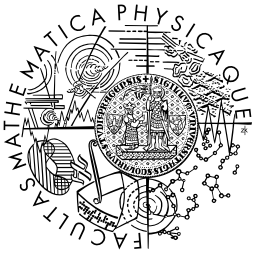

MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

DIPLOMOVÁ PRÁCE

Marek Zpěváček

Důkazy bezpečnosti hashovacích funkcí

Katedra algebry

Vedoucí diplomové práce: doc. Mgr. Pavel Příhoda, Ph.D.

Studijní program: Matematika

Studijní obor: Matematika pro informační technologie

Praha 2019

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V Praze dne 9. května 2019 Podpis autora

Děkuji doc. Pavlu Příhodovi za konzultace a pomoc s vypracováním této práce.

Název práce: Důkazy bezpečnosti hashovacích funkcí

Autor: Marek Zpěváček

Katedra: Katedra algebry

Vedoucí diplomové práce: doc. Mgr. Pavel Příhoda, Ph.D., Katedra algebry

Abstrakt: Tato práce se zaměřuje na důkaz redukce přibližného SBP na SIS. Důkaz provedl již Miklós Ajtai v roce 1996 ve své přelomové práci, avšak jeho důkaz je místy často nejasný a některé kroky nejsou dostatečně rozepsány. Redukce je typu nejhorší případ převeden na průměrný případ. Před zmíněnou prací Ajtaie nebyla známa žádná redukce takového typu. Proto nám přijde vhodné se k důkazu vrátit a rozepsat všechny jeho kroky do většího detailu. Dále je v práci shrnuta složitost základních problémů na mřížkách. Na základě těchto složitostí a dokázané redukce je možné definovat hashovací funkce odolné vůči kolizím. Na takové funkce se tato práce také zběžně zaměřuje.

Klíčová slova: mřížka, redukce, nejhorší případ, průměrný případ

Title: Proving security of hash functions

Author: Marek Zpěváček

Department: Departement of Algebra

Supervisor: doc. Mgr. Pavel Příhoda, Ph.D., Departement of Algebra

Abstract: This thesis focuses on proof of reduction from approximate SBP to SIS. The proof was already accomplished by Miklós Ajtai in 1996 in his groundbreaking work, however his proof lacks level of detail. The reduction is worst-case to average-case and no reduction of this type was known prior to the Ajtai's one. That is the reason why we found appropriate to return to the proof and provide it in more detailed form. Furthermore, the complexity of basic lattice problems is summarized. Based on these complexities and proven reduction, it is possible to define collision-resistant hash functions. This work is also briefly focused on such functions.

Keywords: lattice, reduction, worst-case, average-case

Obsah

Úvod

V roce 1996 přišel Miklós Ajtai s přelomovým důkazem polynomiální redukce hledání přibližně nejkratší báze v celočíselných mřížkách (SBP_f) na problém nazvaný problém krátkého řešení (SIS). Tato redukce je typu nejhorší případ převedený na průměrný případ a byla první redukcí takového typu. Zjednodušeně řečeno: umíme-li efektivně řešit náhodnou instanci problému SIS, potom také umíme efektivně řešit libovolnou instanci problému SBP_f. Spojením „efektivně řešit“ máme na mysli existenci algoritmu, který pracuje v polynomiálním čase ve velikosti vstupu. Velikostí vstupu v této práci máme na mysli vždy počet bitů vstupu. Všechny odhady složitosti tedy provádíme na nižší úrovni, než kdybychom za velikosti vstupu uvažovali například dimenzi.

O problému SBP (v přesné, neaproximační verzi) je dokázáno, že je NP-těžký, což je jedno z nejsilnějších tvrzení o tom, že instance tohoto problému nelze efektivně řešit. V aproximační verzi SBP_f už takový důkaz nemáme, avšak pořád máme silné důvody k předpokladu, že i tyto problémy nelze efektivně řešit.

Takový typ redukce má nepochybně využití v kryptografii. Potřebujeme navrhnout kryptosystém, který je založen na náhodné instanci problému SIS: když dokážeme takový kryptosystém efektivně prolomit, pak dostaneme řešení příslušné instance problému SIS. V případě existence takového útoku bychom z redukce SBP_f na SIS dostali i efektivní algoritmus na řešení SBP_f, o kterém předpokládáme, že je výpočetně náročný. Celkem tedy můžeme takový kryptosystém považovat za bezpečný.

Naprostá většina kryptosystémů, zakládajících bezpečnost na složitosti nějakého výpočetního problému [1], potřebuje pro bezpečnost specifické (nejtěžší) instance problémů [2]. Výhoda přístupu k důkazu bezpečnosti pomocí redukce z průměrného případu je, že kryptosystém je založený na náhodné instanci daného výpočetního problému. Typicky nám tak stačí vygenerovat několik náhodných bitů.

Ajtaiův důkaz postrádá úroveň detailu, spousta kroků v důkazech a odhadech není dostatečně rozepsána. Přínos této práce je zejména v detailním důkazu redukce a všech pomocných lemmat. Dále jsou v této práci určeny konkrétní konstanty v aproximačních faktorech, které v původním důkazu chybí.

V první kapitole si definujeme základní pojmy a představíme vybrané výpočetní problémy na mřížkách. Druhá kapitola se zabývá detailním důkazem redukce SBP_f na SIS. Nejdůležitější částí je věta 18 a celkově je redukce popsána ve větě 19. Ve třetí kapitole pak ukážeme, jak využít redukci a složitost výpočetních problémů k vytvoření bezpečné hashovací funkce.

[1] Některé kryptosystémy dokonce postrádají důkaz, že když je dokážeme prolomit, tak dokážeme řešit příslušný výpočetní problém. Opačná implikace je většinou zřejmá. Příklad může být RSA, kde chybí důkaz implikace „umíme prolomit RSA“$\Rightarrow$ umíme faktorizovat celá čísla.

[2] Jako příklad uvedeme opět RSA: tento kryptosystém potřebuje složené N ze dvou prvočísel: $N = p \cdot q$. Dále máme spoustu dalších požadavků na p a q, například $p-1$ i $q-1$ musí mít dostatečně velkého prvočíselného dělitele (tato podmínka je zde jako ochrana proti Pollardově $p-1$ metodě).

Značení

Nechť $x \in \mathbb{R}$, potom $\lfloor x \rceil$ značí běžné zaokrouhlení, neboli $\lfloor x \rceil = \{z \in \mathbb{Z}\colon |x - z| \text{ je minimální}\}$. Pro $z + \frac{1}{2}, z \in \mathbb{Z}$ definujme $\left\lfloor z + \frac{1}{2} \right\rceil = z + 1$. Dále $\lfloor x \rfloor$ značí zaokrouhlení dolů, tedy $\lfloor x \rfloor = \max\{z \in \mathbb{Z}\colon z \leq x\}$. Nakonec $\lceil x \rceil$ značí zaokrouhlení nahoru, tedy $\lceil x \rceil = \min\{z \in \mathbb{Z}\colon z \geq x\}$.

Vektory budeme značit tučně: $\boldsymbol{x} \in \mathbb{Z}^n$. Matice budeme značit velkým písmenem: $A \in \mathbb{Z}^{n \times m}$. Zápis $A = (a_{ij})$ znamená, že matice A má na pozici (i,j) prvek a_{ij}. Determinant matice A značíme $\det(A)$.

Nechť $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n, p \in \mathbb{N}$. Potom ℓ_p-normou vektoru $\boldsymbol{x}$ máme na mysli hodnotu $\|\boldsymbol{x}\|_p = (\sum_{i=1}^n |x_i|^p)^{\frac{1}{p}}$. Dále označme $\|\boldsymbol{x}\|_\infty = \max_{i=1,\ldots,n} |x_i|$. Symbolem $\|\boldsymbol{x}\|$ máme na mysli ℓ_2-normu vektoru $\boldsymbol{x}$, neboli $\|\boldsymbol{x}\| = \sqrt{\sum_{i=1}^n (x_i)^2}$.

Nechť $B = (b_{ij}) \in \mathbb{Z}^{n \times m}, \boldsymbol{x} = (x_1, \ldots, x_m) \in \mathbb{Z}^m, \boldsymbol{u} = (u_1, \ldots, u_m) \in \mathbb{Z}^n$ a $q \in \mathbb{N}$, potom zápisem $B\boldsymbol{x} \equiv \boldsymbol{u} \bmod q$ máme na mysli, že platí následující: $\sum_{j=1}^m b_{ij} x_j \equiv u_i \bmod q, \forall i = 1, \ldots, n$.

Nechť $A \subset \mathbb{R}^n$. Pak vnitřek množiny A budeme značit $\operatorname{int}(A)$.

Nechť $f, g\colon \mathbb{N} \to [0, \infty)$, potom píšeme $f = \mathcal{O}(g)$ pokud existuje $c \in \mathbb{N}$ takové, že pro všechna $n \in \mathbb{N}$ platí $f(n) \leq c \cdot g(n)$. O algoritmu $\mathcal{A}$ řekneme, že má polynomiální složitost, jestliže existuje $c \in \mathbb{N}$ takové, že pro libovolný vstup x délky n je k výpočtu $\mathcal{A}(x)$ zapotřebí vykonat $\mathcal{O}(n^c)$ jednotkových operací. V této práci považujeme za jednotkové operace sčítání a násobení jednociferných čísel (bez bližší specifikace báze).

1. Mřížky

1.1 Složitost algoritmů lineární algebry

Než se dostaneme k samotným mřížkám, zaměříme se na složitost některých algoritmů lineární algebry: řešení soustavy lineárních rovnic, výpočet ortogonální projekce na podprostor a výpočet Hermitovské normální formy. K tomu si zadefinujme velikost čísel z $\mathbb{Z}$ a $\mathbb{Q}$.

Definice 1. *Nechť $k \in \mathbb{Z} \setminus \{0\}$. Velikostí k máme na mysli počet bitů v binární reprezentaci k, neboli $\lceil \log_2 |k| \rceil + 1$. Značíme* $\operatorname{size}(k)$. *Pro $k = 0$ definujeme* $\operatorname{size}(0) = 1$.

Pro zlomek $\frac{a}{b} \in \mathbb{Q}$ v základním tvaru definujme $\operatorname{size}\left(\frac{a}{b}\right) = \operatorname{size}(a) + \operatorname{size}(b)$.

Nechť $A = (a_{ij}) \in \mathbb{Q}^{n \times m}$, pak definujeme $\operatorname{size}(A) = \sum_{i=1}^{n} \sum_{j=1}^{m} \operatorname{size}(a_{ij})$.

Následující poznámku budeme hojně využívat v odhadech složitosti, kde budeme ignorovat faktor n, například ve for cyklech.

Poznámka 1. Nechť p je polynom. Jelikož pro $\boldsymbol{v} \in \mathbb{Q}^n$ je $\operatorname{size}(\boldsymbol{v}) \geq n$, potom jakýkoliv algoritmus se složitostí polynomiální v $\operatorname{size}(\boldsymbol{v})$ má složitost polynomiální v $p(n) \cdot \operatorname{size}(\boldsymbol{v})$.

Prvně se zaměříme na řešení soustavy lineárních rovnic. Mějme soustavu n lineárních rovnic o n neznámých, neboli pro $A \in \mathbb{Z}^{n \times n}$ a $\boldsymbol{b} \in \mathbb{Z}^n$ hledáme $\boldsymbol{x} \in \mathbb{Z}^n$ takové, že $A\boldsymbol{x} = \boldsymbol{b}$. Takové $\boldsymbol{x}$ lze najít pomocí Gaussovy eliminace v čase $\mathcal{O}(n^3)$, kdy za jednotkovou operaci volíme násobení nebo sčítání v $\mathbb{Q}$, nehledě na velikost sčítaných či násobených čísel. Volíme-li ale za jednotkovou operaci sčítání a násobení jednociferných čísel v $\mathbb{Z}$, potom Gaussova eliminace nemusí mít polynomiální složitost v n. Označíme-li M maximum z absolutních hodnot prvků matice A a vektoru $\boldsymbol{b}$, potom Gaussova eliminace nemusí mít polynomiální složitost ani v $n \operatorname{size}(M)$, jak je uvedeno například v (Fang a Havas, 1997), a tedy ani v $\operatorname{size}(A)$. Důvodem je, že během maticových úprav může velikost koeficientů růst exponenciálně.

Na druhou stranu předpokládejme, že je matice A regulární. Potom z Cramerova pravidla pro i-tou složku vektoru $\boldsymbol{x}$ platí:

$$x_i = \frac{\det(A_i)}{\det(A)},$$

kde A_i je matice, která vznikne nahrazením i-tého sloupce matice A vektorem $\boldsymbol{b}$. Dále z Hadamardovy nerovnosti plyne

$$|\det(A)| \leq \prod_{i=1}^{n} \|\boldsymbol{a}_i\|.$$

Potom platí:

$$|\det(A)| \leq \prod_{i=1}^{n} \sqrt{nM^2} = \left(nM^2\right)^{\frac{n}{2}}$$

a tedy

$$\begin{aligned}\operatorname{size}(\det(A)) = &\left\lceil \log_2 \left(nM^2\right)^{\frac{n}{2}} \right\rceil + 1 = \left\lceil \frac{n}{2} \left(\log_2 n + 2\log_2 (M)\right) \right\rceil + 1 \\ \leq &\frac{n}{2} \left(\log_2 n + 2\operatorname{size}(M)\right) + 2.\end{aligned}$$

Z $\operatorname{size}(M) \leq \operatorname{size}(A)$ a z poznámky 1 platí, že $\operatorname{size}(\det(A))$ je polynomiální v size A. Obdobně platí, že $\operatorname{size}(\det(A_i))$ je polynomiální v $\operatorname{size}(A) + \operatorname{size} \boldsymbol{b}$. Celkem tedy máme, že $\operatorname{size}(\boldsymbol{x})$ je polynomiální v $\operatorname{size}(A) + \operatorname{size} \boldsymbol{b}$.

Na jednu stranu tedy platí, že pro regulární soustavu $A\boldsymbol{x} = \boldsymbol{b}, A \in \mathbb{Z}^{n\times n}, \boldsymbol{b} \in \mathbb{Z}^n$ je $\operatorname{size}(\boldsymbol{x})$ je polynomiální v $\operatorname{size}(A)+\operatorname{size} \boldsymbol{b}$, na druhou stranu během Gaussovy eliminace může velikost koeficientů růst exponenciálně, tedy nemáme zaručenou polynomiální složitost.

Existují jiné algoritmy, například algoritmus popsaný v Dixon (1982), který pracuje v polynomiálním čase v $\operatorname{size}(A) + \operatorname{size} \boldsymbol{b}$.

Dále nás bude zajímat případ, kdy je matice A typu $m \times n, m > n$ a soustava $A\boldsymbol{x} = \boldsymbol{b}$ má právě jedno řešení. V tomto případě budeme řešit soustavu $A^\top A\boldsymbol{x} = A^\top \boldsymbol{b}$. Dostáváme tak následující lemma.

Lemma 1. *Nechť $n, k \in \mathbb{N}, k \leq n, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k \in \mathbb{Z}^n$ jsou lineárně nezávislé vektory a nechť $\boldsymbol{b} \in \langle \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k \rangle, \boldsymbol{b} \in \mathbb{Z}^n$. Potom lze řešení $\boldsymbol{x} \in \mathbb{Q}^k$ soustavy $A\boldsymbol{x} = \boldsymbol{b}$ spočítat v polynomiálním čase v $\sum_{i=1}^k \operatorname{size} \boldsymbol{a}_i + \operatorname{size} \boldsymbol{b}$.*

Dále platí, že $\operatorname{size}(\boldsymbol{x})$ je polynomiální v $\sum_{i=1}^k \operatorname{size} \boldsymbol{a}_i + \operatorname{size} \boldsymbol{b}$.

Lemma 1 lze jednoduše použít na hledání inverzní matice. Chceme-li pro regulární matici A najít A^{-1}, potom i-tý sloupec A^{-1} získáme jako řešení soustavy $A \cdot \boldsymbol{x} = \boldsymbol{e}_i$, kde $\boldsymbol{e}_i$, je i-tý vektor kanonické báze.

Lemma 2. *Nechť $A \in \mathbb{Z}^{n\times n}$ je regulární matice. Potom lze spočítat inverzní matici A^{-1} k matici A v polynomiálním čase v $\operatorname{size}(A)$. Dále platí, že $\operatorname{size}(A^{-1})$ je polynomiální v $\operatorname{size}(A)$.*

Definice 2. *Celočíselnou čtvercovou matici $M \in \mathbb{Z}^{n\times n}$ nazveme unimodulární, pokud $\det(M) = \pm 1$.*

Dále budeme potřebovat následující lemma, že každou celočíselnou matici lze převést vynásobením unimodulární maticí na horní trojúhelníkovou. Splňuje-li horní trojúhelníková matice navíc podmínky, které pro nás nejsou důležité, potom se této úpravě říká Hermitovská normální forma. Tuto formu lze spočítat postupnou eliminací sloupců za pomoci Euklidova algoritmu. Avšak tato metoda, obdobně jako Gaussova eliminace, může produkovat čísla, jejichž velikost není polynomiální ve velikosti vstupu. V Micciancio a Warinschi (2001) je uvedena metoda, jak tyto dvě matice lze spočítat v polynomiálním čase vzhledem v velikosti matice A, složitost je diskutována v sekci 6 výše zmíněného článku.

Lemma 3. *Nechť $A \in \mathbb{Z}^{n\times n}$ je celočíselná čtvercová matice, potom existují unimodulární matice $U \in \mathbb{Z}^{n\times n}$ a horní trojúhelníková matice $T \in \mathbb{Z}^{n\times n}$ tak, že $T = U \cdot A$. Tyto matice lze spočítat v polynomiálním čase v $\operatorname{size}(A)$. Dále platí, že $\operatorname{size}(T)$ a $\operatorname{size}(U)$ je polynomiální v $\operatorname{size}(A)$.*

Dále se zaměříme na výpočet a složitost výpočtu ortogonální projekce. Mějme lineárně nezávislé vektory $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_k \in \mathbb{Z}^n$ a vektor $\boldsymbol{c} \in \mathbb{Z}^n$. Označme $B = \langle \boldsymbol{b}_1,\ldots,\boldsymbol{b}_k\rangle$. Chceme spočítat projekci vektoru $\boldsymbol{c}$ na podprostor B, neboli $\boldsymbol{c}_B \in B$ takové, že $\boldsymbol{c} - \boldsymbol{c}_B \perp B$. Protože $\boldsymbol{c}_B \in B$, existují $a_1,\ldots,a_k \in \mathbb{R}$ takové, že:

$$\boldsymbol{c}_B = a_1\boldsymbol{b}_1 + \cdots + a_k\boldsymbol{b}_k.$$

Dále z $\boldsymbol{c} - \boldsymbol{c}_B \perp B$ platí, že $\boldsymbol{c} - \boldsymbol{c}_B \perp \boldsymbol{b}_i, i = 1,\ldots,k$, a tedy:

$$\begin{aligned} 0 &= \langle \boldsymbol{c} - \boldsymbol{c}_B,\boldsymbol{b}_i\rangle = \langle \boldsymbol{c} - a_1\boldsymbol{b}_1 - \ldots - a_k\boldsymbol{b}_k,\boldsymbol{b}_i\rangle \\ &= \langle \boldsymbol{c},\boldsymbol{b}_i\rangle - a_1\langle \boldsymbol{b}_1,\boldsymbol{b}_i\rangle - \ldots - a_k\langle \boldsymbol{b}_k,\boldsymbol{b}_i\rangle. \end{aligned}$$

Hodnoty $a_1,\ldots,a_k \in \mathbb{R}$ tedy dostaneme jako řešení následující soustavy rovnic:

$$\begin{pmatrix} \langle \boldsymbol{b}_1,\boldsymbol{b}_1\rangle & \langle \boldsymbol{b}_1,\boldsymbol{b}_2\rangle & \cdots & \langle \boldsymbol{b}_1,\boldsymbol{b}_k\rangle \\ \langle \boldsymbol{b}_2,\boldsymbol{b}_1\rangle & \langle \boldsymbol{b}_2,\boldsymbol{b}_2\rangle & \cdots & \langle \boldsymbol{b}_2,\boldsymbol{b}_k\rangle \\ \vdots & \ddots & \vdots & \\ \langle \boldsymbol{b}_k,\boldsymbol{b}_1\rangle & \langle \boldsymbol{b}_k,\boldsymbol{b}_2\rangle & \cdots & \langle \boldsymbol{b}_k,\boldsymbol{b}_k\rangle \end{pmatrix} \cdot \begin{pmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{pmatrix} = \begin{pmatrix} \langle \boldsymbol{c},\boldsymbol{b}_1\rangle \\ \langle \boldsymbol{c},\boldsymbol{b}_2\rangle \\ \vdots \\ \langle \boldsymbol{c},\boldsymbol{b}_k\rangle \end{pmatrix}.$$

Co se týká složitosti výpočtu, označme $\sigma = \sum_{i=1}^{k} \operatorname{size}(\boldsymbol{b}_i) + \operatorname{size}(\boldsymbol{c})$. Potom prvky matice i prvky vektoru pravých stran mají velikost polynomiální v σ. Řešení soustavy můžeme dle lemmatu 1 spočítat v polynomiálním čase v σ. Dále pro a_i platí, že $a_i \in \mathbb{Q}$ a $\operatorname{size}(a_i)$ je polynomiální v σ. Z toho plyne, že i $\operatorname{size}(\boldsymbol{c}_B)$ je polynomiální v σ a $\boldsymbol{c}_B \in \mathbb{Q}^n$. Dostáváme tak následující lemma.

Lemma 4. *Nechť $n,k \in \mathbb{N}, k \leq n, \boldsymbol{b}_1,\ldots,\boldsymbol{b}_k \in \mathbb{Z}^n$ jsou lineárně nezávislé vektory a nechť $\boldsymbol{c} \in \mathbb{Z}^n$. Potom lze projekci $\boldsymbol{c}_B$ vektoru $\boldsymbol{c}$ na podprostor $B = \langle \boldsymbol{b}_1,\ldots,\boldsymbol{b}_k\rangle$ spočítat v polynomiálním čase v $\sum_{i=1}^{k} \operatorname{size}\boldsymbol{b}_i + \operatorname{size}\boldsymbol{c}$.*

Dále platí, že $\operatorname{size}(\boldsymbol{c}_B)$ je polynomiální v $\sum_{i=1}^{k} \operatorname{size}\boldsymbol{b}_i + \operatorname{size}\boldsymbol{c}$ a $\boldsymbol{c}_B \in \mathbb{Q}^n$.

