# Rácselmélet alkalmazása a számelméletben

Szakdolgozat

Radnai András

Témavezető: Grolmusz Vince
Számítógéptudományi tanszék

Eötvös Loránd Tudományegyetem
Természettudományi Kar

# Tartalomjegyzék

# Bevezetés

A szakdolgozatnak célja összefoglalni néhány rácselmélettel kapcsolatos eredményt. Maga a rács, melyet általában „$L$"-el (Lattice=Rács) jelölünk, definíció szerint az „$n$" dimenziós valós térnek egy diszkrét additív részcsoportja. Másképpen megfogalmazva néhány független vektor egészegyütthatós lineáris kombinációinak halmaza. Ezen vektorok által alkotott rendszert a rács bázisának nevezzük. Mint most is, ezentúl is vektorrendszer alatt kizárólag rendezett halmazra fogunk gondolni. Egy rács bázisának vektorai tetszőleges $SO_n(\mathbb{Z})$-beli transzformáció hatására ugyanannak a rácsnak bázisába mennek át, a fogalom tehát korántsem egyértelmű. A felmerülő problémák vizsgálatakor legtöbbször olyan bázist keresünk majd, melyben az adott rácsnak egy bizonyos tulajdonsága viszonylag átlátható.

Természetesen felmerül a kérdés, hogy milyen hosszú a legrövidebb nem-0 vektora egy rácsnak. Erre a Minkowski-tétel ad becslést, viszont nem ad konkrét rövid vektort. Kérdés marad, hogy van-e polinomiális idejű algoritmus, ami megadja a rács legrövidebb vektorát. Ez az egyik alapvető probléma, az SVP.(Shortest Vector Problem) mely a kérdés egyszerűségének ellenére meglepően nehéznek bizonyul: egy sor állítás áll rendelkezésre, mely különböző feltételek melletti NP-nehézségét bizonyítja a problémának. A legrövidebb vektor hosszát a továbbiakban $\lambda(L)$-lel jelöljük majd.

A kérdés 2-dimenziós esete már Gauss számára is felmerült, aki teljességgel megoldotta a problémát: az euklideszi algoritmus általánosításának tekinthető algoritmust adott, mely megtalálja a legrövidebb vektort. A végeredményként kapott bázist Gauss-redukáltnak nevezzük. Ez vázlatosan szerepel is az első fejezetben. Három dimenzióra még mindig átjátszható nagyobb nehézségek nélkül a két dimenziós bázisredukció.

Magasabb dimenziókban a Lovász László, Arjen Lenstra és Hendrik Lenstra által 1982-ben adott LLL-algoritmusnál nem ismert lényegesen jobban közelítő módszer. Látni fogjuk, hogy ennek ellenére a futásidejének polinomialitása egy folyamatosan csökkenő potenciál bevezetésével könnyen levezethető. A probléma bonyolultságát mutatja, hogy ez is a dimenzióban exponenciá-

lis hibakonstanssal dolgozik, mégis a gyakorlatban igen jól használhatónak mutatkozik.

[2]-ben több példa is meg említve van az algoritmus felhasználására. Egy egyszerűbb ilyen feladat például a szimultán approximáció, aminek lényege, hogy adott (az algoritmikusan kezelhetőség miatt racionális) számokat egyszerre „$\varepsilon$"-nál jobban közelítsünk minél kisebb közös nevezőjű törtekkel. Alkalmazható a nyilvános kulcsú RSA kódolás feltörésére is (természetesen csak nagyon erős plusz feltételek mellett).

A következő fejezetben megvizsgáljuk, hogy milyen közeli rácsvektort tudunk adni egy adott tetszőleges vektorra a Lovász-redukált bázis segítségével. A legközelebbi rácsvektor keresésének problémáját nevezzük CVP.(Closest Vector Problem)-nek. Ezek után az alapvető rácselméleti problémák bonyolultságelméleti nehézségéről teszünk néhány említést.

A prímfaktorizációval foglalkozó fejezet [4] nyomán egy kísérleti lehetőséget mutat egy adott nagy szám nem-triviális szorzattá bontására. Ennek alapja az ott részletezett Schnorr-algoritmus, aminek működéséhez szükséges inputot prímrácsokban keresett rövid, illetve közeli vektorok segítségével próbáljuk biztosítani.

# 1. Rácsalgoritmusok

## 1.1. Definíciók

Rácsnak nevezünk $\mathbb{R}^m$ egy független vektorrendszere által kifeszített additív részcsoportot:

**1.1.1. Definíció.** *Legyenek $b_1, b_2, \ldots, b_n \in \mathbb{R}^m$ lineárisan független vektorok ekkor*

$$L(b_1, b_2, \ldots, b_n) := \left\{ \sum_{i=1}^{n} \lambda_i b_i \,|\forall i\ \lambda_i \in \mathbb{Z} \right\}$$

*Itt „n"-et a rács rangjának, „m"-et a dimenziójának, $b_1, b_2, \ldots, b_n$ vektorokat pedig a bázisának nevezzük.*

Vegyük észre, hogy egy adott rácsnak több bázisa is lehet. A bázisvektorokból (mint oszlopvektorokból) álló $n \times m$-es mátrixot jelöljük $B := [\underline{b}_1, \underline{b}_2, \ldots, \underline{b}_n]$-vel! Nem fog félreértésekhez vezetni ha magát a bázist is egyszerűen B-vel jelöljük. Szükségünk lesz még egy fogalomra, a rács determinánsára.

**1.1.2. Definíció.** *Az L rács determinánsának a bázisvektorai által meghatározott Gram-mátrix determinánsának gyökét nevezzük:* $\det(L) := \sqrt{\det(B^T B)}$

Ez az érték megegyezik a bázisvektorok által kifeszített $\{\sum_{i=1}^{n} \lambda_i b_i \,|\forall i\ \lambda_i \in [0,1)\}$ (n-dimenziós) parallelepipedon, az úgynevezett alap blokk térfogatával. Mivel a rács maga mindig benne van egy n dimenziós altérben, ezért nyugodtan áttérhetünk az úgynevezett teljes rangú rácsok vizsgálatára, ahol a dimenzió és a rang megegyezik. Ilyen rácsoknál a determináns egyszerűbben felírható: $\det(L) := |\det(B)|$

Felmerül a kérdés, hogy ez a determináns valóban a rácsra jellemző mennyiség-e, vagy esetleg függhet a bázisvektorok választásától. Legyen B és B' két különböző bázisa a rácsunknak. Ez azt jelenti, hogy elemeik kölcsönösen felírhatók a másik elemeinek egész együtthatós lineáris kombinációjaként. Vagyis mátrixokra áttérve $\exists T \in \mathbb{Z}^{n\times n} \qquad BT = B' \quad \& \quad B'T^{-1} = B$ Mivel itt $T$ és $T^{-1}$ is egész számokból áll, determinánsuk is egész lesz, és tudjuk róluk, hogy reciprokai egymásnak. Következésképp $\det T = \pm 1$, tehát $|\det(B)| = |\det(B')|$, amivel megvan az egyértelműség.

