že u a v sú *ortogonálne*, ak ich skalárny súčin je nulový:

$$\langle u, v \rangle = \sum_{i=1}^n u_i v_i = 0.$$

Definícia 1.2 (Ortogonalita matice). Nech $A \in \mathbb{R}^{n \times n}$ je štvorcová matica. Povieme, že matica A je *ortogonálna*, ak platí:

$$A^T A = A A^T = I_n, \quad (1.1)$$

kde I_n je jednotková matica rádu n .

Definícia 1.3 (Ortonormálna báza). Množina vektorov $\{v_1, v_2, \dots, v_n\}$ v $\mathbb{R}^n$ sa nazýva *ortonormálna báza*, ak platí:

1. Ortogonalita: $\langle v_i, v_j \rangle = 0$, pre všetky $i \neq j$.
2. Normalizácia: $\|v_i\| = 1$, pre všetky i .

1.3.1 Vlastnosti ortogonálnych matíc

Ortogonálne matice majú niekoľko dôležitých vlastností:

- Stĺpce (a riadky) ortogonálnej matice tvoria ortonormálnu bázu v $\mathbb{R}^n$.

- Determinant ortogonálnej matice môže byť ± 1 :

$$|\det A| = 1. \quad (1.2)$$

- Inverzná matica k ortogonálnej matici je jej transponovaná matica:

$$A^{-1} = A^T. \quad (1.3)$$

- Ortogonálne matice zachovávajú euklidovskú normu stĺpcových vektorov, t.j. ak A je ortogonálna matica a $\mathbf{x} \in \mathbb{R}^n$ je stĺpcový vektor, potom platí:

$$\|A\mathbf{x}\| = \|\mathbf{x}\|. \quad (1.4)$$

1.4 Hadamardova matica

Definícia 1.4. Nech H_n je štvorcová matica rádu n , ktorej prvky patria do množiny $\{+1, -1\}$. Matica H_n sa nazýva *Hadamardova matica*, ak spĺňa vzťah:

$$H_n H_n^T = nI_n, \quad (1.5)$$

kde H_n^T je transponovaná matica H_n a I_n je jednotková matica rádu n .

Príklad 1.1. Nasledujúca matica H_4 je Hadamardova matica rádu $n = 4$:

$$H_4 = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & -1 & +1 \end{bmatrix}.$$

Jej transponovaná matica je:

$$H_4^T = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & -1 & +1 \end{bmatrix}.$$

Výpočet súčinu $H_4 H_4^T$:

$$H_4 H_4^T = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} = 4I_4.$$

Keďže podmienka $H_n H_n^T = nI_n$ je splnená, matica H_4 je Hadamardova matica.

Veta 1.1. Nech H_n je Hadamardova matica rádu n . Potom jej riadky a stĺpce tvoria ortogonálnu sústavu vektorov v $\mathbb{R}^n$.

Dôkaz. Z definície Hadamardovej matice vieme, že platí

$$H_n H_n^T = n I_n.$$

To znamená, že skalárny súčin ktoréhokoľvek dvoch rôznych riadkov (alebo stĺpcov) matice H_n je nulový:

$$\langle h_i, h_j \rangle = 0 \quad \text{pre } i \neq j.$$

Zároveň každý riadok má normu $\sqrt{n}$, keďže

$$\langle h_i, h_i \rangle = n.$$

□

Definícia 1.5. Nech H je Hadamardova matica rádu n . Hadamardova matica H sa nazýva **symetrická**, ak platí $H^T = H$ a **anti-symetrická**, ak platí $H^T = -H$.

Príklad 1.2. Nasledujúce matice sú symetrické Hadamardove matice:

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad \begin{bmatrix} -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \end{bmatrix}.$$

Nasledujúce matice sú anti-symetrické Hadamardove matice:

$$\begin{bmatrix} 1 & 1 \\ -1 & 1 \end{bmatrix}, \quad \begin{bmatrix} 1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 \\ -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 \end{bmatrix}.$$

△

Definícia 1.6. Dve Hadamardove matice sa nazývajú **Hadamard ekvivalentné** alebo **H-ekvivalentné**, ak jednu z nich možno získať z druhej vykonaním konečnej postupnosti nasledujúcich elementárnych operácií:

1. Zmenou poradia riadkov alebo stĺpcov.
2. Vynásobením ľubovoľného riadka alebo stĺpca číslom -1 .

Definícia 1.7. Hadamardova matica, ktorá má v prvom riadku a v prvom stĺpci všetky prvky rovné $+1$, sa nazýva **normalizovaná** Hadamardova matica.

Veta 1.2. Nech $n > 2$. Ak $n \times n$ matica H je Hadamardova, tak n je deliteľné 4.

Dôkaz uvádza Zhang vo svojej knihe [5].

1.5 Determinanty a vlastné čísla Hadamardových matíc

Hadamardove matice majú špeciálnu algebraickú štruktúru, ktorá ovplyvňuje ich determinanty a spektrálne vlastnosti.

Veta 1.3 (Hadamardova determinantová nerovnosť). Nech X je ľubovoľná reálna matica rádu n , ktorej prvky patria do množiny $\{+1, -1\}$. Potom platí:

$$|\det X| \leq n^{n/2}. \quad (1.6)$$

Rovnosť nastáva práve vtedy, keď X je Hadamardova matica.

Dôkaz tejto nerovnosti uviedol Hadamard vo svojej práci [2].

Príklad 1.3. Pre Hadamardovu maticu rádu $n = 2$:

$$H_2 = \begin{bmatrix} +1 & +1 \\ +1 & -1 \end{bmatrix},$$

jej determinant je:

$$\det(H_2) = (1 \cdot (-1)) - (1 \cdot 1) = -1 - 1 = -2.$$

Podľa Hadamardovej determinantovej nerovnosti je horná hranica pre $n = 2$ rovná:

$$2^{2/2} = 2.$$

Keďže $|\det H_2| = 2$, táto matica dosahuje maximálnu možnú hodnotu determinantu a teda je Hadamardova. $\triangle$

Táto vlastnosť robí Hadamardove matice dôležitými v oblastiach, kde je potrebné maximalizovať determinanty ako napríklad pri návrhu experimentov alebo pri konštrukcii ortogonálnych kódov.

1.6 Hadamardova domnienka

Hadamardova domnienka patrí medzi najdlhšie otvorené problémy v kombinatorickej matematike. Týka sa existencie Hadamardových matíc pre všetky rozmery, ktoré sú deliteľné štyrmi.

Domnienka 1.1 (Hadamardova domnienka). Pre každé n , ktoré je deliteľné štyrmi, existuje

Hadamardova matica rádu n , teda

$$\forall n \in \mathbb{N}, \quad n \equiv 0 \pmod{4} \implies \exists H_n \in \{+1, -1\}^{n \times n}, \quad H_n H_n^T = nI_n.$$

Táto hypotéza bola prvýkrát explicitne formulovaná Jacquesom Hadamardom v roku 1893 vo svojej práci [2] a odvtedy zostáva nevyriešeným problémom. Hadamardova domnienka má viacero ekvivalentných formulácií a jej pravdivosť by znamenala existenciu rozsiahlej triedy ortogonálnych návrhov s dôležitými aplikáciami v experimentálnom dizajne, spracovaní signálu a teórii kódovania.

2 ZÁKLADNÉ ALGEBRAICKÉ ŠTRUKTÚRY

Táto kapitola sa venuje základným pojmom abstraktnej algebry, ktoré sú nevyhnutné pre pochopenie algebraických metód používaných pri konštrukcii Hadamardových matic. Najprv sú predstavené binárne operácie a ich vlastnosti, následne sú zadané grupy a polia, pričom zvláštna pozornosť je venovaná konečným poliam a ich štruktúre. Ďalej sú definované pojmy kvadratických zvyškov v konečných poliach, multiplikatívnych znakov a ich ortogonálnych vlastností.

2.1 Množiny a binárne operácie

Základy algebraických štruktúr vychádzajú z definície množín a operácií definovaných na týchto množinách.

Definícia 2.1. *Množina* je súbor navzájom rozlíšiteľných objektov, ktoré sa nazývajú *prvky množiny*. Každý prvok je buď v množine, alebo nie je. Ak a je prvkom množiny A , píšeme $a \in A$; ak nie je, píšeme $a \notin A$.

Množinu možno určiť:

- *vymenovaním prvkov*, napr. $A = \{1, 2, 3\}$,
- *vlastnosťou prvkov*, napr. $A = \{x \in \mathbb{Z} \mid x > 0\}$.

Dve množiny A a B sú si rovné ($A = B$), ak obsahujú tie isté prvky, teda

$$\forall x : (x \in A \Leftrightarrow x \in B).$$

Definícia 2.2. Nech $a, b \in \mathbb{Z}$ a $m \in \mathbb{N}$, $m \geq 1$. Čísla a a b nazývame *kongruentné modulo m* , ak rozdiel $a - b$ je deliteľný číslom m . Zapisujeme to ako

$$a \equiv b \pmod{m},$$

pričom m sa nazýva *modul kongruencie*. Ak a a b nie sú kongruentné modulo m , píšeme $a \not\equiv b \pmod{m}$.

Príklad 2.1. Platí, že čísla 10 a 3 sú kongruentné modulo 7, pretože $10 - 3 = 7$ je deliteľné 7, teda

$$10 \equiv 3 \pmod{7}.$$

△

Definícia 2.3. Pre $n \in \mathbb{N}$, $n \geq 2$, definujeme množinu zvyškov modulo n ako

$$\mathbb{Z}_n = \{0, 1, 2, \dots, n-1\}.$$

Každý prvok $a \in \mathbb{Z}_n$ predstavuje zvyšok, ktorý môže vzniknúť po delení celého čísla číslom n . Celé číslo $b \in \mathbb{Z}$ má zvyšok a , ak rozdiel $b - a$ je deliteľný n a teda platí $b \equiv a \pmod{n}$.

Definícia 2.4. Nech $\mathbb{Z}_n$ je množina zvyškov modulo n . Označenie $\mathbb{Z}_n^*$ predstavuje podmnožinu všetkých prvkov $a \in \mathbb{Z}_n$, ku ktorým existuje prvok $b \in \mathbb{Z}_n$ taký, že

$$a \cdot b \equiv 1 \pmod{n}.$$

Taký prvok b nazývame *multiplikatívny inverz* k a modulo n . Množina $\mathbb{Z}_n^*$ teda obsahuje všetky invertovateľné prvky v $\mathbb{Z}_n$ vzhľadom na násobenie modulo n .

Veta 2.1. Ak je n prvočíslo, potom každý nenulový prvok v $\mathbb{Z}_n$ má multiplikatívny inverz. Platí teda:

$$\mathbb{Z}_n^* = \mathbb{Z}_n \setminus \{0\}.$$

Dôkaz je založený na Bézoutovej identite a je uvedený v [6].

Príklad 2.2. Nech $n = 7$. Množina zvyškov modulo 7 je

$$\mathbb{Z}_7 = \{0, 1, 2, 3, 4, 5, 6\}.$$

Invertovateľné prvky modulo 7 sú tie, ku ktorým existuje multiplikatívny inverz v $\mathbb{Z}_7$.
Napríklad:

$$3 \cdot 5 = 15 \equiv 1 \pmod{7},$$

takže 5 je inverzný prvok k 3 modulo 7.

Celá množina invertovateľných prvkov je

$$\mathbb{Z}_7^* = \{1, 2, 3, 4, 5, 6\},$$

keďže všetky nenulové prvky v $\mathbb{Z}_7$ majú inverz (pretože 7 je prvočíslo). △

Na množinách môžeme definovať rôzne typy predpisov, ktorými prvkom množiny alebo ich kombináciám priradíme nové hodnoty. Základným nástrojom na formálne vyjadrenie takéhoto priradovania je zobrazenie, ktoré je základom aj pre definovanie algebraických operácií. Osobitným prípadom takéhoto zobrazenia je binárna operácia, ktorá každému usporiadanému páru prvkov z množiny priraduje opäť prvok tej istej množiny.

Definícia 2.5. Nech A a B sú množiny. *Zobrazenie* (alebo *funkcia*) f z množiny A do množiny B je predpis, ktorý každému prvku $a \in A$ priradí práve jeden prvok $f(a) \in B$. Zapisujeme: $f : A \rightarrow B$.

Definícia 2.6. Nech A je neprázdna množina. *Binárna operácia* na množine A je zobrazenie

$$* : A \times A \rightarrow A,$$

ktoré každým dvom prvkom $a, b \in A$ priradí prvok $a * b \in A$. Hovoríme, že množina A je *uzavretá* vzhľadom na operáciu $*$, ak pre všetky $a, b \in A$ je $a * b \in A$.

Definícia 2.7. Binárna operácia $*$ na množine A sa nazýva:

- *komutatívna*, ak pre všetky $a, b \in A$ platí $a * b = b * a$,
- *asociatívna*, ak pre všetky $a, b, c \in A$ platí $(a * b) * c = a * (b * c)$.

Definícia 2.8. Prvok $e \in A$ sa nazýva *neutrálny prvok* vzhľadom na operáciu $*$, ak pre všetky $a \in A$ platí:

$$a * e = e * a = a.$$

Definícia 2.9. Pre daný prvok $a \in A$ hovoríme, že prvok $b \in A$ je *inverzný k prvku a* vzhľadom na operáciu $*$, ak platí:

$$a * b = b * a = e,$$

kde e je neutrálny prvok.

Príklad 2.3. Uvažujme množinu celých čísel $\mathbb{Z}$ a binárnu operáciu sčítania $+$.

- Pre všetky $a, b \in \mathbb{Z}$ platí $a + b \in \mathbb{Z}$, teda operácia je uzavretá.
- Operácia $+$ je komutatívna, keďže $a + b = b + a$.
- Je tiež asociatívna, pretože $(a + b) + c = a + (b + c)$ pre všetky $a, b, c \in \mathbb{Z}$.
- Neutrálnym prvkom je číslo 0, pretože $a + 0 = 0 + a = a$.
- Inverzný prvok k $a \in \mathbb{Z}$ je $-a$, pretože $a + (-a) = 0$.

Teda $(\mathbb{Z}, +)$ je príklad množiny s asociatívnou, komutatívnou binárnou operáciou s neutrálnym a inverzným prvkom. △

Príklad 2.4. Uvažujme množinu prirodzených čísel $\mathbb{N}$ a operáciu odčítania $-$. Táto operácia nie je binárna operácia na $\mathbb{N}$, pretože nie je uzavretá: napríklad $3 - 5 = -2 \notin \mathbb{N}$. Teda

$$- : \mathbb{N} \times \mathbb{N} \nrightarrow \mathbb{N}.$$

△

Pochopenie vlastností binárnych operácií je dôležité pre definovanie vyšších algebraických štruktúr, ako sú grupy a polia, ktoré budú rozpracované v ďalších častiach.

2.2 Grupy, podgrupy a okruhy

Grupy sú základné algebraické štruktúry. Vychádzajú z množiny, na ktorej je definovaná binárna operácia spĺňajúca určité vlastnosti. Pojem grupy tvorí základ pre definovanie polí, kde sčítanie a násobenie tvoria dve rozdielne grupové štruktúry.

Definícia 2.10. Nech G je neprázdna množina a $*$: $G \times G \rightarrow G$ je binárna operácia. Usporiadanú dvojicu $(G, *)$ nazývame *grupa*, ak sú splnené nasledovné podmienky:

- **Asociativita:** Pre všetky $a, b, c \in G$ platí, že $(a * b) * c = a * (b * c)$.
- Existencia **neutrálneho prvku:** Existuje prvok $e \in G$, taký, že pre každý prvok $a \in G$ platí $a * e = e * a = a$.
- Existencia **inverzného prvku:** Pre každý prvok $a \in G$ existuje inverzný prvok $a^{-1} \in G$, taký, že $a * a^{-1} = a^{-1} * a = e$, kde e je neutrálny prvok.

Definícia 2.11. Grupa $(G, *)$ sa nazýva *komutatívna* alebo *abelovská*, ak pre všetky $a, b \in G$ platí:

$$a * b = b * a.$$

Komutatívne grupy sú základom aditívnej štruktúry v poliach a okruhoch, zatiaľ čo nekomutatívne grupy sa objavujú napríklad pri štúdiu symetrií a permutácií.

Príklad 2.5. Grupa $(\mathbb{Q} \setminus \{0\}, \cdot)$ je množina nenulových racionálnych čísel s operáciou násobenia. Násobenie je asociatívne a komutatívne, neutrálny prvok je 1 a každé $a \in \mathbb{Q} \setminus \{0\}$ má inverzný prvok $\frac{1}{a}$, takže ide o komutatívnu grupu. $\triangle$

Definícia 2.12. *Rád grupy* $(G, *)$ je počet jej prvkov. Označuje sa $|G|$. Grupa sa nazýva *konečná*, ak $|G| < \infty$, inak je *nekonečná*.

Definícia 2.13. Nech $(G, *)$ je grupa a nech $g \in G$. **Rád prvku** g , označený ako $|g|$, je najmenšie kladné celé číslo n , také, že $g^n = e$, kde g^n znamená aplikáciu grupovej operácie $*$ na prvok g celkom n -krát a e je neutrálny prvok grupy (taký prvok, že $e * g = g * e = g$ pre všetky $g \in G$). Ak neexistuje takéto kladné celé číslo n , hovoríme, že rád prvku g je nekonečný.

Príklad 2.6. Uvažujme grupu $(\mathbb{Z}_6, +)$, teda množinu zvyškov modulo 6 so sčítaním modulo 6:

$$\mathbb{Z}_6 = \{0, 1, 2, 3, 4, 5\}.$$

- Grupa $(\mathbb{Z}_6, +)$ má 6 prvkov, teda jej rád je $|\mathbb{Z}_6| = 6$.
- Prvok $[2]$ má rád 3, pretože $2 + 2 + 2 = 6 \equiv 0 \pmod{6}$ a žiadny menší počet sčítaní nedá neutrálny prvok.
- Prvok $[3]$ má rád 2, keďže $3 + 3 = 6 \equiv 0 \pmod{6}$.
- Prvok $[1]$ má rád 6, lebo až 6-násobné sčítanie dá neutrálny prvok:

$$1 + 1 + 1 + 1 + 1 + 1 = 6 \equiv 0 \pmod{6}.$$

△

Po zavedení štruktúry grupy je prirodzené pýtať sa, ako možno porovnávať dve rôzne grupy navzájom. Na to slúžia tzv. zobrazenia, ktoré zachovávajú algebraickú štruktúru – homomorfizmy a izomorfizmy.

Definícia 2.14. Nech $(G, *)$ a $(H, \circ)$ sú grupy. Zobrazenie $\varphi : G \rightarrow H$ sa nazýva *homomorfizmus grup*, ak pre všetky $a, b \in G$ platí:

$$\varphi(a * b) = \varphi(a) \circ \varphi(b).$$

Definícia 2.15. Homomorfizmus $\varphi : G \rightarrow H$ sa nazýva *izomorfizmus*, ak je navyše *bijektívny*, t. j. je súčasne *injektívny* (rôzne prvky sa zobrazujú na rôzne) a *surjektívny* (každý prvok v H je obrazom nejakého prvku z G). V prípade existencie takého izomorfizmu hovoríme, že grupy G a H sú *izomorfné*, a píšeme $G \cong H$.

