

Kodavimo teorijos projekto ataskaita

Justas Baniulis

VU MIF

2023 m. rudens semestras

Turinys

1. Įžanga	2
2. Realizacija	2
3. Trečiųjų šalių bibliotekos	3
4. Atlikimo laikas	3
5. Programos paleidimas	3
6. Programos tekstai ir programiniai sprendimai	4
7. Vartotojo sąsaja	6
8. Eksperimentai	8
8.1. Dekodavimo algoritmo trukmė	8
8.2. Lyderių svorių ir sindromų lentelės sudarymo atmintis	9

1. Įžanga

Realizuota užduotis A11. Užduočiai realizuoti sukurta konsolinė .NET C# programa, kuri modeliuoja nurodytą kodą C virš F_q .

A11 užduoties kodas ir parametrai:

- kūnas $q = 2$ - programa dirba tik su dvejetainiais skaičiais.
- tiesinis kodas $C[n,k]$ - remiamasi paskaitų kosnpektais [2].
- grandininis (step-by-step) dekodavimas - remiamasi pateikta literatūra [1].
- kodo ilgis n , dimensija k , generuojanti matrica G - juos pateikia vartotojas, G standartinio pavidalo.
- klaidos tikimybė p_e - pateikia vartotojas.

Realizuoti 3 scenarijai:

1. Vartotojo įvesto vektoriaus siuntimas nepatikimu kanalu naudojant kodą.
2. Teksto siuntimas nepatikimu kanalu naudojant ir nenaudojant kodą.
3. Paveiksluko siuntimas nepatikimu kanalu naudojant ir nenaudojant kodą.

2. Realizacija

Realizuotos užduoties dalys:

- **Klaidas taisantys kodai:**

- **Baigtiniai kūnai.**

- Dirbant su kodu naudojami baigtiniai kūnai.

- **Kodavimas**

- Tiesinis kodas $C[n,k]$, pagal A11 užduoties sąlygą.

- **Kanalas.**

- Kanalo tikimybę įveda vartotojas programai paprašius. Kanalas pritaiko tikimybę kiekvienam vektoriaus elementui - jeigu programos gautas realusis skaičius mažesnis už tikimybę, vektorius keičiamas, jeigu didesnis, tai nekeičiamas.

- **Dekodavimas.**

- grandininis (step-by-step) dekodavimo algoritmas, pagal A11 užduoties sąlygą.

- **Kita:**

- **Ataskaita.**

- Šis dokumentas.

- **Išeities tekstai.**

- Pateikiami .zip faile. Jie parašyti .NET C# kalba ir yra išskaidyti į klases ir kiekviena turi savo srities metodus. Kiekvienas metodas turi paaškinančius komentarus.

- **1, 2 ir 3 scenarijai.**

- Paleidžiami programoje.

3. Trečiųjų šalių bibliotekos

Trečiųjų šalių bibliotekos nepanaudotos. Visa sistema naudoja bazines .NET C# (6 versijos) konsolinės aplikacijos klases ir metodus.

4. Atlikimo laikas

1 lentelė. Veiklų trukmė

Veikla	Trukmė
Literatūros skaitymas	5 val.
Kodo veikimo aiškinimasis	10 val.
Projektavimas	2 val.
Programavimas	10 val.
Klaidų ieškojimas ir taisymas	10 val.
Ataskaitos ruošimas	5 val.

5. Programos paleidimas

Darbui su programa reikia pasiruošti atlikti šiuos veiksmus:

- Programa yra pateiktos užduoties .rar faile. Paleidimo failas yra pavadinimu „A11.exe“. Išekstrahuoti „A11.exe“ ir failą paleisti. Vykdomasis failas pritaikytas naujausioms Windows OS.
- Reikia nurodyti visus A11 užduoties parametrus - p_e , n , k , G (standartinio pavidalo) ir pasirinkti scenarijaus numerį.
- 1 scenarijui reikia vektoriaus, kuris bus siunčiamas kanalu.
- 2 scenarijui reikia failo, iš kurio jis skaitys tekstą. Programai pateikti kelią iki failo (jis turi būti absoliutus, pvz.: „C:\Users\JustasBaniulis\Desktop\123.txt“). Programa pritaikyta .txt failų tipams, kitų failų tipų veikimas neapibrėžtas.