## 1.2. Gram-Schmidt ortogonalizáció, gyenge redukció

Ismert a Gram-Schmidt ortogonalizációs eljárás, melynek lényege, hogy az adott $b_1, b_2, \ldots, b_n$ bázisból olyan merőleges $b_1^*, b_2^*, \ldots, b_n^*$ vektorokat csinálunk, melyekre igaz, hogy $\forall i \qquad b_i^* - b_i \in \langle b_1, b_2, \ldots, b_{i-1}\rangle$. Ekkor az eredeti bázisvektorokat felírhatjuk

$$b_i = b_i^* + \sum_{j=1}^{i-1} \mu_{i,j} b_j^*$$

alakban, ahol a $\eta_{i,,j}$ $(i > j)$ Gram-Schmidt együtthatók a következőek:

$$\mu_{i,,j} := \frac{\langle b_i, b_j^* \rangle}{\langle b_j^*, b_j^* \rangle} = \frac{\langle b_i, b_j^* \rangle}{\|b_j^*\|^2}$$

Kiterjesztjük a „$\mu$" együtthatók indextartományát a következő módon:

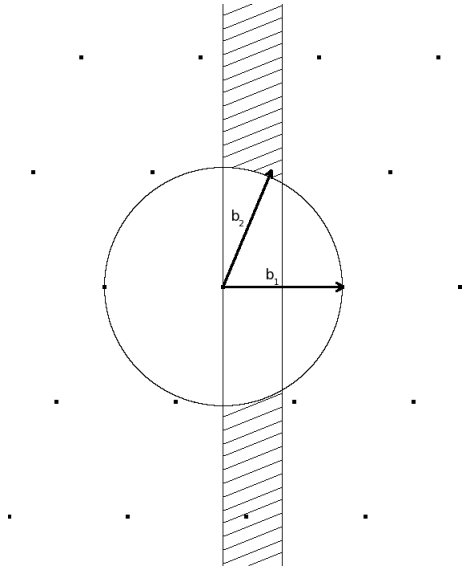
$$\mu_{i,j} := \begin{cases} 1 & , ha\ i = j \\ 0 & , ha\ i < j \end{cases}$$

Ekkor felírhatjuk a következő mátrix egyenletet: $B^* \cdot G = B$, ahol $G$ a $\mu_{j,i}$ együtthatókból kapott mátrix. Könnyen meggondolható, hogy mind $G$, mind $G^{-1}$ olyan felsőháromszögmátrixok, melyek főátlójában mindenhol egyes van.

A következőkben különböző olyan megkötéseket próbálunk tenni bázisokra, melyek segítenek áttekinthetővé tenni a rácsban lévő vektorokat. Egy dimenzióban a bázisunk előjel erejéig meghatározott. Két dimenzióban definiálhatjuk a Gauss-redukáltságot a következőképpen:

**1.2.1. Definíció.** *Azt mondjuk, hogy a $b_1, b_2$ bázis Gauss-redukált, amennyiben $\|b_1\| \leq \|b_2\|$, valamint $0 \leq \mu_{2,1} \leq 1/2$*

Ez a definíció azért kényelmes, mert van gyors (és egyszerű) algoritmus ilyen bázis előállítására. Könnyen meggondolható, hogy $b_1$ legrövidebb vektora lesz a rácsnak: $b_2$-nek az ábrán a satírozott területbe kell mutatnia.



Magasabb dimenziókban nem ismert ilyen erős és algoritmikusan polinomiális időben megvalósítható feltétel. A következő definíciót az alapján az észrevétel alapján vezethetjük be, hogy két dimenzióban a Gauss-redukáltság ereje abban rejlett, hogy a vektoraink egymásra $\pi/3$ és $\pi/2$ közötti szöget zárnak be benne. Ennek kiterjesztéseként általában azt követeljük meg, hogy páronként a bázisvektorok egymásra a „lehető legmerőlegesebbek" legyenek egymásra.

**1.2.2. Definíció.** *Egy $b_1, b_2, \ldots, b_n \in \mathbb{R}^n$ bázist gyengén redukáltnak nevezünk, amennyiben a Gram-Schmidt együtthatók kielégítik az alábbi feltételt:*

$$|\mu_{i,j}| \leq 1/2, \qquad \textit{ahol } 1 \leq j < i \leq n$$

A feltételnek eleget tevő bázis szintén polinomiális időben található egy egyszerű algoritmussal.

**1.2.3. Algoritmus.** *Iteráljuk ezt az egy lépést:*

*Ha nincs olyan (i,j) pár, melyekre $|\mu_{i,j}| > 1/2$, akkor leállunk. Egyébként pedig vegyük azt a párt, amelyikre „i" a legkisebb ilyen, és ezen belül „j" maximális. Erre az „i"-re és „j"-re:*

$b_i := b_i - \lceil \mu_{i,j} \rfloor \cdot b_j$

**1.2.4. Állítás.** *Ez az algoritmus polinomiális időben véget ér, és gyengén redukált bázist hagy végeredményül tetszőleges input bázis esetén.*

**Bizonyítás.** Az output redukáltsága triviális, amennyiben az algoritmus tényleg lefut. Ehhez azt fogjuk belátni, hogy egy lépésben egy együtthatót legfeljebb $1/2$ abszolút értékűre állítunk, és a korábban kijavítottakat nem rontjuk el. Mivel $b_1, b_2, \ldots, b_{i-1}$, valamint $b_1^*, b_2^*, \ldots, b_n^*$nem változnak az eljárás során, ezért az „i"-nél kisebb első indexű Gram-Schmidt együtthatók szintén változatlanok lesznek. A kérdés, hogy mi történhet egy $\mu_{i,k}$ számmal, ahol $j \leq k < i$.

$$\mu'_{i,k} := \frac{\langle b'_i, b_k^* \rangle}{\langle b_k^*, b_k^* \rangle} = \frac{\langle b_i - \lceil \mu_{i,j} \rfloor \cdot b_j, b_k^* \rangle}{\langle b_k^*, b_k^* \rangle} = \frac{\langle b_i, b_k^* \rangle}{\langle b_k^*, b_k^* \rangle} - \lceil \mu_{i,j} \rfloor \frac{\langle b_j, b_k^* \rangle}{\langle b_k^*, b_k^* \rangle}$$

Itt a bal oldali összeadandó maga $\mu_{i,k}$, a jobboldali a $j < k$ feltétel mellett 0, tehát

$$\mu'_{i,k} = \mu_{i,k}$$

$j = k$ esetben $\frac{\langle b_j, b_j^* \rangle}{\langle b_j^*, b_j^* \rangle} = 1$-ből azt kapjuk, hogy $\left|\mu'_{i,j}\right| = |\mu_{i,j} - \lceil \mu_{i,j} \rfloor| \leq 1/2$

## 1.3. Az LLL algoritmus

A Lovász-redukáltság előtt vezessünk be egy jelölést: adott $b_1, b_2, \ldots, b_n \in \mathbb{R}^n$ bázis mellett bármely $a \in \mathbb{R}^n$ vektor felbontható $a = a_1 + a_2$ vektorok összegére úgy, hogy $a_1 \in \langle b_1, b_2, \ldots, b_{i-1} \rangle \quad \& \quad a_2 \in \langle b_1, b_2, \ldots, b_{i-1} \rangle^{\perp}$, mivel ezek direkt kiegészítő alterek. Ekkor jelölje „$a\,(i)$" az $a_2$ komponenst!

