Eötvös Loránd Tudományegyetem

Természettudományi Kar

Baranyi Károly Tamás

Rácsalgoritmusok

MSc Szakdolgozat

Témavezető

Dr. Grolmusz Vince

Számítógéptudományi Tanszék

Budapest, 2017

Tartalomjegyzék

1. Bevezetés

A rácsok periodikus, diszkrét struktúrák egy vektortérben, generáló elemeik egész együtthatós lineáris kombinációiból állnak. Jelentőségük mind az elméleti matematikában, mind az alkalmazott matematikában hatalmas. A csoportelméletben rácsokon keresztül jutottak el bizonyos sporadikus csoportok felfedezéséig, a kódelmélethez a lineáris kódokon keresztül kapcsolódnak, számtalan fontos számelméleti probléma megfogalmazható segítségükkel, a kriptográfiában sok séma rácsokon megfogalmazott feladatok feltételezett nehézségén alapszik, továbbá hasznosnak bizonyultak a Lie-algebrák kutatásában is. Szerepük a szilárdtest-fizikában és a fizika más területein is jelentős, a kémiában pedig a kristályrács elrendeződés miatt fontosak.

A rácsok és a velük kapcsolatos problémák, elsősorban a rács legrövidebb vektorának megtalálása (Shortest Vector Problem, SVP) matematikai kutatása több, mint egy évszázada zajlik [39], olyan nevekkel, mint Korkin és Zolotarev, Hermite, Minkowski és Voronoi, azonban a felmerülő problémák algoritmikus megoldására vonatkozó kutatások relatíve későn, csak az 1980-as években kezdődtek el. Akkoriban a rácsproblémák jelentették a szűk keresztmetszetet a kombinatorikus optimalizálási és algebrai számelméleti feladatokban. Ebbe a környezetbe robbant bele Arjen Klaas Lenstra, Hendrik Willem Lenstra Jr. és Lovász László legendás cikke, melyben egy algoritmust javasolnak a rácsokban legrövidebb vektor keresésére. A később LLL néven híressé vált algoritmus polinomiális futásidejű és bár a legrövidebb vektort csak a dimenzióban exponenciális hibával közelíti, ez lenyűgözően sok alkalmazáshoz elégnek bizonyult. Azóta a cikkre több, mint ezerszer hivatkoztak más kutatásokban [27], az LLL algoritmus valamilyen változatát az összes lényeges matematikai könyvtár és keretrendszer tartalmazza. 2007-ben konferenciát rendeztek az algoritmus születésének 25. évfordulója alkalmából, melyből kiadvány is született [44].

Szakdolgozatomban rácsokkal és azokkal kapcsolatos algoritmusokkal foglalkozom, kitüntetett szerepet szánva az Lenstra–Lenstra–Lovász algoritmusnak.

Az első fejezet a rácsok precíz bevezetéséről szól, alapvető tulajdonságaik és a velük kapcsolatban felmerülő leggyakoribb feladatok érintésével, külön kiemelve a Gram–Schmidt-eljárást, amely, pontosabban a generáló elemek szöge abszolút kulcsa a rácsokkal kapcsolatos problémák hatékony megoldásának.

A második fejezetet teljes egészében az LLL algoritmusnak szenteltem, a hozzá szorosan kapcsolódó részekkel, mint a bázisokat gyengén redukált formába transzformáló algoritmus, ami az LLL algoritmus szubrutinjaként nyilván csak azzal együtt tárgyalható. Itt szerepel az algoritmus pontos leírása, a korrektség alátámasztása, illetve a polinomiális futásidő bizonyítása.

A rácsokkal kapcsolatos feladatok közül SVP-t már említettem, a Closest Vector

Problem (CVP) ezzel szoros viszonyban áll, itt a feladat egy adott, jellemzően rácson kívüli vektorhoz legközelebb található rácsvektor megkeresése. Ezen feladatok számítástudományi értelemben vett komplexitása olyan, fontos része a tárgyalásnak, amelyben sok eredmény született az utóbbi idők matematikai kutatásában. Ebből Peter van Emde Boas mély eredményét emeltem ki és dolgoztam fel részletesen, amelyben bizonyítja a fent említett SVP feladat és CVP feladat NP-nehéz voltát.
A negyedik fejezet az alkalmazásokról szól, egy racionális vektor elemeinek szimultán diofantikus approximációja, ennek közelítő megoldása után bizonyos típusú polinomkongruenciákra térek át, majd kriptográfiai alkalmazásokat veszek sorra, előbb kitérek a jól ismert RSA titkosítás rácsok segítségével történő feltörésének lehetőségeire, majd ellenkező irányból bemutatok egy rácsokon alapuló, hatékony titkosítási sémát.

Köszönetnyilvánítás

Köszönöm témavezetőmnek, Grolmusz Vince tanár úrnak kitartó és alapos munkáját, útmutatását mind a szakdolgozat elkészítésében, mind az irodalom kiválasztásában, a rendszeres átolvasásokat és értékes tanácsait.
Köszönöm továbbá Király Tamás tanár úrnak mélyenszántó megjegyzéseit és hogy új aspektusból mutatta meg a dolgozat fő témáját adó LLL algoritmust.

Az ábrákat – kivéve, ahol máshogy hivatkoztam – Mathworks® MATLAB® segítségével készítettem, felhasználva Vladimir Bondarenko *DrawLA – Draw Toolbox for Linear Algebra* csomagjának egy általam módosított változatát.

1.1. Rácsok, tulajdonságaik

1.1. Definíció. *Legyen adott az n (véges) dimenziós V vektortér $\mathbb{R}$ felett, a vektortérben pedig egy m elemű $b_1, \ldots, b_m$ lineárisan független vektorrendszer. Az egész együtthatós lineáris kombinációik halmazát* ***rácsnak*** *nevezzük és a következőképpen jelöljük:*

$$\mathcal{L} = \mathcal{L}(b_1, \ldots, b_m) := \mathbb{Z}b_1 + \cdots + \mathbb{Z}b_m = \{\lambda_1 b_1 + \cdots + \lambda_m b_m : \lambda_i \in \mathbb{Z}\}.$$

$\mathcal{L}$ a $b_1, \ldots, b_m$ vektorok **által generált rács**, $b_1, \ldots, b_m$ pedig a **rács egy bázisa**. A bázis nyilván egyértelműen meghatározza a rácsot, de egy rácsnak több bázisa is lehet. Rácsot alkotnak például a szokásos $\mathbb{R}^2$ sík egész koordinátájú pontjai, nevezzük ezt a rácsot $\mathbb{Z}^2$-nek. Ennek a rácsnak egy kézenfekvő bázisa a

$$b_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad b_2 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

rendszer, de ugyanezt a rácsot generálja a

$$c_1 = \begin{pmatrix} 3 \\ 7 \end{pmatrix}, \quad c_2 = \begin{pmatrix} 2 \\ 5 \end{pmatrix}$$

rendszer is (mivel $5c_1 - 7c_2 = b_1$ és $-2c_1 + 3c_2 = b_2$). Ellenben nem bázisa például a

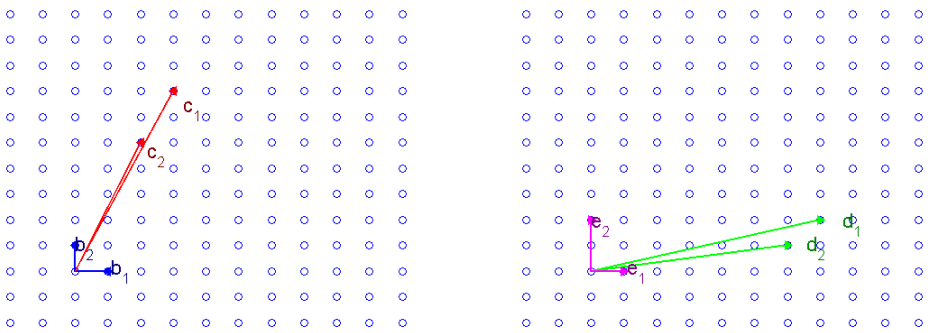
$$d_1 = \begin{pmatrix} 7 \\ 2 \end{pmatrix}, \quad d_2 = \begin{pmatrix} 6 \\ 1 \end{pmatrix}$$

vagy még szembetűnőbben az

$$e_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad e_2 = \begin{pmatrix} 0 \\ 2 \end{pmatrix}$$

rendszer (1. ábra). Láthatjuk tehát, hogy egy rácshoz több bázis is tartozhat, de az nem igaz, hogy bárhogyan választunk lineárisan független rácsbeli vektorokat, azok generálnák a rácsot.

Egy $\mathbb{R}^n$-beli, adott $\mathcal{L}$ ponthalmaz (vektorhalmaz) pontosan akkor rács, ha létezik bázisa, azaz ha léteznek olyan $\mathbb{R}^n$-beli $b_1, \ldots, b_m$ vektorok, melyekre $\mathcal{L} = \mathcal{L}(b_1, \ldots, b_m)$. A következő fejezetben szereplő, 1.5. tétel szerint a rácsokat úgy is definiálhattuk volna (ahogy azt az irodalom egy részében találjuk is), hogy egy rács a befoglaló vektortér egy *diszkrét részcsoportja*. (Diszkrét alatt jelen esetben azt értve, hogy a vektortér bármely korlátos halmazába a rácsnak véges sok eleme esik.) Másfelől, de szintén algebrai oldalról megközelítve egy rács tekinthető egy $b_1, \ldots, b_m$ elemek által generált $\mathbb{Z}$ *feletti unitér, bal oldali modulusnak* is a szokásos vektorösszeadásra, az egészek gyűrűjének műveleteire és a $\mathbb{Z}$-beli skalárral való szorzásra.



1. ábra. $\mathbb{Z}^2$ rács két bázisa (bal) és két nem-bázisa (jobb)

A szakdolgozatban a rácsokat algoritmusok vonatkozásában $\mathbb{R}$ helyett $\mathbb{Q}$ test fölötti vektortérben tekintjük, a szakirodalom túlnyomó részének megfelelően. A definíciók és általános tételek ugyanúgy érvényesek akár valós fölötti vektortérben, akár a racionális számok teste fölött értjük őket, ezeket a valós testre vonatkozóan fogalmazzuk meg.

Ha a vektorrendszer elemei nem feszítik a teljes V vektorteret, vehetjük az általuk feszített alteret és a vizsgálódásunkat korlátozhatjuk erre (az általuk generált rács természetesen az altérben is rács), így a továbbiakban feltesszük, hogy B teljes dimenziós, $n = m$. Jelölje a rács báziselemeiből álló mátrixot B, amelyről tehát feltehető, hogy négyzetes:

$$\mathbb{R}^{n\times n} \ni B = \Big[b_1, b_2, \ldots, b_n\Big],$$

a **bázis determinánsa** pedig legyen ennek a mátrixnak a determinánsa. A továbbiakban olykor magára a bázisra is B-vel fogunk hivatkozni.

Láttuk tehát, hogy egy rácsot több bázis is generálhat, de a következő állítások szerint az ugyanazon rácsot generáló bázisok determinánsa már elég meghatározott.

1.2. Állítás. *A $B = [b_1, \ldots, b_n]$ és $C = [c_1, \ldots, c_n]$ bázisok pontosan akkor generálják ugyanazt az $\mathcal{L}$ rácsot $\mathbb{R}^n$-ben, ha létezik $U \in \mathbb{Z}^{n\times n}$ unimoduláris (tehát ± 1 determinánsú) mátrix, amelyre $C = BU$.*

Bizonyítás: A Cramer-szabály felhasználásával látszik, hogy az unimodularitás ekvivalens azzal, hogy az egészek feletti mátrix az egészek felett invertálható is, azaz $U^{-1} \in \mathbb{Z}^{n\times n}$.

Ha B és C ugyanazon $\mathcal{L}$ rácsot generálják, akkor definíció szerint $\mathcal{L}$ elemei kifejezhetők mind B, mind C elemeinek egész együtthatós lineáris kombinációjaként, speciálisan (és ezzel ekvivalensen) C elemei is kifejezhetők B bázis segítségével és vice

versa, azaz létezik $U \in \mathbb{Z}^{n\times n}$, melyre $C = B\cdot U$ és az ellenkező irányú transzformáció mátrixa, U^{-1} is egészértékű. De ez pontosan akkor áll fenn, ha U unimoduláris. ∎

1.3. Következmény. *Ha $B = [b_1, \ldots, b_n]$ és $C = [c_1, \ldots, c_n]$ bázisok ugyanazt az $\mathcal{L}$ rácsot generálják, akkor* $\det B$ *és* $\det C$ *előjeltől eltekintve azonos.*

Bizonyítás: Az előző állítás alapján, ha B és C ugyanazt a rácsot generálják, akkor létezik U unimoduláris mátrix, melyre $C = BU$. Az unimodularitás szerint $|\det(U)| = 1$, tehát a determinánsok szorzástétele $|\det(C)| = |1 \cdot \det(B)|$ adja az állítást. ∎

Ez alapján tehát jogos a következő definíció.

1.4. Definíció. *Legyen az $\mathcal{L}$ rács egy tetszőleges bázisa B. **$\mathcal{L}$ rács determinánsát** a B determinánsának abszolútértékével definiáljuk:*

$$\det \mathcal{L} := |\det B|.$$

Így tehát rácsok determinánsát csak teljes rangú esetben definiáltuk. Megjegyezzük, hogy $\det \mathcal{L} := \sqrt{\det B^T B}$ képlettel a definíció kiterjeszthető lenne nem teljes rangú rácsokra is, de mi a korábbi megjegyzésre való tekintettel maradunk a négyzetes mátrixú rácsoknál.
Néhány megjegyzés a jelölésekről. Az előbbiekben is és az egész dolgozatban szögletes zárójelben olyan mátrixelemeket sorolunk fel, melyek maguk is vektorok, a gömbölyű zárójelet pedig arra az esetre tartjuk fenn, amikor a mátrix összes elemét (skalárokként) kiírjuk. Olykor, ha hangsúlyozni akarjuk a vektor és skalárelemek közötti különbségtételt, a vektorokat vastagított $\mathbf{v}$ alakban írjuk. Minden logaritmus 2-es alapú.

1.2. Rácsok ekvivalens definíciója, duális rács, példák

Az alábbi tételben belátjuk, hogy az előző fejezetben a rácsok alternatív definíciója tényleg ekvivalens az eredetivel. [22, 8.1. tétel]

1.5. Tétel. *Ha $\mathcal{L}$ egy B által generált rács, akkor elemei a befoglaló V vektortér additív csoportjának egy diszkrét részcsoportját alkotját és megfordítva, V^+ bármely diszkrét részcsoportja rácsot alkot.*

Bizonyítás: Tegyük fel, hogy $B = [\mathbf{b_1}, \ldots, \mathbf{b_m}]$ lineárisan független vektorrendszer V-ben és $\mathcal{L} = \mathcal{L}(B)$ az általa generált rács. Ekkor, mivel $\mathcal{L}$ elemei lineáris kombinációk, nyilvánvalóan részcsoportot alkotnak a vektorösszeadás műveletére. Másrészt

egészítsük ki B-t a teljes vektortér egy $\mathbf{b_1}, \ldots, \mathbf{b_m}, \mathbf{b_{m+1}}, \ldots, \mathbf{b_n}$ bázisává és jelöljük ezt is B-vel. Így B nemszinguláris, tehát létezik B^{-1} inverze. Legyen $R \subset V$ korlátos halmaz. Ha $\mathbf{x} \in \mathcal{L} \cap R$, akkor tehát $\mathbf{x}$ valamely $\lambda = \lambda_1, \ldots, \lambda_n$ segítségével előáll $\mathbf{x} = B\lambda$ alakban, azaz $\lambda = B^{-1}\mathbf{x}$. Tekintsük az $\mathcal{L} \cap R$ halmaz B szerinti ősképét:

$$\Lambda = B^{-1}(\mathcal{L} \cap R) = \{\lambda = B^{-1}\mathbf{x} : \mathbf{x} \in \mathcal{L} \cap R\}.$$

Mivel $\mathcal{L} \cap R$ korlátos halmaz és B^{-1}, mint véges dimenziós vektorterek közötti lineáris operátor, korlátos, ezért $\Lambda \subset \mathbb{Z}^n$ is korlátos, de akkor véges, mert bármely korlátos halmazban csak véges sok egész pont található. Így tehát véges sok λ jöhet csak szóba, ezért a B szerinti képeik számossága, $|\mathcal{L} \cap R|$ is véges.

Megfordítva, tegyük fel, hogy $\mathcal{L}$ diszkrét részcsoport V additív csoportjában. Legyen $\mathbf{g_1}, \ldots, \mathbf{g_m}$ az $\mathcal{L}$-et tartalmazó legszűkebb V-beli altér egy olyan bázisa, melyre minden $\mathbf{g_i} \in \mathcal{L}$. Ekkor, mivel $\mathcal{L}$ csoport, $\mathbf{g_i}$-k egész számszorosait (a szokásos ismételt csoportösszeadás értelmében) és ilyenek összegét, így tehát egész értékű lineáris kombinációikat tartalmazza. Kérdés, hogy tartalmaz-e mást. Tekintsük a következő halmazt:

$$X := \left\{\mathbf{x} \in \mathbb{R}^n : \mathbf{x} = \sum_{i=1}^{m} \lambda_i \mathbf{g_i},\ \lambda_i \in [0,1),\ i = 1, \ldots, m\right\},$$

amely tehát $[0,1)^m$ tégla $G = [\mathbf{g_1}, \ldots, \mathbf{g_m}]$ mátrixszal való transzformáltja, egy félig nyílt paralelepipedon, ami korlátos részhalmaza $\mathbb{R}^n$-nek. Mivel X korlátos, $\mathcal{L}$ pedig diszkrét, ezért $X \cap \mathcal{L}$ véges. Válasszuk meg $[\mathbf{g_1}, \ldots, \mathbf{g_m}]$ rendszert úgy, hogy X a lehető legtöbb rácspontot tartalmazza, tehát $|X \cap \mathcal{L}|$ maximális legyen.

1.6. Állítás. *Ekkor* $X \cap \mathcal{L} = \{\mathbf{0}\}$.

Bizonyítás: Indirekt tegyük fel, hogy $X \cap \mathcal{L}$ tartalmaz nem $\mathbf{0}$ vektort. Mivel $\mathcal{L}$ diszkrét, $X \cap \mathcal{L}$ legalábbis véges sok vektort tartalmaz, legyenek ezek $\sum \lambda_i \mathbf{g_i}$ alakúak és rendezzük sorba őket λ koordinátái szerint lexikografikus sorrendben. Vegyük ebben a sorrendben az első, $\mathbf{x}^* = \sum_{i=1}^{m} \lambda_i \mathbf{g_i} \neq \mathbf{0}$ elemet, tehát λ legyen ennek a rögzített elemnek a koordinátavektora és legyen ebben ℓ a legelső (azaz legkisebb) index, amelyre $\lambda_\ell > 0$ (azaz $\lambda_\ell \neq 0$, ilyen létezik, hiszen $\mathbf{x}^* \neq \mathbf{0}$).

