Úvod do mriežok

Martin Stanek

2025

KI FMFI UK Bratislava

Úvod

- využitie aj mimo kryptológie – teória kódovania, optimalizácia
- základ pre konštrukciu rôznych kryptografických schém
  - klasické schémy, napr. asymetrické šifrovanie alebo podpisy
  - iné: plne homomorfné šifrovanie, šifrovanie založené na identite a pod.
- ťažké problémy súvisiace s mriežkami
  - niektoré odolné voči kvantovým počítačom (PQC konštrukcie)
  - dôkazy bezpečnosti – redukcie s predpokladom worst-case zložitosti problémov
- využitie v kryptoanalýze
  - napr. útoky na RSA schému s malým verejným exponentom

PQC štandardizácia

- viaceré NIST PQC štandardizované konštrukcie sú založené na mriežkach
- FIPS 203 (KEM): ML-KEM (CRYSTALS-Kyber), kde ML = „Module-Lattice“
- FIPS 204 (podpisy): ML-DSA (CRYSTALS-Dilithium)
- pripravovaný štandard FIPS 206 (podpisy): FN-DSA (Falcon),
  kde FN = „FFT over NTRU-Lattice“

Mriežka – definícia

- $v_1, \ldots, v_k \in \mathbb{R}^n$ – množina lineárne nezávislých vektorov, tzv. báza
- **mriežka** je množina celočíselných lineárnych kombinácií vektorov bázy:

$$L = \{a_1 v_1 + \ldots + a_k v_k \mid a_1, \ldots, a_n \in \mathbb{Z}\}$$

  - n – dimenzia mriežky; k – hodnosť mriežky
- ďalej nás budú zaujímať mriežky s plnou hodnosťou $k = n$
- báza mriežky je ľubovoľná množina lineárne nezávislých vektorov, ktorá ju generuje
  - mriežka má veľa rôznych báz, všetky s rovnakým počtom prvkov
- vektorový priestor vs. mriežka
  - lineárne kombinácie s koeficientami z $\mathbb{R}$ vs. koeficienty zo $\mathbb{Z}$
  - vektor vs. bod