**1.3.1. Definíció.** *Egy $b_1, b_2, \ldots, b_n \in \mathbb{R}^n$ bázist Lovász-redukáltnak nevezünk, ha egyrészt gyengén redukált, másrészt*

$$\|b_i\,(i)\|^2 \leq \frac{4}{3} \|b_{i+1}\,(i)\|^2 \qquad \textit{ahol } 1 \leq i < n$$

A Gram-Schmidt ortogonalizációnál felírt egyenletekből itt

$$b_i\,(i) = b_i^*$$

$$b_{i+1}\,(i) = b_{i+1}^* + \mu_{i+1,i} b_i^*$$

**1.3.2. Állítás.** *Legyen $b_1, b_2, \ldots, b_n \in \mathbb{R}^n$ Lovász-redukált bázisa az $L$ rácsnak. Ekkor teljesül az alábbi három állítás:*

$$(1) \qquad \|b_1\| \leq 2^{(n-1)/2} \lambda\,(L)$$

$$(2) \qquad \|b_1\| \leq 2^{(n-1)/4} \sqrt{\det\,(L)}$$

$$(3) \qquad \prod_{i=1}^{n} \|b_i\| \leq 2^{\binom{n}{2}/2} \det\,(L)$$

**1.3.3. Megjegyzés.** *Itt az első állításban szereplő $2^{(n-1)/2}$ együttható a dimenzióban exponenciális hibakonstans. Meglepő, hogy ennek ellenére az algoritmus jól használható. A harmadik állítás azt fejezi ki, hogy a célunk, nevezetesen, hogy minél merőlegesebb bázisvektorokat találjunk, mennyire van biztosítva: a bal oldal minél kisebb. Ideális esetben, vagyis ha a vektorok páronként*

*merőlegesek egymásra, természetesen ez pont a determinánssal egyenlő. Itt egy $2^{\binom{n}{2}/2}$-es hibataggal tudunk becsülni.*

**Bizonyítás.** Legyen $b_1, b_2, \ldots, b_n \in \mathbb{R}^n$ Lovász-redukált bázis! Ekkor teljesül az alábbi egyenlőtlenségrendszer

$$\|b_i^*\|^2 \leq \frac{4}{3}\left\|b_{i+1}^* + \mu_{i+1,i}b_i^*\right\|^2 = \frac{4}{3}\left\|b_{i+1}^*\right\|^2 + \frac{4}{3}\mu_{i+1,i}^2\left\|b_i^*\right\|^2 \leq \frac{4}{3}\left\|b_{i+1}^*\right\|^2 + \frac{1}{3}\left\|b_i^*\right\|^2$$

Innen átrendezéssel adódik az alábbi egyenlőtlenség:

$$\|b_i^*\|^2 \leq 2\left\|b_{i+1}^*\right\|^2$$

Amiből indukcióval látható, hogy

$$\|b_1\|^2 = \|b_1^*\|^2 \leq 2^{i-1}\left\|b_i^*\right\|^2$$

A bizonyítás során csak ezt fogjuk kihasználni, mint többletet a gyengén redukáltsághoz képest. Az eredeti felírás majd az algoritmus lépésszámának ellenőrzésénél lesz kényelmes. Ezeket az egyenlőtlenségeket összeszorozva $i = 1, \ldots, n$-re

$$\|b_1\|^{2n} \leq 2^{\binom{n}{2}}\prod_{i=1}^{n}\|b_i^*\|^2 = 2^{\binom{n}{2}}\det{}^2(L),$$

ami bizonyítja (2) állítást. A (3) bizonyításához elég belátnunk, hogy

$$\|b_i\| \leq 2^{i-1}\|b_i^*\|$$

Ezek szorzata adja az állítást. Ez $i = 1$-re triviális, nagyobb indexre pedig működik a következő érvelés:

$$\|b_i\|^2 = \left\|b_i^* + \sum_{j=1}^{i-1}\mu_{i,j}b_j^*\right\|^2 = \|b_i^*\|^2 + \sum_{j=1}^{i-1}\mu_{i,j}^2\left\|b_j^*\right\|^2 \leq \|b_i^*\|^2 + \sum_{j=1}^{i-1}\frac{1}{4}\left\|b_j^*\right\|^2 \leq$$

$$\leq \left(1 + \frac{1}{4}\left(\sum_{j=1}^{i-1}2^j\right)\right) \leq 2^{i-1}\|b_i^*\|$$

Hogy a legrövidebb vektor hosszára alsó becslést adhassunk, először belátjuk a következő lemmát:

**1.3.4. Lemma.** *A szokásos jelölések mellett*

$$\lambda(L) \geq \min_{1 \leq i \leq n} \{\|b_i^*\|\}$$

**Bizonyítás.**(lemma)Legyen $b \in L$ tetszőleges nem-0 rácsbeli vektor. Ekkor

$$b = \sum_{i=1}^{k} \lambda_i b_i$$

valamely $\lambda_1, \ldots, \lambda_k \in \mathbb{Z}$ számokra, ahol $k \leq n$ és $\lambda_k \neq 0$. Másrészt ugyanez a vektor felírható az ortogonalizált bázisban is

$$b = \sum_{i=1}^{k} \lambda_i' b_i^*$$

alakban, ahol a legnagyobb indexű együtthatókra teljesül, hogy $\lambda_k = \lambda_k'$ (az együtthatómátrix $G^{-1}$ inverzének az ortogonalizációnál megemlített tulajdonságaiból látszik). Így felírható, hogy

$$\|b\|^2 = \sum_{i=1}^{k} \lambda_i'^2 \|b_i^*\|^2 \geq \lambda_k'^2 \|b_k^*\|^2 \geq \|b_k^*\|^2,$$

ami bizonyítja a lemma állítását.

Ezzel

$$\|b_1\|^2 \leq \min_{1 \leq i \leq n} \left\{2^{i-1} \|b_i^*\|^2\right\} \leq 2^{n-1} \min_{1 \leq i \leq n} \left\{\|b_i^*\|^2\right\} \leq 2^{n-1} \lambda^2(L)$$

Ahonnan végül gyökvonással kapjuk az (1) állítást.

