

# Kriptografija bazirana na rešetkama

---

**Radenić, Martina**

**Master's thesis / Diplomski rad**

**2023**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **University of Zagreb, Faculty of Science / Sveučilište u Zagrebu, Prirodoslovno-matematički fakultet**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:217:750030>

*Rights / Prava:* [In copyright](#)/[Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2024-10-14**



*Repository / Repozitorij:*

[Repository of the Faculty of Science - University of Zagreb](#)



**SVEUČILIŠTE U ZAGREBU**  
**PRIRODOSLOVNO–MATEMATIČKI FAKULTET**  
**MATEMATIČKI ODSJEK**

Martina Radenić

**KRIPTOGRAFIJA BAZIRANA NA**  
**REŠETKAMA**

Diplomski rad

Voditelj rada:  
prof. dr. sc. Filip Najman

Zagreb, veljača, 2023.

Ovaj diplomski rad obranjen je dana \_\_\_\_\_ pred ispitnim povjerenstvom u sastavu:

1. \_\_\_\_\_, predsjednik
2. \_\_\_\_\_, član
3. \_\_\_\_\_, član

Povjerenstvo je rad ocijenilo ocjenom \_\_\_\_\_.

Potpisi članova povjerenstva:

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_

*Mojima.* ♡

# Sadržaj

<b>Sadržaj</b>	<b>iv</b>
<b>Uvod</b>	<b>1</b>
<b>1 Osnovni pojmovi</b>	<b>2</b>
1.1 Kriptografija . . . . .	2
1.2 Sigurnost i složenost . . . . .	4
1.3 NIST-ovi kriteriji sigurnosti . . . . .	5
<b>2 Rešetka</b>	<b>8</b>
2.1 Problem najkraćeg vektora . . . . .	14
2.2 Problem najbližeg vektora . . . . .	16
2.3 Problem najmanje baze . . . . .	16
2.4 Problem učenja s greškama . . . . .	16
<b>3 Algoritmi bazirani na rešetkama</b>	<b>18</b>
3.1 Ajtai-Dwork kriptosustav . . . . .	18
3.2 Goldreich-Goldwasser-Halevi kriptosustav . . . . .	22
3.3 NTRU . . . . .	25
3.4 Usporedba algoritama . . . . .	30
3.5 NIST-ovi kandidati . . . . .	31
<b>Bibliografija</b>	<b>34</b>

# Uvod

Kvantna računala svojom efikasnošću uvelike nadilaze klasična računala. Nudeći odgovore na problem faktorizacije cijelih brojeva i problem diskretnog logaritma, predstavljaju opasnost za današnja sigurnosna rješenja. Mnogi od današnjih najvažnijih komunikacijskih protokola uglavnom se oslanjaju na težinu upravo navedenih problema. Ovaj kvantno mehanički sustav naveo je kriptografe na razvoj novih kriptografskih metoda koje bi bile otporne na napade realizirane na kvantnim računalima.

Post-kvantna kriptografija teži ka pronalasku upravo takvog efektivnog rješenja. Traži da sustav bude siguran bez obzira na računalnu moć protivnika. Kriptografske metode bazirane na rešetkama samo su jedno od rješenja koje se nameće.

Algebarska struktura rešetka i problemi vezani uz nju čine odličnu bazu za kreiranje novih kriptosustava. Ovakve kriptografske metode najčešće se baziraju na problemu pronalaska najkraćeg vektora u rešetki. Ovaj problem se smatra teškim, čak i za kvantna računala, te su kriptografski protokoli bazirani na njemu česti kandidati za post-kvantnu kriptografiju.

U ovom radu proučit ćemo glavna svojstva rešetki te probleme bazirane na njima. Navest ćemo kriptosustave koji svoju sigurnost baziraju upravo na problemima vezanim uz rešetke. Proučit ćemo njihove prednosti i nedostatke te se osvrnuti na njihovu složenost.

Diplomski rad napravljen je u sklopu aktivnosti Projekta KK.01.1.1.01.0004 - Znanstveni centar izvrsnosti za kvantne i kompleksne sustave te reprezentacije Liejevih algebri.

# Poglavlje 1

## Osnovni pojmovi

### 1.1 Kriptografija

*Kriptografija* (grč. *kryptós* - skriven, *graphein* - pisanje) je znanstvena disciplina koja se bavi proučavanjem metoda prijenosa informacija u takvom obliku da ih samo onaj kome su namijenjene može pročitati. Dvije osobe (*pošiljatelj* i *primatelj*) moraju biti u mogućnosti komunicirati preko nesigurnog komunikacijskog kanala tako da treća osoba, iako ima pristup istom komunikacijskom kanalu, nije u mogućnosti razumjeti njihove poruke. U literaturi se pošiljatelj i primatelj nazivaju Alice i Bob, dok se treća osoba (njihov protivnik) naziva Eva.

*Otvoreni tekst* je poruka koju pošiljatelj šalje primatelju. *Šifriranje* je postupak transformiranja otvorenog teksta u format nerazumljiv neovlaštenom korisniku koristeći unaprijed dogovoreni *ključ*. Rezultat tog postupka nazivamo *šifrat*.

*Kriptoanaliza* je znanstvena disciplina koja proučava čitanje kriptiranih poruka bez poznavanja tajnog ključa. Grana koja obuhvaća kriptografiju i kriptoanalizu naziva se *kriptologija*. Matematičku funkciju korištenu prilikom šifriranja odnosno dešifriranja nazivamo *kriptografski algoritam* ili *šifra*.

**Definicija 1.1.1.** *Kriptosustav* je uređena petorka  $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ , pri čemu su:

- $\mathcal{P}$  konačan skup svih mogućih osnovnih elemenata otvorenog teksta,
- $\mathcal{C}$  konačan skup svih mogućih osnovnih elemenata šifrata,
- $\mathcal{K}$  konačan skup svih mogućih ključeva,
- $\mathcal{E} = \{e_K : K \in \mathcal{K}\}$  skup svih funkcija šifriranja  $e_K : \mathcal{P} \rightarrow \mathcal{C}$ ,
- $\mathcal{D} = \{d_K : K \in \mathcal{K}\}$  skup svih funkcija dešifriranja  $d_K : \mathcal{C} \rightarrow \mathcal{P}$ .

Za svaki  $K \in \mathcal{K}$ , postoji funkcija  $e_K \in \mathcal{E}$ , i odgovarajuća funkcija dešifriranja  $d_K \in \mathcal{D}$ , sa svojstvom  $d_K(e_K(x)) = x$  za svaki  $x \in \mathcal{P}$ .

Iz definicije slijedi da funkcije  $e_K$  moraju biti injektorije inače primatelj ne zna točno dešifrirati šifrat. Također, ukoliko vrijedi  $\mathcal{P} = \mathcal{C}$ , tada su  $e_K$  permutacije.

Od kriptosustava se očekuje da zadovoljava određena svojstva. Njegovi osnovni ciljevi su:

- Tajnovitost - samo onaj kome je poruka namijenjena može ju pročitati,
- Autentičnost - moguće je dokazati identitet,
- Integritet - poslana poruka ne može biti izmijenjena prije nego što je primljena,
- Neosporavanje - moguće je dokazati da je pošiljalac uistinu poslao poruku.

Postoje kriptosustavi koji zadovoljavaju navedena svojstva, ali zbog drugih svojstava nisu praktična za upotrebu. Neki od čimbenika koji utječu na funkcionalnost kriptosustava su veličina ključeva i vrijeme izvršavanja.

Postoje tri klasifikacije kriptosustava:

### 1. Obzirom na tip operacije pri šifriranju

Postoje *supstitucijske* i *transpozicijske šifre*. Supstitucijske nastaju kada se svaki element otvorenog teksta zamijeni nekim drugi elementom, dok transpozicijske nastaju permutacijom elemenata.

### 2. Obzirom na način obrade otvorenog teksta

U ovoj podjeli razlikujemo *blokovne šifre* u kojima se obrađuju blokovi elemenata otvorenog teksta koristeći isti ključ i *protočne šifre* u kojima se elementi obrađuju koristeći paralelno generirani niz ključeva.

### 3. Obzirom na tajnost i javnost ključeva

Kriptosustave dijelimo na *simetrične* i *asimetrične sustave*. Kod simetričnih kriptosustava ključevi za šifriranje i dešifriranje su najčešće identični ili se pak jedan može lako dobiti iz drugoga. Upravo zbog toga se zovu još i *kriptosustavi s tajnim ključem*. Asimetrični kriptosustavi imaju javni ključ za šifriranje budući se ključ za dešifriranje ne može u razumnom vremenu izračunati iz njega. Još se nazivaju i *kriptosustavi s javnim ključem*.



Upravo kriptografija s javnim ključem čini jedan od glavnih sigurnosnih sustava današnjice. Temelji se na korištenju dva ključa. Pretpostavimo da Alice želi Bobu poslati poruku. Razmjena poruke događa se na sljedeći način:

1. Bob generira svoj privatni ključ,
2. Bob generira javni ključ (na temelju privatnoga),
3. Alice šifrira poruku koristeći javni ključ,
4. Bob dekriptira poruku koristeći privatni ključ.

Zbog različitosti ključeva ova metoda je sigurna. Poznavanjem jednoga nemoguće je u razumnome vremenu izračunati drugi ključ. Razlog je što se ključevi generiraju jednosmjernim funkcijama za koje je još uvijek teško pronaći inverznu funkciju.

RSA algoritam je najpoznatiji primjer upravo ovakve generacije. On koristi problem faktorizacije velikih brojeva kao temelj svoje sigurnosti. Načini na koje računalo može pronaći faktore broja (primjerice grubom silom) zahtijevaju veliku količinu vremena te se stoga smatraju presporima da bi predstavljali ozbiljnu prijetnju.

*Digitalni potpis* je kriptosustav s javnim ključem koji omogućuje autentifikaciju pošiljatelja te nepobitnost podataka. Sustav uključuje tri algoritma:

1. Algoritam za generiranje ključa za potpisivanje,
2. Algoritam za potpisivanje poruke,
3. Algoritam za provjeru potpisa.

## 1.2 Sigurnost i složenost

Sigurnost je jedan od glavnih čimbenika koji utječu na kvalitetu kriptosustava. Ne postoji slučaj za koji možemo garantirati potpunu sigurnost algoritma, ali postoje kriteriji koje algoritam treba zadovoljavati kako bi ga se smatralo sigurnim i funkcionalnim. Cilj kriptografije je pronaći dovoljno "teške" probleme koji nisu efikasno rješivi na računalu te na temelju njih razviti algoritme.

**Definicija 1.2.1.** *Kažemo da problem pripada klasi složenosti  $P$  ukoliko je odlučiv na nekom determinističkom Turingovom stroju polinomne vremenske složenosti.*

**Definicija 1.2.2.** *Kažemo da problem pripada klasi složenosti  $NP$  ukoliko je odlučiv na nekom nedeterminističkom Turingovom stroju polinomne vremenske složenosti.*

**Primjer 1.2.3.** Jednostavni primjer NP-teškog problema je problem particije: Za dani skup prirodnih brojeva  $S$ , postoji li particija zadanog skupa na različite skupove  $S_1$  i  $S_2$  takva da je suma elemenata iz  $S_1$  jednaka sumi elemenata  $S_2$ . Ne postoji efikasni algoritam koji daje odgovor na to pitanje. Međutim, ukoliko neki algoritam odgovori da postoji, tada u polinomnom vremenu možemo provjeriti jesu li sume skupova doista jednake.

