UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY
A INFORMATIKY

Evidenčné číslo: 68b6c096-38ab-4871-b255-d4730489860d

GOLAYOVE PÁRY

2011

Jakub Končok

Univerzita Komenského v Bratislave

Fakulta matematiky, fyziky a informatiky

Golayove páry

(Bakalárska práca)

Študijný program: INF Názov: Informatika
Študijný odbor: 9.2.1 Informatika
Školacie pracovisko: Katedra Algebry
Školiteľ: RNDr. Martin Sleziak, PhD.
Konzultant: RNDr. Martin Sleziak, PhD.

Bratislava, 2011 Jakub Končok

ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Jakub Končok
Študijný program: informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: 9.2.1. informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: slovenský

Názov: Golayove páry

Cieľ: Hlavným cieľom práce je popísať a implementovať algoritmy na vyhľadávanie Golayových párov. Okrem algoritmov známych z literatúry je možné pokúsiť sa aj o vlastné algoritmy alebo vylepšenia známych prístupov.

Literatúra: T. H. Andres and R. G. Stanton. Golay sequences. In C. H. C. Little, editor, Combinatorial Mathematics V, pages 44–54. Springer-Verlag, Berlin, 1977. Lecture notes in mathematics 622.
P. B. Borwein and R. A. Ferguson. A complete description of Golay pairs for lengths up to 100. Mathematics of Computation, 73:967–985, 2003.

Vedúci: RNDr. Martin Sleziak, PhD.
Katedra: FMFI.KAGDM - Katedra algebry, geometrie a didaktiky matematiky

Dátum zadania: 25.10.2010

Dátum schválenia: 02.11.2010

doc. RNDr. Daniel Olejár, PhD.
garant študijného programu

.....
študent

.....
Vedúci

Abstrakt

V mojej práci rozoberám a porovnávam niekoľko známych algoritmov na vyhľadávanie Golayovych párov dĺžky n. Začínam najjednoduchším (a zároveň najpomalším) algoritmom, ktorý postupne vylepšujem. Na implementáciu používam programovací jazyk C++. Dôraz je kladený na minimalizáciu doby výpočtu, pretože optimalizácia pamäťovej náročnosti je v tomto prípade jednoduchá a oproti časovým nárokom zanedbateľná. Pri každom algoritme sa snažím dokázať potrebné matematické tvrdenia, ako aj vysvetliť jeho hlavnú myšlienku.

KĽÚČOVÉ SLOVÁ: Golayove páry, Acyklická autokorelácia, Autokorelácia, Implementácia, Hľadanie

Abstract

This work focuses on analysis and comparison of several known search algorithms for Golay pairs of length n. I start with the simplest (and therefore also the slowest) algorithm, which I gradually improve. For implementation, I use the C++ programming language. Due to the memory requirements of implemented search algorithms being easy to manage, and almost negligible compared to their computational complexity, the emphasis is on improving the run time. Also, explanations of basic ideas and principles behind described algorithms is given, together with proofs of all necessary mathematical statements.

KEYWORDS: Golay pairs, Acyclic autocorrelation, Autocorrelation, Implementation, Search

Obsah

Úvod

Pri (nielen) bezdrôtovom prenose informácií je okrem iného dôležité minimalizovať signálový šum. Takzvané signal-to-noise-ratio (SNR), teda pomer signálu a šumu sa dá minimalizovať nielen technickými prostriedkami, ale aj použitím vhodného spôsobu kódovania. Tu hrajú vďaka svojim vlastnostiam veľkú rolu takzvané komplementárne sekvencie. Využitie nájdu napríklad v radarových systémoch alebo pri dátovom prenose pomocou infračervených vĺn. V diagnostickej medicíne sa uplatnia pri neinvazívnych vyšetreniach pomocou ultrazvuku, čo pomáha získať presnejšie údaje v relatívne vysokom rozlíšení bez nadmerného zaťaženia organizmu pacienta.

Zaujímavé matematické vlastnosti a široká škála praktických využití komplementárnych postupností boli podnetom na venovanie sa tejto téme. Zaujal ma aj fakt, že všetky známe algoritmy na vyhľadávanie komplementárnych postupností majú exponenciálnu časovú zložitosť, čo nám dáva veľký priestor na experimentovanie a snahu o optimalizáciu.

Mojou snahou je zrozumiteľne vysvetliť pojem autokorelácia a zdôvodniť motiváciu pre skúmanie jej matematických vlastností. Nemenej dôležitými cieľmi práce sú aj odvodenie, analýza a implementácia rozličných známych algoritmov na vyhľadávanie Golayovych párov.

Pri skúmaní vyhľadávacích algoritmov budem postupovať prirodzenou cestou od pomalých a jednoduchých k rýchlejším a sofistikovanejším. Tiež sa pokúsim dokázať väčšinu matematických tvrdení využitých pri odvodzovaní a konštrukcii algoritmov, a po ich úspešnej implementácii budem porovnávať ich rýchlosti na rozličných vstupoch.

Kapitola 1

Golayove páry a autokorelácia

1.1 Autokorelácia

Základným princípom fungovania aktívnych radarov je vyslanie signálu a následná detekcia jeho návratu. Podľa časového rozdielu zistíme, ako dlho trvalo signálu doraziť k objektu, od ktorého sa odrazil, a vrátiť sa k zdroju.

Po vynásobení tohoto časového údaju rýchlosťou signálu dostaneme vzdialenosť, ktorú signál prešiel k objektu a späť. Vzdialenosť detekovaného objektu od radarového vysielača v čase dopadu signálu potom vypočítame ako polovicu vzdialenosti prejdenej signálom.

$$d = \frac{(t_1 - t_0) \times v}{2}$$

d - vzdialenosť objektu
t_0 - čas vyslania signálu
t_1 - čas návratu signálu
v - rýchlosť signálu

Ako vidíme, je veľmi dôležité vedieť presne určiť dobu príchodu signálu, musíme ho však odlíšiť od šumu. Prijímač preto ustavične koreluje (spája) prichádzajúce signály s istou formou vyslaného signálu [1]. Pri diskrétnom kódovaní si môžeme signál predstaviť ako postupnosť -1 a 1. V dobe návratu vyslaného signálu korelačná funkcia zaznamená niekoľko miernych (vedľajších) nárastov a jeden centrálny (hlavný) nárast. Tento centrálny nárast zodpovedá návratu signálu, preto chceme, aby bol čo najľahšie detekovateľný,

a teda maximálny možný v porovnaní s externým šumom. Energia signálu je obmedzená technickým vybavením. Preto sa snažíme zvýšiť podiel energie pripadajúci centrálnemu nárastu na úkor vedľajších. To sa dá dosiahnuť zvolením vhodného kódovania.

