PROBLEM UČENJA Z NAPAKAMI IN SODOBNI KRIPTOSISTEMI

TILEN MARC

Fakulteta za matematiko in fiziko
Univerza v Ljubljani

Math. Subj. Class. (2020): 94A60, 68P25, 81P94

Sodobni kriptosistemi so osnovani na matematičnih problemih in njihova varnost je zagotovljena samo, dokler ne obstajajo algoritmi, ki bi te probleme učinkovito rešili. V članku predstavimo nedavno vpeljan algoritmičen problem učenja z napakami, ki se izkaže za izjemno uporabnega v kriptografiji, saj omogoča sestavo novih kriptosistemov z zanimivimi in uporabnimi lastnostmi. Taki kriptosistemi veljajo tudi za varne pred nasprotniki, ki imajo dostop do kvantnega računalnika, kar za večino drugih ne velja. Predstavljena sta kvantno varen kriptosistem z javnim ključem in kriptosistem, ki omogoča računanje na šifriranih podatkih, kar je znano pod imenom homomorfno šifriranje. Konstrukcija slednjega je bila odprt problem več desetletij in dosežena šele s pomočjo problema učenja z napakami.

PROBLEM OF LEARNING WITH ERRORS AND MODERN CRYPTOSYSTEMS

Modern cryptosystems are based on mathematical problems and their security is guaranteed only as long as there are no efficient algorithms solving these problems. We present a recently introduced algorithmic problem of learning with errors (LWE). The problem is crafted for the use in cryptography and allows to construct new cryptosystems with interesting and useful properties. Such cryptosystems are considered safe even against adversaries with access to quantum computers, which does not hold for most of the other systems. We explain how to construct two cryptosystems: a quantumly secure cryptographic scheme with public key, and a scheme that enables computation on encrypted data, known as homomorphic encryption. The construction of the latter was a long standing open problem and was solved only recently with the help of LWE problem.

Uvod

Problem učenja z napakami (ang. *learning with errors – LWE*) je vpeljal Regev v [4] kot nov algoritmičen problem in dokazal, da je vsaj tako težek kot nekateri drugi znani problemi. Članek je povzročil pravo revolucijo, saj je bilo v zadnjem desetletju napisanih na tisoče znanstvenih člankov, ki temeljijo na problemu LWE. Regev je bil leta 2018 za svoj prispevek nagrajen s prestižno Gödelovo nagrado, ki jo vsako leto podelijo za doprinos k teoretičnemu računalništvu.

Glavni razlog za priljubljenost problema LWE je njegova uporabnost v kriptografiji. Eden izmed osnovnih temeljev sodobne kriptografije je t. i. asimetrična kriptografija. Ta omogoča, da uporabnik izračuna svoj javni

ključ, s pomočjo katerega mu lahko vsakdo pošlje šifrirano sporočilo, ki ga lahko dešifrira le lastnik javnega ključa s pomočjo svojega skrivnega ključa. Taki kriptosistemi se dandanes uporabljajo v računalniški komunikaciji, bančništvu in praktično na vseh področjih, kjer je tajnost podatkov pomembna. Varnost asimetrične kriptografije temelji na predpostavki, da iz javnega ključa ne moremo izračunati skrivnega, saj bi v nasprotnem primeru lahko vsak dešifriral sporočila. Zato so kriptosistemi z javnim ključem osnovani na matematičnih problemih, za katere ne poznamo učinkovitih algoritmov za reševanje.

V praksi daleč najbolj uporabljana matematična problema sta t. i. problem razcepa naravnega števila in z njim povezan problem diskretnega logaritma. Problema, ki ju lahko uvrstimo na področje teorije števil, sta omogočila slavne kriptografske sheme, kot so RSA, ElGamalov sistem in DSA, pa tudi sodobnejše konstrukte, ki temeljijo na eliptičnih krivuljah. Največjo grožnjo vsem omenjenim tehnologijam predstavlja razvoj t. i. kvantnega računalnika in kvantnih algoritmov. Glavni razlog je, da je Shor že leta 1999 v [6] opisal kvantni algoritem, ki zmore v polinomskem času razbiti naravno število in poiskati diskretni logaritem. Torej obstaja algoritem, s katerim lahko s pomočjo kvantnega računalnika razbijemo skoraj vsak dandanes uporabljan kriptosistem z javnim ključem, vdremo v bančne račune itd. V trenutku pisanja članka kvantni računalniki že obstajajo, vendar je njihova zmogljivost še zelo omejena. Ali se bo to v prihodnje spremenilo, lahko samo ugibamo.