**Definicija 1.2.4.** Za problem  $p$  kažemo da je NP-potpun ukoliko pripada klasi NP te je svaki problem iz NP polinomno reducibilan na  $p$ .

Pri analizi složenosti određenog kriptografskog algoritama koristit ćemo notaciju veliko  $O$ .

**Definicija 1.2.5.** Neka su  $f, g : \mathbb{N} \rightarrow \mathbb{R}^+$  dvije funkcije. Kažemo da je funkcija  $g$  asimptot-ska gornja međa za funkciju  $f$  ako postoji  $c > 0$  i  $n_0 \in \mathbb{N}$  tako da za svaki  $n \geq n_0$  vrijedi  $f(n) \leq cg(n)$ . Oznaka:  $f(n) = O(g(n))$ .

Nadalje, kažemo da je  $f$

- vremenski konstantna ako  $f(n) = O(1)$ ,
- vremenski linearna ako  $f(n) = O(n)$ ,
- vremenski polinomna ako  $f(n) = O(n^c)$  za  $c > 0$ ,
- vremenski eksponencijalna ako  $f(n) = O(c^n)$  za  $c > 0$ .

Primijetimo da je po definiciji svaki problem iz P ujedno problem iz NP, odnosno vrijedi  $P \subset NP$ . Da bismo kriptografski algoritam smatrali sigurnim želimo ga temeljiti na teškom problemu iz klase NP koji nije u P. Na taj način smatramo da problem nije rješiv na računalu u realnom vremenu. Otvoreno je pitanje vrijedi li  $P = NP$ .

### 1.3 NIST-ovi kriteriji sigurnosti

NIST (engl. *National Institute of Standards and Technology*) je agencija američkog ministarstva trgovine. Između ostalih uloga, NIST je zaslužan za razvijanje standarda sigurnosti kao i za unaprjeđenje razumijevanja i upravljanja rizicima privatnosti. Glavni je čimbenik u razvoju i definiranju raznih kriptografskih algoritama kroz povijest.

2016. godine NIST započinje proces standardizacije post-kvantnih kriptografskih sustava te objavljuje svoje kriterije sigurnosti koje novi standardizirani algoritam mora zadovoljavati.

5 kategorija sigurnosti algoritama, prikazane u tablici 1.1, temelje se na računalnim resursima potrebnim za izvođenje napada grubom silom protiv postojećih NIST standarda za AES <sup>1</sup> i SHA <sup>2</sup> u nizu različitih modela potrošnje resursa, kako običnih tako i kvantnih.

Razbijanje hash funkcija izvodi se napadom kolizije grubom silom, a razbijanje simetrične blokovne šifre napadom pretraživanja ključa grubom silom.

Kategorija	Opis
I	Barem jednako teško razbiti kao AES128 (pretraživanje ključa grubom silom)
II	Barem jednako teško razbiti kao SHA256 (napad kolizije grubom silom)
III	Barem jednako teško razbiti kao AES192 (pretraživanje ključa grubom silom)
IV	Barem jednako teško razbiti kao SHA384 (napad kolizije grubom silom)
V	Barem jednako teško razbiti kao AES256 (pretraživanje ključa grubom silom)

Tablica 1.1: Sigurnosne kategorije

Uz navedene kategorije sigurnosti, važan čimbenik u odluci o standardiziranju kriptografskog algoritma imaju i modeli sigurnosti. Pri definiranju modela promatrat ćemo dvije vrste napada:

- CPA (engl. *Chosen Plaintext Attacks*) - napad odabranim otvorenim tekstom,
- CCA (engl. *Chosen Ciphertext Attacks*) - napad odabranim šifratom.

Najvažniji modeli sigurnosti su sljedeći:

- IND-CPA (engl. *Indistinguishability under chosen plaintext attack*) je model u kojem napadač dobiva dvije riječi te šifrat nasumično odabrane riječi. Napadač odlučuje koja je od početne dvije riječi šifrirana. Model osigurava da sve informacije u polaznom tekstu ostaju skrivene od napadača koji poznaje šifrat.

Kažemo da je algoritam IND-CPA siguran ukoliko svaka odluka unutar polinomnog vremena daje zanemarivo bolje rezultate od nasumičnog pogađanja pod zadanim okolnostima.

- IND-CCA2 (engl. *Indistinguishability under chosen ciphertext attack*) jači je model sigurnosti od prethodnoga. Cilj napadača ostaje isti, ali on ima dodatnu prednost poznavanja ključa sesije. Drugim riječima, napadač ima mogućnost šifrirati ili dešifrirati proizvoljne poruke (uz izuzetak početno zadanih riječi).

<sup>1</sup>Napredni standard šifriranja, engl *Advanced Encryption Standard*.

<sup>2</sup>Sigurni hash algoritam, engl. *Secure Hash Algorithm*.

Kažemo da je algoritam IND-CCA2 siguran ukoliko niti jedan protivnik nema zanemarivu prednost u otkrivanju rješenja pod zadanim okolnostima.

- OW-CPA (engl. *One-wayness under chosen plaintext attack*) je znatno slabiji sigurnosni model od prethodno navedenih. Model osigurava da se šifriranjem proizvoljne poruke ne može dobiti izvorni tekst.

Algoritam je OW-CPA siguran ukoliko je za odabrane napade otvorenog teksta nemoguće otkriti cijeli otvoreni tekst proizvoljno zadanog šifrata.

- EUF-CMA (engl. *Existential unforgeability under chosen message attack*) sigurnosni je model digitalnih potpisa. Napadač dobiva javni ključ te može tražiti potpise na odabranim porukama. Na kraju napadač mora generirati poruku i ispravni potpis tako da poruka nije jedna od ranije generiranih.

Algoritam je EUF-CMA siguran ukoliko niti jedan protivnik nema zanemarivu prednost u ispunjenju gore navedenih uvjeta.

## Poglavlje 2

### Rešetka

Neformalno, rešetke su pravilan raspored točaka u  $n$ -dimenzionalnom prostoru. Prvi puta se proučavaju u 19. stoljeću na područjima teorije brojeva i kristalografije. Od pojave slavnog Lenstra-Lenstra-Lovász algoritma za redukciju baze rešetke, postaju bitni čimbenik u razvoju kriptologije. Donedavno, primjena rešetki u kriptografiji bila je isključivo negativna jer su se koristile kao sredstvo za razbijanje raznih kriptografskih shema. U današnje vrijeme postoje i kriptosustavi temeljeni upravo na težini problema vezanih uz rešetke te one igraju i ključnu ulogu u nekoliko sigurnosnih dokaza.

**Definicija 2.0.1.** *Rešetka je diskretna aditivna podgrupa od  $\mathbb{R}^n$ . Odnosno, to je podgrupa  $\Lambda \subseteq \mathbb{R}^n$  koji zadovoljava sljedeća svojstva:*

- $\Lambda$  je zatvorena na zbrajanje i oduzimanje <sup>1</sup>, (podgrupa)
- Postoji  $\epsilon > 0$  takav da za bilo koje dvije različite točke rešetke  $x \neq y \in \Lambda$  vrijedi  $\|x - y\| \geq \epsilon$ . (diskretna)

Nije svaka podgrupa od  $\mathbb{R}^n$  rešetka.

**Primjer 2.0.2.** *Grupa  $\mathbb{Q}^n$  je podgrupa on  $\mathbb{R}^n$ , ali nije diskretna pa nije rešetka.*

**Primjer 2.0.3.** *Skup  $\mathbb{Z}^n$  je rešetka budući je zatvoren na zbrajanje i oduzimanje te je udaljenost između bilo koja dva elementa barem 1.*

Druge rešetke možemo dobiti iz  $\mathbb{Z}^n$  primjenom (nesingularne) linearne transformacije. Primjerci, ako je  $\mathbf{B} \in \mathbb{R}^{k \times n}$  punog ranga, tada je  $\mathbf{B}(\mathbb{Z}^n) = \{\mathbf{B}\mathbf{x} : \mathbf{x} \in \mathbb{Z}^n\}$  također rešetka. Štoviše sve rešetke se mogu prikazati kao  $\mathbf{B}(\mathbb{Z}^n)$  za neki  $\mathbf{B}$  pa je ekvivalentna definicija rešetke sljedeća:

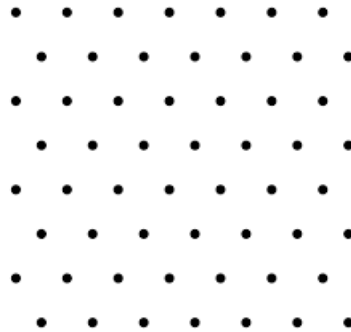
---

<sup>1</sup>Tehnički je dovoljna samo zatvorenost na oduzimanje jer zbrajanje možemo prikazati kao  $a + b = a - (-b)$ .

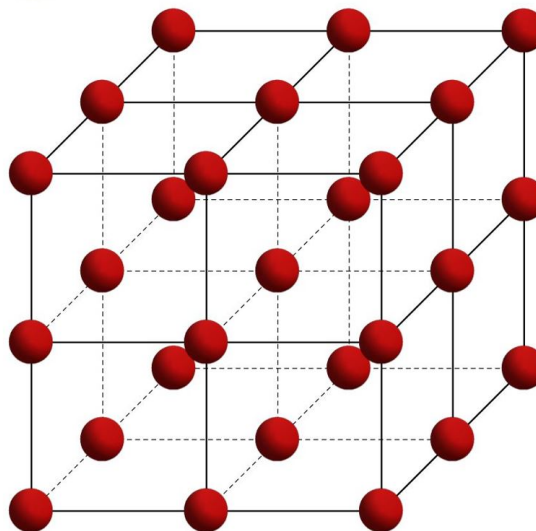
**Definicija 2.0.4.** Neka su  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_k] \in \mathbb{R}^{n \times k}$  linearno nezavisni vektori u  $\mathbb{R}^n$ . Rešetka generirana sa  $\mathbf{B}$  je skup

$$L = L(\mathbf{B}) = \left\{ \mathbf{B}\mathbf{x} : \mathbf{x} \in \mathbb{Z}^k \right\} = \left\{ \sum_{i=1}^k x_i \mathbf{b}_i : x_i \in \mathbb{Z} \right\}$$

svih cjelobrojnih linearnih kombinacija stupaca od  $\mathbf{B}$ . Vektore  $b_1, \dots, b_k$  nazivamo baza rešetke. Brojevi  $n$  i  $k$  su dimenzija odnosno rang rešetke. Ukoliko je  $n = k$  tada kažemo da je rešetka punog ranga.



Slika 2.1: Dvodimenzionalna rešetka



Slika 2.2: Trodimenzionalna rešetka

U računarstvu se najčešće koristi druga definicija jer omogućuje prirodan prikaz rešetke konačnim objektom. Postoji velika sličnost definicije rešetke sa definicijom vektorskog prostora. Naime, vektorski prostor  $V$  generiran sa  $\mathbf{B}$  je skup

$$V(\mathbf{B}) = \left\{ \sum_{i=1}^k x_i b_i : x_i \in \mathbb{R} \right\}.$$

Razlika je u koeficijentima koji u rešetki moraju biti cjelobrojni dok su kod vektorskog prostora dozvoljeni realni.