Definícia 1.1.1. Nech A je postupnosť $\left\{a_i\right\}_{i=0}^{n-1}$, $\forall i \in \mathbb{N} : a_i \in \{\pm 1\}$ a $k \in \mathbb{N}, 0 \leq k \leq n-1$. Potom koeficient k-tej acyklickej autokorelácie postupnosti A vypočítame ako:

$$c_k = \sum_{i=0}^{n-k-1} a_i a_{i+k} = a_0 a_k + a_1 a_{k+1} + \cdots + a_{n-k-1} a_{n-1}.$$

Teda koeficient k-tej acyklickej autokorelácie postupnosti vypočítame jej skopírovaním, posunutím o k a následnou sumáciou súčinov nad sebou sa nachádzajúcich členov.

a_0	$\cdots$	a_k	a_{k+1}	$\cdots$	a_{n-1}		
		a_0	a_1	$\cdots$	a_{n-k-1}	$\cdots$	a_{n-1}

Fundamentálnou vlastnosťou autokorelácie je symetria, čo znamená, že koeficienty k-tej autokorelácie postupností $\left\{a_i\right\}_{i=0}^{n-1}$ a $\left\{-a_i\right\}_{i=0}^{n-1}$ sa rovnajú. Vlastnosť symetrie vyplýva priamo z definície autokorelačného koeficientu.

Príklad 1.1.2. Vypočítajme c_2 pre bipolárnu postupnosť A dĺžky $n = 7$. (Bipolárnou rozumieme každú postupnosť, ktorej členy nadobúdajú len hodnoty z množiny $\{\pm 1\}$.)

$$A = [-1, +1, +1, -1, -1, -1, +1]$$

$$c_2 = \sum_{i=0}^{4} a_i a_{i+2} = -1 - 1 - 1 + 1 - 1 = -3$$

Poznámka 1.1.3. Ak $k = 0$, potom:

$$c_k = c_0 = \sum_{i=0}^{n-1} a_i^2 = \sum_{i=0}^{n-1} 1 = n$$

Druhá mocnina koeficientu nultej autokorelácie c_0 reprezentuje energiu centrálneho nárastu, a hodnoty c_k^2 pre $k > 0$ reprezentujú energiu vedľajších nárastov. V praxi sa snažíme dosiahnuť, aby pomer c_k^2/c_0^2 bol pre všetky nenulové k minimálny. Keďže $c_0 = n$ je pre danú dĺžku n konštanta, zlepšenie pomeru môžeme dosiahnuť jedine minimalizáciou $|c_k|$.

Veta 1.1.4. *Nech $k \in \mathbb{N}, k < n$ a c_k je koeficient k-tej acyklickej autokorelácie postupnosti dĺžky n, potom platí kongruencia:*

$$c_k \equiv n - k \pmod 2$$

Dôkaz. Z definície redukciou modulo 2:

$$c_k = \sum_{i=0}^{n-k-1} a_i a_{i+k} \Rightarrow c_k \equiv \sum_{i=0}^{n-k-1} a_i a_{i+k} \equiv n - k \pmod 2$$

posledná kongruencia platí, keďže $1 \equiv -1 \pmod 2$. □

Dôsledkom predchádzajúcej vety je, že ideálny prípad $c_k = 0$ pre $\forall n > k > 0$ nemôže nastať pre žiadnu postupnosť dĺžky väčšej ako 1. Najlepšie, čo môžeme dosiahnuť je $|c_k| \leq 1$ pre $k > 0$. Postupnosti spĺňajúce túto podmienku sa nazývaju Barkerove postupnosti. Tieto však poznáme len pre dĺžky $n \leq 13$. Existencia Barkerových postupností väčšej dĺžky je otvoreným problémom, predpokladá sa však, že neexistujú. Tento fakt viedol k definovaniu tzv. miery vhodnosti postupnosti [2] na jej použitie pri kódovaní signálu.

Túto mieru vypočítame ako:

$$F = c_0^2 / \sum_{k>0} c_k^2.$$

V praxi sa vyskytujú mnohé situácie, kde aj postupnosti s optimálnou hodnotou F vedú k príliš veľkým stratám energie signálu, čo viedlo k novej stratégii - skúmaniu dvojíc komplementárnych postupností, ktorých koeficienty k-tych autokorelácii sa navzájom vynulujú pre $k > 0$ (teda majú rovnaké absolútne hodnoty, ale opačné znamienka). Dvojiciam s takouto vlastnosťou sa budeme venovať v nasledujúcej podkapitole.

1.2 Golayove páry

Golayove páry, pomenované podľa Marcela E. Golaya, boli prvý krát predstavené v jeho článku Multi-slit spectrometry [4]. Ide o dvojice komplementárnych postupností s užitočnými vlastnosťami, vďaka ktorým sú často vhodné ako kódovacie postupnosti so širokým praktickým využitím.

Definícia 1.2.1. Dvojica rovnako dlhých postupností $A = \{a_i\}_{i=0}^{n-1}$ a $B = \{b_i\}_{i=0}^{n-1}$ pre $a_i, b_i \in \{-1, 1\}$ tvorí Golayov pár práve vtedy, ak pre $\forall k \in \mathbb{N}, n-1 \geq k > 0$ platí

$$c_k + d_k = 0$$

kde c_k a d_k sú koeficienty k-tej acyklickej autokorelácie zodpovedajúce postupnostiam A a B.