Legyen ezután k a legkisebb olyan egész, amelyre λ_ℓ k-szorosa kilóg $[0,1)$ intervallumból, azaz $k\lambda_\ell \geq 1$. Ekkor persze egyúttal $k\mathbf{x}^*$ is legalább a paralelepipedon azon határánál jár, amelyet az nem tartalmaz, $k\mathbf{x}^* \notin X$. Továbbá, mivel k minimális, $(k-1)\lambda_\ell < 1$. Tekintsük a következő vektort

$$\mathbf{y}^* := k\mathbf{x}^* - \sum_{i=\ell}^{m} \lfloor k\lambda_i \rfloor \mathbf{g_i} \in \mathcal{L}, \tag{1}$$

mivel ez csoportelemek egész számszorosainak összege, ill. ($\{\cdot\}$-vel törtrészt jelölve)

$$\mathbf{y}^* = \sum_{i=\ell}^{m} \{k\lambda_i\}\mathbf{g_i} \in X \tag{2}$$

is teljesül, mert $k\lambda_i$ törtrésze nyilván eleme $[0,1)$-nek. Továbbá (1)-ben $\lfloor k\lambda_\ell \rfloor = 1$, ezért $(k-1)\lambda_\ell < 1$, azaz $k\lambda_\ell - 1 < \lambda_\ell$ miatt $\mathbf{y}^*$-nak az ℓ-edik koordinátája szigorúan kisebb, mint $\mathbf{x}^*$-é, ami pedig lexikografikusan minimális volt a nemnulla vektorok között. Tehát csak $\mathbf{y}^* = \mathbf{0}$ lehet, innen

$$\mathbf{g}_\ell = k\mathbf{x}^* - \sum_{i=\ell+1}^{m} \lfloor k\lambda_i \rfloor \mathbf{g_i} = \sum_{i=\ell+1}^{m} \{k\lambda_i\}\mathbf{g_i},$$

$\mathbf{g}_\ell$ kifejezhető $\mathbf{x}^*$-gal és a bázis további elemeivel, mégpedig $[0,1)$-be eső együtthatókkal. Ezért, ha G bázisban $\mathbf{g}_\ell$-et $\mathbf{x}^*$-ra cserélnénk, akkor az $|\mathcal{L} \cap X|$ már tartalmazná $\mathbf{x}^*$-ot is, de $\mathbf{g}_\ell$-t is, tehát szigorúan nagyobb lenne, ellentmondásban G választásával. ∎

Ezzel már befejezhetjük a tétel bizonyítását: ha tehát $X \cap \mathcal{L} = \{\mathbf{0}\}$, akkor $\mathcal{L}$ pontosan $\mathbf{g_i}$-k kombinációit tartalmazza, $\mathcal{L} = \mathcal{L}(G)$, tehát $\mathcal{L}$ valóban rács. ∎

1.7. Példa. *A legkézenfekvőbb $\mathbb{Z}^2$ rácsot már említettük, megjegyezzük, hogy ezek más néven a Gauss-egészek, ha azokra $\mathbb{C}$ helyett $\mathbb{R}^2$-beli pontokként tekintünk. Ezt nagyobb dimenzióra is kiterjeszthetjük: legyen $\mathbb{Z}^n \subset \mathbb{R}^n$ az egész koordinátájú pontok halmaza. Az $\mathbb{R}^n$-beli $(\mathbf{e_i})_{i=1,\dots,n}$ kanonikus bázis láthatóan generálja a $\mathbb{Z}^n$ rácsot is. Valamely $q \in \mathbb{Z}$ esetén tekinthetjük továbbá a $\mathbb{Z}^n$ rács q-szorosát:*

$$q\mathbb{Z}^n := \{\mathbf{z} \in \mathbb{R}^n : \frac{1}{q}\mathbf{z} \in \mathbb{Z}^n\},$$

ennek egy bázisa a kanonikus bázis q-szorosa, $(q\mathbf{e_i})_{i=1,\dots,n}$.

1.8. Definíció. *Legyen $\mathcal{L} = \mathcal{L}(B)$ egy rács az $(\mathbb{R}^n, \langle\cdot,\cdot\rangle)$ térben. Ekkor $\mathcal{L}$ rács $\mathcal{L}^*$ **duális rácsán** azon $z \in \mathbb{R}^n$ vektorok halmazát értjük, melyekre minden $x \in \mathcal{L}$ vektorra $\langle x, z\rangle \in \mathbb{Z}$ teljesül.*

Belátható, hogy $\mathcal{L}^*$ is rács ugyanabban a vektortérben, éspedig a $B^* = (B^{-1})^T$ bázis generálja.
A kriptográfiai és kódelméleti alkalmazásokban nagy szerepet kapnak a q-rácsok (q-ary lattice).

1.9. Definíció. *Azokat az $\mathcal{L}$ rácsokat, amelyekre $q\mathbb{Z}^n \subseteq \mathcal{L} \subseteq \mathbb{Z}^n$ valamely q egész számra, **q-rácsoknak** nevezzük.*

Könnyen látható, hogy egy q-rácshoz az egyes $\mathbf{x} \in \mathbb{Z}^n$ vektorok pontosan $\mathbf{x}$ mod q értéke alapján tartoznak hozzá vagy sem. (Itt $\mathbf{x}$ mod q a koordinátánkénti maradékképzés.) Eszerint a q-rácsok kölcsönösen egyértelműen megfeleltethetők a $\mathbb{Z}_q^n$ feletti lineáris kódoknak [42, 10.1.1. definíció].

További példák rácsokra

Az eddigi példáink $\mathbb{R}^n$ euklideszi térben voltak, a most következő példa az $\mathbb{R}$ feletti, egyváltozós, legfeljebb $n-1$-edfokú polinomok n-dimenziós vektorterében lesz, a skaláris szorzat pedig a polinomok együtthatóinak szorzatösszege lesz, jelöljük ezt a teret P_n-nel.

1.10. Példa. *Ebben a térben az egész együtthatós polinomok rácsot alkotnak, a $\mathcal{P}$ rácsot (egyszerűen belátható, hogy $\mathcal{P}$ diszkrét részcsoportja P_n-nek). $\mathcal{P}$-nek egy lehetséges bázisa a*

$$B = [\mathbf{1}, \mathbf{x}, \mathbf{x^2}, \ldots, \mathbf{x^{n-1}}]$$

rendszer.

Legyen mondjuk $n = 5$, ekkor figyelembe véve, hogy a Pascal-mátrix:

$$S_5 = U_5 L_5 = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 \\ 0 & 0 & 1 & 3 & 6 \\ 0 & 0 & 0 & 1 & 4 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 \\ 1 & 3 & 3 & 1 & 0 \\ 1 & 4 & 6 & 4 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 \\ 1 & 3 & 6 & 10 & 15 \\ 1 & 4 & 10 & 20 & 35 \\ 1 & 5 & 15 & 35 & 70 \end{pmatrix}$$

láthatóan unimoduláris (hiszen unimodulárisak szorzata is az), ezzel transzformálva az eredeti B bázist, kapjuk, hogy az alábbi vektorokból álló $C = [c_1, \ldots, c_5]$ rendszer:

$$\begin{aligned} c_1 &= x^4 + x^3 + x^2 + x + 1 \\ c_2 &= x^4 + 2x^3 + 3x^2 + 4x + 5 \\ c_3 &= x^4 + 3x^3 + 6x^2 + 10x + 15 \\ c_4 &= x^4 + 4x^3 + 10x^2 + 20x + 35 \\ c_5 &= x^4 + 5x^3 + 15x^2 + 35x + 70 \end{aligned}$$

ugyanazt a $\mathcal{P}$ egész együtthatós polinomokból álló rácsot generálja.
A következő két példa ugyan újra a megszokott $\mathbb{R}^n$ euklideszi térből származik, konstrukciójuk azonban jóval bonyolultabb, mint az eddigi példák.

1.11. Példa. *Egy jól ismert példa rácsokra a $\mathcal{C}$ **Leech-rács**, amely a valós test fölötti 24-dimenziós euklideszi térben egy 8-rács $8^{-\frac{1}{2}}$-szerese. Egy bázisának mátrixa*

megtalálható [32] cikkben, de mivel 24×24*-es, itt inkább kihasználjuk, hogy egy* 8*-rács elemei csak a vektorok koordinátáinak modulo* 8 *vett értékeitől függnek és* $\mathcal{C}' = \sqrt{8}\,\mathcal{C}$ 8*-rács. Eszerint bármely* $\mathbf{x} \in \mathbb{R}^{24}$ *vektorra* $\frac{1}{\sqrt{8}}\mathbf{x} \in \mathcal{C}$*, azaz* $\mathbf{x} \in \mathcal{C}'$ *pontosan akkor, ha* $\mathbf{z} := \mathbf{x} \bmod 8$ *vektorra a következők igazak.* $\mathbf{z}$ *minden koordinátáját tekintsük bináris alakban, így tehát van* $3 \cdot 24 = 72$ *darab bitünk:*

$$z_i = z_i^2 \cdot 2^2 + z_i^1 \cdot 2^1 + z_i^0 \cdot 2^0 \qquad (i = 1, \ldots, 24),$$

ekkor azok a vektorok $\mathcal{C}'$*-beliek, melyekhez tartozó* $\mathbf{z}$*-re egyrészt a középső bitek által alkotott* $(z_1^1, z_2^1, \ldots, z_{24}^1)$ *számsorozat eleme a* C_{24} *kiterjesztett Golay-kódnak [42, 10.2.6–7. tétel], másrészt*

$$\begin{array}{c|ccccc} & \overbrace{\qquad\qquad\qquad\qquad}^{z_i^2\text{-kben páros sok 1-es}} & & & & \\ 2^2 & z_1^2 & z_2^2 & \ldots & z_{23}^2 & z_{24}^2 \\ 2^1 & z_1^1 & z_2^1 & \ldots & z_{23}^1 & z_{24}^1 \\ 2^0 & 0 & 0 & \ldots & 0 & 0 \end{array} \in C_{24} \qquad \textit{vagy} \qquad C_{24} \ni \begin{array}{ccccc} \overbrace{\qquad\qquad\qquad\qquad}^{z_i^2\text{-kben páratlan sok 1-es}} & & & & \\ z_1^2 & z_2^2 & \ldots & z_{23}^2 & z_{24}^2 \\ z_1^1 & z_2^1 & \ldots & z_{23}^1 & z_{24}^1 \\ 1 & 1 & \ldots & 1 & 1 \end{array}$$

(ahol minden oszlop egy modulo 8 *maradékosztály* 3 *biten ábrázolva, a vektor egy-egy koordinátája az elsőtől a huszonnegyedikig), azaz az adott vektorra vagy minden* $z_i^0 = 1$ *(*$i = 1, \ldots, 24$*, a vektor minden koordinátája páratlan) vagy minden* $z_i^0 = 0$ *(*$i = 1, \ldots, 24$*, minden koordináta páros) és előbbi esetben* z_i^2*-k (*$i = 1, \ldots, 24$*) között is páratlan sok* 1*-es van, utóbbiban páros sok.*
Tömören megfogalmazva $(z_1^1, \ldots, z_{24}^1) \in C_{24}$ *és*

$$x_1 + \ldots + x_{24} \equiv 4x_1 \equiv 4x_2 \equiv \ldots \equiv 4x_{24} \mod 8.$$

A $\sqrt{8}$*-cal való normálás értelme, hogy így a* $\mathcal{C}$ *rács determinánsa* 1 *lesz.*

A $\mathcal{C}$ Leech-rács avagy Leech-féle lineáris kód jelentősége hatalmas a matematika különböző területein. A csoportelméletben J. H. Conway és J. Thompson a Leech-rács automorfizmus-csoportja segítségével fedeztek fel három új sporadikus csoportot, a Co_1, Co_2, Co_3 Conway-csoportokat [32, 7]. Szerepe jelentős továbbá a gömbpakolási feladatban – melyben adott dimenziós euklideszi térben egységgömbök legsűrűbb átfedés nélküli elhelyezkedését keressük – és az ehhez kapcsolódó Kissing Number problémában is – azaz, hogy adott dimenziós euklideszi térben egy egységgömböt legfeljebb hány másik érinthet átfedés nélkül. C. Henry, A. Kumar, S. D. Miller, D. Radchenko és M. Viazovska 2016-ban belátták [47], hogy 24 dimenzióban nem létezik sűrűbb gömbpakolás, mint a $\mathcal{C}$ rácspontjai által meghatározott.
Hasonló, de egyszerűbb, 8-dimenziós konstrukció az E_8 rács.

1.12. Példa. *Tekintsük a valós test fölötti 8-dimeziós euklideszi térben a következő két halmaz unióját:*

$$E_A := \{\mathbf{x} \in \mathbb{Z}^8 : \sum_{i=1}^{8} x_i \equiv 0 \mod 2\}$$

és

$$E_B := \{\mathbf{x} \in \mathbb{R}^8 : \forall i : \ x_i - \frac{1}{2} \in \mathbb{Z}, \sum_{i=1}^{8} x_i \equiv 0 \mod 2\}.$$

Ekkor $E_8 := E_A \cup E_B$ rácsot alkot $\mathbb{R}^8$-ban, amelynek neve ***Gosset-rács*** *vagy röviden E_8 rács.*

Az E_8 rács egy másik (kiterjesztett) perfekt kóddal, a 8 hosszúságú, 4 dimenziós Hamming kóddal van szoros kapcsolatban, fontossága a 8-dimenziós gömbpakolásban és Kissing Number problémában jelentkezik [46, 48].

1.3. Nevezetes feladatok rácsokon

Rögzítsünk egy V vektortéren értelmezett $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{R}$ skalárszorzatot és az általa indukált $\| \cdot \|$ normát. A rácsokkal kapcsolatos egyik legfontosabb kérdés a rácsban lévő legrövidebb vektor hossza (az indukált normában), azaz a Shortest Vector Problem (SVP), melyben a feladat tehát olyan $\mathbf{0}$ vektortól különböző $u \in \mathcal{L}$ vektor megtalálása, melyre $\|u\|$ minimális. (A $\mathbf{0}$ vektor minden rácsnak eleme, így ezt az érdektelen esetet kizárjuk a legrövidebb vektor problémájából.) Ezzel rokon feladat a Closest Vector Problem (CVP), mely egy rögzített térbeli ponthoz legközelebbi rácspont megtalálását jelenti. (Itt tehát megengedjük, hogy a rögzített pont éppen rácspont – a távolság éppen 0 – legyen.) A feladatokat a rács bázisa segítségével tömören megfogalmazhatjuk:

$$\mathbf{0} \neq x \in \mathbb{Z}^n : \|Bx\| \to \min \tag{SVP}$$

$$x \in \mathbb{Z}^n : \|Bx - t\| \to \min \tag{CVP}$$

ahol $t \in \mathbb{R}^n$ adott. Egy rácsban a legrövidebb (ill. a legközelebbi) vektor hossza nyilván csak a rácspontoktól és így a rácstól függ, a rácsot generáló bázistól nem.
A továbbiakban egy $\mathcal{L}$ rács legrövidebb nem $\mathbf{0}$ vektorának hosszát $\lambda(\mathcal{L})$-lel fogjuk jelölni.
E kettő mellett előfordul még egy harmadik feladat, a legrövidebb független vektorok problémája, a Shortest Independent Vector Problem (SIVP). Ez utóbbiról kevés szó lesz a szakdolgozatban, a teljesség kedvéért azonban definiáljuk. Az SIVP feladatban

egy B bázissal adott $\mathcal{L}(B) \subseteq \mathbb{R}^n$ rácsban keresendő n darab független rácsvektor $S = [s_1, \ldots, s_n]$, melyre

$$\|S\| := \max_i \|s_i\| \to \min,$$

azaz olyan S, ami a leghosszabb S-beli vektor hosszát minimalizálja.
A szakdolgozatban elsősorban az SVP, ill. az ehhez szorosan kapcsolódó CVP feladatról, ezek megoldásáról lesz szó, ehhez azonban erősen támaszkodni fogunk a Gram–Schmidt-együtthatókra, ezért röviden kitérünk rájuk.

1.4. Gram–Schmidt-ortogonalizáció

A jól ismert Gram–Schmidt-ortogonalizáció eljárása adott $b_1, \ldots, b_n$ lineárisan független vektorrendszerhez olyan $b_1^*, \ldots, b_n^*$-ot rendel, melyek páronként merőlegesek egymásra, $b_1^* = b_1$, továbbá $i = 2, \ldots, n$-re $b_i^{\|} := b_i - b_i^*$ megegyezik b_i vektor $\operatorname{span}(b_1, \ldots, b_{i-1})$ altérre vonatkozó merőleges vetületével, másképp fogalmazva b_i^* vektor b_i projekciója $\operatorname{span}(b_1, \ldots, b_{i-1})$ altér ortogonális kiegészítő alterére. Tehát

$$b_i^{\|} \in \operatorname{span}(b_1, \ldots, b_{i-1}), \text{ és}$$
$$b_i^{\perp} := b_i^* \in \operatorname{span}(b_1, \ldots, b_{i-1})^{\perp}.$$

Figyeljük meg, hogy $b_i = b_i^{\|} + b_i^*$ miatt

$$\{b_1, \ldots, b_{i-1}, b_i\} \text{ és } \{b_1, \ldots, b_{i-1}, b_i^*\}$$

ugyanazt az alteret feszítik, tehát indukcióval

$$\operatorname{span}(b_1, \ldots, b_i) \text{ és } \operatorname{span}(b_1^*, \ldots, b_i^*)$$

terek is azonosak, ezért $b_i - b_i^* \in \operatorname{span}(b_1^*, \ldots, b_{i-1}^*)$ is igaz. Vezessük be még a teljesség kedvéért a $b_1^{\|} := \mathbf{0}$ jelölést. Ekkor tehát bármely i-re $b_i = b_i^{\perp} + b_i^{\|}$ a vektor felbontása az i-nél kisebb indexű vektorok által generált altérbe eső, ill. az altérre merőleges összetevőre.
Legyen $b_1, \ldots, b_n$ vektorok Gram–Schmidt-ortogonalizáltja a $b_1^*, \ldots, b_n^*$ rendszer. Mivel $b_i - b_i^* \in \operatorname{span}(b_1^*, \ldots, b_{i-1}^*)$, b_i vektor felírható a következő alakban:

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

$\mu_{ii} := 1$ kiterjesztéssel a b_i^* bevihető a szummába, sőt, μ_{ij} együtthatót $j > i$ esetre 0-ként kiterjesztve az alábbi alakot kapjuk:

$$b_i = \sum_{j=1}^{n} \mu_{ij} b_j^*, \tag{GS1}$$

ahol a μ_{ij} $(i, j = 1, \ldots, n)$ együtthatók minden i, j-re egyértelműen meghatározottak és $j < i$ esetén a következő képlet szolgáltatja őket:

$$\mu_{ij} = \frac{\langle b_i, b_j^* \rangle}{\langle b_j^*, b_j^* \rangle} \quad (j < i) \tag{GS2}$$

(ezeket a Gram–Schmidt-ortogonalizáció algoritmusa ki is számítja).

Így tehát bármely vektorrendszerre értelmes az adott rendszer **Gram–Schmidt-együtthatóiról** beszélni, ami alatt a fenti együtthatókat értjük. Megjegyezzük azonban, hogy a vektorrendszer Gram–Schmidt-együtthatói függnek a vektorrendszer elemeinek sorrendjétől.

1.5. Alapvető rácsparaméterekre vonatkozó becslések

Az alábbi rövid fejezet a rácsok számunkra két legfontosabb paraméterére, $\lambda(\mathcal{L})$-re és $\det(\mathcal{L})$-re vonatkozó ismert becsléseket foglalja össze.

Az Hadamard-egyenlőtlenség szerint tetszőleges lineárisan független $b_1, \ldots, b_n \in \mathbb{R}^n$ vektorokra és $B \in \mathbb{R}^{n \times n}$ mátrixukra

$$|\det(B)| \leq \prod_{i=1}^{n} \|b_i\|,$$

ahol $\|\cdot\|$ az euklideszi norma és egyenlőség pontosan akkor áll, ha a b_i vektorok ortogonálisak. Ez szemléletes tény, figyelembe véve, hogy $|\det(B)|$ azon paralelepipedon térfogata, melynek éleit a B rendszer vektorai adják.

Ha a $b_1, \ldots, b_n$ vektorokat most egy rács bázisának gondoljuk, akkor a determináns lineáritásából, illetve abból a tulajdonságból, hogy lineárisan összefüggő vektorok (így $\{b_1, \ldots, b_k, b_{k+1}^{\|}\}$) esetén a determináns értéke 0, könnyen belátható, hogy egy bázisnak és ortogonalizáltjának a determinánsa megegyezik, következésképpen az ortogonalizáltak hosszával a rács determinánsa kifejezhető.