1.2 Základní definice

Definice 3. *Nechť $n,k \in \mathbb{N}, k \leq n$ a nechť $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_k \in \mathbb{R}^n$ jsou lineárně nezávislé vektory. Potom množinu*

$$L = \{c_1\boldsymbol{b}_1 + \cdots + c_k\boldsymbol{b}_k \colon c_1,\ldots,c_k \in \mathbb{Z}\}$$

nazveme mřížkou. Vektory $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_k$ nazýváme bází mřížky L. Tuto mřížku značíme $L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_k)$.

Definice 4. *Mřížku $L \subset \mathbb{R}^n$ nazveme celočíselnou, pokud $L \subset \mathbb{Z}^n$.*

Definice 5. *Nechť $q \in \mathbb{N}$. Celočíselnou mřížku $L \subset \mathbb{Z}^n$ nazveme q-ární, pokud $q\mathbb{Z}^n \subset L \subset \mathbb{Z}^n$.*

V dalším textu budeme pracovat s následujícími mřížkami: mějme $q,n,m \in \mathbb{N}$, dále mějme matici $B \in \mathbb{Z}_q^{n\times m}$, pak definujme:

$$\begin{aligned} \mathcal{L}_q(B) &= \{\boldsymbol{y} \in \mathbb{Z}^m \colon \boldsymbol{y} \equiv B^T\boldsymbol{s} \bmod q \text{ pro nějaké } \boldsymbol{s} \in \mathbb{Z}^n\} \\ \mathcal{L}_q^{\perp}(B) &= \{\boldsymbol{y} \in \mathbb{Z}^m \colon B\boldsymbol{y} \equiv 0 \bmod q\}. \end{aligned}$$

Pozorování 1. Nechť $q, n, m \in \mathbb{Z}$ a $B \in \mathbb{Z}_q^{n \times m}$. Potom $\mathcal{L}_q(B)$ a $\mathcal{L}_q^\perp(B)$ jsou q-ární mřížky. $\mathcal{L}_q(B)$ je generována řádky matice B a $q\boldsymbol{e}_1, \ldots, q\boldsymbol{e}_m$, kde $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_m$ jsou vektory kanonické báze. Obdobně $\mathcal{L}_q^\perp(B)$ je generována množinou generátorů $\ker B \bmod q$ a vektory $q\boldsymbol{e}_1, \ldots, q\boldsymbol{e}_m$. Snadno nahlédneme, že $q\mathbb{Z}^m \subset \mathcal{L}_q(B)$ a $q\mathbb{Z}^m \subset \mathcal{L}_q^\perp(B)$.

Dále definujeme determinant mřížky. Vystačíme si pouze se zjednodušenou definicí, kdy báze mřížky tvoří zároveň bázi celého prostoru:

Definice 6. *Nechť $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{R}^n$ je báze $\mathbb{R}^n$, potom determinant mřížky $L = L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ definujeme jako $|\det((\boldsymbol{a}_1 | \ldots | \boldsymbol{a}_n))|$. Značíme $\det(L)$.*

Následující lemma budeme potřebovat k důkazu lemmatu 6. Toto lemma potom zaručuje, že determinant mřížky nezávisí na volbě báze.

Lemma 5. *Nechť M je unimodulární, potom i M^{-1} je unimodulární.*

Důkaz. Z $\det(MM^{-1}) = 1$ a z toho, že M je celočíselná matice dostaneme, že $\det(M^{-1}) = \pm 1$. Zbývá dokázat, že M^{-1} je celočíselná. To nahlédneme z toho, že i-tý sloupec M^{-1} lze dostat jako řešení soustavy $M \cdot \boldsymbol{x} = \boldsymbol{e}_i$, kde $\boldsymbol{e}_i$ je i-tý vektor kanonické báze. Dle Cramerova pravidla pro j-tou složku vektoru $\boldsymbol{x}$ platí:

$$x_j = \frac{\det(M_j)}{\det(M)},$$

kde M_j je matice, která vznikne nahrazením j-tého sloupce matice M vektorem $\boldsymbol{e}_i$. Protože je $\det(M) = \pm 1$, potom $x_j \in \mathbb{Z}$ a tedy M^{-1} je celočíselná. □

Lemma 6. *Nechť $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{R}^n$ je báze $\mathbb{R}^n$ a $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^n$ je báze $\mathbb{R}^n$. Potom $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ a $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ tvoří bázi stejné mřížky právě, když matice přechodu od $(\boldsymbol{a}_1 | \ldots | \boldsymbol{a}_n)$ k $(\boldsymbol{b}_1 | \ldots | \boldsymbol{b}_n)$ je unimodulární.*

Důkaz. Označme $A = (\boldsymbol{a}_1 | \ldots | \boldsymbol{a}_n)$ a $B = (\boldsymbol{b}_1 | \ldots | \boldsymbol{b}_n)$ a nechť X je matice přechodu od A k B, neboli $B = AX$. Potom $A = BX^{-1}$.

$\Rightarrow$: Nechť $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ a $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ tvoří bázi stejné mřížky. Potom každé $\boldsymbol{b}_i$ leží v $L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ a tedy X je celočíselná. Obdobně se dokáže, že také X^{-1} je celočíselná. Protože $\det(XX^{-1}) = 1$ a X je celočíselná, potom $\det(X) = \pm 1$.

$\Leftarrow$: Nechť X unimodulární, tedy celočíselná, potom $\boldsymbol{b}_i \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ a tedy $L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$. Z lemmatu 5 plyne, že i X^{-1} je unimodulární, speciálně X^{-1} je celočíselná a tedy $L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \subset L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$. □

Definice 7. *Nechť $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in \mathbb{R}^n, k \leq n$ jsou lineárně nezávislé vektory. Potom k-dimenzionální rovnoběžnostěn $\{\sum_{i=1}^k \alpha_i \boldsymbol{b}_i : 0 \leq \alpha_i \leq 1\}$ značíme $\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$ a množinu $\{\sum_{i=1}^k \alpha_i \boldsymbol{b}_i : 0 \leq \alpha_i < 1\}$ značíme $\mathcal{P}^-(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$.*

Definice 8. *Nechť $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in \mathbb{R}^n, k \leq n$ jsou lineárně nezávislé vektory. Potom definujme objem rovnoběžnostěnu $\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)$ jako:*

$$\mathrm{vol}(\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k)) = \sqrt{\left|\det\left(\left(\boldsymbol{b}_1 | \ldots | \boldsymbol{b}_k\right)^\top \cdot \left(\boldsymbol{b}_1 | \ldots | \boldsymbol{b}_k\right)\right)\right|}$$

Poznámka 2. Pokud $k = n$, potom se objem $\mathrm{vol}\left(\mathcal{P}\left(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\right)\right)$ rovná s determinantu mřížky $L\left(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n\right)$.

Definice 9. *Nechť $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in \mathbb{R}^n, k \leq n$ jsou lineárně nezávislé vektory. Potom definujme povrch rovnoběžnostěnu $\mathcal{P}\left(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\right)$ jako:*

$$\mathrm{sur}\left(\mathcal{P}\left(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\right)\right) = 2 \cdot \sum_{\{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{k-1}\} \subset \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}} \mathrm{vol}\left(\mathcal{P}\left(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{k-1}\right)\right)$$

A minimální výšku téhož rovnoběžnostěnu jako:

$$\mathrm{minh}\left(\mathcal{P}\left(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\right)\right) = \min_{\{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{k-1}\} \subset \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}} \frac{\mathrm{vol}\left(\mathcal{P}\left(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\right)\right)}{\mathrm{vol}\left(\mathcal{P}\left(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{k-1}\right)\right)}$$

Lemma 7. *Nechť $n, k \in \mathbb{N}, k \leq n, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k \in \mathbb{R}^n$ jsou lineárně nezávislé vektory. Pak pro každé $\boldsymbol{b} \in \langle \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k \rangle$ existuje právě jedno $\acute{\boldsymbol{b}} \in \mathcal{P}^-\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$ tak, že $\boldsymbol{b} - \acute{\boldsymbol{b}} \in L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$.*

Pokud navíc $\boldsymbol{b} \in \mathbb{Z}^n$ a $\boldsymbol{a}_i \in \mathbb{Z}^n$, pak lze $\acute{\boldsymbol{b}}$ spočítat v polynomiálním čase v $\sum_{i=1}^k \mathrm{size}\left(\boldsymbol{a}_i\right) + \mathrm{size}\left(\boldsymbol{b}\right)$.

Důkaz. Vyjádřeme $\boldsymbol{b}$ jako lineární kombinaci vektorů $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k$: $\boldsymbol{b} = \sum_{i=1}^k c_i \boldsymbol{a}_i$. Dále nechť $d_i = \lfloor c_i \rfloor$ a položme $\acute{\boldsymbol{b}} = \sum_{i=1}^k \left(c_i - d_i\right) \boldsymbol{a}_i$. Vyjádření $\boldsymbol{b}$ jako lineární kombinace lze nad $\mathbb{R}$ provést například pomocí Gaussovy eliminace. Nad celými čísly máme z lemmatu 1 zaručenou existenci algoritmu, který pracuje v polynomiálním čase v $\sum_{i=1}^k \mathrm{size}\left(\boldsymbol{a}_i\right) + \mathrm{size}\left(\boldsymbol{b}\right)$.

Z $\boldsymbol{b} - \acute{\boldsymbol{b}} = \sum_{i=1}^k d_i \boldsymbol{a}_i$ a $d_i \in \mathbb{Z}$ dostáváme $\boldsymbol{b} - \acute{\boldsymbol{b}} \in L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$, dále z $0 \leq c_i - d_i < 1$ plyne $\acute{\boldsymbol{b}} \in \mathcal{P}^-\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$. Zbývá dokázat jednoznačnost.

Nechť $\boldsymbol{b}_1, \boldsymbol{b}_2 \in \mathcal{P}^-\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$ a $\boldsymbol{b} - \boldsymbol{b}_1 \in L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right), \boldsymbol{b} - \boldsymbol{b}_2 \in L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$. Potom $\boldsymbol{b}_1 - \boldsymbol{b}_2 = -\left(\boldsymbol{b} - \boldsymbol{b}_1\right) + \left(\boldsymbol{b} - \boldsymbol{b}_2\right) \in L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$. Z čehož plyne, že $\boldsymbol{b}_1 - \boldsymbol{b}_2 = 0$. □

Důsledek 1. Nechť $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{R}^n$ jsou lineárně nezávislé vektory a nechť $L = L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n\right)$. Potom:

$$\mathbb{R}^n = \dot{\bigcup}_{\boldsymbol{l} \in L} \boldsymbol{l} + \mathcal{P}^-\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n\right)$$

Důsledek 2. Nechť $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{R}^n$ jsou lineárně nezávislé vektory, $\alpha \in \mathbb{N}$ a nechť $L = L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n\right)$. Potom $\left|L \cap \mathcal{P}^-\left(\alpha \boldsymbol{a}_1, \ldots, \alpha \boldsymbol{a}_n\right)\right| = \alpha^n$.

Definice 10. *Vektor $\acute{\boldsymbol{b}}$ z lemmatu 7 budeme značit $\boldsymbol{b}_{(\mathrm{mod}\, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k)}$.*

Lemma 8. *Nechť $n, k \in \mathbb{N}, k \leq n, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k \in \mathbb{R}^n$ jsou lineárně nezávislé vektory. Pak pro každé $\boldsymbol{b} \in \langle \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k \rangle$ existuje $\acute{\boldsymbol{b}} \in \mathbb{R}^n$ tak, že $\boldsymbol{b} - \acute{\boldsymbol{b}} \in L\left(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k\right)$ a $\left\|\acute{\boldsymbol{b}}\right\| \leq \frac{1}{2} \sum_{i=1}^k \left\|\boldsymbol{a}_i\right\|$.*

Pokud navíc $\boldsymbol{b} \in \mathbb{Z}^n$ a $\boldsymbol{a}_i \in \mathbb{Z}^n$, pak lze $\acute{\boldsymbol{b}}$ spočítat v polynomiálním čase v $\sum_{i=1}^k \mathrm{size}\left(\boldsymbol{a}_i\right) + \mathrm{size}\left(\boldsymbol{b}\right)$.

Důkaz. Vyjádřeme $\boldsymbol{b}$ jako lineární kombinaci vektorů $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k$: $\boldsymbol{b} = \sum_{i=1}^k c_i \boldsymbol{a}_i$. Dále nechť $d_i = \lfloor c_i \rceil$ a položme $\acute{\boldsymbol{b}} = \sum_{i=1}^k \left(c_i - d_i\right) \boldsymbol{a}_i$. Vyjádření $\boldsymbol{b}$ jako lineární kombinace lze nad $\mathbb{R}$ provést například pomocí Gaussovy eliminace. Nad celými

čísly máme z lemmatu 1 zaručenou existenci algoritmu, který pracuje v polynomiálním čase v $\sum_{i=1}^{k} \text{size}(\boldsymbol{a}_i) + \text{size}(\boldsymbol{b})$.

Z $\boldsymbol{b} - \acute{\boldsymbol{b}} = \sum_{i=1}^{k} d_i \boldsymbol{a}_i$ a $d_i \in \mathbb{Z}$ dostáváme $\boldsymbol{b} - \acute{\boldsymbol{b}} \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_k)$, dále z $|c_i - d_i| \leq \frac{1}{2}$ plyne:

$$\left\|\acute{\boldsymbol{b}}\right\| = \left\|\sum_{i=1}^{k} (c_i - d_i)\, \boldsymbol{a}_i\right\| \leq \sum_{i=1}^{k} |c_i - d_i| \, \|\boldsymbol{a}_i\| \leq \frac{1}{2} \sum_{i=1}^{k} \|\boldsymbol{a}_i\| .$$

□

Definice 11. *Vektor $\acute{\boldsymbol{b}}$ z důkazu v lemmatu 8 budeme značit $\acute{\boldsymbol{b}} = \boldsymbol{b}_{\lfloor \boldsymbol{a}_1, \ldots, \boldsymbol{a}_k \rceil}$.*

1.3 Výpočetní problémy

Připomeňme, že bez bližší specifikace normou myslíme ℓ_2-normu, neboli $\|\boldsymbol{x}\| = \sqrt{\sum_{i=1}^{n} (x_i)^2}$.

Definice 12. *Nechť $L = L(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ je mřížka v $\mathbb{R}^n$. Délkou báze $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ máme na mysli hodnotu* $\max_{i=1,\ldots,n} \|\boldsymbol{b}_i\|$.

Definice 13. *Nechť L je mřížka v $\mathbb{R}^n$. Potom nejkratší bází máme na mysli bázi s nejkratší délkou. Délku nejkratší báze mřížky L budeme značit* $\text{bl}(L)$. *Jedná se tedy o hodnotu:*

$$\text{bl}(L) = \min_{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in \mathbb{R}^n \text{ báze } L} \max \|\boldsymbol{b}_i\| .$$

Definice 14. *Nechť L je mřížka v $\mathbb{R}^n$. Potom $\lambda_i(L)$ značí:*

$$\lambda_i(L) = \min_{r \in \mathbb{R}} \{L \text{ obsahuje } i \text{ lineárně nezávislých vektorů kratších než } r\}$$

Poznámka 3. Obecně neplatí, že $\lambda_n(L) = bl(L)$. Protipříkladem je mřížka v $L \subset \mathbb{Z}^5$ s následující bází:

$$B = \left\{ \begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{pmatrix} \right\}.$$

Potom $\boldsymbol{v} = (v_1, v_2, v_3, v_4, v_5) \in L \Leftrightarrow v_1 \equiv v_2 \equiv v_3 \equiv v_4 \equiv v_5 \mod 2$. Z toho plyne, že $\lambda_1 = \lambda_2 = \lambda_3 = \lambda_4 = \lambda_5 = 2$, kde λ_5 plyne z toho, že jediná pětice lineárně nezávislých vektorů délky 2 je až na násobek -1:

$$C = \left\{ \begin{pmatrix} 2 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 2 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 2 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 2 \end{pmatrix} \right\}.$$

C ale není báze, protože vektor (1,1,1,1,1) neleží v lineárním obalu C. Potom $\text{bl}(L) = \sqrt{5}$.

Definice 15 (Problém nejkratšího vektoru - **Shortest vector problem** (SVP))**.** *Dána báze* $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ *mřížky* $L = L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$*, úkolem je najít nejkratší nenulový vektor v mřížce* L*, neboli najít vektor* $\boldsymbol{v}$ *takový, že* $\|\boldsymbol{v}\| = \lambda_1(L)$*.*

Definice 16 (Problém nejkratších lineárně nezávislých vektorů - **Shortest independent vectors problem** (SIVP))**.** *Dána libovolná báze* $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ *mřížky* $L = L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$*, úkolem je najít lineárně nezávislé vektory* $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n \in L$ *pro které:* $\|\boldsymbol{v}_i\| \leq \lambda_n(L)$*.*

Definice 17 (Problém nejkratší báze - **shortest basis problem** (SBP))**.** *Dána libovolná báze* $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ *mřížky* $L = L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$*, úkolem je najít nejkratší bázi* $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n \in L$*, neboli:* $\|\boldsymbol{c}_i\| \leq \mathrm{bl}(L)$*.*

Všechny tři zmíněné problémy budeme uvažovat také v jednodušší aproximační verzi. K tomu uvažme, že f je funkce $\mathbb{N} \to \mathbb{R}$.

Definice 18 (Problém přibližně nejkratšího vektoru - **Approximate shortest vector problem** (SVP_f))**.** *Dána báze* $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ *mřížky* $L = L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$*, úkolem je najít nenulový vektor* $v \in L$ *pro který:* $\|\boldsymbol{v}\| \leq f(n) \cdot \lambda_1(L)$*.*

Definice 19 (Problém přibližně nejkratších lineárně nezávislých vektorů - **Approximate shortest independent vectors problem** (SIVP_f))**.** *Dána libovolná báze* $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ *mřížky* $L = L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$*, úkolem je najít lineárně nezávislé vektory* $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n \in L$ *pro které:* $\|\boldsymbol{v}_i\| \leq f(n) \cdot \lambda_n(L)$*.*

Definice 20 (Problém přibližně nejkratší báze - **Approximate shortest basis problem** (SBP_f))**.** *Dána libovolná báze* $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ *mřížky* $L = L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$*, úkolem je najít bázi* $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n \in L$ *pro kterou platí:* $\|\boldsymbol{c}_i\| \leq f(n) \cdot \mathrm{bl}(L)$*.*

Následující lemma popisuje vztah mezi SBP_f a SIVP_f, konkrétně platí, že SBP_{f+n} může být redukován na SIVP_f. Důkaz lemmatu vychází z Micciancio a Goldwasser (2002, Lemma 7.1).

Lemma 9. *Nechť* $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n \in \mathbb{Z}^n$ *jsou lineárně nezávislé vektory,* $\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n \in L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$ *jsou lineárně nezávislé vektory a* $\max_{i=1}^{n} \|\boldsymbol{d}_i\| \leq M$*. Potom existuje báze* $L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$ *obsahující vektory kratší než* nM*.*

Tuto bázi lze spočítat v polynomiálním čase v $\sum_{i=1}^{n} (\mathrm{size}(\boldsymbol{a}_i) + \mathrm{size}(\boldsymbol{d}_i))$*.*

Důkaz. Označme $A = (\boldsymbol{a}_1|\ldots|\boldsymbol{a}_n)$. Obdobně $D = (\boldsymbol{d}_1|\ldots|\boldsymbol{d}_n)$. Protože $\boldsymbol{d}_i \in L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$, potom existuje celočíselná matice Q taková, že $D = AQ$. Z lemmatu 3 plyne, že existují matice $T, U \in \mathbb{Z}^{n\times n}$ takové, že $T = UQ$, kde U je unimodulární a T je horní trojúhelníková. Protože D i A mají hodnost n, potom i Q a T mají hodnost n, tedy T má na diagonále nenulové hodnoty. Položme $B = AU^{-1}$. Protože U je unimodulární, potom dle lemmatu 5 je i U^{-1} unimodulární a tedy dle lemmatu 6 tvoří sloupce matice $B = (\boldsymbol{b}_1|\ldots|\boldsymbol{b}_n)$ bázi mřížky $L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$. Nyní z vektorů $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ vytvoříme vektory $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n$ o kterých

tvrdíme, že splňují požadavky lemmatu. Postup pro vytvoření vektorů $\boldsymbol{c}_i$ je popsán v algoritmu Baze:

Algoritmus 1: Baze

Input : $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n \in \mathbb{Z}^n$ LN, $\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n \in L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$ LN
Output: $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n$ báze $L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n), \max\|\boldsymbol{c}_i\| \leq n\max\|\boldsymbol{d}_i\|$
$A = (\boldsymbol{a}_1|\ldots|\boldsymbol{a}_n)$
$D = (\boldsymbol{d}_1|\ldots|\boldsymbol{d}_n)$
spočti Q: $D = AQ$
spočti T,U: $T = UQ$
$B = (\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n) = AU^{-1}$
$\boldsymbol{c}_1 = \boldsymbol{b}_1$
for i=2,...,n **do**
  spočti $\boldsymbol{b}_i^d$ projekci $\boldsymbol{b}_i$ na $\langle \boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1}\rangle$
  spočti $\boldsymbol{b}^d_{i(\mathrm{mod}\,\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}$
  $\boldsymbol{c}_i = \boldsymbol{b}_i - \left(\boldsymbol{b}_i^d - \boldsymbol{b}^d_{i(\mathrm{mod}\,\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}\right)$
end for
return $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n$

Nejdříve dokážeme, že vektory $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n$ tvoří bázi mřížky $L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$. K tomu nahlédneme, že platí následující rovnost:

$$D = AQ = AU^{-1}UQ = BT.$$

Protože je T horní trojúhelníková s nenulovými prvky na diagonále, potom platí:

$$\boldsymbol{d}_i = \sum_{j=1}^{i} t_{ji}\boldsymbol{b}_j = \sum_{j=1}^{i-1} t_{ji}\boldsymbol{b}_j + t_{ii}\boldsymbol{b}_i, \tag{1.1}$$

kde t_{ij} je prvek matice T na pozici (i,j) a $t_{ii} \in \mathbb{Z}\setminus\{0\}$. Speciálně platí:

$$\boldsymbol{d}_i \in L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_i). \tag{1.2}$$

Dále dokážeme, že pro každé $i = 1,\ldots,n$ platí:

$$L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_i) = L(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_i). \tag{1.3}$$

Pro $i = 1$ rovnost triviálně platí. Dále budeme pokračovat indukcí. Nechť tedy rovnost platí pro $i-1$. Nechť nejprve $\boldsymbol{z} \in L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_i)$, potom $\boldsymbol{z} = \sum_{j=1}^{i} u_j\boldsymbol{b}_j$ pro nějaká $u_j \in \mathbb{Z}$ a tedy

$$\boldsymbol{z} = \sum_{j=1}^{i} u_j\boldsymbol{b}_j = \sum_{j=1}^{i-1} u_j\boldsymbol{b}_j + u_i\boldsymbol{b}_i = \sum_{j=1}^{i-1} u_j\boldsymbol{b}_j + u_i\boldsymbol{c}_i + u_i\left(\boldsymbol{b}_i^d - \boldsymbol{b}^d_{i(\mathrm{mod}\,\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}\right).$$

Dále dle lemmatu 7, 1.2 a z indukčního předpokladu platí

$$\boldsymbol{b}_i^d - \boldsymbol{b}^d_{i(\mathrm{mod}\,\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})} \in L(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1}) \subset L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_{i-1}) = L(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_{i-1}). \tag{1.4}$$

Dále z indukčního předpokladu plyne, že $\sum_{j=1}^{i-1} u_j\boldsymbol{b}_j \in L(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_{i-1})$ a tedy $\boldsymbol{z} \in L(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_i)$. Druhá inkluze se dokáže obdobně. Vektor $\boldsymbol{z} \in L(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_i)$ rozepíšeme jako

$$\boldsymbol{z} = \sum_{j=1}^{i} u_j\boldsymbol{c}_j = \sum_{j=1}^{i-1} u_j\boldsymbol{c}_j + u_i\boldsymbol{c}_i = \sum_{j=1}^{i-1} u_j\boldsymbol{c}_j + u_i\boldsymbol{b}_i - u_i\left(\boldsymbol{b}_i^d - \boldsymbol{b}^d_{i(\mathrm{mod}\,\boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}\right).$$

Z indukčního předpokladu plyne, že $\sum_{j=1}^{i-1} u_j \boldsymbol{c}_j \in L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_{i-1})$, z 1.4 máme $\left(\boldsymbol{b}_i^d - \boldsymbol{b}^d_{i(\bmod \boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}\right) \in L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_{i-1})$, a tedy $\boldsymbol{z} \in L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_i)$. Dokázali jsme, že 1.3 platí pro $i = 1,\ldots,n$, a tedy $L(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n) = L(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n)$. Protože $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ tvoří bázi mřížky, potom jsou i vektory $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n$ bází této mřížky.