Poznámka. Izomorfné grupy majú rovnaké algebraické vlastnosti – napríklad rovnaký rád, rovnakú štruktúru podgrup a podobne. Líšia sa len zápisom prvkov alebo konkrétnou reprezentáciou operácie.

Definícia 2.16. Nech $(G, *)$ je grupa a $H \subseteq G$ je neprázdna podmnožina. Ak H spolu s operáciou $*$ tvorí samu o sebe grupu, potom H nazývame *podgrupa* grupy G .

Príklad 2.7. Uvažujme grupu $(\mathbb{Z}, +)$, teda množinu celých čísel so sčítaním. Množina párnych čísel

$$H = \{\dots, -4, -2, 0, 2, 4, \dots\} = 2\mathbb{Z}$$

tvorí podgrupu grupy $(\mathbb{Z}, +)$, pretože:

- Je neprázdna: $0 \in H$,
- Je uzavretá na sčítanie: ak $a, b \in H$, potom $a + b \in H$,

- Ku každému $a \in H$ existuje $-a \in H$, teda každý prvok má inverz v H .

Teda $(2\mathbb{Z}, +)$ je podgrupa $(\mathbb{Z}, +)$.

△

Príklad 2.8. Zobrazenie $\varphi : (\mathbb{Z}, +) \rightarrow (2\mathbb{Z}, +)$ definované predpisom $\varphi(n) = 2n$ je izomorfizmus grup, pretože:

$$\varphi(a + b) = 2(a + b) = 2a + 2b = \varphi(a) + \varphi(b),$$

a zároveň je bijektívne. Teda $(\mathbb{Z}, +) \cong (2\mathbb{Z}, +)$.

△

Definícia 2.17. Nech $(G, *)$ je grupa. Hovoríme, že G je *cyklická grupa*, ak existuje prvok $g \in G$, taký, že každý prvok v G môže byť zapísaný ako g^n pre nejaké celé číslo n . Prvok g sa potom nazýva *generátor* cyklickej grupy G . Formálne zapisujeme, že $G = \langle g \rangle = \{g^n | n \in \mathbb{Z}\}$, kde $\mathbb{Z}$ je množina celých čísel a g^n je aplikácia grupovej operácie $*$ na prvok g , n -krát.

Príklad 2.9. Uvažujme množinu $\mathbb{Z}_7 = \{0, 1, 2, 3, 4, 5, 6\}$ so sčítaním modulo 7. Táto množina spolu s operáciou $+$ tvorí komutatívnu cyklickú grupu $(\mathbb{Z}_7, +)$ rádu 7. Generátorom tejto grupy je napríklad prvok $1 \in \mathbb{Z}_7$, pretože:

$$\langle 1 \rangle = \{1, 1 + 1 = 2, 1 + 1 + 1 = 3, \dots, 1 + \dots + 1 = 0 \pmod{7}\} = \mathbb{Z}_7.$$

Každý prvok v $\mathbb{Z}_7$ možno teda vyjadriť ako $n \cdot 1$ (sčítanie prvku 1 so sebou n -krát), čo formálne znamená $\mathbb{Z}_7 = \langle 1 \rangle$.

Podobne, množina $\mathbb{Z}_7^* = \{1, 2, 3, 4, 5, 6\}$ spolu s operáciou násobenia modulo 7 tvorí multiplikatívnu cyklickú grupu $(\mathbb{Z}_7^*, \cdot)$ rádu 6. Generátorom tejto grupy je napríklad prvok 3, pretože:

$$\langle 3 \rangle = \{3^1 = 3, 3^2 = 2, 3^3 = 6, 3^4 = 4, 3^5 = 5, 3^6 = 1\} \pmod{7} = \mathbb{Z}_7^*.$$

Tento príklad ukazuje obe základné formy cyklickej grupy – aditívnu aj multiplikatívnu.

△

Definícia 2.18. Nech R je neprázdna množina vybavená dvoma binárnymi operáciami: sčítaním $+$ a násobením $\cdot$. Usporiadanú trojicu $(R, +, \cdot)$ nazývame *okruh*, ak platí:

1. $(R, +)$ je komutatívna grupa s neutrálnym prvkom 0,
2. $\cdot$ je asociatívna binárna operácia na R ,
3. platí distribučný zákon: pre všetky $a, b, c \in R$ platí

$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{a} \quad (a + b) \cdot c = a \cdot c + b \cdot c.$$

Ak navyše násobenie $\cdot$ je komutatívne, hovoríme o *komutatívnom okruhu*, a ak existuje neutrálny prvok 1 vzhľadom na násobenie, ide o *okruh s jednotkou*.

Na grupy nadväzujú ďalšie algebraické štruktúry – polia. Ich aditívna a multiplikatívna časť sú komutatívne grupy definované nad tou istou množinou. Nasleduje ich presná definícia a základné vlastnosti.

2.3 Polia a ich vlastnosti

Polia predstavujú základnú algebraickú štruktúru pre aritmetiku, ktorá sa využíva v mnohých oblastiach – lineárnej algebre, teórii kódovania ale aj pri konštrukcii Hadamardových matic.

Definícia 2.19. Nech F je neprázdna množina vybavená dvoma binárnymi operáciami: sčítaním $+: F \times F \rightarrow F$ a násobením $\cdot: F \times F \rightarrow F$. Usporiadanú trojicu $(F, +, \cdot)$ nazývame *pole*, ak sú splnené nasledujúce podmienky:

1. $(F, +)$ je komutatívna grupa s neutrálnym prvkom $0 \in F$,
2. $(F \setminus \{0\}, \cdot)$ je komutatívna grupa s neutrálnym prvkom $1 \in F$, teda každý nenulový prvok má multiplikatívny inverz,
3. Platí *distributívny zákon*: pre všetky $a, b, c \in F$ platí

$$a \cdot (b + c) = a \cdot b + a \cdot c.$$

Poznámka. V komutatívnom okruhu s jednotkou sú definované obe operácie, sčítanie tvorí komutatívnu grupu a násobenie je asociatívne, komutatívne a má neutrálny prvok. Nie každý nenulový prvok však musí mať multiplikatívny inverz.

Pole je špeciálny typ komutatívneho okruhu, v ktorom navyše každý nenulový prvok má inverz vzhľadom na násobenie. Inými slovami, v poli je možné vykonávať delenie nenulovými prvkami, čo v okruhu vo všeobecnosti nie je možné.

Príklad 2.10. Množina celých čísel $(\mathbb{Z}, +, \cdot)$ tvorí komutatívny okruh s jednotkou. Sčítanie aj násobenie sú asociatívne, komutatívne a distribučné, pričom neutrálnym prvkom pre sčítanie je 0 a pre násobenie je 1.

Avšak nie každý nenulový prvok má inverz vzhľadom na násobenie. Napríklad číslo 2 nemá multiplikatívny inverz v množine $\mathbb{Z}$, pretože neexistuje celé číslo x , pre ktoré by platilo $2 \cdot x = 1$. Preto $\mathbb{Z}$ nie je polom.

Naopak, množina racionálnych čísel $(\mathbb{Q}, +, \cdot)$ tvorí pole, pretože každé nenulové racionálne číslo a má inverz $\frac{1}{a}$. Napríklad inverzný prvok k 2 je $\frac{1}{2}$, keďže $2 \cdot \frac{1}{2} = 1$. $\triangle$

Definícia 2.20. Pole F sa nazýva *konečné*, ak obsahuje konečný počet prvkov. Počet prvkov v konečnom poli sa nazýva *řád pol'a* a označuje sa $|F|$.

Definícia 2.21. *Galoisovo pole* (angl. *Galois field*) je konečné pole, ktoré obsahuje práve $q = p^k$ prvkov, kde p je prvočíslo a $k \in \mathbb{N}$, $k \geq 1$. Takéto pole označujeme ako $\mathbb{F}_q$ alebo $GF(q)$.

Číslo p sa nazýva *charakteristika pol'a* a predstavuje najmenšie kladné celé číslo, pre ktoré platí

$$p \cdot 1 = 0$$

v danom poli, teda súčet jednotky so sebou samou p -krát dáva nulový prvok. Všetky konečné polia majú charakteristiku rovnu nejakému prvočíslu p .

Číslo k sa nazýva *stupeň rozšírenia* a určuje, o koľko je pole $\mathbb{F}_{p^k}$ väčšie než základné pole $\mathbb{F}_p$. Pole $\mathbb{F}_{p^k}$ tvorí vektorový priestor dimenzie k nad základným pol'om $\mathbb{F}_p$.

Poznámka. Ak $k = 1$, teda $q = p$, potom Galoisovo pole $\mathbb{F}_p$ je izomorfné s pol'om zvyškov modulo p , teda $\mathbb{F}_p \cong \mathbb{Z}_p$.

Veta 2.2. Pre každé prvočíslo p a každé celé číslo $k \geq 1$ existuje (a až na izomorfizmus jediné) konečné pole s p^k prvkami. Označuje sa $\mathbb{F}_{p^k}$.

Dôkaz je možné nájsť v [7].

Poznámka. Veta o jedinečnosti konečného pol'a hovorí, že ak existuje konečné pole s p^k prvkami, potom je (až na izomorfizmus) jediné.

Galoisove polia, najmä tie s charakteristikou 2, sa využívajú pri konštrukcii Hadamardových matíc pomocou kvadratických zvyškov alebo Walshových transformácií.

2.4 Kvadratické zvyšky v konečných poliach

Kvadratické a nekvadratické zvyšky predstavujú základný pojem v aritmetike konečných polí. Vzhľadom na ich symetrické vlastnosti a štruktúru v multiplikatívnej grupe pol'a sa ukazujú ako veľmi užitočné pri konštrukcii pravidelných a ortogonálnych štruktúr. Práve tieto vlastnosti ich predurčujú na využitie v algebraických metódach pri generovaní Hadamardových matíc. Štúdium kvadratických zvyškov tak tvorí fundamentálny krok k pochopeniu, ako algebraické vlastnosti konečných polí môžu viesť ku konkrétnym konštrukciám Hadamardových matíc.

Definícia 2.22. Nech $\mathbb{F}_q$ je konečné pole s q prvkami, kde q je mocnina nepárneho prvočísla. Prvok $a \in \mathbb{F}_q^\times$ (t. j. $a \neq 0$) nazývame *kvadratický zvyšok*, ak existuje $y \in \mathbb{F}_q$ také, že $y^2 = a$. V opačnom prípade sa a nazýva *nekvadratický zvyšok*.

Množinu kvadratických zvyškov v $\mathbb{F}_q$ označujeme ako $QR(q)$, množinu nekvadratických zvyškov ako $QNR(q)$.

Veta 2.3. Nech q je mocnina nepárneho prvočísla a $a \in \mathbb{F}_q^\times$. Potom platí:

$$a \in QR(q) \iff a^{\frac{q-1}{2}} = 1, \quad a \in QNR(q) \iff a^{\frac{q-1}{2}} = -1.$$

Okrem toho:

$$|QR(q)| = |QNR(q)| = \frac{q-1}{2}.$$

Dôkaz je uvedený v [8].

Poznámka. Z predchádzajúcej vety vyplýva, že množina všetkých nenulových prvkov konečného poľa $\mathbb{F}_q^\times$ sa rozdeľuje na dve disjunktné množiny: kvadratické zvyšky a nekvadratické zvyšky. Platí, že tieto dve množiny majú dohromady $q-1$ prvkov.

Príklad 2.11. Nech $\mathbb{F}_7 = \{0, 1, 2, 3, 4, 5, 6\}$ je konečné pole so siedmimi prvkami, kde operácie sčítania a násobenia sa vykonávajú modulo 7. Určíme, ktoré nenulové prvky z tejto množiny sú druhé mocniny iných prvkov — teda ktoré z nich sú kvadratické zvyšky.

Vypočítajme druhé mocniny všetkých prvkov poľa okrem nuly:

$$\begin{aligned} 1^2 &\equiv 1 \pmod{7}, \\ 2^2 &\equiv 4 \pmod{7}, \\ 3^2 &\equiv 2 \pmod{7}, \\ 4^2 &\equiv 2 \pmod{7}, \\ 5^2 &\equiv 4 \pmod{7}, \\ 6^2 &\equiv 1 \pmod{7}. \end{aligned}$$

Zo získaných hodnôt môžeme určiť množinu všetkých kvadratických zvyškov, teda čísel, ktoré sa dajú zapísať ako druhá mocnina nejakého nenulového prvku poľa:

$$QR(7) = \{1, 2, 4\}.$$

Zvyšné nenulové prvky poľa, ktoré sa medzi druhými mocninami nevyskytli, teda $\{3, 5, 6\}$, nazývame nekvadratické zvyšky:

$$QNR(7) = \{3, 5, 6\}.$$

Vidíme, že počet kvadratických aj nekvadratických zvyškov je rovnaký a rovný $3 = \frac{7-1}{2}$, čo je v súlade s teóriou. $\triangle$

Definícia 2.23 (Legendreov symbol). Nech p je nepárne prvočíslo a $\mathbb{F}_p$ je konečné pole rádu p . Legendreov symbol je zobrazenie

$$\left(\frac{a}{p}\right) : \mathbb{F}_p \rightarrow \{0, 1, -1\},$$

ktoré je definované nasledovne:

$$\left(\frac{a}{p}\right) = \begin{cases} 0 & \text{ak } a = 0, \\ 1 & \text{ak } a \text{ je kvadratický zvyšok v } \mathbb{F}_p, \\ -1 & \text{ak } a \text{ nie je kvadratický zvyšok v } \mathbb{F}_p. \end{cases}$$

Legendreov symbol umožňuje jednoduchým spôsobom určiť, či daný prvok konečného poľa je kvadratickým zvyškom.

Legendreov symbol možno efektívne implementovať aj programovo. V nasledujúcom zápise v jazyku Python je využité tzv. *Eulerovo kritérium*, podľa ktorého pre prvočíslo p platí:

$$\left(\frac{a}{p}\right) \equiv a^{\frac{p-1}{2}} \pmod{p}.$$

Výsledná hodnota je potom 1 pre kvadratické zvyšky, -1 pre nekvadratické zvyšky a 0 pre $a = 0$.

```

1 def legendre_symbol(a, p):
2     a = a % p
3     if a == 0:
4         return 0
5     return 1 if pow(a, (p - 1) // 2, p) == 1 else -1

```

Kód 2.1 Výpočet Legendreovho symbolu v jazyku Python.

Veta 2.4 (Multiplikativita Legendreovho symbolu). Nech p je nepárne prvočíslo. Potom pre všetky $a, b \in \mathbb{F}_p$ platí:

$$\left(\frac{ab}{p}\right) = \left(\frac{a}{p}\right) \cdot \left(\frac{b}{p}\right).$$

Dôkaz tejto vlastnosti možno nájsť v klasickej literatúre, napríklad v [9].

Veta 2.5 (Ortogonalita Legendreovho symbolu). Nech p je nepárne prvočíslo. Potom platí:

$$\sum_{a \in \mathbb{F}_p} \left(\frac{a}{p}\right) = 0.$$

Príklad 2.12. Uvažujme pole $\mathbb{F}_7$. Kvadratické zvyšky v tomto poli sú $\{1, 2, 4\}$ a nekvadra-

tické zvyšky sú $\{3, 5, 6\}$ (pozri **Príklad 2.11**). Legendreov symbol nad $\mathbb{F}_7$ má nasledovné hodnoty:

$$\begin{aligned}\left(\frac{0}{7}\right) &= 0, \\ \left(\frac{1}{7}\right) &= 1, \quad \left(\frac{2}{7}\right) = 1, \quad \left(\frac{4}{7}\right) = 1, \\ \left(\frac{3}{7}\right) &= -1, \quad \left(\frac{5}{7}\right) = -1, \quad \left(\frac{6}{7}\right) = -1.\end{aligned}$$

Multiplikativitu môžeme demonštrovať napríklad takto:

$$\left(\frac{2 \cdot 4}{7}\right) = \left(\frac{8}{7}\right) = \left(\frac{1}{7}\right) = 1, \quad \text{a pritom} \quad \left(\frac{2}{7}\right) \cdot \left(\frac{4}{7}\right) = 1 \cdot 1 = 1.$$

Zároveň platí ortogonalita: súčet všetkých hodnôt Legendreovho symbolu nad poľom $\mathbb{F}_7$ je nulový:

$$\sum_{x \in \mathbb{F}_7} \left(\frac{x}{7}\right) = \left(\frac{0}{7}\right) + \left(\frac{1}{7}\right) + \cdots + \left(\frac{6}{7}\right) = 0 + 1 + 1 + (-1) + 1 + (-1) + (-1) = 0.$$

△

2.5 Cirkulačné matice

Definícia 2.24. *Cirkulačná matica* je špeciálny typ štvorcovej matice, pri ktorej každý riadok vznikne cyklickým posunutím predchádzajúceho riadku doprava. Ak $a = (a_0, a_1, \dots, a_{n-1})$ je riadkový vektor, potom cirkulačná matica C vytvorená z a má tvar:

$$C = \begin{bmatrix} a_0 & a_1 & \cdots & a_{n-1} \\ a_{n-1} & a_0 & \cdots & a_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ a_1 & a_2 & \cdots & a_0 \end{bmatrix}.$$

Cirkulačné matice sa často využívajú v kombinatorike, spracovaní signálov a najmä pri konštrukcii Hadamardových matíc.

Príklad 2.13. Nech $a = (1, 2, 3)$. Potom cirkulačná matica C vytvorená z a má tvar:

$$C = \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{bmatrix}.$$

Každý riadok je cyklickým posunom predchádzajúceho. Takéto matice sa neskôr využijú pri generovaní Hadamardových matíc pomocou sekvencií. △

2.6 Tenzorový súčin matíc

Tenzorový súčin (často označovaný aj ako Kroneckerov súčin) je základná operácia na maticiach, ktorá umožňuje konštruovať väčšie matice z menších. V kontexte Hadamardových matíc sa používa na rekurzívne generovanie väčších matíc pri zachovaní ortogonalít.

Definícia 2.25. Nech $A = [a_{ij}]$ je matica veľkosti $m \times n$ a B je matica veľkosti $p \times q$. *Tenzorový súčin* $A \otimes B$ je matica veľkosti $(mp) \times (nq)$ definovaná ako:

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix}.$$

Príklad 2.14. Uvažujme dve matice:

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 5 \\ 6 & 7 \end{bmatrix}.$$

Tenzorový súčin $A \otimes B$ má veľkosť 4×4 a vypočíta sa tak, že každý prvok matice A sa nahradí príslušnou násobnou maticou B :

$$A \otimes B = \begin{bmatrix} 1 \cdot B & 2 \cdot B \\ 3 \cdot B & 4 \cdot B \end{bmatrix} = \begin{bmatrix} 0 & 5 & 0 & 10 \\ 6 & 7 & 12 & 14 \\ 0 & 15 & 0 & 20 \\ 18 & 21 & 24 & 28 \end{bmatrix}.$$

△

3 KONŠTRUKCIA HADAMARDOVÝCH MATÍC

Existuje niekoľko rodín konštrukčných algoritmov, ktoré sa líšia nielen matematickým pozadím, ale aj praktickým rozsahom použitia a generovanými triedami Hadamardových matíc. Medzi najznámejšie patria rekurzívne konštrukcie, ktoré vytvárajú Hadamardove matice vyšších rádov na základe menších maticových jadier, typickým príkladom je Sylvesterova rekurencia. Ďalšiu významnú skupinu tvoria konštrukcie založené na konečných poliach, predovšetkým tzv. Paleyho typy, ktoré využívajú vlastnosti kvadratických zvyškov a Legendreovho symbolu nad poľami $\mathbb{F}_q$. Tieto metódy umožňujú vytvárať Hadamardove matice so špecifickou symetriou a sú pevne prepojené s algebraickou štruktúrou príslušných polí [10].