Dostávame teda sústavu $n-1$ rovníc reprezentujúcich autokorelačné podmienky pre komplementaritu postupností $A = \{a_i\}_{i=0}^{n-1}$, $B = \{b_i\}_{i=0}^{n-1}$:

$$\begin{array}{ccccc}
a_0a_{n-1} & + & b_0b_{n-1} & = & 0 \\
a_0a_{n-2} + a_1a_{n-1} & + & b_0b_{n-2} + b_1b_{n-1} & = & 0 \\
a_0a_{n-3} + a_1a_{n-2} + a_2a_{n-1} & + & b_0b_{n-3} + b_1b_{n-2} + b_2b_{n-1} & = & 0 \\
\cdots & & \cdots & & \\
a_0a_1 + a_1a_2 + \cdots + a_{n-2}a_{n-1} & + & b_0b_1 + b_1b_2 + \cdots + b_{n-2}b_{n-1} & = & 0
\end{array}$$

i-ta rovnica predstavuje podmienku $c_{n-i} + d_{n-i} = 0$.

Príklad 1.2.2. Triviálne všetky dvojice postupností dĺžky jedna sú Golayovým párom (napríklad $A = [1]$, $B = [1]$ alebo $A = [1]$, $B = [-1]$).

Príklad 1.2.3. ($n = 8$)

$$A = [+1, -1, +1, +1, +1, +1, +1, -1]$$

$$B = [+1, -1, +1, +1, -1, -1, -1, +1]$$

Overením autokorelačných podmienok zistíme, či A a B tvoria Golayov pár: $k = 5$

$$c_5 = 1 - 1 - 1 = -1 \text{ a } d_5 = -1 + 1 + 1 = 1$$

dostaneme teda $c_5 + d_5 = -1 + 1 = 0$. Postup zopakujeme pre všetky relevantné hodnoty k ($1 \leq k \leq 7$).

k	0	1	2	3	4	5	6	7
c_k	8	1	2	1	0	−1	2	−1
d_k	8	−1	−2	−1	0	1	−2	1
$c_k + d_k$	16	0	0	0	0	0	0	0

Z tabuľky je vidieť, že nami definované postupnosti A, B tvoria Golayov pár. Pre úplnosť sú uvedené aj hodnoty pre $k = 0$, kedy podľa poznámky 1.1.3 platí $c_0 + d_0 = n + n = 2n$.

Príklad 1.2.4. $(n = 8)$

$$A = [+1, -1, -1, +1, +1, +1, -1, -1]$$

$$B = [-1, -1, -1, +1, -1, -1, -1, -1]$$

k	0	1	2	3	4	5	6	7
c_k	8	1	-4	-3	0	3	2	-1
d_k	8	3	2	1	2	3	2	1
$c_k + d_k$	16	4	-2	-2	2	6	4	0

zjavne A a B netvoria Golayov pár.

Z definície Golayových párov je vidieť, že majú veľmi užitočné vlastnosti pre kódovanie signálov. Teoreticky umožňujú minimalizovať stratovú energiu. V praxi je množstvo aplikácií komplementárnych postupností obmedzené potrebou vysielať 2 signály zároveň. Oba signály musia byť totiž veľmi dobre zosynchronizované. Napriek tomu existuje mnoho situácií, kde výhody kódovania pomocou Golayových párov stoja za komplikácie spôsobené nutnosťou vysielať dvojicu signálov. V ďalšej kapitole budeme popisovať algoritmy na hľadanie Golayových párov a dokazovať matematické tvrdenia potrebné na ich konštrukciu.

Kapitola 2

Vyhľadávacie algoritmy

V tejto kapitole popíšeme a porovnáme niekoľko rôznych algoritmov na vyhľadávanie Golayových párov. Prvý bude prirodzený algoritmus hrubou silou (brute-force). Ten nebude vôbec optimalizovaný a bude slúžiť len na porovnávanie. Ďalšou snahou bude vytvoriť čo najefektívnejší vyhľadávací algoritmus postupným vylepšovaním prirodzeného postupu.

Vstupom pre algoritmus bude jedno celé číslo n, dĺžka, pre ktorú chcem hľadať Golayove páry. Keďže už pre malé dĺžky je počet Golayových párov relatívne veľký, nebudeme vypisovať páry danej dĺžky, ale len uvedieme ich počet.

Všetky popísané algoritmy budú kandidátov na Golayove páry testovať v princípe rovnako (až na drobné úpravy kvôli rozličnej reprezentácii dát). Rozdiely budú v tom, na akej veľkej podmnožine všetkých dvojíc postupností danej dĺžky n sa spustí testovacia sekvencia.

Výpočtovú zložitosť budeme často popisovať známou O-notáciou [5]. Budeme sa snažiť odhadnúť počet vykonaných aritmetických a logických operácií. Ten bude pri všetkých tu uvedených algoritmoch exponenciálne závislý od n, a teda pomerne veľký (rádovo 2^{2n} až $2^{n/2}$).

Vo všeobecnosti budeme vedieť časovú zložitosť odhadnúť výrazom

$$p(n) \times 2^{q(n)},$$

kde $q(n)$ je polynóm prvého stupňa a $p(n)$ je polynóm. Pamäťová zložitosť bude zväčša lineárna, čo sa týka počtu zapamätaných celých čísiel (počet potrebných bitov bude potom $O(n \log(n))$. Jej optimalizácii sa veľmi venovať

nebudeme, pri exponenciálnej časovej zložitosti budeme samozrejme klásť prioritu na rýchlosť.

2.1 Brute-force algoritmus

Princíp brute-force algoritmu bude jednoduchý a intuitívny. Vychádzať bude len z definície Golayovho páru. Pre všetky možné dvojice bipolárnych postupností danej dĺžky n sa overia jednotlivé autokorelačné podmienky z definície 1.2.1.