Dále dokážeme odhad na normu vektorů $\boldsymbol{c}_i$. K tomu se zaměřme na kolmici $\boldsymbol{d}_i^\perp$ vektoru $\boldsymbol{d}_i$ na podprostor $\langle \boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1}\rangle$. Z 1.2 plyne, že $\langle \boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1}\rangle = \langle \boldsymbol{b}_1,\ldots,\boldsymbol{b}_{i-1}\rangle$ a to spolu s 1.1 dává, že kolmice $\boldsymbol{d}_i^\perp$ je rovna kolmici $\boldsymbol{b}_i^\perp$ vektoru $t_{ii}\boldsymbol{b}_i$ na podprostor $\langle \boldsymbol{b}_1,\ldots,\boldsymbol{b}_{i-1}\rangle$. Protože $t_{11} \in \mathbb{Z}\setminus\{0\}$ pak: $\left\|\boldsymbol{b}_i^\perp\right\| \le \left\|\boldsymbol{d}_i^\perp\right\| \le \|\boldsymbol{d}_i\|$.

Nyní můžeme provést samotný odhad normy:

$$\begin{aligned}\|\boldsymbol{c}_i\| &= \left\|\boldsymbol{b}_i - \left(\boldsymbol{b}_i^d - \boldsymbol{b}^d_{i(\bmod \boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}\right)\right\| \le \left\|\boldsymbol{b}_i - \boldsymbol{b}_i^d\right\| + \left\|\boldsymbol{b}^d_{i(\bmod \boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}\right\| \\ &= \left\|\boldsymbol{b}_i^\perp\right\| + \left\|\boldsymbol{b}^d_{i(\bmod \boldsymbol{d}_1,\ldots,\boldsymbol{d}_{i-1})}\right\| \le \|\boldsymbol{d}_i\| + \sum_{j=1}^{i-1}\|\boldsymbol{d}_j\| \le iM \le nM,\end{aligned}$$

kde předposlední nerovnost plyne z toho, že vektory $\boldsymbol{d}_i$ jsou navzájem kolmé.

Nakonec dokážeme polynomiální složitost algoritmu Baze. K tomu označme $\sigma = \sum_{i=1}^n (\operatorname{size}(\boldsymbol{a}_i) + \operatorname{size}(\boldsymbol{d}_i))$. Matici Q vypočítáme vynásobením inverzní matice A^{-1} a matice D a: $Q = A^{-1}D$, to lze dle lemmatu 2 spočítat v polynomiálním čase v σ. Matice T a U lze dle lemmatu 3 spočítat v polynomiálním čase v σ, matici B obdobně jako Q také. Všechny matice Q, U, T, B mají polynomiální velikost v σ. Nakonec $n-1$-krát provedeme smyčku for-cyklu, kde první dva kroky lze dle lemmat 4 a 7 provést v polynomiálním čase v σ. Třetí krok v cyklu je standardní sčítání vektorů. □

V dimenzi 2 máme pro hledání nejkratší báze algoritmus, který najde nejkratší bázi přesně a pracuje v polynomiální čase - Gaussovu redukci mřížky:

Algoritmus 2: Gaussova redukce mřížky

Input : $\boldsymbol{a}_1, \boldsymbol{a}_2 \in \mathbb{Z}^n$ LN
Output: nejkratší báze $L(\boldsymbol{a}_1, \boldsymbol{a}_2)$
do
  if $\|\boldsymbol{a}_1\| > \|\boldsymbol{a}_2\|$ **then**
    Prohoď $\boldsymbol{a}_1, \boldsymbol{a}_2$
  end if
  $x = \left\lfloor \frac{\boldsymbol{a}_1 \cdot \boldsymbol{a}_2}{\|\boldsymbol{a}_1\|^2} \right\rceil$
  $\boldsymbol{a}_2 = \boldsymbol{a}_2 - x \cdot \boldsymbol{a}_1$
while $x \neq 0$;
return $\boldsymbol{a}_1, \boldsymbol{a}_2$

Algoritmus Gaussovy redukce mřížky pracuje v polynomiálním čase vzhledem k velikosti vstupních vektorů.

Polynomiální algoritmus pro přesné hledání nejkratší báze je znám až do dimenze 4, jak je uvedeno v Nguyen a Stehlé (2004). Požadujeme-li obecný algoritmus pro libovolnou dimenzi s polynomiální složitostí, máme k dispozici například velmi známý LLL algoritmus. Ten vrátí bázi s aproximačním faktorem $2^{\mathcal{O}(n)}$. Na druhou stranu všechny dnes známé algoritmy, které najdou nejkratší bázi přesně (s aproximačním faktorem 1), mají exponenciální složitost.

Obdobně je tomu tak i se složitostí pro další dva výše zmíněné problémy: SVP, SIVP. V Micciancio a Goldwasser (2002, Sekce 7.4) je shrnuta složitost

těchto problémů, zejména je zde uvedeno, že všechny tři zmíněné problémy jsou NP-těžké. To koresponduje s tím, že nejsou známy žádné algoritmy, které řeší SVP v polynomiálním čase. Takové algoritmy ani neexistují, pokud P≠NP. Dále je dokázáno, že tyto problémy jsou NP-těžké až pro aproximační faktor $f = n^{1/\log\log n}$. Dále je třeba zmínit, že nejsou známy žádné algoritmy s polynomiální složitostí, které řeší výše zmíněné problémy v aproximační verzi, kde faktor je polynomiální funkce v dimenzi mřížky.

Výše zmíněné nám dává základní předpoklad ohledně složitosti SVP_f, SIVP_f a SBP_f:

Hypotéza 1. *Neexistuje žádný polynomiální algoritmus, který by řešil přibližné SVP_f, $SIVP_f$ nebo SBP_f pro aproximační faktor polynomiální v dimenzi mřížky.*

V této práci budeme pracovat ještě s dalším výpočetním problémem týkajícím se mřížek a to hledáním krátkého vektoru v mřížkách $\mathcal{L}_q^\perp(A)$ popsaných pod definicí 5. Pro naše potřeby pro délku vektoru volíme ℓ_1-normu, avšak protože pro každé $\boldsymbol{x} \in \mathbb{R}^n$ platí vztah:

$$\|\boldsymbol{x}\|_2 \leq \|\boldsymbol{x}\|_1 \leq \sqrt{n}\,\|\boldsymbol{x}\|_2\,,$$

lze volit standardní ℓ_2-normu a další postupy příslušně upravit (zejména se příslušně změní aproximační faktory).

Definice 21 (Problém krátkého řešení - **Short integer solution** (SIS))**.** *Dána matice $A \in \mathbb{Z}_q^{n\times m}$ a $\beta < q$, úkolem je najít vektor $\boldsymbol{v} \in \mathbb{Z}^m, \boldsymbol{v} \neq \vec{0}$ pro který: $\|\boldsymbol{v}\|_1 \leq \beta$ a $A\boldsymbol{v} \equiv 0 \bmod q$.*

Poznámka 4. Podmínka $\beta < q$ vyloučí triviální řešení typu $\boldsymbol{v} = (q,0,\ldots,0)$. Dále platí, že vektor $\boldsymbol{v}$ leží v $\mathcal{L}_q^\perp(A)$. Jedná se tedy o krátký vektor v $\mathcal{L}_q^\perp(A)$.

Lemma 10. *Nechť $n, c_1, c_2 \in \mathbb{N}, c_1 > c_2, m = \lceil c_1 n \log_2 n\rceil, q = n^{c_2}, q > \beta \geq m$ a $A \in \mathbb{Z}_q^{n\times m}$. Potom existuje řešení problému SIS.*

Důkaz. Uvažme zobrazení $f\colon \mathbb{Z}_2^m \to \mathbb{Z}_q^n, f(\boldsymbol{v}) = A\boldsymbol{v}$. Protože platí:

$$2^m = 2^{\lceil c_1 n \log_2 n\rceil} > 2^{c_2 n \log_2 n} = n^{c_2 n} = q^n,$$

existují $\boldsymbol{v}_1, \boldsymbol{v}_2 \in \mathbb{Z}_2^m, \boldsymbol{v}_1 \neq \boldsymbol{v}_2$ takové, že $A\boldsymbol{v}_1 = A\boldsymbol{v}_2$. Potom pro $\boldsymbol{v} = \boldsymbol{v}_1 - \boldsymbol{v}_2$ platí $A\boldsymbol{v} = 0$, a protože $\boldsymbol{v} \in \{-1,0,1\}^m$, dostáváme: $\|\boldsymbol{v}\|_1 \leq m \leq \beta$. □

Dále budeme pracovat s pravděpodobnostními algoritmy, které vždy skončí. Definujme dva typy pravděpodobnosti úspěchu algoritmu.

Definice 22. *Nechť $\mathcal{A}$ je pravděpodobnostní algoritmus a $p \in [0,1]$. Potom řekneme, že úspěšnost algoritmu $\mathcal{A}$ v **nejhorším případě** je alespoň p, pokud platí:*

$$P(\mathcal{A} \text{ uspěl na vstupu } x) \geq p,$$

pro každý vstup x algoritmu $\mathcal{A}$.

Definice 23. *Nechť $\mathcal{A}$ je pravděpodobnostní algoritmus, $p \in [0,1]$ a nechť pro každé $n \in \mathbb{N}$ je S_n množina všech možných vstupů $\mathcal{A}$ velikosti n. Potom řekneme, že úspěšnost algoritmu $\mathcal{A}$ v* ***průměrném případě*** *je alespoň p, pokud platí:*

$$\sum_{x \in S_n} P\left(\mathcal{A} \text{ uspěl na vstupu } x\right) \geq p\,|S_n|\,,$$

pro každé $n \in \mathbb{N}$.

Poznamenejme, že v průměrném případě mohou existovat vstupy, kde algoritmus neuspěje vůbec.

2. Redukce přibližného SBP na SIS

V této kapitole si ukážeme polynomiální redukci přibližného SBP_f na SIS. Tato redukce je typu nejhorší případ redukovaný na průměrný případ. Jinak řečeno: předpokládejme, že máme orákulum, které s nenulovou pravděpodobností řeší efektivně SIS v průměrné případě, potom lze za pomocí tohoto orákula řešit libovolnou instanci problému SBP_f s dostatečně velkou pravděpodobností, kde aproximační faktor je polynomiální v dimenzi mřížky: $f = \mathcal{O}(n^c), c \in \mathbb{N}$.

K důkazu existence této redukce budeme potřebovat několik technických lemmat. Následující lemma vychází z (Ajtai, 1996, lemma 3).

Lemma 11. *Mějme lineárně nezávislé vektory* $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{R}^n, \max_{i=1}^n \|\boldsymbol{a}_i\| \leq M, n \geq 3$. *Potom existují lineárně nezávislé vektory* $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$, *pro které platí:*

1. $\left(n^3 - \frac{1}{2}n\right) M \leq \|\boldsymbol{b}_i\| \leq \left(n^3 + \frac{1}{2}n\right) M, i = 1, \ldots, n$
2. $\frac{8}{27}(n^3 M)^n \leq \mathrm{vol}(\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)) \leq 3(n^3 M)^n$
3. $\frac{2}{3}n^3 M \leq \mathrm{minh}(\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n))$
4. $\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n) \subset K$, *kde* K *je krychle o hraně délky* $(n^3 + n^2) M$.

Pokud navíc $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{Z}^n$, *pak* $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ *lze spočítat v polynomiálním čase v* $\sum_{i=1}^n \mathrm{size}(\boldsymbol{a}_i)$.

Důkaz. Mějme vektory $\boldsymbol{f}_1, \ldots, \boldsymbol{f}_n \in \mathbb{R}^n$ splňující: $\boldsymbol{f}_i = n^3 M \boldsymbol{e}_i, i = 1, \ldots, n$, kde $\boldsymbol{e}_i$ jsou vektory kanonické báze. Dále položme $\boldsymbol{b}_i = \boldsymbol{f}_i - \boldsymbol{f}_{i\lfloor \boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \rceil}, i = 1, \ldots, n$.

Algoritmus 3: Pseudo-krychle

Input : $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{Z}^n$ LN, $\|\boldsymbol{a}_i\| \leq M$
Output: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$
$\boldsymbol{f}_i = n^3 M \boldsymbol{e}_i, i = 1, \ldots, n$
$\boldsymbol{b}_i = \boldsymbol{f}_i - \boldsymbol{f}_{i\lfloor \boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \rceil}, i = 1, \ldots, n$
return $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$;

Z lemmatu 8 plyne: $\boldsymbol{b}_i \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$.

Dále dokážeme, že vektory $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ splňují požadavky lemmatu. Z lemmatu 8 máme:

$$\|\boldsymbol{f}_i - \boldsymbol{b}_i\| = \left\|\boldsymbol{f}_{i\lfloor \boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \rceil}\right\| \leq \frac{1}{2}\sum_{i=1}^n \|\boldsymbol{a}_i\| \leq \frac{1}{2}nM.$$

Odhad délky vektorů plyne z $\|\boldsymbol{f}_i\| = n^3 M$ a z $\|\boldsymbol{f}_i - \boldsymbol{b}_i\| \leq \frac{1}{2}nM$.

Označme $Q = \mathcal{P}(\boldsymbol{f}_1, \ldots, \boldsymbol{f}_n)$. Potom vzdálenost příslušných vrcholů Q a $\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ je nejvýše:

$$\sum_{i=1}^n \|\boldsymbol{f}_i - \boldsymbol{b}_i\| \leq \sum_{i=1}^n \frac{n}{2}M \leq \frac{n^2}{2}M. \tag{2.1}$$

Definujme krychli Q_0, která vznikne zvětšením krychle Q o faktor $\left(1+\frac{1}{n}\right)$ a má stejný střed jako střed krychle Q:

$$Q_0 = -\frac{1}{2n}\sum_{i=1}^{n}\boldsymbol{f}_i + \mathcal{P}\left(\left(1+\frac{1}{n}\right)\boldsymbol{f}_1,\ldots,\left(1+\frac{1}{n}\right)\boldsymbol{f}_n\right)$$

Obdobně definujme krychli Q_1, která vznikne zmenšením krychle Q o faktor $\left(1-\frac{1}{n}\right)$ a má stejný střed jako střed krychle Q:

$$Q_1 = \frac{1}{2n}\sum_{i=1}^{n}\boldsymbol{f}_i + \mathcal{P}\left(\left(1-\frac{1}{n}\right)\boldsymbol{f}_1,\ldots,\left(1-\frac{1}{n}\right)\boldsymbol{f}_n\right)$$

Potom délka hrany krychle Q_0 je $(n^3+n^2)M$ a vzdálenost příslušných stěn Q a Q_0 je $\frac{n^2}{2}M$. Stejně tak vzdálenost příslušných stěn Q a Q_1 je $\frac{n^2}{2}M$. To spolu s 2.1 dává:

$$Q_1 \subset \mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n) \subset Q_0,$$

čímž je dokázána existence krychle K. Z $Q_1 \subset \mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$ dostáváme, že $\mathrm{vol}(Q_1) \leq \mathrm{vol}(\mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n))$.

$$\mathrm{vol}(Q_1) = \left(\left(n^3-n^2\right)M\right)^n = \left(1-\frac{1}{n}\right)^n\left(n^3M\right)^n \geq \frac{8}{27}\left(n^3M\right)^n,$$

kde poslední nerovnost plyne z toho, že $\left(1-\frac{1}{n}\right)^n$ je rostoucí na $\mathbb{N}$ a $n \geq 3$. Tím je dokázán spodní odhad na objem. Obdobně dokážeme i horní odhad: z $\mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n) \subset Q_0$ plyne $\mathrm{vol}(\mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)) \leq \mathrm{vol}(Q_0)$.

$$\mathrm{vol}(Q_0) = \left(\left(n^3+n^2\right)M\right)^n = \left(1+\frac{1}{n}\right)^n\left(n^3M\right)^n \leq 3\left(n^3M\right)^n,$$

kde poslední nerovnost plyne z toho, že $\left(1+\frac{1}{n}\right)^n$ je rostoucí na $\mathbb{N}$ a z toho, že $\lim_{n\to\infty}\left(1+\frac{1}{n}\right)^n = e \leq 3$.

Jelikož Q_1 obsahuje n-dimenzionální kouli o průměru $(n^3-n^2)M$, pro $n \geq 3$ dostáváme:

$$\mathrm{minh}(\mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)) \geq \left(n^3-n^2\right)M \geq \frac{2}{3}n^3M.$$

Lineární nezávislost vektorů $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ plyne z toho, že $0 < \frac{8}{27}(n^3M)^n \leq \mathrm{vol}(\mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n))$.

Nakonec dokážeme, že vektory $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n$ lze spočítat v polynomiálním čase. Snadno nahlédneme, že $\mathrm{size}(\boldsymbol{f}_i) = \mathcal{O}(\log(n^3M))$ a tedy $\mathrm{size}(\boldsymbol{f}_i)$ je polynomiální v $\sum_{i=1}^{n}\mathrm{size}(\boldsymbol{a}_i)$. Potom $\boldsymbol{f}_{i\lfloor\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\rceil}$ lze dle lemmatu 8 spočítat v polynomiálním čase v $\mathrm{size}(\boldsymbol{f}_i)$ a tedy i v $\sum_{i=1}^{n}\mathrm{size}(\boldsymbol{a}_i)$. Nakonec $\boldsymbol{b}_i$ získáme odečtením vektorů velikosti polynomiální v $\sum_{i=1}^{n}\mathrm{size}(\boldsymbol{a}_i)$, to lze provést v polynomiálním čase v $\sum_{i=1}^{n}\mathrm{size}(\boldsymbol{a}_i)$. □

Následující lemma a lemma 14 vycházejí z (Ajtai, 1996, lemma 5), avšak jsou formulována trochu jinak.

Lemma 12. *Nechť $L = L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$ je mřížka v $\mathbb{R}^n$, $\mathrm{bl}(L) = B, \boldsymbol{b}_1,\ldots,\boldsymbol{b}_n \in \mathbb{R}^n$ lineárně nezávislé, V je objem a H je minimální výška $\mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n), \boldsymbol{c} \in \mathbb{R}^n$ a nechť $\frac{2Bn}{H} < 1$.*

Označme $k_0 = |L \cap (\boldsymbol{c} + \mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n))|$ *a* $k_1 = |L \cap \operatorname{int}(\boldsymbol{c} + \mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n))|$, *potom:*

$$\left(1 - \frac{2Bn}{H}\right)^n \frac{V}{\det(L)} \leq k_j \leq \left(1 + \frac{2Bn}{H}\right)^n \frac{V}{\det(L)}, j = 0{,}1$$

Důkaz. Mějme $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n$ nejkratší bázi mřížky L, pro kterou platí $\max_{i=1}^n \|\boldsymbol{c}_i\| = B$ a označme $W = \boldsymbol{c} + \mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$. Dále definujme množinu $\mathcal{C}$:

$$\mathcal{C} = \{\boldsymbol{v} + \mathcal{P}(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n) : \boldsymbol{v} \in L, (\boldsymbol{v} + \mathcal{P}(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n)) \cap W \neq \emptyset\},$$

potom k_0 lze ze shora odhadnout velikostí množiny $\mathcal{C}$.

Dále nechť W_0 je rovnoběžnostěn vzniklý z W zvětšením o faktor $1 + \frac{2Bn}{H}$ se stejným středem jako W. Obdobně nechť W_ϵ je rovnoběžnostěn vzniklý z W zmenšením o faktor $1 - \frac{2Bn}{H} - \epsilon$, pro $\epsilon > 0$. Potom minimální výška W_0 je $H + 2Bn$, obdobně minimální výška W_ϵ je $H - 2Bn - H\epsilon$ a tedy vzdálenost stěn W a W_0 respektive W_ϵ je alespoň Bn.