1.13. Állítás. $\det(\mathcal{L}) = |\det(B)| = |\det(B^*)| = \prod_{i=1}^{n} \|b_i^*\|$

Másrészt Hermite belátta, hogy bármely $\mathcal{L}$ rácsnak van olyan $b_1, \ldots, b_n$ bázisa, melyre

$$\prod_{i=1}^{n} \|b_i\| \leq c_n \det(\mathcal{L}),$$

ahol c_n csak a dimenziótól függő konstans, továbbá azt is tudjuk, hogy c_n legkisebb értéke n^n-nél kisebb.

Megjegyezzük, hogy azon felül, hogy a bevezetőben láttuk, hogy tetszőleges $c_1, \ldots, c_n$ független vektorokat választva az $\mathcal{L}$ rácsból, a kapott rendszer általában nem bázisa a rácsnak, az intuíciónak némiképpen ellentmondva olyan példa is adható ilyen rendszerre, ahol a vektorok normáinak szorzata kisebb $\det(\mathcal{L})$-nél [23, 16. o.].

Minkowski híres tétele a legnagyobb olyan konvex, szimmetrikus testről szól, amely csakis egyetlen rácspontot tartalmaz [14, 1.4. tétel] (illetve speciálisan 2 dimenzióra és elemi eszközökkel bizonyítva lásd [28, 8.2.1. tétel]).

1.14. Tétel (Minkowski). *Legyen adott $\mathcal{L}(B)$ rács és egy, az origóra vonatkozóan középpontosan szimmetrikus, konvex S test a bázisvektorok által feszített térben. Ha az S test térfogatára $vol(S) > 2^n \det(\mathcal{L})$, akkor a test tartalmaz $\mathbf{0}$ vektoron kívül legalább még egy rácspontot.*

A következő lemma már a báziselemek és a legrövidebb vektor kapcsolatáról szól: kimondja, hogy egy rács legrövidebb vektora nem lehet hosszabb, mint a bázis Gram–Schmidt-ortogonalizáltjainak bármelyike.

1.15. Lemma. *Legyen $\mathcal{L}$ rács bázisa $b_1, \ldots, b_n$, ennek ortogonalizáltja $b_1^*, \ldots, b_n^*$. Ekkor a rács legrövidebb vektorának hosszára fennáll a következő becslés:*

$$\lambda(\mathcal{L}) \leq \min\{\|b_1^*\|, \ldots, \|b_n^*\|\}.$$

Bizonyítás: Legyen $v \in \mathcal{L}$, $v \neq 0$, ekkor v felírható a báziselemek lineáris kombinációjaként:

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

ahol a végéről 0 együtthatókat elhagyva esetleg $k < n$, viszont így $\lambda_k \neq 0$ és ilyen k mindig létezik $v \neq 0$ miatt. Továbbá b_i bázisvektorok kifejezhetők az ortogonalizáltjaikkal, így kapjuk:

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

Utóbbi felírásban $\lambda_k' \in \mathbb{Z}$ és $\lambda_k' \neq 0$, mert $\mu_{kk} = 1$. Innen:

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

és ezzel megkaptuk a bizonyítandó állítást. ∎

2. A Lenstra–Lenstra–Lovász algoritmus

Ebben a részben a rácsok vizsgálatában mérföldkőnek számító, 1982-ben született LLL algoritmust mutatjuk be, amely polinomiális futásidejű és bár végeredményben az SVP feladat megoldását csak a dimenzióban exponenciális hibával közelíti, ez a gyakorlatban lehengerlően sok esetben elegendőnek bizonyul. Innentől a dolgozat további részeiben rács alatt a racionális számtest feletti vektortérben generált rácsot fogunk érteni, ahogy azt a bevezetőben már jeleztük.
A legrövidebb vektor meghatározásának egyik módja, ha belátjuk, hogy az adott rácsban a legrövidebb vektor történetesen éppen a megadott rácsbázis egyik eleme, vagy legalábbis a megadott bázis átalakítható úgy, hogy ez teljesüljön. Az utóbbiról szól a következő részben szereplő a 2.6. algoritmus, az előbbi esetre pedig a legegyszerűbb példa a korábban említett $\mathbb{Z}^n$, vagy eggyel általánosabban bármely olyan $\mathcal{L}(C)$ rács, melynek C bázisában a vektorok egymásra páronként merőlegesek. Ilyenkor persze a bázis megegyezik az ortogonalizáltjával, $C^* = C$. A merőlegesség miatt egyik báziselemhez másikat adva csak az eredetinél hosszabbat kaphatunk, ha ezt összevetjük az 1.15. lemmával, felhasználva, hogy itt $c_i^* = c_i$, azonnal kapjuk, hogy a rács legrövidebb vektorát a báziselemek között kell keresnünk.
Ez tehát bizonyos értelemben a legjobb eset. Az ettől az esettől való távolság mérésére bevezetjük a rácsbázisra vonatkozó, következő mennyiséget.

2.1. Definíció. *Legyen egy $B = [b_1, \ldots, b_n]$ bázis által generált $\mathcal{L}(B)$ rácsra a*

$$\Delta(B) = \frac{\prod_{i=1}^{n} \|b_i\|}{\det \mathcal{L}(B)},$$

a bázis ún. ***merőlegességi defektusa****.*

Az Hadamard-egyenlőtlenség és az 1.13. állítás összevetéséből következik, hogy bármely rács bármely bázisára $\Delta(B) \geq 1$, az előző példában pedig láthatóan $\Delta(C) = 1$. Két dimenzióban azonban nem kell ilyen szigorú feltételeknek teljesülnie ahhoz, hogy az egyik báziselem legyen a rács legrövidebb vektora. Erre a következő részben egy jóval enyhébb elégséges feltételt fogunk adni.

2.1. Gyenge redukált

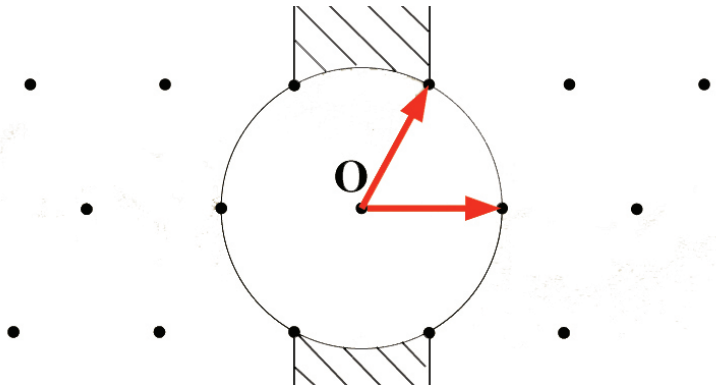
Legyen $\mathcal{L}$ egy rács a $\mathbb{Q}^2$ térben. A rács bázisának Gram–Schmidt-együtthatói segítségével megfogalmazható egy olyan, egyszerű elégséges feltétel, amelynek fennállásakor a rács legrövidebb vektora szükségképpen a bázis egyik eleme.

2.2. Definíció. *Legyen $\mathcal{L}$ kétdimenziós rács a b_1, b_2 bázissal adott. Ha a*

$$\|b_1\| \leq \|b_2\|$$
$$|\mu_{21}| \leq \frac{1}{2}$$

feltételek fennállnak, akkor a bázist ***Gauss- vagy gyengén redukáltnak*** *nevezzük. [13], [34]*

Fogalmazzuk meg szóban is a gyenge redukáltság feltételét: az kell tehát hozzá, hogy a két bázisvektor közül a második legyen hosszabb (legalábbis nem rövidebb) és az elsőre vonatkozó vetületének hossza az első vektor hosszának legfeljebb fele legyen. Utóbbi feltételt úgy is fogalmazhatjuk, hogy a két bázisvektor által bezárt szög 60° és 90° között van. A gyenge redukáltságból már következik, hogy a rács legrövidebb vektora éppen a bázis első eleme, ez az alábbi, 2. ábráról is könnyen látszik, ugyanis a második bázisvektornak szükségképpen a staírozott részbe kell esnie. (Az ábra forrása [27, p76].)



2. ábra. Gauss-redukált bázis 2 dimenzióban

Ezt az észrevételt fontossága miatt külön állításként is megfogalmazzuk.

2.3. Állítás. *Legyen $\mathcal{L}$ rács $B = [b_1, b_2]$ bázisa gyengén redukált. Ekkor a rács legrövidebb vektora éppen b_1.*

Láttuk, hogy a gyenge redukáltság 2 dimenzióban kiemelkedően hasznosnak bizonyul: fennállása esetén a rács legrövidebb vektorát a báziselemek között találjuk. Hasonlóan erős állításokat csak alacsony dimenziójú rácsok esetén ismerünk. 3 dimenzióban lásd Brigitte Vallée, ill. Igor Semaev munkáit [43, 35], 4 dimenzióban Phong Q. Nguyễn és Damien Stehlé cikkét [40].
Mindazonáltal a gyenge redukáltság fogalmára magasabb dimenzióban is szükségünk lesz.

2.4. Definíció. *Legyen az $\mathcal{L}$ rács dimenziója n, bázisa $b_1, \ldots, b_n$, a bázis Gram–Schmidt-együtthatói: μ_{ij} $(i, j = 1, \ldots, n)$. Ha a Gram–Schmidt-együtthatók teljesítik a következő feltételt:*

$$|\mu_{ij}| \leq \frac{1}{2}$$

minden $j < i$ esetén, akkor a bázist ***gyengén redukáltnak*** *nevezzük.*

Bármely bázis polinomiális időben gyengén redukálttá tehető, erre algoritmikus bizonyítást adunk.

2.5. Állítás. *Legyen $\mathcal{L}$ rács egy bázisa $b_1, \ldots, b_n$, ennek Gram–Schmidt-ortogonalizáltja $b_1^*, \ldots, b_n^*$, a hozzá tartozó együtthatók μ_{ij} $(i, j = 1, \ldots, n)$. Ekkor $\mathcal{L}$ rácsnak létezik egy $\overline{b}_1, \ldots, \overline{b}_n$ gyengén redukált bázisa, melynek ortogonalizáltja szintén $b_1^*, \ldots, b_n^*$.*

2.6. Algoritmus.

1) Kezdetben legyen $b_1, \ldots, b_n$ az $\mathcal{L}$ rács egy tetszőleges bázisa, $b_1^*, \ldots, b_n^*$ a bázis ortogonalizáltja és μ_{ij} a bázishoz tartozó Gram–Schmidt-együtthatók.

2) Amíg létezik $(i, j) \in \{1, \ldots, n\} \times \{1, \ldots, n\}$, melyre $j < i$ és $|\mu_{ij}| > \frac{1}{2}$:

> *válasszunk ezen (i, j)-k közül egy olyat, ahol j maximális (ezen belül i tetszőleges). Jelölje $\lfloor \mu_{ij} \rfloor$ és $\lceil \mu_{ij} \rceil$ közül a μ_{ij}-hez közelebbit $[\mu_{ij}]$, az i-edik báziselemet pedig változtassuk a következőre:*
>
> $$b_i := b_i - [\mu_{ij}] \cdot b_j.$$

3) Adjuk vissza $(\overline{b}_i) := (b_i)$ bázist.

2.7. Állítás. *A 2.6. algoritmus felfeljebb $\binom{n}{2}$ számú iteráció után véget ér és eredményül gyengén redukált bázist az vissza.*

Bizonyítás: Mivel az algoritmus fő ciklusának kiugrási feltétele éppen a gyenge redukáltság definíciója, világos, hogy az algoritmus ilyen bázissal tér vissza, már ha véges időben valóban lefut.
A végesség bizonyításához belátjuk, hogy az algoritmus minden szóba jövő (i, j) párt legfeljebb egyszer javít meg és a továbbiakban már nem rontja azt el. Pontosabban azt látjuk be, hogy az éppen aktuális μ_{ij} együtthatót a fő lépés legfeljebb $\frac{1}{2}$ abszolútértékűre állítja, közben esetleg elront egyes együtthatókat, de a korábbiakban kezelésbe vett együtthatókat már nem rontja el.

Ennek első lépéseként belátjuk, hogy az ortogonalizáltak végig változatlanok maradnak az algoritmus futása során.
Tekintsuk az iteráció fő lépését egy adott b_i báziselemre vonatkozóan. A Gram–Schmidt-ortogalizációról szóló részben szereplő

$$b_i = b_i^{\perp} + b_i^{\parallel} \qquad (b_i^{\parallel} \in \operatorname{span}(b_1, \ldots, b_{i-1}))$$

alakot felhasználva látjuk, hogy $b_j \in \operatorname{span}(b_1, \ldots, b_{i-1})$ miatt a b_j-vel párhuzamos eltolás legalábbis $b_i^* = b_i^{\perp}$ vektort változatlanul hagyja. Emiatt azonban az is igaz, hogy ugyanettől a lépéstől az i-nél nagyobb k indexekre μ_{ki} értéke nem változik meg, tehát a b_k^* ortogonalizáltak sem. Az i-nél korábbi ($k < i$ indexű) báziselemek pedig nyilvánvalóan nem érintettek a változtatásban, tehát egyik ortogalizált sem változott.
Az algoritmus tehát veszi a rossz (i, j) párokat és j szerint csökkenő (legalábbis nem növekvő) sorrendben sorban végigveszi őket. Az iterációs lépés utáni új bázist és együtthatókat jelöljük vesszővel: b_i' és μ_{ij}'. (Mivel azt már beláttuk, hogy az ortogonalizáltak nem változnak, ezért az egyszerűbb jelölés kedvéért azokra változatlanul b_i^*-ként fogunk hivatkozni.)
A $[\mu_{ij}] \cdot b_j$-vel való eltolás, b_i vektor $b_i^{\parallel}$ összetevőjét már megváltoztatja, méghozzá:

$$b_i' = b_i - [\mu_{ij}]b_j = \sum_{t=1}^{i} \mu_{it} b_t^* - [\mu_{ij}] \sum_{t=1}^{j} \mu_{jt} b_t^* = \sum_{t=1}^{n} \mu_{it} b_t^* - [\mu_{ij}] \sum_{t=1}^{n} \mu_{jt} b_t^*$$

(kihasználva, hogy $t > i$-re, ill. $t > j$-re a μ_{it}, ill. μ_{jt} együtthatók 0-k), tehát

$$b_i' = \sum_{t=1}^{n} \Big(\underbrace{\mu_{it} - [\mu_{ij}] \cdot \mu_{jt}}_{\mu_{it}'} \Big) b_t^*,$$

ahol tehát

$$\mu_{it}' = \begin{cases} \mu_{it} - [\mu_{ij}] \cdot \mu_{jt}, & \text{ha } t = 1, \ldots, j; \\ \mu_{it}, & \text{ha } t = j+1, \ldots, i; \\ 0, & \text{ha } t = i+1, \ldots, n. \end{cases}$$

Ezek közül $\mu_{i1}', \ldots, \mu_{i,j-1}'$ együtthatókról nem tudunk semmit, de egyrészt j maximalitása miatt tudjuk, hogy $t = j+1, \ldots, i$-re (sőt, $t = j+1, \ldots, n$-re) $|\mu_{it}'| = |\mu_{it}| \leq \frac{1}{2}$, másrészt most már $|\mu_{ij}'| = |\mu_{ij} - [\mu_{ij}] \cdot \mu_{jj}| = |\mu_{ij} - [\mu_{ij}] \cdot 1| \leq \frac{1}{2}$ is fennáll, tehát az adott i-hez a legnagyobb j, ami rossz (i, j) párt eredményez, csökkent. Persze ugyanez a j más i-kkel is alkothat rossz (i, j) párokat, de csak $i > j$ feltételnek eleget tevő i-kre (ahogy ezt a főciklus feltételében is olvashatjuk), így az adott j-hez legfeljebb $n - j$ rossz i található, ezeket tetszőleges sorrendben eliminálva, a maximális j végül is csökken.

Összességében véve legfeljebb $\binom{n}{2}$ rossz pár lehetséges, ezeken sorban végigmenve, az algoritmus legfeljebb $n(n-1)/2$ iteráció után leáll. ∎

2.2. LLL-redukció

Lovász-redukált

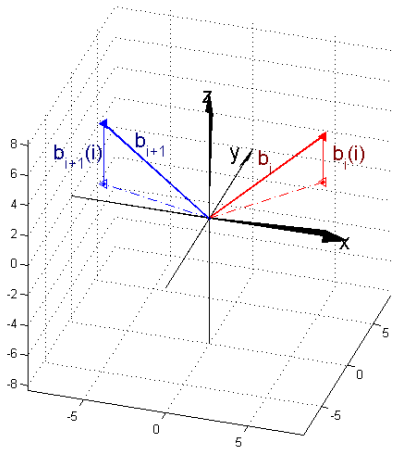
A Lovász-redukált definíciója előtt bevezetünk egy újabb jelölést. Legyen $B = (b_1, \ldots, b_n)$ bázis $V = \mathbb{Q}^n$-ben. A Gram–Schmidt-ortogonalizáltnál láttuk egy vektor $b_k = b_k^{\parallel} + b_k^{\perp}$ direkt felbontását az őt megelőző vektorok által feszített térbe eső, ill. arra merőleges részre, melyben a merőleges rész éppen a vektor ortogonalizáltját szolgáltatta. Ez a felbontás persze megtehető $\mathrm{span}(b_1, \ldots, b_{k-1})$ altéren kívül bármely, kevesebb vektor által kifeszített térre is, mégpedig: ha v egy tetszőleges vektor a teljes B rendszer által feszített térben, $V_j = \mathrm{span}(b_1, \ldots, b_j)$ az első j vektor által feszített altér, akkor v felírható ezen altér szerinti direkt összegként is: $v = v_{V_j}^{\parallel} + v_{V_j}^{\perp}$, ahol tehát $v_{V_j}^{\parallel}$ jelöli v vektor V_j altérre vonatkozó vetületét, $v_{V_j}^{\perp}$ pedig az altérre merőleges összetevőt. Ezzel a jelöléssel tehát az ortogonalizált másképpen így is írható: $b_k = (b_k)_{V_{k-1}}^{\parallel} + (b_k)_{V_{k-1}}^{\perp} = b_k^{\parallel} + b_k^*$.
A továbbiakban rögzítettnek gondolt $B = (b_1, \ldots, b_n)$ mellett tetszőleges v vektor azon komponensét, ami a j-nél kisebb indexű vektorok által generált altérre (azaz V_{j-1}-re) merőleges, jelöljük $v(j)$-vel. (Azaz $v(j) := v_{V_{j-1}}^{\perp}$ és így $b_k^* = (b_k)_{V_{k-1}}^{\perp} = b_k(k)$.) Az imént bevezetett jelöléssel már egyszerűen megfogalmazhatjuk a Lovász-redukált definícióját.

2.8. Definíció. *Legyen $B = (b_1, \ldots, b_n)$ bázis $\mathbb{Q}^n$-ben, V_j az első j eleme által feszített altér és $b_k(j) = (b_k)_{V_{j-1}}^{\perp}$, mint a bevezetőben. Azt mondjuk, hogy B **Lovász-redukált bázis**, ha egyrészt gyengén redukált, másrészt szomszédos elemei teljesítik a következő egyenlőtlenséget:*

$$\|b_i(i)\|^2 \leq \frac{4}{3}\|b_{i+1}(i)\|^2 \tag{LR}$$

minden $i = 1, \ldots, n-1$ esetén. [23, 21]

Vegyük észre, hogy (LR) egyenlőtlenség bal oldalán az i-edik, jobb oldalán az $i+1$-edik bázisvektor V_{i-1} altérre merőleges összetevője, ezek normanégyzete van. Úgy is tekinthetjük, hogy a bal oldalon az i-edik bázisvektor ortogonalizáltja szerepel, a jobb oldali vektor pedig akkor lenne az ott szereplő, $i+1$-edik bázisvektor ortogonalizáltja, ha az i-edik és $i+1$-edik bázisvektor sorrendjét felcserélnénk (lásd a 3. ábrát).