Existuje 2^{2n} rôznych dvojíc bipolárnych postupností dĺžky n. Overenie konkrétnej dvojice vyžaduje v najhoršom prípade rádovo n^2 operácii sčítania a násobenia (stačí si uvedomiť, že postupnosti počtov týchto aritmetických operácií v jednotlivých autokorelačných podmienkach tvoria aritmetický rad). Počet vykonaných násobení (aj sčítaní) je teda $p(n) \times 2^{2n}$ kde $p(n)$ je polynóm druhého stupňa. Keďže násobenie je zložitejšia operácia ako sčítanie a všetky násobenia vykonávame na množine $\{-1, 1\}$, mohli by sme ich nahradiť jednoduchým podmieneným výrazom. Tento algoritmus však slúži výhradne na ilustráciu problému a na porovnávanie so značne vylepšenými verziami, takže jeho ďalšou optimalizáciou sa nebudeme zaoberať.

Testovaní kandidáti sú pre názornosť reprezentovaní ako celočíselné polia. Pri behu programu sa teda využíva $O(n)$ pamäte. Program implementujúci brute-force algoritmus je súčasťou prílohy, nachádza sa v súbore **algoritmus1.cpp**.

Pri testovaní rýchlosti algoritmu som nameral nasledujúce hodnoty:

vstup n	čas výpočtu	počet párov
1	$< 1s$	4
2	$< 1s$	8
3	$< 1s$	0
4	$< 1s$	32
5	$< 1s$	0
6	$< 1s$	0
7	$< 1s$	0
8	$< 1s$	192
9	$< 1s$	0
10	$< 1s$	128
11	$1s$	0
12	$2s$	0
13	$11s$	0
14	$1min$	0
15	$5min\,35s$	0
16	$24min\,9s$	1536

Symbol $< 1s$ značí, že výpočet trval menej ako jednu sekundu.

2.2 Quady

Aby sme mohli vylepšiť brute-force algoritmus, budeme musieť dokázať niekoľko matematických tvrdení. Pri dôkazoch budeme často využívať modulárne redukcie. Vďaka nim získame množstvo užitočných nutných (vo všeobecnosti nie postačujúcich) podmienok na komplementaritu dvoch postupností.

Pojmom modulárna redukcia myslíme nahradenie rovnice nejakou kongruenciou (prípadne sústavy rovníc systémom kongruencií). Platí totiž:

$$(z_1 = z_2) \Rightarrow (z_1 \equiv z_2 \pmod{n})$$

kde z_1 a z_2 sú ľubovoľné celé čísla a $n \in \mathbb{N}, n > 0$.

Tento vzťah sme už využili pri dôkaze vety 1.1.4. Jeho platnosť je zrejmá, ak sú dve celé čísla rovnaké, zrejme budú dávať aj rovnaké zvyšky po delení ľubovoľným kladným číslom. Množina riešení sústavy rovníc sa po redukcii na sústavu kongruencií určite nezmenší (v zmysle inklúzie), môže len narásť.

Lema 2.2.1. *Pre $a, b \in \{\pm 1\}$ platí nasledovná kongruencia:*

$$ab \equiv a + b - 1 \pmod{4}.$$

Dôkaz. Jednoducho dosadíme všetky možné kombinácie ± 1 za a, b (máme len 4 rôzne možnosti). □

Dôsledok 2.2.2. *Pre Golayov pár (A, B) dĺžky n platí:*

$$\forall i, 0 \leq i \leq n-1 : a_i a_{n-i-1} + b_i b_{n-i-1} = 0 \tag{2.1}$$

Dôkaz. Keďže (A, B) je Golayov pár, potom pre $\forall k \in \mathbb{N}, 1 \leq k \leq n-2$ musí platiť

$$c_{k+1} + d_{k+1} = 0 \text{ a } c_k + d_k = 0$$

Najprv upravíme prvú rovnicu, zrejme

$$c_{k+1} + d_{k+1} \equiv 0 \pmod 4$$

dosadením za c_{k+1} a d_{k+1} dostávame

$$\sum_{i=0}^{n-k-2} a_i a_{i+k+1} + \sum_{i=0}^{n-k-2} b_i b_{i+k+1} \equiv 0 \pmod 4$$

teraz využijeme lemu 2.2.1 , aplikujeme kongruenciu na každý člen sumácie samostatne

$$\sum_{i=0}^{n-k-2} (a_i + a_{i+k+1} - 1) + \sum_{i=0}^{n-k-2} (b_i + b_{i+k+1} - 1) \equiv 0 \pmod 4$$

pri ďalších úpravách použijeme základné vlastnosti sumy

$$\sum_{i=0}^{n-k-2} a_i + \sum_{i=0}^{n-k-2} a_{i+k+1} + \sum_{i=0}^{n-k-2} (-1) + \sum_{i=0}^{n-k-2} b_i + \sum_{i=0}^{n-k-2} b_{i+k+1} + \sum_{i=0}^{n-k-2} (-1) \equiv 0 \pmod 4$$

$$\sum_{i=0}^{n-k-2} a_i + \sum_{i=0}^{n-k-2} a_{i+k+1} + \sum_{i=0}^{n-k-2} b_i + \sum_{i=0}^{n-k-2} b_{i+k+1} - 2(n-k-1) \equiv 0 \pmod 4$$

podobne budeme postupovať pri úprave druhej rovnice (rovnice k-tej autokorelácie)

$$c_k + d_k \equiv 0 \pmod 4$$

dosadíme

$$\sum_{i=0}^{n-k-1} a_i a_{i+k} + \sum_{i=0}^{n-k-1} b_i b_{i+k} \equiv 0 \pmod 4$$

substituujeme členy sumácie podľa lemy 2.2.1

$$\sum_{i=0}^{n-k-1} (a_i + a_{i+k} - 1) + \sum_{i=0}^{n-k-1} (b_i + b_{i+k} - 1) \equiv 0 \pmod 4$$

upravíme sumy

$$\sum_{i=0}^{n-k-1} a_i + \sum_{i=0}^{n-k-1} a_{i+k} + \sum_{i=0}^{n-k-1} (-1) + \sum_{i=0}^{n-k-1} b_i + \sum_{i=0}^{n-k-1} b_{i+k} + \sum_{i=0}^{n-k-1} (-1) \equiv 0 \pmod 4$$