Primijetimo također da budući da su vektori  $\mathbf{b}_1, \dots, \mathbf{b}_k$  linearno nezavisni, svaki element  $\mathbf{y} \in V(\mathbf{B})$  se može prikazati kao linearna kombinacija  $\mathbf{y} = x_1 \mathbf{b}_1 + \dots + x_k \mathbf{b}_k$  na jedinstven način. Stoga vrijedi  $\mathbf{y} \in L(\mathbf{B})$  ako i samo ako  $x_1, \dots, x_k \in \mathbb{Z}$ .

Ako je  $\mathbf{B}$  baza za rešetku  $L(\mathbf{B})$  onda je baza i za vektorski prostor  $V(\mathbf{B})$ . Međutim, obrat ne vrijedi. Primjerice,  $2\mathbf{B}$  je baza za  $V(\mathbf{B})$ , ali nije baza za  $L(\mathbf{B})$  jer vektor  $\mathbf{b}_i \in L(\mathbf{B})$  (za bilo koji  $i$ ) nije cjelobrojna linearna kombinacija vektora iz  $2\mathbf{B}$ .

Duljinom baze rešetke smatrat ćemo maksimum po svim normama baznih vektora:

$$\max_{b \in \mathbf{B}} \|b\|.$$

**Definicija 2.0.5.** *Prostor razapet rešetkom  $L(\mathbf{B})$  definiran je na sljedeći način:*

$$\text{span}(L) := \left\{ \sum_{i=1}^k x_i \cdot b_i \mid x_i \in \mathbb{R} \right\}.$$

**Teorem 2.0.6.** *Neka su  $\mathbf{B}$  i  $\mathbf{C}$  dvije baze. Tada je  $L(\mathbf{B}) = L(\mathbf{C})$  ako i samo ako postoji unimodularna<sup>2</sup> matrica  $\mathbf{U}$  takva da  $\mathbf{B} = \mathbf{C}\mathbf{U}$ .*

Jednostavan način za dobiti jednu bazu rešetke iz druge je primjenom niza elementarnih transformacija nad stupcima odnosno niza sljedećih operacija:

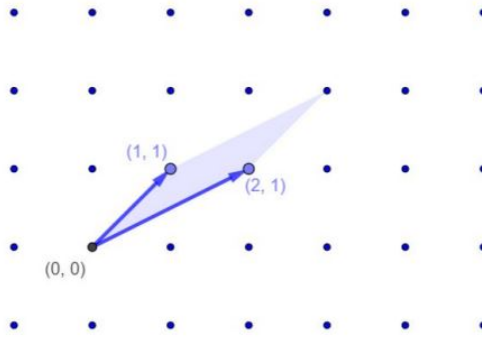
- Zamjena dva stupca matrice,
- Množenje stupca sa -1,
- Dodavanje cjelobrojnog višekratnika stupca drugom stupcu.

---

<sup>2</sup>Cjelobrojna kvadratna matrica čija je determinanta  $\pm 1$ .

**Definicija 2.0.7.** Neka je  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_k]$  baza. Fundamentalni paralelopiped od  $\mathbf{B}$  je skup točaka

$$\mathcal{P}(\mathbf{B}) = \left\{ \sum_{i=1}^k x_i \mathbf{b}_i : 0 \leq x_i < 1 \right\}.$$



Slika 2.3: Fundamentalni paralelopiped

Primijetimo da je  $\mathcal{P}(\mathbf{B})$  poluotvoren tako da translacije  $\mathcal{P}(\mathbf{B}) + \mathbf{v}$  (za  $\mathbf{v} \in L(\mathbf{B})$ ) čine particiju cijelog prostora  $\mathbb{R}^k$ .

**Definicija 2.0.8.** Za proizvoljnu točku  $\mathbf{r} := \mathbf{u} + \mathbf{p} \in \text{span}(L)$ , gdje je  $\mathbf{u} \in L$  i  $\mathbf{p} \in \mathcal{P}(L)$  definiramo

$$\mathbf{p} \equiv \mathbf{r} \pmod{\mathcal{P}(L)}$$

kao redukciju od  $\mathbf{r}$  modulo paralelopiped  $\mathcal{P}(L)$ .

Generalno, definiramo relaciju ekvivalencije za  $\mathbf{a}, \mathbf{b} \in \text{span}(L)$  :

$$\mathbf{a} \equiv \mathbf{b} \iff \mathbf{a} - \mathbf{b} \in L.$$

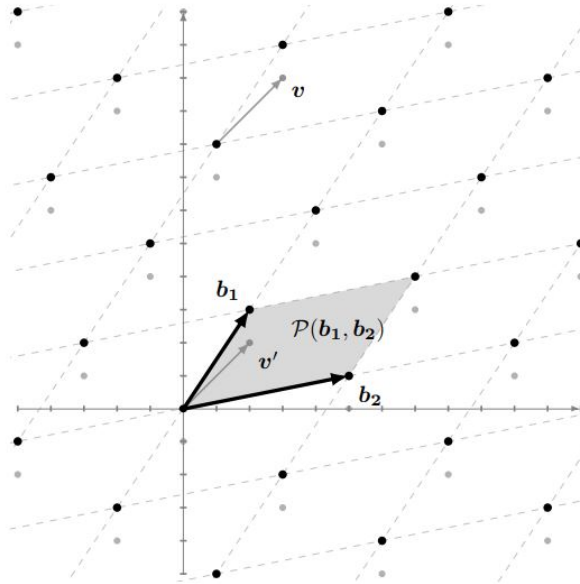
**Primjer 2.0.9.** Slika 2.4 prikazuje rešetku generiranu sa  $\mathbf{b}_1 = (2, 3)$  i  $\mathbf{b}_2 = (5, 1)$ . Crne točke predstavljaju točke rešetke. Osjenčano područje predstavlja paralelopiped  $\mathcal{P}(\mathbf{B}) = \mathcal{P}(\mathbf{b}_1, \mathbf{b}_2)$ . Sive točke predstavljaju točke ekvivalentne

$$\mathbf{v} = 3 \cdot \mathbf{b}_1 - 1 \cdot \mathbf{b}_2 + (2, 2).$$

Točka  $\mathbf{v}'$  je jedinstvena točka u paralelopipedu ekvivalentna  $\mathbf{v}$ :

$$(2, 2) = \mathbf{v}' \equiv \mathbf{v} \pmod{\mathcal{P}(\mathbf{B})}.$$



Slika 2.4: Rešetka generirana sa  $\mathbf{b}_1 = (2, 3)$  i  $\mathbf{b}_2 = (5, 1)$ 

**Definicija 2.0.10.** Neka je  $\mathbf{B}$  baza rešetke  $L = L(\mathbf{B})$ . Volumen rešetke zadan je kao:

$$\text{vol}(L) = \sqrt{\det(\mathbf{B}^T \mathbf{B})}.$$

Ukoliko je  $\mathbf{B}$  ortogonalna,  $\text{vol}(L)$  jednak je  $|\det \mathbf{B}|$ .

**Napomena 2.0.11.** Sada lako vidimo da volumen ne ovisi o izboru baze. Ukoliko imamo dvije baze  $\mathbf{B}_1$  i  $\mathbf{B}_2$  i vrijedi  $L(\mathbf{B}_1) = L(\mathbf{B}_2)$ , tada postoji unimodularna matrica  $\mathbf{U}$  takva da  $\mathbf{B}_1 = \mathbf{U}\mathbf{B}_2$ . Stoga imamo sljedeće jednakosti:

$$\begin{aligned} \text{vol}(L) &= \sqrt{\det(\mathbf{B}_1^T \mathbf{B}_1)} \\ &= \sqrt{\det((\mathbf{U}\mathbf{B}_2)^T (\mathbf{U}\mathbf{B}_2))} \\ &= \sqrt{\det(\mathbf{B}_2^T \mathbf{U}^T \mathbf{U} \mathbf{B}_2)} \\ &= \sqrt{\det(\mathbf{B}_2^T) \det(\mathbf{U}^T \mathbf{U}) \det(\mathbf{B}_2)} \\ &= \sqrt{\det(\mathbf{B}_2^T) \det(\mathbf{B}_2)} \\ &= \sqrt{\det(\mathbf{B}_2^T \mathbf{B}_2)}. \end{aligned}$$

**Definicija 2.0.12.** Za proizvoljnu rešetku  $L = L(\mathbf{B})$ , minimalna udaljenost od  $L$  je najmanja udaljenost dviju točaka rešetke:

$$\lambda_1(L) = \min \{ \|\mathbf{x} - \mathbf{y}\| : \mathbf{x}, \mathbf{y} \in L, \mathbf{x} \neq \mathbf{y} \}.$$

Primijetimo da se minimalna udaljenost može definirati i kao duljina najkraćeg ne-nul vektora rešetke:

$$\lambda_1(L) = \min \{ \|\mathbf{v}\| : \mathbf{v} \in L \setminus \{0\} \}.$$

Ova definicija slijedi iz činjenice da su rešetke aditivne podgrupe od  $\mathbb{R}^n$ . Dakle, ako su  $\mathbf{x}$  i  $\mathbf{y}$  dvije različite točke rešetke, tada  $\mathbf{x} - \mathbf{y}$  pripada rešetki i različito je od 0. Definiciju možemo generalizirati tako da definiramo  $i$ -ti minimum kao najmanji  $\lambda_i$  takav da kugla  $\mathcal{B}_{\lambda_i} = \{x \mid \|x\| \leq \lambda_i\}$  sadrži barem  $i$  linearno nezavisnih vektora rešetke. Označavamo ga s  $\lambda_i(L)$ .

**Definicija 2.0.13.** Neka je  $A \subset \mathbb{R}^n$  ograničen skup i neka je  $\chi : \mathbb{R}^n \rightarrow \{0, 1\}$  njegova karakteristična funkcija definirana s  $\chi(x) = 1$  za  $x \in A$  i  $\chi(x) = 0$  za  $x \in \mathbb{R}^n \setminus A$ . Kažemo da  $A$  ima volumen ako je  $\chi$  integrabilna funkcija. Broj

$$\int_A \chi = \text{vol}(A)$$

nazivamo volumen skupa  $A$ .

**Teorem 2.0.14** (Blichfeldt [13]). Za danu rešetku  $L(\mathbf{B})$  i skup  $S \subseteq \mathbb{R}^m$ , ako je  $\text{vol}(S) > \det(\mathbf{B})$ , tada  $S$  sadrži dvije točke  $z_1, z_2 \in S$  takve da  $\mathbf{z}_1 - \mathbf{z}_2 \in L(\mathbf{B})$ .

Iz prethodnog teorema lako dobivamo gornju ogradu na duljinu najkraćeg vektora rešetke.

**Korolar 2.0.15** (Minkowskijev teorem o konveksnom tijelu [13]). Neka je  $L(\mathbf{B})$  rešetka. Ako je  $S$  konveksno ishodišno-simetrično tijelo za čiji volumen vrijedi  $\text{vol}(S) > 2^m \det(\mathbf{B})$ , tada  $S$  sadrži ne-nul točku rešetke.