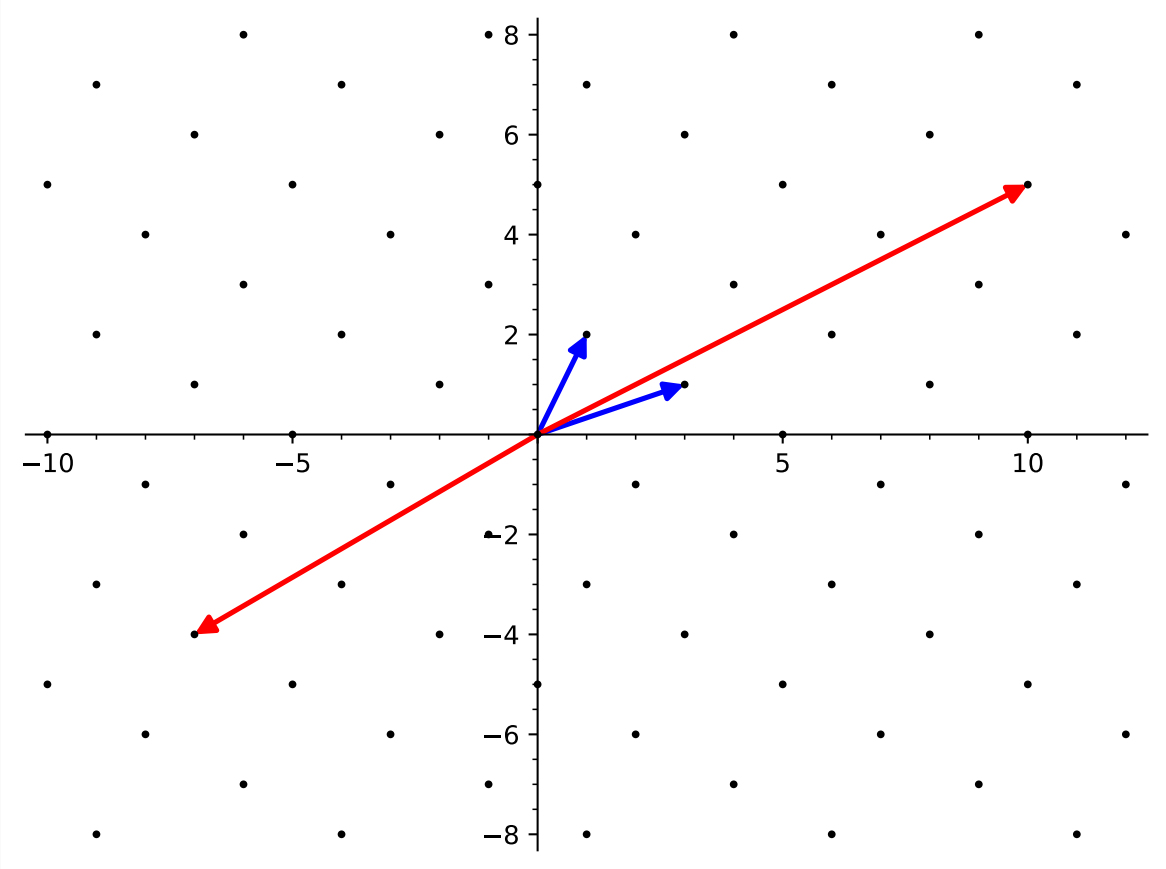
Iné pohľady na mriežku

- iná ekvivalentná definícia: mriežka je diskrétna aditívna podgrupa $\mathbb{R}^n$
- väčšinou pracujeme s celočíselnými mriežkami, kde všetky vektory bázy sú zo $\mathbb{Z}^n$
  - aditívna podgrupa $\mathbb{Z}^n$
- triviálna mriežka je celé $\mathbb{Z}^n$
- mriežky získame zo $\mathbb{Z}^n$ použitím lineárnej transformácie $\boldsymbol{B}\mathbb{Z}^n$
  - kde $\boldsymbol{B} \in \mathbb{R}^{n \times n}$ je regulárna matica (báza mriežky, riadky sú vektory bázy)

Základné pojmy

- skalárny súčin vektorov: $x \cdot y = (x_1, \ldots, x_n) \cdot (y_1, \ldots, y_n) = \sum_{i=1}^{n} x_i y_i$
- (Euklidovská) dĺžka vektora $\|x\| = \sqrt{x_1^2 + \ldots + x_n^2}$, $\|x\|^2 = x \cdot x$
- základný rovnobežnosten mriežky L s bázou $v_1, \ldots, v_n$:
  - $\mathcal{P} = \sum_{i=1}^{n} v_i \cdot \langle 0, 1) = \{t_1 v_1 + \ldots + t_n v_n \mid t_i \in \langle 0, 1)\}$
- dve bázy $\boldsymbol{B}_1, \boldsymbol{B}_2$ generujú rovnakú mriežku $\Leftrightarrow \boldsymbol{B}_1 = \boldsymbol{U}\boldsymbol{B}_2$, kde $\boldsymbol{U} \in \mathbb{Z}^{n \times n}$ je unimodulárna matica (teda $\det(\boldsymbol{U}) = \pm 1$)
- determinant mriežky: $\det(L)$ je objem základného rovnobežnostena $\mathrm{Vol}(\mathcal{P})$
  - pre mriežky s plnou hodnosťou: $\det(L) = \mathrm{Vol}(\mathcal{P}) = |\det(\boldsymbol{B})|$, kde $\boldsymbol{B}$ je báza L
  - $\det(L)$ je invariant mriežky, nemení sa zmenou bázy
  - rôzne bázy → rôzne základné rovnobežnosteny, všetky s rovnakým objemom