$$\sum_{i=0}^{n-k-1} a_i + \sum_{i=-1}^{n-k-2} a_{i+k+1} + \sum_{i=0}^{n-k-1} b_i + \sum_{i=-1}^{n-k-2} b_{i+k+1} - 2(n-k) \equiv 0 \pmod 4$$

osamostatníme niektoré členy sumácií

$$a_{n-k-1} + a_k + b_{n-k-1} + b_k - 2 +$$

$$+ \sum_{i=0}^{n-k-2} a_i + \sum_{i=0}^{n-k-2} a_{i+k+1} + \sum_{i=0}^{n-k-2} b_i + \sum_{i=0}^{n-k-2} b_{i+k+1} - 2(n-k-1) \equiv 0 \pmod 4$$

z prvej rovnice vieme, že miesto výrazu

$$\sum_{i=0}^{n-k-2} a_i + \sum_{i=0}^{n-k-2} a_{i+k+1} + \sum_{i=0}^{n-k-2} b_i + \sum_{i=0}^{n-k-2} b_{i+k+1} - 2(n-k-1)$$

môžeme substituovať nulu, z čoho dostávame

$$a_k + a_{n-k-1} + b_k + b_{n-k-1} - 2 \equiv 0 \pmod 4$$

čo je podľa lemy 2.2.1 ekvivalentné s podmienkou

$$a_k a_{n-k-1} + b_k b_{n-k-1} = 0,$$

ktorú sme potrebovali dokázať. □

Dôsledok 2.2.3. *Nech existuje Golayov pár dĺžky n, potom buď $n = 1$ alebo n je párne.*

Dôkaz. Nech $n > 1$ je nepárne, potom z dôsledku 2.2.2 vyplýva

$$a_{(n-1)/2}^2 + b_{(n-1)/2}^2 = 0$$

$$1 + 1 = 0$$

čo samozrejme neplatí - spor. □

Z tvrdení 2.2.1 a 2.2.2 vyplýva $a_i a_{n-i-1} b_i b_{n-i-1} = -1$. Teda práve 3 z členov a_i, a_{n-i-1}, b_i a b_{n-i-1} budú mať rovnaké znamienko (z parity n vyplýva, že $i \neq n-i-1$). Toto znamienko a trojicu jemu prislúchajúcich prvkov budeme nazývať dominantným znamienkom (resp. prvkami). Podobne jediný odlišný prvok (a jeho znamienko) budeme nazývať nedominantným.

Golayov pár budeme zapisovať ako postupnosť $\frac{n}{2}$ tzv. quadov:

$$(A, B) = \left\{X_i\right\}_{i=0}^{n/2-1}$$

kde quad budeme zapisovať ako maticu 2×2

$$X_i = \begin{bmatrix} a_i & a_{n-i-1} \\ b_i & b_{n-i-1} \end{bmatrix}$$

Ďalej budeme quadmi nazývať len tie štvorice, ktoré spĺňajú podmienku 2.1.

Existuje 8 rôznych quadov, budeme ich označovať nasledovne:

$$P = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, P^*, -P, -P^*$$

$$Q = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}, Q^*, -Q, -Q^*$$

pričom hviezdička (*) označuje výmenu stĺpcov v matici.

Keďže ich je 8, môžeme každý identifikovať pomocou $\lg(8) = 3$ bitov. Jeden bit reprezentuje dominantné znamienko, ďalší typ quadu (P, Q) a posledný hviezdičku (*), resp. jej absenciu. Všetkých možných štvoríc je $|\{-1, 1\}|^4 = 16$. Pôvodne sme teda reprezentovali dvojicu postupností pomocou $4 \times (n/2) = 2n$ bitov. Ak by sme na reprezentáciu použili len quady, potrebovali by sme len $3 \times (n/2)$ bitov.

To ale znamená, že z pôvodných 2^{2n} možností nám stačí testovať len $2^{3/2n}$. Na tomto fakte môžeme postaviť základy nového algoritmu, najprv si však zavedieme ešte niekoľko pomocných tvrdení a definícií.

Definícia 2.2.4. Operáciu násobenia quadov definujeme ako

$$XY = \begin{bmatrix} x_1 & x_2 \\ x_3 & x_4 \end{bmatrix} \begin{bmatrix} y_1 & y_2 \\ y_3 & y_4 \end{bmatrix} = \frac{1}{4}(x_1y_1 + x_2y_2 + x_3y_3 + x_4y_4)$$

(deliteľnosť 4 môžeme dokázať napríkld pomocou liem 2.2.1 a 2.2.2) ako vidíme, operácia násobenia quadov nie je násobením matíc, ale je v podstate skalárnym súčinom štvorrozmerných vektorov.

Pre násobenie quadov dostávame tabuľku:

$\times$	P	P^*	Q	Q^*
P	1	0	0	0
P^*	0	1	0	0
Q	0	0	1	0
Q^*	0	0	0	1

ďalej platí $-(XY) = (-X)Y = X(-Y)$ a $(-X)(-Y) = XY$, čím sú už určené hodnoty XY pre všetky quady.

Veta 2.2.5. *Pri notácii pomocou quadov budú autokorelačné podmienky ekvivalentné s*

$$\begin{aligned}
X_0X_1^* &= 0\\
X_0X_2^* &= 0\\
X_0X_3^* + X_1X_2^* &= 0\\
X_0X_4^* + X_1X_3^* &= 0\\
X_0X_5^* + X_1X_4^* + X_2X_3^* &= 0\\
X_0X_6^* + X_1X_5^* + X_2X_4^* &= 0\\
&\vdots\\
X_0X_{n-3}^* + X_1X_{n-4}^* + \cdots + X_{(n/2)-1}X_{(n/2)}^* &= 0\\
X_0X_{n-2}^* + X_1X_{n-3}^* + \cdots + X_{(n/2)-1}X_{(n/2)+1}^* &= 0
\end{aligned}$$

Dôkaz. Rovnica $a_0a_{n-1} + b_0b_{n-1} = 0$ platí, pretože X_0 je jedným z 8 prípustných quadov.