Problem LWE ne temelji na zgoraj omenjenih problemih in je zaenkrat varen pred kvantnimi računalniki, saj (trenutno) ne poznamo kvantnega algoritma, ki bi ga v polinomskem času uspel rešiti. Še več – kot bomo videli, je Regev dokazal, da je problem LWE vsaj tako težek kot nekateri problemi na t. i. rešetkah. Ti so poznani že dlje časa in veljajo za težke, zato je zaupanje v neobstoj polinomskih (kvantnih) algoritmov za reševanje problema LWE še večje.

V članku bomo predstavili enega izmed načinov, kako je mogoče sestaviti kriptosistem z javnim ključem, katerega varnost temelji na težavnosti problema LWE. Tak kriptosistem torej velja za kvantno varnega, razvoj tovrstnih kriptosistemov pa je ena izmed najpomembnejših tem sodobne kriptografije. To dokazuje tudi dejstvo, da v času pisanja članka poteka pomemben izbor ameriškega Nacionalnega inštituta za standarde in tehnologijo (NIST) [7] za določitev kvantnih kriptografskih standardov, ki se bodo uporabljali v bližnji prihodnosti. Večina shem, ki so se uvrstile v ožji izbor, temelji na problemu LWE ali njegovih izpeljankah.

Problem LWE poleg kvante varnosti kriptosistemov z javnim ključem prinaša tudi druge novosti. Tako lahko sestavimo nove kriptosisteme, katerih varnost temelji na težavnosti problema LWE, z lastnostmi, ki jih starejši

kriptosistemi ne omogočajo. Področje simetrične kriptografije omogoča šifriranje podatkov s pomočjo skrivnega ključa, tako da jih lahko dešifriramo samo ob poznavanju le-tega. Taki sistemi se uporabljajo za hitro prenašanje sporočil med uporabniki, ki so si izmenjali skrivni ključ, pa tudi za varno hrambo podatkov. V sodobnem svetu veliko podatkov hranimo v oblaku, in če želimo ohraniti osebne podatke varne, morajo biti taki podatki šifrirani. Poleg hrambe ponudniki omogočajo tudi urejanje, analizo, napovedi, priporočila in druge storitve, ki temeljijo na računanju s podatki. A če so podatki šifrirani, take storitve niso mogoče, saj ponudnik storitev v oblaku podatkov ne pozna. Želeli bi torej na videz nemogoče – računanje (in s tem vse storitve, ki smo jih vajeni) brez poznavanja podatkov. Čeprav opisano zveni kot sodobni problem, je bil izziv sestaviti kriptosistem, ki bi omogočal računanje s šifriranimi podatki, brez poznavanja le-teh, predlagan že leta 1978 v [5]. Tak kriptosistem imenujemo homomorfno šifriranje. Problem je bil dolgo časa odprt, dokler ni prvi tak sistem opisal Gentry v [3]. Objavljen kriptosistem je bil osnovan na problemu LWE in od njegove objave je bilo predlaganih več različic in izboljšav. Eno izmed teh bomo predstavili v članku.

Preostanek članka je strukturiran na naslednji način: v razdelkih Problem LWE in Težavnost problema LWE definiramo problem LWE in pokažemo, zakaj je to težek problem, v razdelku Kriptosistem z javnim ključem, osnovan na problemu LWE predstavimo kriptosistem z javnim ključem, ki temelji na problemu LWE, in ga nato v razdelku Homomorfno šifriranje nadgradimo v kriptosistem za homomorfno šifriranje. Da lahko to storimo, konstruiramo t. i. generator psevdonaključnih vektorjev s stranskimi vrati in razložimo, kako uporabiti dvojiško kodiranje pri konstrukciji.

Problem LWE

Začnimo s preprostim matematičnim problemom reševanja linearnih enačb. Naj bo p praštevilo in z $\mathbb{Z}_p$ označimo polje s p elementi. Torej, elementi $\mathbb{Z}_p$ so števila $\{0, 1, \ldots, p-1\}$, operaciji seštevanja in množenja pa izvajamo po modulu p. Naj bo $A \in \mathbb{Z}_p^{m \times n}$ poljubna matrika in $b \in \mathbb{Z}_p^m$ vektor za $m \geq n \geq 1$. Problem iskanja $x \in \mathbb{Z}_p^n$ v matrični enačbi nad $\mathbb{Z}_p$

$$Ax = b$$

ni težek, saj ga lahko rešimo z Gaussovo eliminacijo, četudi je p velik.