Príklad (celočíselnej) mriežky



- modrá báza: $(1, 2), (3, 1)$
- červená báza: $(10, 5), (-7, -4)$
- $\det(L) = 5$

$$\begin{pmatrix} 1 & 3 \\ -1 & -2 \end{pmatrix} \cdot \begin{pmatrix} 1 & 2 \\ 3 & 1 \end{pmatrix} = \begin{pmatrix} 10 & 5 \\ -7 & -4 \end{pmatrix}$$

Minimálne vzdialenosti

- prvé minimum mriežky: $\lambda_1 = \min_{x \neq y \in L} \|x - y\| = \min_{x \in L \setminus \{0\}} \|x\|$
  - najkratšia vzdialenosť medzi bodmi
  - najkratší nenulový vektor
- ďalšie minimá $\lambda_2, \ldots, \lambda_n$:
  - λ_i – najmenšie $r \in \mathbb{R}$ také, že L obsahuje i nezávislých vektorov dĺžky $\leq r$
  - $\lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n$
- príklady:
  - naša mriežka: $\lambda_1 = \|(1, 2)\| = \sqrt{5}, \lambda_2 = \|(2, -1)\| = \sqrt{5}$
  - pre $\mathbb{Z}^n$: $\lambda_1 = \lambda_2 = \ldots = \lambda_n = 1$
- vzdialenosť bodu $z \in \mathbb{R}^n$ od mriežky L: $\mu(z, L) = \min_{x \in L} \|x - z\|$