Može se pokazati da za proizvoljnu rešetku  $L(\mathbf{B})$  punog ranga postoji točka  $x \in L(\mathbf{B}) \setminus \{0\}$  takva da

$$\|\mathbf{x}\|_\infty \leq \det(\mathbf{B})^{1/n}.$$

Koristeći nejednakost  $\|\mathbf{x}\| \leq \sqrt{n}\|\mathbf{x}\|_\infty$  dobivamo odgovarajuću ogradu za Euklidsku normu.

**Korolar 2.0.16** (Gornja ograda duljine najkraćeg vektora rešetke [13]). Za proizvoljnu rešetku  $L(\mathbf{B})$  postoji točka  $x \in L(\mathbf{B}) \setminus \{0\}$  takva da

$$\|\mathbf{x}\| \leq \sqrt{n} \det(\mathbf{B})^{1/n}.$$

Rešetka međutim može sadržavati vektore proizvoljno kraće od prethodno navedene ograde.

**Primjer 2.0.17.** *Promatramo dvodimenzionalnu rešetku generiranu vektorima  $(1, 0)^T$  i  $(0, N)^T$ , gdje je  $N$  proizvoljan veliki cijeli broj. Rešetka sadrži kratki vektor duljine  $\lambda = 1$ . Međutim, determinanta rešetke je  $N$  pa je gornja ograda definirana u korolaru  $\sqrt{2}N^{1/2}$  što je mnogo veće od 1.*

## 2.1 Problem najkraćeg vektora

Problem najkraćeg vektora ili **SVP** (eng. *Shortest Vector Problem*) najkorišteniji je u kriptografskim metodama. Neformalno, dana nam je baza rešetke i želimo pronaći najkraći ne-nul vektor te rešetke.

**Definicija 2.1.1** (Problem najkraćeg vektora). *Za danu bazu rešetke  $L(\mathbf{B})$  treba pronaći  $\mathbf{u} \in L \setminus \{0\}$  takav da*

$$\|\mathbf{u}\| = \lambda_1(L).$$

**Napomena 2.1.2.** *Najkraći vektor  $\mathbf{u}$  nije jedinstven. Očito ukoliko je  $\mathbf{u}$  kratki vektor tada je to i  $-\mathbf{u}$ . Uz to u rešetci mogu postojati i drugi vektori iste norme kao  $\mathbf{u}$  i  $-\mathbf{u}$ .*

**Definicija 2.1.3** (Problem aproksimacije najkraćeg vektora  $\gamma$ SVP). *Za danu bazu  $\mathbf{B}$  rešetke  $L(\mathbf{B})$  i faktor aproksimacije  $\gamma$  treba pronaći  $\mathbf{v} \in L \setminus \{0\}$  takav da*

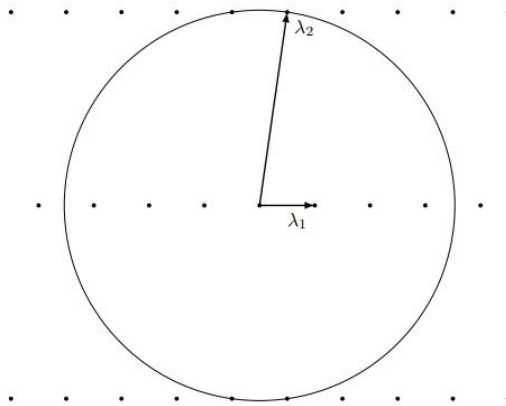
$$\|\mathbf{v}\| \leq \gamma \lambda_1(L).$$

Varijanta problema SVP koja se popularizirala zbog Ajtai-Dwork kriptosustava naziva se problem jedinstvenog najkraćeg vektora. Problem kao takav nije drugačiji već su drugačije pretpostavke o rešetci. Naime rešetka mora sadržavati jedinstveni najkraći vektor. Odnosno, neka je  $\gamma = n^{O(1)}$ , gdje je  $n$  rang rešetke. Rešetka  $L$  ima  $\gamma$ -jedinstveni najkraći vektor ako je  $\lambda_2(L) > \gamma \lambda_1(L)$ .

**Definicija 2.1.4** (Problem jedinstvenog najkraćeg vektora uSVP). *Za danu rešetku  $L$  koja sadrži jedinstveni najkraći vektor treba pronaći  $\mathbf{v} \in L$  takav da*

$$\|\mathbf{v}\| = \lambda_1(L).$$

Koristeći LLL algoritam, možemo riješiti  $\gamma$ SVP za  $\gamma = 2^{\theta(n)}$ . Schnorr 1987. godine predstavlja proširenje LLL algoritma koji dovodi do nešto boljeg faktora aproksimacije. Glavna ideja tog algoritma je zamjena  $2 \times 2$  blokova LLL algoritma sa blokovima većih dimenzija. Povećanje dimenzije bloka dovodi do smanjenja faktora aproksimacije po cijenu



Slika 2.5: Rešetka koja sadrži jedinstveni najkraći vektor

duljeg vremena izvođenja. Postoji nekoliko varijanti Schnorrovog algoritma. Nažalost, sve varijante imaju podjednake eksponencijalne aproksimacije.

Ukoliko želimo pronaći rješenje za SVP, ili samo aproksimaciju  $p(n)$ , najbolji algoritam ima vremensku složenost  $2^{O(n)}$ . Prostorna složenost algoritma je također eksponencijalna što ga čini nepraktičnim. Drugi algoritmi imaju polinomnu prostornu složenost, ali  $2^{n \log n}$  vremensku složenost.

To nas dovodi do sljedeće slutnje:

**Slutnja 2.1.5.** *Ne postoji algoritam polinomne vremenske složenosti koji rješava problem aproksimacije najkraćeg vektora polinomnim faktorom.*

Drugim riječima, problem aproksimacije najkraćeg vektora polinomnim faktorom je težak problem. Sigurnost mnogih kriptosustava baziranih na rešetkama temelji se upravo na ovoj slutnji. Također, od 1980tih nema velikog napretka u performansama što dalje potvrđuje slutnju. Treba napomenuti da aproksimacije faktorima iznad  $\sqrt{n/\log n}$  nisu NP-težak problem. Samo za mnogo manje faktore kao  $n^{O(1/\log \log n)}$  je pokazano da su aproksimacije NP-teški problemi.

U primjeni, algoritmi za redukciju rešetke daju bolje rezultate. Eksperimentima je pokazano da algoritmi daju približno  $\delta^n$  bolje rezultate od očekivanog, gdje je  $n$  dimenzija rešetke a  $\delta$  konstanta koja ovisi o algoritmu. Najbolji  $\delta$  postignut u razumnom vremenu izvršavanja je blizu 1.012.

## 2.2 Problem najbližeg vektora

Problem najbližeg vektora ili **CVP** (eng. *Closest Vector Problem*) još se naziva i problem najbliže točke. Neformalno, za danu bazu rešetke  $\mathbf{B}$  i vektor  $\mathbf{t}$ , želimo pronaći  $\mathbf{v} \in L(\mathbf{B})$  najbliži zadanom vektoru  $\mathbf{t}$ . Zadani vektor ne mora nužno biti element rešetke.

**Definicija 2.2.1** (Problem najbližeg vektora). *Za danu bazu  $\mathbf{B}$  rešetke i zadani vektor  $\mathbf{t}$  (ne nužno element rešetke) treba pronaći  $\mathbf{v} \in L(\mathbf{B})$  za koji vrijedi*

$$\|\mathbf{v} - \mathbf{t}\| = \min_{\mathbf{y} \in L(\mathbf{B})} \|\mathbf{y} - \mathbf{t}\|$$

**Definicija 2.2.2** (Problem aproksimacije najbližeg vektora). *Za danu bazu  $\mathbf{B}$  rešetke, faktor aproksimacije  $\gamma$  i zadani vektor  $\mathbf{t}$  (ne nužno element rešetke) treba pronaći  $\mathbf{v} \in L(\mathbf{B})$  za koji vrijedi*

$$\|\mathbf{v} - \mathbf{t}\| \leq \gamma \min_{\mathbf{y} \in L(\mathbf{B})} \|\mathbf{y} - \mathbf{t}\|.$$

Problem najbližeg vektora je NP-težak. Međutim, postoji algoritam polinomne vremenske složenosti koji rješava problem aproksimacije najbližeg vektora.

## 2.3 Problem najmanje baze

Problem najmanje baze ili **SBP** (eng. *Smallest Basis Problem*) ima više varijanti ovisno o značenju riječi "najmanje".

Jedna varijanta problema neformalno glasi: pronađi bazu rešetke koja minimizira maksimume duljina svojih elemenata.

**Definicija 2.3.1.** *Za danu rešetku  $L(\mathbf{B})$  treba pronaći bazu  $a_1, \dots, a_n$  čija je duljina, definirana kao  $\max_{i=1}^n \|a_i\|$ , najmanja moguća do na polinomni faktor.*

Geometrizirana varijanta problema traži minimizaciju produkta duljina, budući je produkt uvijek veći od volumena rešetke, a jednak ako i samo ako je baza rešetke ortogonalna.

**Definicija 2.3.2.** *Za dane linearno nezavisne vektore  $a_1, \dots, a_n \in \mathbb{Q}^n$  treba pronaći bazu  $b_1, \dots, b_n$  rešetke  $L(a_1, \dots, a_n)$  takvu da je produkt  $\|b_1\| \cdot \dots \cdot \|b_n\|$  minimalan.*

## 2.4 Problem učenja s greškama

Problem učenja s greškama ili **LWE** (eng. *Learning With Errors*) temelji se na sustavu jednadžbi sa namjerno ubačenom greškom. Upravo ta generirana greška problem

rješavanja sustava čini izrazito teškim. Postoji nekoliko varijanti problema od kojih izdvajamo modularni LWE.

Modularni LWE može se svesti na jednadžbu:

$$B = As + e \pmod{q}$$

pri čemu su  $A$  i  $B$  javni ključevi,  $s$  tajni ključ,  $e$  ubačena greška i  $q$  proizvoljni veliki prosti broj.

# Poglavlje 3

## Algoritmi bazirani na rešetkama

Rešetke čine bazu mnogih algoritama za koje se smatra da su otporni na napade realizirane na kvantnim računalima. Sustavi bazirani na njima imaju snažne dokaze sigurnosti, a izrazito jednostavnu implementaciju. U ovom poglavlju istražiti ćemo neke od takvih algoritama, njihovu implementaciju, sigurnost i efikasnost.

### 3.1 Ajtai-Dwork kriptosustav

Svaku instancu kriptosustava treba biti jednako teško riješiti kao i uSVP u najgorem slučaju. Upravo to je temelj Ajtai-Dwork kriptosustava [2]. Nadalje, zbog činjenice da zasada ne postoji kvantni algoritam za redukciju rešetki, ovaj kriptosustav je zanimljiv s teorijske točke gledišta. Nažalost, originalni predložak kriptosustava je izuzetno neučinkovit.

#### Opis sustava

Neka je  $n \in \mathbb{N}$ . Zovemo ga sigurnosni parametar jer određuje dimenziju vektorskog prostora koji koristimo za rešetku kao i za preciznost binarne ekspanzije realnih brojeva. Za dani  $n$  definiramo  $m := n^3$  i  $\rho_n := 2^{n \log_2^n} = n^n$ .