Otežimo zgornji problem z dodajanjem napake. Za vrednost $y \in \mathbb{Z}_p$ označimo $|y| = \min(y, p-y)$ in recimo, da je vrednost $y \in \mathbb{Z}_p$ *majhna*, če je vrednost $|y| = \min(y, p-y)$ občutno manjša od p: za vse uporabe v članku lahko bralec predpostavi, da je $|y|$ manjši od $\sqrt{p}$. Naj bo $e \in \mathbb{Z}_p^m$ vektor z majhnimi vrednostmi, torej je vrednost vsake komponente v e občutno

manjša od p. Naj bosta $A \in \mathbb{Z}_p^{m\times n}$ in $b \in \mathbb{Z}_p^m$ dana kot prej. Iskanje vrednosti x, da velja

$$Ax + e = b,$$

kjer je e neznan, je osnova problema učenja z napakami (LWE).

Oglejmo si zgornji problem nekoliko natančneje. Na prvi pogled je problem enostavnejši, saj imamo m linearnih enačb in $m+n$ neznank – neznana sta tako x kot e. V primeru, da je $m = n$, lahko izberemo $e = 0$ in rešimo $Ax = b$, kot v zgornjem primeru z Gaussovo eliminacijo. Problem postane težji, če je m večji kot n, saj nismo zadovoljni z vsako rešitvijo, ampak samo s tistimi, v katerih je e majhen. Reševanje problema s standardnimi metodami linearne algebre nam ne zagotavlja take rešitve. Na problem lahko pogledamo tudi drugače: podprostor vseh vektorjev $\{Ay \mid y \in \mathbb{Z}_p^n\}$ je največ n-dimenzionalni podprostor m-dimenzionalnega prostora $\mathbb{Z}_p^m$. Če želimo najti x, da velja $Ax + e = b$, potem moramo najti tak majhen e, da je $b - e \in \{Ay \mid y \in \mathbb{Z}_p^n\}$.

Definirajmo sedaj iskalni problem LWE natančneje. Da bo problem uporaben v kriptografiji, mora biti take narave, da lahko enostavno generiramo naključne vrednosti (recimo nove skrite in javne ključe), za katere je problem težek. Zato problem definiramo za naključne vrednosti, izbrane iz vnaprej določenih porazdelitev. Torej problema LWE ne bomo definirali za poljubne vrednosti A, x, e, b, ampak za naključne, kar je precej drugače od mnogih drugih algoritmičnih problemov, ki jih rešujemo za poljubne vhodne podatke. Razlog za tako definicijo je, da je marsikateri (tudi NP-poln) problem težek, če so vhodni podatki poljubni, vendar lahek za naključne.

Označimo s χ porazdelitev naključnih majhnih vektorjev, izbranih iz $\mathbb{Z}_p$ (v naslednjem razdelku bomo natančneje definirali, kako izbrati χ). Za vrednost (vektor, matriko itd.) x bomo rekli, da je izbrana enakomerno naključno, če so verjetnosti izbire vseh mogočih vrednosti enake.

Definicija 1. Naj bo p praštevilo, $m \geq n \geq 1$, $A \in \mathbb{Z}_p^{m\times n}$ enakomerno naključno izbrana matrika, $x \in \mathbb{Z}_p^n$ enakomerno naključno izbran vektor in $e \in \mathbb{Z}_p^n$ vektor, naključno izbran iz porazdelitve χ. Definirajmo $b \in \mathbb{Z}_p^m$ kot $b := Ax + e$. *Iskalni problem* $LWE_{n,m,p,\chi}$ je problem najti x, če poznamo samo A in b.

Pozoren bralec bi lahko opazil, da ima sistem $b = Ax+e$ lahko več rešitev. Spomnimo se – najti moramo tak majhen e, da je $b-e \in \{Ay \mid y \in \mathbb{Z}_p^n\}$. Če določimo parametre tako, da je m občutno večji od n, je po številu elementov prostor $\{Ay \mid y \in \mathbb{Z}_p^n\}$ občutno manjši od celega prostora. Intuicija nam zato pravi, da je verjetnost (za naključno izbrane A, x, e), da ima enačba več kot eno rešitev, majhna. Kot bomo videli, je za kriptografsko uporabo pomembno le, da je problem tako težek, da ne moremo najti nobene rešitve x, e, kjer je e majhen.