**1.3.5. Algoritmus.** *(Lovász-Lenstra-Lenstra)Az algoritmusban két lépést hajtunk végre felváltva: Az elsőben az aktuális bázisunkat átalakítjuk gyengén-redukálttá az előzőekben tárgyalt módon. A másodikban, ha találunk két szomszédos indexű bázisvektort, mely megszegi a Lovász-redukáltság feltételét, akkor ezeket felcseréljük.*

**1.3.6. Állítás.** *Legyen a $b_1, b_2, \ldots, b_n \in \mathbb{Q}^n$ bázis által generált rács $L$. Ekkor az LLL algoritmus polinomiális időben előállít egy Lovász-redukált bázist.*

**Bizonyítás.**Nyilvánvaló, hogy amennyiben az algoritmus valóban véget ér, egy Lovász-redukált bázist ad.

Az algoritmus lépésszámát egy egyszerű potenciálfüggvény bevezetésének segítségével fogjuk felülről becsülni.

$$D\left(b_1, b_2, \ldots, b_n\right) := \prod_{i=1}^{n} \left\|b_i^*\right\|^{n-i}$$

Ez a potenciál felírható az alábbi módon is:

$$D\left(b_1, b_2, \ldots, b_n\right) := \prod_{i=1}^{n-1} \det\left(L\left(b_1, b_2, \ldots, b_i\right)\right)$$

Feltehetjük, hogy a kiinduló bázisvektoraink koordinátái egészek: ha nem azok, akkor felszorzunk a nevezők legkisebb közös többszörösével. Vegyük észre, hogy ekkor az algoritmus végig egész-koordinátás bázisokon mozog. Ha a jobb oldalon szereplő rács bázisához tartozó mátrixot $B_i$-vel jelöljük, akkor a determináns definíciójában szereplő Gram mátrix $\mathbb{Z}^{i\times i}$-beli lévén pozitív egész determinánsú, következésképp

$$\det\left(L\left(b_1, b_2, \ldots, b_i\right)\right) = \sqrt{\det\left(B_i^T B_i\right)} \geq 1$$

minden i-re, azaz a potenciál függvény is mindig $\geq 1$ lesz. Mivel az első lépésünk fixen hagyja az ortogonalizált vektorokat, a potenciál csak a második lépés során változhat, mégpedig a következő módon csökken:

**1.3.7. Lemma.** *Ha alkalmaznunk kell a második lépést, azaz akad két vektor a bázisban, melyek sértik a Lovász-redukáltság feltételét, akkor ezek megcserélésének hatására az új* $D' = D\left(b_1, \ldots, b_{i-1}, b_{i+1}, b_i, b_{i+2}, \ldots, b_n\right)$ *potenciálra* $\frac{D'}{D} \leq \frac{\sqrt{3}}{2}$

**Bizonyítás.**(lemma)Legyen $b_i$és $b_{i+1}$melyekre sérül a feltétel:

$$\left\|b_i\left(i\right)\right\|^2 > \frac{4}{3}\left\|b_{i+1}\left(i\right)\right\|^2$$

A Gram-Schmidt ortogonalizált vektorok közül csak az i-edik és i+1-edik változik. $b_i\left(i\right)$ és $b_{i+1}\left(i\right)$vektorok által bezárt szöget jelöljük „$\alpha$”-val. Ekkor

a potencilok hányadosa

$$\frac{D'}{D} = \frac{\|b_{i+1}(i)\|^{n-i} \cdot \|b_i(i) \cdot \sin\alpha\|^{n-i-1}}{\|b_i(i)\|^{n-i} \cdot \|b_{i+1}(i) \cdot \sin\alpha\|^{n-i-1}} = \frac{\|b_{i+1}(i)\|}{\|b_i(i)\|} < \sqrt{\frac{3}{4}}$$

Adjunk becslést az algoritmus lépésszámára a lemma segítségével! „h" lépés után:

$$1 \leq D(b'_1, b'_2, \ldots, b'_n) \leq \left(\frac{\sqrt{3}}{2}\right)^h D(b_1, b_2, \ldots, b_n) \leq \left(\frac{\sqrt{3}}{2}\right)^h \left(\max_{1\leq i\leq n}\{\|b_i\|\}\right)^{\binom{n}{2}}$$

Ennek logaritmusát véve:

$$h \leq \frac{1}{\log {}^2/_{\sqrt{3}}} \binom{n}{2} \max_{1\leq i\leq n}\{\log\|b_i\|\}$$

# 2. Bonyolultságelméleti vonatkozások

## 2.1. CVP: Egy másik alapvető rácselméleti probléma

Ebben az alfejezetben azt vizsgáljuk, hogy az LLL-algoritmus hogyan tud segíteni egy adott vektorhoz egy adott rácsban viszonylag közeli vektorok keresésében. A ténylegesen legközelebbi vektor megtalálásának problémáját nevezzük CVP.(Closest Vector Problem)-nek.

Nevezzük el a B bázis vektorai által kifeszített origóba tolt centrumú parallelepipedont:

$$\mathcal{P}(B) := \left\{ \sum_{i=1}^{n} \lambda_i b_i \,\middle|\, \forall i\ \lambda_i \in \left[-\frac{1}{2}, \frac{1}{2}\right) \right\}$$

Ekkor könnyű meggondolni, hogy mind $\mathcal{P}(B) + L(B)$, mind $\mathcal{P}(B^*) + L(B)$ egyrétűen fedik le $\mathbb{R}^n$-t. Ebből kifolyólag tetszőleges $t \in \mathbb{R}^n$ vektorra $t + \mathcal{P}(B^*)$ pontosan egy $L(B)$-beli rácsvektort fog tartalmazni.

Azt szeretnénk megmutatni, hogy ez a rács vektor gyors algoritmussal megtalálható, valamint, hogy amennyiben Lovász-redukált bázisból indulunk

ki, viszonylag jól megközelíti a „$t$” célvektort. Rögzítsük tehát a $t \in \mathbb{R}^n$ vektort, és tegyük fel, hogy „$B$” Lovász-redukált bázis! A rácsvektort amit a $t+\mathcal{P}(B^*)$ téglatestben találtunk jelöljük $Bx$-el, a „$t$”-hez legközelebbit pedig $B\widehat{x}$-al ($x, \widehat{x} \in \mathbb{Z}^n$). Belátjuk, hogy ekkor

$$\|Bx - t\|_2 < 2^{\frac{n}{2}} \|B\widehat{x} - t\|_2 .$$

Az eltérés-vektorokat felírhatjuk a $B^*$ bázisban: $\exists z, \widehat{z} \in \mathbb{R}^n \qquad Bx - t = B^*z \quad \& \quad B\widehat{x} - t = B^*\widehat{z}$. „$x$” definíciója miatt „$z$” vektor minden koordinátájára teljesülni fog, hogy $\lceil z_i \rfloor = 0$. Tegyük fel, hogy $\widehat{x} \neq x$, és legyen „$s$” a legnagyobb index, ahol eltérés van: $\widehat{x}_s \neq x_s$.