- 3 scenarijui vykdyti yra reikalingas aplankalas laikiniams failams. Jis turi būti sukurtas „C:\temp_JustoBaniulio“ lokacijoje (kitais žodžiais, „C“ disko lokacijoje, reikalingas aplankalas „temp_JustoBaniulio“). Taip pat, reikalinga .bmp nuotrauka (kitų failų tipų veikimas neapibrėžtas), programa paprašys kelio iki jos (absoliutaus, pvz.: „C:\Users\JustasBaniulis\Desktop\123.bmp“)

6. Programos tekstai ir programiniai sprendimai

- **Program.cs**

Pagrindinis programos failas, kuris paprašo konsolėje įvesti parametrus, pasirinkti scenarijų, tęsti ar baigti darbą.

Programa tikrina įvestis (Pvz.: $n < k$, tai vartotojui reikia vesti duomenis iš naujo)

- **Scenarijai/Scenarijus1.cs**

Skirtas 1 scenarijaus vykdymui, t.y iš konsolės paimti vektorių, jį užkoduoti, siųsti, dekoduoti ir parodyti rezultatą

Programa tikrina įvestis (pvz.: įvedus naują iš kanalo gautą žinutę ne n ilgio, tai vartotojui reikia vesti žinutę iš naujo).

- **Scenarijai/Scenarijus2.cs**

Skirtas 2 scenarijaus vykdymui, t.y nuskaityti nurodytą tekstinį failą, tada tekstą siųsti kanalu ir po to tą patį tekstą koduoti ir siųsti kanalu. Konsolėje rodomi rezultatai tekstai.

Tekstas skaidomas į 16 bitų vektorių (pagal unicode, tiek vienam simboliui skiriama). Jeigu siunčiamas vektorius nėra k ilgio, tada užpildomas nuliais. Saugomas kiekis užpildymų kiekvienam vektoriui. Dirbama tik su .txt failais.

- **Scenarijai/Scenarijus3.cs**

Skirtas 3 scenarijaus vykdymui, t.y nuskaityti ir atidaryti nurodytą paveiklėlio failą, tada suskaidytą vektoriais paveikslėlį siųsti kanalu ir po to tą patį paveikslėlį koduoti ir siųsti kanalu. Rezultatai išsaugomi laikiname aplankale ir atidaromi.

Konsolė negeba parodyti paveiklėlio, todėl jie atidaromi numatyta paveiklėlių programa. Paveikslėlis skaidomas į 8 bitų vektorių. Jeigu siunčiamas vektorius nėra k ilgio, tada užpildomas nuliais. Saugomas kiekis užpildymų kiekvienam vektoriui. Dirbama tik su .bmp failais, kurių antraštė 54 baitai, jie išsaugomi prieš siunčiant ir naudojami po viso proceso, kad atgaminti paveikslėlį.

- **Scenarijai/Pernaudojama.cs**

Realizuoja binarinių skaičių skaidymą ir užkodavimą. Taip pat realizuoja siuntimą kanalu ir dekodavimą.

Dekoduojant atsižvelgiama kiek kiekvienas vektorius buvo užpildytas nuliais. Užkoduoti ir neužkoduoti vektoriai saugomi atkiruose sąrašuose, skaidomi kanalui yra visi.

- **Classes/Matricos/Matrica.cs**

Realizuoja matricos funkcionalumus, kaip vienetinės matricos generavimas, transponavimas ir parametrus, kaip ilgis, dimensija, elementų reikšmės.

Matricos duomenys saugomi kaip sveikųjų skaičių 2D masyvas.

- **Classes/Matricos/GeneruojantiMatrica.cs**

Realizuoja generuojančią matricą G . Galima inicializuoti matricą atsitiktinai arba pačiam nurodant duomenis. Taip pat realizuota ne vienetinės matricos dalies išgavimas, reikalingas kontrolinės matricos gavimui.

Programa dirba tik su standartinio pavidalo matricomis. Generuoja tik standartinio pavidalo.

- **Classes/Matricos/KontrolineMatrica.cs**

Realizuota kontrolinė matrica K .

Gaunama padavus G , paėmus iš jos ne vienetinę dalį, ją transponavus, sukūrus naujų parametrų vienetinę ir suliejus su transponuotąja.

- **Classes/Matricos/OperacijosMatricos.cs**

Realizuoja matricų suliejimą ir binarinę daugybą

Daugyboje dauginimo rezultatai yra sudedami XOR operatoriumi, kadangi dirbame su dvejetainiais kūnais.