Pre $1 \leq i \leq n-2$ je i-ta rovnica ekvivalentná s podmienkou pre koeficient $(n-i-1)$-vej autokorelácie - pri dosadení za súčiny quadov podľa definície 2.2.4 dostaneme pre nepárne i priamo autokorelačnú podmienku, pre párne potrebujeme ešte pripočítať rovnicu $a_{i/2}a_{n-i/2-1} + b_{i/2}b_{n-i/2-1} = 0$ tá platí, pretože $X_{i/2}$ je quad. □

Poznámka 2.2.6. Pre $i \geq n/2$ quady X_i nie sú potrebné na jednoznačné určenie kandidátov na Golayove páry. Každý z nich je totiž symetrický s jedným z quadov s indexom nižším ako $n/2$. V autokorelačných rovniciach sú použité hlavne z estetických dôvodov. Pri výpočtoch sa jednoducho nahradia quadom podľa rovnosti:

$$X_i^* = \begin{bmatrix} a_ia_{n-i-1} \\ b_ib_{n-i-1} \end{bmatrix} = X_{n-i-1}.$$

Zápis autokorelačných podmienok iným spôsobom je uvedený napríklad v [3].

Definícia 2.2.7. Postupnosť $\{X_i\}_{i=0}^{n/2-1}$ môžeme charakterizovať pomocou trojice $(n/2)$-rozmerných binárnych vektorov (H, V, S) (analogicky ako pri reprezentácii jednolivých quadov pomocou trojice bitov).
Vektor horizontálnej orientácie $H = [h_0, h_1, \ldots, h_{n/2-1}]$, kde h_i je 0 ak sa nedominantný prvok i-teho quadu vyskytuje napravo a 1 ak sa vyskytuje naľavo.
Vektor vertikálnej orientácie $V = [v_0, v_1, \ldots, v_{n/2-1}]$, kde v_i je 0 ak sa nedominantný prvok i-teho quadu vyskytuje hore a 1 ak sa vyskytuje dole.
Znamienkový vektor $S = [s_0, s_1, \ldots, s_{n/2-1}]$, kde $s_i \in \{\pm 1\}$ určuje znamienko i-teho quadu. Analogicky môžeme zadefinovať *binárny vektor* $B = [b_0, b_1, \ldots, b_{n/2-1}]$, $b_i = \frac{1}{2}(s_i + 1)$.

Pri programovaní budeme používať zo zjavných dôvodov hlavne B, pri niektorých matematických tvrdeniach sa však lepšie pracuje s vektorom S.

Takto definované vektory budeme využívať na reprezentáciu kandidátov na Golayove páry vo väčšine algoritmov. V nasledujúcej podkapitole ju použijeme na návrh mierne vylepšeného vyhľadávacieho algoritmu.

2.3 Quad algoritmus

Tento algoritmus využíva reprezentáciu testovaných kandidátov pomocou vektorov H, V a B. Tieto vektory sú uložené v pamäti ako celé čísla bez znamienka. Testovacia sekvencia bude pozostávať rádovo z $O(n^2)$ operácii. Operácie využité pri testovaní majú všetky zložitosť lineárnu od počtu bitov (a teda sú logaritmicky závislé od n). Jedná sa o bitové operácie (napríklad bitový AND, SHIFT, XOR), sčítanie a unárnu aritmetickú negáciu. Táto sekvencia sa spustí pre všetkých $2^{3/2n}$ možných hodnôt (H,V,B). Pamäťová náročnosť je minimálna, potrebujeme si uložiť len hodnoty vektorov H, V, S a maximálne nejaký ten medzivýsledok pri testovaní.

Súčasťou práce je samozrejme aj priložený program **algoritmus2.cpp**. Namerané hodnoty trvania behu programu sú opäť uvedené vo forme tabuľky:

vstup n	čas výpočtu	počet párov
2	$< 1s$	8
⋮	⋮	⋮
12	$< 1s$	0
14	$1s$	0
16	$5s$	1536
18	$35s$	0
20	$8min\,10s$	1088
22	$1h\,14min\,44s$	0

Merania sme vykonali len pre párne dĺžky, pre nepárne vráti algoritmus automaticky nulový počet nájdených párov.

2.4 Ďalšie redukcie

Opäť budeme využívať modulárne redukcie modulo 2 a 4. Budeme ich aplikovať na sústavu rovníc predstavujúcich autokorelačné podmienky zapísané pomocou quadov.

Lema 2.4.1. *Pre $i, j < n/2$ platí:*

$$X_i X_j \equiv (1 + v_i + v_j)(1 + h_i + h_j) \pmod 2$$

$$X_i X_j^* \equiv (1 + v_i + v_j)(h_i + h_j) \pmod 2$$

Dôkaz. Z tabuľky pre násobenie quadov vidíme, že ich súčin je nenulový ak sa líšia len znamienkom (teda ak $h_i = h_j$ a $v_i = v_j$). Zároveň všetky nenulové hodnoty súčinov sú nepárne (± 1). Druhá rovnica je dôsledkom faktu, že ak $X_i = X_j^*$, potom $h_i = 1 - h_j$. □

Redukciou autokorelačných podmienok modulo 2 a substitúciou za $X_i X_j$ podľa lemy 2.4.1 získame systém nelineárnych kongruencií. Ak si ale pevne zvolíme jeden z vektorov H (resp. V), dostaneme systém lineárnych kongruencií pre druhý z vektorov. Tento systém vieme riešiť napríklad pomocou metódy Gaussovej eliminácie v polynomiálnom čase. Po vypočítaní vektorov H a V potrebujeme ešte zistiť vektor S.

Poznámka 2.4.2. Pevným zvolením jedného z vektorov myslíme, že za tento vektor postupne dosadzujeme všetky možné vektory $\{0, 1\}^{n/2}$. Pre každé konkrétne dosadenie potom zredukujeme autokorelačné vzťahy a získame systém

lineárnych kongruencií, ktorý musí druhý z vektorov spĺňať, aby dvojica postupností určená vektormi H,V,S mohla byť kandidátom na Golayov pár.

Nasledujúce redukcie budú predpokladať znalosť vektorov H a V, vďaka čomu získame nutné podmienky pre S (resp. B).