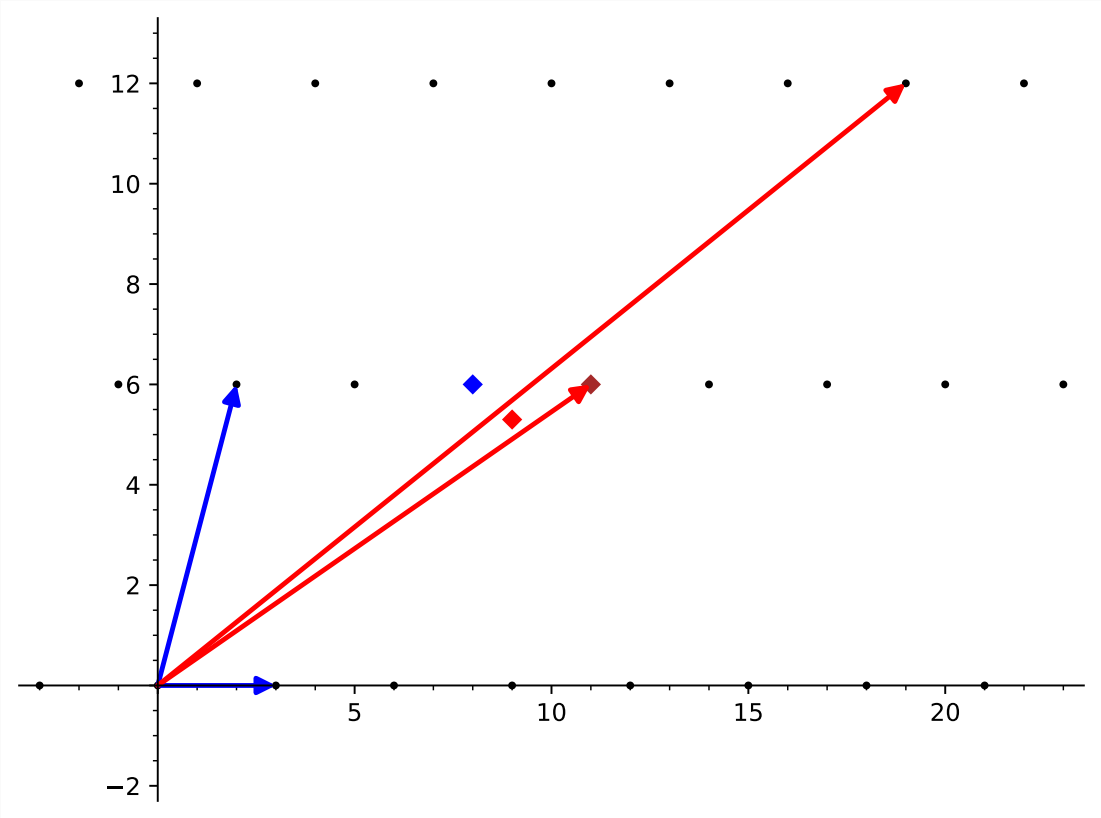
Výpočtové problémy na mriežkach

- daná je mriežka L dimenzie n (určená svojou bázou)
- **SVP** (Shortest Vector Problem): nájsť najkratší nenulový vektor, teda $v \in L$: $\|v\| = \lambda_1$
  - NP ťažký problém
  - najlepší algoritmus má heuristickú časovú zložitosť $2^{0.292+o(n)}$
- **SVP$_\gamma$** (aproximačná verzia SVP): nájsť nenulový vektor $v \in L$: $\|v\| \leq \gamma\lambda_1$
  - ťažké pre malé γ
- **SIVP** (Shortest Independent Vectors Problem): nájsť n lineárne nezávislých vektorov $v_1, \ldots, v_n \in L$: $\max_{i=1,\ldots,n}\|v_i\| = \lambda_n$
- **SIVP$_\gamma$** (aprox. verzia SIVP): nájsť lin. nezávislé $v_1, \ldots, v_n \in L$: $\max_{i=1,\ldots,n}\|v_i\| = \gamma\lambda_n$.
- **CVP** (Closest Vector Problem): pre bod $z \in \mathbb{R}^n$ nájsť vektor $v \in L$: $\|z - v\| \leq \mu(z, L)$
- **CVP$_\gamma$** (aprox. verzia CVP): pre bod $z \in \mathbb{R}^n$ nájsť vektor $v \in L$: $\|z - v\| \leq \gamma\mu(z, L)$.

Poznámky k problémom

- každý z uvedených problémov môže mať viac riešení
  - napr. SVP má pre mriežku $\mathbb{Z}^2$ riešenia $(0, 1)$, $(1, 0)$, $(0, -1)$ a $(-1, 0)$
  - ľubovoľné riešenie je OK
- nepoznáme polynomiálne (PPT) algoritmy na riešenie uvedených problémov
  - ani pre aproximačné verzie (pre vhodné γ)
- existujú a sú využívané aj iné problémy na mriežkach
  - BDD (Bounded Distance Decoding), GapSVP, GapCVP, atď.
- *dobrá* báza uľahčuje riešenie
  - dobrá $\sim$ vhodne ortogonálne vektory (a prípadne vhodne krátke)
  - nájdenie dobrej bázy je ťažké

Dobrá báza, zlá báza a CVP problém



- *dobrá* modrá báza:
  - $v_1 = (3, 0), v_2 = (2, 6)$
- *zlá* červená báza:
  - $w_1 = (11, 6), w_2 = (19, 12)$
- pokus o CVP riešenie pre červený bod z
  - určme rovnobežnosten, kam bod patrí a nájdime najbližší vrchol
  - hnedý bod (pre zlú bázu) nie je správne riešenie
- SVP pre červenú bázu vyžaduje viac úsilia ako pre modrú bázu

Knapsack problém

Knapsack problém ako základ asymetrickej šifry

- subset-sum (knapsack) problém:
  - dané $r_1, \ldots, r_n \in \mathbb{Z}^+$ a $s \in \mathbb{Z}$
  - konštrukčný problém: nájsť $x_1, \ldots, x_n \in \{0,1\}$: $\sum_{i=1}^{n} x_i r_i = s$
  - rozhodovací problém: určiť, či také $x_1, \ldots, x_n \in \{0,1\}$ existujú
- NP-úplný problém
  - triviálne riešiteľné v čase $O(2^n)$ vyskúšaním všetkých možností
  - inak: môžeme hľadať kolízie medzi množinami (napr. hashtable)
    $A = \{\sum_I r_i \mid I \subseteq \{1, \ldots, \lfloor n/2 \rfloor\}\}$ a
    $B = \left\{s - \sum_J r_i \mid J \subseteq \{\lfloor n/2 \rfloor + 1, \ldots, n\}\right\}$
  - v každej množine $\approx 2^{n/2}$ prvkov, celková zložitosť $O\left(2^{n/2}\right)$