Pro vzdálenost dvou bodů $\boldsymbol{a}, \boldsymbol{b} \in \mathcal{P}(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n)$ platí:

$$\|\boldsymbol{a} - \boldsymbol{b}\| \leq \left\|\sum_{i=1}^n \boldsymbol{c}_i\right\| \leq \sum_{i=1}^n \|\boldsymbol{c}_i\| \leq Bn.$$

Z toho dostáváme, že pro každé $C \in \mathcal{C}$ platí zároveň také: $C \subset W_0$. Počet prvků $C \in \mathcal{C}$ obsažených ve W_0 je maximálně:

$$\frac{\operatorname{vol}(W_0)}{\operatorname{vol}(\mathcal{P}(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n))} = \frac{\operatorname{vol}(W_0)}{\det(L)} = \left(1 + \frac{2Bn}{H}\right)^n \frac{V}{\det(L)},$$

čímž je dokázán horní odhad k_0 a tím také k_1.

Obdobně dostaneme, že každá množina $C \in \mathcal{C}$, pro kterou $C \cap W_\epsilon \neq \emptyset$ je obsažena ve W. Dolní odhad k_1 lze tedy odhadnout počtem prvků $C \in \mathcal{C}$ pro které $C \cap W_\epsilon \neq \emptyset$. To je alespoň:

$$\frac{\operatorname{vol}(W_\epsilon)}{\det(L)} = \left(1 - \frac{2Bn}{H} - \epsilon\right)^n \frac{V}{\det(L)}.$$

Odhad platí pro všechna $\epsilon > 0$ a limitním přechodem $\epsilon \to 0$ dostaneme požadovaný spodní odhad na k_1 a tedy také na k_0. □

Následující lemma je převzato z Ball (1986).

Lemma 13. *Nechť* $Q_n = \left[-\frac{1}{2}, \frac{1}{2}\right]^n$ *je* n*-dimenzionální jednotková krychle a nechť* F *je* $(n-1)$*-dimenzionální podprostor* $\mathbb{R}^n$. *Potom pro* $(n-1)$*-dimenzionální objem* $\operatorname{vol}(Q_n \cap F)$ *platí:*

$$\operatorname{vol}(Q_n \cap F) \leq \sqrt{2}.$$

Následující důsledek je opět ze stejného zdroje.

Důsledek 3. Nechť $Q_n = \left[-\frac{1}{2}, \frac{1}{2}\right]^n$ je n-dimenzionální jednotková krychle. Dále nechť $\boldsymbol{b} \in \mathbb{R}^n$ a F je $(n-1)$-dimenzionální podprostor $\mathbb{R}^n$. Potom pro $(n-1)$-dimenzionální objem $\operatorname{vol}(Q_n \cap (\boldsymbol{b} + F))$ platí:

$$\operatorname{vol}(Q_n \cap (\boldsymbol{b} + F)) \leq \sqrt{2}.$$

Jelikož je objem invariantní vůči posunutí, platí následující důsledek.

Důsledek 4. Nechť $Q_n = \left[-\frac{1}{2},\frac{1}{2}\right]^n$ je n-dimenzionální jednotková krychle a nechť $\boldsymbol{b},\boldsymbol{c} \in \mathbb{R}^n$ a F je $(n-1)$-dimenzionální podprostor R^n. Potom pro $(n-1)$-dimenzionální objem $\mathrm{vol}\left((\boldsymbol{c}+Q_n)\cap(\boldsymbol{b}+F)\right)$ platí:

$$\mathrm{vol}\left((\boldsymbol{c}+Q_n)\cap(\boldsymbol{b}+F)\right) \leq \sqrt{2}.$$

Důsledek 5. Nechť $a \in \mathbb{R}, a > 0, P_n = \left[-\frac{a}{2},\frac{a}{2}\right]^n$ je n-dimenzionální krychle s délkou hrany a a nechť $\boldsymbol{b},\boldsymbol{c} \in \mathbb{R}^n$ a F je $(n-1)$-dimenzionální podprostor R^n. Potom pro $(n-1)$-dimenzionální objem $\mathrm{vol}\left((\boldsymbol{c}+P_n)\cap(\boldsymbol{b}+F)\right)$ platí:

$$\mathrm{vol}\left((\boldsymbol{c}+P_n)\cap(\boldsymbol{b}+F)\right) \leq a^{n-1}\sqrt{2}.$$

Důkaz následujícího lemmatu je jedním z těch, který je rozepsán do detailu oproti původnímu důkazu v (Ajtai, 1996, lemma 5).

Lemma 14. *Mějme lineárně nezávislé vektory* $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n, \max_{i=1}^n \|\boldsymbol{a}_i\| \leq M, n \geq 3, L = L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n), \mathrm{bl}(L) = B$. *Dále nechť* $\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n \in \mathbb{R}^n$ *jsou vektory z lemmatu 11 a* $\boldsymbol{c} \in \mathbb{R}^n$. *Nechť* H *je minimální výška* $\mathcal{P}(\boldsymbol{b}_1,\ldots,\boldsymbol{b}_n)$. *Potom pro každé* $\boldsymbol{d} \in \mathbb{R}^n$ *a každou nadrovinu* F *v* $\mathbb{R}^n$ *platí:*

$$\left|(\boldsymbol{d}+F)\cap L\cap\left(\boldsymbol{c}+\mathcal{P}\left(\frac{1}{q}\boldsymbol{b}_1,\ldots,\frac{1}{q}\boldsymbol{b}_n\right)\right)\right| \leq \frac{8Bn\left(n^3M\right)^{n-1}}{\det(L)\,q^{n-1}}\left(1+\frac{2Bnq}{H}\right)^{n-1}.$$

Důkaz. Mějme $\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n$ nejkratší bázi mřížky L, pro kterou platí $\max_{i=1}^n \|\boldsymbol{c}_i\| = B$ a označme $W = \boldsymbol{c}+\mathcal{P}\left(\frac{1}{q}\boldsymbol{b}_1,\ldots,\frac{1}{q}\boldsymbol{b}_n\right)$. Dále definujme množinu $\mathcal{D}$:

$$\mathcal{D} = \{\boldsymbol{v}+\mathcal{P}(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n) : \boldsymbol{v} \in L, (\boldsymbol{v}+\mathcal{P}(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n))\cap W\cap(\boldsymbol{d}+F) \neq \emptyset\}$$

potom lze odhadovaný výraz ze shora odhadnout velikostí množiny $\mathcal{D}$.

Dále nechť W_0 je rovnoběžnostěn vzniklý z W zvětšením o faktor $1+\frac{2Bnq}{H}$ se stejným středem jako W. Tedy:

$$W_0 = \acute{\boldsymbol{c}}+\mathcal{P}\left(\left(1+\frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_1,\ldots,\left(1+\frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_n\right),$$

pro nějaké $\acute{\boldsymbol{c}} \in \mathbb{R}^n$. Potom minimální výška W_0 je $\frac{H}{q}+2Bn$ a tedy vzdálenost stěn W a W_0 je alespoň Bn.

Dále platí, že pro libovolné dva body $\boldsymbol{a},\boldsymbol{b} \in \mathcal{P}(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n)$ platí:

$$\|\boldsymbol{a}-\boldsymbol{b}\| \leq \left\|\sum_{i=1}^n \boldsymbol{c}_i\right\| \leq \sum_{i=1}^n \|\boldsymbol{c}_i\| \leq Bn.$$

Nechť π je ortogonální projekce $\mathbb{R}^n$ na $(\boldsymbol{d}+F)$. Definujme množinu G:

$$G = \{x \in \mathbb{R}^n : \pi(x) \in (\boldsymbol{d}+F)\cap W_0, \text{vzdálenost } x \text{ a } (\boldsymbol{d}+F) \leq Bn\}$$

Z následujícího:

1. každé dva body v $D \in \mathcal{D}$ jsou vzdáleny maximálně Bn

2. $\forall D \in \mathcal{D}\colon D \cap W \cap (\boldsymbol{d} + F) \neq \emptyset$

3. vzdálenost stěn W a W_0 je alespoň Bn,

dostáváme:

$$\forall D \in \mathcal{D}\colon \pi(D) \subset W_0 \cap (\boldsymbol{d} + F).$$

Dále z

1. každé dva body v $D \in \mathcal{D}$ jsou vzdáleny maximálně Bn

2. $\forall D \in \mathcal{D}\colon D \cap (\boldsymbol{d} + F) \neq \emptyset$

plyne:

$$\forall D \in \mathcal{D}\colon \forall \boldsymbol{e} \in D\colon \text{ vzdálenost } \boldsymbol{e} \text{ a } (\boldsymbol{d} + F) \text{ je maximálně } Bn.$$

Celkem tak dostáváme:

$$\forall D \in \mathcal{D} : D \subset G$$

Počet prvků $D \in \mathcal{D}$ obsažených v množině G je maximálně $\frac{\mathrm{vol}(G)}{\mathrm{vol}(\mathcal{P}(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n))}$. Zbývá odhadnout objem množiny G.

Objem G lze spočítat jako $(n-1)$-dimenzionální objem základny vynásobený výškou:

$$\mathrm{vol}(G) = \mathrm{vol}((\boldsymbol{d} + F) \cap W_0) \cdot 2Bn.$$

Nyní odhadneme objem základny:

$$\begin{aligned}&\mathrm{vol}((\boldsymbol{d} + F) \cap W_0)\\ =&\mathrm{vol}\left((\boldsymbol{d} + F) \cap \left(\acute{\boldsymbol{c}} + \mathcal{P}\left(\left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_1, \ldots, \left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_n\right)\right)\right).\end{aligned}$$

Jelikož vektory $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ splňují bod 4 z lemmatu 11, lze $\mathcal{P}(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n)$ ohraničit krychlí o délce hrany $\left(1 + \frac{1}{n}\right) n^3 M$ a tedy $\mathcal{P}\left(\left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_1, \ldots, \left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_n\right)$ lze ohraničit krychlí K o délce hrany $\left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\left(1 + \frac{1}{n}\right) n^3 M$. Potom existuje $\hat{\boldsymbol{c}}$ takové, že:

$$\left(\acute{\boldsymbol{c}} + \mathcal{P}\left(\left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_1, \ldots, \left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_n\right)\right) \subset (\hat{\boldsymbol{c}} + K),$$

a tedy

$$\begin{aligned}&(\boldsymbol{d} + F) \cap \left(\acute{\boldsymbol{c}} + \mathcal{P}\left(\left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_1, \ldots, \left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_n\right)\right) \subset\\ &(\boldsymbol{d} + F) \cap (\hat{\boldsymbol{c}} + K).\end{aligned}$$

Potom:

$$\begin{aligned}&\mathrm{vol}\left((\boldsymbol{d} + F) \cap \left(\acute{\boldsymbol{c}} + \mathcal{P}\left(\left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_1, \ldots, \left(1 + \frac{2Bnq}{H}\right)\frac{1}{q}\boldsymbol{b}_n\right)\right)\right)\\ \leq&\mathrm{vol}((\boldsymbol{d} + F) \cap (\hat{\boldsymbol{c}} + K)).\end{aligned}$$

Z posledního důsledku 5 dostáváme:

$$\mathrm{vol}\left((\boldsymbol{d}+F)\cap(\hat{\boldsymbol{c}}+K)\right)\leq\sqrt{2}\left(\left(1+\frac{2Bnq}{H}\right)\frac{1}{q}\left(1+\frac{1}{n}\right)n^3M\right)^{n-1}.$$

Celkem dostáváme:

$$\begin{aligned}\mathrm{vol}\,(G)\leq&\sqrt{2}\left(\left(1+\frac{2Bnq}{H}\right)\frac{1}{q}\left(1+\frac{1}{n}\right)n^3M\right)^{n-1}2Bn\\&=\left(1+\frac{2Bnq}{H}\right)^{n-1}\frac{1}{q^{n-1}}\left(1+\frac{1}{n}\right)^{n-1}\left(n^3M\right)^{n-1}2\sqrt{2}Bn\\&\leq\left(1+\frac{2Bnq}{H}\right)^{n-1}\frac{1}{q^{n-1}}\left(1+\frac{1}{n}\right)^{n}\left(n^3M\right)^{n-1}2\sqrt{2}Bn\\&\leq\left(1+\frac{2Bnq}{H}\right)^{n-1}\frac{1}{q^{n-1}}\left(n^3M\right)^{n-1}2\sqrt{2}eBn\\&\leq\left(1+\frac{2Bnq}{H}\right)^{n-1}\frac{1}{q^{n-1}}\left(n^3M\right)^{n-1}8Bn.\end{aligned}$$

a tedy

$$\begin{aligned}&\left|(\boldsymbol{d}+F)\cap L\cap\left(\boldsymbol{c}+\mathcal{P}\left(\frac{1}{q}\boldsymbol{b}_1,\ldots,\frac{1}{q}\boldsymbol{b}_n\right)\right)\right|\leq\mathcal{D}\leq\frac{\mathrm{vol}\,(G)}{\mathrm{vol}\left(\mathcal{P}\left(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_n\right)\right)}\\&\leq\left(1+\frac{2Bnq}{H}\right)^{n-1}\frac{1}{q^{n-1}}\left(n^3M\right)^{n-1}\frac{8Bn}{\det\left(L\right)}.\end{aligned}$$

□

Lemma 15. *Pro každé $x\in(-1,\infty)$ a $n\in\mathbb{N}$ platí:*

$$(1+x)^n\geq1+nx$$

Důkaz. Lemma dokážeme indukcí podle n. Pro $n=1$ je tvrzení triviální. Dále platí:

$$\begin{aligned}(1+x)^{n+1}=&(1+x)^n(1+x)\geq(1+nx)(1+x)=1+x+nx+nx^2\\\geq&1+x+nx=1+(n+1)x,\end{aligned}$$

kde v druhé nerovnosti jsme využili indukční předpoklad a fakt, že $(1+x)\geq0$. □

Následující důkaz vychází z (Ajtai, 1996, lemma 8), kde chybí detailní postup pro všechny odhady v důkazu.

Lemma 16. *Pro každé $a\in\mathbb{N}$ existuje $b\in\mathbb{N}$, pro které platí:*

Nechť $n\in\mathbb{N},\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\in\mathbb{Z}^n$ jsou lineárně nezávislé vektory a mějme mřížku $L=L\left(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\right)$. Dále nechť $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\in L$ jsou lineárně nezávislé vektory, pro které platí: $\|\boldsymbol{a}_i\|\leq2^{n^a}$ a $\|\boldsymbol{d}_i\|\leq2^{n^a}$. Dále nechť $\mu_1,\ldots,\mu_n$ jsou nezávislé náhodné veličiny s uniformním rozdělením na $\{0,1,2\ldots,2^{n^b}-1\}$.

Položme $\chi=\left(\sum_{i=1}^n\mu_i\boldsymbol{d}_i\right)_{(\mathrm{mod}\,\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)}$. Potom χ je skoro uniformní na $L\cap\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)$ v následujícím smyslu: označme $k=\left|\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)\cap L\right|$. Pak $\sum_{\boldsymbol{v}\in L\cap\mathcal{P}^-(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)}\left|P\left(\chi=\boldsymbol{v}\right)-\frac{1}{k}\right|\leq2^{-n^a}$.

Dále platí, že b stačí volit lineární v a, neboli $b=\mathcal{O}\left(a\right)$.

Důkaz. Pro každé $\alpha \in \mathbb{N}$ označme $W_\alpha = \mathcal{P}^- (\alpha \boldsymbol{d}_1, \ldots, \alpha \boldsymbol{d}_n)$. Potom dle důsledku 2 W_α obsahuje právě α^n mřížových bodů. Označme $\tau = \sum_{i=1}^n \mu_i \boldsymbol{d}_i$ a nechť $t = 2^{n^b}$, kde konkrétní hodnotu b určíme později. Potom τ má uniformní rozdělení na $W_t \cap L$.

Definujme množinu $\mathcal{X}$, která obsahuje disjunktní rovnoběžnostěny tvaru $\boldsymbol{u} + \mathcal{P}^- (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \subset W_t$:

$$\mathcal{X} = \{\boldsymbol{u} + \mathcal{P}^- (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) : \boldsymbol{u} = \sum_{i=1}^n c_i \boldsymbol{a}_i, c_i \in \mathbb{Z}, \boldsymbol{u} + \mathcal{P}^- (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \subset W_t\}.$$

Protože $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in L(\boldsymbol{d}_1, \ldots, \boldsymbol{d}_n)$, dostáváme, že každá množina $X \in \mathcal{X}$ obsahuje stejně mřížových bodů:

$$\left|\left(\boldsymbol{u} + \mathcal{P}^- (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)\right) \cap L\right| = \left|\mathcal{P}^- (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \cap L\right| = k, \boldsymbol{u} \in L$$

Dále platí:

$$W_t \cap L = \left(\left(W_t \cap \bigcup_{X \in \mathcal{X}} X\right) \cap L\right) \dot{\bigcup} \left(\left(W_t \setminus \bigcup_{X \in \mathcal{X}} X\right) \cap L\right),$$

to spolu s $|W_t \cap L| = t^n$ dává:

$$t^n = |\mathcal{X}|\, k + z, \tag{2.2}$$

pro nějaké $z \in \mathbb{N}_0$.

Nyní můžeme odhadnout $P(\chi = \boldsymbol{v}), \boldsymbol{v} \in \mathcal{P}^- (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n) \cap L$:

$$P(\chi = \boldsymbol{v}) = P\left(\chi = \boldsymbol{v}, \tau \in \bigcup_{X \in \mathcal{X}} X\right) + P\left(\chi = \boldsymbol{v}, \tau \notin \bigcup_{X \in \mathcal{X}} X\right)$$
$$= \sum_{X \in \mathcal{X}} P(\chi = \boldsymbol{v} | \tau \in X)\, P(\tau \in X) + P\left(\chi = \boldsymbol{v} | \tau \notin \bigcup_{X \in \mathcal{X}} X\right) P\left(\tau \notin \bigcup_{X \in \mathcal{X}} X\right).$$

Protože τ má uniformní rozdělení na $W_t \cap L$, platí:

$$P(\tau \in X) = \frac{k}{t^n}.$$

Dále platí:

$$P(\chi = \boldsymbol{v} | \tau \in X) = \frac{P(\chi = \boldsymbol{v}, \tau \in X)}{P(\tau \in X)} = \frac{P(\tau = \acute{\boldsymbol{v}})}{P(\tau \in X)} = \frac{\frac{1}{t^n}}{\frac{k}{t^n}} = \frac{1}{k},$$

kde $\acute{\boldsymbol{v}}$ je dáno jednoznačně vztahem $\acute{\boldsymbol{v}} \in X$ a $\acute{\boldsymbol{v}}_{(\mathrm{mod}\, \boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)} = \boldsymbol{v}$. Potom platí:

$$P(\chi = \boldsymbol{v}) = |\mathcal{X}|\, \frac{1}{k} \frac{k}{t^n} + \delta_{\boldsymbol{v}},$$

pro nějaké $\delta_{\boldsymbol{v}}$ splňující:

$$0 \leq \delta_{\boldsymbol{v}} = P\left(\chi = \boldsymbol{v} | \tau \notin \bigcup_{X \in \mathcal{X}} X\right) P\left(\tau \notin \bigcup_{X \in \mathcal{X}} X\right) \leq P\left(\tau \notin \bigcup_{X \in \mathcal{X}} X\right) = \frac{z}{t^n}.$$

Nyní odhadněme výraz $\left|P(\chi=\boldsymbol{v})-\frac{1}{k}\right|$:

$$\left|P(\chi=\boldsymbol{v})-\frac{1}{k}\right|=\left||\mathcal{X}|\frac{1}{k}\frac{k}{t^n}+\delta_{\boldsymbol{v}}-\frac{1}{k}\right|=\left|\delta_{\boldsymbol{v}}+\frac{1}{kt^n}\left(k|\mathcal{X}|-t^n\right)\right|\overset{2.2}{=}\left|\delta_{\boldsymbol{v}}-\frac{z}{kt^n}\right|.$$

Protože $0\leq\delta_{\boldsymbol{v}}\leq\frac{z}{t^n}$, dostáváme:

$$\left|P(\chi=\boldsymbol{v})-\frac{1}{k}\right|\leq\frac{z}{t^n},$$

a tedy:

$$\sum_{\boldsymbol{v}\in L\cap\mathcal{P}^-(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)}\left|P(\chi=\boldsymbol{v})-\frac{1}{k}\right|\leq\frac{kz}{t^n}.$$

Zbývá dokázat, že kz je dostatečně malé vůči t^n.

Nechť $\acute{W}_t$ je rovnoběžnostěn se stejným středem jako W_t, vzniklý z W_t zmenšením o faktor[1]

$$\gamma=1-\frac{2\sum_{i=1}^n\|\boldsymbol{a}_i\|}{\operatorname{minh}(W_t)}.$$

Potom pro minimální výšku $\operatorname{minh}\left(\acute{W}_t\right)$ platí:

$$\operatorname{minh}\left(\acute{W}_t\right)=\operatorname{minh}(W_t)-2\sum_{i=1}^n\|\boldsymbol{a}_i\|,$$

a tedy vzdálenost stěn W_t a $\acute{W}_t$ je alespoň $\sum_{i=1}^n\|\boldsymbol{a}_i\|$. Z toho plyne, že:

$$\acute{W}_t\subset\bigcup_{X\in\mathcal{X}}X.$$

Potom $|\mathcal{X}|\,k=|(\bigcup_{X\in\mathcal{X}}X)\cap L|\geq\left|\acute{W}_t\cap L\right|$. Dle lemmatu 12 dostáváme:

$$\begin{aligned}\left|\acute{W}_t\cap L\right|&\geq\left(1-\frac{2n\operatorname{bl}(L)}{\operatorname{minh}\left(\acute{W}_t\right)}\right)^n\frac{\operatorname{vol}\left(\acute{W}_t\right)}{\det(L)}=\left(1-\frac{2n\operatorname{bl}(L)}{t\operatorname{minh}(W_1)\gamma}\right)^n\frac{\operatorname{vol}(W_1)t^n\gamma^n}{\operatorname{vol}(W_1)}\\&=\left(1-\frac{2n\operatorname{bl}(L)}{t\operatorname{minh}(W_1)\left(1-\frac{2\sum_{i=1}^n\|\boldsymbol{a}_i\|}{t\operatorname{minh}(W_1)}\right)}\right)^n\left(1-\frac{2\sum_{i=1}^n\|\boldsymbol{a}_i\|}{t\operatorname{minh}(W_1)}\right)^n t^n.\end{aligned}$$

Z $\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\in\mathbb{Z}^n$ plyne $\det(L)=\operatorname{vol}(\mathcal{P}(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n))=|\det(\boldsymbol{d}_1|\ldots|\boldsymbol{d}_n)|\in\mathbb{N}$. Speciálně tedy $\operatorname{vol}(W_1)\geq1$. Potom:

$$\begin{aligned}\operatorname{minh}(W_1)=&\operatorname{minh}(\mathcal{P}(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n))=\min_{\{\boldsymbol{c}_1,\ldots,\boldsymbol{c}_{n-1}\}\subset\{\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\}}\frac{\operatorname{vol}(\mathcal{P}(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n))}{\operatorname{vol}(\mathcal{P}(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_{n-1}))}\\=&\frac{\operatorname{vol}(\mathcal{P}(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n))}{\max_{\{\boldsymbol{c}_1,\ldots,\boldsymbol{c}_{n-1}\}\subset\{\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\}}\operatorname{vol}(\mathcal{P}(\boldsymbol{c}_1,\ldots,\boldsymbol{c}_{n-1}))}\\\geq&\frac{1}{\prod_{i=1}^n\|\boldsymbol{d}_i\|}\geq2^{-nn^a}\geq2^{-n^{a+1}}.\end{aligned}$$

[1]Faktor je nezáporný, jak vyplyne dále z důkazu.