Lema 2.4.3. *Pre všetky $i, j < n$ platí:*

$$X_i X_j \equiv (2b_i + 2b_j + 1)|X_i X_j| \pmod 4$$

Dôkaz. Z vlastností násobenia quadov vyplýva, že platí

$$X_i X_j = s_i s_j |X_i X_j|$$

na túto rovnicu aplikujeme redukciu modulo 4, a za $s_i s_j$ dosadíme podľa 2.2.1

$$X_i X_j \equiv (s_i + s_j - 1)|X_i X_j| \pmod 4$$

teraz podľa definície binárneho vektora dosadíme za s_i a s_j a dostaneme hľadaný vzťah

$$X_i X_j \equiv (2b_i + 2b_j + 1)|X_i X_j|$$

□

Teraz autokorelačné podmienky zredukujeme modulo 4 dosadíme za $X_i X_j$ podľa lemy 2.4.3. Hodnoty $|X_i X_j|$ vieme zistiť z 2.4.1, keďže poznáme H a V. Výsledkom bude systém lineárnych kongruencií modulo 4 pre vektor B. Počet nenulových súčinov $X_i X_j$ v každej autokorelačnej rovnici musí byť párny (inak by sa ich súčet nemohol rovnať nule).

Z toho ale vyplýva, že všetky získané kongruencie modulo 4 majú všetky členy párne, teda ich môžeme vydeliť dvomi, a získame tak systém kongruencií modulo 2. Ten vieme opäť riešiť v polynomiálnom čase, a jeho množinou riešení budú kandidáti na vektory B.

Na lepšie pochopenie predchádzajúcich úvah uvedieme jednoduchý príklad:

Príklad 2.4.4. (n=10) Najprv si pevne zvolíme H. Napríklad nech $H = [0, 0, 0, 0, 1]$. Teraz postupne redukujeme všetkých $(n - 2) = 8$ autokorelačných podmienok a následne do nich dosadzujeme pomocou 2.4.1:

$$X_0 X_1^* = 0 \quad \Rightarrow \quad (0 + 0)(1 + v_0 + v_1) \equiv 0 \pmod 2$$
$$X_0 X_2^* = 0 \quad \Rightarrow \quad (0 + 0)(1 + v_0 + v_2) \equiv 0 \pmod 2$$

a podobne aj tretia rovnica sa nám redukuje na $0 \equiv 0 \pmod 2$.
Ďalšie rovnice sa ale už redukujú na použiteľné kongruencie:

$$\begin{aligned}
&X_0X_4^* + X_1X_3^* = 0 \Rightarrow 1 + v_0 + v_4 \equiv 0 \pmod 2 \\
&X_0X_5^* + X_1X_4^* + X_2X_3^* = X_0X_4 + X_1X_4^* + X_2X_3^* = 0 \Rightarrow \\
&\Rightarrow 1 + v_1 + v_4 \equiv 0 \pmod 2 \\
&X_0X_6^* + X_1X_5^* + X_2X_4^* = X_0X_3 + X_1X_4 + X_2X_4^* = 0 \Rightarrow \\
&\Rightarrow (1 + v_0 + v_3) + (1 + v_2 + v_4) \equiv v_0 + v_2 + v_3 + v_4 \equiv 0 \pmod 2
\end{aligned}$$

Analogickým postupom zredukujeme aj zvyšné dve rovnice a vektory koeficientov v nenulových riadkoch zapíšeme do matice:

$$\left(\begin{array}{ccccc|c} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{array}\right) \sim \left(\begin{array}{ccccc|c} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right)$$

Maticu sme rovno upravili na trojuholníkový redukovaný tvar. Vďaka tomuto tvaru matice vidíme, že sústava lineárnych kongruencií pre V má jednu voľnú premennú (jediný stĺpec bez pivota). Počet riešení systému je 2^k, kde k je počet voľných premenných sústavy. Pre všetky (v našom prípade pre obe) riešenia sústavy potom budeme redukovať systém pre vektor S (resp. B).

Riešeniami pre V sú $[1,1,1,0,0]$ a $[0,0,0,1,1]$. My si ukážeme postup na prvom riešení, pri druhom bude analogický.

Po redukcii autokorelačných podmienok modulo 4 a dosadení 2.4.3 dostávame sústavu

$$\begin{array}{ccccccccccc} 2b_0 & + & 2b_2 & + & 2b_3 & + & 2b_4 & + & 2 & \equiv & 0 \pmod 4 \\ 2b_0 & + & 2b_2 & & & & & + & 2 & \equiv & 0 \pmod 4 \end{array}$$

Obe rovnice vydelíme 2 a redukujeme modulo 2 (to môžeme, keďže sú modulo 4, a 4 je deliteľné 2). Dostaneme maticu koeficientov, ktorú upravíme na trojuholníkový redukovaný tvar

$$\left(\begin{array}{ccccc|c} 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{array}\right) \sim \left(\begin{array}{ccccc|c} 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{array}\right)$$

V trojuholníkovom redukovanom tvare matice sa nachádzajú dva pivoty (vedúce prvky) a 3 voľné premenné. Táto sústava má teda $2^3 = 8$ rôznych riešení. Pre všetky tieto riešenia nám už len stačí overiť, či sú Golayovými pármi.

Teraz sa pokúsime implementovať algoritmus využívajúci postup z príkladu 2.4.4.

2.5 $2^{n/2}$ algoritmus

Myšlienka tohto algoritmu je jednoduchá. Snažíme sa eliminovať čo najviac kandidátov na Golayov pár ešte pred ich testovaním. Hlavný cyklus programu sa spustí $2^{n/2}$-krát, raz pre každú možnú hodnotu vektora H. Pri každej iterácii sa potom vytvorí matica systému kongruencií, ktoré musí vektor V spĺňať. Mohutnosť množiny riešení tohoto systému je exponenciálne závislá od počtu voľných premenných v systéme.

Následne sa pre každý validný vektor V spustí vnútorný cyklus. Pri jeho iterácii sa vytvorí matica pre druhý systém kongruencií, tentoraz pre vektor B. Po vyriešení tohoto systému sa spustí štandardný testovací algoritmus s polynomiálnou časovou zložitosťou, identický s tým použitým v quad $(2^{3/2n})$ algoritme.