Uvodimo sljedeću notaciju:

Neka je  $\alpha < \frac{1}{2}$ . Sa  $\mathbb{Z}_{\pm\alpha}$  označavamo skup

$$\bigcup_{z \in \mathbb{Z}} [z - \alpha, z + \alpha].$$

**Definicija 3.1.1.** Veliku  $n$ -dimenzionalnu kocku sa stranicama duljine  $\rho_n$  definiramo kao

$$B_n := \left\{ x \in \mathbb{R}^n \mid \forall i \in \{1, \dots, n\} : -\frac{\rho_n}{2} \leq x_i \leq \frac{\rho_n}{2} \right\}$$

a malu  $n$ -dimenzionalnu kuglu radijusa  $n^{-c}$  kao

$$S_n := \left\{ x \in \mathbb{R}^n \mid \|x\| \leq n^{-c} \right\}$$

gdje je  $c > 0$  cijeli broj.

**Definicija 3.1.2.** Neka su  $\mathbf{a}, \mathbf{b} \in \mathbb{R}^n$  dva vektora. Definiramo skalarni produkt, u oznaci  $\langle \mathbf{a}, \mathbf{b} \rangle$ , kao:

$$\langle \mathbf{a}, \mathbf{b} \rangle = \sum_{i=1}^n \mathbf{a}_i \cdot \mathbf{b}_i.$$

### Privatni ključ

Privatni ključ sustava je uniformno odabrani vektor  $\mathbf{u}$  iz  $n$ -dimenzionalne jedinične kugle

$$U_n := \left\{ x \in \mathbb{R}^n \mid \|x\| < 1 \right\}.$$

### Javni ključ

Za dani privatni ključ  $\mathbf{u}$   $\mathcal{H}_u$  označava distribuciju točaka u  $B_n$  induciranu sljedećom konstrukcijom:

- Uniformno odaberi točku  $a \in \{x \in B_n \mid \langle x, u \rangle \in \mathbb{Z}\}$ ,
- Uniformno odaberi  $\delta_1, \dots, \delta_n$  iz  $S_n$ ,
- Izlaz je točka  $v = a + \sum_{i=1}^n \delta_i$ .

Dakle,  $\mathcal{H}_u$  možemo promatrati kao skup  $\{a + \sum_{i=1}^n \delta_i \mid \langle a, u \rangle \in \mathbb{Z}, \delta_i \in S_n\}$ . Javni ključ sada dobivamo nezavisnim odabirom točaka  $(w_1, \dots, w_n)$  i  $(v_1, \dots, v_m)$  iz distribucije  $\mathcal{H}_u$  sa ograničenjem da je širina paralelopipeda  $\mathcal{P}(w_1, \dots, w_n)$  barem  $n^{-2}\rho_n$ .

### Šifriranje

Šifriranje ovisi o bitu koji se trenutno šifrira. Ukoliko je to 0, uniformno se odabere  $b_1, \dots, b_m \in \{0, 1\}$  i reducira vektor  $\sum_{i=1}^m b_i \cdot v_i$  modulo paralelopiped  $\mathcal{P}(W)$ . Za šifriranje bita 1, kao šifrat nasumično odaberemo vektor iz paralelopipeda  $\mathcal{P}(W)$ .



## Dešifriranje

Dešifriranje također ovisi o bitu. Za dešifriranje komponente  $c_i$  poruke  $c = (c_1, \dots, c_k)$  sa privatnim ključem  $\mathbf{u}$ , prvo računamo  $\tau := \langle c_i, \mathbf{u} \rangle$ . Ukoliko je  $\tau \in \mathbb{Z}_{p^{mn}-1}$ , tada  $c_i$  dešifriramo kao 0, inače kao 1.

**Lema 3.1.3.** Šifrat bita 0 uvijek će se dešifrirati kao 0. Šifrat bita 1 ima vjerojatnost  $2 \cdot n^{-1}$  da se dešifrira kao 0.

## Eliminacija grešaka dešifriranja

1997. godine Goldreich, Goldwasser i Halevi objavljuju modificiranu verziju Ajtai-Dwork kriptosustava koja eliminira greške dešifriranja [7]. Također pokazuju kako modifikacije nemaju utjecaja na sigurnost sustava.

Kao i u originalnoj shemi, modificirana preslikava 0 u vektor  $\mathbf{x}$  takav da  $\langle \mathbf{x}, \mathbf{u} \rangle$  bude blizu cijeloga broja. Međutim, modificirana shema osigurava da šifrirani vektor jedinice bude daleko od cjelobrojnih vrijednosti. Drugim riječima, šifrat jedinice može se raspoznati od šifrata nule.

Parametri  $n, m, \rho_n$ , skupovi  $S_n, B_n$  te vektori  $u, w_1, \dots, w_n, v_q, \dots, v_m$  odabrani su na isti način kao u originalnom sustavu.

### Javni ključ (modifikacija)

Uz originalni javni ključ  $(\mathbf{w}, \mathbf{v})$ , uniformno se odabire indeks  $i_1$  iz skupa indeksa koji zadovoljavaju sljedeći uvjet:

$$\mathbf{v}_i = \mathbf{a}_i + \sum_{j=1}^n \delta_j \text{ takav da } \langle \mathbf{a}_i, \mathbf{u} \rangle \in 2\mathbb{Z} + 1, \delta_j \in S_n,$$

gdje  $2\mathbb{Z} + 1$  označava neparne cijele brojeve. Takav indeks postoji sa vjerojatnošću  $1 - 2^{-m}$ . Javni ključ je sada  $(\mathbf{w}, \mathbf{v}, i_1)$ .

### Šifriranje (modifikacija)

Šifriranje bita 0 izvodi se kao i u originalnoj metodi, uniformnim odabirom  $b_1, \dots, b_m \in \{0, 1\}$  i reduciranjem vektora  $\sum_{i=1}^m b_i \cdot v_i$  modulo paralelopiped  $\mathcal{P}(W)$ . Vektor

$$\sum_{i=1}^m b_i \cdot v_i \pmod{\mathcal{P}(W)}$$

je šifrat bita 0.

Razlika je u šifriranju bita 1. U ovom slučaju uniformno biramo  $b_1, \dots, b_m \in \{0, 1\}$  te se vektor  $\frac{1}{2}\mathbf{v}_{i_1} + \sum_{i=1}^m b_i \cdot v_i$  reducira modulo paralelopiped. Vektor

$$\mathbf{c}_i = \frac{1}{2}\mathbf{v}_{i_1} + \sum_{i=1}^m b_i \cdot v_i \quad \text{mod } \mathcal{P}(W)$$

odgovara šifratu bita 1.

### Dešifriranje (modifikacija)

Dešifriranje vektora  $\mathbf{c}_i$  koji odgovara vrijednosti bita  $\sigma \in \{0, 1\}$  vrši se na sljedeći način:

$$\sigma = \begin{cases} 0 & \langle \mathbf{c}_i, \mathbf{u} \rangle \in \mathbb{Z}_{\pm 1/4}, \\ 1 & \text{inače.} \end{cases}$$

Modificirana verzija osigurava da šifriranje jedinica zadovoljava  $\langle \mathbf{c}_i, \mathbf{u} \rangle \in \mathbb{Z} + \frac{1}{2} \pm \frac{1}{n}$ .

**Propozicija 3.1.4** (Dešifriranje bez grešaka). *Neka je  $c=8$ . Za svaki  $\sigma \in \{0, 1\}$ , svaki odabir privatnog i javnog ključa te svaki odabir  $b_i$ -ova algoritmom šifriranja, šifrat  $\mathbf{x}$  zadovoljava*

$$\langle \mathbf{x}, \mathbf{u} \rangle \in \mathbb{Z} + \frac{\sigma}{2} \pm \frac{1}{n^3}.$$

*Posebno, dešifriranje je bez grešaka za dimenzije  $n \geq 2$ .*

### Vremenska i prostorna složenost

Prisjetimo se kako  $n$  predstavlja dimenziju rešetke i preciznost binarne ekspanzije.

Privatni ključ je nasumično odabrani vektor  $n$ -dimenzionalne jedinične kugle sa preciznošću od  $n$  bitova. To nam ukupno daje  $n^2$  bitova.

Za javni ključ, prvo računamo prostor potreban za binarnu reprezentaciju elementa  $v \in \mathcal{H}_u$ . Ovaj  $n$ -dimenzionalni vektor je oblika  $v = a + \sum_{i=1}^n \delta_i$ . Stoga su komponente vektora ograničene sa  $-\rho_n/2 - n \cdot n^{-c} \leq |v_i| \leq \rho_n/2 + n \cdot n^{-c}$ , što znači da svaka komponenta vektora treba  $\log_2 \rho_n = \log_2 n^n = n \cdot \log_2 n$  bitova plus  $n$  bitova za binarnu ekspanziju. Ukupno  $n$ -dimenzionalni vektor  $v \in \mathcal{H}_u$  treba  $n^2 \cdot (1 + \log_2 n)$  bitova. Javni ključ  $(w_1, \dots, w_n, v_1, \dots, v_{n^3})$  se sastoji od  $n+n^3$  vektora iz  $\mathcal{H}_u$ . Dakle zauzima  $(n+n^3) \cdot (n^2 \cdot (1 + \log_2 n)) = (n^3 + n^5) \cdot (1 + \log_2 n)$  bitova.

Šifriranje jednog bita je vektor iz paralelopipeda  $\mathcal{P}(w_1, \dots, w_n)$ , koji je ograničen sa:  $\forall i \in$

$\{1, \dots, n\} : v_i \leq \rho_n$ . Ukupno zauzima  $n^2 \cdot (1 + \log_2 n)$  bitova.

Privatni ključ može biti generiran u linearnom vremenu ukoliko pretpostavimo da se slučajne vrijednosti mogu odabrati iz izračunatog skupa slučajnih vrijednosti u konstantnom vremenu. Za javni ključ trebamo generirati  $n^3$  elemenata iz distribucije  $\mathcal{H}_u$  koja sama zahtijeva  $O(n^2)$  operacija. Dakle cijeli korak ima vremensku složenost  $O(n^3 \cdot n^2) = O(n^5)$ . Šifriranje se sastoji od zbrajanja  $n^3$  vektora i modularne operacije. Vremenska složenost navedenoga je  $O(n^4)$ . Dešifriranje zahtijeva skalarni produkt dva vektora, odnosno  $n$  množenja i  $n$  zbrajanja. Dakle vremenska složenost je  $O(n^2)$ .

Prostorna složenost		Vremenska složenost	
Objekt	Složenost	Objekt	Složenost
Privatni ključ	$O(n^2)$	Generiranje privatnog ključa	$O(n)$
Javni ključ	$O(n^5)$	Generiranje javnog ključa	$O(n^5)$
Šifrat	$O(n^2 \log n)$	Šifriranje	$O(n^4)$
		Dešifriranje	$O(n^2)$

Tablica 3.1: Složenost Ajtai-Dwork kriptosustava

## Zaključno

Ajtai-Dwork kriptosustav koristi činjenicu da ukoliko se navedeni kriptosustav riješi u polinomnom vremenu, tada se mogu riješiti i sljedeći poznati problemi nad rešetkama: aproksimacija najkraćeg vektora, pronalaženje jedinstvenog najkraćeg vektora i pronalaženje najmanje baze rešetke do na polinomni faktor. Nažalost, sustav nije praktičan u dimenzijama u kojima je navedena tri problema neizvedivo riješiti. Njegova vremenska i prostorna složenost čine ga neefikasnim u velikim dimenzijama. Ipak, izum Ajtai-Dwork kriptosustava bila je velika prekretnica u proučavanju novih polja za kriptografiju, pogotovo onih temeljenih na rešetkama.