- **Classes/Sindromai/LyderiuLentele.cs**

Realizuoja klasės lyderių sindromų ir jų svorių lentelę, reikalingą dekodavimo algoritmui

Sugeneruojamos visos įmanomos iš kanalo atėjusios žinutės, išgaunami sindromai ir pridedami į žodyno tipo sąrašą, kur raktas yra sindromas, o reikšmė yra žinutės svoris. Rezultate gaunami mažiausio žinutės svorio sindromai. Galimų žinučių generavimui pasitelkiama dešimtainės į dvejetainės sistemos konvertavimas.

- **Classes/Dekodavimas.cs**

Realizuotas grandininis dekodavimo algoritmas

Einama per kiekvieną vektoriaus bitą, jis keičiamas, gaunamas sindromas ir tikrinama ar naujos žinutės svoris pamažėjo, jei ne nekeičiam, jei pamažėjo keičiam. Praėjus visus bitus arba gavus svorį 0, dekodavimas nutraukiamas, grąžinamas pakeistas vektorius.

- **Classes/Kanalas.cs**

Realizuoja vektoriaus siuntimą nepatikimu kanalu su nurodyta tikimybe p_e

Atsitiktinis skaičius generuojamas intervale $0 \leq p_e \leq 1$ su 0,000000001 tikslumu. Keičiama, jei atsitiktinis skaičius mažesnis už p_e . Realizuotas atskiras metodas atsitiktiniam skaičiui, nes .NET C# kalboje atsitiktinio skaičiaus a realizacija yra $a < 1$.

- **Classes/Kodavimas.cs**
Realizuotas tiesinis kodavimas

koduojama sudauginus G matricą su vektoriumi.

7. Vartotojo sąsaja

Programa veikia konsolėje, todėl po kiekvieno pasirinkimo/įvedimo reikia paspausti „enter“ klavišą.

1. Nurodyti n .

Pvz.: 6

2. Nurodyti k .

Pvz.: 3. Jeigu $n < k$, tuomet programa gražina į 1. žingsnį, nes tai nevalidūs duomenys.

3. Pasirenkti generuojančios matricos G kūrimą :

(a) Pasirinkimas 1 (įvedus 1):

sugeneruojama atsitiktinė generuojanti matrica G . Išvedimo pvz.:

1	0	0	1	1	0
0	1	0	0	1	0
0	0	1	1	1	1

(b) Pasirinkimas 2 (įvedus 2):

vartotojas suveda generuojančią matricą G . Įveda $k * n$ skaičių. Įvedimas yra skaidomas eilutėmis geresniam skaitomumui. Programa paprašo įvesti po eilutę, praneša kiek skaičių joje įvesti. Jeigu įvestas netinkamas kiekis skaičių, kartojamas eilutės įvedimas.

Pvz.: paprašo 3 kartus įvesti po 6 skaičius, jeigu $n = 6$ ir $k = 3$. Vienos eilutės įvedimo pvz.: „100110“ (dėl patogumo kabelių tarp elementų nevedame)

Atspausdinama gauta G matrica.

4. Nurodyti p_e

Vartotojas įveda klaidos tikimybę. Programa priima skaičius su „.“ (tašku) ir „.“ (kableliu). Pvz.: 0.05 arba 0,05

5. Pasirenkamas scenarijus - vartotojas įveda scenarijaus numerį - 1, 2, 3.

(a) Scenarijus 1.

- Įvesti k ilgio žinutę. Pvz.: „101“ (dėl patogumo kabelių tarp elementų nevedame).
- Programa išveda n ilgio vektorių, kuris gautas užkoduojant žinutę tiesiniu kodu. Išvedimo pvz.: „101000“.

- iii. Programa siunčia vektorių kanalu su p_e tikimybe. Praneša kur, kokias ir kiek klaidų padaro, ir koks vektorius išejo is kanalo.
Pvz.: „Klaida 1 pozicijoje. 1 -> 0“. Gauta iš kanalo vektoriaus išvedimo pvz.: „011000“.
- iv. Programa leidžia pasirinkti ar keisti gautą vektorių (jeigu netenkina padarytos klaidos) ar nekeisti ir tęsti dekodavimą. Pasirinkus vektorių pakeisti, programa paprašo įvesti ilgio vektorių ir tęsia dekodavimą su įvestu vektoriumi. Įvedimo pvz.: „101011“ (dėl patogumo kablelių tarp elementų nevedame).
- v. Parodomas dekodotas vektorius ir atitikmuo procentais su originaliu užkoduotu vektoriumi. Dekoduotos žinutės išvesties Pvz.: „010111“ ir „Originalaus ir dekoduito vektoriu atitikmuo: 0%“

(b) Scenarijus 2.