3. ábra. $b_i(i)$ és $b_{i+1}(i)$ viszonya (V_{i-1} altér az x–y sík)

2.9. Tétel. *Legyen $(b_1, \ldots, b_n)$ Lovász-redukált bázisa az $\mathcal{L}$ rácsnak. Ekkor fennállnak az alábbi becslések:*

(1) $\|b_1\| \leq 2^{\frac{n-1}{2}} \lambda(\mathcal{L})$

(2) $\|b_1\| \leq 2^{\frac{n-1}{4}} \sqrt[n]{\det(\mathcal{L})}$

(3) $\|b_1 \cdot \cdots \cdot b_n\| \leq 2^{\frac{1}{2}\binom{n}{2}} \det(\mathcal{L})$

Bizonyítás: Tegyük fel, hogy B Lovász-redukált bázis. Feltehető, hogy B elemei egész vektorok, ugyanis a közös nevezővel való beszorzással ez elérhető és mivel a kapott rács az eredetinek egyszerűen egy egész számmal vett megnyújtása, az minden számunkra lényeges szempontból hasonló az eredetihez. (Mind a gyenge, mind a Lovász-redukáltság nyilván skálázás-invariáns tulajdonság, továbbá ha a közös nevező $m \in \mathbb{Z}_+$, akkor (1) és (2) mindkét oldala m-mel, (3) mindkét oldala m^n-nel szorzódik.) Ha tehát B Lovász-redukált, akkor

$$\begin{aligned}\|b_i^*\|^2 = \|b_i(i)\|^2 &\leq \frac{4}{3}\|b_{i+1}(i)\|^2 = \\ &= \frac{4}{3}\|b_{i+1}^* + \mu_{i+1,i} b_i^*\|^2 = \\ &= \frac{4}{3}\|b_{i+1}^*\|^2 + \frac{4}{3}\mu_{i+1,i}^2 \|b_i^*\|^2 \leq \\ &\leq \frac{4}{3}\|b_{i+1}^*\|^2 + \frac{1}{3}\|b_i^*\|^2,\end{aligned}$$

(sorban (LR), a merőlegesség és a gyenge redukáltság felhasználásával) és így:

$$\|b_{i+1}^*\|^2 \geq \frac{1}{2}\|b_i^*\|^2,$$

majd innen indukcióval:

$$\|b_i^*\|^2 \geq 2^{1-i}\|b_1^*\|^2 = 2^{1-i}\|b_1\|^2 \qquad (i = 1, \ldots, n). \tag{*}$$

Mivel ez minden i-re fennáll:

$$\|b_1\|^2 \leq \min_i\{2^{i-1}\|b_i^*\|^2\} \leq 2^{n-1}\min_i \|b_i^*\|^2 \leq 2^{n-1}\lambda(\mathcal{L})^2$$

az 1.15. lemma alapján kaptuk (1)-et. Másrészt összeszorozva minden i-re az egyenlőtlenségeket:

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

az utolsó egyenlőségnél felhasználva az 1.13. állítást. Ennek $2n$-edik gyöke:

$$\|b_1\| \leq 2^{\frac{n-1}{2}}\sqrt[n]{\det(\mathcal{L})},$$

azaz éppen (2). Végül (3) bizonyításához belátjuk, hogy $\|b_i\|$ becsülhető $\|b_i^*\|$ segítségével, mégpedig a Lovász-redukáltság:

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

majd pedig (*) felhasználásával:

$$\sum_{j=1}^{i-1}\frac{1}{4}\|b_j^*\|^2 + \|b_i^*\|^2 \leq \Big(1 + \frac{1}{4}\big(2 + \ldots + 2^{i-1}\big)\Big)\|b_i^*\| \leq 2^{i-1}\|b_i^*\|^2.$$

Ezt minden i-re összeszorozva:

$$\prod_{i=1}^{n}\|b_i\|^2 \leq 2^{\frac{n-1}{2}n}\prod_{i=1}^{n}\|b_i^*\|^2 = 2^{\binom{n}{2}}\big(\det(\mathcal{L})\big)^2.$$

ezzel (3)-at is beláttuk. ∎

Az LLL algoritmus

Ez alapján megadhatunk egy polinomiális algoritmust, mely adott rács legrövidebb vektorát a dimenzióban exponenciális hibával közelíti. Az alábbi algoritmus Arjen Klaas Lenstra, Hendrik Willem Lenstra Jr. és Lovász László eredménye. [21]

2.10. Algoritmus (Lenstra–Lenstra–Lovász, LLL).

1) Kezdetben legyen $b_1, \ldots, b_n$ az $\mathcal{L}$ rács egy tetszőleges bázisa, $b_1^*, \ldots, b_n^*$ a bázis ortogonalizáltja és μ_{ij} a bázishoz tartozó Gram–Schmidt-együtthatók.

2) Ismételjük felváltva az alábbi két lépést, amíg lehetséges (ha egyik feltétele sem áll fenn, ugorjunk ki a ciklusból).

I. Ha a jelenlegi bázis nem gyengén redukált, akkor a 2.6. algoritmus segítségével alakítsuk gyengén redukálttá.

II. Ha pedig a jelenlegi bázis a Lovász-redukáltság feltételének második felét nem teljesíti, azaz $\|b_i(i)\|^2 \not\leq \frac{4}{3}\|b_{i+1}(i)\|^2$ *valamilyen* i*-re, akkor (bármelyik, de mondjuk a legkisebb ilyen* i*-re) az* i*-edik és* $i+1$*-edik báziselemet cseréljük fel.*

3) Adjuk vissza (b_i) *bázist.*

Az algoritmus során, bár többször megváltoztatjuk $b_1, \ldots, b_n$ bázist, végig fenntartjuk, hogy a rendszer az $\mathcal{L}$ rácsnak bázisa legyen. Ilyenkor az ortogonalizáltak és a Gram–Schmidt-együtthatók is megváltoznak, ezek újraszámolását a változtatásokba beleértjük. Így b_i^* vektorokat és μ_{ij} együtthatókat úgy tekintjük, hogy mindig az algoritmus által aktuálisan fenntartott bázishoz tartoznak.
A következő fogalomra és a hozzátartozó két állításra az algoritmus futásidejének becsléséhez lesz szükségünk.
Legyen $\mathcal{L}(B)$ egy $B = (b_1, \ldots, b_n)$ bázis által generált rács, $B^* = (b_1^*, \ldots, b_n^*)$ a báziselemek Gram–Schmidt-ortogonalizáltjai. Legyenek továbbá $B_i = (b_1, \ldots, b_i)$ és $B_i^* = (b_1, \ldots, b_i^*)$ az első i báziselemre vonatkozó részrendszerek. Ezután definiáljuk a következő ún. **potenciált**:

$$\Phi(B) = \prod_{i=1}^{n} d_i, \tag{P1}$$

ahol

$$d_i = \prod_{j=1}^{i} \|b_j^*\|^2. \tag{P2}$$

Behelyettesítve és átrendezve világos:

$$\Phi(B) = \prod_{i=1}^{n}\prod_{j=1}^{i} \|b_j^*\|^2 = \prod_{k=1}^{n} \left(\|b_k^*\|^{n-k}\right)^2. \tag{P*}$$

$\Phi(B)$ potenciált az ortogonalizáltak helyett magukkal (b_i) báziselemekkel is kifejezhetjük, ezt mondja ki a következő lemma.

2.11. Lemma.

$$d_i = \det(B_i^T B_i), \tag{P3}$$

ezért d_i *és így maga* $\Phi(B)$ *is felírható* B *elemei segítségével.*

Bizonyítás: Ha a Gram–Schmidt-ortogonalizáltakra és együtthatóikra vonatkozó (GS1) és (GS2) egyenleteket mátrixalakban írjuk fel:

$$B_i = B_i^* R_i,$$

ahol R_i mátrix tartalmazza μ_{ij} együtthatókat, akkor R_i alsóháromszög-mátrix, a főátlóban 1-esekkel, így determinánsa is 1, valamint a transzponáltjáé is. Ezért:

$$\det(B_i^T B_i) = \det((B_i^* R_i)^T B_i^* R_i) = \det(R_i^T (B_i^*)^T B_i^* R_i) = \det((B_i^*)^T B_i^*) =$$
$$= \det \begin{pmatrix} \|b_1^*\|^2 & 0 & \dots & 0 \\ 0 & \|b_2^*\|^2 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & 0 & \|b_i^*\|^2 \end{pmatrix} = \prod_{j=1}^{i} \|b_j^*\|^2 = d_i,$$

ahol felhasználtuk, hogy B_i^* oszlopai merőlegesek egymásra. ∎

Ezzel a potenciálfüggvény egy másik fontos tulajdonságát is megkapjuk, amely az eredeti, B^*-os felírásból egyáltalán nem látszott.

2.12. Következmény. $\Phi(B) > 0$*, továbbá ha* B *elemei egészértékű vektorok, akkor* $\Phi(B)$ *is egészértékű.*

Bizonyítás: Tudjuk, hogy $\Phi(B) = \prod_{i=1}^{n} d_i$. Másrészt d_i 2.11. lemmabeli alakjából látszik, hogy egészértékű, ha b_j bázisvektorok azok. $\Phi(B) > 0$ egyszerű következménye annak, hogy mivel B_i elemei lineárisan függetlenek voltak, semelyik ortogonalizált hossza sem lehet 0. ∎

2.13. Tétel. *A 2.10. (LLL) algoritmus polinomiális időn belül véget ér és Lovász-redukált bázist ad vissza.*

Bizonyítás: Mint a 2.9. tétel bizonyításában, itt is feltehető, hogy a kiindulási bázis egész vektorokból áll. Vegyük észre, hogy ezen az algoritmus nem változtat: egész vektorokból kiindulva a vektorok egészértékűségét a lépések fenntartják.
Előbb belátjuk, hogy az algoritmus véges, sőt polinomiális, utána, hogy Lovász-redukált bázist ad végeredményül.
Tekintsük az algoritmus során az aktuális bázishoz rendelt $\Phi(B)$ potenciált. Belátjuk, hogy ez a mennyiség alulról és felülről is korlátos, valamint, hogy az algoritmus futása során szigorúan csökken. Mivel ezen felül egészértékű is, innen a végesség már azonnal következik. Végül egy polinomiális becslést is megfogalmazunk a futásidőre.
A potenciál eredeti (P1)+(P2) alakját felhasználva látjuk, hogy az I. típusú lépés során nem változik meg, mivel a 2.6. algoritmus a (b_i^*) ortogonalizáltakat helyben

hagyja. Belátjuk, hogy a II. lépés minden egyes végrehajtása során pedig szigorúan csökken. Legyen tehát i olyan index, melyre b_i és b_{i+1} sérti (LR)-t: $\|b_i(i)\|^2 > \frac{4}{3}\|b_{i+1}(i)\|^2$ és ezért a II. lépés végrehajtása során felcseréljük őket. Legyen az új (sorrendű) bázis $(c_i)_{i=1,\dots,n}$. Tehát $c_i = b_{i+1}$, $c_{i+1} = b_i$, a többi báziselemet pedig a II. lépés változatlanul hagyja: $c_j = b_j$ $(j \neq i, i+1)$.
Ekkor az (LR) egyenlőtlenség megsértése éppen azt jelenti, hogy

$$\|b_i^*\|^2 > \frac{4}{3}\|c_i^*\|^2$$

(lásd a 2.8. definíció utáni megjegyzést). A felcserélés során d_i-ben $\|b_i^*\|$ helyére $\|c_i^*\|$ kerül, a többi tényező pedig változatlan marad, mert $j < i$, ill. $j > i+1$ esetén $\operatorname{span}(b_1, \dots, b_j) = \operatorname{span}(c_1, \dots, c_j)$, így tehát:

$$\frac{d_i'}{d_i} = \frac{\prod_{j=1}^{i} \|c_i^*\|^2}{\prod_{j=1}^{i} \|b_i^*\|^2} = \frac{\|b_1^*\|^2 \cdots \|b_{i-1}^*\|^2 \cdot \|c_i^*\|^2}{\|b_1^*\|^2 \cdots \|b_{i-1}^*\|^2 \cdot \|b_i^*\|^2} = \frac{\|c_i^*\|^2}{\|b_i^*\|^2} < \frac{3}{4}$$

azaz az adott i-re d_i legfeljebb $\frac{3}{4}$-szeresére változik.
Másrészt d_n-t (P2) alakban felírva, ezt az 1.13. állítással összevetve látjuk, hogy

$$d_n = \prod_{j=1}^{n} \|b_j^*\|^2 = \det(\mathcal{L}) = \prod_{j=1}^{n} \|c_j^*\|^2 = d_n',$$

innen $c_j^* = b_j^*$ $(j \neq i, i+1)$ újbóli felhasználásával:

$$\|b_i^*\|\|b_{i+1}^*\| = \|c_i^*\|\|c_{i+1}^*\|,$$

ezért d_{i+1} nem változik meg, sőt, d_i-n kívül minden más is változatlan marad: $d_j' = d_j$ $(j \neq i)$.
Kaptuk, hogy $\Phi(B) = \prod_{k=1}^{n} d_k$ kifejezésben d_i az egyetlen, ami megváltozott, méghozzá kevesebb, mint $\frac{3}{4}$-részére, tehát $\Phi(B)$ értéke is kevesebb, mint $\frac{3}{4}$-szeresére változott.
Az ortogonalizáltak projekciók, így nem lehetnek hosszabbak az eredeti vektoroknál. Ezért felhasználva, hogy Φ potenciál az algoritmus során monoton csökken, a kiindulási bázist $A = (a_1, \dots, a_n)$-nel jelölve, egy felső korlátként az alábbit kapjuk:

$$\Phi(B) \leq \Phi(A) = \prod_{i=1}^{n} \left(\|a_i^*\|^{n-i}\right)^2 \leq \prod_{i=1}^{n} \left(\|a_i\|^{n-i}\right)^2 \leq \left(\max_i \|a_i\|\right)^{n(n+1)}$$

(itt a potenciál (P*) egyszeres produktum alakját használtuk).
Egy alsó korlát azonnal látszik a 2.12. következményből: mivel $\Phi(B)$ egészértékű és pozitív, azonnal kapjuk, hogy

$$\Phi(B) \geq 1.$$

Mindezekből (az egészértékűség figyelembe vételével) a végesség azonnal látszik, azonban felhasználva a látottakat, polinomiális becslés is adható az algoritmus futási idejére. Tegyük fel ugyanis, hogy az algoritmus p ciklust hajt végre, mielőtt leáll és végeredményként a $B = (b_1, \ldots, b_n)$ bázist adja (a kezdeti bázist most is jelöljük $A = (a_1, \ldots, a_n)$-nel). Ekkor:

$$1 \leq \Phi(B) \leq \left(\frac{3}{4}\right)^p \Phi(A) \leq \left(\frac{3}{4}\right)^p \left(\max_i \|a_i\|\right)^{n(n+1)},$$

ennek logaritmusát véve

$$\begin{aligned}
0 &\leq p \log\left(\frac{3}{4}\right) + n(n+1) \log\left(\max_i \|a_i\|\right) \\
-p \log\left(\frac{3}{4}\right) &\leq n(n+1) \log\left(\max_i \|a_i\|\right) \\
p \log\left(\frac{4}{3}\right) &\leq n(n+1) \max_i \left(\log(\|a_i\|)\right) \\
p &\leq \frac{1}{\log(4/3)} n(n+1) \max_i \left(\log(\|a_i\|)\right) \leq \\
&\leq 3n(n+1) \max_i \left(\log(\|a_i\|)\right).
\end{aligned}$$

Tehát a végrehajtott ciklusok p száma legfeljebb négyzetes a dimenziószámban. Egy ciklus mindkét (I. és II.) lépése polinomiálisan sok aritmetikai műveletből áll, így az algoritmus által összességében végrehajtott aritmetikai műveletek száma is polinomiális.

Hátravan még annak megmutatása, hogy amikor az algoritmus polinomiális időn belül megáll, a végeredményként kapott bázis Lovász-redukált.

Mivel az algoritmus fő ciklusának kiugrási feltételei éppen a Lovász-redukáltság definíciójában foglaltak, az nyilvánvalóan teljesül, hogy ha az algoritmus véges időn belül tényleg lefut, akkor Lovász-redukált bázist ad vissza, másrészt a véges futásidőt már beláttuk, tehát a végeredmény tényleg Lovász-redukált bázis. ∎

Megjegyzések.

1) A végeredmény Lovász-redukáltságával kapcsolatban megjegyezzük, hogy önmagában az, hogy a ciklus kiugrási feltétele az, hogy a bázis Lovász-redukált legyen, még nem lenne elég. Az problémát okozhatna, hogy a két fő lépés (I. és II.) kölcsönösen elrontják egymás eredményét, felváltva lehetetlenné téve a ciklusból való kiugrást, egyszer az egyik, máskor a másik feltétel miatt, így az algoritmust ciklizálásra kényszerítve. Pontosan ezt a lehetőséget zárja azonban ki amit beláttunk: a potenciál a II. lépés minden egyes végrehajtásakor szigorúan csökken, az I. lépés során pedig változatlan marad. Ezért, bár a fő lépések

általában könnyen elronthatják egymás eredményét, véges időn belül elérkezik az állapot, hogy a potenciál eléri a minimumát, így további II. lépés lefuttatása lehetetlen (és szükségtelen), esetleg az I. lépés (egyszeri) végrehajtása még hátravan, de utána az algoritmus megáll.

2) A Lovász-redukált definíciójában szereplő $\frac{4}{3}$ konstans értékét tetszőleges $1+\varepsilon$-ra változtathatnánk az $\varepsilon \in (0, \frac{1}{2})$ intervallumból, a tétel továbbra is érvényben maradna, az algoritmus működne. Továbbá $1+\varepsilon$ megfelelő választásával a 2.9. tételben az összes $2^{\frac{n-1}{4}}$-ben a 2-es alapot kicserélhetnénk bármely $c > 2/\sqrt{3}$-ra, ezzel élesebb becslést adva (és természetesen a futásidőt is megnövelve).

3) A gyenge redukáltság feltételéből csak $|\mu_{i+1,i}| \leq \frac{1}{2}$-et használtuk ki, a további gyenge redukáltsági feltételek ahhoz kellenek, hogy az algoritmus során előforduló számok ne nőjenek túl nagyra, ezzel biztosítva az input bitméretében és nem csak a dimenzióban való polinomialitást.

4) A konkrét futásidő-korlátról belátható, hogy $O(n^6 \log^3 (\max_i \|a_i\|^2))$, ahol (a_i) a kezdeti bázis. Ez a korlát azonban elég pesszimista, a gyakorlatban sokszor ennél sokkal gyorsabb lefutást tapasztaltak. [21, 17]

3. Bonyolultságelméleti vonatkozások

Mind az SVP, mind a CVP feladat bonyolultsága sokat kutatott kérdés. Ezek eredeti megfogalmazásukban optimalizációs feladatok, de könnyen definiálható hozzájuk eldöntési feladat.

> Shortest Vector Problem (eldöntési változat)
> Feladatpéldány: adott egy $B = (b_1, \ldots, b_n)$ bázis $\mathbb{Q}^n$-ben, egy $K \in \mathbb{Q}$ szám és egy norma $\mathbb{Q}^n$-ben.
> Kérdés: A B bázis által generált $\mathcal{L}(B)$ rácsban van-e olyan vektor, melynek normája kisebb, mint K.