## 3.2 Goldreich-Goldwasser-Halevi kriptosustav

Nadahnuti rezultatima Ajtaija, Oded Goldreich, Shafi Goldwasser i Shai Halevi 1997. godine objavljuju kriptosustav temeljen na CVP problemu poznat pod imenom GGH kriptosustav [8].

## Opis sustava

Kriptosustav ovisi o dva (javna) parametra: dimenziji  $n \in \mathbb{N}$  i sigurnosnom parametru  $\sigma \in \mathbb{R}$ . Generiranje ključeva događa se na sljedeći način:

1. Bob na slučajan način bira bazu  $\mathbf{R}$  kratkih vektora  $n$ -dimenzionalne rešetke. Ta baza je privatni ključ.
2. Bob generira drugu bazu  $\mathbf{B}$  iste rešetke, ali sa ne tako kratkim vektorima kao  $\mathbf{R}$ .
3. Bob objavljuje bazu  $\mathbf{B}$  kao javni ključ.

## Šifriranje

Alice kodira svoju poruku kao element  $m \in \mathbb{Z}^n$  i na slučajan način bira vektor pogreške  $e \in \{\pm\sigma\}^n$  te šalje sljedeći šifrat Bobu:

$$c = \mathbf{B} \cdot m + e.$$

## Dešifriranje

Koristeći privatnu bazu, pomoću Babaijevog algoritma dobijemo vektor rešetke najbliži vektoru  $c$ . Taj vektor je zapravo  $c - p$ .

Polaznu poruku dobijemo tako da vektor  $c - p$  slijeva pomnožimo sa  $\mathbf{B}^{-1}$ .

## Analiza grešaka i odabira $\sigma$

Postoje ograničenja prilikom odabira parametara jer oni utječu na greške nastale prilikom dekripcije. Može se pokazati da elementi vektora pogreške moraju biti elementi  $[-\sigma, \sigma]$  gdje je  $\sigma \in \mathbb{R}$ . Označimo sa  $\lfloor a \rfloor$  najbliži cijeli broj broju  $a$ .

**Lema 3.2.1.** *Neka je  $\mathbf{R}$  privatna baza. Greška prilikom dekripcije nastaje ako i samo ako vrijedi  $\lfloor \mathbf{R}^{-1}e \rfloor \neq 0$ .*

**Teorem 3.2.2.** *Neka je  $\mathbf{R}$  privatna baza te neka je  $\rho$  maksimum  $L_1$ -norme redaka u  $\mathbf{R}^{-1}$ . Tada ne dolazi do pogrešaka prilikom dešifriranja ukoliko vrijedi  $\sigma < 1/(2\rho)$ .*

## Vremenska i prostorna složenost

Budući da GGH kriptosustav nije potpuno određen teško je dati točnu analizu složenosti. Otvorena pitanja kriptosustava su:

- Kako odabrati R?
- Kako generirati B iz R?
- Kako odabrati vektor m?

Pri analizi ćemo koristiti sljedeće vrijednosti. Privatni ključ je dobra baza rešetke konstruirana kao  $R = k \cdot I_n + Q$ , gdje je  $k = \lfloor \sqrt{n} \cdot l \rfloor$  a Q proizvoljna perturbacija matrice sa elementima iz  $\{-l, \dots, l\}$  za  $l = 4$ . Zadana matrica ima prostornu složenost  $O(n^2 \log_2 n)$ . Javni ključ je "loša" baza iste rešetke dobivena množenjem R unimodularnom matricom. Elementi su obično većeg reda od n pa ova matrica zauzima  $O(n^3 \log n)$  prostora. Šifrat kao produkt poruke i javnog ključa zauzima  $O(n^2 \log n)$  prostora.

Minimalni trošak kreiranja privatnog ključa zahtijeva generiranje  $n^2$  cijelih brojeva. Pretpostavljamo da je ovo izvedivo u linearnom vremenu ukoliko uzimamo brojeve iz već slučajno generiranog skupa cijelih brojeva. Ovaj korak stoga zahtijeva  $O(n^2)$  vremena. Međutim, ukoliko reduciramo i invertiramo bazu radi ubrzanja dešifriranja, ovaj korak ima vremensku složenost  $O(n^3)$ .

Izračunavanje javnog ključa ovisi o odabranoj metodi. Pretpostavljamo da zahtijeva barem množenje dviju matrica (R sa unimodularnom matricom) pa za donju ogradu vremenske složenosti imamo  $O(n^3)$ .

Šifriranje je množenje matrice i vektora te ima vremensku složenost  $O(n^2)$ . Dešifriranje je primjena Babaijevog algoritma na prethodni produkt što je zapravo množenje matrice i vektora. Ovo ponovno zahtijeva  $O(n^2)$  vremena. Ovdje pretpostavljamo da je inverz izračunat. U suprotnome dešifriranje zahtijeva  $O(n^3)$  vremena.

Prostorna složenost		Vremenska složenost	
Objekt	Složenost	Objekt	Složenost
Privatni ključ	$O(n^2 \log n)$	Generiranje privatnog ključa	$O(n^2)$
Javni ključ	$O(n^3 \log n)$	Generiranje javnog ključa	$\geq O(n^3)$
Šifrat	$O(n^2 \log n)$	Šifriranje	$O(n^2)$
		Dešifriranje	$O(n^2)$

Tablica 3.2: Složenost GGH kriptosustava

## Zaključno

GGH kriptosustav ima jednu ključnu manu: način na koji se bira vektor pogreške može se iskoristiti za napad na kriptosustav za dimenzije manje od 400. Vektor pogreške se lako

generira neranjivim na ovakve napade ukoliko dozvolimo vrijednosti između  $-\sigma$  i  $\sigma$ . Ovo dovodi do slabijeg CVP problema. Međutim, i ovaj problem se lako razriješi dozvoljavanjem većih vrijednost  $\sigma$  što drastično povećava težinu CVP-a. Sustav više ne bi bio bez greške, ali pri velikim dimenzijama (već oko 400) stopa pogrešaka je toliko mala da se lako mogu ukloniti ili bar detektirati pomoću kodova za ispravljanje grešaka.

Uspoređujući GGH kriptosustav sa Ajtai-Dwork sustavom, možemo primijetiti veliki napredak u složenosti. Javni ključ je gotovo dva reda manji, a isto vrijedi i za vremenske složenosti generiranja javnog ključa, šifriranje i dešifriranje. Jedini aspekt u kojem se GGH čini lošijim je generiranje privatnog ključa.

### 3.3 NTRU

NTRU<sup>1</sup> [11] je prvi puta predstavljen 1996. godine te je bio među prvim poznatim kriptosustavima baziranim na rešetkama. Iako su se kroz godine pojavile razne varijante NTRU sustava, bazna ideja ostaje ista i prisutna je u svima varijacijama.

U ovom odjeljku promatramo  $R := \mathbb{Z}[X]/(X^N - 1)$ , gdje je  $N \in \mathbb{Z}$  sigurnosni parametar. Drugim riječima, promatramo prsten polinoma s cjelobrojnim koeficijentima nad varijablom  $X$  modulo polinom  $X^N - 1$ . Primijetimo da je maksimalni stupanj polinoma  $N - 1$  budući da imamo jednadžbu

$$X^N = 1 \in R.$$

Prsten ima klasično zbrajanje po komponentama:

Neka su  $a(X) = \sum_{i=0}^{N-1} a_i \cdot X^i$  i  $b(X) = \sum_{i=0}^{N-1} b_i \cdot X^i \in R$ , tada je

$$a(X) + b(X) = \sum_{i=0}^{N-1} (a_i + b_i) \cdot X^i,$$

i konvolucijsko množenje, definirano kao  $c(X) := a(X) \otimes b(X)$ , gdje je

$$c_k := \sum_{i=0}^k a_i \cdot b_{k-i} + \sum_{i=k+1}^{N-1} a_i \cdot b_{N+k-i} = \sum_{i+j \equiv k \pmod{N}} a_i \cdot b_j.$$

**Napomena 3.3.1.** *Konvolucijsko množenje može se opisati kao matrična operacija. Prvo definiramo cikličnu rotaciju  $C$  koja transformira vektor  $(x_1, \dots, x_n)$  u  $(x_n, x_1, \dots, x_{n-1})$ . Tako*

<sup>1</sup>engl. *N-th degree TRUncated polynomial ring.*

za  $x \in \mathbb{R}^n$  imamo cirkularnu matricu definiranu kao

$$[C^*x] := [x, Cx, \dots, C^{n-1}x] = \begin{bmatrix} x_1 & x_n & \cdots & x_2 \\ x_2 & x_1 & \cdots & x_3 \\ \vdots & \vdots & \ddots & \vdots \\ x_n & x_{n-1} & \cdots & x_1 \end{bmatrix}$$

Sada imamo da je  $f \otimes g$  ekvivalentno matricnoj operaciji  $[C^*f]g$  ako se  $f$  i  $g$  sastoje od koeficijenata od  $f$  i  $g$ . Operacija je asocijativna, komutativna i distributivna, dakle  $(R, +, \otimes)$  čini prsten. Nadalje, operacija zadovoljava svojstva:

$$\begin{aligned} [C^*f](g_1 + g_2) &= [C^*f]g_1 + [C^*f]g_2, \\ [C^*f][C^*g] &= [C^*([C^*f]g)], \\ [C^*(Cf)]g &= [C^*f](Cg). \end{aligned}$$

**Definicija 3.3.2.** Kada reduciramo polinom  $p(X)$  modulo  $q$ , zapravo reduciramo koeficijente od  $p(X)$  modulo  $q$  tako da je rezultat  $\tilde{p}(X) \in R/(q) = \mathbb{Z}[X]/(X^N - 1, q)$ .

## Opis sustava

Slijedi opis originalnog kriptosustava. Pretpostavljamo da Alice želi poslati poruku Bobu. Kao i u ostalim kriptosustavima sa javnim ključem, Bob generira privatni i javni ključ, Alice zatim šifrira poruku javnim ključem te naposljetku Bob dešifrira poruku koristeći svoj privatni ključ.

NTRU kriptosustav ima tri cjelobrojna parametra  $(N, p$  i  $q)$  za kontrolu sigurnosti i praktičnosti sustava. Pri tome  $p$  i  $q$  moraju biti relativno prosti.

Cjelobrojni koeficijenti  $d_f$ ,  $d_g$  i  $d_r$  daju ograde na koeficijente polinoma  $f$ ,  $g$  i  $r$  u  $R$  i kontroliraju vjerojatnost pogrešaka pri dešifriranju.

Definiramo sljedeću oznaku:

$$\mathcal{L}(d_1, d_2) := \{p(X) \in R \mid p \text{ ima } d_1 \text{ jedinicu, } d_2 \text{ negativne jedinice, a preostali koeficijenti su nula}\}$$

i u skladu s tim prostore:

$$\mathcal{L}_f := \mathcal{L}(d_f, d_f - 1), \quad \mathcal{L}_g := \mathcal{L}(d_g, d_g), \quad \mathcal{L}_r := \mathcal{L}(d_r, d_r).$$

Primijetimo kako, za razliku od  $\mathcal{L}_g$  i  $\mathcal{L}_r$ ,  $\mathcal{L}_f$  ima jednu negativnu jedinicu manje od pozitivnih. Razlog tomu je što  $f \in \mathcal{L}_f$  mora biti invertibilan. Kada bi imao isti broj, vrijedilo bi  $f(1) = 0$ , odnosno  $f$  ne bi bio invertibilan.