Superrastúca postupnosť

- postupnosť $r = (r_1, \ldots, r_n) \in (\mathbb{Z}^+)^n$ je *superrastúca* (SR-postupnosť), ak

$$r_k > r_{k-1} + \ldots + r_1, \text{ pre } k \geq 2$$

- ukážme, že knapsack problém je pre SR-postupnosť ľahký:
  1. Ak existuje riešenie, tak je toto riešenie jedinečné.
  2. Riešenie je možné nájsť greedy algoritmom.

Jednoznačnosť riešenia knapsack problému s SR-postupnosťou

- nech $(x_1, \ldots, x_n)$ a $(y_1, \ldots, y_n)$ sú rôzne riešenia
- nech k je najväčší index, pre ktorý $x_k \neq y_k$
  - WLOG $x_k = 1, y_k = 0$
- položme $s' = s - \sum_{i=k+1}^{n} x_i r_i = s - \sum_{i=k+1}^{n} y_i r_i$
  - $(x_1, \ldots, x_k)$ a $(y_1, \ldots, y_k)$ sú riešenia pre knapsack so sumou s'
  - $s' = r_k + \sum_{i=1}^{k-1} x_i r_i \overset{(*)}{=} \sum_{i=1}^{k-1} y_i r_i$
  - avšak $r_k > r_{k-1} + \ldots + r_1$ (SR-postupnosť), teda rovnosť $(*)$ nemôže platiť (aj keby boli všetky $y_i = 1$ a $x_i = 0$ pre $i \leq k - 1$)
- spor

Greedy algoritmus pre knapsack problém s SR-postupnosťou

- určíme x_r:
  - ak $s \geq r_n > r_{n-1} + \ldots + r_1$, tak nutne $x_n = 1$
  - ak $s < r_n$, tak $x_n = 0$
- analogicky postupujeme ďalej
- greedy algoritmus (vráti „false“ ak riešenie neexistuje):

for $i = n, \ldots, 1$:
  if $s \geq r_i$: $x_i \leftarrow 1$; $s \leftarrow s - r_i$
  else: $x_i \leftarrow 0$
if $s \neq 0$: return false

Merkleho-Hellmanova schéma

Šifrovacia schéma postavená na knapsack probléme

- schéma s verejným kľúčom: $\langle \text{Gen}, \text{Enc}, \text{Dec} \rangle$
- idea: modifikovať SR-postupnosť tak, aby výsledná postupnosť vyzerala náhodne
- súkromný kľúč umožní efektívne riešiť knapsack problém
- Merkle, Hellman (1978)

Gen:

1. zvolíme SR-postupnosť $r = (r_1, \ldots, r_n)$
2. zvolíme $m, w \in \mathbb{Z}^+$: $\text{nsd}(m, w) = 1$ a $m > \sum_{i=1}^{n} r_i$
3. vypočítame $a = (a_1, \ldots, a_n)$, kde $a_i = r_i w \bmod m$
   - triviálne $a_i \neq 0$, keďže m a w sú nesúdeliteľné

- verejný kľúč: $\text{pk} = a$
- súkromný kľúč: $\text{sk} = (m, w, r)$

Šifrovanie a dešifrovanie v Merkleho-Hellmanovej schéme

Enc

- otvorený text $x \in \{0,1\}^n$
- $\mathrm{Enc}_{\mathrm{pk}}(x) = x \cdot a$ (skalárny súčin)