> Closest Vector Problem (eldöntési változat)
> Feladatpéldány: adott egy $B = (b_1, \ldots, b_n)$ bázis $\mathbb{Q}^n$-ben, egy $v \in \mathbb{Q}^n$ vektor, egy $K \in \mathbb{Q}$ szám és egy norma $\mathbb{Q}^n$-ben.
> Kérdés: A B bázis által generált $\mathcal{L}(B)$ rácsban van-e olyan w vektor, melyre $\|v - w\|$ kisebb, mint K.

A CVP feladat NP-nehéz voltát van Emde Boas igazolta bármely normában [45], ugyanebben a cikkben kitért az SVP feladatra is, azt látta be, hogy az ℓ_∞-normában NP-nehéz és megfogalmazta a sejtést, hogy ℓ_2-normában is NP-nehéz.[1] Ajtai Miklós 1998-ban bebizonyította, hogy az SVP ℓ_2 normában randomizált visszavezetésre vonatkozóan NP-nehéz, azaz létezik randomizált Turing-gép, amely polinomiális időben bármely NP-beli problémát visszavezet az SVP feladat megoldására [3]. Nyitott kérdés viszont, hogy az SVP feladat valamely nem ℓ_∞ normában a szokásos értelemben (determinisztikus visszavezetésre) NP-nehéz-e. Továbbá megjegyezük, hogy az SVP feladat α-approximációja $\alpha > \sqrt{n/\log n}$ esetén nem NP-nehéz (ahol n szokás szerint a rács dimenziója), hacsak a polinomiális hierarchia össze nem omlik [33, 15, 30, 29]. Mindezek alapján az SVP probléma pontos megoldására polinomiális idejű algoritmus nem várható (hacsak $P \neq NP$). A pontos megoldásra a jelenleg ismert leggyorsabb algoritmus exponenciális, $2^{O(n)}$ futásidejű [38]. Mindezek fényében kitüntetett szerepe van az LLL algoritmusnak [21], mely az első és leghíresebb ismert algoritmus, amely polinomiális futásidejű és az SVP problémát $2^{O(n)}$ hibával

[1] P. van Emde Boas eredeti, 1981-es cikkében a jelenleg megszokottól kissé eltérő nevezéktant használt a felmerülő problémákra. Ami nála Closest Vector Problem, az a mi szóhasználatunkban SVP ℓ_∞ normában; továbbá ő Nearest Vector Problem-ként hivatkozik arra, amire mi CVP-ként. A további nevezéktanbeli különbségeket is jelezni fogjuk, külön lábjegyzetekben.

közelíti. Az elmúlt évtizedek kutatásai a témában a következő sejtés megfogalmazásához vezettek, amely az LLL algoritmus jelentőségét még inkább kiemeli.

3.1. Sejtés. *Nem létezik polinomiális idejű algoritmus, amely a rácsproblémákat polinomiális hibával közelíti.*

Az alábbiakban CVP probléma NP-nehéz voltát és SVP probléma ℓ_∞ normában (a továbbiakban SVP$^\infty$) NP-nehéz voltát látjuk be Peter van Emde Boas eredeti cikke alapján [45].

3.1. Három NP-teljes probléma

A CVP probléma NP-teljességének megmutatásához az Összeg Kettébontása[2] problémát, amely egyike Karp jól ismert 21 NP-teljes problémájának [20] polinomiálisan visszavezetjük az SVP$^\infty$ problémára, a Korlátos Homogén Lineáris Egyenlet (KHLE) probléma közbeiktatásával. Hasonlóan, a CVP probléma NP-teljességéhez belátjuk, hogy az Összeg Kettébontása probléma polinomiálisan visszavezethető CVP-re, itt az Összeg Gyenge Kettébontása[3] köztes probléma segítségével. Az említett problémákat pontosan definiáljuk majd használat előtt. A bizonyítás ötlete a következő alapvető technikán alapul, amellyel kombinatorikai problémák (mint például a halmazfedés) visszavezethetők speciálisan megkonstruált számokra vonatkozó egyenlőségek vizsgálatára. A technikát a Részletösszeg probléma[4] NP-teljességének bizonyításán keresztül mutatjuk be.

3.2. Tétel. *A* Részletösszeg *probléma NP-teljes.*

Bizonyítás: Pontosabban azt látjuk be, hogy az NP-teljes Partíció probléma[5] polinomiálisan visszavezethető a Részletösszeg problémára és hogy Részletösszeg NP-beli. A Partíció probléma NP-teljességét illetően lásd [24, 4.5.2. tétel]. Mindenekelőtt pontosan definiáljuk a problémákat.

Partíció probléma
Feladatpéldány: adott A alaphalmaz és részhalmazainak egy $\mathcal{S} \subseteq \mathcal{P}(A)$ rendszere.
Kérdés: Létezik-e $\mathcal{Z} \subseteq \mathcal{S}$, melyre $\dot{\bigcup_{Z \in \mathcal{Z}}} Z = A$, azaz $\mathcal{Z}$ elemei A-nak éppen egy partícióját adják.

[2] van Emde Boas cikkében (és az angol szakirodalomban) Partition
[3] van Emde Boas cikkében Weak Partition
[4] van Emde Boas cikkében Knapsack problem
[5] van Emde Boas cikkében Exact Cover

RÉSZLETÖSSZEG probléma
Feladatpéldány: adott egy $X = (x_1, \ldots, x_n) \in \mathbb{Z}^n$ vektor és $k \in \mathbb{Z}$ szám.
Kérdés: Létezik-e $I \subseteq \{1, \ldots, n\}$, melyre

$$\sum_{i \in I} x_i = k.$$

Kiindulunk a PARTÍCIÓ probléma egy példányából és polinomiális időben megkonstruáljuk belőle a RÉSZLETÖSSZEG probléma egy példányát. Legyen tehát $A = \{a_1, \ldots, a_n\}$, $\mathcal{S} = \{S_1, \ldots, S_m\}$ és $(X, \mathcal{S})$ a PARTÍCIÓ probléma egy példánya. Legyen $d := m + 1$, minden egyes S_j halmazhoz $(j = 1, \ldots, m)$ rendelünk egy s_j egész számot, méghozzá azt, amelynek alakja d alapú számrendszerben:

$$s_j = (s_{j1} \ldots s_{jn})_{\textcircled{d}}$$

ahol

$$s_{ji} = \begin{cases} 1, & \text{ha } a_i \in S_j \\ 0, & \text{egyébként} \end{cases} \qquad (i = 1, \ldots, n, \;\; j = 1, \ldots, m).$$

Továbbá definiáljuk $k \in \mathbb{Z}$ számot a következőképpen:

$$k = (\overbrace{11 \ldots 1}^{n \text{ db}})_{\textcircled{d}}.$$

Ekkor tehát minden j-re $s_j = \sum_{i=1}^{n} s_{ji} d^{i-1}$ és $k = \sum_{i=1}^{n} d^{i-1}$ és figyeljük meg, hogy mivel bármely $a_i \in A$ elemet legfeljebb m darab $S_j \subseteq \mathcal{S}$ halmaz tartalmazhat, $d = m + 1$ választása miatt az s_j számok bármilyen összegzése esetén összeadási átvitel a számjegyek között nem fordulhat elő. Éppen ezért könnyen látható, hogy pontosan akkor létezik olyan $J \subseteq \{1, \ldots, m\}$, melyre

$$\sum_{j \in J} s_j = \sum_{j \in J} \sum_{i=1}^{n} s_{ji} d^{i-1} = \sum_{i=1}^{n} d^{i-1} = k,$$

azaz $X := (s_1, \ldots, s_m)$ választással a RÉSZLETÖSSZEG probléma (X, k) példányának pontosan akkor létezik megoldása, ha az eredeti PARTÍCIÓ problémának létezett megoldása.

A RÉSZLETÖSSZEG probléma NP-beliségéhez csak azt kell látnunk, hogy az I halmaz, mely könnyen láthatóan megfelel tanúnak az NP-beliségre, megadható olyan formában, amelynek mérete input méretének polinomjával korlátozható. Valójában az input elemszámának logaritmusával korlátozható, hiszen az eredeti vektor minden elemének megfeleltethetünk egy bitet aszerint, hogy I-nek eleme-e vagy sem. A

leellenőrzés ideje nyilván polinomiális: összeadjuk a megfelelő elemeket és ellenőrizzük, hogy az összeg egyenlő-e k-val. ∎

Hasonló, de bonyolultabb gondolatmeneten alapszik a következő két tétel is, melyeket közvetlenül az SVP, ill. CVP probléma NP-teljességének bizonyításához fogunk felhasználni. Először is definiálunk három további problémát, melyekre hivatkozni fogunk.

Összeg Kettébontása probléma
Feladatpéldány: Egy $(a_1, \ldots, a_m) \in \mathbb{Z}^m$ vektor.
Kérdés: Létezik-e $I \subseteq \{1, \ldots, m\}$, melyre

$$\sum_{i \in I} a_i = \sum_{j \notin I} a_j.$$

Ezt a problémát fogjuk tehát visszavezetni külön-külön az alábbi két problémára, melyeket pedig következő lépcsőként majd az SVP, ill. CVP problémára vezetünk vissza.

Korlátos Homogén Lineáris Egyenlet probléma
Feladatpéldány: Egy $(a_1, \ldots, a_m) \in \mathbb{Z}^m$ vektor, $K \in \mathbb{Z}$ pozitív szám.
Kérdés: Létezik-e a következő egyenletnek:

$$\sum_{i=1}^{m} x_i a_i = 0$$

nemtriviális, korlátos, egész x megoldása, azaz olyan megoldása, melyben x nem a nulla vektor és minden i-re $|x_i| \leq K$, $x_i \in \mathbb{Z}$.

Összeg Gyenge Kettébontása probléma
Feladatpéldány: Egy $(a_1, \ldots, a_m) \in \mathbb{Z}^m$ vektor.
Kérdés: Létezik-e olyan $\mathbf{0} \neq x \in \{-1, 0, +1\}^m$ vektor, melyre

$$\sum_{i=1}^{n} x_i a_i = 0,$$

azaz az egyenletnek van-e nemtriviális megoldása (tehát olyan megoldása, melyben nem minden x_i egyenlő 0-val).

Az Összeg Kettébontása probléma definíciójában szereplő egyenlőséget

$$\sum_{i \in I} a_i - \sum_{j \notin I} a_j = 0$$

alakba írva azonnal láthatjuk, hogy az majdnem megegyezik az Összeg Gyenge Kettébontása problémával, annyi különbséggel, hogy a gyenge esetben megengedjük az $x_i = 0$ koordinátákat, azaz bizonyos számok kihagyását az összegből. Továbbá vegyük észre, hogy az Összeg Gyenge Kettébontása probléma a Korlátos Homogén Lineáris Egyenlet problémának speciális esete $K = 1$-re.
Az alábbi lemma formalizálja és kissé általánosítja a 3.2. tételben szereplő összeadás-átvitelre vonatkozó gondolatot. A következő tételekben többször is támaszkodunk majd rá.

3.3. Lemma. *Legyenek a_i, b_i, c_i $(i = 1, \ldots, n)$ egész számok a következő alakúak:*

$$a_i = b_i + M \cdot c_i$$

ahol M is egész szám. Legyen továbbá K olyan egész, melyre $M > K \cdot \sum_{i=1}^{n} |b_i|$ fennáll. Ekkor az alábbi két rendszer megoldhatósága az egészek fölött ekvivalens:

$$\sum_{i=1}^{n} x_i a_i = 0, \qquad |x_i| \leq K, \tag{1}$$

illetve

$$\sum_{i=1}^{n} x_i b_i = 0, \qquad \sum_{i=1}^{n} x_i c_i = 0, \qquad |x_i| \leq K. \tag{2}$$

Bizonyítás: Osszuk el (1) egyenlet mindkét oldalát M-mel és vegyük észre, hogy a feltételek miatt (2) éppen a kapott egyenlet egész- és törtrészéről szól. Ha $(x_i)_{i=1,\ldots,n}$ megoldása (1) rendszernek, akkor az egész- és törtrészeknek nyilván külön-külön is egyenlőnek kell lenniük 0-val, ezért (x_i) megoldása (2)-nek is. Visszafelé nyilvánvaló.
■

3.4. Tétel. Összeg Gyenge Kettébontása *probléma NP-teljes.*

Bizonyítás: Vegyük az Összeg Kettébontása probléma egy példányát és polinomiális időben konstruáljuk meg belőle az Összeg Gyenge Kettébontása probléma egy példányát. Tegyük fel tehát, hogy adott az Összeg Kettébontása probléma egy $(a_1, \ldots, a_m)$ példánya. Legyen továbbá $d > 4$ egész szám és

$$M := 2 \left(\sum_{i=1}^{m} |a_i| \right) + 1.$$

Minden i-re bevezetjük a $b_{i1}, \ldots, b_{i5}$ számokat:

$$\begin{array}{rcl} b_{i1} &=& a_i + M\left(d^{4i-4} + d^{4i-3} + 0 + d^{4i-1} + 0 \right) \\ b_{i2} &=& 0 + M\left(0 + d^{4i-3} + 0 + 0 + d^{4i} \right) \\ b_{i3} &=& 0 + M\left(d^{4i-4} + 0 + d^{4i-2} + 0 + 0 \right) \\ b_{i4} &=& a_i + M\left(0 + 0 + d^{4i-2} + d^{4i-1} + d^{4i} \right) \\ b_{i5} &=& 0 + M\left(0 + 0 + 0 + d^{4i-1} + 0 \right) \end{array} \tag{A1}$$

azzal a kivétellel, hogy b_{m2}-ben és b_{m4}-ben az $M \cdot d^{4m}$ tag helyett $M \cdot 1$ álljon, később részletezendő okból. Könnyen látható, hogy M választása miatt minden b_{ij} szám két olyan tag összege, melyben az elsők még minden i, j-re összeadva is szigorúan kisebbek M-nél, a második tagok pedig M egész számú többszörösei, így a 3.3. lemma feltételei – $K := 1$ választással – fennállnak. (M definíciójában a 2-es szorzó nem $K = 2$-re utal, mint ahogy formailag a 3.3. lemmában szereplő képlet alapján tűnhet, hanem azért van ott, mert bár itt $K = 1$, viszont egy adott i-re a fenti b_{ij} számokban 2-szer szerepel a_i, így ha összeadjuk az összes b_{ij}-t, mint ahogy (A2)-ben szerepel, akkor $M > \sum 2|a_i|$ a megfelelő korlát, ami el tudja választani az M-szorzós tagokat az anélküliektől.) A lemma alkalmazásával kapjuk, hogy a

$$\sum_{i=1}^{m} \sum_{j=1}^{5} x_{ij} b_{ij} = 0, \qquad x_{ij} \in \{-1, 0, +1\} \tag{A2}$$

rendszer megoldhatósága ekvivalens az alábbi rendszerével:

$$\begin{array}{ll} (i) & \displaystyle\sum_{i=1}^{m} (x_{i1} + x_{i4}) a_i = 0 \\ (ii) & (x_{11} + x_{13} + x_{m2} + x_{m4}) \cdot 1 + \\ & \displaystyle\sum_{i=2}^{m} (x_{i1} + x_{i3} + x_{i-1,2} + x_{i-1,4}) \cdot d^{4i-4} + \\ & \displaystyle\sum_{i=1}^{m} (x_{i1} + x_{i2}) \cdot d^{4i-3} + \\ & \displaystyle\sum_{i=1}^{m} (x_{i3} + x_{i4}) \cdot d^{4i-2} + \\ & \displaystyle\sum_{i=1}^{m} (x_{i1} + x_{i4} + x_{i5}) \cdot d^{4i-1} = 0 \\ & x_{ij} \in \{-1, 0, +1\} \end{array} \tag{A3}$$

ahol M már nem szerepel és a második egyenlet tagjait csoportosítottuk d hatványai szerint. Vegyük észre, hogy a korábbi (A1) rendszerben egy adott b_{ij}-re az M többszörösét adó tagban M együtthatói egy szám d alapú számrendszerbeli ábrázolásának számjegyei (növekvő helyiérték szerint feltüntetve), melyek a kialakítás szerint mind 1-esek vagy 0-k. Továbbá x_{ij} számok abszolútértéke is legfeljebb 1. Másrészt amikor (A3)-ban összegzünk, d-hatványonként legfeljebb 4 tagot adunk össze, ezért

megengedett (x_{ij}) megoldások esetén $d > 4$ miatt nem történhet összeadási átvitel. (Ez magyarázza $d > 4$ feltételt, lásd még a 3.2. tétel bizonyításában d választását.) De ugyanígy hivatkozhattunk volna a 3.3. lemma ismételt alkalmazására is $M' := d$ és $K' := 1$ választásával.

Következésképpen az összeadandók itt is szeparálhatók és (A2) rendszer megoldhatósága végül is ekvivalens az alábbi, $4m+1$ egyenletből álló rendszer megoldhatóságával, melyből tehát az elsőben az eredeti Összeg Kettébontása feladatpéldány paraméterei szerepelnek, a többi $4m$ darab egyenletben pedig csak az $M \cdot d^j$ alakú tagok, d-hatványonként csoportosítva. Itt látszik igazán jól, hogy az a_m-hez tartozó b_{m2}, ill. b_{m4} számokban M együtthatójának d^{4m}-ről 1-re változtatásával az egyenletek egyfajta „körbe zárásának" feladatát (szándékolt összeadási átvitel kényszerítésével a szabadsági fok csökkentését) látjuk el.

$$\begin{array}{lcll}
(0) & \sum_{i=1}^{m}(x_{i1}+x_{i4})a_i & = 0 & \\
(i1) & x_{11}+x_{13}+x_{m2}+x_{m4} & = 0 & (i=1 \text{ eset}) \\
 & x_{i1}+x_{i3}+x_{i-1,2}+x_{i-1,4} & = 0 & i=2,3,\ldots,m \\
(i2) & x_{i1}+x_{i2} & = 0 & i=1,2,\ldots,m \\
(i3) & x_{i3}+x_{i4} & = 0 & i=1,2,\ldots,m \\
(i4) & x_{i1}+x_{i4}+x_{i5} & = 0 & i=1,2,\ldots,m \\
 & x_{ij} \in \{-1,0,+1\} & &
\end{array} \tag{A4}$$

Itt (i2) és (i3) egyenletekből $x_{i2} = -x_{i1}$ és $x_{i4} = -x_{i3}$, ezt összevetve (i1)-gyel kapjuk, hogy egy tetszőleges $(x_{ij})_{i,j\in[m]\times[5]}$ megoldásban $x_{i1} + x_{i3}$ értéke i-től független, nevezzük ezt az értéket az (x_{ij}) megoldás súlyának. Mivel a -1-gyel szorzás egy megoldást nem ront el, feltehetjük, hogy a megoldás súlya nemnegatív.

Továbbá megtehetjük, hogy azon megoldások vizsgálatára szorítkozunk, amelyekben $x_{13} = 0$, ugyanis végeredményben az Összeg Gyenge Kettébontása probléma egy példányát szeretnénk kapni; ez is az Összeg Gyenge Kettébontása probléma egy példánya, csak eggyel kevesebb számra (b_{13} kihagyásával).

Ezekkel a feltételekkel tehát egy megoldás súlya megegyezik x_{11} együttható értékével és ezért feltehető, hogy 0 vagy 1 (ugyanis $x_{ij} \in \{-1,0,1\}$ és feltettük, hogy nemnegatív).