## Generiranje ključeva

Za kreiranje privatnog i javnog ključa, Bob slučajno odabire polinome  $f \in \mathcal{L}_f$  i  $g \in \mathcal{L}_g$ . Polinom  $f$  treba biti invertibilan modulo  $p$  i  $q$  te te inverze označimo sa  $f_p^{-1}$  i  $f_q^{-1}$  redom. Stoga imamo:

$$\begin{aligned} f_p^{-1} \otimes f &= 1 \pmod{p}, \\ f_q^{-1} \otimes f &= 1 \pmod{q}. \end{aligned}$$

Zatim Bob računa javni ključ

$$h := p \cdot f_q^{-1} \otimes g \pmod{q}$$

ili u matričnoj notaciji

$$\mathbf{h} := p \cdot [C^* \mathbf{f}_q^{-1}] \mathbf{g} \pmod{q},$$

a  $f$  i  $g$  ostaju privatni.

## Šifriranje

Neka je  $\mathbb{Z}_q$  prsten cijelih brojeva modulo  $q$ . Za šifriranje poruke  $\mathbf{m} \in \mathbb{Z}_q^n$  (označimo odgovarajući polinom poruke sa  $m$ ) Alice na slučajan način odabire polinom  $r \in \mathcal{L}_r$  te računa

$$c := h \otimes r + m \pmod{q}$$

ili u matričnoj notaciji

$$\mathbf{c} := [C^* \mathbf{h}] \mathbf{r} + \mathbf{m} \pmod{q}.$$

## Dešifriranje

Bob, prilikom dešifriranja poruke  $c$ , prvo reducira ulazni vektor

$$t \equiv f \otimes c \pmod{q},$$

tako da su sve vrijednosti  $t_i$  ograničene sa  $-\frac{q}{2} \leq t_i \leq \frac{q}{2}$ . Originalna poruka sada se lako dobije kao

$$m \equiv f_p^{-1} \otimes c \pmod{q}$$



ili ekvivalentno u matričnoj notaciji

$$\mathbf{t} \equiv [C^* \mathbf{f}] \mathbf{c} \pmod{q},$$

$$\mathbf{m} \equiv [C^* \mathbf{f}]_p^{-1} \mathbf{c} \pmod{q}.$$

Tablica 3.3 rezimira sve parametre NTRU sustava.

Parametar	Domena	Dostupnost
$N$	$\mathbb{N}$	javno
$p$	$\mathbb{N}$	javno
$q$	$\mathbb{N}$	javno
$d_f$	$\mathbb{N}$	javno
$d_g$	$\mathbb{N}$	javno
$d_r$	$\mathbb{N}$	javno
$f$	$\mathbb{Z}[X]/(X^N - 1)$	privatno
$g$	$\mathbb{Z}[X]/(X^N - 1)$	privatno
$h$	$\mathbb{Z}[X]/(X^N - 1)$	privatno

Tablica 3.3: Parametri NTRU kriptosustava

Neke od verzija NTRU sustava razlikuju se upravo po izboru navedenih parametara. Tablica 3.4 prikazuje tri takve verzije. Primijetimo kako je  $p$  uvijek 3 što čini invertiranje polinoma  $f$  mnogo jednostavnijim. Također,  $N$  je uvijek prost broj, a  $q$  potencija broja 2.

	$N$	$p$	$q$	$d_f$	$d_g$	$d_r$
NTRU167	167	3	128	61	20	18
NTRU263	263	3	128	50	24	16
NTRU503	503	3	256	216	72	55

Tablica 3.4: Parametri u vajantama NTRU kriptosustava

## Analiza grešaka

Prilikom procesa dešifriranja može se dogoditi da dešifrirana poruka ne odgovara u potpunosti originalnoj. Vjerojatnost pogreške ovisi o  $N$ ,  $d_f$ ,  $d_g$ ,  $d_r$  i  $m$ . Prvo pokažimo zašto

ovakva ideja dešifriranja funkcionira. Pogledajmo  $t = f \otimes c$ . Vrijedi:

$$\begin{aligned}
 t &= f \otimes c \\
 &= f \otimes (h \otimes r + m) \\
 &= f \otimes (h \otimes r) + f \otimes m \\
 &= f \otimes (p \cdot f_q^{-1} \otimes g) \otimes r + f \otimes m \\
 &\equiv p \cdot g \otimes r + f \otimes m \pmod{q}.
 \end{aligned} \tag{3.1}$$

Sljedeći korak je reducirati  $t$  tako da svi koeficijenti leže u intervalu  $[-\frac{q}{2}, \frac{q}{2}]$ . Ovisno o parametrima sustava, vrlo je vjerojatno kako će ovaj proces vratiti originalne koeficijente od  $p \cdot g \otimes r + f \otimes m$  u  $\mathbb{Z}[X]$ , iako je moguće da dešifriranje ne vrati originalnu poruku. Greške dešifriranja može koristiti Eva kako bi dobila informacije o privatnom ključu  $f$ .

Množeći  $t$  sa  $f_p^{-1}$  dobivamo:

$$\begin{aligned}
 f_p^{-1} \otimes t &= f_p^{-1}(p \cdot g \otimes r + f \otimes m) \\
 &= 0 + f_p^{-1} \otimes f \otimes m \\
 &\equiv m \pmod{p}.
 \end{aligned} \tag{3.2}$$

Zašto dolazi do pogrešaka?

Neka je  $\mathcal{P}$  skup svih polinoma stupnja najviše  $N - 1$ , a  $\mathcal{P}_q$  skup svih polinoma stupnja najviše  $N - 1$  kojima su koeficijenti ostaci modulo  $q$ . Ključna točka dešifriranja je (3.1). Polinom  $t$  treba biti jednak

$$p \cdot r \otimes g \otimes m \otimes f \in \mathcal{P}.$$

Jednakost modulo  $q$  ( $t \equiv p \cdot r \otimes g \otimes m \otimes f \in \mathcal{P}$ ) nije dovoljna. Pretpostavimo da je jednadžba istinita u  $\mathcal{P}_q$  ali nije u  $\mathcal{P}$ . Drugim riječima, postoji  $\mathbf{e} \in \mathbb{Z}^N$ , takav da

$$t = (p \cdot r \otimes g \otimes m \otimes f) + q \cdot \mathbf{e} \in \mathcal{P}.$$

Budući su  $p$  i  $q$  relativno prosti, izraz  $(q \cdot \mathbf{e})$  sa velikom vjerojatnošću nije 0 u  $\mathbb{Z}_p^N$ . Stoga generalno vrijedi:

$$\begin{aligned}
 f_p^{-1} \otimes t &= f_p^{-1}(p \cdot g \otimes r + f \otimes m) + f_p^{-1} \otimes (q \cdot \mathbf{e}) \\
 &= 0 + f_p^{-1} \otimes f \otimes m + f_p^{-1} \otimes (q \cdot \mathbf{e}) \\
 &= m + f_p^{-1} \otimes (q \cdot \mathbf{e}) \\
 &\neq m \pmod{p}.
 \end{aligned}$$

## Vremenska i prostorna složenost

Privatni ključ kriptosustava se sastoji od dva polinoma  $f$  i  $g$ . Za optimiziranje performansi pohranjujemo i inverze polinoma  $f$  modulo  $p$  i  $q$ . Zaključno, privatni ključ treba četiri vektora duljine  $n$  sa vrijednostima manjim od  $q$ , odnosno  $O(n \log q)$  prostora. Budući je  $q$  najčešće slične veličine kao  $N$ , pojednostavljamo izraz na  $O(n \log n)$ . Javni ključ i šifrat također su vektori iste strukture pa trebaju  $O(n \log n)$  prostora.

Generiranje privatnog ključa uključuje odabir polinoma  $f$  i  $g$ , gdje  $f$  mora biti invertibilan, i računanje inverza  $f$  modulo  $p$  i  $q$ . Vremenski zahtjevna operacija u ovom slučaju je računanje inverza što se može postići u vremenu  $O(n^2)$ . Generiranje javnog ključa i šifriranje se sastoje samo od množenja polinoma što ima vremensku složenost  $O(n^2)$ . Dešifriranje zahtijeva dva množenja polinoma te stoga također traži  $O(n^2)$ .

Prostorna složenost		Vremenska složenost	
Objekt	Složenost	Objekt	Složenost
Privatni ključ	$O(n \log n)$	Generiranje privatnog ključa	$O(n^2)$
Javni ključ	$O(n \log n)$	Generiranje javnog ključa	$O(n^2)$
Šifrat	$O(n \log n)$	Šifriranje	$O(n^2)$
		Dešifriranje	$O(n^2)$

Tablica 3.5: Složenost NTRU kriptosustava

## Zaključno

NTRU kriptosustav dobra je zamjena za sustave bazirane na problemu faktorizacije ili diskretnog logaritma kao što su RSA i ECC. Veličina ključa istog je reda kao u RSA, a performanse su znatno bolje u odnosu na RSA iste kriptografske sigurnosti.

Vremenska složenost najbolja je od svih dosad navedenih kriptosustava, a prostorna je gotovo linearna u svim aspektima. Nadalje, napadi na kriptosustav praktični su samo u malim dimenzijama.

## 3.4 Usporedba algoritama

U ovom odjeljku rezimiramo složenosti i svojstva navedenih kriptografskih sustava baziranih na rešetkama. Vidljivo u tablici 3.6, GGH i NTRU sustavi imaju istu vremensku složenost generiranja privatnog ključa. Svi sustavi imaju vremensku složenost šifriranja i dešifriranja  $O(n^2)$  sa izuzetkom Ajtai-Dwork kriptosustava čije je vrijeme šifriranja  $O(n^4)$ . Veličine ključeva smanjuju se iz sustava u sustav.

	Ajtai-Dwork	GGH	NTRU
Generiranje privatnog ključa	$O(n)$	$O(n^2)$	$O(n^2)$
Generiranje javnog ključa	$O(n^5)$	$\geq O(n^3)$	$O(n^2)$
Šifriranje	$O(n^4)$	$O(n^2)$	$O(n^2)$
Dešifriranje	$O(n^2)$	$O(n^2)$	$O(n^2)$
Veličina privatnog ključa	$O(n^2)$	$O(n^2 \log n)$	$O(n \log n)$
Veličina javnog ključa	$O(n^5)$	$O(n^3 \log n)$	$O(n \log n)$
Veličina šifrata	$O(n^2 \log n)$	$O(n^2 \log n)$	$O(n \log n)$

Tablica 3.6: Rezime složenosti algoritama

Kao što možemo vidjeti, NTRU sustav ima najbolje ocjene. Međutim, Ajtai-Dwork kriptosustav je jedini koji u originalnoj verziji dolazi sa dokazom sigurnosti.