Dec

- šifrový text $c \in \mathbb{Z}$
- vypočítame $s = cw^{-1} \bmod m$
- otvorený text je riešenie knapsack problému pre SR-postupnosť r a sumu s

Korektnosť:

$$s \equiv cw^{-1} \equiv \sum_{i=1}^{n} x_i a_i w^{-1}$$

$$\equiv \sum_{i=1}^{n} x_i r_i w w^{-1} \equiv \sum_{i=1}^{n} x_i r_i \mod m$$

- keďže $s \in \mathbb{Z}_m$ a $m > \sum_{i=1}^{n} r_i$, tak $s = \sum_{i=1}^{n} x_i r_i$, odkiaľ riešením knapsack problému získame x

Poznámky k schéme

- znalosť m alebo w vedie k rozbitiu schémy
  - napr. pre $r_1 = 1$ máme $a_1 = w$
  - preto má byť aj r_1 dostatočne veľké, povedzme $\sim 2^n$
- varianty:
  - pridanie súkromnej permutácie π, pričom $a = \left(a_{\pi(1)}, \ldots, a_{\pi(n)}\right)$
  - viacnásobné transformácie postupností
- keďže máme $2^{n/2}$ algoritmus pre knapsack, potrebujeme $n \geq 2k$ pre k-bitovú bezpečnosť
- prečo je schéma atraktívna
  - jednoduchosť a rýchlosť, napr. v porovnaní s RSA
  - pri šifrovaní stačí sčitovať (relatívne krátke) čísla
- nevýhody („okrem“ rozbitia viacerých variantov)
  - dlhé verejné aj súkromné kľúče, napr. v porovnaní s RSA
  - pre $n = 256$ a 512-bitové m je verejný kľúč viac ako 16 KB

Útok na Merkleho-Hellmanovu schému

- rôzne útoky na rôzne varianty knapsack schém
  - útok na pôvodnú schému – Shamir (1984)
  - polynomiálny algoritmus, ktorý nájde (m', w') vedúce k SR-postupnosti
- ukážeme kryptonalýzu pomocou riešenia problému na mriežkach
  - ilustrácia použiteľnosti mriežok, nie ako najlepší/najvhodnejší útok

Útok na Merkleho-Hellmanovu schému (2)

- uvažujme mriežku generovanú bázou (vektory bázy $v_1, \ldots, v_{n+1}$ sú riadky matice)

$$\begin{pmatrix} 1 & 0 & \ldots & 0 & -a_1 \\ 0 & 1 & \ldots & 0 & -a_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \ldots & 1 & -a_n \\ 0 & 0 & \ldots & 0 & c \end{pmatrix}$$

- nech x je riešenie pre knapsack problém (a, c): $c = x_1 a_1 + \ldots + x_n a_n$
- potom v mriežke existuje vektor $u = x_1 v_1 + \ldots + x_n v_n + v_{n+1} = (x_1, \ldots, x_n, 0)$
- dĺžka u: $\|u\| = \sqrt{x_1^2 + \ldots + x_n^2} \leq \sqrt{n}$

Útok na Merkleho-Hellmanovu schému (3)

- iné krátke vektory okrem u v mriežke?
  - závisí na *hustote* problému $n / \max_i \{\log a_i\}$
  - čím nižšia hustota, tým väčšia pravdepodobnosť úspechu
- použitie algoritmov na riešenie SVP problému – nájdenie u (demo)
- niekedy sú nájdené vektory kratšie ako u ale s nesprávnymi koeficientami
  - potrebujeme koeficienty z $\{0, 1\}$
- LLL algoritmus
  (A. Lenstra, H. Lenstra, L. Lovász)
  - redukcia bázy – LLL nájde *lepšiu* bázu pre mriežku
  - aproximácia najkratšieho vektora
  - v praxi často lepšie výsledky ako v teórii
- BKZ algoritmus
  (Block KZ; A. Korkine, Y. Zolotarev)
  - iný algoritmus na redukciu bázy