Zložitejšie, ale veľmi univerzálne sú kombinatorické a návrhové metódy, ako napríklad Williamsonova konštrukcia, ktoré využívajú systémy binárnych matíc spĺňajúcich ortogonálne vzťahy. Tieto prístupy sú vhodné pre hľadanie Hadamardových matíc vo vyšších radoch a často vedú k tzv. kocyklickým alebo blokovým štruktúram. Osobitnú úlohu zohráva aj tenzorové rozšírenie, ktoré umožňuje efektívne kombinovať už známe matice a vytvárať tak nové, väčšie Hadamardove matice. Výhodou týchto metód je, že sú dobre implementovateľné a zachovávajú ortogonalitu [11].

Doteraz neexistuje univerzálny prístup vhodný pre všetky veľkosti matíc, preto sa vyvíjali rozličné techniky, ktoré dnes tvoria široké spektrum konštrukčných metód. Niektoré z nich sú teoreticky motivované, iné využívajú výpočtové postupy alebo heuristiky.

Táto práca sa zameriava na vybraných zástupcov hlavných konštrukčných rodín Hadamardových matíc, konkrétne na Sylvesterovu rekurzívnu konštrukciu (vrátane interpretácie cez tenzorový súčin), Paleyho konštrukcie typu I a II, Williamsonovu metódu, Turynove postupy a kocyklické metódy založené na grupových zobrazeniach.

Pri návrhu algoritmov, generovaní výpočtových ukážok, ladení Python kódu bol ako podporný nástroj využitý jazykový model ChatGPT 4o(OpenAI) [12]. Model bol použitý interaktívne, pričom výstupy boli ďalej upravované autorom. Vo všetkých vizualizáciách platí, že biele políčka zodpovedajú hodnotám $+1$, a čierne hodnotám -1 .

3.1 Sylvesterova rekurzívna konštrukcia

Jednou z najstarších a najjednoduchších metód konštrukcie Hadamardových matíc je Sylvesterova rekurzívna konštrukcia, ktorú v roku 1867 objavil James Joseph Sylvester. Táto metóda umožňuje generovanie Hadamardových matíc pre rozmery, ktoré sú mocninami dvojky.

Veta 3.1 (Sylvesterova rekurzia). Nech H je Hadamardova matica rádu n . Potom bloková matica

$$\begin{bmatrix} H & H \\ H & -H \end{bmatrix}$$

je Hadamardova matica rádu $2n$.

Táto konštrukcia zabezpečuje, že každá nová matica si zachováva ortogonálne vlastnosti Hadamardových matíc. Každý nový blok H_{2n} je zostavený z matice H_n , pričom v spodnej pravej časti sa negujú všetky prvky.

Príklad 3.1. Vygenerujme Hadamardove matice pre $n = 1$, $n = 2$ a $n = 4$ pomocou Sylvesterovej rekurzcie:

Pre $n = 1$:

$$H_1 = \begin{bmatrix} 1 \end{bmatrix}.$$

Pre $n = 2$:

$$H_2 = \begin{bmatrix} H_1 & H_1 \\ H_1 & -H_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Pre $n = 4$:

$$H_4 = \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}.$$

Týmto spôsobom možno postupne rekurzívne konštruovať Hadamardove matice ľubovoľného rádu $n = 2^k$.

Nasleduje kód, ktorý generuje Hadamardove matice podľa Sylvesterovej rekurzcie.

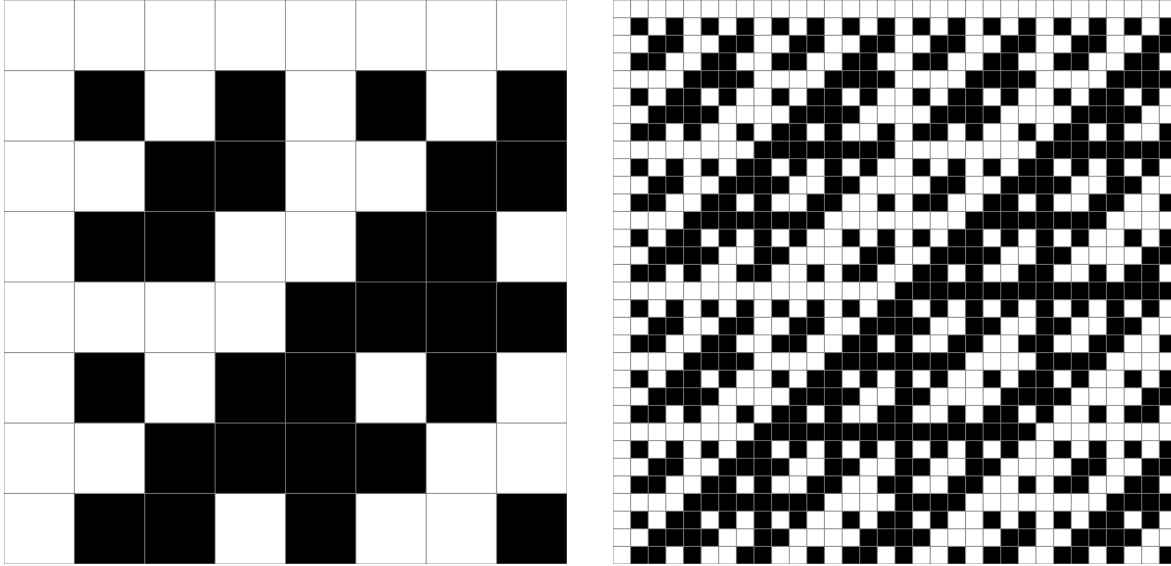
```

1 def sylvester_hadamard(n):
2     if n == 1:
3         return np.array([[1]])
4     H_n = sylvester_hadamard(n // 2)
5     return np.block([[H_n, H_n], [H_n, -H_n]])

```

Kód 3.1 Generovanie Hadamardových matíc pomocou Sylvesterovej konštrukcie

Tento algoritmus začína s H_1 a rekurzívne vytvára väčšie Hadamardove matice zdvojitím blokov a negáciou pravého dolného bloku.



Obrázok 3.1 Matice H_8 a H_{32} vygenerované Sylvesterovou konštrukciou.

Výhodou Sylvesterovej metódy je jej jednoduchá implementácia a rekurzívna štruktúra, čo umožňuje generovanie veľmi veľkých matíc. Metóda je však obmedzená len na rády tvaru $n = 2^k$, a preto je pri iných hodnotách n potrebné použiť alternatívne konštrukčné prístupy.

3.1.1 Tenzorový súčin ako všeobecný princíp

Tenzorový súčin predstavuje prirodzený spôsob, ako z menších Hadamardových matíc vytvárať väčšie. Ak H_m a H_n sú Hadamardove matice rádu m a n , potom ich tenzorový súčin $H = H_m \otimes H_n$ je Hadamardova matica rádu mn .

Základná vlastnosť Hadamardových matíc — ortogonalita — sa pritom zachováva:

Dôkaz.

$$\begin{aligned}
 HH^T &= (H_m \otimes H_n)(H_m \otimes H_n)^T \\
 &= (H_m \otimes H_n)(H_m^T \otimes H_n^T) \quad (\text{použitím } (A \otimes B)^T = A^T \otimes B^T) \\
 &= (H_m H_m^T) \otimes (H_n H_n^T) \quad (\text{použitím } (A \otimes B)(C \otimes D) = (AC) \otimes (BD)) \\
 &= (mI_m) \otimes (nI_n) \quad (\text{z definície } H_m, H_n) \\
 &= mn(I_m \otimes I_n) \quad (\text{vyňatím skalárov}) \\
 &= mnI_{mn}
 \end{aligned}$$

Týmto je dokázané, že tenzorový súčin dvoch Hadamardových matíc je opäť Hadamardova matica. $\square$

Príklad 3.2. Nech

$$H_2 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad H'_2 = \begin{bmatrix} 1 & -1 \\ -1 & -1 \end{bmatrix}.$$

Potom tenzorový súčin týchto dvoch matíc je:

$$H_4 = H_2 \otimes H'_2 = \begin{bmatrix} 1 \cdot H'_2 & 1 \cdot H'_2 \\ 1 \cdot H'_2 & -1 \cdot H'_2 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 1 & -1 \\ -1 & -1 & -1 & -1 \\ 1 & -1 & -1 & 1 \\ -1 & -1 & 1 & 1 \end{bmatrix}.$$

3.2 Paleyho konštrukcie

Dve rodiny Hadamardových matíc objavil Paley [3] pomocou kvadratických zvyškov v konečnom poli $\mathbb{F}_q$ rádu q , kde q je mocnina nepárneho prvočísla.

Definícia 3.1 (Jacobsthalova matica). Nech $\mathbb{F}_q$ je konečné pole s usporiadanými prvkami $g_0, \dots, g_{q-1}$. *Jacobsthalovu maticu* $Q \in \{\pm 1\}^{q \times q}$ definujeme pomocou Legendreovho symbolu ako:

$$Q_{ij} = \left(\frac{g_i - g_j}{q} \right).$$

Lema 3.1 (Symetria matice). Matica Q má tieto vlastnosti:

- je *symetrická*, ak $q \equiv 1 \pmod{4}$, teda $Q^\top = Q$,
- je *antisymetrická*, ak $q \equiv 3 \pmod{4}$, teda $Q^\top = -Q$.

Jacobsthalove matice sú využité v oboch Paleyho metódach. Ich prvky sú určené pomocou Legendreovho symbolu, ktorý odlišuje kvadratické zvyšky od nekvadratických v konečnom poli $\mathbb{F}_q$. Vzhľadom na aritmetické vlastnosti Legendreovho symbolu nad prvočíslom q matica Q vykazuje buď symetriu, alebo antisymetriu v závislosti od zvyšku $q \pmod{4}$. Táto vlastnosť umožňuje jednotný algoritmus generovania pre obe metódy, v ktorom štruktúra výslednej matice vzniká prirodzene bez potreby dodatočného rozlišovania.

```

1 def jacobsthal_matrix(q):
2     J = np.zeros((q, q), dtype=int)
3     for i in range(q):
4         for j in range(q):
5             if i != j:
6                 J[i, j] = legendre_symbol(i - j, q)
7     return J

```

Kód 3.2 Generovanie Jacobsthalovej matice v jazyku Python

Veta 3.2 (Paleyho konštrukcia typu I). Nech $q \equiv 3 \pmod{4}$ je mocnina nepárneho prvočísla. Nech Q je antisymetrická Jacobsthalova matica podľa vyššie uvedenej definície. Potom maticu

$$H = \begin{bmatrix} 1 & -\mathbf{1}^T \\ \mathbf{1} & Q + I_q \end{bmatrix}$$

zostrojíme tak, že k matici Q pripočítame jednotkovú maticu I_q , čím zabezpečíme, že všetky diagonálne prvky budú rovné 1. Vektory $\mathbf{1}$ a $\mathbf{1}^T$ predstavujú stĺpcový, resp. riadkový vektor samých jednotiek. Takto skonštruovaná matica H obsahuje iba prvky z množiny $\{\pm 1\}$ a je Hadamardova.

Príklad 3.3. Uvažujme Paleyho konštrukciu typu I pre $q = 7$, kde $q \equiv 3 \pmod{4}$. Pole $\mathbb{F}_7$ obsahuje prvky $\{0, 1, 2, 3, 4, 5, 6\}$. Kvadratické zvyšky modulo 7 sú:

$$1^2 = 1, \quad 2^2 = 4, \quad 3^2 = 2 \Rightarrow \{1, 2, 4\}.$$

Z toho vyplývajú hodnoty Legendreovho symbolu:

$$\left(\frac{0}{7}\right) = 0, \quad \left(\frac{1}{7}\right) = 1, \quad \left(\frac{2}{7}\right) = 1, \quad \left(\frac{3}{7}\right) = -1, \quad \left(\frac{4}{7}\right) = 1, \quad \left(\frac{5}{7}\right) = -1, \quad \left(\frac{6}{7}\right) = -1.$$

Na základe toho zostavíme antisymetrickú Jacobsthalovu maticu $Q \in \mathbb{R}^{7 \times 7}$, kde

$$Q_{ij} = \left(\frac{i-j}{7}\right), \quad \text{pričom } Q_{ii} = 0.$$

Pripočítaním jednotkovej matice I_7 k Q dostávame diagonálu s hodnotami 1.

Následne Hadamardovu maticu H_8 zostavíme ako blokovú maticu podľa vety 3.2:

$$H_8 = \begin{bmatrix} 1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & 1 & -1 & -1 & 1 & -1 & 1 & 1 \\ 1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \\ 1 & -1 & 1 & 1 & 1 & -1 & -1 & 1 \\ 1 & 1 & -1 & 1 & 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 & 1 & 1 & 1 & -1 \\ 1 & -1 & -1 & 1 & -1 & 1 & 1 & 1 \end{bmatrix}.$$

Táto matica je Hadamardova, teda spĺňa podmienku ortogonalít $H_8 H_8^\top = 8I$. △

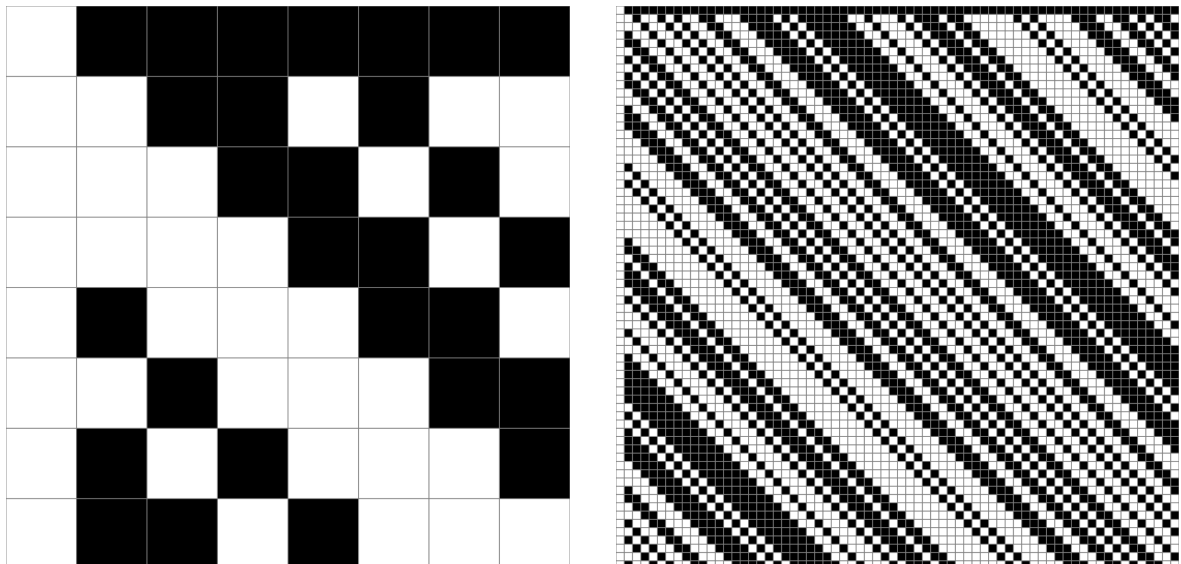
```
1 def paley_type_I(p):
2     if p % 4 != 3:
```

```

3         raise ValueError("Paley type I requires p == 3 mod 4"
4           )
5     Q = np.zeros((p, p), dtype=int)
6     for i in range(p):
7         for j in range(p):
8             if i != j:
9                 Q[i, j] = legendre_symbol(i - j, p)
10
11     Q += np.identity(p, dtype=int) # Pridáme jednotkovú
12     maticu, aby diagonála bola 1
13
14     H = np.ones((p + 1, p + 1), dtype=int)
15     H[0, 1:] = -1 # prvý riadok okrem H[0,0] bude -1
16     H[1:, 0] = 1  # prvý stlpec sú samé jednotky
17     H[1:, 1:] = Q # jadro je Q + I
18     return H

```

Kód 3.3 Paleyho konštrukcia typu I v jazyku Python

Obrázok 3.2 Matice H_8 a H_{68} vygenerované Paleyho konštrukciou 1. typu.

Paleyho konštrukcia typu II sa používa pre prípady, keď je q prvočíslo také, že $q \equiv 1 \pmod{4}$. Tento prístup umožňuje vytvoriť Hadamardove matice rádu $2(q + 1)$. Na rozdiel od typu I sa v tomto prípade Jacobsthalova matica používa ako súčasť pomocnej matice S , ku ktorej sa pridáva nulová diagonála a jednotkový stĺpec a riadok.

Veta 3.3 (Paleyho konštrukcia typu II). Nech $q \equiv 1 \pmod{4}$ je prvočíslo a nech Q je symetrická Jacobsthalova matica nad $\mathbb{F}_q$. Označme

$$S = \begin{bmatrix} 0 & \mathbf{1}^T \\ \mathbf{1} & Q \end{bmatrix},$$

kde $\mathbf{1} \in \mathbb{R}^q$ je stĺpcový vektor jednotiek a $Q \in \{\pm 1\}^{q \times q}$. Potom dostaneme Hadamardovu

maticu rádu $2(q+1)$, ktorá má tvar:

$$H = \begin{bmatrix} S + I_{q+1} & S - I_{q+1} \\ S - I_{q+1} & -S - I_{q+1} \end{bmatrix},$$

kde I_{q+1} je jednotková matica rádu $q+1$.