**2.1.1. Lemma.** *Minden $i > s$ indexre $\widehat{z}_i = z_i$, és $\widehat{z}_i - z_i \in \mathbb{Z}\{0\}$*

**Bizonyítás.**(lemma)Vegyük a „$G$” Gram-Schmidt együttható-mátrixot! $B = B^* \cdot G$ miatt felírhatjuk, hogy

$$B^* \cdot Gx - t = B^*z;$$

$$B^* \cdot G\widehat{x} - t = B^*\widehat{z}.$$

Ezek különbségét véve kapjuk, hogy

$$B^* \cdot G(\widehat{x} - x) = B^*(\widehat{z} - z).$$

Mivel $B^*$ invertálható, ennek következményeképp

$$B^* \cdot G(\widehat{x} - x) = B^*(\widehat{z} - z).$$

"$G$" tulajdonságaiból (felső háromszögmátrix, egyesekkel a diagonálisában) következik a lemma állítása.

Szűkségünk lesz még a $|\widehat{z}_s| \geq \frac{1}{2}$ egyenlőtlenségre. amennyiben ez nem lenne igaz, állna, hogy $|\widehat{z}_s - z_s| \leq |\widehat{z}_s| + |z_s| < \frac{1}{2} + \frac{1}{2} = 1$, ami ellentmondásban állna azzal, hogy $\widehat{z}_i - z_i$ nem-0 egész szám.

Ezek alapján

$$\|Bx - t\|^2 = \|B^*z\|^2 = \sum_{i=1}^{n} \|b_i^*\|^2 z_i^2 =$$

$$= \sum_{i=1}^{s} z_i^2 \left\|b_i^*\right\|^2 + \sum_{i=s+1}^{n} z_i^2 \left\|b_i^*\right\|^2 \leq$$

$$\leq \sum_{i=1}^{s} \frac{2^{s-i}}{4} \left\|b_s^*\right\|^2 + \sum_{i=s+1}^{n} z_i^2 \left\|b_i^*\right\|^2 =$$

$$= \frac{2^s - 1}{4} \left\|b_s^*\right\|^2 + \sum_{i=s+1}^{n} \widehat{z}_i^2 \left\|b_i^*\right\|^2 <$$

$$< 2^s \left( \widehat{z}_s^2 \left\|b_s^*\right\|^2 + \sum_{i=s+1}^{n} \widehat{z}_i^2 \left\|b_i^*\right\|^2 \right) \leq$$

$$\leq 2^s \left( \sum_{i=s+1}^{n} \widehat{z}_i^2 \left\|b_i^*\right\|^2 \right) =$$

$$= 2^s \left\|B^* \widehat{z}\right\|^2 \leq 2^n \left\|B^* \widehat{z}\right\|^2 = 2^n \left\|B\widehat{x} - t\right\|^2$$

Jelöljük $\lambda_i$-vel a $\frac{\left\langle t - Bx, b_i^* \right\rangle}{\left\langle b_i^*, b_i^* \right\rangle}$mennyiséget! A $t + \mathcal{P}(B^*)$ téglán belüli rácspont kereséshez vegyük a következő algoritmust:

**2.1.2. Algoritmus.** *Vegyük bemenetként az $x \in \mathbb{Z}^n$ vektort! Ha minden indexre $\lceil \lambda_i \rfloor = 0$, akkor leállunk. Különben pedig vegyük a legnagyobb „i" indexet, melyre $\lceil \lambda_i \rfloor \neq 0$ erre $x_i := x_i + \lceil \lambda_i \rfloor$.*

**2.1.3. Állítás.** *Ez az algoritmus polinomiális időben visszaad egy olyan „x" vektort, melyre $Bx \in t + \mathcal{P}(B^*)$ tetszőleges input vektor esetén*

**Bizonyítás.**A bizonyítás hasonlóan működik mint a 1.2.3 algoritmus működésének bizonyítása.

## 2.2. Bonyolultság

Lagarias megmutatta, hogy az SVP NP-nehéz $\|.\|_\infty$ normában. Van Embde Boas belátta, hogy a CVP probléma minden p-normában nézve NP-nehéz. Hosszú ideje kérdés, hogy az SVP euklideszi normában NP-nehéz-e. 1997-ben Ajtai bebizonyította az NP-nehézséget véletlen redukció alatt, az általánosan vett feladat bonyolultságelméleti osztálya viszont még mindig

nyitott kérdés. A CVP alapvetően nem eldöntési kérdés, hanem optimalizálási feladat, azonban könnyen átalakíthatjuk a következő kérdéssé: van-e a rácsban a célvektorhoz legfeljebb „$r$” távolságra található vektor. Természetesen, amikor a következő alfejezetben a CVP NP-teljességéről beszélünk, ezt a problémát fogjuk visszavezetni más NP-teljes feladatra.

## 2.3. CVP NP-teljes

Ismert, az NP-teljes részhalmaz-összeg(subset-sum) probléma: adott $\mathbf{a} \in \mathbb{Z}^n, b, M \in \mathbb{Z}$ paraméterekre keresendő az $\langle \mathbf{a}, \mathbf{x} \rangle \equiv b \pmod{M}$ kongruenciát kielégítő $\{0,1\}^n$-beli „$\mathbf{x}$” vektor. Ennek segítségével fogjuk belátni a CVP probléma NP-teljességét.

Vegyük a

$$B := \begin{bmatrix} M & a_1 & \cdots & a_n \\ 0 & & & \\ \vdots & & I_n & \\ 0 & & & \end{bmatrix}$$

bázis által generált rácsot. Keressük ebben a

$$t := \begin{pmatrix} b \\ 1/2 \\ \vdots \\ 1/2 \end{pmatrix}$$

vektorhoz legközelebbi rácsvektort. Mivel az $L(B)$ elemei mind egész koordinátásak a t-től való távolság legalább $\sqrt{\frac{n}{4}}$.

Tegyük fel, hogy a részhalmaz-összeg problémánk kielégíthető, találtunk olyan $\mathbf{x} \in \mathbb{Z}^n$-et és $y \in \mathbb{Z}$-t, melyekre $\langle \mathbf{a}, \mathbf{x} \rangle = b - yM$. Ekkor

$$\left\| B \begin{pmatrix} y \\ \mathbf{x} \end{pmatrix} - t \right\|^2 = \left\| \begin{pmatrix} \langle \mathbf{a}, \mathbf{x} \rangle + yM - b \\ \mathbf{x} - 1/2 \end{pmatrix} \right\|^2 = \left\| \begin{pmatrix} 0 \\ \mathbf{x} - 1/2 \end{pmatrix} \right\|^2 = \frac{n}{4},$$

azaz $B \begin{pmatrix} y \\ \mathbf{x} \end{pmatrix}$ „$t$”-től minimális távolságra van.