Dále platí $\mathrm{bl}(L) \leq \max_{i=1}^{n} \|\boldsymbol{d}_i\| \leq 2^{n^a}$. Potom:

$$\begin{aligned}\left|\acute{W}_t \cap L\right| &\geq \left(1 - \frac{2n \cdot 2^{n^a}}{t2^{-n^{a+1}}\left(1 - \frac{2n2^{n^a}}{t2^{-n^{a+1}}}\right)}\right)^n \left(1 - \frac{2n2^{n^a}}{t2^{-n^{a+1}}}\right)^n t^n \\ &= \left(1 - \frac{2n2^{n^a(n+1)}}{\left(2^{n^b}-1\right)\left(1 - \frac{2n2^{n^a(n+1)}}{2^{n^b}-1}\right)}\right)^n \left(1 - \frac{2n2^{n^a(n+1)}}{2^{n^b}-1}\right)^n t^n.\end{aligned}$$

Pro $n \geq 2, b \in \mathbb{N}$ platí: $2^{n^b} - 1 \geq 2^{n^b-1}$ a tedy:

$$\begin{aligned}&\left(1 - \frac{2n2^{n^a(n+1)}}{2^{n^b}-1}\right) \geq \left(1 - \frac{2n2^{n^a(n+1)}}{2^{n^b-1}}\right) = \left(1 - 4n2^{n^a(n+1)-n^b}\right) \\ &= \left(1 - 2^{n^a(n+1)+2+\log_2 n - n^b}\right) \geq \left(1 - 2^{n^a(n+n)+n+n-n^b}\right) \geq \left(1 - 2^{n^a 2n+2n-n^b}\right) \\ &\geq \left(1 - 2^{2n(n^a+1)-n^b}\right) \geq \left(1 - 2^{n^2 n^{a+1}-n^b}\right) \geq \left(1 - 2^{n^{a+3}-n^b}\right) \geq \frac{1}{2},\end{aligned}$$

kde poslední nerovnost lze zajistit dostatečně velkou volbou b, například $b = a+4$. Dále dostáváme:

$$\begin{aligned}\left|\acute{W}_t \cap L\right| &\geq \left(1 - \frac{2n2^{n^a(n+1)}}{2^{n^b-1}\frac{1}{2}}\right)^n \left(1 - \frac{2n2^{n^a(n+1)}}{2^{n^b-1}}\right)^n t^n \\ &\geq \left(1 - \frac{4n2^{n^a(n+1)}}{2^{n^b-1}}\right)^n \left(1 - \frac{4n2^{n^a(n+1)}}{2^{n^b-1}}\right)^n t^n \\ &= \left(1 - \frac{4n2^{n^a(n+1)}}{2^{n^b-1}}\right)^{2n} t^n.\end{aligned}$$

Z lemmatu 15 máme: $\forall x \in (0,1)$ a $n \in \mathbb{N}$:

$$(1-x)^n \geq 1 - nx$$

a tedy pro b dostatečně velké (například jako níže), aby $2^{n^b-1} \geq 4n2^{n^a(n+1)}$ máme:

$$\begin{aligned}\left|\acute{W}_t \cap L\right| &\geq \left(1 - 2n\frac{4n2^{n^a(n+1)}}{2^{n^b-1}}\right) t^n = \left(1 - \frac{16n^2 2^{n^a(n+1)}}{2^{n^b}}\right) t^n \\ &= \left(1 - 2^{n^a(n+1)+4+2\log_2 n - n^b}\right) t^n \geq \left(1 - 2^{n^a(n+n)+2n-n^b}\right) t^n \\ &= \left(1 - 2^{2n(n^a+1)-n^b}\right) t^n \geq \left(1 - 2^{n^2 n^{a+1}-n^b}\right) t^n \geq \left(1 - 2^{n^{a+3}-n^b}\right) t^n.\end{aligned}$$

Zvolíme-li opět $b = a + 4$, dostaneme:

$$\begin{aligned}\left|\acute{W}_t \cap L\right| &\geq \left(1 - 2^{n^{a+3}-n^{a+4}}\right) t^n = \left(1 - 2^{n^{a+3}(1-n)}\right) t^n \geq \left(1 - 2^{-n^{a+3}}\right) t^n \\ &\geq \left(1 - 2^{-2n^{a+1}}\right) t^n.\end{aligned}$$

Prozatím máme:

1. $t^n = |\mathcal{X}|\, k + z$
2. $\sum_{\boldsymbol{v} \in L \cap \mathcal{P}^-(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)} \left|P(\chi = \boldsymbol{v}) - \frac{1}{k}\right| \leq \frac{kz}{t^n}$
3. $|\mathcal{X}|\, k = |(\bigcup_{X \in \mathcal{X}} X) \cap L| \geq \left|\acute{W}_t \cap L\right| \geq t^n \left(1 - 2^{-2n^{a+1}}\right)$

Celkem tedy:

$$\sum_{\boldsymbol{v}\in L\cap\mathcal{P}^-(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)}\left|P\left(\chi=\boldsymbol{v}\right)-\frac{1}{k}\right|\leq\frac{k}{t^n}\left(t^n-\left|\mathcal{X}\right|k\right)\leq\frac{k}{t^n}\left(t^n-t^n\left(1-2^{-2n^{a+1}}\right)\right)$$
$$=k2^{-2n^{a+1}}$$

Protože $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\in L\left(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\right)$, potom:

$$\left|\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)\cap L\right|=\frac{\mathrm{vol}\left(\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)\right)}{\det\left(L\right)}$$

Výše jsme odhadli, že $\det\left(L\right)\geq 1$ a tedy:

$$k=\left|\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)\cap L\right|=\frac{\mathrm{vol}\left(\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)\right)}{\det\left(L\right)}\leq\mathrm{vol}\left(\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)\right)$$
$$\leq\prod_{i=1}^{n}\left\|\boldsymbol{a}_i\right\|=2^{nn^a}=2^{n^{a+1}}.$$

a tedy

$$\sum_{\boldsymbol{v}\in L\cap\mathcal{P}^-(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)}\left|P\left(\chi=\boldsymbol{v}\right)-\frac{1}{k}\right|\leq 2^{n^{a+1}}2^{-2n^{a+1}}=2^{-nn^a}\leq 2^{-n^a}.$$

□

Lemma 16 nám dává následující algoritmus pro výběr náhodného vektoru z $L\left(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\right)\cap\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)$:

Algoritmus 4: Náhodný vektor

Input : $a\in\mathbb{N},\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\in\mathbb{Z}^n$ LN, $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\in L\left(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\right)$ LN, $\left\|\boldsymbol{a}_i\right\|\leq 2^{n^a},\left\|\boldsymbol{d}_i\right\|\leq 2^{n^a}$
Output: $\boldsymbol{b}\in L\left(\boldsymbol{d}_1,\ldots,\boldsymbol{d}_n\right)\cap\mathcal{P}^-\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)$
$b=a+4$
vyber r_i uniformně náhodně z $\{0,1,2\ldots,2^{n^b}-1\}\forall i=1,\ldots,n$
$\boldsymbol{c}=\sum_{i=1}^{n}r_i\boldsymbol{d}_i$
$\boldsymbol{b}=c_{(\mathrm{mod}\,\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)}$
return $\boldsymbol{b}$;

Lemma 17. *Složitost algoritmu 4 je polynomiální v n^a.*

Důkaz. Nejprve si všimneme, že z $\left\|\boldsymbol{a}_i\right\|\leq 2^{n^a}$ plyne $\mathrm{size}\left(\boldsymbol{a}_i\right)\leq n\cdot n^a$, protože z $\left\|\boldsymbol{a}_i\right\|\leq 2^{n^a}$ plyne, že každá složka vektoru $\boldsymbol{a}_i$ musí být v absolutní hodnotě menší jak 2^{n^a}.

V algoritmu celkem n-krát vybereme náhodné číslo délky $n^b=\mathcal{O}\left(n^a\right)$, dále provádíme operace na číslech či vektorech velikosti menší než $n\left(n^a+n^b\right)$, tedy vše lze spočítat v čase polynomiálním v n^a. □

Následuje nejdůležitější věta pro naši redukci. Máme-li k dispozici orákulum, které řeší SIS, potom dokážeme z lineárně nezávislých vektorů mřížky vyrobit jiné lineárně nezávislé vektory s menší normou. Důkaz vychází z (Ajtai, 1996, lemma 13), navíc jsme zde provedli diskuzi konstant.

Věta 18. *Existují konstanty $c_1, c_2, c_3 \in \mathbb{N}$ takové, že pro všechna $n \in \mathbb{N}$ platí:*

Nechť existuje pravděpodobnostní polynomiální algoritmus $\mathcal{A}$, který na vstupu dostane matici $B \in \mathbb{Z}_{n^{c_2}}^{n \times \lceil c_1 n \log_2 n \rceil}$ a s pravděpodobností alespoň $1/2$ v průměrném případě vrátí vektor $\boldsymbol{v} \in \mathcal{L}_q^{\perp}(B)$, pro který $\|\boldsymbol{v}\|_1 \leq \lceil c_1 n \log_2 n \rceil$.

Potom existuje pravděpodobnostní algoritmus $\mathcal{C}$, který na vstupu dostane lineárně nezávislé vektory $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{Z}^n$ a lineárně nezávislé vektory $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$, $\|\boldsymbol{u}_1\| \leq \|\boldsymbol{u}_2\| \leq \cdots \leq \|\boldsymbol{u}_n\|$, $\|\boldsymbol{u}_n\| > n^{c_3} \operatorname{bl}(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$. Výstupem algoritmu $\mathcal{C}$ je vektor $\boldsymbol{g} \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ lineárně nezávislý na vektorech $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n-1}$ a pro který platí: $\|\boldsymbol{g}\| \leq \frac{1}{2}\|\boldsymbol{u}_n\|$.

Označíme-li $\sigma = \sum_{i=1}^{n} (\operatorname{size}(\boldsymbol{a}_i) + \operatorname{size}(\boldsymbol{u}_i))$, pak složitost algoritmu je polynomiální v σ. Pravděpodobnost úspěchu algoritmu $\mathcal{C}$ je alespoň $\frac{1}{3}$ v nejhorším případě.

Poznámka 5. Chceme-li důkaz provést tak, jak je věta formulovaná, pro všechna $n \in \mathbb{N}$, potom konstanty vyjdou následovně: $c_1 = 10, c_2 = 8, c_3 = 14$.

Důkaz lze však upravit tak, aby věta platila pro $n \geq n_0, n_0 \geq 3$. Potom konstanty vyjdou menší. V tabulce uvádíme některé hodnoty, které lze získat mírnou modifikací důkazu.

n_0	c_1	c_2	c_3
3	10	8	14
5	8	7	12
16	7	6	10
20	7	6	9
700	7	6	8
35000	6	5	7

Tabulka 2.1: Alternativní konstanty

Důkaz. Pro $n = 2$ máme k dispozici deterministický algoritmus 2: Gaussovu redukci mřížky. V celé další části důkazu budeme uvažovat, že $n \geq 3$. Začneme popisem algoritmu $\mathcal{C}$.

Pomocí lemmatu 11 aplikovaného na vektory $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n$ a $M = \max_{i=1}^{n} \|\boldsymbol{u}_i\| = \|\boldsymbol{u}_n\|$ dostaneme vektory $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n \in L(\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n) \subset L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$, pro které platí:

1. $\left(n^3 - \frac{1}{2}n\right) M \leq \|\boldsymbol{v}_i\| \leq \left(n^3 + \frac{1}{2}n\right) M, i = 1, \ldots, n$
2. $\frac{8}{27}(n^3 M)^n \leq \operatorname{vol}(\mathcal{P}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)) \leq 3(n^3 M)^n$
3. $\frac{2}{3}n^3 M \leq \operatorname{minh}(\mathcal{P}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n))$.
4. $\mathcal{P}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n) \subset K$, kde K je krychle o hraně délky $(n^3 + n^2) M$.

V důkazu budeme používat následující značení: $q = n^{c_2}$ a $m = \lceil c_1 n \log_2 n \rceil$, $L = L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$, $B = \operatorname{bl}(L)$, $M = \max_{i=1}^{n} \|\boldsymbol{u}_i\| = \|\boldsymbol{u}_n\|$. Dále budeme značit $V = \operatorname{vol}(\mathcal{P}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n))$, $S = \operatorname{sur}(\mathcal{P}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n))$, $H = \operatorname{minh}(\mathcal{P}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n))$. V tomto důkazu uvažujeme, že konstanty nabývají následujících hodnot: $c_1 = 10, c_2 = 8, c_3 = 14$.

Pomocí náhodné veličiny χ z lemmatu 16 vybereme náhodný vektor $\boldsymbol{w} \in \mathcal{P}^{-}(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n) \cap L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$. Z $\boldsymbol{w} \in \mathcal{P}^{-}(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n)$ dostáváme, že $\boldsymbol{w}$ lze zapsat jako $\boldsymbol{w} = \sum_{j=1}^{n} \alpha_j \boldsymbol{v}_j$, $\alpha_j \in [0,1)$. Označme $t_j = \lfloor q\alpha_j \rfloor$ a $\boldsymbol{t} = (t_1,\ldots,t_n) \in \mathbb{Z}_q^n$. Dále položme $\boldsymbol{r} = \boldsymbol{w} - \left(\sum_{j=1}^{n} \frac{t_j}{q} \boldsymbol{v}_j\right)$.

Opakováním tohoto postupu m-krát dostaneme vektory $\boldsymbol{w}_i$, $\boldsymbol{t}_i = (t_{i1},\ldots,t_{in})$ a $\boldsymbol{r}_i$, $i = 1,\ldots,m$.

Dále aplikujeme algoritmus $\mathcal{A}$ na mřížku $\mathcal{L}_q^{\perp}(T)$, kde $T = (\boldsymbol{t}_1|\ldots|\boldsymbol{t}_m)$. Výstupem je buď vektor $\boldsymbol{s} = (s_1,\ldots,s_m)$ nebo algoritmus neuspěl a pak položíme $s = (0,\ldots,0)$. Výstupem algoritmu $\mathcal{C}$ je vektor $\boldsymbol{g} = \sum_{i=1}^{m} s_i \boldsymbol{r}_i$.

Algoritmus 5: Algoritmus $\mathcal{C}$

Input : $\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n \in \mathbb{Z}^n$ LN, $\boldsymbol{u}_1,\ldots,\boldsymbol{u}_n \in L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$ LN, $\|\boldsymbol{u}_1\| \leq \|\boldsymbol{u}_2\| \leq \cdots \leq \|\boldsymbol{u}_n\|$

Output: $\boldsymbol{g} \in L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$, $\|\boldsymbol{g}\| \leq \frac{\|\boldsymbol{u}_n\|}{2}, \boldsymbol{g} \notin \langle \boldsymbol{u}_1,\ldots,\boldsymbol{u}_{n-1} \rangle$

Vytvoř z vektorů $\boldsymbol{u}_1,\ldots,\boldsymbol{u}_n$ vektory $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n$; `// pomocí algoritmu 3`

for *i=1,...,m* **do**

  Vyber náhodný vektor $\boldsymbol{w}_i \in \mathcal{P}^{-}(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n) \cap L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$; `// pomocí algoritmu 4`

  Zapiš $\boldsymbol{w}_i$ jako $\boldsymbol{w}_i = \sum_{j=1}^{n} \alpha_{ij} \boldsymbol{v}_j$, $\alpha_{ij} \in [0,1)$;

  $t_{ij} = \lfloor q\alpha_{ij} \rfloor, \forall j = 1,\ldots,n$;

  $\boldsymbol{t}_i = (t_{i1},\ldots,t_{in})$;

  $\boldsymbol{r}_i = \boldsymbol{w}_i - \left(\sum_{j=1}^{n} \frac{t_{ij}}{q} \boldsymbol{v}_j\right)$

end for

$T = (\boldsymbol{t}_1|\ldots|\boldsymbol{t}_m)$;

$\boldsymbol{s} = \mathcal{A}\left(\mathcal{L}_q^{\perp}(T)\right)$; `// Pokud` $\mathcal{A}$ `neuspěje, pak` $\boldsymbol{s} = 0$

$\boldsymbol{g} = \sum_{i=1}^{m} s_i \boldsymbol{r}_i$;

return $\boldsymbol{g}$;

Poznamenejme, že k tomu, aby měla instance problému pro algoritmus $\mathcal{A}$ řešení, potřebujeme dle lemmatu 10, aby $c_1 > c_2$ a $n^{c_2} = q > m$. Což platí pro všechny hodnoty z tabulky 2.1.

Tvrdíme, že pro $n \geq 3$ platí:

1. $\boldsymbol{g} \in L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$,
2. $\|\boldsymbol{g}\| \leq \frac{1}{2}\|\boldsymbol{u}_n\|$,
3. vektor $\boldsymbol{g}$ lineárně nezávislý na vektorech $\boldsymbol{u}_1,\ldots,\boldsymbol{u}_{n-1}$ s pravděpodobností alespoň $\frac{1}{3}$,
4. složitost algoritmu je polynomiální v σ,

potom je věta dokázána.

Nejprve dokážeme, že $\boldsymbol{g} \in L(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n)$. Ze vztahu $t_{ij} = \lfloor q\alpha_{ij} \rfloor$ dostáváme:

$$\alpha_{ij} = \frac{t_{ij}}{q} + r_{ij}, \text{pro nějaké } r_{ij} \in \left[0, \frac{1}{q}\right).$$

A tedy:

$$\boldsymbol{r}_i = \boldsymbol{w}_i - \left(\sum_{j=1}^{n} \frac{t_{ij}}{q} \boldsymbol{v}_j\right) = \sum_{j=1}^{n} \alpha_{ij} \boldsymbol{v}_j - \left(\sum_{j=1}^{n} \frac{t_{ij}}{q} \boldsymbol{v}_j\right) = \sum_{j=1}^{n} r_{ij} \boldsymbol{v}_j.$$

Uvažme nyní vektor

$$\boldsymbol{f} = \sum_{i=1}^{m} s_i \boldsymbol{w}_i,$$

protože $\boldsymbol{s} \in \mathbb{Z}^n$ a $\boldsymbol{w}_i \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ platí:

$$\boldsymbol{f} \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n).$$

Dále platí:

$$\boldsymbol{f} = \sum_{i=1}^{m} s_i \boldsymbol{w}_i = \sum_{i=1}^{m} s_i \sum_{j=1}^{n} \frac{t_{ij}}{q} \boldsymbol{v}_j + \sum_{i=1}^{m} s_i \boldsymbol{r}_i = \sum_{j=1}^{n} \sum_{i=1}^{m} s_i \frac{t_{ij}}{q} \boldsymbol{v}_j + \sum_{i=1}^{m} s_i \boldsymbol{r}_i$$

Protože $\boldsymbol{s}$ je výstup z algoritmu $\mathcal{A}$, platí:$\sum_{i=1}^{m} s_i \boldsymbol{t}_i \equiv \boldsymbol{0} \bmod q$, a tedy: $\sum_{i=1}^{m} s_i t_{ij} \equiv 0 \bmod q \ \forall j = 1, \ldots, n$. To spolu s $t_{ij} \in \{0,1, \ldots, q-1\}$ dává:

$$\sum_{i=1}^{m} s_i \frac{t_{ij}}{q} \in \mathbb{Z}$$

Celkem tedy máme:

$$\sum_{j=1}^{n} \sum_{i=1}^{m} s_i \frac{t_{ij}}{q} \boldsymbol{v}_j \in L(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n),$$

a protože $\boldsymbol{v}_i \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$, pak:

$$\sum_{j=1}^{n} \sum_{i=1}^{m} s_i \frac{t_{ij}}{q} \boldsymbol{v}_j \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n),$$

a tedy:

$$\boldsymbol{g} = \sum_{i=1}^{m} s_i \boldsymbol{r}_i \in L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n).$$

Dále dokážeme odhad délky vektoru $\boldsymbol{g}$. Z $r_{ij} \in \left[0, \frac{1}{q}\right)$ plyne:

$$\boldsymbol{r}_i \in \mathcal{P}^- \left(\frac{1}{q} \boldsymbol{v}_1, \ldots, \frac{1}{q} \boldsymbol{v}_n \right).$$

Ze vztahu $\boldsymbol{r}_i \in \mathcal{P}^- \left(\frac{1}{q} \boldsymbol{v}_1, \ldots, \frac{1}{q} \boldsymbol{v}_n \right)$ a $\mathcal{P}(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n) \subset K$, kde K je krychle o hraně délky $(n^3 + n^2) M$ dostáváme následující odhad:

$$\|\boldsymbol{r}_i\| \leq \left(n^3 + n^2\right) \frac{\sqrt{n} M}{q},$$

to spolu s $\|\boldsymbol{s}\|_1 \leq m = \lceil c_1 n \log_2 n \rceil$ dává:

$$\begin{aligned} \left\| \sum_{i=1}^{m} s_i \boldsymbol{r}_i \right\| &\leq \sum_{i=1}^{m} |s_i| \, \|\boldsymbol{r}_i\| \leq \sum_{i=1}^{m} |s_i| \left(n^3 + n^2\right) \frac{\sqrt{n} M}{q} \\ &= \lceil c_1 n \log_2 n \rceil \left(n^3 + n^2\right) \frac{\sqrt{n} M}{q}. \end{aligned}$$

Platí-li $n \geq 3, c_2 = 8$ a $c_1 = 10$, potom:[2]

$$\begin{aligned}\|\boldsymbol{g}\| &= \left\|\sum_{i=1}^{m} s_i \boldsymbol{r}_i\right\| \leq \lceil c_1 n \log_2 n \rceil \left(n^3 + n^2\right) \frac{\sqrt{n}M}{q} \\ &= (c_1 n \log_2 n + 1)\left(n^3 + n^2\right) \frac{\sqrt{n}M}{n^{c_2}} = (10n \log_2 n + 1)\left(\frac{1}{n^5} + \frac{1}{n^6}\right)\sqrt{n}M \\ &\leq (10 \cdot 3 \log_2 3 + 1)\left(\frac{1}{3^5} + \frac{1}{3^6}\right)\sqrt{3}M \leq 85\left(\frac{1}{3^5} + \frac{1}{3^6}\right)M \leq \frac{1}{2}M.\end{aligned}$$

čímž je dokázán odhad normy vektoru $\boldsymbol{g}$.