Príklad 3.4. Uvažujme Paleyho konštrukciu typu II pre $q = 5$, kde $q \equiv 1 \pmod{4}$. Pole $\mathbb{F}_5$ obsahuje prvky $\{0, 1, 2, 3, 4\}$. Kvadratické zvyšky modulo 5 sú všetky hodnoty $x^2 \pmod{5}$ pre $x \in \{1, 2, 3, 4\}$:

$$1^2 = 1, \quad 2^2 = 4, \quad 3^2 = 9 \equiv 4, \quad 4^2 = 16 \equiv 1 \pmod{5}.$$

Teda množina kvadratických zvyškov v $\mathbb{F}_5$ je $\{1, 4\}$. Z toho vyplývajú hodnoty Legendreovho symbolu:

$$\left(\frac{0}{5}\right) = 0, \quad \left(\frac{1}{5}\right) = 1, \quad \left(\frac{2}{5}\right) = -1, \quad \left(\frac{3}{5}\right) = -1, \quad \left(\frac{4}{5}\right) = 1.$$

Pomocou týchto hodnôt zostavíme symetrickú Jacobsthalovu maticu $Q \in \mathbb{R}^{5 \times 5}$, kde:

$$Q_{ij} = \left(\frac{i-j}{5}\right), \quad \text{pričom } Q_{ii} = 0.$$

Zostrojíme maticu $S \in \mathbb{R}^{6 \times 6}$ ako:

$$S = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & -1 & -1 & 1 \\ 1 & 1 & 0 & 1 & -1 & -1 \\ 1 & -1 & 1 & 0 & 1 & -1 \\ 1 & -1 & -1 & 1 & 0 & 1 \\ 1 & 1 & -1 & -1 & 1 & 0 \end{bmatrix}.$$

Hadamardova matica H_{12} sa potom zostaví ako:

$$H_{12} = \begin{bmatrix} S + I_6 & S - I_6 \\ S - I_6 & -S - I_6 \end{bmatrix}.$$

Výsledná matica je rádu 12 a obsahuje iba prvky ± 1 . Táto matica je Hadamardova, tzn. spĺňa podmienku ortogonalít $H_{12}H_{12}^\top = 12I$.

$$\begin{bmatrix}
 1 & 1 & 1 & 1 & 1 & 1 & -1 & 1 & 1 & 1 & 1 & 1 \\
 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 & 1 \\
 1 & 1 & 1 & 1 & -1 & -1 & 1 & 1 & -1 & 1 & -1 & -1 \\
 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\
 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 & 1 \\
 1 & 1 & -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \\
 -1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 \\
 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\
 1 & 1 & -1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 & 1 & 1 \\
 1 & -1 & 1 & -1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 \\
 1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & -1 & -1 \\
 1 & 1 & -1 & -1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 & -1
 \end{bmatrix}$$

△

```

1 def paley_type_II(q):
2     J = jacobsthal_matrix(q)
3     ones_row = np.ones((1, q), dtype=int)
4     ones_col = np.ones((q, 1), dtype=int)
5     S = np.block([
6         [np.array([[0]]), ones_row],
7         [ones_col, J]
8     ])
9     I = np.identity(q + 1, dtype=int)
10    H = np.block([
11        [S + I, S - I],
12        [S - I, -S - I]
13    ])
14    return H
15
16
17
18

```

Kód 3.4 Paleyho konštrukcia typu II v jazyku Python

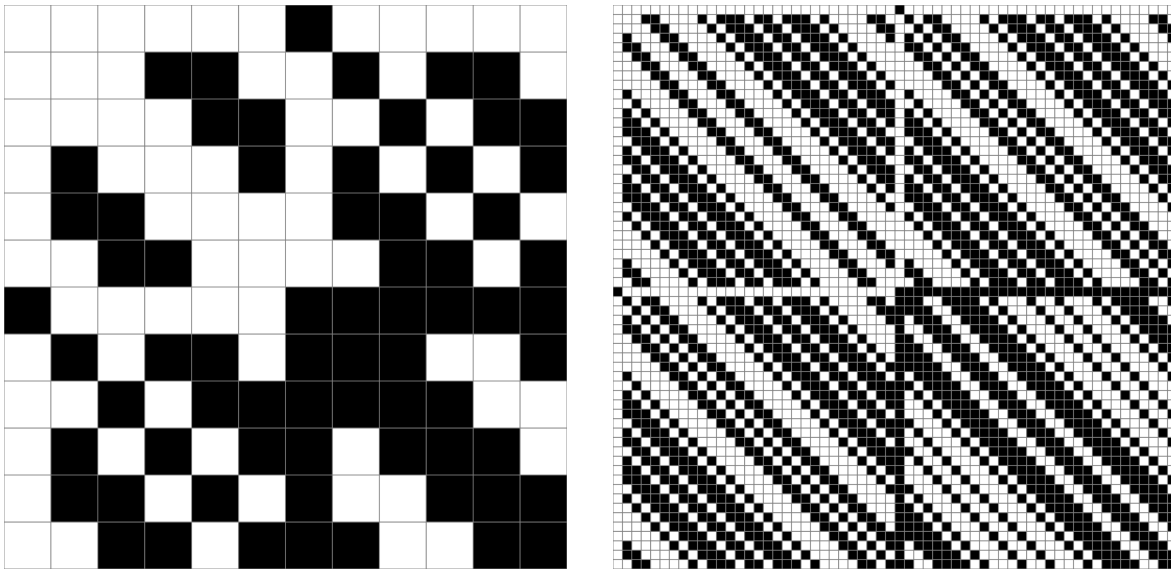
3.3 Williamsonova konštrukcia

Ďalšou z klasických metód konštrukcie Hadamardových matíc je tzv. *Williamsonova konštrukcia*, ktorú zaviedol J. Williamson v roku 1944. Táto metóda umožňuje vytvoriť Hadamardovu maticu rádu $4n$ pomocou štyroch binárnych symetrických matíc [13].

Definícia 3.2. Štvorica binárnych symetrických matíc $A, B, C, D \in \{\pm 1\}^{n \times n}$ sa nazýva *Williamsonova štvorica*, ak spĺňa nasledujúce podmienky:

1.

$$AA^T + BB^T + CC^T + DD^T = 4nI_n,$$



Obrázok 3.3 Matice H_{12} a H_{60} vygenerované Paleyho konštrukciou 2. typu.

kde I_n je jednotková matica rádu n .

2. pre každé rôzne $X, Y \in \{A, B, C, D\}$ platí:

$$XY^{\top} = YX^{\top}.$$

Pomocou takejto štvorice môžeme zostaviť Hadamardovu maticu H rádu $4n$ podľa nasledujúceho blokového usporiadania:

$$H = \begin{bmatrix} A & B & C & D \\ -B & A & -D & C \\ -C & D & A & -B \\ -D & -C & B & A \end{bmatrix}.$$

Lema 3.2. Ak matice A, B, C, D tvoria Williamsonovu štvoricu, potom matica H skonštruovaná vyššie je Hadamardova.

Pôvodne sa predpokladalo, že Williamsonove matice existujú pre každé celé číslo n , čo by poskytovalo univerzálnu cestu k dôkazu Hadamardovej domnienky, podľa ktorej existujú Hadamardove matice rádu $4n$ pre všetky $n \in \mathbb{N}$. Tento predpoklad bol známy ako *Williamsonova domnienka*. V roku 1993 však Dragomir Ž. Đoković [14] pomocou počítačového prehľadávania ukázal, že pre $n = 35$ neexistuje žiadna štvorica symetrických binárnych

matic, ktorá by spĺňala Williamsonovu podmienku, čím túto domnienku vyvrátil. Neskôr boli nájdené ďalšie rozmery, ktoré nespĺňali túto podmienku ($n = 47, 53, 59$) [15].

Príklad 3.5. Uvažujme $n = 2$ a nasledujúcu Williamsonovu štvoricu binárnych symetrických matíc:

$$A = \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}.$$

Tieto matice spĺňajú dôležité podmienky Williamsonovej konštrukcie:

1. Sú binárne, symetrické a ich súčet štvorcov je:

$$AA^T + BB^T + CC^T + DD^T = 8I_2,$$

teda presne $4nI_n$.

2. Každé dve matice z množiny $\{A, B, C, D\}$ spĺňajú komutačnú podmienku:

$$XY^T = YX^T \quad \text{pre všetky } X, Y \in \{A, B, C, D\}, X \neq Y.$$

Napríklad:

$$AB^T = \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} = BA^T,$$

Po dosadení konkrétnych matíc dostaneme:

$$H_8 = \left[\begin{array}{cc|cc|cc|cc} -1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 & -1 & 1 & -1 & 1 \\ \hline -1 & 1 & -1 & -1 & 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & -1 & 1 & -1 & -1 & -1 \\ \hline 1 & -1 & 1 & 1 & -1 & 1 & -1 & 1 \\ -1 & 1 & 1 & -1 & 1 & -1 & 1 & -1 \\ \hline 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 \\ -1 & 1 & -1 & -1 & 1 & -1 & -1 & 1 \end{array} \right].$$

Táto matica je Hadamardova, pretože spĺňa podmienku $H_8 H_8^T = 8I_8$.

Pri hľadaní Williamsonových štvoric je potrebné vygenerovať všetky možné binárne symetrické matice určitého rádu. Symetrická matica $M \in \pm 1^{n \times n}$ spĺňa podmienku $M = M^T$, takže je postačujúce zvoliť hodnoty iba v hlavnom trojuholníku (vrátane diagonály) a zvyšné hodnoty zrkadlovo doplniť.

Počet nezávislých prvkov v takejto matici je daný vzt'ahom:

$$t = \frac{n(n+1)}{2},$$

kde t predstavuje počet pozícií, ktoré určujú celú symetrickú maticu. Keďže každý prvok môže nadobúdať hodnotu $+1$ alebo -1 , celkový počet všetkých binárnych symetrických matíc rádu n je 2^t .

V nasledujúcom výpise je uvedená funkcia v jazyku Python, ktorá generuje všetky binárne symetrické matice daného rádu:

```

1 def generate_binary_symmetric_matrices(n):
2     total_entries = (n * (n + 1)) // 2
3     for values in itertools.product([-1, 1], repeat=
4         total_entries):
5         M = np.zeros((n, n), dtype=int)
6         idx = 0
7         for i in range(n):
8             for j in range(i, n):
9                 M[i][j] = values[idx]
10                M[j][i] = values[idx]
11                yield M
                idx += 1

```

Kód 3.5 Generovanie binárnych symetrických matíc

Pre každú štvoricu kandidátskych matíc musíme overiť, či spĺňa dve základné podmienky z definície:

```

1 def check_williamson_condition(A, B, C, D):
2     n = A.shape[0]
3     lhs = A @ A + B @ B + C @ C + D @ D
4     rhs = 4 * n * np.eye(n, dtype=int)
5     return np.array_equal(lhs, rhs)
6
7 def check_commutativity(A, B, C, D):
8     matrices = [A, B, C, D]
9     for i in range(4):
10        for j in range(i+1, 4):
11            X = matrices[i]
12            Y = matrices[j]
13            if not np.array_equal(X @ Y.T, Y @ X.T):
14                return False
15    return True

```

Kód 3.6 Podmienky na overenie Williamsonovej štvorice

Po splnení týchto podmienok môžeme pristúpiť k zostaveniu samotnej Hadamardovej matice. Nasledujúci kód prehľadáva všetky možné kombinácie binárnych symetrických matíc a vracia prvú platnú Hadamardovu maticu:

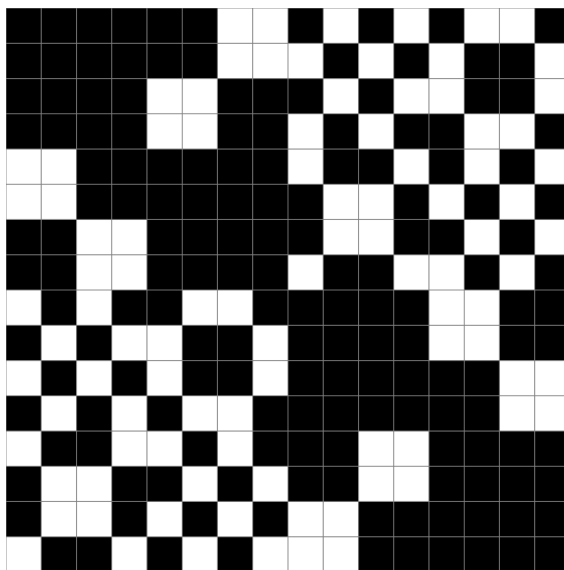
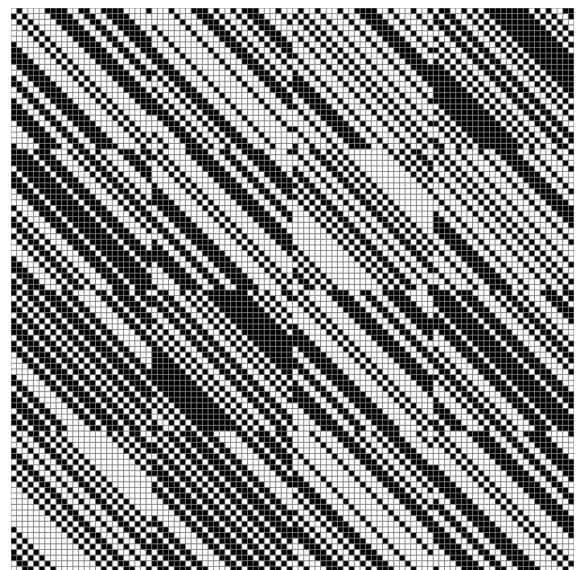
```

1 def build_williamson_hadamard(A, B, C, D):
2     top = np.hstack((A, B, C, D))
3     mid1 = np.hstack((-B, A, -D, C))
4     mid2 = np.hstack((-C, D, A, -B))
5     bot = np.hstack((-D, -C, B, A))
6     return np.vstack((top, mid1, mid2, bot))
7
8
9 def find_williamson_until_hadamard(n):
10    matrices = list(generate_binary_symmetric_matrices(n))
11    counter = 0
12
13    for A, B, C, D in itertools.product(matrices, repeat=4):
14        counter += 1
15        if check_williamson_condition(A, B, C, D) and
16            check_commutativity(A, B, C, D):
17            H = build_williamson_hadamard(A, B, C, D)
18            if is_hadamard_matrix(H):
19                return A, B, C, D, H, counter
20    return None, None, None, None, None, counter

```

Kód 3.7 Hľadanie Williamsonovej štvorice vedúcej k Hadamardovej matici

Tento prístup je síce neefektívny pre veľké hodnoty n , ale postačuje na ilustračné a výskumné účely pre malé rozmery (napr. $n \leq 4$).

(a) Hadamardova matica H_{16} hľadaná hrubou silou.(b) Hadamardova matica H_{100} prevzatá z databázy [16].

Obrázok 3.4 Hadamardove matice vygenerované Williamsonovou metódou.

Počet binárnych symetrických matíc $n \times n$ je $2^{n(n+1)/2}$. Pre $n = 4$ teda algoritmus prechádza $2^{10} = 1024$ možností pre každú z matíc A, B, C, D , čo predstavuje veľmi vysoký počet kombinácií (až 2^{40}). Preto je táto metóda vhodná len pre malé hodnoty n . Existujú optimalizované algoritmy, ktoré využívajú heuristiky alebo algebraické vlastnosti na efektívnejšie hľadanie Williamsonových štvoríc, ale ich implementácia presahuje rámec tejto práce.

3.4 Turynove metódy

Ďalšou významnou triedou konštrukcií Hadamardových matic sú tzv. *Turynove metódy*, pomenované po R. J. Turynovi [17], ktorý významne prispel k ich rozvoju v 70. rokoch 20. storočia. Tieto metódy sa často opierajú o existenciu špeciálnych sekvencií s dobrými autokorelačnými vlastnosťami.

Na rozdiel od iných konštrukcií, ktoré priamo pracujú s maticami, Turynove metódy začínajú so štyrmi binárnymi sekvenciami. Tieto sekvencie musia spĺňať špeciálne autokorelačné podmienky. Ak sa podarí nájsť takúto štvoricu dĺžky n , môžeme z nej skonštruovať Hadamardovu maticu veľkosti $4n$. Nasledujúca definícia formalizuje podmienky, ktoré takáto štvorica musí spĺňať.

Definícia 3.3. Štvorica sekvencií $A = (a_1, a_2, \dots, a_n)$, $B = (b_1, b_2, \dots, b_n)$, $C = (c_1, c_2, \dots, c_n)$, $D = (d_1, d_2, \dots, d_n)$ s prvkami z množiny $\{\pm 1\}$ sa nazýva *Turynova štvorica dĺžky n* , ak spĺňa nasledujúce podmienky pre všetky $k = 1, 2, \dots, n-1$:

$$\sum_{i=1}^{n-k} (a_i a_{i+k} + b_i b_{i+k} + c_i c_{i+k} + d_i d_{i+k}) = 0.$$

Navyše, ak n je párne, vyžaduje sa, aby $\sum_{i=1}^n a_i = 0$.

Takéto štyri binárne sekvencie A, B, C, D môžu byť použité v Turynovej konštrukcii Hadamardových matic. Ich prvky nadobúdajú hodnoty ± 1 a sú navrhnuté tak, aby ich vzájomné posuny boli vzájomne „neinterferujúce“ – formálne povedané, aby súčet ich posunutých skalárnych súčinov bol nulový pre každé $k = 1, 2, \dots, n-1$. Tieto podmienky zabezpečujú ortogonalitu blokov v matici, ktorá z nich bude vytvorená.

Overením, že konkrétna štvorica sekvencií túto podmienku spĺňa, možno rozhodnúť, či dané sekvencie vedú ku konštrukcii Hadamardovej matice. Nasledujúci príklad ilustruje, ako takúto kontrolu vykonať.

Príklad 3.6. Uvažujme $n = 3$. Skúsme overiť, či štvorica sekvencií

$$A = (+1, -1, +1), \quad B = (+1, +1, -1), \quad C = (+1, -1, -1), \quad D = (+1, +1, +1)$$

je Turynova štvorica. Vypočítajme sumy pre $k = 1$ a $k = 2$:

- pre $k = 1$:

$$\sum_{i=1}^2 (a_i a_{i+1} + b_i b_{i+1} + c_i c_{i+1} + d_i d_{i+1}) =$$

$$\begin{aligned}
&= (a_1a_2 + a_2a_3) + (b_1b_2 + b_2b_3) + (c_1c_2 + c_2c_3) + (d_1d_2 + d_2d_3) \\
&= (-1 - 1) + (+1 - 1) + (-1 + 1) + (+1 + 1) = -2 + 0 + 0 + 2 = 0
\end{aligned}$$

- pre $k = 2$:

$$\begin{aligned}
&\sum_{i=1}^1 (a_i a_{i+2} + b_i b_{i+2} + c_i c_{i+2} + d_i d_{i+2}) = \\
&= a_1 a_3 + b_1 b_3 + c_1 c_3 + d_1 d_3 \\
&= (+1 - 1 - 1 + 1) = 0
\end{aligned}$$

Sekvence teda splňajú podmienky definície pre všetky potrebné k .

△

Turynove metódy boli úspešné pri konštrukcii Hadamardových matíc pre mnohé rády. Existencia Turynových sekvencií je však otvorený problém pre niektoré dĺžky n . Pôvodná Turynova domnienka tvrdila, že Turynove sekvencie dĺžky n existujú pre všetky $n \in \mathbb{N}$, no táto domnienka bola neskôr vyvrátená [18]. Napriek tomu sa Turynove konštrukcie ukázali ako veľmi plodné pri hľadaní Hadamardových matíc.

Pri konštrukcii Hadamardových matíc založených na Turynových metódach sa využívajú rôzne typy špeciálnych sekvencií s prvkami ± 1 alebo 0 , ktoré majú štruktúru zabezpečujúcu ortogonálne vlastnosti potrebné pre vznik Hadamardovej matice. Nasledujúci prehľad popisuje hlavné typy sekvencií využívaných v rámci Turynových metód.

1. Turynove sekvencie typu TT(n)

Ide o štvoricu binárnych sekvencií, kde tri majú dĺžku n a jedna má dĺžku $n - 1$. Tieto sekvencie boli navrhnuté tak, aby ich kombinácia umožňovala vytvoriť Hadamardovu maticu veľkosti $4n$. Ich vnútorná štruktúra spočíva v tom, že súčet súčinov prvkov v jednotlivých sekvenciách pri ľubovoľnom nenulovom posunutí je nulový, čo zabezpečuje, že vzájomné korelácie medzi sekvenciami sa vyrušia a výsledná matica bude ortogonálna.

2. Turynove sekvencie (TS)

Pod týmto označením sa v literatúre rozumejú sekvencie podobné TT(n), často s inou konfiguráciou dĺžok a dodatočnými symetrickými vlastnosťami. Používajú sa v špeciálnych konštrukciách, kde sa kladie dôraz na rovnováhu a zrkadlovú štruktúru.