A másik irányhoz tegyük fel, hogy a rácsunk egy adott $B\begin{pmatrix} y \\ \mathbf{x} \end{pmatrix}$ vektora $\frac{n}{4}$-nél közelebb van a célvektorhoz. Ekkor felírható, hogy

$$\frac{n}{4} \geq \left\| B\begin{pmatrix} y \\ \mathbf{x} \end{pmatrix} - t \right\|^2 = \left(\langle \mathbf{a}, \mathbf{x} \rangle + yM - b\right)^2 + \left\| \mathbf{x} - {}^1\!/_{\mathbf{2}} \right\|^2 .$$

Itt az $\mathbf{x} - {}^1\!/_{\mathbf{2}}$ vektor hossza akkor minimális, ha „$\mathbf{x}$" valóban egy 0-1 karakterisztikus vektor. Ekkor a hossz-négyzete $\frac{n}{4}$. Ilyenkor az egyenlőtlenség továbbalakítható:

$$0 \geq \left(\langle \mathbf{a}, \mathbf{x} \rangle + yM - b\right)^2 \Rightarrow \langle \mathbf{a}, \mathbf{x} \rangle + yM - b = 0,$$

ami azt jelenti, hogy „$\mathbf{x}$" megoldása az adott részhalmaz-összeg kérdésnek.

# 3. Kísérletek prímfaktorizálásra

## 3.1. Alapok, Schnorr algoritmusa

Ebben a fejezetben az lesz a fő célunk, hogy polinomiális időben megtaláljuk egy adott $N \in \mathbb{N}$ szám faktorait. Módszerünk alapját az képzi, hogy ha az

$$x^2 \equiv y^2 \pmod{N}$$

kongruenciára találunk $(x, y) \in \mathbb{Z} \times \mathbb{Z}$ nem-triviális megoldást, azaz olyat ahol $x \not\equiv \pm y \pmod{N}$, akkor

$$N \mid (x - y)(x + y) \qquad \& \qquad N \nmid (x \pm y)$$

miatt $(x + y, N)$ „$N$"-nek valódi faktora lesz.

Míg ha „$N$" prím szám, minden megoldás triviális, a következő állítás azt mutatja, hogy összetett szám esetén reménykedhetünk abban, hogy egy véletlenszerűen vett megoldást felhasználhatunk

**3.1.1. Állítás.** *Tegyük fel, hogy „$N$" páratlan összetett szám, és az $(x, y) \in \mathbb{Z}^2$ $N$-el külön-külön relatív prím számpár kielégíti a kongruenciát. Ekkor legalább 1/2 valószínűsséggel teljesül, hogy $x \not\equiv \pm y \pmod{N}$*

**Bizonyítás.**Bontsuk fel „$N$"-et: $N = P \cdot Q$, ahol $(P, Q) = 1$. Rögzítsük a megoldásból „$x$"-et. Ekkor a két triviális megoldáshoz ($\pm x$), mutatunk két nem-triviálisat. A kínai maradéktétel szerint a

$$y \equiv x \pmod{P}$$

$$y \equiv -x \pmod{Q}$$

rendszernek „$y$"-ra pontosan egy megoldása van ($mod\ N$). Ekkor $(x, \pm y)$ két nem-triviális megoldása lesz az elsőnek (minden feltétel könnyen ellenőrizhető).

Vezessünk be néhány jelölést!

**3.1.2. Definíció.** *Nevezzünk K-simának (K-smooth) az N természetes számot, amennyiben N mentes a K-nál nagyobb prímfaktoroktól.*

Jelölje $p_i$ az „$i$"-edik prímszámot!

**3.1.3. Algoritmus.** *(Schnorr)*

*(1) Bemenetként vegyük az N számot, amit faktorizálni szeretnénk.*

*(2) Állítsuk be a „d" dimenziót, és a $C > 1$ konstanst. Ha N faktorai között szerepel a $p_1, p_2 \ldots, p_d$ prímek valamelyike, akkor adjuk vissza, és álljunk le. Ellenkező esetben vegyük a „d" és „C" értékekhez tartozó (a későbbiekben részletezendő) $S_p$ rácsot, majd egészítsük ki az első d darab prímet tartalmazó listánkat a $p_0 := -1$-el.*

*(3) Keressünk $d + 2$ darab $(u_i, k_i) \in \mathbb{N} \times \mathbb{Z}$ számpárt, ahol $u_i$ $p_d$-sima, és teljesül az*

$$|u_i - k_i N| \leq p_d$$

*egyenlőtlenség. Faktorizáljuk az $u_i$, valamint az $u_i - k_i N$ számokat:*

$$u_i = \prod_{j=0}^{d} p_j^{a_{i,j}}, \qquad a_{i,0} = 0$$

$$u_i - k_i N = \prod_{j=0}^{d} p_j^{bi,j}$$

*(4) Vegyük az $a_{j,i}$, valamint a $b_{j,i}$, kitevők által generált „A", illetve „B" $\mathbb{N}^{(d+1)\times(d+2)}$-beli mátrixokat („j" a sor-, "i" az oszlopindex)*

*(5) Tekintsük az*

$$(A+B)\,\mathbf{c} \equiv \mathbf{0} \pmod 2$$

*egyenlet egy $\mathbf{c} \in \{0,1\}^{d+2}$ megoldását.*

*(6) Vegyük az*

$$x := \prod_{i=0}^{d} p_j^{((A+B)\mathbf{c})_j/2}$$

*és*

$$y := \prod_{i=0}^{d} p_j^{(A\mathbf{c})_j}$$

*számokat. Amennyiben $x \neq \pm y \pmod N$, adjuk vissza értékként $(x+y, N)$-et. Különben pedig próbálkozzunk meg az 5. lépésben lévő egyenlet egy másik megoldásával.*

**3.1.4. Megjegyzés.** *Első ránézésre nem biztos, hogy világos a „$\mathbf{c}$" vektor szerepe. Ő egy karakterisztikus vektor: kiválasztja, hogy mely $u_i$ számok szorzata legyen az „y" szám:*

$$y = \prod_{i=0}^{d} p_j^{(A\mathbf{c})_j} = \prod_{i=1}^{d+2} u_i^{c_i} \equiv \prod_{i=1}^{d+2} (u_i - k_i N)^{c_i} \equiv \prod_{i=0}^{d} p_j^{(B\mathbf{c})_j} \pmod N .$$

*Ez alapján máris megvan, hogy $(x,y)$ megfelel az elsődleges célunknak, azaz*

$$x^2 = \prod_{i=0}^{d} p_j^{((A+B)\mathbf{c})_j} = \prod_{i=0}^{d} p_j^{(A\mathbf{c})_j} \cdot \prod_{i=0}^{d} p_j^{(B\mathbf{c})_j} \equiv y^2 \pmod N$$

*Az algoritmus működéséhez persze kell, hogy a 3. lépésben megfelelő számpárokat találjunk. Valójában ez a lépés viszi el majdnem a teljes futásidejét az algoritmusnak. A következőkben ilyen jellegű számpároknak az észlelhetőségéhez fogunk feltételt adni.*