Nyní dokážeme odhad pravděpodobnosti. K tomu rozepišme $\boldsymbol{w}_i = \boldsymbol{\sigma}_i + \boldsymbol{r}_i$, kde $\boldsymbol{\sigma}_i = \sum_{j=1}^{n} \frac{t_{ij}}{q} \boldsymbol{v}_j$ a nahlížejme na $\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m$, $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m$, $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m$, $\boldsymbol{s}$ a $\boldsymbol{g}$ jako na hodnoty náhodných veličin. Tyto veličiny označme $\boldsymbol{\Sigma}_1, \ldots, \boldsymbol{\Sigma}_m$, $\boldsymbol{R}_1, \ldots, \boldsymbol{R}_m$, $\boldsymbol{W}_1, \ldots, \boldsymbol{W}_m$, $\boldsymbol{S} = (S_1, \ldots, S_m)$ a $\boldsymbol{G}$. Označme $F = \langle \boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n-1} \rangle$. Potřebujeme tedy ukázat, že:

$$P\left(\boldsymbol{G} \notin F\right) \geq \frac{1}{3}.$$

Důležitým krokem důkazu je pozorování, že výsledek algoritmu $\mathcal{A}$ nezávisí na vektorech $\boldsymbol{r}_i$, ale pouze na vektorech $\boldsymbol{\sigma}_i$. Platí tedy:

$$\begin{aligned}&P\left(\boldsymbol{S} = \boldsymbol{s} | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_m = \boldsymbol{r}_m\right) \\ =&P\left(\boldsymbol{S} = \boldsymbol{s} | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right),\end{aligned} \tag{2.3}$$

pro všechna $\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m$, $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_m$, $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_m$, $\boldsymbol{s}$, pro která jsou výše zmíněné pravděpodobnosti definovány. Obdobně také platí:

$$\begin{aligned}&P\left(\boldsymbol{S} = \boldsymbol{s} | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{R}_m \in F\right) \\ =&P\left(\boldsymbol{S} = \boldsymbol{s} | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right),\end{aligned} \tag{2.4}$$

pro F takové, že je první pravděpodobnost definována.

Při odhadu pravděpodobnosti $P\left(\boldsymbol{G} \notin F\right)$ můžeme uvažovat, že náhodné vektory $\boldsymbol{w}_i$ byly vybrány tak, že nejdříve byly vybrány vektory $\boldsymbol{\sigma}_i$, dále byl zavolán algoritmus $\mathcal{A}$ a nakonec vybrány vektory $\boldsymbol{r}_i \in \mathcal{P}^{-}\left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)$.

Dále uvažme, že máme vybrané $\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m$ a $\boldsymbol{s} = (s_1, \ldots, s_m)$ je výstup algoritmu $\mathcal{A}$. Uvažme navíc, že algoritmus $\mathcal{A}$ uspěl, tedy $\boldsymbol{s} \neq \vec{0}$. Označme k největší i takové, že $s_i \neq 0$. Bez újmy na obecnosti můžeme uvažovat, že $k = m$. Toho lze docílit příslušnou permutací vektorů $\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m$ a složek vektoru $\boldsymbol{s}$. Dále nechť máme vybrané i $\boldsymbol{r}_1, \ldots, \boldsymbol{r}_{m-1}$ a odhadněme pravděpodobnost $\boldsymbol{G} \in F$ za těchto podmínek:

$$\begin{aligned}&P\left(\boldsymbol{G} \in F | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\ =&P\left(\sum_{i=1}^{m} S_i \boldsymbol{R}_i \in F \middle| \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\ =&\frac{P\left(\sum_{i=1}^{m} S_i \boldsymbol{R}_i \in F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)} \\ =&\frac{P\left(\boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)},\end{aligned}$$

[2]Zde je třeba provést úpravy, chceme-li změnit konstanty dle tabulky 2.1. Jelikož pro celý výraz platí $\|\boldsymbol{g}\| \in \mathcal{O}\left(\frac{n^4 \sqrt{n} \log n}{n^{c_2}} M\right)$, nejmenší přípustná hodnota c_2 je 5.

kde $\boldsymbol{t} = \frac{-\sum_{i=1}^{m-1} s_i \boldsymbol{r}_i}{s_m}$. Dále předpokládejme, že

$$P\left(\boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right) > 0,$$

V případě, že je výše zmíněná pravděpodobnost rovna nule, potom:

$$P\left(\boldsymbol{G} \in F | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)$$

$$= \frac{P\left(\boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)}$$

$$= 0.$$

V případě nenulové pravděpodobnosti máme:

$$P\left(\boldsymbol{G} \in F | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)$$

$$= \frac{P\left(\boldsymbol{S} = \boldsymbol{s} | \boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)}$$
$$\cdot P\left(\boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)$$

$$\overset{2.4}{=} \frac{P\left(\boldsymbol{S} = \boldsymbol{s} | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)}$$
$$\cdot P\left(\boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)$$

$$= \frac{P\left(\boldsymbol{S} = \boldsymbol{s}, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)}$$
$$\cdot \frac{P\left(\boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right)}$$

$$= \frac{P\left(\boldsymbol{R}_m \in \boldsymbol{t} + F, \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)}{P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}\right)}$$

$$= \frac{P\left(\boldsymbol{W}_1 = \boldsymbol{w}_1, \ldots, \boldsymbol{W}_{m-1} = \boldsymbol{w}_{m-1}, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_m \in \boldsymbol{t} + F\right)}{P\left(\boldsymbol{W}_1 = \boldsymbol{w}_1, \ldots, \boldsymbol{W}_{m-1} = \boldsymbol{w}_{m-1}, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right)}.$$

Snadno nahlédneme, že platí:

$$\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m \& \boldsymbol{R}_m \in \boldsymbol{t} + F \Leftrightarrow$$
$$\boldsymbol{W}_m \in ((\boldsymbol{t} + \boldsymbol{\sigma}_m) + F) \cap \left(\boldsymbol{\sigma}_m + \mathcal{P}^- \left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)\right),$$

označme $\Omega = ((\boldsymbol{t} + \boldsymbol{\sigma}_m) + F) \cap \left(\boldsymbol{\sigma}_m + \mathcal{P}^- \left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)\right)$. Jelikož jsou náhodné

veličiny $\boldsymbol{W}_1,\ldots,\boldsymbol{W}_m$ nezávislé, platí:

$$
\begin{aligned}
&P\left(\boldsymbol{G}\in F|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)\\
=&\frac{P\left(\boldsymbol{W}_1=\boldsymbol{w}_1,\ldots,\boldsymbol{W}_{m-1}=\boldsymbol{w}_{m-1},\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{W}_m\in\Omega\right)}{P\left(\boldsymbol{W}_1=\boldsymbol{w}_1,\ldots,\boldsymbol{W}_{m-1}=\boldsymbol{w}_{m-1},\boldsymbol{W}_m\in\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)}\\
=&\frac{P\left(\boldsymbol{W}_1=\boldsymbol{w}_1\right)\cdot\ldots\cdot P\left(\boldsymbol{W}_{m-1}=\boldsymbol{w}_{m-1}\right)\cdot P\left(\boldsymbol{W}_m\in\Omega\right)}{P\left(\boldsymbol{W}_1=\boldsymbol{w}_1\right)\cdot\ldots\cdot P\left(\boldsymbol{W}_{m-1}=\boldsymbol{w}_{m-1}\right)\cdot P\left(\boldsymbol{W}_m\in\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)}\\
=&\frac{P\left(\boldsymbol{W}_m\in((\boldsymbol{t}+\boldsymbol{\sigma}_m)+F)\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right)}{P\left(\boldsymbol{W}_m\in\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)}.
\end{aligned}
$$

Protože je $\boldsymbol{W}_m$ skoro uniformní na $\mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\cap L\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)$ ve smyslu znění lemmatu 16, stačí nám k odhadu pravděpodobnosti odhadnout velikosti příslušných množin a poté přičíst, respektive odečíst celkovou odchylku:

$$
\begin{aligned}
&P\left(\boldsymbol{W}_m\in((\boldsymbol{t}+\boldsymbol{\sigma}_m)+F)\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right)\\
\leq&\frac{\left|L\cap((\boldsymbol{t}+\boldsymbol{\sigma}_m)+F)\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{\left|L\cap\mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\right|}+2^{-n^a}
\end{aligned}
$$

a

$$
P\left(\boldsymbol{W}_m\in\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\geq\frac{\left|L\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{\left|L\cap\mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\right|}-2^{-n^a},
$$

kde konkrétní hodnotu a určíme později. Celkem dostáváme:

$$
\begin{aligned}
&P\left(\boldsymbol{G}\in F|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)\\
\leq&\frac{\frac{\left|L\cap((\boldsymbol{t}+\boldsymbol{\sigma}_m)+F)\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{\left|L\cap\mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\right|}+2^{-n^a}}{\frac{\left|L\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{\left|L\cap\mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\right|}-2^{-n^a}}.
\end{aligned}
$$

Označme:

$$
\begin{aligned}
x&=\left|L\cap((\boldsymbol{t}+\boldsymbol{\sigma}_m)+F)\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|\\
y&=\left|L\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|\\
z&=\left|\mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\cap L\right|
\end{aligned}
$$

Zvolíme-li a dostatečně velké (hodnotu a upřesníme níže při důkazu, že algoritmus pracuje v polynomiálním čase), aby platilo:

$$
2^{-n^a}\leq\frac{y}{28z},\tag{2.5}
$$

dostaneme:

$$P\left(\boldsymbol{G}\in F|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)$$

$$\leq\frac{x+2^{-n^a}z}{y-2^{-n^a}z}\leq\frac{x+z\dfrac{y}{28z}}{y-z\dfrac{y}{28z}}=\frac{28x+y}{27y}=\frac{28x}{27y}+\frac{1}{27},$$

dosadíme-li zpět za x a y:

$$P\left(\boldsymbol{G}\in F|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)$$

$$\leq\frac{28\left|L\cap\left(\left(\boldsymbol{t}+\boldsymbol{\sigma}_m\right)+F\right)\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{27\left|L\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}+\frac{1}{27}.$$

Aplikováním lemmatu 14 na mřížku L, nadrovinu F, vektory $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n,\boldsymbol{\sigma}_m$ a $\boldsymbol{t}+\boldsymbol{\sigma}_m$ dostáváme:

$$\left|L\cap\left(\left(\boldsymbol{t}+\boldsymbol{\sigma}_m\right)+F\right)\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|\leq$$

$$\frac{8Bn}{q^{n-1}\det\left(L\right)}\left(n^3M\right)^{n-1}\left(1+\frac{2Bnq}{H}\right)^{n-1}$$

Dále z lemmatu 12 aplikovaného na mřížku L, vektory $\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n$ a $\boldsymbol{b}=\boldsymbol{\sigma}_m$ dostaneme:

$$\left|L\cap\left(\boldsymbol{\sigma}_m+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|\geq\left(1-\frac{2Bnq}{H}\right)^n\frac{V}{q^n\det\left(L\right)}.$$

A tedy:

$$P\left(\boldsymbol{G}\in F|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)$$

$$\leq\frac{28\frac{8Bn}{q^{n-1}}\left(n^3M\right)^{n-1}\left(1+\frac{2Bnq}{H}\right)^{n-1}}{27\left(1-\frac{2Bnq}{H}\right)^n\frac{V}{q^n}}+\frac{1}{27}=\frac{28\cdot8Bnq\left(n^3M\right)^{n-1}\left(1+\frac{2Bnq}{H}\right)^n}{27V\left(1+\frac{2Bnq}{H}\right)\left(1-\frac{2Bnq}{H}\right)^n}+\frac{1}{27}$$

$$\leq\frac{28\cdot8Bnq\left(n^3M\right)^{n-1}\left(1+\frac{2Bnq}{H}\right)^n}{27V\left(1-\frac{2Bnq}{H}\right)^n}+\frac{1}{27}$$

Z lemmatu 11 dostáváme odhad H, dále z předpokladu $M\geq n^{c_3}B, n\geq3$ máme:

$$\frac{2Bnq}{H}\leq\frac{2Bnn^{c_2}}{\frac{2}{3}n^3M}\leq\frac{2Bnn^{c_2}}{\frac{2}{3}n^3n^{c_3}B}=\frac{3n}{n^{c_3-c_2+3}}\leq\frac{n^2}{n^{c_3-c_2+3}}=\frac{1}{n^{c_3-c_2+1}},\tag{2.6}$$

dále máme odhad na $V\geq\frac{8}{27}\left(n^3M\right)^n$ a z předpokladu $M\geq n^3B$ dostaneme:

$$P\left(\boldsymbol{G}\in F|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)$$

$$\leq\frac{28\cdot8Bnn^{c_2}\left(n^3M\right)^{n-1}\left(1+\frac{1}{n^{c_3-c_2+1}}\right)^n}{27\frac{8}{27}\left(n^3M\right)^n\left(1-\frac{1}{n^{c_3-c_2+1}}\right)^n}+\frac{1}{27}$$

$$\leq\frac{28Bnn^{c_2}\left(1+\frac{1}{n^{c_3-c_2+1}}\right)^n}{n^3M\left(1-\frac{1}{n^{c_3-c_2+1}}\right)^n}+\frac{1}{27}\leq\frac{28Bnn^{c_2}\left(1+\frac{1}{n^{c_3-c_2+1}}\right)^n}{n^3n^{c_3}B\left(1-\frac{1}{n^{c_3-c_2+1}}\right)^n}+\frac{1}{27}$$

$$=\frac{28\left(1+\frac{1}{n^{c_3-c_2+1}}\right)^n}{n^{c_3-c_2+2}\left(1-\frac{1}{n^{c_3-c_2+1}}\right)^n}+\frac{1}{27}\leq\frac{28\left(1+\frac{1}{n^{c_3-c_2+1}}\right)^n}{n^{c_3-c_2+2}\left(1-\frac{1}{n^{c_3-c_2+1}}\right)^n}+\frac{1}{27}$$

[3] Jelikož je pro $c_2 = 8, c_3 = 14$ posloupnost $\left(1+\frac{1}{n^{c_3-c_2+1}}\right)^n = \left(1+\frac{1}{n^7}\right)^n$ klesající na $\mathbb{N}$ a obdobně $\left(1-\frac{1}{n^{c_3-c_2+2}}\right)^n = \left(1-\frac{1}{n^7}\right)^n$ je rostoucí na $\mathbb{N}\setminus\{1\}$ a za předpokladu $n \geq 3$ dostáváme následující odhad:

$$\begin{aligned} &P\left(\boldsymbol{G}\in F|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)\\ \leq&\frac{28\left(1+\frac{1}{3^7}\right)^3}{3^8\left(1-\frac{1}{3^7}\right)^3}+\frac{1}{27}\leq\frac{1}{20}. \end{aligned} \tag{2.7}$$

Připomeňme, že odhad výše platí pro $\boldsymbol{s}\neq\vec{0}$. Pro $\boldsymbol{s}=\vec{0}$ dostáváme:

$$P\left(\boldsymbol{G}\in F\middle|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\vec{0}\right)=1, \tag{2.8}$$

jelikož $\boldsymbol{G}=\vec{0}$.

Nechť $\mathcal{M}$ je množina všech $2m$-tic, kterých může $(\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_m,\boldsymbol{r}_1,\ldots,\boldsymbol{r}_{m-1},\boldsymbol{s})$ nabývat a pro kterou platí

$$P\left(\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)\neq 0,$$

pro každé $(\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_m,\boldsymbol{r}_1,\ldots,\boldsymbol{r}_{m-1},\boldsymbol{s})\in\mathcal{M}$. Dále rozdělme množinu $\mathcal{M}$ na množiny $\mathcal{M}_0$ a $\mathcal{M}_1$, kde prvky $\mathcal{M}_0$ mají v poslední složce nulový vektor $\boldsymbol{s}$ a prvky $\mathcal{M}_1$ nenulový vektor. Množinu $\mathcal{M}_0$ můžeme rozložit jako $\mathcal{M}_0=\mathcal{W}\times\{\vec{0}\}$ a množinu $\mathcal{W}$ dále rozložit jako

$$\mathcal{W}=\dot{\bigcup}_{(\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_m)\in\mathcal{S}}(\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_m)\times R_{(\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_m)}=\dot{\bigcup}_{\boldsymbol{\sigma}\in\mathcal{S}}\boldsymbol{\sigma}\times R_{\boldsymbol{\sigma}},$$

kde $(\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_m)$ budeme zkráceně zapisovat jako $\boldsymbol{\sigma}$, $\mathcal{S}$ je množina všech možných $\boldsymbol{\sigma}$, neboli

$$\mathcal{S}=\left\{\left(\sum_{i=1}^n\frac{t_{1i}}{q}\boldsymbol{v}_i,\ldots,\sum_{i=1}^n\frac{t_{mi}}{q}\boldsymbol{v}_i\right):t_{ij}\in\{0,1,\ldots,q-1\},i=1,\ldots,n,j=1,\ldots,m\right\}$$

a $R_{\boldsymbol{\sigma}}$ je množina všech možných $\boldsymbol{r}_1,\ldots,\boldsymbol{r}_{m-1}$ příslušných k $(\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\sigma}_{m-1})$, neboli

$$R_{\boldsymbol{\sigma}}=\left\{(\boldsymbol{r}_1,\ldots,\boldsymbol{r}_{m-1}):\boldsymbol{\sigma}_i+\boldsymbol{r}_i\in L\cap\left(\boldsymbol{\sigma}_i+\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right\}.$$

Odhadněme následující výraz:

$$\begin{aligned} c=&\sum_{\mathcal{M}_0}P\left(\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\boldsymbol{s}\right)\\ =&\sum_{\mathcal{W}}P\left(\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1},\boldsymbol{S}=\vec{0}\right)\\ =&\sum_{\mathcal{W}}P\left(\boldsymbol{S}=\vec{0}\middle|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1}\right)\\ &\cdot P\left(\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m,\boldsymbol{R}_1=\boldsymbol{r}_1,\ldots,\boldsymbol{R}_{m-1}=\boldsymbol{r}_{m-1}\right)\\ \overset{2.3}{=}&\sum_{\mathcal{W}}P\left(\boldsymbol{S}=\vec{0}\middle|\boldsymbol{\Sigma}_1=\boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m\right)\\ &\cdot P\left(\boldsymbol{W}_1=\boldsymbol{\sigma}_1+\boldsymbol{r}_1,\ldots,\boldsymbol{W}_{m-1}=\boldsymbol{\sigma}_{m-1}+\boldsymbol{r}_{m-1},\boldsymbol{\Sigma}_m=\boldsymbol{\sigma}_m\right). \end{aligned}$$

[3] Zde je třeba provést úpravy, chceme-li změnit konstanty dle tabulky 2.1. V případě $n_0 = 700$, je třeba navíc v nerovnosti 2.5 místo hodnoty 28 volit alespoň 100 a následně upravit nerovnosti v průběhu výpočtu. Odhad $\frac{1}{20}$ je pak třeba nahradit menší hodnotou.

Protože byly $\boldsymbol{w}_1,\ldots,\boldsymbol{w}_m$ vybrány nezávisle na sobě a $\mathcal{W}$ je disjunktní sjednocení $\mathcal{W} = \dot{\bigcup}_{\boldsymbol{\sigma}\in\mathcal{S}} \boldsymbol{\sigma} \times R_{\boldsymbol{\sigma}}$:

$$\begin{aligned}
c = &\sum_{\boldsymbol{\sigma}\in\mathcal{S}} \sum_{R_{\boldsymbol{\sigma}}} P\left(\boldsymbol{S} = \vec{0} \middle| \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \\
&\cdot P\left(\boldsymbol{W}_1 = \boldsymbol{\sigma}_1 + \boldsymbol{r}_1\right) \cdot \ldots \cdot P\left(\boldsymbol{W}_{m-1} = \boldsymbol{\sigma}_{m-1} + \boldsymbol{r}_{m-1}\right) \cdot P\left(\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \\
= &\sum_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{S} = \vec{0} \middle| \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \cdot P\left(\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \\
&\sum_{R_{\boldsymbol{\sigma}}} P\left(\boldsymbol{W}_1 = \boldsymbol{\sigma}_1 + \boldsymbol{r}_1\right) \cdot \ldots \cdot P\left(\boldsymbol{W}_{m-1} = \boldsymbol{\sigma}_{m-1} + \boldsymbol{r}_{m-1}\right) \\
= &\sum_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{S} = \vec{0} \middle| \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \cdot P\left(\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \\
&P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1\right) \cdot \ldots \cdot P\left(\boldsymbol{\Sigma}_{m-1} = \boldsymbol{\sigma}_{m-1}\right) \\
= &\sum_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{S} = \vec{0} \middle| \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \cdot P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \\
\leq &\max_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \sum_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{S} = \vec{0} \middle| \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \\
= &\max_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \sum_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\mathcal{A} \text{ neuspěl na vstupu } \boldsymbol{\sigma}\right) \\
\leq &\max_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right) \frac{q^{nm}}{2},
\end{aligned}$$

kde poslední nerovnost plyne z toho, že úspěšnost algoritmu $\mathcal{A}$ je alespoň $\frac{1}{2}$ v průměrném případě a $|\mathcal{S}| = q^{nm}$.