- i. Įvesti absoliutų kelią iki failo, pvz.: „C:\Users\JustasBaniulis\Desktop\123.txt“. Įvykus klaidai, pavyzdžiui neradus failo ar aplankalo, .NET C# kalba išmes klaidą ir programa stabdoma. Naudojamas visas failo tekstas.
- ii. Programa parodo tekstą, kurio suskaidyti vektoriai nebuvo užkoduoti ir išsiųti pro kanalą. Išvedimo pvz.: „????????)??g????????q“.
- iii. Programa parodo tekstą, kurio suskaidyti vektoriai buvo užkoduoti, išsiųti pro kanalą ir dekodotas. Išvedimo pvz.: „kod??imo teorij?“.

(c) Scenarijus 3.

- i. Įvesti absoliutų kelią iki failo, pvz.: „C:\Users\JustasBaniulis\Desktop\nuotrauka.bmp“. Įvykus klaidai, pavyzdžiui neradus failo ar aplankalo, .NET C# kalba išmes klaidą ir programa stabdoma. Suradus failą, programa jį atidarys su numatyta nuotraukų atidarymo programa („dažniausiu atveju tai būna „Photos“ programėlė).
- ii. Programa suskaido nuotrauką į vektorių, siunčia juos kanalu ir gautą rezultatą išsaugo laikinoje „C:\temp_JustoBaniulio“ lokacijoje. Failo pavadinimas yra „test_nedekoduotas.bmp“. Failas atidaromas numatyta nuotraukų peržiūros programa.
- iii. Programa suskaido nuotrauką į vektorių, juos užkoduoja, siunčia kanalu, dekoduoja ir gautą rezultatą išsaugo laikinoje „C:\temp_JustoBaniulio“ lokacijoje. Failo pavadinimas yra „test_dekodotas.bmp“. Failas atidaromas numatyta nuotraukų peržiūros programa.

6. Darbo tęsimas arba nutraukimas.

Programa leidžia pasirinkti vykdyti naują scenarijų arba baigti darbą. Norint tęsti darbą su nauju scenarijumi spaudžiamas bet koks simbolis ir enter, baigiamas suvedus „end“ ir paspaudus enter. Darbas tęsimas su tais pačiais parametrais n , k , G , p_e .

Prieš kiekvieną išvedimą konsolėje atspausdinamas jo paaiškinimas.

8. Eksperimentai

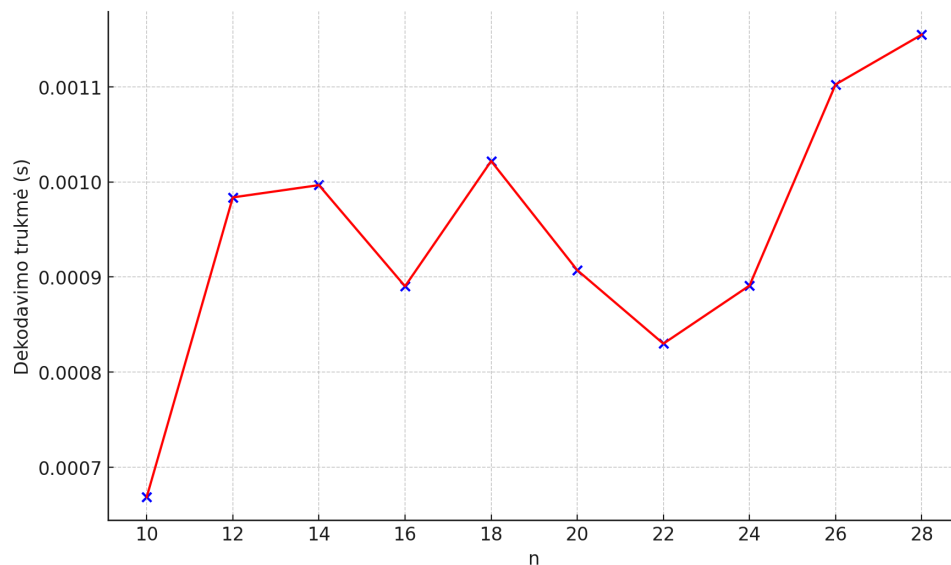
Atlikti du ekspreimentai. Eksperimentuose tirta dekodavimo algoritmo trukmės ir klasės lyderių svorių ir sindormų lentelės sudarymo atminties priklausomybė nuo n parametro.