Az

$$|u_i - k_i N| \leq p_d$$

feltételt általánosítva vehetjük a következő problémát:

**3.1.5. Probléma.** *Legyen $d \geq 1$ rögzített állandó, és „N" legfeljebb $p_d$ faktoroktól mentes természetes szám. Keresendő $d+2$ darab $(u, v, k, \gamma)$ számnégyes, ahol „u" és "v" $p_d$-sima egészek, „k" és "N" relatív prímek, valamint $\gamma \in \mathbb{N}^+$, amelyek teljesítik a következő feltételt:*

$$u = v + kN^{\gamma} \tag{1}$$

## 3.2. Rácsok

Adleman prím-rácsa az alábbi $A_p$ mátrix (oszlopai) által generált rács.

$$A_p := \begin{bmatrix} \sqrt[p]{\ln p_1} & 0 & 0 & 0 & 0 \\ 0 & \sqrt[p]{\ln p_2} & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & \sqrt[p]{\ln p_d} & 0 \\ C \ln p_1 & C \ln p_2 & \cdots & C \ln p_d & C \ln N \end{bmatrix}$$

Ekkor a rács elemeit felírhatjuk

$$A_p \mathbf{z} = \begin{bmatrix} z_1 \sqrt[p]{\ln p_1} \\ z_2 \sqrt[p]{\ln p_2} \\ \vdots \\ z_n \sqrt[p]{\ln p_n} \\ C \left( \sum_{i=1}^{d} z_i \ln p_i + z_{d+1} \ln N \right) \end{bmatrix}$$

alakban. A rácsvektorok hossza:

$$\|A_p \mathbf{z}\|_p^p = \sum_{i=1}^{d} |z_i^p| \ln p_i + C^p \left| \sum_{i=1}^{d} z_i \ln p_i + z_{d+1} \ln N \right|^p$$

**3.2.1. Tétel.** *Legyen $C > 1$ konstans és $\mathbf{z} \in \mathbb{Z}^{d+1}$ nem-0 utolsó koordinátával, ahol az utolsó koordinátának abszolútértékét jelöljük $\gamma := |z_{d+1}|$-val. Ekkor amennyiben találunk a rácsban megfelelő rövid*

$$\|A_1 \mathbf{z}\|_1 \leq 2 \ln C + 2\sigma \ln p_d - \gamma \ln N, \tag{2}$$

*tudunk mutatni ehhez a „$\mathbf{z}$" vektorhoz tartozó megfelelő $u, k$ számokat, hogy*

$$|u - kN^{\gamma}| \leq p_d^{\sigma}$$

**Bizonyítás.**Először is feltehetjük, hogy az utolsó $z_{d+1}$ koordináta negatív, ellenkező esetben vehetjük a vektorunk additív inverzét. Legyenek az $u, k, \gamma$ számok a következőek:

$$u := \prod_{i,\, z_i>0} p_i^{z_i}, \qquad k := \prod_{i,\, z_i<0} p_i^{|z_i|}$$

$$\gamma := |z_{d+1}|$$

Célunk belátni, hogy ezek eleget tesznek az állításnak. Az áttekinthetőség érdekében nevezzük el a (2) egyenlőtlenség jobb oldalát:

$$\varepsilon := 2\ln C + 2\sigma \ln p_d - \gamma \ln N$$

Ezek alapján

$$\|A_1\mathbf{z}\|_1 = \ln u + \ln k + C\,|\ln u - \ln(kN^\gamma)|^p < \varepsilon$$

Vegyük észre, hogy ez az egyenlőtlenség szimmetrikus $u$-ban és $kN^\gamma$-ban, ezért feltehetjük, hogy $u \geq kN^\gamma$, így elhagyhatjuk az abszolút értéket, és átrendezve az egyenlőtlenséget ezt kapjuk:

$$kN^\gamma \leq u \leq k^{\frac{C-1}{C+1}} \cdot N^{\frac{C\tau}{C+1}} \cdot \exp\left(\frac{\varepsilon}{C+1}\right)$$

Tehát $u - kN^\gamma$ értékét felülről becsülhetjük ezzel a különbséggel:

$$u - kN^\gamma \leq k^{\frac{C-1}{C+1}} \cdot N^{\frac{C\tau}{C+1}} \cdot \exp\left(\frac{\varepsilon}{C+1}\right) - kN^\gamma \tag{3}$$

Vegyük ezt az értéket, mint $k$ függvényét és vizsgáljuk, hogy hol veheti fel a maximumhelyét, vagyis, hogy hol nulla a k szerinti deriváltja:

$$\left(\frac{C-1}{C+1}\right) \cdot k_0^{-\frac{2}{C+1}} \cdot N^{\frac{C\tau}{C+1}} \cdot \exp\left(\frac{\varepsilon}{C+1}\right) - N^\gamma = 0$$

$$k_0^{-\frac{2}{C+1}} = \left(\frac{C-1}{C+1}\right)^{-1} \cdot N^{\frac{\tau}{C+1}} \cdot \exp\left(-\frac{\varepsilon}{C+1}\right)$$

$$k_0 = \left(\frac{C-1}{C+1}\right)^{\frac{C+1}{2}} \cdot N^{-\frac{\tau}{2}} \cdot \exp\left(\frac{\varepsilon}{2}\right)$$

Behelyettesítve (3)-be

$$u - kN^{\gamma} \leq$$

$$\leq \left( \left( \frac{C-1}{C+1} \right)^{\frac{C-1}{2}} \cdot N^{-\frac{\tau(C-1)}{2(C+1)}} \cdot \exp\left( \frac{\varepsilon\,(C-1)}{2\,(C+1)} \right) \right) \cdot N^{\frac{C\tau}{C+1}} \cdot \exp\left( \frac{\varepsilon}{C+1} \right) -$$

$$- \left( \frac{C-1}{C+1} \right)^{\frac{C+1}{2}} \cdot N^{-\frac{\tau}{2}} \cdot \exp\left( \frac{\varepsilon}{2} \right) N^{\gamma} =$$

$$= \left( \frac{C-1}{C+1} \right)^{\frac{C-1}{2}} \cdot N^{\frac{\tau}{2}} \cdot \exp\left( \frac{\varepsilon}{2} \right) - \left( \frac{C-1}{C+1} \right)^{\frac{C+1}{2}} \cdot N^{\frac{\tau}{2}} \cdot \exp\left( \frac{\varepsilon}{2} \right) =$$

$$\left( \frac{C-1}{C+1} \right)^{\frac{C-1}{2}} \cdot N^{\frac{\tau}{2}} \cdot \exp\left( \frac{\varepsilon}{2} \right) \cdot \left( \frac{2}{C+1} \right)$$

**3.2.2. Lemma.**

$$C > 1 \Longrightarrow \left( \frac{C-1}{C+1} \right)^{\frac{C-1}{2}} \left( \frac{2}{C+1} \right) \leq \frac{1}{C}$$

**Bizonyítás.**

$$(C-1) \cdot \frac{C-1}{2} + (2C) \cdot 1 = (C+1) \cdot \frac{C+1}{2}$$

Ezért felírhatjuk a logaritmus függvényre az alábbi (súlyozott) Jensen-esgyenlőtlenséget:

$$\ln (C-1) \cdot \frac{C-1}{2} + \ln(2C) \cdot 1 \leq \ln (C+1) \cdot \frac{C+1}{2}$$

$$(C-1)^{\frac{C-1}{2}} \cdot 2C \leq (C+1)^{\frac{C+1}{2}}$$

Amiből átrendezve kapjuk a lemmánk állítását.