Dále odhadněme $\max_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1,\ldots,\boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right)$. K tomu pro $\boldsymbol{\sigma}_i$ spočteme $P\left(\boldsymbol{\Sigma}_i = \boldsymbol{\sigma}_i\right)$:

$$P\left(\boldsymbol{\Sigma}_i = \boldsymbol{\sigma}_i\right) = P\left(\boldsymbol{W}_i \in \left(\boldsymbol{\sigma}_i + \mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right)$$

Jelikož je $\boldsymbol{W}_i$ skoro uniformní na $\mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right) \cap L\left(\boldsymbol{a}_1,\ldots,\boldsymbol{a}_n\right)$ ve smyslu znění lemmatu 16, stačí nám k odhadu pravděpodobnosti odhadnout velikosti příslušných množin a poté přičíst celkovou odchylku:

$$P\left(\boldsymbol{W}_i \in \boldsymbol{\sigma}_i + \mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right) \leq \frac{\left|L \cap \left(\boldsymbol{\sigma}_i + \mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{\left|L \cap \mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\right|} + 2^{-n^a}.$$

Z lemmatu 12 aplikovaného na mřížku L a vektory $\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n$ dostáváme:

$$\left|L \cap \mathcal{P}^-\left(\boldsymbol{v}_1,\ldots,\boldsymbol{v}_n\right)\right| \geq \left(1 - \frac{2Bn}{H}\right)^n \frac{V}{\det\left(L\right)}$$

a

$$\left|L \cap \left(\boldsymbol{\sigma}_i + \mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1,\ldots,\frac{1}{q}\boldsymbol{v}_n\right)\right)\right| \leq \left(1 + \frac{2Bnq}{H}\right)^n \frac{V}{q^n \det\left(L\right)}.$$

Potom:

$$P\left(\boldsymbol{\Sigma}_i = \boldsymbol{\sigma}_i\right) \leq \frac{\left(1 + \frac{2Bnq}{H}\right)^n \frac{V}{q^n \det\left(L\right)}}{\left(1 - \frac{2Bn}{H}\right)^n \frac{V}{\det\left(L\right)}} + 2^{-n^a} \leq \frac{\left(1 + \frac{2Bnq}{H}\right)^n}{q^n \left(1 - \frac{2Bnq}{H}\right)^n} + 2^{-n^a}.$$

Dále z 2.6 máme:

$$P\left(\boldsymbol{\Sigma}_i = \boldsymbol{\sigma}_i\right) \leq \frac{\left(1+\frac{1}{n^{c_3-c_2+1}}\right)^n}{q^n\left(1-\frac{1}{n^{c_3-c_2+1}}\right)^n} + 2^{-n^a} = \frac{\left(n^{c_3-c_2+1}+1\right)^n}{\left(n^{c_3-c_2+1}-1\right)^n q^n} + 2^{-n^a}$$
$$= \left(1+\frac{2}{n^{c_3-c_2+1}-1}\right)^n \frac{1}{q^n} + 2^{-n^a}$$

Označme $d = c_3 - c_2 + 1$, potom dle tabulky 2.1 platí $d \leq 7$. Zvolíme-li $a \geq 4$ pak pro $n \geq 3$ a $c_2 \leq 8$ platí:

$$(d + c_2 n)\log_2 n \leq n^a,$$

potom

$$2^{-n^a} \leq 2^{-(d+c_2 n)\log_2 n} = n^{-(d+c_2 n)} = \frac{1}{n^d q^n},$$

a tedy

$$P\left(\boldsymbol{\Sigma}_i = \boldsymbol{\sigma}_i\right) \leq \left(1+\frac{2}{n^d-1}\right)^n \frac{1}{q^n} + \frac{1}{n^d q^n} \leq \frac{1}{q^n}\left(\left(1+\frac{2}{n^d-1}\right)^n + \frac{1}{n^d-1}\right).$$

Dále

$$\left(1+\frac{2}{n^d-1}\right)^n + \frac{1}{n^d-1} = 1 + n\frac{2}{n^d-1} + \sum_{i=2}^{n}\binom{n}{i}\left(\frac{2}{n^d-1}\right)^i + \frac{1}{n^d-1}$$
$$\leq 1 + n\frac{3}{n^d-1} + \sum_{i=2}^{n}\binom{n}{i}\left(\frac{2}{n^d-1}\right)^i \leq 1 + n\frac{3}{n^d-1} + \sum_{i=2}^{n}\binom{n}{i}\left(\frac{3}{n^d-1}\right)^i$$
$$= \left(1+\frac{3}{n^d-1}\right)^n$$

Protože vektory σ_i byly vybrány nezávisle na sobě, potom pravděpodobnost vybrané matice $(\boldsymbol{\sigma}_1|\ldots|\boldsymbol{\sigma}_m)$ je maximálně

$$\frac{1}{q^{nm}}\left(1+\frac{3}{n^d-1}\right)^{nm} = \frac{1}{q^{nm}}\left(1+\frac{3}{n^{c_3-c_2+1}-1}\right)^{n\lceil c_1 n\log_2 n\rceil},$$

a tedy:

$$c \leq \max_{\boldsymbol{\sigma}\in\mathcal{S}} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m\right)\frac{q^{nm}}{2}$$
$$\leq \frac{q^{nm}}{2}\frac{1}{q^{nm}}\left(1+\frac{3}{n^{c_3-c_2+1}-1}\right)^{n\lceil c_1 n\log_2 n\rceil} \leq \frac{1}{2}\left(1+\frac{3}{n^{c_3-c_2+1}-1}\right)^{n(c_1 n\log_2 n+1)}.$$

[4] Jelikož je pro zvolené konstanty $c_1 = 10, c_2 = 8$ a $c_3 = 14$ posloupnost $\left(1+\frac{3}{n^{c_3-c_2+1}-1}\right)^{n(c_1 n\log_2 n+1)} = \left(1+\frac{3}{n^7-1}\right)^{n(10n\log_2 n+1)}$ klesající na $\mathbb{N}\setminus\{1\}$ a za předpokladu $n \geq 3$ dostáváme následující odhad:

$$c \leq \frac{1}{2}\left(1+\frac{3}{3^7-1}\right)^{3(10\cdot 3\log_2 3+1)} \leq \frac{1.23}{2} \leq 0.62.$$

[4] Zde je třeba provést úpravy, chceme-li změnit konstanty dle tabulky 2.1. Jelikož pro celý výraz platí, že se asymptoticky chová jako $\mathcal{O}\left(\left(\frac{1}{n^{c_3-c_2+1}}\right)^{n^2\log n}\right)$, nejmenší přípustná hodnota $c_3 - c_2$ je 2.

Připomeňme, že

$$c = \sum_{\mathcal{M}_0} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right),$$

kde $\mathcal{M}$ je množina všech $2m$-tic, kterých může $(\boldsymbol{\sigma}_1, \ldots, \boldsymbol{\sigma}_m, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_{m-1}, \boldsymbol{s})$ nabývat a pro kterou platí

$$P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \neq 0,$$

Dále z $\mathcal{M} = \mathcal{M}_0 \dot{\bigcup} \mathcal{M}_1$ dostáváme, že:

$$\sum_{\mathcal{M}_1} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) = 1 - c$$

Nyní můžeme provést dokazovaný odhad pravděpodobnosti:

$$\begin{aligned}
& P\left(\boldsymbol{G} \in F\right) \\
=& \sum_{\mathcal{M}} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\
& \cdot P\left(\boldsymbol{G} \in F | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\
=& \sum_{\mathcal{M}_0} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\
& \cdot P\left(\boldsymbol{G} \in F | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\
& + \sum_{\mathcal{M}_1} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\
& \cdot P\left(\boldsymbol{G} \in F | \boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \\
\overset{2.7,\, 2.8}{\leq}& \sum_{\mathcal{M}_0} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \cdot 1 \\
& + \sum_{\mathcal{M}_1} P\left(\boldsymbol{\Sigma}_1 = \boldsymbol{\sigma}_1, \ldots, \boldsymbol{\Sigma}_m = \boldsymbol{\sigma}_m, \boldsymbol{R}_1 = \boldsymbol{r}_1, \ldots, \boldsymbol{R}_{m-1} = \boldsymbol{r}_{m-1}, \boldsymbol{S} = \boldsymbol{s}\right) \frac{1}{20} \\
=& c \cdot 1 + (1 - c) \frac{1}{20} = \frac{19c}{20} + \frac{1}{20} \leq 0.639 \leq \frac{2}{3},
\end{aligned}$$

čímž je dokázán odhad na $P\left(\boldsymbol{G} \notin F\right) \geq \frac{1}{3}$ a tedy i pravděpodobnost úspěchu algoritmu v nejhorším případě.

Nakonec dokážeme, že algoritmus pracuje v polynomiálním čase v σ, kde $\sigma = \sum_{i=1}^{n} \left(\text{size}\left(\boldsymbol{a}_i\right) + \text{size}\left(\boldsymbol{u}_i\right)\right)$. V prvním kroku vytvoříme vektory $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$. Ty lze dle lemmatu 11 spočítat v polynomiálním čase v $\sum_{i=1}^{n} \text{size}\left(\boldsymbol{u}_i\right)$, tedy i σ. Pro vektory $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n$ platí $\|\boldsymbol{v}_i\| \leq \left(n^3 + \frac{1}{2}n\right) M = \left(n^3 + \frac{1}{2}n\right) \max_{i=1}^{n} \|\boldsymbol{u}_i\|$ a tedy platí, že $\text{size}\left(\boldsymbol{v}_i\right)$ je polynomiální v $\text{size}\left(\boldsymbol{u}_i\right)$.

Dále m-krát provedeme smyčku for cyklu. Jelikož $m = \lceil c_1 n \log_2 n \rceil$ je polynomiální v n a $\text{size}\left(\boldsymbol{a}_i\right) \geq n$, je $m = \lceil c_1 n \log_2 n \rceil$ polynomiální v $\text{size}\left(\boldsymbol{a}_i\right)$ a tedy i v σ. Dále dokážeme, že každý krok ve smyčce má polynomiální složitost.

V for cyklu nejprve vybereme náhodný vektor $\boldsymbol{w}_i$. K odhadu složitosti tohoto kroku nejprve určíme hodnotu a z nerovnosti 2.5. Chceme tedy, aby platilo:

$$2^{-n^a} \leq \frac{y}{28z} = \frac{\left|L \cap \left(\boldsymbol{\sigma}_m + \mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{27\left|\mathcal{P}^-\left(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\right) \cap L\right|}.$$

Odhadněme tedy zespodu

$$\frac{\left|L \cap \left(\boldsymbol{\sigma}_m + \mathcal{P}^- \left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{28\left|\mathcal{P}^-\left(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\right) \cap L\right|}.$$

Z lemmatu 12 a 2.6 dostáváme:

$$\begin{aligned}\frac{\left|L \cap \left(\boldsymbol{\sigma}_m + \mathcal{P}^- \left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{28\left|\mathcal{P}^-\left(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\right) \cap L\right|} \geq & \frac{\left(1 - \frac{2Bnq}{H}\right)^n \frac{V}{q^n \det(L)}}{28\left(1 + \frac{2Bn}{H}\right)^n \frac{V}{\det(L)}} \geq \frac{\left(1 - \frac{2Bnq}{H}\right)^n}{28q^n\left(1 + \frac{2Bnq}{H}\right)^n} \\ \geq & \frac{\left(1 - \frac{1}{n^{c_3 - c_2 + 1}}\right)^n}{28q^n\left(1 + \frac{1}{n^{c_3 - c_2 + 1}}\right)^n}.\end{aligned}$$

[5] Jelikož je pro $c_2 = 8, c_3 = 14$ posloupnost $\left(1 + \frac{1}{n^{c_3 - c_2 + 1}}\right)^n = \left(1 + \frac{1}{n^7}\right)^n$ klesající na $\mathbb{N}$ a obdobně $\left(1 - \frac{1}{n^{c_3 - c_2 + 1}}\right)^n = \left(1 - \frac{1}{n^7}\right)^n$ je rostoucí na $\mathbb{N} \setminus \{1\}$ a za předpokladu $n \geq 3$ dostáváme následující odhad:

$$\begin{aligned}\frac{\left|L \cap \left(\boldsymbol{\sigma}_m + \mathcal{P}^- \left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)\right)\right|}{28\left|\mathcal{P}^-\left(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n\right) \cap L\right|} \geq & \frac{\left(1 - \frac{1}{3^7}\right)^3}{28q^n\left(1 + \frac{1}{3^7}\right)^3} \geq \frac{1}{32q^n} = \frac{1}{32n^{c_2 n}} \\ = & \frac{1}{32 \cdot 2^{c_2 n \log_2 n}} = \frac{1}{2^{5 + 8n \log_2 n}} \geq \frac{1}{2^{8n + 8n \log_2 n}} \geq \frac{1}{2^{8n^2}} \geq \frac{1}{2^{9n^2}} \geq 2^{-n^4}.\end{aligned}$$

Potřebujeme tedy, aby platilo $2^{-n^a} \leq 2^{-n^4}$, neboli $a \geq 4$. Dále, aby byly splněny předpoklady lemmatu 16, potřebujeme, aby $\|\boldsymbol{a}_i\| \leq 2^{n^a}$ a $\|\boldsymbol{v}_i\| \leq 2^{n^a}$. Zvolme tedy $a = \max\{4, \log_n \log_2 \|\boldsymbol{a}_i\|, \log_n \log_2 \|\boldsymbol{v}_i\|\}$. Výběr náhodného vektoru lze dle lemmatu 17 provést v polynomiálním čase v n^a. Protože je $\operatorname{size}(\boldsymbol{a}_i) \geq n$, je n^a polynomiální v $\operatorname{size}(\boldsymbol{a}_i)$, a tedy složitost výběru náhodného vektoru $\boldsymbol{w}_i$ je polynomiální v $\sum_{i=1}^n (\operatorname{size}(\boldsymbol{a}_i) + \operatorname{size}(\boldsymbol{v}_i))$. Jelikož vektor $\boldsymbol{w}_i$ leží v $\mathcal{P}^-(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$, máme $\operatorname{size}(\boldsymbol{w}_i)$ je polynomiální v $\sum_{i=1}^n \operatorname{size}(\boldsymbol{v}_i)$.

Následuje krok výpočtu α_{ij} tak, aby $\boldsymbol{w}_i = \sum_{j=1}^n \alpha_{ij} \boldsymbol{v}_j$. To lze dle lemmatu 1 spočítat v polynomiálním čase v $\sum_{i=1}^n \operatorname{size} \boldsymbol{v}_i + \operatorname{size} \boldsymbol{w}_i$ a tedy i v σ. Výsledné čitatele i jmenovatele koeficientů α_{ij} jsou opět polynomiálně velké v $\operatorname{size}(\boldsymbol{w}_i) + \sum_{i=1}^n \operatorname{size}(\boldsymbol{v}_i)$.

Další kroky for-cyklu jsou standardní operace na celých číslech nebo vektorech, pro které platí, že jejich velikost je polynomiální v σ. To lze vše provést v polynomiálním čase v σ.

Dále zavoláme algoritmus $\mathcal{A}$, který dle předpokladu pracuje v polynomiálním čase. Nakonec spočteme vektor $\boldsymbol{g}$, což lze opět provést v polynomiálním čase v σ. □

Poznámka 6. Z důkazu odhadu normy vektoru $\boldsymbol{g}$ plyne, že algoritmus $\mathcal{C}$ vrátí vždy vektor kratší jak $\frac{1}{2}\|\boldsymbol{u}_n\|$. Neúspěch tedy znamená, že algoritmus vrátí vektor lineárně závislý na vektorech $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_{n-1}$.

Následuje věta samotné redukce přibližného SBP_f na SIS.

[5] Zde je třeba provést úpravy, chceme-li změnit konstanty dle tabulky 2.1.

Věta 19. *Existují konstanty* $c_1, c_2, \acute{c}_3 \in \mathbb{N}$ *takové, že pro všechna* $n \in \mathbb{N}$ *platí:*

Nechť existuje pravděpodobnostní polynomiální algoritmus $\mathcal{A}$*, který na vstupu dostane matici* $B \in \mathbb{Z}_{n^{c_2}}^{n \times \lceil c_1 n \log_2 n \rceil}$ *a s pravděpodobností alespoň* $1/2$ *v průměrném případě vrátí vektor* $\boldsymbol{v} \in \mathcal{L}_q^{\perp}(B)$*, pro který* $\|\boldsymbol{v}\|_1 \leq \lceil c_1 n \log_2 n \rceil$*.*

Potom existuje pravděpodobnostní algoritmus $\mathcal{B}$*, který na vstupu dostane lineárně nezávislé vektory* $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{Z}^n$*, a výstupem tohoto algoritmu je báze* $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ *mřížky* $L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$ *splňující:* $\max_{i=1}^{n} \|\boldsymbol{b}_i\| \leq n^{\acute{c}_3} \operatorname{bl}(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$*.*

Označíme-li $\sigma = \sum_{i=1}^{n} \operatorname{size}(\boldsymbol{a}_i)$*, pak složitost algoritmu je polynomiální v* σ*. Pravděpodobnost úspěchu algoritmu* $\mathcal{B}$ *je alespoň* $1 - 2^{-\sigma}$ *v nejhorším případě.*

Poznámka 7. Obdobně, jako je uvedené v poznámce 5, lze provést důkaz tak, aby věta platila pro $n \geq n_0, n_0 \geq 3$. Konstanty $c_1, c_2, \acute{c}_3$ volíme jako v tabulce 2.1 z věty 18 s tím rozdílem, že $\acute{c}_3$ zvolíme o 1 větší než c_3:

n_0	c_1	c_2	$\acute{c}_3$
3	10	8	15
5	8	7	13
16	7	6	11
20	7	6	10
700	7	6	9
35000	6	5	8

Tabulka 2.2: Alternativní konstanty

Důkaz. Myšlenka algoritmu $\mathcal{B}$ je následující: začneme s vektory $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n = \boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$, které setřídíme vzestupně dle jejich normy a budeme stále dokola volat algoritmus $\mathcal{C}$. Z poznámky 6 dostáváme, že algoritmus $\mathcal{C}$ vrátí vždy vektor kratší než $\frac{\|\boldsymbol{c}_n\|}{2}$, avšak pravděpodobnost, že vektor $\boldsymbol{g}$ na výstupu bude lineárně nezávislý na $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{n-1}$, je alespoň $\frac{1}{3}$. Neúspěch tedy znamená, že vektor $\boldsymbol{g}$ je lineárně závislý na $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{n-1}$.

Pokud byl algoritmus $\mathcal{C}$ úspěšný, vektor $\boldsymbol{c}_n$ nahradíme výstupem algoritmu, vektory $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n$ znovu setřídíme dle normy a opět zavoláme algoritmus $\mathcal{C}$. Jestliže algoritmus $\mathcal{C}$ nebyl úspěšný, zkusíme ho zavolat znovu. Jestliže algoritmus $\mathcal{C}$ nebyl úspěšný k-krát v řadě za sebou, je velká pravděpodobnost, že vektory $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n$ nesplňují předpoklady algoritmu $\mathcal{C}$ (jejich norma je větší než $n^{c_3} \operatorname{bl}(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$, kde c_3 koresponduje s hodnotami v tabulce 2.1), a tedy už máme dostatečně krátké vektory. V takovém případě zastavíme volání algoritmu $\mathcal{C}$ a na vektory $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n$ zavoláme algoritmus Baze. Tento algoritmus vytvoří z lineárně nezávislých vektorů bázi mřížky $L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$, která je výstupem našeho algoritmu. Tato báze je nejvýše n-krát delší než $\max_{i=1}^{n} \|\boldsymbol{c}_i\|$, proto potřebujeme

$\acute{c_3} = c_3 + 1$. Dostáváme tak algoritmus $\mathcal{B}$.

Algoritmus 6: Algoritmus $\mathcal{B}$

Input : $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n \in \mathbb{Z}^n$ LN
Output: $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$ báze mřížky $L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$:
$\max_{i=1}^{n} \|\boldsymbol{b}_i\| \leq n^{\acute{c_3}} \operatorname{bl}(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$
$k = 3\sigma$
$\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n = \boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$
setřiď $\boldsymbol{c}_i$ podle normy
$i = 0$
$j = 0$; `// pomocná proměnná`
while $i \leq k$ **do**
  $\boldsymbol{g} = \mathcal{C}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_n)$
  if $\boldsymbol{g} \in \langle \boldsymbol{c}_1, \ldots, \boldsymbol{c}_{n-1} \rangle$ **then**
    i=i+1
  end if
  else
    $\boldsymbol{c}_n = \boldsymbol{g}$
    setřiď $\boldsymbol{c}_i$ podle normy
    $j = j + 1$
    $i = 0$
  end if
end while
$\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n = \text{Báze}(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n, \boldsymbol{c}_1, \ldots, \boldsymbol{c}_n)$
return $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_n$

Dále potřebujeme určit hodnotu k tak, aby pravděpodobnost úspěchu algoritmu $\mathcal{B}$ byla alespoň $1 - 2^{-\sigma}$. Neúspěch algoritmu nastane právě když skončí while cyklus, ale stále nemáme dostatečně krátké vektory $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n$.