Tegyük fel, hogy az (x_{ij}) megoldás súlya 0. Így tehát minden i-re: $x_{i1} + x_{i3} = 0$, továbbá (i4) és (i3) alapján

$$x_{i5} = -x_{i1} - x_{i4} = x_{i3} - x_{i1},$$

de $x_{i3} - x_{i1}$ paritása megegyezik $x_{i3} + x_{i1} = 0$ paritásával:

$$x_{i5} = x_{i3} - x_{i1} \equiv x_{i3} + x_{i1} = 0 \mod 2$$

összevetve $x_{i5} \in \{-1, 0, 1\}$ lehetséges értékeivel kapjuk, hogy $x_{i5} = 0$. Így x_{i1} és x_{i3} egyenlők és összegük 0, tehát maguk is egyenlők 0-val. (i2) és (i3) egyenletekből már korábban láttuk, hogy $x_{i2} = -x_{i1}$ és $x_{i4} = -x_{i3}$, így tehát ezek is 0-k. Az érvelés minden i-re elmondható, kaptuk tehát, hogy a megoldás az azonosan 0 megoldás, viszont azt már kizártuk az elfogadható megoldások köréből.
Marad tehát a másik eset, hogy az (x_{ij}) megoldás súlya 1. Hasonló érveléssel kapjuk, hogy ekkor x_{i5} páratlan, tehát $+1$ vagy -1. Az első esetben a fentiből

$$x_{i1} = 0, \quad x_{i3} = 1, \text{ és így } x_{i4} = 0,$$

a másodikban

$$x_{i1} = 1, \quad x_{i3} = 0, \text{ és így } x_{i4} = -1.$$

E megfigyeléseket felhasználva látjuk, hogy a (0) egyenletben szereplő $x_{i1} + x_{i4}$ együttható értéke $+1$ vagy -1, tehát a fenti rendszerben (0) megoldhatóságából, és így a teljes $5m-1$ számból álló Összeg Gyenge Kettébontása feladat megoldhatóságából következik $(a_1, \ldots, a_m)$ Összeg Kettébontása feladatpéldány megoldhatósága.
Megfordítva, $(a_1, \ldots, a_m)$ Összeg Kettébontása feladathoz tartozó megoldásból könnyen látható módon, csak a definiáló egyenletek követésével előállítható $(b_{11}, b_{12}, b_{14}, \ldots, b_{m5})$ Összeg Gyenge Kettébontása feladatpéldány egy megoldása. Az is világosan látszik, hogy ez az előállítás polinomiális időben elvégezhető. Az NP-beliség igazolása hasonló módon történhet, mint a 3.2. esetében (itt két bitvektorra van szükség).
Megjegyezzük, hogy a konstrukcióból az is látszik, hogy az Összeg Gyenge Kettébontása feladat egy megoldásában pontosan $3m$ darab nemnulla szám lesz. (x_{i5} nemnulla, x_{i1} és x_{i4} közül pontosan az egyik 0, továbbá $x_{i1} \neq 0 \Leftrightarrow x_{i2} \neq 0$ és $x_{i3} \neq 0 \Leftrightarrow x_{i4} \neq 0$.) ∎

3.5. Tétel. Korlátos Homogén Lineáris Egyenlet *probléma NP-teljes.*

Bizonyítás: Az előző tétel gondolatmenetének csekély módosításával belátható. Láttuk, hogy az Összeg Gyenge Kettébontása probléma $K = 1$-re vonatkozó speciális esete a Korlátos Homogén Lineáris Egyenlet problémának, ennek alapján fogjuk módosítani a konstrukciót és a 3.4. tétel bizonyítását.
Legyen tehát $K \geq 1$ tetszőleges egész. Ahhoz, hogy a 3.3. lemmát itt is alkalmazhassuk, először is d helyett $d' := Kd$ megfelelő hatványait kell venni. Ezután az $M(d')^{4i-j} = M \cdot (Kd)^{4i-j}$ alakú tagokat kicseréljük $KM \cdot (Kd)^{4i-j}$ alakúakra, kivéve b_{i5} együtthatóit, amelyeket változatlanul hagyunk. Tehát a következő számokat

definiáljuk:

$$\begin{array}{lllllllllllll}
b_{i1} & = a_i & + & KM\Big(& (Kd)^{4i-4} & + & (Kd)^{4i-3} & + & 0 & + & (Kd)^{4i-1} & + & 0 \Big) \\
b_{i2} & = 0 & + & KM\Big(& 0 & + & (Kd)^{4i-3} & + & 0 & + & 0 & + & (Kd)^{4i} \Big) \\
b_{i3} & = 0 & + & KM\Big(& (Kd)^{4i-4} & + & 0 & + & (Kd)^{4i-2} & + & 0 & + & 0 \Big) \\
b_{i4} & = a_i & + & KM\Big(& 0 & + & 0 & + & (Kd)^{4i-2} & + & (Kd)^{4i-1} & + & (Kd)^{4i} \Big) \\
b_{i5} & = 0 & + & M\Big(& 0 & + & 0 & + & 0 & + & (Kd)^{4i-1} & + & 0 \Big)
\end{array} \tag{B1}$$

azzal, hogy itt is, mint 3.4. (A1) rendszer esetében, b_{m2}-ben és b_{m4}-ben az $KM \cdot (Kd)^{4m}$ tag helyett $KM \cdot 1$ szerepeljen.

Ebből az előző tételéhez egészen hasonló levezetés segítségével egy 3.4. (A4) rendszerhez hasonlóhoz jutunk, annyi különbséggel, hogy más a változók lehetséges értékeinek halmaza: $\{x_{ij} \in \mathbb{Z} : |x_{ij}| \leq K\}$ és (i4) egyenletek helyett

$$(i4') \quad K\left(x_{i1} + x_{i4}\right) + x_{i5} \;\; = 0 \qquad (i = 1, 2, \ldots, m)$$

szerepel, azaz a következő rendszert kapjuk:

$$\begin{array}{lcll}
(0) & \sum_{i=1}^{m}(x_{i1} + x_{i4})a_i & = 0 & \\
(i1) & x_{11} + x_{13} + x_{m2} + x_{m4} & = 0 & (i = 1 \text{ eset}) \\
 & x_{i1} + x_{i3} + x_{i-1,2} + x_{i-1,4} & = 0 & i = 2, 3, \ldots, m \\
(i2) & x_{i1} + x_{i2} & = 0 & i = 1, 2, \ldots, m \\
(i3) & x_{i3} + x_{i4} & = 0 & i = 1, 2, \ldots, m \\
(i4') & K\left(x_{i1} + x_{i4}\right) + x_{i5} & = 0 & i = 1, 2, \ldots, m \\
 & |x_{ij}| \leq K, x_{ij} \in \mathbb{Z} & &
\end{array} \tag{B4}$$

A feltételek szerint $|x_{i5}| \leq K$, ezért (i4')-ből

$$|x_{i1} + x_{i4}| \leq 1$$

és mivel $x_{14} = -x_{13} = 0$ itt is feltehető (a 3.4. tételhez hasonlóan), itt is minden megoldás súlya $-1, 0$, vagy $+1$, éspedig x_{11} értéke szerint. Sőt, mivel a -1-gyel szorzás itt sem ront el egy megoldást, itt is feltehető, hogy a megoldás súlya nemnegatív.

Az egyetlen 0 súlyú megoldás itt is a triviális, csupa 0 megoldás, ugyanis egyrészt

$$x_{i1} + x_{i3} = 0,$$

másrészt

$$|x_{i1} - x_{i3}| = |x_{i1} + x_{i4}| \leq 1,$$

ezért $x_{i1} = x_{i3} = 0$, a többi változó 0 voltát pedig innen már a 3.4. tételhez hasonlóan kapjuk.

Végül az 1 súlyú megoldásra is a korábbi tételhez hasonlóan:

$$x_{i1} + x_{i4} = \pm 1,$$

ezért a (B4) rendszerben (0) megoldhatóságából, és így a teljes $5m-1$ számból álló Korlátos Homogén Lineáris Egyenlet feladat megoldhatóságából $(a_1,\ldots,a_m)$ Összeg Kettébontása feladatpéldány megoldhatósága következik. Megfordítva, $(a_1,\ldots,a_m)$ Összeg Kettébontása feladathoz tartozó megoldásból itt is polinomiális időben konstruálható a fenti $((b_{11},b_{12},b_{13},b_{15},\ldots,b_{m5}),K)$ Korlátos Homogén Lineáris Egyenlet feladatpéldány egy megoldása. Az NP-beliséghez azt kell látni, hogy egy adott megoldás megfelel polinomiális méretű és idejű tanúnak. ∎

3.2. CVP és SVP bonyolultsága

Ennyi előkészítés után már beláthatjuk a fejezet két fő eredményét, SVP^∞ és CVP NP-teljességét.

3.6. Tétel. Shortest Vector Problem ℓ_∞ *normában NP-teljes.*

Bizonyítás: A Korlátos Homogén Lineáris Egyenlet (KHLE) problémát fogjuk polinomiálisan visszavezetni SVP-re. Legyen tehát $((b_1,\ldots,b_m),K)$ a KHLE probléma egy példánya, legyenek $K' := K+1$, $K'' := 2K'\sum_{j=1}^{m}|b_j|$ és tekintsük a következő, $m+1$ elemből álló vektorrendszert:

$$U=(u_1,\ldots,u_{m+1})=\left(\begin{array}{ccc|c} & & & 0\\ & I_{m\times m} & & \vdots\\ & & & \vdots\\ & & & \vdots\\ & & & 0\\ \hline K'b_1 & K'b_2 & \ldots\ K'b_m & K'' \end{array}\right),$$

ahol $I_{m\times m}$ az $m\times m$-es identitásmátrix. Most ahhoz, hogy az SVP^∞ probléma (eldöntési változatának) egy példányához jussunk, tegyük fel, hogy w vektor az (u_i) vektorok egy nemtriviális egész lineáris kombinációja, azaz

$$x_1u_1+\ldots+x_mu_m+x_{m+1}u_{m+1}=w,\quad x_i\in\mathbb{Z},\ \exists i: x_i\neq 0$$

és tegyük fel (itt kap szerepet az ℓ_∞ norma), hogy w minden $(w_i)_{i=1,\ldots,m+1}$ koordinátájára $|w_i|\le K$. Innen az identitásmátrix miatt minden $j\le m$-re $w_j=x_j$, így $|x_j|\le K$ is fennáll. Ezenkívül w_{m+1}-re:

$$w_{m+1}=K'\cdot(x_1b_1+\ldots+x_mb_m)+K''\cdot x_{m+1},$$

de mivel $|w_{m+1}| \leq K < K' < K''$, innen

$$x_1 b_1 + \ldots + x_m b_m = 0 \quad \text{és} \quad x_{m+1} = 0.$$

Tehát kaptuk, hogy $((u_1, \ldots, u_{m+1}), K)$ SVP feladatpéldány pontosan akkor megoldható, ha az eredeti $((b_1, \ldots, b_m), K)$ KHLE feladatpéldány megoldható volt, a konstrukció pedig nyilván polinomiális idejű.
Kell még, hogy SVP$^\infty$ NP-beli, de könnyű látni, hogy maga a legrövidebb vektor megfelel polinomiális bithosszúságú tanúnak, a koordinátáit pedig sorra leellenőrizhetjük polinom időben, hogy teljesítik-e a megadott korlátot.
(Azzal kapcsolatban, hogy a látott visszavezetés ℓ_∞ norma helyett más normában miért nem alkalmazható, lásd [45, „parasitic weight zero solutions"].) ∎

3.7. Tétel. Closest Vector Problem *bármely p-normában NP-teljes.*

Bizonyítás: Az Összeg Kettébontása problémát Összeg Gyenge Kettébontása problémán keresztül polinomiálisan vissza fogjuk vezetni CVP-re. A bizonyítás során feltesszük, hogy $p = 2$, de a gondolatmenet könnyen láthatóan alkalmazható bármely $1 \leq p \leq \infty$ esetére.
Tekintsük tehát az Összeg Kettébontása probléma egy $(a_1, \ldots, a_{m'})$ számokkal megadott példányát és a 3.4. tétel segítségével konstruáljunk belőle $(b_1, \ldots, b_m)$ számokkal adott Összeg Gyenge Kettébontása feladatpéldányt, ahol $m = 5m' - 1$ és legyen (a tételben szereplő) $d > 2m$. Ha ennek a feladatnak van $(x_i)_{i=1,\ldots,m}$ nemtriviális megoldása, akkor erre a megoldásra

$$x_1 b_1 + \ldots + x_m b_m = 0$$

és $|x_i| \leq 1$ minden i-re, továbbá a megoldás a súlya megegyezik x_1-gyel, ami nem 0. Ezenkívül a nemnulla x_i-k száma pontosan $3m'$, ahogy azt a 3.4. tétel végén megjegyeztük.
Most legyen $K := m + 1$, ebből $K' = K + 1$ és $K'' = 2K' \sum_{j=1}^{m} |b_j|$ az előző tételhez hasonlóan és vegyük az előző tételben látott $m + 1$ elemű vektorrendszert: $u_1, \ldots, u_{m+1}$. Továbbá legyen még $\mathbb{Z}^{m+1} \ni v = (K'', 0, \ldots, 0)^T$ új vektor. Ezután az U rendszer első elemét v-re cserélve tekintsük az alábbi, módosított rendszert:

$$U' = (v, u_2, \ldots, u_{m+1}) = \left(\begin{array}{c|ccc|c} K'' & 0 \ldots 0 & & & 0 \\ \hline 0 & & & & 0 \\ \vdots & & I_{(m-1)\times(m-1)} & & \vdots \\ \vdots & & & & \vdots \\ 0 & & & & 0 \\ \hline 0 & K'b_2 & K'b_2 & \ldots \; K'b_m & K'' \end{array}\right), u_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \\ K'b_1 \end{pmatrix}$$

és az ezzel megfogalmazott CVP feladatpéldányt:

$$((v, u_2, \ldots, u_{m+1}), \sqrt{3m'}, v_1).$$

Ha az eredeti Összeg Kettébontása feladatpéldány megoldható volt, akkor az abból konstruált Összeg Gyenge Kettébontása feladatpéldánynak lesz olyan megoldása, melynek (esetleges negálás után) a súlya $x_1 = -1$. K'' választása (mérete) miatt a megoldásban biztosan nem szerepel, ezért a megoldást a következő alakban is felírhatjuk:

$$b_1 = x_2 b_2 + \ldots + x_m b_m,$$

éspedig pontosan $3m' - 1$ nemnulla x_i együtthatóval az egyenlet jobb oldalán. Ha ezután w vektort úgy definiáljuk, hogy ugyanígy x_i szerinti összege legyen, méghozzá az $u_2, \ldots, u_m$ vektoroknak:

$$w = x_2 u_2 + \ldots + x_m u_m,$$

akkor w utolsó koordinátájában $K'b_j$ kifejezések éppen $K'b_1$-re összegződnek, ezért $u_1 - w$ vektor utolsó koordinátája 0 lesz, a többi koordináta közül pedig pontosan $3m'$ helyen lesz ± 1, tehát $|u_1 - w| \leq \sqrt{3m'}$, azaz a CVP feladatpéldány is megoldható. Megfordítva, tegyük fel, hogy

$$w = x_1 u_1 + \ldots + x_{m+1} u_{m+1}$$

egy megoldása a CVP feladatnak, ekkor K'' mérete miatt u_1 és u_{m+1} kizárható a megoldásból, azaz $x_1 = x_{m+1} = 0$ feltehető. Innen könnyen láthatóan

$$b_1 = x_2 b_2 + \ldots + x_m b_m,$$

következik, ahol $|x_i| \leq \sqrt{3m'} < m < \frac{1}{2}d$ tehát alkalmazhatjuk a 3.3. lemmát, amiből kapjuk, hogy (x_i) egy -1 súlyú valódi megoldás, azaz az eredeti Összeg Kettébontása problémának is van megoldása.
A konstrukció szemmel láthatóan polinomiális időben elvégezhető, beláttuk tehát, hogy CVP NP-nehéz. Az NP-beliséget a 3.6. tételhez hasonlóan kell belátni. ∎

4. Alkalmazások

4.1. Szimultán diofantikus approximáció

A szimultán diofantikus approximáció feladata abban áll, hogy egy valós számokból álló, adott vektort közelítünk egy ugyanekkora méretű, racionális számokból álló vektorral, ahol az utóbbi vektor elemeinek ugyanaz a nevezője. Legyen n tetszőleges pozitív egész, $\alpha_1, \ldots, \alpha_n$ tetszőleges valós számok, $0 < \varepsilon < 1$ valós. Ekkor a jól ismert eredmény szerint léteznek $p_1, \ldots, p_n$ és q egészek, melyekre

$$|p_i - q\alpha_i| \leq \varepsilon \quad \forall 1 \leq i \leq n \qquad \text{és}$$

$$1 \leq q \leq \varepsilon^{-n}.$$

Ennél egy kicsit gyengébb feltételnek eleget tevő egész számok a bázisredukció alkalmazásával azonnal megtalálhatók.

4.1. Állítás. *Létezik olyan polinomiális algoritmus, amely adott n egész számhoz és $\alpha_1, \ldots, \alpha_n, \varepsilon$ $(0 < \varepsilon < 1)$ racionális számokhoz megad $p_1, \ldots, p_n, q$ egészeket, amelyekre:*

$$|p_i - q\alpha_i| \leq \varepsilon \quad \forall 1 \leq i \leq n \qquad \textit{és}$$

$$1 \leq q \leq 2^{n(n+1)/4}\varepsilon^{-n}.$$

Bizonyítás: Legyen $\mathcal{L}$ rács bázisa a következő $(n+1) \times (n+1)$ méretű mátrix oszlopvektoraiból álló rendszer:

$$B = \begin{pmatrix} 1 & 0 & \ldots & 0 & -\alpha_1 \\ 0 & 1 & \ldots & 0 & -\alpha_2 \\ \vdots & \vdots & \ddots & \vdots & \\ 0 & 0 & \ldots & 1 & -\alpha_n \\ 0 & 0 & \ldots & 0 & 2^{-n(n+1)/4}\varepsilon^{n+1} \end{pmatrix}.$$

Erre a rácsra alkalmazhatjuk a 2.10. algoritmust, mely polinomiális idő után megáll és Lovász-redukált bázist ad vissza, legyen ez $\overline{B} = (\overline{b}_1, \ldots, \overline{b}_n)$. Ekkor tehát 2.9. (2) alapján $|\overline{b}_1| \leq 2^{n/4}\sqrt[n+1]{2^{-n(n+1)/4}\varepsilon^{n+1}} = \varepsilon$.
Mivel $b_1 \in \mathcal{L}$, felírhatjuk az eredeti báziselemek lineáris kombinációjaként, azaz:

$$\overline{b}_1 = \begin{pmatrix} p_1 - q\alpha_1 \\ p_2 - q\alpha_2 \\ \vdots \\ p_n - q\alpha_n \\ q \cdot 2^{-n(n+1)/4}\varepsilon^{n+1} \end{pmatrix},$$

ahol $p_1, \ldots, p_n, q \in \mathbb{Z}$, melyekre:

$$|p_i - q\alpha_i| \leq \varepsilon \quad \forall 1 \leq i \leq n \qquad \text{és}$$

$$|q| \leq 2^{n(n+1)/4}\varepsilon^{-n},$$

hiszen $|\bar{b}_1| < \varepsilon$ volt. Végül $\varepsilon < 1$ és $\bar{b}_1 \neq 0$ miatt $q \neq 0$ is fennáll, $\bar{b}_1$ esetleges negálásával pedig elérhető, hogy $q > 0$ legyen. ∎