Ezt felhasználva az egyenlőtlenséget tovább alakíthatjuk:

$$u - kN^{\gamma} \leq \frac{1}{C} \cdot N^{\frac{\tau}{2}} \cdot \exp\left( \frac{\varepsilon}{2} \right)$$

Ebbe visszaírva $\varepsilon$-t, pont a bizonyítandó állítást kapjuk.

**3.2.3. Megjegyzés.** *Ha a tételben szereplő „$\sigma$" paraméter 1, az garantálja, hogy megfelelő $p_d$-sima $u$, $v := u - kN^\gamma$ számokat kapunk, melyek így megoldását adják (1) egyenletnek. Viszont amennyiben nem lényegesen nagyobb mint 1, még mindig reménykedhetünk benne, hogy teljesülni fog a $p_d$-simaság. Ez alapján a következő lehetőségünk adódik a faktorizálásra : 1-es normában rövid vektorokat keresünk az $A_1$ rácsban, majd ellenőrizzük, hogy az ezek segítségével legyártott számok valódi megoldását adják-e (1)-nek egészen addig, míg legalább d+2 ilyen össze nem gyűlik. Ha ezt sikerült elérni, akkor tudjuk alkalmazni a Schnorr-algoritmust ezek segítségével.*

**3.2.4. Megjegyzés.** *A tétel segíthet annak vizsgálatában, hogy van-e d+2 darab megoldás a (1) egyenlőségre a Minkovski-tétel felhasználásával, ami explicit becslést ad a legrövidebb vektor hosszára.*

**3.2.5. Megjegyzés.** *Mivel a rácsalgoritmusok többsége euklideszi normával dolgozik, hasznos lenne egy az előbbiekhez hasonló tétel, mely $\|.\|_2$-ban mérve kis vektorokból indul ki.*

Egy másik hasonló útként vegyük a következő $S_p$ mátrix (oszlopai) által generált rácsot(Sperner prím rácsa ).

$$S_p := \begin{bmatrix} \sqrt[p]{\ln p_1} & 0 & 0 & 0 \\ 0 & \sqrt[p]{\ln p_2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & \sqrt[p]{\ln p_d} \\ C \ln p_1 & C \ln p_2 & \cdots & C \ln p_d \end{bmatrix}$$

A rácsvektorok között keressünk a

$$\mathbf{t} := \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ C \ln N \end{bmatrix}$$

vektorhoz közel lévőket. A „$\mathbf{t}$”-től való távolságvektor:

$$S_p\mathbf{z}-\mathbf{t}=\begin{bmatrix} z_1\sqrt[p]{\ln p_1} \\ z_2\sqrt[p]{\ln p_2} \\ \vdots \\ z_n\sqrt[p]{\ln p_n} \\ C\left(\sum_{i=1}^{d} z_i \ln p_i - \ln N\right) \end{bmatrix}$$

$$\|S_p\mathbf{z}-\mathbf{t}\|_p^p=\sum_{i=1}^{d}|z_i^p|\ln p_i + C^p\left|\sum_{i=1}^{d} z_i\ln p_i - \ln N\right|^p$$

**3.2.6. Tétel.** *Legyen $C>1$ konstans és $\mathbf{z}\in\mathbb{Z}^d$, Ekkor amennyiben találunk a rácsban „$\mathbf{t}$”-hez elegendően közeli $S_1\mathbf{z}$ vektort:*

$$\|S_1\mathbf{z}-\mathbf{t}\|_1 \le 2\ln C + 2\sigma\ln p_d - \ln N,$$

*tudunk mutatni ehhez a „$\mathbf{z}$” vektorhoz tartozó $u,k$ számokat*

$$|u-kN|\le p_d^{\sigma}$$

**Bizonyítás.**A tétel és a bizonyítás egy az egyben az Adleman rácsára vonatkozó megfelelője a $\gamma=1$ esetben.

**3.2.7. Megjegyzés.** *$N$ faktorizálásához itt „$\mathbf{t}$”-hez közeli $S_1$-beli rácsvektorokat kell keresnünk. Adleman módszerének megvan az az előnye, hogy nagyobb területen van esélye vektorokat találni. A gyakorlat azonban azt mutatja, hogy az így megtalált valóban használható megoldások - amelyek kielégítik az (1)-hez tartozó feltételeket - pont azok, amelyeket Schnorr megközelítése ad.*

**3.2.8. Megjegyzés.** *Csakúgy mint az előbbiekben, ennek a tételnek euklideszi normával számoló megfelelője is segítene a gyakorlatban.*

# 4. Összefoglalás

# A. Jelölések

| | |
|---|---|
| log | 2-es alapú logaritmus |
| $(a, b)$ | az „$a$” és „$b$” természetes számok legnagyobb közös osztója |
| $\lfloor x \rceil$ | az „$x$” valós szám kerekített értéke |
| $\mathbb{Z}$ | az egész számok halmaza |
| $\mathbb{Q}$ | a racionális számok halmaza |
| $\mathbb{R}$ | a valós számok halmaza |
| $p_i$ | az „$i$”-edik prím szám |
| $\lambda(L)$ | Az „$L$” rácsban a legrövidebb nem-nulla vektor hossza |

# B. Felhasznált irodalom

# Hivatkozások


[1] J.-Y. Cai. Some Recent Progress on Complexity of Lattice Problems. In Proc of FCRC, 1999.

[2] Kajtár M. Cons. Grolmusz V. Lattice Theory in Cryptography, 2008

[3] Lovász L. An Algorithmic Theory of Numbers, Graphs and Convexity. a CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM, Philadelphia, 1986.

[4] D. Micciancio, A. Yannakopoulos and N. Segerlind. Lattices in Cryptography and Cryptanalysis. Lecture 10, CSE 291., 1999.

[5] C. P. Schnorr. Factoring integers and computing discrete logarithms via diophantine approximation. In Advances in Computational Complexity Theory, J.-Y. Cai, Ed., vol. 13 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science. AMS, 1993.

[6] A. I. Vera. A Note on Integer Factorrization Using Lattices. CACAO Project, INRIA Nancy Grand-Est, 2010