3. T-sekvencie

Tieto sekvencie obsahujú okrem hodnôt ± 1 aj nuly a majú spoločnú dĺžku. Ich špecifická vlastnosť spočíva v tom, že v každej pozícii len jeden zo štyroch prvkov je nenulový.

4. Bázové sekvencie (Base Sequences)

Ide o štyri binárne sekvencie, z ktorých dve majú typicky o jeden prvok viac než ostatné. Tieto sekvencie predstavujú základný stavebný kameň pre ďalšie typy (ako sú T-sekvencie alebo TT(n)), a hrajú významnú úlohu v systematickom vytváraní väčších matíc.

Typ sekvencie	Dĺžky sekvencií	Hodnoty	Použitie
TT(n)	$n, n, n, n-1$	± 1	Konštrukcia Hadamardovej matice
TS	$n, n, n-1, n-1$	± 1	Varianta TT(n), symetria
T-sekvencie	n, n, n, n	$0, \pm 1$	Goethals–Seidelova konštrukcia
Bázové sekvencie	$n+1, n+1, n, n$	± 1	Základ pre ďalšie konštrukcie

Tabulka 3.1 Základné typy sekvencií používané v Turynových metódach

Hľadanie Turynových sekvencií a následná konštrukcia Hadamardových matíc často vyžaduje systematické prehľadávanie alebo využitie špeciálnych algebraických techník. Na rozdiel od Williamsonovej konštrukcie, ktorá priamo pracuje s maticami, Turynove metódy sa zameriavajú na vlastnosti sekvencií.

Jedna zo známych Turynových metód je tzv. *Goethals–Seidelova konštrukcia*, ktorá využíva cirkulačné matice, ktorých prvé riadky sú dané sekvenciami A, B, C, D . Nech teda A, B, C, D označujú cirkulačné matice rádu n generované zodpovedajúcimi binárnymi sekvenciami (každá sekvencia tvorí prvý riadok danej matice). Potom Hadamardova matica rádu $4n$ môže byť skonštruovaná nasledovne:

$$H = \begin{bmatrix} A & -B^R & -C^R & -D^R \\ B^R & A & -D^R & C^R \\ C^R & D^R & A & -B^R \\ D^R & -C^R & B^R & A \end{bmatrix},$$

kde X^R označuje maticu získanú z X obrátením poradia riadkov.

Príklad 3.7. Uvažujme $n = 2$ a štyri binárne sekvencie

$$A = (-1, +1), \quad B = (-1, -1), \quad C = (-1, -1), \quad D = (-1, +1).$$

Z týchto sekvencií zostrojme cirkulačné matice rádu 2:

$$A = \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}, \quad B = \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix}, \quad C = \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix}, \quad D = \begin{bmatrix} -1 & +1 \\ +1 & -1 \end{bmatrix}.$$

Obrátením riadkov týchto matíc dostaneme:

$$B^R = \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix}, \quad C^R = \begin{bmatrix} -1 & -1 \\ -1 & -1 \end{bmatrix}, \quad D^R = \begin{bmatrix} +1 & -1 \\ -1 & +1 \end{bmatrix}.$$

Podľa Goethals–Seidelovej konštrukcie zostrojíme Hadamardovu maticu rádu $4n = 8$:

$$H = \begin{bmatrix} A & -B^R & -C^R & -D^R \\ B^R & A & -D^R & C^R \\ C^R & D^R & A & -B^R \\ D^R & -C^R & B^R & A \end{bmatrix} = \begin{bmatrix} -1 & 1 & 1 & 1 & -1 & 1 & 1 & -1 \\ 1 & -1 & 1 & 1 & 1 & -1 & 1 & -1 \\ -1 & -1 & -1 & 1 & -1 & 1 & -1 & -1 \\ -1 & -1 & 1 & -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & -1 & 1 & 1 \\ 1 & -1 & 1 & 1 & -1 & -1 & -1 & 1 \\ -1 & 1 & 1 & 1 & -1 & -1 & 1 & -1 \end{bmatrix}.$$

Overením môžeme vidieť, že H je Hadamardova matica, pretože platí $HH^T = 8I_8$. $\triangle$

Nasleduje implementácia *Goethals–Seidelovej konštrukcie* pri generovaní Hadamardovej matice hrubou silou. Použijeme štvoricu binárnych T-sekvencií rovnakej dĺžky n , z ktorých vytvoríme cirkulačné matice. Tie následne skombinujeme do štandardnej blokovej štruktúry $4n \times 4n$, pričom niektoré bloky budú negované a/alebo zrkadlovo otočené.

Nasledujúca funkcia overuje, či štyri binárne sekvencie dĺžky n spĺňajú podmienky Tury-novej štvorice. Kontroluje sa ortogonalita cez súčet korelácií pre všetky nenulové posuny a v prípade párneho n aj nulový súčet prvej sekvencie A .

```
1 def check_turyn_condition(A, B, C, D):
2     n = len(A)
3
```

```

4     if n % 2 == 0 and sum(A) != 0:
5         return False
6
7     for k in range(1, n):
8         s = 0
9         for i in range(n - k):
10            s += A[i] * A[i + k] + B[i] * B[i + k] + C[i] * C
11                [i + k] + D[i] * D[i + k]
12            if s != 0:
13                return False
14
15     return True

```

Kód 3.8 Overenie Turynovej štvorice

Nasledujúca funkcia vytvára Hadamardovu maticu rádu $4n$ z danej štvorice binárnych sekvencií dĺžky n . Každá sekvencia je skonvertovaná na cirkulačnú maticu a tieto sa následne skombinujú do blokovej štruktúry podľa klasickej Turynovej schémy.

```

1 def build_hadamard_from_turyn(A_seq, B_seq, C_seq, D_seq):
2     A = circulant(A_seq)
3     B = circulant(B_seq)
4     C = circulant(C_seq)
5     D = circulant(D_seq)
6
7     BR = np.flip(B, axis=1)
8     CR = np.flip(C, axis=1)
9     DR = np.flip(D, axis=1)
10
11     top_row = np.hstack((A, -BR, -CR, -DR))
12     mid1_row = np.hstack((BR, A, -DR, CR))
13     mid2_row = np.hstack((CR, DR, A, -BR))
14     bottom_row = np.hstack((DR, -CR, BR, A))
15
16     return np.vstack((top_row, mid1_row, mid2_row, bottom_row
17                        ))

```

Kód 3.9 Konštrukcia Hadamardovej matice z Turynovej štvorice

Nasledujúci kód využíva *T-sekvencie* s dĺžkou $n = 5$ a hrubou silou hľadá štvoricu binárnych sekvencií spĺňajúcich ortogonálnu podmienku vedúcu ku konštrukcii Hadamardovej matice.

```

1 n = 5
2 found = False
3 for A_seq in product([-1, 1], repeat=n):
4     if n % 2 == 0 and sum(A_seq) != 0:
5         continue
6     for B_seq in product([-1, 1], repeat=n):
7         for C_seq in product([-1, 1], repeat=n):
8             for D_seq in product([-1, 1], repeat=n):
9                 if check_turyn_condition(A_seq, B_seq, C_seq,
10                    D_seq):
11                     H = build_hadamard_from_turyn(A_seq,
12                        B_seq, C_seq, D_seq)
13                     if is_hadamard_matrix(H):
14                         print("Found a Turyn quadruple that
15                             generates a Hadamard matrix:")
16                         print(f"A = {A_seq}")

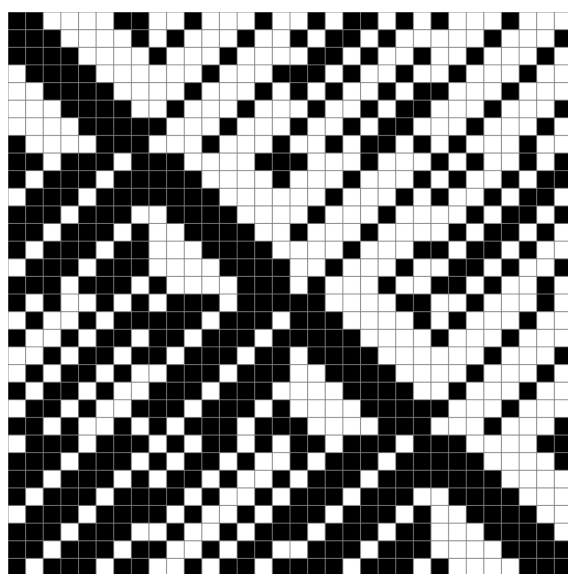
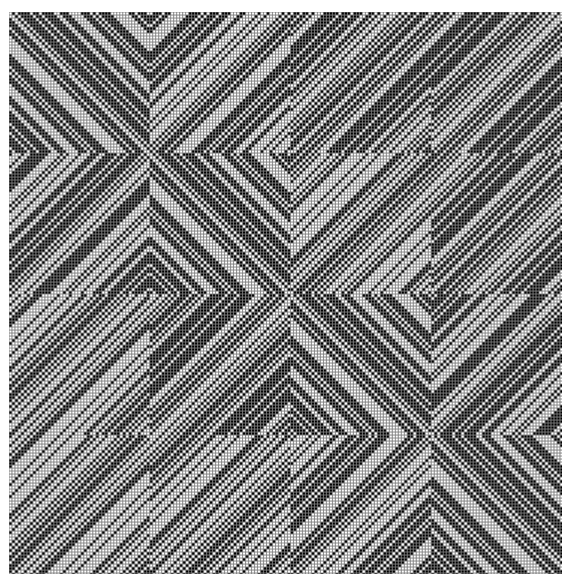
```

```

15         print(f"B = {B_seq}")
16         print(f"C = {C_seq}")
17         print(f"D = {D_seq}")
18         print("Hadamard matrix H:")
19         print(H)
20         plot_hadamard(H, f"turyn_n{n}")
21         found = True
22         break
23     if found: break
24 if found: break
25

```

Kód 3.10 Hľadanie Hadamardovej matice hrubou silou

(a) Hadamardova matica H_{32} hľadaná hrubou silou.(b) Hadamardova matica H_{188} prevzatá z databázy [16].

Obrázok 3.5 Hadamardove matice skonštruované Turynovými metódami.

Turynove metódy poskytujú silný nástroj na konštrukciu Hadamardových matíc, hoci existencia Turynových sekvencií nie je zaručená pre všetky dĺžky. Výskum v tejto oblasti pokračuje a nové konštrukcie založené na Turynových myšlienkach sa neustále objavujú.

3.5 Kocyklické metódy

Kocyklické konštrukcie [11] predstavujú v súčasnosti jednu z najúspešnejších a najvšeobecnejších metód generovania Hadamardových matíc, predovšetkým pre rozmery, ktoré nie sú pokryté klasickými algoritmi ako Sylvesterova alebo Paleyho konštrukcia. Výhodou tejto metódy je možnosť systematického prehľadávania pomocou výpočtových prostriedkov a možnosť overovania existencie Hadamardových matíc aj pre zložitejšie prípady [10, 19].

Základom kocyklickej konštrukcie je špeciálna binárna funkcia nazývaná *kocyklus*, ktorá

sa definuje nad konečnou grupou G . Z nej sa zostaví matica H tak, že každý prvok H_{ij} závisí iba od dvojice prvkov $g_i, g_j \in G$. Táto matica má hodnoty ± 1 a v prípade, že je ortogonálna, t. j. spĺňa $HH^\top = nI_n$, ide o Hadamardovu maticu.

Definícia 3.4. Nech G je konečná grupa rádu n . Zobrazenie $\psi : G \times G \rightarrow \{\pm 1\}$ sa nazýva *kocyklus*, ak platí *kocyklická podmienka*:

$$\psi(x, y)\psi(xy, z) = \psi(x, yz)\psi(y, z), \quad \forall x, y, z \in G.$$

Základnou myšlienkou kocyklickej konštrukcie Hadamardových matíc je hľadanie vhodného kocykla $\psi : G \times G \rightarrow \{\pm 1\}$ nad pevne zvolenou konečnou grupou G , pričom sa následne zostaví matica $H = [\psi(x, y)]_{x, y \in G}$ a overí sa, či je ortogonálna. Kocyklus musí spĺňať kocyklickú podmienku, ktorá zabezpečuje algebraickú konzistenciu. Úspešnosť konštrukcie často závisí nielen od samotného tvaru funkcie ψ , ale aj od výberu grupy G , nad ktorou je definovaná. V nasledujúcich príkladoch je použitý rovnaký typ kocyklu, no nad rôznymi grupami – raz nad $(\mathbb{Z}_4, +)$, kde konštrukcia zlyhá, a následne nad $(\mathbb{Z}_2 \times \mathbb{Z}_2, +)$, kde vedie k platnej Hadamardovej matici.

Príklad 3.8 (Kocyklická matica nad grupou $(\mathbb{Z}_4, +)$). Uvažujme cyklickú grupu $G = \mathbb{Z}_4 = \{0, 1, 2, 3\}$. Definujme funkciu $\psi : G \times G \rightarrow \{\pm 1\}$ nasledovne:

$$\psi(x, y) = (-1)^{xy}.$$

Tento zápis znamená, že $\psi(x, y) = +1$ práve vtedy, keď je celočíselný súčin xy párny, inak $\psi(x, y) = -1$.

Preto výpočty pre jednotlivé hodnoty $x \in \mathbb{Z}_4 = \{0, 1, 2, 3\}$ vyzerajú nasledovne:

- Pre $x = 0$: $xy = 0 \cdot y = 0$ pre všetky $y \Rightarrow \psi(0, y) = +1$.
- Pre $x = 1$: $xy = y \Rightarrow \psi(1, y) = (-1)^y$.
- Pre $x = 2$: $xy = 2y$, čo je vždy párne $\Rightarrow \psi(2, y) = +1$.
- Pre $x = 3$: $xy = 3y$. Čísla $3 \cdot 0 = 0, 3 \cdot 1 = 3, 3 \cdot 2 = 6, 3 \cdot 3 = 9$ majú paritu rovnakú ako $y \Rightarrow \psi(3, y) = (-1)^y$.

Zostavíme teda hodnoty matice $H = [\psi(x, y)]_{x, y \in \mathbb{Z}_4}$:

$$H = \begin{bmatrix} \psi(0,0) & \psi(0,1) & \psi(0,2) & \psi(0,3) \\ \psi(1,0) & \psi(1,1) & \psi(1,2) & \psi(1,3) \\ \psi(2,0) & \psi(2,1) & \psi(2,2) & \psi(2,3) \\ \psi(3,0) & \psi(3,1) & \psi(3,2) & \psi(3,3) \end{bmatrix} = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \end{bmatrix}$$

Skontrolujme ortogonalitu výpočtom $HH^\top$:

$$HH^\top = \begin{bmatrix} 4 & 0 & 4 & 0 \\ 0 & 4 & 0 & 4 \\ 4 & 0 & 4 & 0 \\ 0 & 4 & 0 & 4 \end{bmatrix} \neq 4I_4$$

Táto matica teda nie je Hadamardova, pretože nie je ortogonálna (napr. riadky 0 a 2 sú identické, rovnako ako riadky 1 a 3). $\triangle$

Príklad 3.9 (Kocyklická Hadamardova matica nad grupou $(\mathbb{Z}_2 \times \mathbb{Z}_2, +)$). V tomto príklade použijeme rovnaký tvar kocyklu $\psi(x,y) = (-1)^{x \cdot y}$ ako v predchádzajúcom príklade, avšak nad inou grupou – konkrétne nad $(\mathbb{Z}_2 \times \mathbb{Z}_2, +)$.

Uvažujme grupu $G = (\mathbb{Z}_2 \times \mathbb{Z}_2, +) = \{(0,0), (0,1), (1,0), (1,1)\}$. Prvky tejto grupy sú usporiadané dvojice (x_1, x_2) , kde $x_i \in \{0, 1\}$. Operácia $+$ je definovaná ako sčítanie po zložkách v $\mathbb{Z}_2$:

$$x + y = (x_1 + y_1, x_2 + y_2).$$

Definujme kocykklus:

$$\psi(x,y) = (-1)^{x \cdot y},$$

kde $x \cdot y = x_1 y_1 + x_2 y_2$ je skalárny súčin v $\mathbb{Z}_2^2$.

Hodnoty skalárneho súčinu $x \cdot y$ v $\mathbb{Z}_2$ sú uvedené v nasledujúcej tabuľke:

$x \backslash y$	(0,0)	(0,1)	(1,0)	(1,1)
(0,0)	0	0	0	0
(0,1)	0	1	0	1
(1,0)	0	0	1	1
(1,1)	0	1	1	0

Tým získame maticu:

$$H = [\psi(x,y)]_{x,y \in G} = \begin{bmatrix} +1 & +1 & +1 & +1 \\ +1 & -1 & +1 & -1 \\ +1 & +1 & -1 & -1 \\ +1 & -1 & -1 & +1 \end{bmatrix}$$

Vypočítajme súčin $HH^\top$:

$$HH^\top = \begin{bmatrix} 4 & 0 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 0 & 0 & 4 & 0 \\ 0 & 0 & 0 & 4 \end{bmatrix} = 4I_4$$

V tomto prípade vedie rovnaký kocyklus k platnej Hadamardovej matici H , keďže štruktúra grupy umožňuje ortogonalitu. $\triangle$

Prístup uvedený v Príklade 3.9 možno zobecniť na vyššie rády, čím vzniknú tzv. *Walsh–Hadamardove matice*. Z algebraického hľadiska ide o prípad kocyklickej konštrukcie, kde konečná grupa je $G = (\mathbb{Z}_2^k, +)$ a kocyklus ψ je definovaný ako:

$$\psi(x, y) = (-1)^{x \cdot y}$$

kde $x, y \in \mathbb{Z}_2^k$ a skalárny súčin $x \cdot y = \sum_{i=1}^k x_i y_i$ sa počíta v aritmetike poľa $\mathbb{Z}_2$.

Výsledná matica $H = [\psi(x, y)]$ má rozmery $2^k \times 2^k$ a spĺňa ortogonálnu podmienku $HH^\top = 2^k I_{2^k}$, takže ide o Hadamardovu maticu.

Takto vygenerovaná matica je *identická* s maticou získanou pomocou klasickej Sylvestrovej rekurzívnej konštrukcie. Obe metódy vedú na tú istú maticu – rozdiel spočíva v prístupe: Sylvestrova metóda ju buduje rekurzívne, zatiaľ čo kocyklická konštrukcia ju definuje explicitne pomocou skalárneho súčinu v $\mathbb{Z}_2^k$.