Proměnná j v algoritmu $\mathcal{B}$ je pouze pomocná proměnná pro potřeby důkazu. Hodnota j vyjadřuje, kolikrát byl algoritmus $\mathcal{C}$ doposud úspěšný. Jestliže platí $\max_{i=1}^{n} \|\boldsymbol{c}_i\| > n^{c_3} \operatorname{bl}(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$, potom pravděpodobnost, že algoritmus $\mathcal{C}$ neuspěje k-krát v řadě za sebou (a tedy while cyklus skončí) je maximálně $\left(\frac{2}{3}\right)^k$. Odhad této pravděpodobnosti, za předpokladu $\max_{i=1}^{n} \|\boldsymbol{b}_i\| > n^{c_3} \operatorname{bl}(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$, je nezávislý na hodnotě j. Označme A_l jev, který nastane, pokud $j = l$, $i = k$ a $\max_{i=1}^{n} \|\boldsymbol{b}_i\| > n^{c_3} \operatorname{bl}(L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$, neboli po právě l úspěších algoritmu $\mathcal{C}$ nastane k neúspěchů algoritmu v řadě a while cyklus skončí. Potom pravděpodobnost jevu A_l lze odhadnout jevem, že $\mathcal{C}$ neuspěje k-krát v řadě za sebou:

$$P(A_l) \leq \left(\frac{2}{3}\right)^k$$

Celkový neúspěch algoritmu $\mathcal{B}$ je pak sjednocením jevů A_l. Označme m maximální hodnotu l, pro kterou může nastat jev A_l. Označíme-li $M = \max_{i=1}^{n} \|\boldsymbol{a}_i\|$, potom zřejmě $m \leq n \log_2 M$, jinak bychom měli, že norma jednoho z vektorů $\boldsymbol{c}_i \in \mathbb{Z}^n$ je menší jak 1. Nyní můžeme odhadnou pravděpodobnost neúspěchu algoritmu $\mathcal{B}$:

$$P(\mathcal{B} \text{ neuspěl}) \leq \sum_{j=0}^{m} P(A_j) \leq \sum_{j=0}^{m} \left(\frac{2}{3}\right)^k \leq \left(\frac{2}{3}\right)^k n \log_2 M.$$

Potřebujeme tedy, aby:

$$\begin{aligned}
\left(\frac{2}{3}\right)^k n \log_2 M &\leq 2^{-\sigma} \\
\left(\frac{2}{3}\right)^k &\leq 2^{-\sigma} \frac{1}{n \log_2 M} \\
k \log_2 \frac{2}{3} &\leq \log_2 \left(2^{-\sigma} \frac{1}{n \log_2 M}\right) \\
k &\geq \frac{-\log_2 \left(2^{\sigma} n \log_2 M\right)}{\log_2 \frac{2}{3}} \\
k &\geq \frac{\log_2 2^{\sigma} + \log_2 n + \log_2 \log_2 M}{\log_2 3 - \log_2 2} \\
k &\geq \frac{\sigma + \log_2 n + \log_2 \log_2 M}{\log_2 3 - 1}
\end{aligned}$$

Dále nahlédneme, že platí $M = \max_{i=1}^{n} \|\boldsymbol{a}_i\| \leq 2^{\sigma}$ a tedy $\log_2 \log_2 M \leq \log_2 \sigma$. Dále z $\operatorname{size}(\boldsymbol{a}_i) \geq n$ dostaneme $\sigma \geq n^2$. Potřebujeme tedy, aby

$$k \geq \frac{\sigma + \log_2 \sigma + \log_2 \sigma}{\log_2 3 - 1},$$

pro $n \geq 3$, a tedy $\sigma \geq 9$ stačí volit $k \geq 3\sigma$. Tím je dokázána pravděpodobnost úspěchu algoritmu.

Co se týče složitosti algoritmu, smyčka while cyklu proběhne celkem maximálně $k \cdot n \log_2 M$-krát, což je polynomiální v σ. Uvnitř cyklu voláme algoritmus $\mathcal{C}$, který pracuje v polynomiálním čase ve velikosti svého vstupu. Zde si všimneme, že velikost vstupu algoritmu $\mathcal{C}$ neroste, neboť norma vektorů $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n$ neroste. Dále v cyklu testujeme zda $\boldsymbol{g} \in \langle \boldsymbol{c}_1, \ldots, \boldsymbol{c}_{n-1} \rangle$, to lze otestovat například spočtením projekce a porovnáním zda se projekce rovná původnímu vektoru. To dle lemmatu 4 lze spočíst v polynomiálním čase. Dále v cyklu používáme standardní vektorové operace, které pracují v polynomiálním čase. Nakonec zavoláme algoritmus Baze, který pracuje v polynomiálním čase. □

Dokázali jsme polynomiální redukci přibližného SBP_f na SIS, kde v plné obecnosti (pro všechna $n \in \mathbb{N}$) jsme dostali aproximační faktor $f(n) = n^{15}$. Pro $n \geq 35000$ je možné aproximační faktor snížit na $f(n) = n^8$. Redukce byla následně několikrát vylepšena a je důležité zmínit, že autoři v důkazech používají asymptotické odhady, tedy jejich výsledky je třeba srovnávat s hodnotou 8 v našem důkazu.

V (Cai a Nerurkar, 1997) autoři dokázali redukci s $\acute{c}_3 = 3{,}5 + \epsilon$. Princip jejich redukce je stejný jako ve větě 18, vylepšení bylo dosaženo zejména v kroku výběru náhodných vektorů $\boldsymbol{w}_i \in \mathcal{P}^-(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n) \cap L(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$. Autoři zde zvolili jiný rovnoběžnostěn $\mathcal{P}^-(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$ (symetrický kolem počátku a menší) a tomu i přizpůsobili algoritmus výběru vektorů $\boldsymbol{w}_i$.

Věta 18 i výše zmíněné vylepšení k výběru uniformně náhodných vektorů $\boldsymbol{t}_i \in \mathbb{Z}_q^n, i = 1, \ldots, m$ využívají rovnoběžnostěn $\mathcal{P}^-(\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n)$, který následně rozdělí na q^n menších rovnoběžnostěnů $\mathcal{P}^-\left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)$. Dále se vybírají náhodné body

$\boldsymbol{w}_i \in \mathcal{P}^- (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n) \cap L (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n)$, které potom určí hodnoty $\boldsymbol{t}_i$ podle toho, do jakého menšího rovnoběžnostěnu spadají. $\boldsymbol{w}_i$ dále určí vektory $\boldsymbol{r}_i$, které jsou použity na sestrojení krátkého vektoru $\boldsymbol{g}$. Nyní máme dva protichůdné požadavky: za prvé chceme, aby byly rovnoběžnostěny $\mathcal{P}^- \left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)$ co nejmenší, protože potom dostaneme krátké vektory $\boldsymbol{r}_i$. Na druhou stranu potřebujeme mít rovnoběžnostěny $\mathcal{P}^- \left(\frac{1}{q}\boldsymbol{v}_1, \ldots, \frac{1}{q}\boldsymbol{v}_n\right)$ dostatečně velké vůči $\mathrm{bl}\,(L\,(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n))$, aby bylo zajištěno, že počet mřížových bodů v nich obsažených je zhruba stejný. Potom jsou $\boldsymbol{t}_i$ vybrány dostatečně uniformně z $\mathbb{Z}_q^n$ a dostaneme redukci na průměrný případ. Těmto protichůdným požadavkům se vyhnuli autoři v (Micciancio a Regev, 2007) odlišnou myšlenkou redukce. Celkem tak dosáhli konstanty $\acute{c}_3 = 1{,}5 + \epsilon$, což je dnes nejlepší známý výsledek.

3. Hashovací funkce

3.1 Úvod

Kryptografickou hashovací funkcí rozumíme zobrazení $h\colon D \to R$, kde D je množina všech možných vstupů a R je konečná množina všech možných výstupů. Množina D může být nekonečná, v případě konečné množiny požadujeme, aby $|D| > |R|$, potom má funkce h kompresní vlastnost a mimo jiné tak existují kolize.

Na kryptografické hashovací funkce máme tyto tři požadavky:

1. Odolnost vůči získání vzoru: pro daný obraz $r \in R$ je obtížné najít $d \in D$ takové, že $h(d) = r$.

2. Odolnost vůči získání jiného vzoru: pro daný vzor $d \in D$ je obtížné najít $e \in D$ takové, že $d \neq e$ a $h(d) = h(e)$.

3. Odolnost vůči získání kolize: je obtížné najít $d, e \in D$ takové, že $d \neq e$ a $h(d) = h(e)$.

Uvažme nyní, že pro každý obraz $r \in R$ existují alespoň dva vzory $d_1 \neq d_2 \in D$ takové, že $h(d_1) = h(d_2) = r$. Potom dostaneme, že požadavek 3 je nejslabším požadavkem. Skutečně, umíme-li efektivně řešit 1, potom s určitou pravděpodobností umíme řešit i 3: vybereme náhodný vstup $d_1 \in D$, spočteme $r = h(d_1)$ a z předpokladu, že umíme řešit 1, spočteme $d_2 \in D\colon h(d_2) = r$. Protože d_1 bylo vybráno náhodně a pro každé $r \in R$ existují alespoň dva vzory, s pravděpodobností alespoň $\frac{1}{2}$ platí $d_1 \neq d_2$.

Obdobně dostaneme, že umíme-li řešit 2, potom umíme řešit i 3. Chceme-li dokázat bezpečnost hashovací funkce, stačí dokázat, že je taková funkce odolná vůči kolizím. Potom funkce splňuje i body 1 a 2 za předpokladu, že pro každý obraz existuje více vzorů.

Hashovací funkce jsou často také definovány jako rodiny funkcí $\{h_k\colon k \in K\}$, kde každá funkce je parametrizována hodnotou (klíčem) $k \in K$. Od bezpečnosti takové rodiny hashovacích funkcí potom požadujeme, aby bylo obtížné najít kolize funkce h_k pro náhodně zvolené $k \in K$. Takovým funkcím se budeme dále věnovat.

3.2 Základní případ

Na základě složitosti problému SIS můžeme definovat rodinu hashovací funkcí, které jsou odolné vůči kolizím. Prvně zvolíme bezpečnostní parametr n, dále položme $m = \lceil c_1 n \log_2 n \rceil$ a $q = n^{c_2}$, kde c_1 a c_2 můžeme volit na základě n podle tabulky 2.1. Dále vybereme náhodnou matici $A \in \mathbb{Z}_q^{n \times m}$. Funkce potom vypadá následovně:

$$f_A : \mathbb{Z}_2^m \to \mathbb{Z}_q^n, \boldsymbol{x} \mapsto A\boldsymbol{x} \,(\mathrm{mod} q)\,.$$

Celkem dostáváme rodinu funkcí $\left\{f_A\colon A \in \mathbb{Z}_q^{n \times m}\right\}$.

Volbou $c_1 > c_2$ zajistíme, že platí:

$$2^m = 2^{\lceil c_1 n \log_2 n \rceil} > 2^{c_2 n \log_2 n} = n^{c_2 n} = q^n.$$

Z toho plyne, že takto definované funkce obsahují alespoň dva různé vstupy se stejným výstupem, neboli existují kolize. Najdeme-li pro $f_A, A \in \mathbb{Z}^{n\times m}$ kolizi, neboli $\boldsymbol{x}_1 \neq \boldsymbol{x}_2 \colon f_A(\boldsymbol{x}_1) = f_A(\boldsymbol{x}_2)$, potom:

$$f_A(\boldsymbol{x}_1 - \boldsymbol{x}_1) = A(\boldsymbol{x}_1 - \boldsymbol{x}_2) = A\boldsymbol{x}_1 - A\boldsymbol{x}_2 = \vec{0}.$$

Protože $\boldsymbol{x}_1 - \boldsymbol{x}_2 \in \{-1,0,1\}^m$, potom $\|\boldsymbol{x}_1 - \boldsymbol{x}_2\| \leq m$ a to spolu s $A(\boldsymbol{x}_1 - \boldsymbol{x}_2) = \vec{0}$ dává, že $\boldsymbol{x}_1 - \boldsymbol{x}_2$ je řešením instance problému SIS.

Máme-li tedy efektivní (polynomiální) algoritmus pro hledání kolizí pro výše definovanou rodinu hashovacích funkcí, potom máme efektivní algoritmus, který řeší náhodnou instanci problému SIS. Dle věty 19 potom máme efektivní algoritmus, který řeší přibližný SBP v nejhorším případě.

Předpokládáme-li naopak, že není možné efektivně řešit přibližný SBP pro daný aproximační faktor, potom z výše uvedeného plyne, že není možné efektivně hledat kolize pro výše definovanou rodinu hashovacích funkcí.

Dále se zaměříme na efektivitu takto definovaných funkcí. K vypočítání $f_A(\boldsymbol{x})$ potřebujeme spočíst $A\boldsymbol{x} \pmod q$, a protože $\boldsymbol{x} \in \{0,1\}^m$, potřebujeme pouze operace sčítání celých čísel modulo q. Celkem je potřeba tedy vynásobit matici a vektor $\boldsymbol{x} \in \{0,1\}^m$, což vyžaduje $\mathcal{O}(nm)$ operací. Paměťová náročnost spočívá zejména v načtení matice A do paměti, což vyžaduje $mn \log_2 q$ bitů. V dalších sekcích popíšeme varianty, které jsou časově i paměťově méně náročné.

3.3 Případ s využitím ideálních mřížek

Pro začátek potřebujeme mít definované f-ideální mřížky:

Definice 24. *Mějme $f \in \mathbb{Z}[x]$ monický polynom stupně n. Nahlížejme na mřížové body mřížky $L \subset \mathbb{Z}^n$ jako na polynomy stupně menšího než n, neboli uvažme zobrazení*

$$\Phi \colon \mathbb{Z}^n \to \mathbb{Z}[x]/(f)$$
$$(z_1, \ldots, z_n) \mapsto z_1 + z_2 x + \cdots + z_n x^{n-1}.$$

Mřížku L potom nazveme f-ideální, pokud $\Phi(L)$ je ideálem v $Z[x]/(f)$.

Dále pro monický polynom stupně n

$$h = x^n + h_n x^{n-1} + \cdots + h_2 x + h_1 \in \mathbb{Z}_q[x]$$

a $\boldsymbol{b} \in \mathbb{Z}_q^n$ definujme následující operaci:

$$H * \boldsymbol{b} = \left(\boldsymbol{b}, H\boldsymbol{b}, \ldots, H^{n-1}\boldsymbol{b}\right), \text{ kde } H = \begin{pmatrix} 0 & 0 & \ldots & 0 & 0 & -h_1 \\ 1 & 0 & \ldots & 0 & 0 & -h_2 \\ 0 & 1 & \ldots & 0 & 0 & -h_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 & -h_{n-1} \\ 0 & 0 & \ldots & 0 & 1 & -h_n \end{pmatrix}.$$

Nyní zafixujme monický polynom stupně n $h \in \mathbb{Z}_q[x]$. Rodinu hashovacích funkcí potom definujeme následovně: pro uniformně náhodné vektory $\boldsymbol{b}^{(1)}, \ldots, \boldsymbol{b}^{(m/n)} \in \mathbb{Z}_q^n$ vytvoříme matici $B \in \mathbb{Z}_q^{n\times m}$, kde $n|m$:

$$B = \left(H * \boldsymbol{b}^{(1)} | \ldots | H * \boldsymbol{b}^{(m/n)}\right).$$

Hashovací funkce potom vypadá analogicky, jako v předchozím případě: $f_B(\boldsymbol{x}) = B\boldsymbol{x} \,(\mathrm{mod} q)$.

Rodina funkcí potom vypadá následovně:

$$\left\{ f_B \colon B = \left(H * \boldsymbol{b}^{(1)} | \ldots | H * \boldsymbol{b}^{(m/n)} \right), \boldsymbol{b}^{(1)}, \ldots, \boldsymbol{b}^{(m/n)} \in \mathbb{Z}_q^n \right\}.$$

V (Lyubashevsky a Micciancio, 2006) autoři dokázali, že splňuje-li polynom h určité podmínky, potom je hledání kolizí pro takto definovanou rodinu funkcí alespoň tak těžké, jako problém SIS v f-ideálních mřížkách v aproximační verzi s polynomiálním aproximačním faktorem. Tyto podmínky zde specifikovat nebudeme, vystačíme si pouze s návrhem autorů na dva takové polynomy, které podmínky splňují pro daná n:

1. $h = x^n + 1$ a n je mocnina 2
2. $h = x^n + x^{n-1} + \cdots + x + 1$ a $n + 1$ je prvočíslo

Obdobně jako v případě obecných mřížek není znám žádný algoritmus, který by efektivně řešil přibližný SIS v f-ideálních mřížkách. Avšak na rozdíl od obecných mřížek chybí důkaz, že řešit (přesný) SIS v f-ideálních mřížkách je NP-těžké. I přes tento nedostatek panuje hypotéza, že neexistuje polynomiální algoritmus, který by řešil přibližný SIS v f-ideálních mřížkách.

3.4 SWIFFT

Rodina hashovacích funkcí SWIFFT (Lyubashevsky a kol., 2008) je speciálním případem popsaného v předchozí sekci. Autoři zvolili následující parametry: $n = 64, m = 1024, q = 257, h = x^n + 1$. Matice H potom vypadá následovně:

$$H = \begin{pmatrix} 0 & 0 & \ldots & 0 & 0 & -1 \\ 1 & 0 & \ldots & 0 & 0 & -0 \\ 0 & 1 & \ldots & 0 & 0 & -0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & 0 & -0 \\ 0 & 0 & \ldots & 0 & 1 & -0 \end{pmatrix},$$

a snadno nahlédneme, že pro vektor $\boldsymbol{b} = (b_1, \ldots, b_n) \in \mathbb{Z}_q^n$ platí:

$$H * \boldsymbol{b} = \begin{pmatrix} b_1 & -b_n & \ldots & -b_3 & -b_2 \\ b_2 & b_1 & \ldots & -b_4 & -b_3 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ b_{n-1} & b_{n-2} & \ldots & b_1 & -b_n \\ b_n & b_{n-1} & \ldots & b_2 & b_1 \end{pmatrix}.$$

K vytvoření hashovací funkce je tedy zapotřebí vybrat uniformně náhodné vektory $\boldsymbol{b}^{(1)}, \ldots, \boldsymbol{b}^{(16)} \in \mathbb{Z}_{257}^{64}$, tedy $1024 \log_2 257 \approx 8192$ bitů. Pro vstup $\boldsymbol{x} \in \{0,1\}^n$ je výstupem $\left(H * \boldsymbol{b}^{(1)} | \ldots | H * \boldsymbol{b}^{(m/n)} \right) \boldsymbol{x}$. Vstup má tedy $m = 1024$ bitů a výstup $n \log_2 q = 64 \log_2 257 \approx 512$ bitů.

Autoři také navrhli efektivní implementaci takto navržené funkce. Výpočet může být proveden v $\mathcal{O}(n \log n)$ čase za pomocí rychlé Fourierovy transformace

(FFT) a paměťová náročnost je $\mathcal{O}(m \log_2 q)$ bitů. Praktická implementace je pak rychlostí srovnatelná s funkcí SHA-1.

Parametry $n = 64, m = 1024, q = 257, h = x^n + 1$ se následně (Buchmann a Lindner, 2009) ukázaly jako nedostatečné. Ve stejném zdroji autoři navrhli jiné parametry, které mají zaručit dostatečnou bezpečnost.

Rodina funkcí SWIFFT je využita v hashovací funkci SWIFFTX. Ta je založena na HAIFA konstrukci[1], a tedy pracuje se vstupy délky až 2^{64} bitů. SWIFFT zde tvoří základní stavební kámen v kompresní funkci. Hashovací funkce SWIFFTX byla přijata jako kandidát na hashovací funkci SHA-3 v soutěži pořádané americkým Národním institutem standardů a technologie (NIST). Funkce však neprošla prvním kolem, neboť s původními parametry nesplňovala požadavky.

[1]HAIFA je alternativou k Merkle–Damgårdově konstrukci.

Závěr

V této práci jsme definovali základní pojmy a výpočetní problémy týkající se mřížek. Dále jsme uvedli čtenáře do problematiky úspěchu algoritmů v průměrném a v nejhorším případě a vysvětlili důležitost redukce typu nejhorší případ na průměrný případ, zejména pak využití v kryptografii. V kapitole 2 jsme popsali redukci SBP_f na SIS v detailní a matematicky korektní formě. Důkaz popsaný v této práci se od Ajtaiova důkazu v některých krocích odchyluje, avšak myšlenka zůstává stejná. Dále jsme určili hodnoty konstant $c_1, c_2, \acute{c}_3$ v aproximačních faktorech, zdůraznili, kde se tyto konstanty v důkazu uplatňují a provedli diskuzi, proč v tomto důkazu nelze volit konstanty menší. Nakonec jsme představili aplikace v kryptografii, konkrétně jak vytvořit hashovací funkce odolné vůči kolizím.

Tato práce může posloužit jako základ pro další diplomovou práci, která by se mohla zaměřit na další kryptografické aplikace, jejichž bezpečnost je založená na složitosti problémů na mřížkách a na redukci popsané v druhé kapitole. Takovými aplikacemi mohou být další kryptografická primitiva jako asymetrická šifra nebo podpisové schéma.

Seznam použité literatury

Ajtai, M. (1996). Generating hard instances of lattice problems (extended abstract). In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing*, STOC '96, pages 99–108, New York, NY, USA, 1996. ACM. ISBN 0-89791-785-5.

Ball, K. (1986). Cube slicing in Rn. *Proceedings of the American Mathematical Society*, **97**(3), 465–473. ISSN 00029939, 10886826.

Buchmann, J. a Lindner, R. (2009). Secure parameters for swifft. In *Proceedings of the 10th International Conference on Cryptology in India: Progress in Cryptology*, INDOCRYPT '09, pages 1–17, Berlin, Heidelberg, 2009. Springer-Verlag. ISBN 978-3-642-10627-9. doi: 10.1007/978-3-642-10628-6_1.

Cai, J.-y. a Nerurkar, A. P. (1997). An improved worst-case to average-case connection for lattice problems. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science*, FOCS '97, pages 468–, Washington, DC, USA, 1997. IEEE Computer Society. ISBN 0-8186-8197-7.

Dixon, J. D. (1982). Exact solution of linear equations using p-adic expansions. *Numer. Math.*, **40**(1), 137–141. ISSN 0029-599X.

Fang, X. G. a Havas, G. (1997). On the worst-case complexity of integer gaussian elimination. In *Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation*, ISSAC '97, pages 28–31, New York, NY, USA, 1997. ACM. ISBN 0-89791-875-4.

Lyubashevsky, V. a Micciancio, D. (2006). Generalized compact knapsacks are collision resistant. In *Proceedings of the 33rd International Conference on Automata, Languages and Programming - Volume Part II*, ICALP'06, pages 144–155, Berlin, Heidelberg, 2006. Springer-Verlag. ISBN 3-540-35907-9, 978-3-540-35907-4.

Lyubashevsky, V., Micciancio, D., Peikert, C. a Rosen, A. (2008). Swifft: A modest proposal for fft hashing. In *Fast Software Encryption, 15th International Workshop, FSE 2008, Lausanne, Switzerland, February 10-13, 2008, Revised Selected Papers*, volume 5086 of *Lecture Notes in Computer Science*, pages 54–72. Springer.

Micciancio, D. a Goldwasser, S. (2002). *Complexity of Lattice Problems : a Cryptographic Perspective*. Springer, Boston, MA. ISBN 978-1-4615-0897-7.

Micciancio, D. a Regev, O. (2007). Worst-case to average-case reductions based on gaussian measures. *SIAM J. Comput.*, **37**(1), 267–302. ISSN 0097-5397.

Micciancio, D. a Warinschi, B. (2001). A linear space algorithm for computing the hermite normal form. In *Proceedings of the 2001 International Symposium on Symbolic and Algebraic Computation*, ISSAC '01, pages 231–236, New York, NY, USA, 2001. ACM. ISBN 1-58113-417-7.

Nguyen, P. Q. a Stehlé, D. (2004). Low-dimensional lattice basis reduction revisited. In *Algorithmic Number Theory*, pages 338–357, Berlin, Heidelberg, 2004. Springer Berlin Heidelberg. ISBN 978-3-540-24847-7.