Težavnost problema LWE

V tem razdelku bomo definirali odločitveno verzijo problema LWE, natančneje določili porazdelitev χ, ki se uporablja za izbiro vektorja e, in predstavili pomemben rezultat glede težavnosti problema LWE.

Odločitveni problem LWE

Izkaže se, da je za kriptografsko uporabo prikladno, da definiramo *odločitveno* verzijo problema LWE.

Definicija 2. *Odločitveni problem* je problem, ki ima le dve možni rešitvi: 0 ali 1.

Torej, če želimo rešiti odločitveni problem, želimo izpeljati algoritem, ki za vhodne podatke vrne odgovor 0 ali 1. V primeru odločitvenega problema LWE je naloga razlikovati med vrednostmi, izbranimi iz dveh različnih porazdelitev.

Definicija 3. Naj bo p praštevilo, $m \geq n \geq 1$, $A \in \mathbb{Z}_p^{m \times n}$ enakomerno naključno izbrana matrika, $x \in \mathbb{Z}_p^n$ enakomerno naključno izbran vektor in $e \in \mathbb{Z}_p^m$ vektor, naključno izbran iz porazdelitve χ. Naj bo $b \in \mathbb{Z}_p^m$ definiran kot $b := Ax + e$ in $c \in \mathbb{Z}_p^m$ izbran iz enakomerne porazdelitve. *Odločitveni problem* $LWE_{n,m,p,\chi}$ je problem, katerega vhod je ena izmed vrednosti (A, b) ali (A, c), algoritem pa mora določiti, katera izmed vrednosti je bila vhod. Torej, algoritem prejme vrednost (A, z) in mora določiti, ali je vektor z vektor vrednosti enakomerne porazdelitve ali je z dobljen iz porazdelitve enačbe LWE.

Za razliko od iskalnega problema tukaj naloga ni, da dobimo (A, z) in poiščemo x, e, da velja $Ax + e = z$, ampak da lahko razločimo, ali je bil z naključno izbran prek izbire A, x, e, ali je preprosto vektor enakomerno naključnih vrednosti. Ti dve porazdelitvi seveda nista enaki; recimo, vektor z, izbran iz prve porazdelitve, je vedno tak, da obstaja rešitev enačbe $Ax + e = z$, kjer je e majhen, medtem ko za enakomerno naključen vektor z rešitev morda (in tudi zelo verjetno) ne obstaja. Izkaže se, da sta iskalni in odločitveni problem podobne težavnosti, saj lahko prevedemo enega na drugega [4]. Za uporabo v kriptografiji bi želeli, da je odločitveni problem LWE (in zato tudi iskalni) težek, torej da algoritem, ki bi deloval dovolj hitro in rešil problem pravilno v več primerih kot napačno, ne obstaja.

Definirajmo matematično, kaj pravzaprav mislimo, ko rečemo, da je odločitveni problem $\mathrm{LWE}_{n,m,p,\chi}$ težek. Označimo z $\Pr(X)$ verjetnost dogodka X. Za algoritem $\mathcal{A}$, ki rešuje odločitveni problem LWE, označimo z $\mathcal{A}(A, z)$ vrednost, ki jo vrne $\mathcal{A}$ ob vhodu (A, z).

Definicija 4. Naj bo $n > 0$ parameter in naj bo $m \geq n$, p praštevilo in χ porazdelitev, ki so lahko določeni v odvisnosti od n. Naj bosta (A, b) in (A, c) para, kjer je $b \in \mathbb{Z}_p^m$ definiran kot $b := Ax + e \in \mathbb{Z}_p^m$ in $c \in \mathbb{Z}_p^m$ enakomerno naključen, torej sta (A, b) in (A, c) generirana kot možna vhoda odločitvenega problema $\mathrm{LWE}_{n,m,p,\chi}$. *Predpostavka LWE* pravi, da za vsak polinomski algoritem $\mathcal{A}$, ki vrača 0 ali 1, velja

$$|\Pr(\mathcal{A}(A, b) = 1) - \Pr(\mathcal{A}(A, c) = 1)| < 2^{-Cn}$$

za neko konstanto $C > 0$.