Izvršena su testiranja za navedene kriptosustave i vremena i memoriju potrebnu za njihovo uspješno izvršavanje u dimenzijama gdje najboljem napadu klasičnim računalom treba oko 10 godina da uspješno probije sustav. Rezultati su prikazani u tablici 3.7.

	Ajtai-Dwork	GGH	NTRU
Dimenzija	48	600	300
Generiranje privatnog ključa	10ms	100ms	50ms
Generiranje javnog ključa	2h	1min	5ms
Šifriranje	6min	50ms	0.5ms
Dešifriranje	6sec	1sec	5ms
Veličina privatnog ključa	250KB	25KB	3KB
Veličina javnog ključa	24MB	500KB	1KB
Veličina šifrata	200KB	20KB	1KB

Tablica 3.7: Usporedba vremena i prostora potrebnog za izvršavanje algoritama

### 3.5 NIST-ovi kandidati

NIST u zadnjim krugovima svog natjecanja ima nekoliko finalista koji se temelje na problemima vezanim uz rešetke. U ovom odjeljku navodimo neke od njih, komentirajući njihovu sigurnost prema NIST-ovim standardima.

## NTRU

NTRU kriptosustav iz prethodnog odjeljka jedan je od finalista koje ćemo analizirati. U ovom dijelu sagledat ćemo sustav NIST-ovim kriterijima.

Kandidati koji su se prijavili IND-CCA2 su sigurni budući da svoju sigurnost većinom temelje na prstenastom LWE problemu.

Promatramo dvije verzije kriptosustava:

- NTRU-HRSS verzija podrazumijeva da je svaki koeficijent polinoma uniformno odabran iz skupa  $\{-1, 0, 1\}$ .
- NTRU-HRP verzija zahtijeva da gledajući koeficijente polinoma, ukupan broj pozitivnih i negativnih jedinica bude fiksiran.

	Javni ključ	Tajni ključ	Šifrat	Razina sigurnosti
NTRU-HPS2048677	930	1234	930	1
NTRU-HRSS701	1138	1450	1138	1
NTRU-HPS4096821	1230	1590	1230	3
NTRU-HPS40961229	1842	2366	1842	5
NTRU-HRSS1373	2401	2983	2401	5

Tablica 3.8: Veličina objekata u varijacijama NTRU-sustava

## SABER

SABER je familija algoritama koji se temelje na *Module Learning With Rounding* problemu (Mod-LWR). Razlika sa LWE problemom leži u tome što se same pogreške u LWR dobivaju deterministički, zaokruživanjem poruke na modulo manji od onog korištenog u LWE. Problemi su iste sigurnosti, ali je LWR jednostavniji, učinkovitiji (ne treba uzrokovati pogreške distribucije) te robusniji (smanjuje se veličina ključeva i šifrata zaokruživanjem).

Postoje tri varijante SABER-a od kojih svaka odgovara određenoj razini sigurnosti. To su LightSABER, SABER i FireSABER.

SABER je IND-CPA siguran. IND-CCA razinu sigurnosti postiže pomoću Fujisaki-Okamoto transformacija.

	core-SVP (klasično)	core-SVP (kvantno)	Razina sigurnosti
LightSaber	$2^{118}$	$2^{107}$	1
Saber	$2^{189}$	$2^{172}$	3
FireSaber	$2^{260}$	$2^{236}$	5

Tablica 3.9: Sigurnost SABER sustava

## CRYSTALS-Kyber

Naposljetku navodimo algoritam kojeg je NIST 5. srpnja 2022. godine standardizirao. CRYSTALS-Kyber je skup algoritama za razmjenu ključeva. Svoju sigurnost postiže koristeći modularni LWE problem.

Ocjena performansi Kyber algoritma ovisi o performansama korištenih kriptografskih metoda. Razlog tome je što Kyber koristi brzu varijantu diskretne Fourierove transformacije u mnogim svojim koracima. Ova činjenica uzrokuje da je algoritam iznimno brz na računalima koja imaju sklopovsku podršku za izvedbu određenih kriptografskih funkcija. Uz sve navedeno, algoritam je i izrazito skalabilan budući da povećanje razine sigurnosti zahtijeva samo promjenu dimenzija matrice.

CRYSTALS-Kyber je IND-CCA2 siguran te postoje tri varijante algoritma: Kyber512, Kyber768 i Kyber1024. Svaka od navedenih predstavlja određenu razinu sigurnosti.

	core-SVP (klasično)	core-SVP (kvantno)	Razina sigurnosti
Kyber512	111	100	1
Kyber768	181	164	3
Kyber1024	254	230	5

Tablica 3.10: Sigurnost CRYSTALS-Kyber sustava

# Bibliografija

- [1] Miklós Ajtai, *Generating hard instances of the short basis problem*, International Colloquium on Automata, Languages, and Programming, Springer, 1999, str. 1–9.
- [2] Miklós Ajtai i Cynthia Dwork, *A public-key cryptosystem with worst-case/average-case equivalence*, Proceedings of the twenty-ninth annual ACM symposium on Theory of computing, 1997, str. 284–293.
- [3] Gorjan Alagic, Daniel Apon, David Cooper, Quynh Dang, Thinh Dang, John Kelsey, Jacob Lichtinger, Carl Miller, Dustin Moody, Rene Peralta et al., *Status report on the third round of the NIST post-quantum cryptography standardization process*, US Department of Commerce, NIST (2022).
- [4] Johannes Blömer i Jean Pierre Seifert, *On the complexity of computing short linearly independent vectors and short bases in a lattice*, Proceedings of the thirty-first annual ACM symposium on Theory of computing, 1999, str. 711–720.
- [5] Lily Chen, Lily Chen, Stephen Jordan, Yi Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner i Daniel Smith-Tone, *Report on post-quantum cryptography*, sv. 12, US Department of Commerce, National Institute of Standards and Technology . . . , 2016.
- [6] Andrej Dujella i Marcel Maretić, *Kriptografija*, Element, 2007.
- [7] Oded Goldreich, Shafi Goldwasser i Shai Halevi, *Eliminating decryption errors in the Ajtai-Dwork cryptosystem*, Annual International Cryptology Conference, Springer, 1997, str. 105–111.
- [8] ———, *Public-key cryptosystems from lattice reduction problems*, Annual International Cryptology Conference, Springer, 1997, str. 112–131.
- [9] Matthew Green, *EUF-CMA and SUF-CMA*, 2018, <https://blog.cryptographyengineering.com/euf-cma-and-suf-cma/>, posjećena 2022-12-10.

- [10] Michael Hartmann i Joachim Rosenthal, *The Ajtai-Dwork Cryptosystem and Other Cryptosystems Based on Lattices*, Universite de Zurich **29** (2015).
- [11] Jeffrey Hoffstein, Jill Pipher i Joseph H Silverman, *NTRU: A ring-based public key cryptosystem*, International algorithmic number theory symposium, Springer, 1998, str. 267–288.
- [12] Manish Kumar, *Post-Quantum Cryptography Algorithms Standardization and Performance Analysis*, arXiv preprint arXiv:2204.02571 (2022).
- [13] Daniele Micciancio, *Lattices Algorithms and Applications (Spring 2007)*, <https://cseweb.ucsd.edu/classes/sp07/cse206a/>, posjećena 2022-12-10.
- [14] Daniele Micciancio i Oded Regev, *Lattice-based cryptography*, Post-quantum cryptography, Springer, 2009, str. 147–191.
- [15] Phong Q Nguyen i Jacques Stern, *The two faces of lattices in cryptology*, International Cryptography and Lattices Conference, Springer, 2001, str. 146–180.
- [16] Oded Regev, *Lattice-based cryptography*, Annual International Cryptology Conference, Springer, 2006, str. 131–141.
- [17] Yang Wang i Mingqiang Wang, *Module-lwe versus ring-lwe, revisited*, Cryptology ePrint Archive (2019).



# Sažetak

Pojavom LLL algoritma za redukciju baze rešetke, one postaju bitan čimbenik u razvoju kriptografskih sustava. Kriptografija bazirana na rešetkama temelji se na složenosti problema rešetki, čija je osnova problem najkraćeg vektora. Ovdje nam je kao ulaz dana rešetka zadana svojom bazom, a cilj nam je pronaći najkraći vektor rešetke različit od nule. Ovakvi sustavi pružaju obećavajuće rješenje za post-quantnu kriptografiju jer se vjeruje da su upravo problemi nad rešetkama NP-teški čak i za kvantna računala.

Začetnikom ideje o kriptosustavima baziranim na rešetkama smatra se Ajtai-Dwork kriptosustav. Ima veliku teoretsku važnost te svojim idejama i rezultatima složenosti služi kao preteča mnogih drugih kriptosustava. Njegovim stopama idu sustavi poput GGH i NTRU sustava koji uz odlične rezultate složenosti imaju i bolju efikasnost.

Veliku većinu NIST-ovih finalista čine upravo sustavi temeljeni na problemima vezanim uz rešetke. Čak tri takva algoritma (CRYSTALS-Kyber, NTRU i SABER) proglašena su finalistima za mehanizme enkapsulacije ključa. Upravo CRYSTALS-Kyber je u srpnju 2022. godine i standardiziran od strane NIST-a.

Uzevši u obzir složenost problema rešetki i vremenske i prostorne složenosti algoritama baziranih na njoj, da se naslutiti da će uloga rešetki u kriptografiji samo dobivati na značajnosti. Možemo pretpostaviti da u rešetkama leži naša sigurna budućnost, barem dok se ne dokaže suprotno.

# Summary

With the emergence of the LLL algorithm for lattice basis reduction, lattices became an important factor in the development of cryptographic systems. Lattice-based cryptography is based on the complexity of the lattice problem, the basis of which is the shortest vector problem. Here, we are given a lattice with its base as input, and our goal is to find the shortest non-zero lattice vector. Such systems provide a promising solution for post-quantum cryptography because it is believed that problems concerning lattices are NP-hard even for quantum computers.

The initial idea of lattice-based cryptosystems is considered to be introduced in the Ajtai-Dwork cryptosystem. It has great theoretical importance and with its ideas and complexity results serves as inspiration for many others. Following in its footsteps are systems like the GGH and NTRU, which, in addition to excellent complexity results, also have better efficiency.

The vast majority of NIST's finalists are systems based on lattice problems. As many as three such algorithms (CRYSTALS-Kyber, NTRU and SABER) were declared finalists for key encapsulation mechanisms. It was CRYSTALS-Kyber that was standardized by NIST in July of 2022.

Taking into account the complexity of the problem of lattices and the temporal and spatial complexity of algorithms based on it, one can guess that the role of lattices in cryptography will only gain more importance. It is reasonable to assume that the future of cybersecurity lies in the use of lattices, at least until proven otherwise.

# Životopis

Rođena sam 4. lipnja 1997. godine u Sisku. Pohađala sam Osnovnu školu Dragutina Tadijanovića u Petrinji, područni razredni odjel Mošćenica. Nakon toga upisujem Opću gimnaziju u Sisku. Ljubav prema matematici se razvila već u osnovnoj školi, pa nakon srednjoškolskog obrazovanja upisujem preddiplomski studij Matematike na Prirodoslovno-matematičkom fakultetu u Zagrebu. Tu započinje moje zanimanje prema računarstvu te završivši preddiplomski studij upisujem diplomski studij Računarstva i matematike također na Matematičkom odsjeku Prirodoslovno-matematičkog fakulteta u Zagrebu.