4.2. Polinomkongruenciák

Az alábbi fejezetben az LLL algoritmus egy önmagában is érdekes alkalmazását mutatjuk be, amely ezenfelül a következő fejezetben tárgyalt, RSA nyilvános kulcsú titkosítás elleni támadásban betöltött szerepe miatt is igen lényeges.
Bevezetésképpen legyen adott egy $p(x)$ egyváltozós, egész együtthatós polinom és oldjuk meg a következő, N-re vonatkozó d-edfokú kongruenciát: célunk az összes kis $g \in \mathbb{Z}$ megkeresése, melyre $|g| < K$ valamilyen K-ra és

$$p(g) = g^d + a_{d-1}g^{d-1} + \ldots + a_1 g + a_0 \equiv 0 \mod N \qquad \text{(PK)}$$

is teljesül. Itt K-t szeretnénk minél nagyobbra választani, p-ről feltesszük, hogy 1 főegyüttható és irreducibilis $\mathbb{Z}$ fölött, továbbá úgy vesszük, hogy N-nek ismeretlen a prímfaktorizációja. A feladat nehézségét nyilván a kongruencia okozza, anélkül egy közönséges polinom gyökeit keresnénk, melyre számos gyors módszer létezik. (Bairstow-módszer, Aberth-módszer [5, 1, 12], ezek iteratív, kvadratikusan konvergens módszerek, de mivel egész gyököket keresünk, azonnal leállhatunk, amint a közelítés hibája $\frac{1}{2}$-en belülre kerül.) A fenti polinomkongruencia problémáját Johan Håstad, Don Coppersmith és Nicholas Howgrave-Graham munkáján [18, 8, 17, 9] alapulva tárgyaljuk. Lásd még a témában Glenn Durfee [10] disszertációját.
Először [18] alapján tegyük fel, hogy g egy kis egész gyök a fenti értelemben és tekintsük a következő polinomokat:

$$C_1 := \{x^i : 0 \leq i < d\} \cup \{p(x)/N\}.$$

Könnyen láthatóan bármely C_1-beli polinomba, valamint bármely lineáris kombinációjukba g-t behelyettesítve egészet kapunk. Ezután tekintsük az euklideszi normával ellátott $\mathbb{Q}^{d+1}$ térben valamely $K > 0$-ra a következő B_1 bázist, valamint a bázissal

generált $\mathcal{L}(B_1)$ rácsot:

$$B_1 = \begin{pmatrix} 1 & 0 & 0 & \dots & 0 & 0 & a_0/N \\ 0 & K & 0 & \dots & 0 & 0 & a_1K/N \\ 0 & 0 & K^2 & \dots & 0 & 0 & a_2K^2/N \\ \vdots & & & & & & \\ 0 & 0 & 0 & \dots & K^{d-2} & 0 & a_{d-2}K^{d-2}/N \\ 0 & 0 & 0 & \dots & 0 & K^{d-1} & a_{d-1}K^{d-1}/N \\ 0 & 0 & 0 & \dots & 0 & 0 & 1K^d/N \end{pmatrix},$$

amiben minden oszlopvektor a fenti C_1 egy-egy $q(x)$ elemének felel meg, méghozzá

$$(x^i/K^i)_{i=0,\dots,d}$$

$\mathbb{Q}[x]$-beli polinombázis szerint felírva, azaz a mátrix i-edik sora az adott $q(x)$ polinomban az x^i-hez tartozó együttható K^i-szerese. A bázis elemei jól látható módon lineárisan függetlenek, a kapott rács $d+1$-dimenziós, a bázis determinánsa a felsőháromszög-mátrix alak miatt a főátlóbeli elemek szorzata: $N^{-1}K^{d(d+1)/2}$. Ha erre a rácsra (az adott B_1 kezdőbázissal) alkalmazzuk a 2.10. (LLL) algoritmust, akkor az a visszaadott redukált bázisban egy

$$|\mathbf{v}| < c_1(d)(\det(\mathcal{L}))^{1/(d+1)} = c_1(d)N^{-1/(d+1)}K^{d/2}$$

vektort szolgáltat, ahol $c_1(d)$ csak a d dimenziótól függő konstans. A $\mathbf{v}$ vektor ugyanazon vektortér-izomorfizmus mentén (a (x^i/K^i) bázis segítségével) nyilván megfeleltethető $v(x) = (v_dK^d)x^d + \dots + (v_1K)x + v_0$ polinomnak. Ha ezután föltesszük, hogy

$$c_1(d)(\det(\mathcal{L}))^{1/(d+1)} < \frac{1}{1+d}$$

vagy ezzel ekvivalensen

$$K \le c_1'(d)N^{\frac{2}{d(d+1)}},$$

ahol $c_1'(d)$ megint egy csak d-től függő konstans, akkor innen $\|\mathbf{v}\|_\infty \le |\mathbf{v}| < \frac{1}{d+1}$ és így $\mathbf{v}$ minden koordinátájára is:

$$|v_iK^i| < \frac{1}{d+1}.$$

Ezért, ha a $\mathbf{v}$-nek megfelelő $v(x)$ polinomba behelyettesítjük (PK) egy kis g gyökét, akkor $|g| \le K$ figyelembe vételével:

$$|v(g)| = \left|\sum_{i=0}^{d} v_ig^i\right| \le \sum_{i=0}^{d} \left|v_ig^i\right| \le \sum_{i=0}^{d} \left|v_iK^i\right| < \sum_{i=0}^{d} \frac{1}{d+1} = 1,$$

azaz $|v(g)| < 1$. Másrészt korábban láttuk, hogy C_1 elemeinek bármely lineáris kombinációjába és így v-be is, g-t behelyettesítve egészet kapunk. E kettő csak úgy teljesülhet egyszerre, ha $v(g) = 0$.

Kaptuk tehát, hogy (PK) kongruencia bármely $K = c_1'(d)N^{\frac{2}{d(d+1)}}$-nél kisebb abszolútértékű megoldását $v(x) \in \mathbb{Q}[x]$ polinomba helyettesítve 0-t kapunk, azaz $v(x)$ racionális polinom gyökei között találjuk (PK) összes kis megoldását, ráadásul ezek számát egyúttal $\dim \mathcal{L}(B_1)$-ben korlátoztuk.

Ha a kiindulási polinomok C_1 halmazát [9] szerint (két lépcsőben) tovább bővítjük, előbb

$$C_2 := \{x^i : 0 \le i < d\} \cup \{\big(p(x)/N\big) \cdot x^i : 0 \le i < d\}$$

halmazra, a hozzá tartozó B_2 bázist és $\mathcal{L}(B_2)$ rácsot analóg módon definiálva

$$B_2 = \begin{pmatrix} 1 & 0 & \dots & 0 & \frac{a_0}{N} & 0 & 0 & \dots & 0 \\ 0 & K & \dots & 0 & \frac{a_1K}{N} & \frac{a_0K}{N} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & \frac{a_2K^2}{N} & \frac{a_1K^2}{N} & \frac{a_0K^2}{N} & \dots & 0 \\ \vdots & & & & & & & & \\ 0 & 0 & \dots & 0 & \frac{a_{d-2}K^{d-2}}{N} & \frac{a_{d-3}K^{d-2}}{N} & \frac{a_{d-4}K^{d-2}}{N} & \dots & 0 \\ 0 & 0 & \dots & K^{d-1} & \frac{a_{d-1}K^{d-1}}{N} & \frac{a_{d-2}K^{d-1}}{N} & \frac{a_{d-3}K^{d-1}}{N} & \dots & 0 \\ 0 & 0 & \dots & 0 & \frac{1K^d}{N} & \frac{a_{d-1}K^d}{N} & \frac{a_{d-2}K^d}{N} & \dots & 0 \\ 0 & 0 & \dots & 0 & 0 & \frac{1K^{d+1}}{N} & \frac{a_{d-1}K^{d+1}}{N} & \dots & \frac{a_0K^{d+1}}{N} \\ 0 & 0 & \dots & 0 & 0 & 0 & \frac{1K^{d+2}}{N} & \dots & \frac{a_1K^{d+2}}{N} \\ \vdots & & & & & & & & \\ 0 & \dots & \dots & \dots & \dots & \dots & \dots & 0 & \frac{1K^{2d-1}}{N} \end{pmatrix},$$

$$\dim \mathcal{L}(B_2) = 2d, \quad \det \mathcal{L}(B_2) = N^{-d}K^{2d(2d-1)/2}$$

adódik, továbbá

$$c_2(d)\big(N^{-d}K^{2d(2d-1)/2}\big)^{1/(2d)} < 1/(2d)$$

feltételt, ill. $K \le c_2'(d)N^{1/(2d-1)}$ korlátot kapjuk K-ra. Ha ezután még tovább bővítjük C_2-t és valamely rögzített h pozitív egészre a következő polinomhalmazt vesszük:

$$C_3 := \{\big(p(x)/N\big)^j x^i : 0 \le i < d, 0 \le j < h\}$$

akkor a gondolatmenet úgyanígy működik: C_3 elemeinek kombinációi továbbra is 0-t adnak kis g-t behelyettesítve, B_3 mátrix a következő polinomok mátrixa (változatlanul (x^i/K^i) bázisban):

$$B_3 = \left[x^0, x^1, \dots, x^{d-1}, \frac{x^0p(x)}{N}, \frac{x^1p(x)}{N}, \dots, \frac{x^{d-1}p(x)}{N}, \frac{x^0p^2(x)}{N^2}, \dots, \frac{x^{d-1}p^{h-1}(x)}{N^{h-1}}\right]$$

azaz első $2d$ oszlopát B_2 mátrix oszlopai adják 0-kkal kiegészítve, a teljes B_3 pedig a következő alakú:

$$B_3 = \left(\begin{array}{ccc|ccccc} & & & * & \dots & \dots & \dots & 0 \\ & B_2 & & * & \dots & \dots & \dots & 0 \\ & & & * & \dots & \dots & \dots & 0 \\ \hline 0 & 0 & 0 & \frac{1K^{2d}}{N^2} & * & * & * & * \\ 0 & 0 & 0 & 0 & \frac{1K^{2d+1}}{N^2} & * & * & * \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & \dots & 0 & \frac{1K^{dh-1}}{N^h} \end{array}\right),$$

ahol a $*$-gal jelöltek (a mátrix nem feltétlenül 0 elemei) az $x^i(p(x)/N)^j$ polinomok nem-főegyüttható, például $1K^{2d}/N^2$ elem fölött $\frac{K^{2d-1}}{N^2}\sum_{\ell=0}^{2d-1} a_\ell a_{2d-1-\ell}$ szerepel, ami $x^0p^2(x)/N^2$ polinomban a $2d-1$ fokú tag együtthatója, K^{2d-1}-gyel felszorozva a bázisnak megfelelően. A többi elemet is hasonlóan kapjuk.
Könnyen láthatóan $\dim \mathcal{L}(B_3) = dh$ és

$$\det \mathcal{L}(B_3) = N^{-\frac{1}{2}dh(h+1)} B^{\frac{1}{2}dh(dh-1)}.$$

Emiatt a feltétel tovább enyhül, a K-ra vonatkozó korlát tovább nő:

$$\left(N^{-\frac{1}{2}dh(h+1)} B^{\frac{1}{2}dh(dh-1)}\right)^{\frac{1}{dh}} < c_3(d,h)$$

és

$$K \le c_3'(d,h) N^{\frac{h-1}{dh-1}}.$$

Itt persze c_3 és c_3' a d-n kívül a választott h-tól is függ. A kapott eredményt tételként is megfogalmazzuk.

4.2. Tétel. *Legyen $p(x) \in \mathbb{Z}[x]$ egy d fokú, 1 főegyütthatós, $\mathbb{Z}$ fölött irreducibilis polinom és tekintsük a következő kongruenciát:*

$$p(x) \equiv 0 \mod N.$$

Ekkor bármely pozitív egész h-ra létezik olyan, csak d-től és h-tól függő $c_3'(d,h)$ konstans, hogy bármely $K \le c_3'(d,h) N^{\frac{h-1}{dh-1}}$ korlátra a fenti kongruencia összes $|g| < K$ megoldása gyöke egy legfeljebb $dh-1$ fokú, racionális polinomnak, amely ráadásul polinomiális időben meghatározható, tehát maguk a megoldások is polinom időben megadhatók. Speciálisan a megoldások száma legfeljebb $dh-1$.

Megjegyzések.

1) Ha a legutóbbi konstrukcióban megvizsgáljuk a gyökök tartományának K korlátját, ott N kitevőjében $N^{\frac{h-1}{dh-1}}$ szerepel. Ezt $N^{\frac{1}{d}}$-hez viszonyítva:

$$\frac{1}{d} - \frac{h-1}{d-1} = \frac{d-1}{dh-1} < \frac{1}{dh},$$

emiatt h növelésével $N^{\frac{1}{d}}$ tetszőlegesen megközelíthető – persze megnövekedett futásidő árán.

2) A konstansok (c_3, c_3') növelése érdekében a polinomhalmazhoz Lenstra, ill. (tőle függetlenül) Howgrave-Graham ötlete alapján hozzávehetjük a

$$b_k(x) := \frac{x(x-1)\cdots(x-k+1)}{k!}$$

polinomokat, valamint (x/K^i) polinomok helyett a Csebisev-polinomok használatával is nyerhetünk további konstans szorzót. Részleteket illetően lásd az eredeti cikket [9].

4.3. RSA feltörése speciális esetekben

Az RSA titkosítás

Az RSA nyilvános kulcsú titkosítási algoritmus [2], melynek során az üzenet titkosításához egy (e, N) nyilvános kulcsot használunk, a titkosított üzenet megfejtése pedig egy (d, N) titkos kulcs segítségével történik. Mind a kulcsokat, mind az üzeneteket reprezentálhatjuk számként és a továbbiakban így is gondolunk rájuk. A kulcsok generálása a következőképpen történik: N-et két nagy prím, p és q szorzatának választjuk, majd N-hez kiválasztjuk e, d pár úgy, hogy teljesítsék a következő kongruenciát:

$$ed \equiv 1 \mod \varphi(N),$$

ahol $\varphi(x)$ az Euler-féle φ függvény, itt N prímtényezős alakja miatt $\varphi(N) = (p-1)(q-1)$, másképp fogalmazva a $\mathbb{Z}_N^*$ multiplikatív csoport rendje. A kulcsgenerálásban p, q prímek kiválasztása jól ismert (és polinomiális idejű) prímtesztek segítségével történhet, amelyekkel a kellő tartományból véletlenül választott számok prím voltát ellenőrizhetjük. Adott $\varphi(pq) = (p-1)(q-1)$-hez e és d közül egyiket, mondjuk e-t tetszőlegesen választhatjuk ($e < \varphi(N)$ mellett), az e-hez tartozó d pár, azaz $d \equiv e^{-1} \mod \varphi(N)$ kiszámítása pedig a kiterjesztett euklideszi algoritmussal történhet.

Legyen ezután az $m \in \mathbb{Z}_N^*$ szám a titkosítatlan üzenet *(plaintext)*, c szám pedig ennek titkosított változata *(ciphertext)*. A titkosítás és a visszafejtés folyamata a következőképpen zajlik:

$$C(m) := c = m^e \mod N,$$
$$D(c) := d = c^d \mod N,$$

ahol a kis Fermat-tétel miatt

$$D(c) = c^d = m^{ed} \equiv m \mod N,$$

vagyis a titkosított üzenet visszafejtése (d, N) ismeretében valóban lehetséges. Ha d-t nem is ismerjük, de N faktorizációjáról (azaz p, q prímekről) rendelkezünk információval, a visszafejtés akkor is könnyen lehetséges, hiszen akkor $\varphi(N)$ értékét is tudjuk, így adott (e, N) párhoz a kulcsgenerálásnál látott módon d-t a kiterjesztett euklideszi algoritmussal hatékonyan kiszámolhatjuk.
A lehetséges kulcsok végigpróbálgatása (brute-force) természetesen kínálkozó lehetőség, azonban futásideje $O(N)$, az input bitméretében exponenciális. A jelenleg ismert leggyorsabb általános algoritmus, amely egy szám prímfelbontását szolgáltatja a General Number Field Sieve (GNFS), melynek futásideje

$$O\left(\exp\left(\sqrt[3]{\frac{64}{9}\log N \log\log N}\right)\right),$$

továbbra is szubexponenciális [19]. Nyitott kérdés, hogy létezik-e polinomiális idejű prímfaktorizációs eljárás, másrészt az is, hogy létezik-e az RSA titkosítás általános visszafejtésére a prímfaktorizációnál lényegesen gyorsabb, prímfaktorizáció nélküli módszer. [6]
A továbbiakban az LLL algoritmus két olyan alkalmazását vizsgáljuk, amelyek d, ill. p és q konkrét ismerete nélkül, de azokról valamilyen kiegészítő információ birtokában lehetővé teszik az RSA titkosírás visszafejtését. [8]

RSA alacsony kitevővel

Az RSA algoritmus e, d pár megválasztásában elég nagy szabadságot hagy, egyedül az $ed \equiv 1 \mod \varphi(N)$ kongruencia teljesítését szabja feltételként, e és d közül egyik szabadon választható. Mivel a titkosítás folyamatában:

$$C(m) = m^e \mod N,$$

a modulo N hatványozás időigénye e-től erősen függ, elterjedt szokás e értékét minél kisebbnek, például egy kisebb Fermat-prímnek, 3-nak, 17-nek vagy 65537-nek választani. Fermat-prímek esetén a speciális, $2^{2^n} + 1$-es alak miatt a számokban összesen

csak két 1-es bit van, ez a hatványozást még gyorsabbá teszi. Azonban ha ilyen alapfelállás mellett az üzenet egy részéről tudomásunk van, mondjuk m a következő alakú:

$$m = (m_1, m_2),$$

ahol m_1 és m_2 az üzenet, mint bitsorozat első és második része, és mondjuk ugyan m_2-t nem ismerjük, de m_1-ről rendelkezünk információval, akkor m_2, mindaddig, amíg nem túl nagy, hatékonyan meghatározható m_1 és $c = C(m)$ segítségével.
Ha például $m_1 =$ „a mai jelszó: ”, $m_2 =$ „titok456” (ill. ezek numerikus reprezentációi), akkor $e = 3$ nyilvános kitevő mellett $|m_2| < N^{\frac{1}{3}}$ esetén m_2 könnyen meghatározható m_1 és $c = C(m) = m^3 \bmod N$ értékéből, éspedig a 4.2. tételt a

$$p(m) = m^3 - c \equiv 0 \mod N$$

polinomra alkalmazva.
Egy másik alkalmazási lehetőség olyan titkosított üzenetek visszafejtése, ahol az RSA-titkosítást a feltörés további megnehezítése érdekében ún. *véletlenszerű kitöltésnek* (*random padding*) vetik alá. Ennek során az eredeti titkosítatlan üzenetet balra tolják valahány bittel, az új, alacsony helyiértékű helyekre pedig random biteket tesznek, tehát:

$$m = (m_1, r) = 2^k m_1 + r,$$

ahol r k-bites véletlen szám, majd a kapott m számot titkosítják RSA-val. Itt is föltesszük, hogy a nyilvános exponens $e = 3$, továbbá a hosszáadott véletlen szám mérete nem túl nagy: $|r| = k < N^{\frac{1}{9}}$.
Most tegyük fel, hogy ugyanazt a üzenetet kétszer titkosítják, persze más-más véletlen kitöltéssel és a kapott c, c' titkosított üzenetek rendelkezésünkre állnak. Legyen $d := r' - r$ a két véletlen kitöltő szám különbsége. Belátjuk, hogy előbb d, majd ebből m meghatározható, ha d elég kicsi. Felírva a c-re és c'-re vonatkozó kongruenciát:

$$\begin{aligned} c &\equiv m^3 = (2^k m_1 + r)^3 && \bmod N, \\ c' &\equiv (m')^3 = (2^k m_1 + r')^3 = (m+d)^3 && \bmod N. \end{aligned}$$

A fenti két kongruenciából kiküszöbölhetjük m-et, méghozzá a következő két polinom:

$$p(m) = m^3 - c$$

és

$$q(m) = (m+d)^3 - c' = m^3 + 3m^2 d + 3md^2 + (d^3 - c')$$

rezultánsát véve:

$$\mathrm{Res}(p,q) = \det \begin{pmatrix} 1 & 0 & 0 & -c & 0 & 0 \\ 0 & 1 & 0 & 0 & -c & 0 \\ 0 & 0 & 1 & 0 & 0 & -c \\ 1 & 3d & 3d^2 & d^3 - c' & 0 & 0 \\ 0 & 1 & 3d & 3d^2 & d^3 - c' & 0 \\ 0 & 0 & 1 & 3d & 3d^2 & d^3 - c' \end{pmatrix} =$$

$$= d^9 + (3c - 3c')d^6 + (3c^2 + 21cc' + 3(c')^2)d^3 + (c - c')^3.$$

A kapott kifejezésre, mint d polinomjára alkalmazva a 4.2. tételt megkaphatjuk

$$\mathrm{Res}(f,g)(d) \equiv 0 \mod N$$

kongruencia $|d| \leq N^{\frac{1}{9}}$ megoldásait és így d visszafejthető.
Mivel mindkét titkosítatlan üzenet jelentést hordozó (determinisztikus) része ugyanazon m_1 volt, ezért $d = r' - r$ mellett $d = m' - m$ is fennáll, ezért tekintve újra a c, c'-re vonatkozó kongruenciát:

$$\begin{aligned} c &\equiv m^3 && \mod N, \\ c' &\equiv (m+d)^3 = m^3 + 3m^d + 3md^2 + d^3 && \mod N, \end{aligned}$$

innen elemi átalakítással [8]

$$m \equiv \frac{d(c' + 2c - d^3)}{c' - c + 2d^3} \equiv \frac{d(3m^3 + 3m^2d + 3md^2)}{3m^2d + 3md^2 + 3d^3} \mod N,$$

az eredeti titkosítatlan üzenet megkapható.
Láthatjuk tehát, hogy az RSA nyilvános kulcsú titkosítási algoritmus bizonyos erős feltételek fennállása esetén feltörhető rácsok, illetve az LLL algoritmus segítségével. Az RSA titkosítással szembeni további (nem csak rácselméleti) támadási lehetőségeket tekintik át Dan Boneh [6] és Glenn Durfee [10] munkái, a rácselmélet és a kriptológia összefüggéseinek általános vizsgálatáról szól Daniele Micciancio és Shafi Goldwasser könyve [14].
A következő fejezetben a rácsok ellenkező irányú kriptográfiai alkalmazására térünk ki: megvizsgáljuk, hogyan lehet rácsok segítségével nehezen feltörhető titkosítási rendszereket létrehozni.