Predpostavka LWE pravi, da ne obstaja algoritem, ki bi mu v polinomskem času uspelo razlikovati (torej vrniti 0 ali 1) med naključnima vhodoma odločitvenega problema $\mathrm{LWE}_{n,m,p,\chi}$ bolje kot z eksponentno majhno verjetnostjo. Intuitivno to v neformalnem jeziku pomeni, da če predpostavka LWE drži in izračunamo $b = Ax + e$ z naključno izbranimi A, x, e, potem se b zdi kot enakomerno naključen vektor. Če predpostavka LWE drži, potem bomo rekli, da je $b = Ax + e$ *računsko neločljiv* od enakomerno naključnega vektorja.

Definicija vsebuje verjetnosti, da algoritem vrne pravilno ali napačno odločitev, saj je problem definiran na naključnih podatkih in bi lahko naključno sprejemal odločitve. Da bi bolje razumeli definicijo, si oglejmo preveč preprost algoritem za reševanje odločitvenega problema LWE. Recimo, da algoritem $\mathcal{B}$ problem reši tako, da preprosto vrže kovanec in izbere 0 ali 1. Seveda tak algoritem ne bi bil preveč uporaben. Oglejmo si vrednost:

$$|\Pr(\mathcal{B}(A, b) = 1) - \Pr(\mathcal{B}(A, c) = 1)| = \left|\frac{1}{2} - \frac{1}{2}\right| = 0.$$

Slednje lahko razumemo tako, da tak algoritem ne bi uspel razlikovati pravilne odločitve od napačne. Če predpostavka LWE drži, potem vsak algoritem, ki bi deloval v polinomskem času, ne bi deloval veliko bolje, kot da vržemo kovanec.

V nadaljevanju razdelka bomo razložili, za kakšne parametre n, m, p, χ se verjame, da predpostavka LWE drži.

Diskretna Gaussova porazdelitev

Do sedaj smo vprašanje, kako izbrati m, p, χ, da bo predpostavka LWE smiselna, pustili odprto. V tem podrazdelku bomo opisali, kako izbrati porazdelitev χ, s katero izberemo vektor e v $Ax + e$.

Definicija 5. *Diskretna Gaussova porazdelitev* D_σ, ki ima vrednosti v $\mathbb{Z}$, je taka porazdelitev, da je za vsak $x \in \mathbb{Z}$ verjetnost izbire x enaka $Ce^{-\frac{x^2}{2\sigma^2}}$,

kjer je $\sigma > 0$ in C taka konstanta, da se verjetnosti vseh dogodkov seštejejo v 1.

Z $\bar{D}_\sigma$ označimo diskretno Gaussovo porazdelitev, ki ima vrednosti v $\mathbb{Z}_p$, ki jih dobimo tako, da izberemo $y \in \mathbb{Z}$ iz D_σ in izračunamo $\bar{y} = (y \bmod p) \in \mathbb{Z}_p$.

Verjetnost izbire vrednosti iz D_σ, katerih absolutna vrednost je večja od σ, eksponentno hitro pada. Če je σ občutno manjši kot praštevilo p, bodo tudi absolutne vrednosti števil, generiranih z D_σ, z zelo veliko verjetnostjo občutno manjše od p. To pomeni, da bodo vrednosti y, generirane z $\bar{D}_\sigma$, majhne. Torej, vrednost $\min(y, p - y)$ bo občutno manjša od p. Take vrednosti želimo v problemu LWE.

Problemi na rešetkah

V tem podrazdelku bomo odgovorili, kako izbrati preostale parametre problema $\mathrm{LWE}_{n,m,p,\chi}$ s pomočjo v uvodu omenjenega rezultata Regeva [4]. Najprej potrebujemo dve definiciji:

Definicija 6. Naj bodo $v_1, \ldots, v_n$ linearno neodvisni vektorji v $\mathbb{R}^n$. Množico

$$L(v_1, \ldots, v_n) = \{a_1 v_1 + \cdots + a_n v_n \mid \ a_i \in \mathbb{Z} \text{ za } 1 \leq i \leq n\}$$

imenujemo *rešetka*.