Kiekviename eksperimente atlikti 10 bandymų. Visuose bandymuose G matrica yra atsitiktinė, kanalas sugadina pusę užkuoto vektoriaus, $k = 8$. n reikšmės yra 10, 12, 14, 16, 18, 20, 22, 24, 26, 28.

8.1. Dekodavimo algoritmo trukmė

2 lentelė. Dekodavimo algoritmo priklausomybė nuo n

n	Dekodavimo trukmė (sekundėmis)
10	0.0006688
12	0.0009837
14	0.0009966
16	0.0008903
18	0.0010219
20	0.0009072
22	0.00083
24	0.000891
26	0.0011024
28	0.0011547



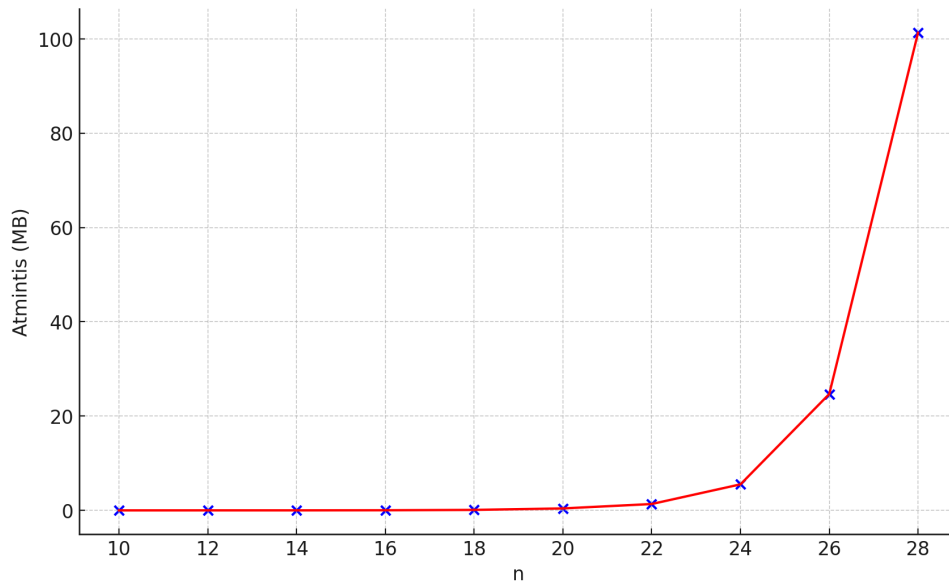
1 pav. Dekodavimo trukmės priklausomybės nuo n grafikas

Nors ir dekodavimo trukmė šiek tiek svyruoja, apibendrinus bandymo duomenis matome, kad trukmė ilgėja su vis didesniu n . Išvada - dekodavimo trukmė ilgėja kartu su didėjančiu n .

8.2. Lyderių svorių ir sindromų lentelės sudarymo atmintis

3 lentelė. Lyderių svorių ir sindromų lentelės sudarymo atminties priklausomybė nuo n

n	Atmintis (MB)
10	0.0078
12	0.0115
14	0.0126
16	0.0321
18	0.1082
20	0.4223
22	1.3553
24	5.526
26	24.6777
28	101.2754



2 pav. Dekodavimo trukmės priklausomybės nuo n grafikas

Iš eksperimento duomenų matome, kad $n < 26$ reikšmėse atminties sunaudojama labai nedaug, tačiau kyla kone eksponentiškai ir nuo 26 pradeda užimti labai didelius kiekius atminties. Tokiu greičiu kylant, ties maždaug $n = 32$ lentelės sudarymo algoritmas turėtų užimti beveik visą prieinamą vidutinio nešiojamojo 16GB RAM kompiuterio atmintį. Išvada - lentelės sudarymo algoritmas užima eksponentiškai daugiau vietos su didėjančiu n .

Literatūra

- [1] P.C. van Oorschot S.A.Vanstone. *An introduction to error correcting codes with applications*. Kluwer Academic Publishers, Boston, 1989.
- [2] Gintaras Skersys. *Klaidas taisančių kodų teorija. Paskaitų konspektai*. 2021.