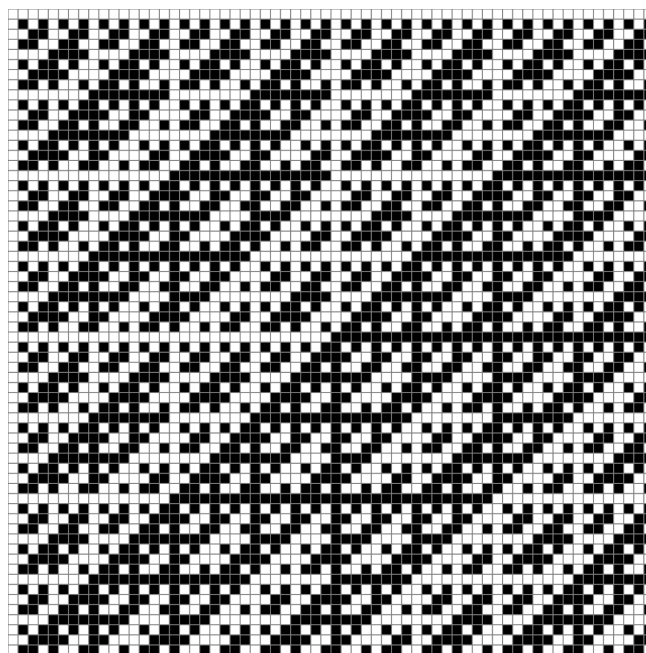
Nasledujúca implementácia generuje Walsh–Hadamardovu maticu pomocou skalárneho súčinu nad $\mathbb{Z}_2^k$, teda kocyklickou metódou pre grupu $G = \mathbb{Z}_2^k$:

```

1 def scalar_product(x, y):
2     return sum([a * b for a, b in zip(x, y)]) % 2
3
4 def generate_hadamard_cocycle(k):
5     G = list(product([0, 1], repeat=k))
6     n = len(G)
7     H = np.zeros((n, n), dtype=int)
8
9     for i, x in enumerate(G):
10        for j, y in enumerate(G):
11            dot = scalar_product(x, y)
12            H[i, j] = (-1)**dot
13
14     return H

```

Kód 3.11 Generovanie Walsh–Hadamardovej matice pomocou kocyklu



Obrázok 3.6 Walsh–Hadamardova matica H_{64} vytvorená pomocou kocyklu nad $\mathbb{Z}_2^6$.

Všeobecný algoritmus pre generovanie kocyklických Hadamardových matíc

Proces generovania Hadamardovej matice pomocou kocyklického prístupu možno zhrnúť do nasledujúcich krokov:

1. **Výber konečnej grupy G rádu n :** Voľba grupy G je kľúčovým krokom, ktorý ovplyvňuje štruktúru a typy Hadamardových matíc, ktoré možno nájsť. Často sa používajú abelovské grupy, ako napríklad cyklické grupy $\mathbb{Z}_n$, grupy typu $\mathbb{Z}_2^k$, alebo ich priame súčiny. Avšak kocyklické metódy nie sú obmedzené len na abelovské grupy; aj neabelovské grupy môžu viesť k novým Hadamardovým maticiam. Rád grupy n musí byť taký, aby Hadamardova matica rádu n mohla existovať (t. j. $n = 1$, $n = 2$ alebo n deliteľné štyrmi).
2. **Generovanie možných funkcií $\psi : G \times G \rightarrow \{\pm 1\}$:** Pri návrhu funkcie $\psi : G \times G \rightarrow \{\pm 1\}$ sa zvyčajne neprezerajú všetky možné kombinácie, keďže ich počet rastie exponenciálne. Namiesto toho sa používajú cielenejšie stratégie. Jednou z možností je navrhnúť algebraický predpis pre pomocnú funkciu $f(x, y)$ a potom definovať $\psi(x, y)$ ako $(-1)^{f(x, y)}$. Takéto funkcie môžu byť napríklad súčiny jednotlivých súradníc alebo ich jednoduché kombinácie. Iným prístupom je odvodenie funkcie ψ z inej jednoduchšej funkcie, pričom sa využívajú vzťahy medzi hodnotami pre jednotlivé prvky

grupy. Niekedy sa hľadanie obmedzí len na tie funkcie, ktoré vykazujú určitú pravidelnosť alebo symetriu, pretože tie majú väčšiu šancu viesť k ortogonálnej matici. Pre väčšie grupy sa používajú sofistikované algoritmy, ktoré kombinujú algebraické obmedzenia s heuristickým prehľadávaním.

3. Overenie podmienky kocykla

Funkcia ψ sa považuje za kocyklus, ak pre všetky $x, y, z \in G$ platí:

$$\psi(x, y)\psi(xy, z) = \psi(x, yz)\psi(y, z).$$

4. Zostavenie matice na základe kocykla

Matica $H = [\psi(x, y)]_{x, y \in G}$ sa vytvorí tak, že riadky a stĺpce zodpovedajú prvkom grupy G . Takto zostavená matica sa ďalej analyzuje ako kandidát na Hadamardovu maticu.

5. Overenie ortogonalít matice

Matica H sa považuje za Hadamardovu, ak spĺňa podmienku ortogonalít:

$$HH^T = nI_n,$$

t. j. ak sú jej riadky navzájom ortogonálne a všetky prvky patria do množiny $\{\pm 1\}$.

Nasledujúci algoritmus ilustruje vyčerpávajúci prístup k hľadaniu kocyklikých Hadamardových matíc pomocou enumerácie všetkých možných binárnych kocyklikých zobrazení $\psi : G \times G \rightarrow \{\pm 1\}$, kde $G = \mathbb{Z}_2 \times \mathbb{Z}_2$. Tento prístup je realizovateľný len pre veľmi malé grupy, pretože počet všetkých možných zobrazení rastie exponenciálne s veľkosťou grupy (2^{n^2} pre $n = |G|$).

```

1 import numpy as np
2 from itertools import product
3 from utils import is_hadamard_matrix, plot_hadamard
4
5 # Define the group Z2 x Z2 = {(0,0), (0,1), (1,0), (1,1)}
6 G = list(product([0, 1], repeat=2))
7 n = len(G)
8
9 # G -> {-1, +1}
10 def generate_all_maps(G):
11     domain = list(product(G, G)) # G x G
12     for values in product([-1, 1], repeat=len(domain)):
13         psi = dict(zip(domain, values))
14         yield psi
15
16 # Check the cocycle condition
17 def is_cocycle(psi, G):
18     for x in G:
19         for y in G:
20             for z in G:
21                 xy = ((x[0] ^ y[0]), (x[1] ^ y[1])) # group
22                 operation: XOR
23                 yz = ((y[0] ^ z[0]), (y[1] ^ z[1]))

```

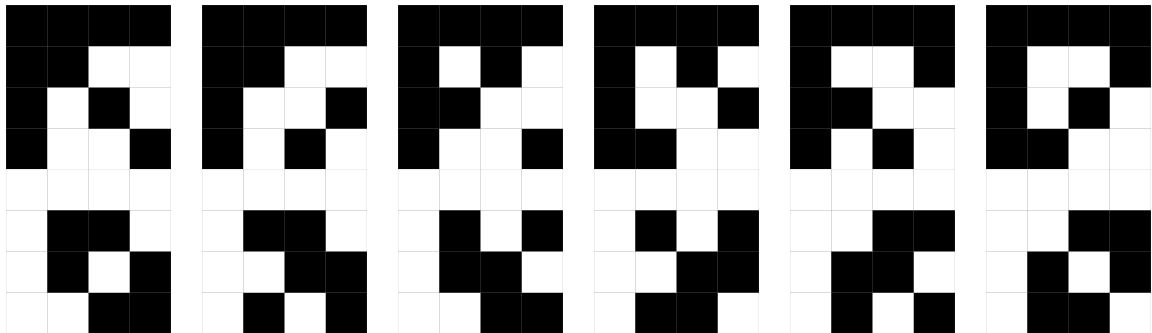
```

23         xyz = ((x[0] ^ y[0] ^ z[0]), (x[1] ^ y[1] ^ z
24             [1]))
25         lhs = psi[(x, y)] * psi[(xy, z)]
26         rhs = psi[(x, yz)] * psi[(y, z)]
27         if lhs != rhs:
28             return False
29     return True
30 # Build matrix H = [psi(x, y)]
31 def build_matrix(psi, G):
32     H = np.zeros((n, n), dtype=int)
33     for i, x in enumerate(G):
34         for j, y in enumerate(G):
35             H[i, j] = psi[(x, y)]
36     return H
37
38 # Main loop
39 count = 0
40 for psi in generate_all_maps(G):
41     if is_cocycle(psi, G):
42         H = build_matrix(psi, G)
43         if is_hadamard_matrix(H):
44             plot_hadamard(H, filename=f"cocycle_bruteforce{
45                 count+1}.png")
46             count += 1
47 print(f"\nTotal found: {count}")

```

Kód 3.12 Generovanie kocyklických Hadamardových matic pre $G = \mathbb{Z}_2 \times \mathbb{Z}_2$ hrubou silou.

Tento kód našiel všetky kocyklické Hadamardove matice nad grupou $\mathbb{Z}_2 \times \mathbb{Z}_2$. Celkový počet prehľadaných zobrazení bol $2^{16} = 65\,536$, čo je zvládnuteľné pomocou bežného počítača. Získané matice boli automaticky uložené ako obrázky, čím je možné vizuálne porovnať ich štruktúru.



Obrázok 3.7 Kocyklické Hadamardove matice rádu $n = 4$ generované hrubou silou nad grupou $\mathbb{Z}_2 \times \mathbb{Z}_2$.

Pre väčšie grupy (napr. $\mathbb{Z}_2^3$) však počet možných zobrazení rastie na 2^{64} , čo je výpočtovo neúnosné. Preto sa pri hľadaní Hadamardových matic pre vyššie rády využívajú algebraické obmedzenia, heuristiky alebo špeciálne konštrukčné schémy namiesto hrubého prehľadávania celého priestoru.

Kocyklové Hadamardove matice dnes predstavujú najvšeobecnejší a najvýkonnejší známy matematický aparát pre konštrukciu ortogonálnych ± 1 -matic. Na rozdiel od klasických kon-

štrukcií, ktoré sú často obmedzené na určité triedy rozmerov alebo algebraické podmienky (napr. mocniny dvojky či prvočísla určitého typu), kocyklové prístupy umožňujú generovať Hadamardove matice pre široké spektrum rádov, vrátane tých, pre ktoré doteraz neexistovali známe explicitné konštrukcie [19].

Okrem čisto teoretickej hodnoty majú kocyklové Hadamardove matice praktické využitie v rôznych oblastiach, ako sú digitálna signalizácia, radarové kódy, experimentálny dizajn, chybovo odolné kódy, kvantové algoritmy, teória grafov či spektroskopia. V databázach známych Hadamardových matíc možno nájsť stovky nových matíc získaných práve pomocou kocyklických metód, pričom mnohé z nich sa nepodarilo odvodiť inými spôsobmi. Samotná úloha generovania kocykla je algoritmicky zvládnuteľná a dobre paralelizovateľná. Práve na tento účel vznikli nástroje ako *Coca* [20], ktoré umožňujú automatizované vyhľadávanie a klasifikáciu kocyklických Hadamardových matíc.

4 SÚČASNÝ STAV VÝSKUMU HADAMARDOVEJ DOMNIENKY

Napriek širokej škále známych konštrukčných metód nie je zatiaľ známa univerzálna metóda, ktorá by dokázala generovať Hadamardove matice pre všetky násobky štyroch, t.j. pre všetky $n = 4k$. Táto skutočnosť je jedným z hlavných dôvodov, prečo Hadamardova domnienka zostáva dodnes nevyriešená. Táto kapitola primárne čerpá zo zdrojov [11], [10] a [4].

Rôzne konštrukčné metódy majú svoje prirodzené limity, ktoré bránia ich uplatneniu pre všetky hodnoty k . Medzi najdôležitejšie obmedzenia patria:

- **Paleyho metódy** sú závislé od existencie konečných polí $\mathbb{F}_q$ s $q \equiv 3 \pmod{4}$ alebo $q \equiv 1 \pmod{4}$ a neposkytujú pokrytie všetkých k .
- **Williamsonove konštrukcie** neexistujú pre niektoré konkrétne hodnoty k , napríklad $k = 35, 47, 53, 59$.
- **Turynove metódy** sú limitované existenciou špeciálnych postupností (napr. Golayových, T-sekvencií), ktoré neexistujú pre všetky rozmery.

Pri výpočtovej konštrukcii Hadamardových matíc narážame na tzv. *kombinatorickú explóziu*. Počet ± 1 matíc veľkosti $n \times n$ je 2^{n^2} , čo robí úplné prehľadávanie neuskutočniteľným už pri stredne veľkých n . Navyše, ani algebraické metódy nie sú schopné generovať riešenia pre všetky prípady, keďže:

- klasifikácia všetkých neekvivalentných Hadamardových matíc je známa len pre $n \leq 32$,
- neexistencia vhodných „stavebných blokov“ (napr. Williamsonových postupností) znemožňuje aplikáciu mnohých metód,
- štruktúrne obmedzenia limitujú použiteľnosť jednoduchých konštrukcií.

Z tabuľky 4.1 vyplýva, že pre všetky hodnoty $n < 40$ deliteľné štyrmi existujú Hadamardove matice, pričom počet neekvivalentných riešení rýchlo narastá so zvyšujúcim sa rádom. Napríklad pre $n = 32$ je známych viac než 13 miliónov neekvivalentných matíc.

Rád n	Typ	Existuje H_n ?	Počet neekvivalentných
1	1	Áno	1
2	2	Áno	1
4	$4k$	Áno	1
8	$4k$	Áno	1
12	$4k$	Áno	1
16	$4k$	Áno	5
20	$4k$	Áno	3
24	$4k$	Áno	60
28	$4k$	Áno	487
32	$4k$	Áno	13 710 027
36	$4k$	Áno	neúplne známe

Tabulka 4.1 Prehľad známych Hadamardových matic pre $n < 40$ (a $n = 1, 2$)

Rád n	Faktor k	Vlastnosti k
668	167	Prvočíslo $\equiv 3 \pmod{4}$
716	179	Prvočíslo
892	223	Prvočíslo
1132	283	Prvočíslo
1244	311	Prvočíslo
1388	347	Prvočíslo
1436	359	Prvočíslo
1676	419	Prvočíslo
1772	443	Prvočíslo
1916	479	Prvočíslo
1948	487	Prvočíslo
1964	491	Prvočíslo

Tabulka 4.2 Zoznam neznámych rádov $n = 4k$ pre $n < 2000$

Naopak, tabuľka 4.2 uvádza vybrané hodnoty $n = 4k < 2000$, pre ktoré doposiaľ neboli Hadamardove matice konštruované. Pozoruhodným javom je, že všetky uvedené hodnoty k sú prvočísla. Toto môže naznačovať, že prvočíselná povaha k (alebo jeho faktorizácia na veľké prvočísla) je faktorom obtiažnosti, najmä pre metódy založené na Williamsonových maticiach alebo doplnkových diferenčných množinách, ktoré vyžadujú konštrukciu objektov rádu k . Ak k je veľké prvočíslo, možnosti pre algebraické manipulácie a rozklady sú obmedzenejšie.

- klasické algebraické konštrukcie (napr. Williamsonova, Paleyho) zlyhávajú, pretože vyžadujú rozklad k alebo špecifické aritmetické vlastnosti (napr. existencia kvadratických zvyškov),

- výpočtové hľadanie je extrémne náročné, keďže prehľadávanie priestoru 2^{k^2} možností je prakticky nemožné pre také veľké hodnoty,
- chýbajú známe sekvencie, diferenčné množiny alebo ortogonálne štruktúry požadovanej veľkosti.

Skutočnosť, že rády ako 668 zostávajú nevyriešené napriek desaťročiam výskumu a pokroku vo výpočtovej technike ukazuje na hĺbku a zložitosť Hadamardovej domnienky. Neznamená to nevyhnutne, že matice týchto rádov neexistujú, ale skôr to, že ležia mimo dosahu súčasných konštrukčných a výpočtových paradigiem. Vyriešenie existencie pre rád 668 by pravdepodobne vyžadovalo nový teoretický prielom alebo výrazne výkonnejšie výpočtové stratégie. Hlavné problémy pre rád $n = 668 = 4 \times 167$:

- **Prvočíselný faktor $k = 167$:** Ako bolo uvedené, 167 je prvočíslo. Mnoho štandardných konštrukcií, ako je Williamsonova metóda, vyžaduje existenciu štyroch Williamsonových matíc rádu $k = 167$. Hľadanie takýchto matíc pre prvočíselné rády je notoricky ťažké. Williamsonove matice neexistujú pre niektoré prvočísla, napríklad 47, 53, 59. Nie je známe, či existujú pre $k = 167$ [18].
- **Nevhodné modulo:** Prvočíslo $167 \equiv 3 \pmod{4}$. Paleyho konštrukcia typu I by mohla byť relevantná pre $q = 167$, čo by viedlo k H_{168} , ale nie priamo k H_{167} (ktorá by bola stavebným blokom) [18].
- **Obmedzenia metód založených na postupnostiach:** Goethals-Seidelova metóda by vyžadovala existenciu vhodných postupností (napr. T -postupností alebo doplnkových diferenčných množín) dĺžky 167.
- **Výpočtová náročnosť:** Priame prehľadávanie všetkých ± 1 matíc rádu 668 je prakticky nemožné (2^{668^2} možností). Aj keby sa hľadanie zúžilo na Williamsonove postupnosti dĺžky 167, počet možností je stále astronomický ($2^{4 \times 167}$ pre binárne postupnosti, o niečo menej pre ± 1 symetrické postupnosti). Súčasné heuristické metódy zatiaľ nepriniesli výsledok pre $k = 167$.

Z týchto dôvodov sa zdá, že tieto problematické rády prekračujú súčasný výpočtový horizont. Pokrok si pravdepodobne vyžiada buď nové algebraické konštrukcie, alebo nové paradigmy výpočtového prehľadávania.

4.1 Nedávno skonštruované rády

Napriek pretrvávajúcim výzvam sa v posledných desať ročiach podarilo skonštruovať Hadamardove matice pre niekoľko predtým otvorených rádo-
v:

- **Hadamardova matica rádu 428:** Skonštruovaná v roku 2005 H. Kharaghani a B. Tayfeh-Rezaie [21]. Dovtedy bol $n = 428$ najmenším neznámym rádom. Jeho vyriešenie posunulo hranicu najmenšieho neznámeho rádu na 668.
- **Hadamardove matice rádo-
v 1004 a 2524:** V roku 2014 Dragomir Ž. Đoković, Oleg Golubitsky a Ilias S. Kotsireas [22] publikovali konštrukcie pre tieto rády. Použili Goethals-Seidelovo pole a vhodné doplnkové diferenčné množiny (SDS) nad cyklickou grupou.
- **Šikmé Hadamardove matice rádo-
v 956 a 1324:** V roku 2016 Dragomir Ž. Đoković a Ilias S. Kotsireas [23] skonštruovali prvé známe Hadamardove matice týchto rádo-
v. Použili cyklické diferenčné rodiny $(v; k_1, k_2, k_3; \lambda)$ pre prvočísla $v \equiv 3 \pmod{4}$ a kombinovali ich s Paley-Toddovými diferenčnými množinami. Týmto spôsobom vytvorili Hadamardove matice rádu $4 \cdot 239 = 956$ a $4 \cdot 331 = 1324$.

Tieto úspechy zdôrazňujú dôležitosť spojenia medzi teoretickým vývojom a výpočtovými schopnosťami. Každý nový skonštruovaný rád je výsledkom špecializovaného prístupu.