4.4. Rácsalapú titkosítások

A rácsalapú titkosítások azon a feltételezésen alapulnak, hogy a rácsproblémák (SVP, CVP, illetve SIVP) megoldása nehéz feladat. A nagy számok faktorizációja, vagy éppen a diszkrét logaritmus probléma, melyek nehézségén sok titkosítási

algoritmus alapszik, bár valóban nehezen megoldható problémák, mindazonáltal az utóbbi évtizedekben számítástudományi értelemben sokat fejlődött a megoldásuk. A prímtenyezőkre bontásra rendelkezünk szubexponenciális idejű algoritmussal (a General Number Field Sieve (GNFS) ilyen, [19]), ugyanez elmondható a diszkrét logaritmus problémáról is. Ezek jelentős eredmények, amikkel összevetve a rácsproblémák megoldására vonatkozó eddigi erőfeszítések sokkal kevésbé sikeresek: több évtizednyi kutatás után sem sikerült az SVP probléma megoldásához szükséges futásidőt $2^{O(n)}$ alá szorítani, sőt, a 3.1. sejtés szerint még polinomiális hibával közelítő, polinomális idejű algoritmus sem létezik.
A kriptográfiai rendszerek alapjaként használt, fent említett két probléma további gyengesége, hogy mindkettőre hatékony kvantumalgoritmus ismert [36]. Ehhez kapcsolódóan megjegyezzük, hogy valamennyi ismert és széles körben alkalmazott titkosítás egy adott probléma *átlagos esetben vett* nehézségén alapul, míg az alábbiakban olyan rendszerekre is kitérünk, melyek adott problémák *legrosszabb esetben vett* nehézségén alapulnak.
Ebben a fejezetben [29] alapján bemutatunk egy egyszerű rácsalapú titkosítási rendszert és rövid áttekintést adunk a jelenleg kutatott és elérhető rácsalapú titkosításokról.

A GGH/HNF titkosítás

A GGH titkosítási sémát eredetileg Oded Goldreich, Shafi Goldwasser és Shai Halevi dolgozta ki [16], ez McEliece korábbi titkosítási rendszerének [25] egy, a hálókra megfogalmazott változata. A titkosítás a következő kézenfekvő gondolaton alapszik. Vegyünk egy rácsot, nyilvános kulcsként használjuk egy tetszőleges, (ortogonalitás tekintetében) nem túl jó bázisát, titkos kulcsként pedig ugyanezen rácsnak egy jó bázisát. Titkosításkor helyezzünk el egy vektort ebben a térben az egyik rácsvektor közelében – az információt ennek a közeli rácsvektorhoz képesti relatív helyzete hordozza –, így visszafejtéskor egy CVP feladattal állunk szemben: az adott *(ciphertext)* vektorhoz közeli rácsvektort kell találnunk, hogy különbségükből kiszűrhessük az eredeti *(plaintext)* információt. A CVP probléma (általános bázisban) nehéz, bázisokat merőlegessé transzformálni nehéz, ez alapján a titkos kulcs nélkül az eredeti üzenet megfejtése nehéz feladat. A titkosítási séma pontosabban a következő.

Kulcsgenerálás

> A titkos kulcs egy B bázis, amely elég ortogonális ahhoz, hogy a CVP feladat effektíven megoldható legyen $\mathcal{L}(B)$-ben.
> A nyilvános kulcs egy ennél sokkal nagyobb ortogonalitási defektussal

(2.1. definíció) rendelkező H bázis, amelyre $\mathcal{L}(H) = \mathcal{L}(B)$.
Daniele Micciancio [26] javaslata alapján a nyilvános kulcs legyen az eredeti B bázis ún. Hermite Normálformája (HNF). Ez az eredeti B bázisból polinomálisan megkapható a Gauss-elimináció egy változatának segítségével és eredményeképpen a H bázis mátrixa felsőháromszög-mátrix alakú. (Innen a GGH/HNF elnevezés.)

Titkosítás

Válasszunk az $\mathcal{L}(H)$ rácsból egy $\mathbf{v}$ vektort és perturbáljuk egy kis normájú $\mathbf{r}$ vektorral, amely tartalmazza a titkosítani szánt információt. A rácsvektor alkalmas megválasztásával kapcsolatban az eredeti cikk véletlen választást javasol, Micciancio erre is megfogalmazott egy javaslatot, lásd [26]. A kapott vektort nevezzük $\mathbf{r}$ mod H-nak, ez a titkosított üzenet.

Visszafejtés

A visszafejtés egy CVP feladatpéldány, amely a rendelkezésre álló bázis függvényében nagyon nehéz is lehet, de a jóval ortogonálisabb B bázisban már könnyű: $\mathcal{L}(B)$ rácsban keressük meg az $\mathbf{r}$ mod H vektorhoz legközelebbi $\mathbf{v}$ rácsvektort, a különbségük az eredeti, titkosítatlan üzenet. Az algoritmus helyessége azon múlik, hogy $\mathbf{r}$ vektor elég rövid legyen ahhoz, hogy a $\mathbf{v} + \mathbf{r}$-hez legközelebbi rácsvektor egyértelmű legyen.

Megjegyezzük, hogy a GGH/HNF titkosítás bizonyos gyakorlati jellegű támadásokkal feltörhető [41], mégis ezt mutattuk be részletesen, mivel egyszerűsége miatt nagyon jól mutatja a rácselmélet alkalmazhatóságát a kriptográfiában. Továbbá az is igaz, hogy aszimptotikusan hatékony támadás nem ismert ellene, azaz a biztonsági paraméter (itt a rács dimenziója) növelésével az említett támadások kizárhatók. Ami mégis ellene szól, hogy a rács dimenziójának növelésével maga a GGH/HNF titkosítás is impraktikussá válhat: mivel egy általános bázis tárolásához $\Omega(n^2)$ tár szükséges, így a titkosítás/visszafejtés futásideje is négyzetesen nő a dimenzióval. Az áttekintő részben szereplő további rácsalapú titkosítások gyakorlati használhatósága éppen azon múlik, hogy a rácsok bázisai speciális struktúrát követnek, melyek tárolása jóval effektívebben megoldható.

A további rendszerekről

A rácsalapú titkosításokat általában két osztályba sorolhatjuk. Az egyik osztályba az olyan titkosítási sémák tartoznak, amelyek konkrét gyakorlati megvalósítása belátható futásidőt garantál, de nem rendelkeznek a biztonságosságukra vonatkozó erős

matematikai bizonyítékokkal, ezek feltörhetőségéről csak annyit tudunk, hogy az eddig ismert támadásokkal szemben védettek. Ilyenek a fent ismertetett GGH/HNF (de a támadhatóságával kapcsolatban lásd [41]), vagy az NTRU rendszer, amely a jelenleg ismert, futásidő tekintetében leghatékonyabb rácsalapú titkosítás [37].
A másik osztályba olyan titkosítási sémák tartoznak, amelyek biztonságosságára erős matematikai bizonyítékokkal rendelkezünk, ám a gyakorlati használhatóságukat korlátozza a nem túl kedvező futásidejük. Ilyen pl. az Ajtai–Dwork titkosítás [11] vagy az Oded Regev által kidolgozott LWE (Learning with errors) [31]. Az Ajtai–Dwork rendszer feltöréséről belátható, hogy ekvivalens több, különböző rácsra vonatkozó SVP feladat megoldásával, éspedig nem az átlagos esetekkel, hanem *legrosszabbakkal*. Az LWE titkosítás feltörhetőségéből pedig olyan polinomiális idejű kvantumalgoritmus létezése következne, amely az SVP feladatot tetszőleges approximációs faktorral megoldja [29, 19. o.]. Futásidőben az LWE elmarad az NTRU rendszertől, mindazonáltal az LWE az egyetlen jelenleg ismert rácsalapú titkosítás, amelynek biztonságosságára erős matematikai garanciával rendelkezünk, de amely egyúttal futásideje alapján a gyakorlatban is használható.
Végül megemlítjük, hogy Shor prímtényezőkre bontásra és diszkrét logaritmusra adott 1997-es kvantumalgoritmusa [36] óta folyik a kutatás rácsfeladatok kvantumalgoritmusokkal való megoldására, eddig gyakorlatilag semmilyen sikerrel. A problémát az okozza, hogy Shor algoritmusának egy lényeges belső tulajdonsága (a „periodikusság-kereső technika") a rácsproblémákra lényegében alkalmazhatatlannak tűnik.
A rácsfeladatok ideális jelöltnek tűnnek kvantumalgoritmusokkal való megközelítésre – mivel elfogadott álláspont, hogy bizonyos tipikus approximációs faktorokra nem NP-nehezek, továbbá a periodikus struktúrájuk alapján és mert a kvantumalgoritmusokban jelentős szerepet játszó Fourier-transzformáció erősen kötődik a duális rács fogalmához. Mindezek ellenére jelenleg nem ismert olyan kvantumalgoritmus, amely rácsfeladatokat a klasszikus (nem-kvantum) algoritmusoknál lényegesen gyorsabban oldana meg. Mindezek a tapasztalatok a következő sejtés megfogalmazásához vezettek [29, Conj. 1.2.].

4.3. Sejtés. *Nem létezik polinomiális idejű kvantumalgoritmus, amely a rácsproblémákat polinomiális hibával közelíti.*

Összefoglalás

Szakdolgozatomban bemutattam a matematikai rácsokat, a jellemző rácsfeladatokat, azok bonyolultságát. Az eddigi kutatások alapján a közelítő megoldásukra gyakorla-

tilag egyedül rendelkezésre álló polinomiális algoritmust, a Lenstra–Lenstra–Lovász-algoritmust, valamint ennek számtalan alkalmazásából a teljesség igénye nélkül válogatva kriptográfiai és diofantikus approximációra vonatkozó példákat, végül röviden kitértem a rácsokon alapuló biztonságos kriptográfiai rendszerekre.

Irodalomjegyzék

[1] Oliver Aberth. Iteration methods for finding all zeros of a polynomial simultaneously. *Journal of Mathematics and Computer Science*, 27(122):339–344, April 1973.

[2] R.L. Rivest; A. Shamir; L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, Volume 21 Issue 2:120–126, 1978.

[3] M. Ajtai. The shortest vector problem in L_2 is np-hard for randomized reductions (extended abstract). In *Proceedings of the thirtieth annual ACM symposium on Theory of computing - STOC '98*, pages 10–19, 1998.

[4] Radnai András. *Rácselmélet alkalmazása a számelméletben.* Szakdolgozat, ELTE, 2010.

[5] L. Bairstow. Investigations relating to the stability of the aeroplane. *Reports and Memoranda, Advisory Committee for Aeronautics*, 154:51–64, 1914.

[6] Dan Boneh. Twenty years of attacks on the RSA cryptosystem. *Notices of the American Mathematical Society (AMS)*, 46(2):203–213, 1999.

[7] J. H. Conway. A perfect group of order 8,315,553,613,086,720,000 and the sporadic simple groups. In *Proceedings of the National Academy of Sciences of the United States of America*, volume 61, pages 398–400, 1968.

[8] Don Coppersmith. Small solutions to polynomial equations and low exponent RSA vulnerabilities. *Journal of Cryptology*, 10:233–260, 1997.

[9] Don Coppersmith. Finding small solutions to small degree polynomials. In *Proceedings of Cryptography and Lattice Conference*, volume 2146. Springer-Verlag, 2001.

[10] Glenn Durfee. *Cryptanalysis of RSA Using Algebraic and Lattice Methods.* PhD thesis, Stanford University, 2002.

[11] M. Ajtai; C. Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, pages 284–293, 1997.

[12] L. W. Ehrlich. A modified newton method for polynomials. *Communications of the ACM*, 10(2):107–108, 1967.

[13] C. F. Gauss. *Disquisitiones Arithmeticae*. Fleischer, Leipzig, 1801.

[14] Daniele Micciancio; Shafi Goldwasser. *Complexity of Lattice Problems: A Cryptographic Perspective*. The Springer International Series in Engineering and Computer Science. Springer, softcover reprint of the original 1st ed. 2002 edition, 2012.

[15] Oded Goldreich; Shafi Goldwasser. On the limits of nonapproximability of lattice problems. *Journal of Computer and System Sciences*, 60:540–563, 2000.

[16] Oded Goldreich; Shafi Goldwasser; Shai Halevi. Public-key cryptosystems from lattice reduction problems. In *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, pages 112–131, 1997.

[17] Nicholas Howgrave-Graham. Finding small roots of univariate modular equations revisited. In *Proceedings of Cryptography and Coding*, volume 1355, pages 131–142. Springer-Verlag.

[18] Johan Håstad. Solving simultaneous modular equations of low degree. *SIAM Journal on Computing*, 17(2):336–341, 1988.

[19] A. K. Lenstra; H. W. Lenstra Jr. The development of the number field sieve. *Lecture Notes in Mathematics*, 1554, 1993.

[20] R. M. Karp. Reducibility among combinatorial problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[21] A. K. Lenstra; H. W. Lenstra; L. Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261:515–534, 12 1982.

[22] Király Tamás; Kis Tamás; Szegő László. *Online jegyzet az Egészértékű Programozás I. és II. tárgyhoz*. ELTE, 2017.

[23] Lovász László. *An Algorithmic Theory of Numbers, Graphs and Convexity*. CBMS-NSF Regional Conference Series in Appl. Math. Society for Industrial and Applied Mathematics, Philadelphia, 1986.

[24] Lovász László. *Algoritmusok Bonyolultsága*. Typotex Kiadó, 2014.

[25] R. J. McEliece. A public-key cryptosystem based on algebraic coding theory. *Technical report, Jet Propulsion Laboratory*, 1978.

[26] Daniele Micciancio. Improving lattice based cryptosystems using the Hermite Normal Form. In Joseph Silverman, editor, *Cryptography and Lattices Conference — CaLC 2001*, volume 2146 of *Lecture Notes in Computer Science*, pages 126–145, Providence, Rhode Island, 29–30 March 2001. Springer-Verlag.

[27] Phong Q. Nguyễn. The LLL algorithm. 2010.

[28] Freud Róbert. *Számelmélet*. Nemzeti Tankönyvkiadó, Budapest, 2000.

[29] D. Micciancio; O. Regev. Lattice-based cryptography. 2008.

[30] Dorit Aharonov; Oded Regev. Lattice problems in NP intersect co-NP. *Journal of the ACM*, 52:749–765, 09 2005.

[31] O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the Thirty-seventh Annual ACM Symposium on Theory of Computing*, STOC '05, pages 84–93, New York, NY, USA, 2005. ACM.

[32] A. Roberts. *Properties of Leech Lattice*. University of Puget Sound, Washington, 2006.

[33] J. C. Lagarias; H. W. Lenstra; C. P. Schnorr. Korkin-Zolotarev bases and successive minima of a lattice and its reciprocal lattice. *Combinatorica*, 10:333–348, 1990.

[34] Catherine Goldstein; Norbert Schappacher; Joachim Schwermer. *Shaping of Arithmetic after C. F. Gauss's Disquisitiones Arithmeticae*. Springer, 2007.

[35] Igor Semaev. In proceedings of the 2001 cryptography and lattices conference (CALC'01): A 3-dimensional lattice reduction algorithm. *Lecture Notes in Computer Science*, 2146:181–193, 2001.

[36] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Review*, 41:303–332, 01 1999.

[37] J. Hoffstein; J. Pipher; J. H. Silverman. NTRU: A ring-based public key cryptosystem. In Joe Buhler, editor, *ANTS*, volume 1423 of *Lecture Notes in Computer Science*, pages 267–288. Springer, 1998.

[38] M. Ajtai; R. Kumar; D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing - STOC '01*, pages 601–610, 2001.

[39] Guillaume Hanrot; Xavier Pujol; Damien Stehlé. Algorithms for the shortest and closest lattice vector problem. 2013.

[40] Phong Q. Nguyễn; Damien Stehlé. Low-dimensional lattice basis reduction revisited. *ACM Transactions on Algorithms*, 5:1–48, 10 2009.

[41] Phong Q. Nguyễn; Jacques Stern. Cryptanalysis of the Ajtai-Dwork cryptosystem. In *Advances in Cryptology - CRYPTO '98, 18th Annual International Cryptology Conference, Santa Barbara, California, USA, August 23-27, 1998, Proceedings*, pages 223–242, 1998.

[42] T. Szőnyi. *Szimmetrikus struktúrák.* Typotex, Budapest, 2013.

[43] Brigitte Vallée. *Une approche géométrique de la réduction des réseaux en petite dimension.* PhD thesis, Université Caen-Normandie, 1986.

[44] Phong Q. Nguyễn; Brigitte Vallée, editor. *The LLL Algorithm - Survey and Applications.* Information Security and Cryptography. Springer, 2010.

[45] P. van Emde Boas. Another NP-complete partition problem and the complexity of computing short vectors in a lattice. *Technical report*, 81-04, 1981.

[46] N. M. Vetčinkin. Uniqueness of classes of positive quadratic forms on which values of the Hermite constant are attained for $6 \leq n \leq 8$. *Geometry of positive quadratic forms, Trudy Math. Inst. Steklov.*, 152:34–86, 1980.

[47] H. Cohn; A. Kumar; S. D. Miller; D. Radchenko; M. Viazovska. The sphere packing problem in dimension 24. 2016.

[48] M. Viazovska. The sphere packing problem in dimension 8. 2016.

NYILATKOZAT

Név: Baranyi Károly Tamás
ELTE Természettudományi Kar, szak: Alkalmazott matematikus MSc
NEPTUN azonosító: COSMIU
Szakdolgozat címe: Rácsalgoritmusok

A szakdolgozat szerzőjeként fegyelmi felelősségem tudatában kijelentem, hogy a dolgozatom önálló munkám eredménye, saját szellemi termékem, abban a hivatkozások és idézések standard szabályait következetesen alkalmaztam, mások által írt részeket a megfelelő idézés nélkül nem használtam fel.

Budapest, 2017. május 30.

_____

a hallgató aláírása