Algoritmus založený na tejto myšlienke je implementovaný v programe **algoritmus3.cpp**. Tu uvedieme len značne zjednodušený pseudokód.

```
pre všetky H urob {
   redukuj pomocou H;
   vyrieš sústavu;
      pre všetky riešenia V urob {
          redukuj pomocou (H, V);
          vyrieš sústavu;
          pre všetky riešenia S urob {
              otestuj(H, S, V);
          }
      }
}
```

Redukcie budú prebiehať v $O(n^2)$, mechanicky po jednom člene sa bude dosadzovať podľa liem 2.4.1 a 2.4.3.

Sústavy sú riešené pomocou Gauss-Jordanovej eliminácie. Za voľné premenné sa postupne dosádzajú všetky možné kombinácie hodnôt a hodnoty pivotov sa dopočítavajú.

Výsledná zložitosť závisí najmä od počtu voľných premenných v riešených sústavách (a to exponenciálne).

Opäť uvádzame aj tabuľku výsledkov a doby trvania výpočtu.

vstup n	čas výpočtu	počet párov
2	$< 1s$	8
⋮	⋮	⋮
24	$< 1s$	0
26	$< 1s$	64
28	$< 1s$	0
30	$< 1s$	0
32	$6s$	15360
34	$2s$	0
36	$5s$	0
38	$10s$	0
40	$1min\,24s$	9728
42	$58s$	0
44	$2min\,27s$	0
46	$5min\,9s$	0
48	$44min\,51s$	0
50	$30min\,6s$	0
52	$1h\,11min\,40s$	512
54	$2h\,33min\,7s$	0

2.6 Stručné zhrnutie algoritmov

V krátkosti si zopakujme základné informácie o jednolivých algoritmoch.

Brute-force má časovú zložitosť odhadnuteľnú funkciou a pamäťovú lineárnu.

Quad algoritmus má časovú zložitosť $2^{3/2n}$ a logaritmickú pamäťovú zložitosť.

Posledný algoritmus vieme zdola ohraničiť $2^{n/2}$, reálna zložitosť závisí od počtu voľných premenných v jednotlivých riešených sústavách. jeho pamäťová zložitosť je $n \times \log(n)$.

Porovnávanie reálnych hodnôt dĺžiek behov programov je nepraktické porovnávať, keďže najpomalší z algoritmov má dobu výpočtu v rozsahu 1 se-

kunda a 1 hodina na vstupoch $11 \leq n \leq 16$, stredne rýchly $14 \leq n \leq 20$ a najrýchlejší $32 \leq n \leq 50$.

Vidíme však, že rozdiely v ich efektivite sú naozaj nezanedbateľne veľké, čo potvrdzujú reálne hodnoty aj odhady výpočtovej zložitosti.

Záver

Prvým cieľom práce bolo zrozumiteľne vysvetliť pojmy autokorelácia a Golayove páry.

V prvej kapitole sme sa najskôr venovali pojmu autokorelácia. Zadefinovali sme ju, opísali sme jej základné vlastnosti a zároveň sme zdôvodnili motiváciu na skúmanie tejto veličiny.Zaoberali sme sa aj bipolárnymi postupnosťami a ich vhodnosťou na kódovanie signálov.

V druhej časti prvej kapitoly sa zameriavame na dvojice komplementárnych postupností, uviedli sme definíciu Golayovho páru a niekoľko jednoduchých príkladov. Spomenuli sme tiež hlavné klady a zápory kódovania signálov pomocou Golayovych párov.

V druhej kapitole sme sa zaoberali vyhľadávacími algoritmi pre Golayove páry..

Najpv sme sa pozreli na principiálne veľmi jednoduchý brute-force algoritmus, ktorý je samozrejme zo všetkých najmenej efektívny, ale práve vďaka svojej jednoduchosti a názornosti nám poslúžil ako štartovný bod pri tvorbe a chápaní pokročilejších postupov. Stručne sme zanalyzovali jeho výpočtovú zložitosť a formou tabuľky sme uviedli namerané dĺžky behu programu na vybraných vstupoch.

Ďalej sme sa dostali k pojmu modulárnej redukcie, a aj vďaka značným rozdielom v efektivite jednotlivých algoritmovsme mohli vidieť, že sa jedná o nástroj, ktorý je v špecifických situáciach veľmi silný. Redukovanie modulo 2 a 4 nám umožnilo zistiť z nelineárnych vzťahov lineárne, ľahko overiteľné nutné podmienky pre komplementaritu postupností.

V poslednej podkapitole sme uviedli finálnu verziu algoritmu, popísali sme ju stručným pseudokódom a opätovne porovnali s ostatnými. Pri porovnávaní nám spôsobili problémy rozdiely v množinách vstupov, na ktorých vieme merať čas bežania jednotlivých programov s dostatočnou presnosťou.

Nepodarilo sa nám dosiahnuť úplne všetky ciele práce (návrh vlastného algoritmu), napriek tomu sme sa však veľa dozvedeli o niektorých už existujúcich algoritmoch, ich výpočtových zložitostiach a princípe fungovania.

Literatúra

[1] Andres, R. G. - Stanton, T. H. 1997. Golay sequences. In *Lecture notes in mathematics*. New York: Springer-Verlag, 1997. s. 44-54.

[2] BORWEIN P. 2002. *Computational Excursions in Analysis and Number Theory*. New York: Springer-Verlag, 2002. ISBN 0-387-95444-9.

[3] BORWEIN R. A. - FERGUSON, P. B. 2004. A complete description of golay pairs for lengths up to 100. In *Mathematics of Computation*. 2004, roč. 62, č. 73, s. 967-985.

[4] GOLAY, M. J. E. 1949 Multi-slit spectrometry. In *Journal of the optical society of America*. 1949, roč. 33, č. 39, s. 437-437.

[5] ĎURIŠ, P. 2009. *Tvorba efektívnych algoritmov*. Bratislava: Knižničné a edičné centrum FMFI UK, 2009. ISBN 978-80-89186-50-1.