4.2 Stratégie prehľadávania

Pri rádoch $n \gg 100$ je úplné prehľadanie priestoru všetkých ± 1 matíc nereálne. Preto sa používajú inteligentné výpočtové stratégie:

- **Heuristické a metaheuristické metódy:** Hľadajú (takmer) optimálne riešenia bez záruky globálneho optima.
- **Genetické algoritmy:** Populácia matíc sa iteratívne vylepšuje pomocou kríženia a mutácií. Fitness funkcia hodnotí mieru ortogonal-
ity. GPU-akcelero-
vané verzie umožňujú prácu s tisíckami matíc naraz.
- **Optimalizačné prístupy:** Hľadanie Hadamardových matíc možno formulovať ako optimalizačný problém, kde cieľom je nájsť maticu, ktorá čo najviac spĺňa požadované ortogonálne vlastnosti. Riešenie sa hľadá minimalizáciou alebo maximalizáciou

vhodne zvolenej cieľovej funkcie v rámci definovaného množinového alebo algebraického priestoru.

- **Geometrické prehľadávanie:** Pri náhodnom vynulovaní niektorých prvkov Hadamardovej matice je možné ich doplniť späť. Táto metóda bola použitá na hľadanie symetrických matíc H_{116} a iných.
- **Prehľadávanie špecifických štruktúr:** Zameriava sa na konkrétne algebraické bloky ako Williamsonove postupnosti, T-postupnosti a SDS, čím sa znižuje priestor riešení.
- **Distribované výpočty:** Využívajú paralelné výpočtové siete a boli nasadené napr. pri prehľadávaní Williamsonových matíc pre $n = 100$ až 148 [24].
- **Kvantové algoritmy:** Predpokladá sa, že v budúcnosti môžu pomôcť pri prehľadávaní veľkých priestorov riešení, hoci zatiaľ ide o teoretický prístup.

Úspech výpočtových metód závisí od schopnosti navrhnuť efektívne heuristiky alebo zúžiť priestor riešení pomocou silných algebraických obmedzení.

5 APLIKÁCIE HADAMARDOVÝCH MATÍC

Hadamardove matice sú známe svojou jednoduchou štruktúrou a silnými matematickými vlastnosťami, najmä ortogonalitou. Táto vlastnosť znamená, že jednotlivé riadky (alebo stĺpce) matice sa navzájom nerušia, čo je výhodné v rôznych technických aplikáciách.

Vďaka tomu sa Hadamardove matice využívajú v širokom spektre oblastí: od digitálnej komunikácie a kódovania dát, cez plánovanie experimentov, až po moderné technológie ako kvantové výpočty. Táto kapitola je prehľad vybraných aplikačných oblastí.

5.1 Kódovanie a komunikačné systémy

Pri prenose digitálnych informácií cez rôzne komunikačné kanály, ako sú rádiové vlny, optické káble alebo dokonca vesmírny priestor, môže dôjsť k chybám. Tieto chyby sú spôsobené šumom, rušením alebo inými nedokonalosťami prenosového média. Kódovanie s opravou chýb (Error Correction Coding – ECC) je technika, ktorá pridáva k pôvodnej správe redundantné informácie takým spôsobom, aby bolo možné na strane prijímača tieto chyby nielen detegovať, ale aj opraviť, čím sa zabezpečí spoľahlivý prenos.

5.2 Hadamardove kódy

Hadamardove kódy sú špeciálnym typom lineárnych blokových kódov, ktoré sa dajú priamo odvodiť z Hadamardových matíc. Kódové slová Hadamardovho kódu sú tvorené riadkami príslušnej Hadamardovej matice H . Niekedy sa do sady kódových slov zahŕňajú aj riadky matice $-H$ (t.j. pôvodné riadky s opačnými znamienkami všetkých prvkov). Pre použitie v binárnych systémoch sa prvky $+1$ a -1 Hadamardovej matice mapujú na binárne hodnoty 0 a 1. Bežné mapovanie je $+1 \rightarrow 0$ a $-1 \rightarrow 1$, alebo naopak.

Parametre Hadamardovho kódu odvodeného z Hadamardovej matice rádu n (kde $n = 2^k$ pre matice Sylvesterovho typu) sú nasledovné:

- **Dĺžka kódového slova (n):** Je rovná rádu Hadamardovej matice.
- **Počet kódových slov:** Ak sa použijú len riadky matice H , je to n . Ak sa použijú riadky H aj $-H$ (tzv. rozšírený Hadamardov kód), je to $2n$.
- **Dĺžka pôvodnej správy (dimenzia kódu, k):** Pre n kódových slov je to $k = \log_2(n)$. Pre rozšírený kód s $2n$ slovami je to $k' = \log_2(2n) = k + 1$.

- **Minimálna Hammingova vzdialenosť (d):** Pre Hadamardove kódy je $d = n/2$. Táto veľká vzdialenosť je dôsledkom ortogonalít riadkov – každý pár rôznych riadkov sa líši v presne $n/2$ pozíciách.

Schopnosť kódu opravovať chyby závisí od jeho minimálnej Hammingovej vzdialenosti. Kód s minimálnou vzdialenosťou d dokáže zaručene opraviť až

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

chýb v jednom kódovom slove. Pre Hadamardove kódy to znamená schopnosť opraviť

$$t = \left\lfloor \frac{n/2-1}{2} \right\rfloor$$

chýb. Napríklad Hadamardov kód odvodený z matice H_{32} (rád $n = 32$) má minimálnu Hammingovu vzdialenosť $d = 32/2 = 16$. Takýto kód dokáže opraviť až

$$t = \left\lfloor \frac{16-1}{2} \right\rfloor = 7$$

chýb. Práve takýto kód využila NASA na prenos obrázkov Marsu z vesmírnej sondy *Marsiner 9* v roku 1971 [25], čo demonštruje praktickú hodnotu a robustnosť týchto kódov aj v extrémne náročných podmienkach s vysokou mierou šumu.

Hadamardove kódy, najmä tie odvodené zo Sylvesterových matíc, majú síce veľkú minimálnu Hammingovu vzdialenosť, čo im dáva skvelú schopnosť opravovať chyby, no platia za to nižšou informačnou mierou. Informačná miera kódu, definovaná ako pomer dĺžky pôvodnej správy k dĺžke kódového slova,

$$\frac{k}{n} = \frac{\log_2(n)}{n},$$

je pre Hadamardove kódy relatívne nízka a s rastúcim n klesá. To znamená, že pridávajú značné množstvo redundancie.

Napríklad:

- pre $n = 32$ je $k = 5$ a miera je $\frac{5}{32} \approx 0,156$,
- pre $n = 256$ je $k = 8$ a miera je $\frac{8}{256} \approx 0,031$.

Veľká časť prenášaných bitov je teda redundantná. Z tohto dôvodu sú Hadamardove kódy obzvlášť vhodné pre veľmi zašumené komunikačné kanály, kde je spoľahlivosť prenosu kritickejšia ako jeho efektivita – presne ako v prípade spomínanej vesmírnej komunikácie.

Príklad 5.1. Pre ilustráciu princípu kódovania a opravy chýb pomocou Hadamardovho kódu použijeme maticu H_8 (pozri Obr. 3.1). Táto matica má rád $n = 8$, takže kód bude mať dĺžku

kódového slova 8. Počet kódových slov bude 8 (ak použijeme len riadky H_8), čo znamená, že môžeme zakódovať $k = \log_2(8) = 3$ bity informácie. Minimálna Hammingova vzdialenosť je $d = n/2 = 4$. Takýto kód dokáže opraviť $t = \lfloor \frac{4-1}{2} \rfloor = 1$ chybu.

1. Výber Hadamardovej matice a kódových slov.

Použijeme Sylvesterovu Hadamardovu maticu H_8 rádu $n = 8$. Pre účely binárneho kódovania mapujeme jej prvky podľa konvencie:

$$+1 \rightarrow 0, \quad -1 \rightarrow 1$$

Takto upravené binárne riadky matice H_8 budú slúžiť ako naše kódové slová. Každému 3-bitovému vstupu (správe) priradíme jeden riadok matice podľa jeho indexu.

Informačný vektor u	Kódové slovo $E(u)$
000	00000000
001	01010101
010	00110011
011	01100110
100	00001111
101	01011010
110	00111100
111	01101001

Tabulka 5.1 Hadamardov kód (8, 3)

Poznámka: Kódové slová v tabuľke sú odvodené priamo z riadkov matice H_8 po binarizácii. Pre konzistentnosť v ďalšom texte budeme používať napr. $E(101) = (0, 1, 0, 1, 1, 0, 1, 0)$ a $E(110) = (0, 0, 1, 1, 1, 1, 0, 0)$ ako referenčné prípady.

Poznámka: V praxi sa Hadamardove kódy často generujú pomocou skalárnych súčín modulo 2 medzi vstupným vektorom x a všetkými binárnymi vektormi $z_i \in \{0, 1\}^k$ v lexikografickom poradí. Výsledné kódové slovo má tvar:

$$E(x) = (z_0 \cdot x, z_1 \cdot x, \dots, z_{n-1} \cdot x) \mod 2,$$

kde $n = 2^k$ je dĺžka kódového slova. Tento výpočet je výhodný z hľadiska efektívnej implementácie, pretože nevyžaduje uchovávanie celej Hadamardovej matice. Oba prístupy sú matematicky ekvivalentné.

2. Kódovanie správy.

Povedzme, že chceme odoslať 3-bitovú správu "101". Podľa konštrukcie vyššie je kódové slovo pre "101":

$$c = (0, 1, 0, 1, 1, 0, 1, 0)$$

3. Prenos a vznik chyby.

Predpokladajme, že počas prenosu dôjde k jednej chybe – tretí bit sa zmení z 0 na 1. Prijaté chybné slovo je:

$$r = (0, 1, \mathbf{1}, 1, 1, 0, 1, 0)$$

4. Dekódovanie (metóda najbližšieho suseda).

Na strane prijímača porovnáme prijaté slovo r so všetkými 8 platnými kódovými slovami z tabuľky a vypočítame Hammingovu vzdialenosť (počet bitov, v ktorých sa líšia). Napríklad:

$$d(r, E(000)) = d((01111010), (00000000)) = 5$$

$$d(r, E(001)) = d((01111010), (01010101)) = 3$$

$$\vdots$$

$$d(r, E(101)) = d((01111010), (01011010)) = 1$$

5. Oprava chyby.

Kódové slovo, ktoré má najmenšiu Hammingovu vzdialenosť od prijatého slova r , je $E(101) = (0, 1, 0, 1, 1, 0, 1, 0)$ s hodnotou 1. Keďže kód dokáže opraviť $t = 1$ chybu, dekodér určí, že pôvodne bolo odoslané toto kódové slovo. Tým sa tretí bit opraví ($1 \rightarrow 0$) a obnoví sa pôvodná správa "101".

Poznámka: Vzhľadom na minimálnu Hammingovu vzdialenosť $d = 4$ je Hadamardov kód schopný zaručene detegovať až $d - 1 = 3$ chyby v jednom kódovom slove. To znamená, že ak sa počas prenosu zmenia najviac tri bity, dekodér vždy rozpozna, že prijaté slovo nepatrí medzi platné kódové slová (nebude existovať žiadne s Hammingovou vzdialenosťou nula). V prípade viacerých ako jednej chyby však dekodér nemusí určiť pôvodné slovo jednoznačne – zabezpečená je len detekcia chyby, nie jej korekcia.

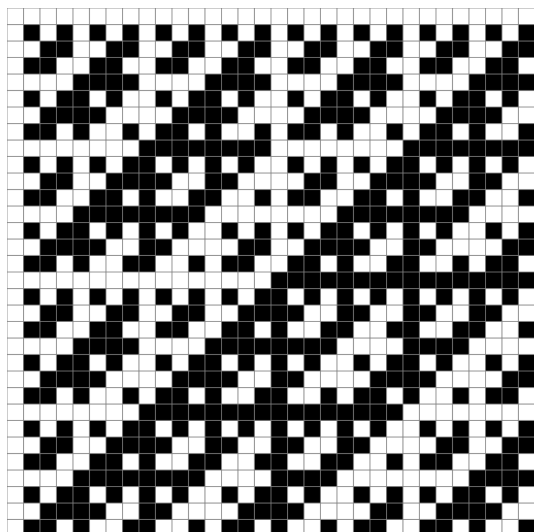
Tento príklad názorne ukazuje, ako ortogonalita Hadamardovej matice vedie k veľkej Hammingovej vzdialenosti medzi kódovými slovami, čo umožňuje detekciu a opravu chýb vzniknutých počas prenosu.

5.3 Walshove matice a ich využitie v CDMA

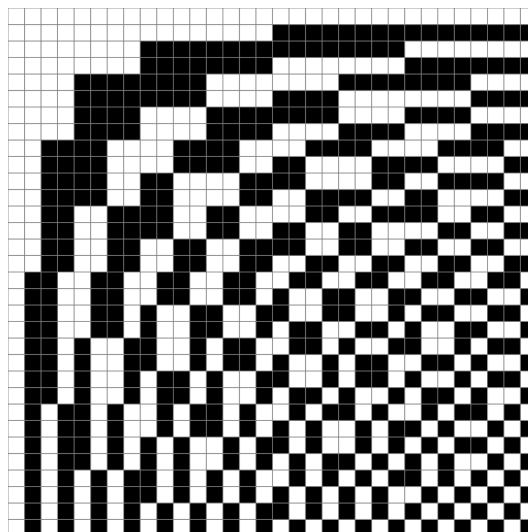
Walshova matica je usporiadaná Hadamardova matica, ktorej riadky sú zoradené podľa tzv. *sekvencného poradia* (*sequency ordering*) – teda podľa počtu prechodov medzi hodnotami $+1$ a -1 v každom riadku. V prípade, že dva alebo viaceré riadky majú rovnaký počet prechodov, poradie medzi nimi nie je jednoznačne určené a môže byť stanovené napríklad po-

mocou lexikografického poradia. Riadky Walshovej matice sú navzájom ortogonálne, vďaka čomu môžu byť použité ako rozprestieracie kódy v komunikačných systémoch.

Walshove matice sa najčastejšie generujú zo Sylvesterových Hadamardových matíc a následne sa ich riadky zoradia podľa počtu znamienkových zmien.



(a) Hadamardova matica H_{32}



(b) Walshova matica W_{32}

Obrázok 5.1 Porovnanie Hadamardovej a Walshovej matice rádu 32.

Walshove funkcie, teda riadky Walshovej matice, sa využívajú najmä v systémoch typu *Code Division Multiple Access* (CDMA) [26]. V takýchto systémoch všetci používatelia prenášajú dáta naraz cez rovnaké frekvenčné pásmo a časový interval, ale každý s použitím iného Walshovho kódu. Vďaka ortogonalite týchto kódov sa dá signál konkrétneho používateľa z prijatého signálu extrahovať pomocou korelácie, pričom sa minimalizuje vzájomné rušenie medzi používateľmi (tzv. *multiple access interference*).

Walshove matice sú obzvlášť výhodné na downlinku (vysielanie zo základňovej stanice), kde je možné zabezpečiť synchronizáciu signálov. Riadky Walshovej matice s nízkym počtom prechodov sú energeticky menej náročné a menej náchylné na rušenie, a preto sa často pridelujú používateľom so slabším signálom.

5.4 Experimentálny dizajn

Hadamardove matice sa v experimentálnom dizajne využívajú na tvorbu ortogonálnych plánov, v ktorých možno presne a nezávisle odhadnúť účinky viacerých faktorov pri malom počte experimentov.

Pri dvojúrovňových faktoroch (napr. zapnuté/vypnuté, nízka/vysoká teplota) sa jednotlivé úrovne často reprezentujú ako $+1$ a -1 . Experimentálny plán je potom tvorený maticou X , kde každý stĺpec zodpovedá jednému faktoru a každý riadok predstavuje konkrétnu kombináciu nastavení vo vykonanom experimente.

Ak sú stĺpce tejto matice navzájom ortogonálne, t. j. $X^T X = nI$ (kde n je počet riadkov a I jednotková matica), znamená to, že medzi faktormi nie je žiadna lineárna závislosť. Ortogonalita zabezpečuje, že účinok každého faktora možno odhadnúť nezávisle od ostatných, čo výrazne zjednodušuje štatistické vyhodnotenie výsledkov.

V lineárnom modeli

$$Y = \beta_0 + \beta_1 x_1 + \dots + \beta_k x_k + \varepsilon,$$

kde Y je vektor meraných hodnôt a ε náhodná chyba, možno odhady parametrov β_i vypočítat pomocou metódy najmenších štvorcov. V prípade ortogonálneho dizajnu má riešenie jednoduchý tvar:

$$\hat{\beta} = (X^T X)^{-1} X^T Y = \frac{1}{n} X^T Y.$$

To znamená, že každý koeficient $\hat{\beta}_i$ závisí len od zodpovedajúceho faktora x_i , nie od ostatných, čo eliminuje problém multikolinearity a zvyšuje spoľahlivosť modelu.

Hadamardove matice, po odstránení prvého (konštantného) stĺpca, sa používajú ako experimentálne plány s n pokusmi pre $n - 1$ dvojúrovňových faktorov [27].

Príklad 5.2. Uvažujme Hadamardovu maticu H_8 veľkosti 8×8 , ktorá spĺňa podmienku $HH^T = 8I_8$. Odstránením prvého stĺpca (ktorý obsahuje samé $+1$) získame dizajnovú maticu so 7 stĺpcami, ktorá sa dá použiť ako plán pre experiment so 7 dvojúrovňovými faktormi a 8 experimentálnymi jednotkami.

Experiment	F_1	F_2	F_3	F_4	F_5	F_6	F_7
1	+1	+1	+1	+1	+1	+1	+1
2	+1	-1	+1	-1	+1	-1	+1
3	+1	+1	-1	-1	+1	+1	-1
4	+1	-1	-1	+1	+1	-1	-1
5	+1	+1	+1	+1	-1	-1	-1
6	+1	-1	+1	-1	-1	+1	-1
7	+1	+1	-1	-1	-1	-1	+1
8	+1	-1	-1	+1	-1	+1	+1

Tabulka 5.2 Hadamardov dizajn H_8 pre 7 faktorov

V tomto príklade môžeme faktory interpretovať napríklad nasledovne:

- F_1 : typ použitého materiálu (A/B),
- F_2 : teplota (nízka/vysoká),
- F_3 : doba ohrevu (krátka/dlhá),
- F_4 : tlak (nízky/vysoký),
- F_5 : chladenie (áno/nie),
- F_6 : druh katalyzátora (I/II),
- F_7 : rýchlosť miešania (pomalá/rýchla).

Matica H_8 môže byť zostrojená napríklad pomocou Sylvestrovej rekurzívnej konštrukcie. Každý riadok zodpovedá jednému konkrétnemu experimentálnemu nastaveniu, kde $+1$ označuje vyššiu úroveň (napr. vyššiu teplotu, prítomnosť chladenia), a -1 nižšiu úroveň. Výsledkom experimentu je zvyčajne meranie nejakej odozvy (napr. výťažok reakcie, pevnosť materiálu, presnosť zariadenia). $\triangle$

Výhodou tohto plánu je, že aj napriek tomu, že testujeme až 7 faktorov, postačuje len 8 meraní. Všetky stĺpce sú navzájom ortogonálne, takže vplyv každého faktora na výstup sa dá odhadnúť nezávisle od ostatných. Takýto plán je vhodný najmä v počiatočnej fáze výskumu, keď sa hľadá, ktoré faktory majú významný vplyv na výsledok [28].

5.5 Kvantové výpočty

Kvantové počítače predstavujú úplne nový spôsob výpočtu, ktorý využíva zákony kvantovej fyziky. Na rozdiel od klasických bitov, ktoré môžu byť iba v stave 0 alebo 1, kvantové bity – tzv. *qubity* — môžu byť v superpozícii oboch stavov naraz. To znamená, že môžu „predstavovať“ 0 aj 1 zároveň, čo umožňuje riešiť niektoré problémy oveľa rýchlejšie ako klasické počítače.

V kvantovej informatike sa na označenie stavov qubitov používa tzv. *Diracova notácia*. Napríklad výraz $|0\rangle$ označuje stav qubit, ktorý zodpovedá klasickej logickej nule, a $|1\rangle$ stav zodpovedajúci jednotke. Ide o tzv. výpočtovú bázu. Kvantový systém však môže existovať aj v superpozícii týchto stavov – napríklad stav $|+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ predstavuje rovnomernú kombináciu oboch báзовých stavov. Podobne stav $|-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ je tiež superpozícia, ale s opačným znamienkom pri $|1\rangle$. Tieto stavy $|+\rangle$ a $|-\rangle$ tvoria tzv. diagonálnu bázu (alebo *X-bázu*), ktorá je ortogonálna k výpočtovej báze a často sa používa napríklad v kvantovej kryptografii.

5.5.1 Hadamardova brána: Kvantová minca

Základným nástrojom kvantového výpočtu je tzv. *Hadamardova brána*. Ide o operáciu, ktorá qubit v klasickom stave (napr. 0 alebo 1) prevedie do superpozície. Funguje podobne ako hod mincou – výsledkom je rovnomerná zmes 0 a 1. V maticovej reprezentácii majú tieto stavy tvar:

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}.$$

Hadamardova brána zapisuje ako matica

$$H = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}.$$

Príklad 5.3 (Aplikácia Hadamardovej brány na základné stavy). Uvažujme dva základné stavy qubitů: $|0\rangle$ a $|1\rangle$. **1. Aplikácia na stav $|0\rangle$:**

$$H|0\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = |+\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

2. Aplikácia na stav $|1\rangle$:

$$H|1\rangle = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = |-\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$$

Oba výsledné stavy $|+\rangle$ a $|-\rangle$ sú tzv. superpozície – kvantové stavy, ktoré predstavujú neurčitý výsledok merania: s pravdepodobnosťou 50 % zmeriate $|0\rangle$ a s pravdepodobnosťou 50 % $|1\rangle$.

Poznámka: V kvantových výpočtoch sa priamo používa len Hadamardova matica rádu 2 (t.j. H_2), pretože Hadamardova brána je jednoqubitová operácia. Hadamardove matice vyšších rádov (napr. H_4 , H_8) vznikajú ako Kroneckerov súčin viacerých H_2 , ak sa táto brána aplikuje na viacero qubitov súčasne.

Schopnosť vytvárať superpozície je základom mnohých kvantových algoritmov. Ak sa Hadamardova brána aplikuje na každý qubit v kvantovom registri (napr. 3 qubity), vytvorí sa superpozícia všetkých možných kombinácií bitov (napr. 000, 001, ..., 111). Kvantový počítač potom môže naraz pracovať so všetkými týmito stavmi, čo sa nazýva *kvantový paralelizmus*.

Tento princíp je využitý v kvantových algoritmoch, ako je *Groverovo kvantové vyhľadávanie* alebo *kvantová Fourierova transformácia*, ktoré ponúkajú rýchlosti nepredstaviteľné

pre klasické stroje.

Hadamardova brána je ako kvantová verzia hodu mincou – vytvára neurčitost', ktorá umožňuje kvantovým počítačom robiť výpočty paralelne a bezpečne kódovať informácie.

5.6 Stručný prehľad ďalších oblastí

Okrem vyššie uvedených hlavných oblastí nachádzajú Hadamardove matice uplatnenie aj v ďalších špecializovaných poliach:

Kryptografia. Vlastnosti Hadamardových matíc boli skúmané pre ich potenciálne využitie v rôznych kryptografických schémach. Niektoré návrhy vychádzajú z teórie kódovania – napríklad McElieceov kryptosystém [29], ktorý môže využívať kódy odvodené z Hadamardových matíc.

Spektroskopia. V Hadamardovej transformačnej spektroskopii sa používajú špeciálne navrhnuté masky, ktorých vzory sú odvodené z riadkov Hadamardovej matice. Tieto masky umožňujú súčasné meranie viacerých spektrálnych zložiek signálu namiesto tradičného sekvencného skenovania jednotlivých frekvencií alebo vlnových dĺžok. Tento prístup, známy ako *multiplexná metóda*, vedie k významnému zlepšeniu pomeru signálu k šumu – tzv. *Fellgettova výhoda* [30].

Kombinatorika. Štúdium Hadamardových matíc tvorí významnú súčasť kombinatoriky. Ich existencia a konštrukcia sú úzko prepojené s ďalšími kombinatorickými objektmi, ako sú symetrické blokové dizajny (SBIBD), konferenčné matice a ortogonálne dizajny [11]. Zvláštnu pozornosť si zasluhuje ich vzťah k tzv. konferenčným maticiam, ktoré slúžia ako základ pri generovaní určitých typov Hadamardových matíc, a tiež k ortogonálnym dizajnom, ktoré umožňujú tvorbu Hadamardových matíc s vyšším stupňom symetrie.

ZÁVER

Hadamardove matice, napriek svojej jednoduchej definícii založenej na prvkoch $+1$ a -1 a vzájomnej ortogonalite riadkov, majú dôležitú úlohu v mnohých oblastiach matematiky a techniky. Ich jednoduchosť ide ruka v ruke s prekvapivou silou využitia v rôznych aplikáciách. V teórii kódovania pomáhajú pri navrhovaní kódov, ktoré dokážu odhaliť a opraviť chyby počas prenosu dát. Pri spracovaní signálov sa používajú na transformáciu a úpravu údajov tak, aby boli lepšie čitateľné či úspornejšie uložené. V experimentálnom výskume pomáhajú navrhovať merania tak, aby sme z nich získali čo najviac informácií s čo najmenším počtom pokusov. A v kvantovom výpočtovom modeli tvoria základ mnohých operácií, ktorými sa vytvárajú stavy umožňujúce paralelné výpočty. To, čo spája všetky tieto oblasti, je schopnosť Hadamardových matic „rozložiť“ informáciu tak, že sa jednotlivé časti navzájom neovplyvňujú – či už ide o signály, merania, alebo kvantové stavy.

Ťažiskom tejto práce bola otázka existencie Hadamardových matic pre rôzne hodnoty rádu n a prehľad konštrukčných metód, ktoré ich dokážu generovať. Táto otázka súvisí s tzv. Hadamardovou domnienkou, podľa ktorej existuje Hadamardova matica práve vtedy, keď jej rozmer n je 1, 2 alebo je násobkom štyroch. Napriek tomu, že táto podmienka je nutná, jej dostatočnosť zostáva otvorenou otázkou, a práve konštrukčné metódy predstavujú hlavný nástroj, ako sa k jej vyriešeniu priblížiť.

V práci boli predstavené rozličné konštrukčné metódy, pričom každá z nich priniesla nové pohľady na to, ako možno Hadamardove matice vytvárať pre čoraz vyššie rády. Tieto prístupy zároveň rozširujú celkové chápanie štruktúr, ktoré za existenciou Hadamardových matic stoja. Keďže neexistuje jednotná ani univerzálna metóda, ktorá by pokrývala všetky prípady $n = 4k$, je dôležité pokračovať vo vývoji nových konštrukčných a výpočtových metód. Ich úspešnosť často závisí od kombinácie viacerých matematických disciplín. Tam, kde algebraické prístupy prestávajú byť účinné, sa čoraz viac uplatňujú heuristické a numerické metódy, ktoré umožňujú preskúmať zložitejšie prípady.

Otvorená zostáva otázka, či Hadamardova domnienka platí všeobecne. Najmenší rád, pre ktorý zatiaľ nie je známa Hadamardova matica, je $n = 668$. Tento stav však nie je dôkazom neexistencie, ale skôr dôsledkom výpočtových obmedzení, keďže pri takýchto veľkých rádoch už tradičné aj heuristické metódy narážajú na hranice dostupných výpočtových kapacít.

Pokroky v tejto oblasti závisia jednak od objavenia nových konštrukčných schém, jednak od lepšieho pochopenia štruktúr, ktoré tieto matice umožňujú.

Hadamardove matice teda nie sú len prostriedkom pre aplikácie – predstavujú zároveň zaujímavý výskumný problém na rozhraní algebraických štruktúr, kombinatoriky a výpočtovej techniky. Práca na ich konštrukcii ponúka nielen praktické výstupy, ale aj nové poznatky o hraniciach našich súčasných matematických nástrojov.

ZOZNAM POUŽITEJ LITERATURY

- [1] SYLVESTER, J. J., 1867. LX. Thoughts on inverse orthogonal matrices, simultaneous sign-successions, and tessellated pavements in two or more colours, with applications to Newton's rule, ornamental tile-work, and the theory of numbers. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*. 1867. 34, č. 232, s. 461–475. Dostupné z: 10.1080/14786446708639826.
- [2] HADAMARD, Jacques, 1893. Résolution d'une question relative aux déterminants. *Bulletin des Sciences Mathématiques*. 1893. 17, s. 240–246.
- [3] PALEY, Raymond E. A. C., 1933. On orthogonal matrices. *Journal of Mathematics and Physics*. 1933. 12, č. 1-4, s. 311–320.
- [4] CATI, Matteo a PASECHNIK, Dmitrii V., 2024. A database of constructions of Hadamard matrices. *arXiv preprint arXiv:2411.18897*. 2024. Dostupné z: <https://arxiv.org/abs/2411.18897>.
- [5] ZHANG, Fuzhen, 2010. *Matrix Theory: Basic Results and Techniques*. Springer-Verlag.
- [6] ROSEN, Kenneth, 2018. *Discrete mathematics and its applications*. New York: Random House. 8th vyd. ISBN 978-1-260-09199-1.
- [7] DUMMIT, David S. a FOOTE, Richard M., 2004. *Abstract Algebra*. John Wiley & Sons. 3rd vyd. ISBN 978-0471433347.
- [8] SZABÓ, Tibor, 2007. Extremal Combinatorics, Recitation 2: Quadratic Residues. <http://www.ti.inf.ethz.ch/ew/courses/extremal07/recitation2.pdf>. Electronic lecture notes, Institute of Theoretical Computer Science, ETH Zürich. Citované 12.3.2025.
- [9] LANG, Serge, 2002. *Algebra*. Roč. 211 / *Graduate Texts in Mathematics*. Springer. Revised third edition vyd. ISBN 978-0-387-95385-4.
- [10] SEBERRY, Jennifer a YAMADA, Mieko, 2020. *Hadamard Matrices: Constructions using Number Theory and Algebra*. Wiley Series in Discrete Mathematics and Optimization. John Wiley & Sons. ISBN 9781119517550.

- [11] HORADAM, Kathy J., 2007. *Hadamard Matrices and Their Applications*. Princeton University Press. ISBN 978-0691119566.
- [12] ČERNOHORSKÝ, Matej, 2025. *Kvadratické zvyšky a Hadamard*. Online. Dostupné z: <https://chatgpt.com/c/67ee4154-9328-8007-a5d5-3a287e8a5d66>. In: ChatGPT. GPT-4o, 26. 3. 2025. Dostupné z: OpenAI [cit. 2025-04-22].
- [13] WILLIAMSON, J., 1944. Hadamard's determinant theorem and the sum of four squares. *Duke Mathematical Journal*. 1944. 11, č. 1, s. 65–81.
- [14] ĐOKOVIĆ, Dragomir Ž., 1993. Williamson matrices of order $n = 33, 35, 39$. *Discrete Mathematics*. 1993. 115, č. 1-3, s. 267–271. Dostupné z: 10.1016/0012-365X(93)90495-F.
- [15] HOLZMANN, W. H.; KHARAGHANI, H. a TAYFEH-REZAIE, B., 2008. Williamson matrices up to order 59. *Designs, Codes and Cryptography*. 2008. 46, č. 3, s. 343–352. Dostupné z: 10.1007/s10623-007-9163-5.
- [16] SLOANE, N. J. A. A Library of Hadamard Matrices. <http://neilsloane.com/hadamard/>. Citované: 10.4.2025.
- [17] TURYN, R. J., 1974. Hadamard matrices, Baumert-Hall units, four-symbol sequences, pulse compression, and surface wave encodings. *Journal of Combinatorial Theory, Series A*. 1974. 16, č. 3, s. 313–333.
- [18] SEBERRY, Jennifer a YAMADA, Mieko, 1992. Hadamard matrices, sequences, and block designs. In: DINITZ, J. H. a STINSON, D. R. (Ed.) *Contemporary Design Theory: A Collection of Surveys*. Wiley-Interscience. s. 431–560.
- [19] LAUNEY, Warwick a FLANNERY, Dane, 2011. *Algebraic Design Theory*. Oxford University Press. ISBN 9780199605532.
- [20] FLANNERY, D. L. a HORADAM, C. M., 1996. COCOA: A system for constructing cocyclic Hadamard matrices. *Australasian Journal of Combinatorics*. 1996. 14, s. 163–176. ISSN 1034-4942.

- [21] KHARAGHANI, Hadi a TAYFEH-REZAIE, Bahman, 2005. A Hadamard matrix of order 428. *Journal of Combinatorial Designs*. 2005. 13, č. 6, s. 435–440. Dostupné z: 10.1002/jcd.20052.
- [22] DJOKOVIĆ, Dragomir Ž.; GOLUBITSKY, Oleg a KOTSIREAS, Ilias S., 2014. Some new orders of Hadamard and skew-Hadamard matrices. *Journal of Combinatorial Designs*. 2014. 22, č. 6, s. 278–287.
- [23] DJOKOVIĆ, Dragomir Ž. a KOTSIREAS, Ilias S., 2016. A class of cyclic $(v; k_1, k_2, k_3; \lambda)$ difference families with $v \equiv 3 \pmod{4}$ a prime. *Special Matrices*. 2016. 4, s. 317–325.
- [24] SEBERRY, Jennifer a YAMADA, Mieko, 2002. A Search for Hadamard Matrices Constructed from Williamson Matrices. Technical report, University of Wollongong. Dostupné z: https://documents.uow.edu.au/~jennie/WEBPDF/2002_07.pdf.
- [25] JET PROPULSION LABORATORY, 1973. Mariner Mars 1971 Project Final Report. Technical Report 32-1550, Jet Propulsion Laboratory, California Institute of Technology. Dostupné z: <https://ntrs.nasa.gov/api/citations/19730012128/downloads/19730012128.pdf>. Volume I: Project Development Through Launch and Trajectory Correction Maneuver.
- [26] VITERBI, Andrew J., 1995. *CDMA: Principles of Spread Spectrum Communication*. Addison-Wesley. ISBN 978-0201633742.
- [27] WU, C. F. Jeff a HAMADA, Michael, 2009. *Experiments: Planning, Analysis, and Optimization*. Wiley. ISBN 9780471699460.
- [28] MONTGOMERY, Douglas C., 2017. *Design and Analysis of Experiments*. Wiley. 9 vyd. ISBN 9781119113478.
- [29] BERNSTEIN, Daniel J a LANGE, Tanja, 2008. McEliece public-key cryptosystem. In: *Post-Quantum Cryptography*. Springer. s. 21–40.
- [30] TAKEDA, Nobuo a NODA, Isao, 1999. Hadamard transform infrared spectroscopy: instrumentation and data analysis. *Applied Spectroscopy*. 1999. 53, č. 10, s. 1212–1219.

ZOZNAM OBRÁZKOV

Obrázok 3.1	Matice H_8 a H_{32} vygenerované Sylvesterovou konštrukciou.....	32
Obrázok 3.2	Matice H_8 a H_{68} vygenerované Paleyho konštrukciou 1. typu.....	35
Obrázok 3.3	Matice H_{12} a H_{60} vygenerované Paleyho konštrukciou 2. typu.....	38
Obrázok 3.4	Hadamardove matice vygenerované Williamsonovou metódou.....	41
Obrázok 3.5	Hadamardove matice skonštruované Turynovými metódami.	47
Obrázok 3.6	Walsh–Hadamardova matica H_{64} vytvorená pomocou kocyklu nad $\mathbb{Z}_2^6$. .	51
Obrázok 3.7	Kocyklické Hadamardove matice rádu $n = 4$ generované hrubou silou nad grupou $\mathbb{Z}_2 \times \mathbb{Z}_2$	53
Obrázok 5.1	Porovnanie Hadamardovej a Walshovej matice rádu 32.	64

ZOZNAM TABULIEK

Tabuľka 3.1	Základné typy sekvencií používané v Turynových metódach	44
Tabuľka 4.1	Prehľad známych Hadamardových matíc pre $n < 40$ (a $n = 1, 2$)	56
Tabuľka 4.2	Zoznam neznámych rádov $n = 4k$ pre $n < 2000$	56
Tabuľka 5.1	Hadamardov kód $(8, 3)$	62
Tabuľka 5.2	Hadamardov dizajn H_8 pre 7 faktorov	65

ZOZNAM POUŽITÝCH SYMBOLOV A SKRATEK

SDS	Supplementary Difference Set (doplnková diferenčná množina)
OA	Orthogonal Array (ortogonálne pole)
$QR(q)$	Quadratic Residue (kvadratický zvyšok v $\mathbb{F}_q$)
$QNR(q)$	Quadratic Non-Residue (nekvadratický zvyšok v $\mathbb{F}_q$)
TS	Turynova sekvencia
$TT(n)$	Turynova sekvencia typu $TT(n)$
CDMA	Code Division Multiple Access
H_n	Hadamardova matica rádu n
I_n	Jednotková (identická) matica rádu n
A^T	Transponovaná matica A
A^{-1}	Inverzná matica k A
$\mathbb{R}^n$	n -rozmerný reálny vektorový priestor
$u \cdot v, \langle u, v \rangle$	Skalárny súčin vektorov u a v
$\ v\ $	Norma (dĺžka) vektora v
$\mathbb{N}, \mathbb{Z}, \mathbb{Q}$	Množiny prirodzených, celých, racionálnych čísel
$\mathbb{F}_q$	Galoisovo pole s q prvkami
$A \times A$	Karteziánsky súčin množiny A so sebou samou
$\mathbb{Z}_n$	Zvyškové triedy modulo n
$\mathbb{Z}_n^*$	Inverzné prvky v $\mathbb{Z}_n$
$\left(\frac{a}{p}\right)$	Legendreov symbol
$(G, *)$	Grupa G s binárnou operáciou $*$
$(F, +, \cdot)$	Pole s operáciami sčítania a násobenia
$\otimes$	Tenzorový (Kroneckerov) súčin
$\circ$	Binárna operácia v cieľovej štruktúre

ZOZNAM PRÍLOH

`cocycle_walsh.py` – Walsh–Hadamardova matica generovaná kocyklickou metódou.

`cocycle_bruteforce.py` – Hľadanie všetkých kocyklických zobrazení vedúcich k Hadamardovej matici.

`paley.py` – Implementácia Paleyho konštrukcie (typ I a II).

`sylvester.py` – Rekurzívna implementácia Sylvestrovej konštrukcie.

`turyn.py` – Generovanie Hadamardových matic pomocou Turynových sekvencií.

`williamson.py` – Generovanie pomocou Williamsonovej konštrukcie.

`utils.py` – Pomocné funkcie (kontrola ortogonality, vizualizácia, export, ...).