Spomnimo se, da je 2-norma vektorja x definirana kot $||x|| = \sqrt{x_1^2 + \cdots + x_n^2}$, kjer so $x_1, \ldots, x_n$ koordinate x. Recimo, da imamo podane vektorje $v_1, \ldots, v_n$, ki imajo vsi 2-normo večjo od nekega $a \in \mathbb{R}$. S sestavljanjem linearnih kombinacij $a_1 v_1 + \cdots + a_n v_n$, z $a_i \in \mathbb{Z}$ za vsak $1 \leq i \leq n$, je v določenih primerih možno sestaviti neničelen vektor, ki ima normo manjšo od a. Kot preprost primer navedimo $v_1 = (100, 99), v_2 = (100, 100)$, kjer ima $-v_1 + v_2 = (0, 1)$ manjšo normo od začetnih vektorjev.

Če za dane $v_1, \ldots, v_n$ lahko najdemo koeficiente $a_i \in \mathbb{Z}$, da je $a_1 v_1 + \cdots + a_n v_n$ neničeln vektor z majhno normo, potem velja, da v rešetki $L(v_1, \ldots, v_n)$ obstaja element z majhno normo. To, kako učinkovito najti take koeficiente oziroma se odločiti, ali sploh obstajajo, je pomemben algoritmičen problem.

Definicija 7. *Odločitveni problem GapSVP(n, s)*, kjer je $n \in \mathbb{N}$ in $s \geq 1$, je problem, katerega vhod so poljubni neodvisni vektorji $v_1, \ldots, v_n$, ki definirajo rešetko $L(v_1, \ldots, v_n)$, in pravilni izhod je vrednost 1, če obstaja $v \in L(v_1, \ldots, v_n) \backslash \{(0, \ldots, 0)\}$ z $||v|| \leq 1$, in 0, če za vsak $v \in L(v_1, \ldots, v_n) \backslash \{(0, \ldots, 0)\}$ velja $||v|| > s$. Če noben izmed pogojev ne velja, je izhod lahko poljuben.

Opomba: Ime GapSVP je bilo izbrano, saj v problemu odločamo o obstoju najkrajšega neničelnega vektorja (ang. *shortest vector problem*) z razmakom (ang. *gap*) s. Natančneje, treba je določiti, ali v rešetki obstaja kratek neničeln vektor z normo največ 1, ali pa so vsi neničelni vektorji daljši od s. Večji kot je s, lažji je problem.

Z naslednjim izrekom je Regev pokazal, da sta problema $\mathrm{LWE}_{n,m,p,\chi}$ in $\mathrm{GapSVP}(n,s)$ povezana.

Izrek 8 ([4]). *Naj bo $n \in \mathbb{N}$ in p, m, α vrednosti, ki so odvisne od n, da velja: p je praštevilo, $m \in \mathbb{N}$ ne več kot polinomsko večji od n in $\alpha \in (0,1)$ tak, da je $\sigma := \alpha p > 2\sqrt{n}$. Če obstaja algoritem, ki za vsak n reši odločitveni problem $LWE_{n,m,p,\bar{D}_\sigma}$ v polinomskem času (v parametru n), potem obstaja kvantni polinomski algoritem, ki reši problem GapSVP$(n, n/\alpha)$.*

Izrek pravi, da je problem $\mathrm{LWE}_{n,m,p,\bar{D}_\sigma}$ vsaj tako težek, kot je problem $\mathrm{GapSVP}(n, n/\alpha)$, če imamo dostop do kvantnega računalnika. Ker je problem $\mathrm{GapSVP}(n,s)$ že dlje časa preučevan in ne poznamo kvantnega algoritma, ki bi rešil $\mathrm{GapSVP}(n, n/\alpha)$ v polinomskem času, izrek vliva upanje, da polinomskega algoritma za reševanje $\mathrm{LWE}_{n,m,p,\bar{D}_\sigma}$ ni.

Zgornji rezultat nam torej zagotavlja, da vsi znani algoritmi za reševanje $\mathrm{LWE}_{n,m,p,\bar{D}_\sigma}$ s parametri, izbranimi tako, da zadoščajo pogojem iz izreka 8, potrebujejo eksponentno mnogo korakov (glede na n), da rešijo problem. Za praktično uporabo v kriptografiji bi želeli konkretne parametre: npr. na podlagi natančne analize znanih algoritmov za reševanje problemov na rešetkah v [1] ocenjujejo, da je za parametre $n = 256$, p 16-bitno praštevilo in $\sigma \approx 26$ časovna zahtevnost reševanja ustreznega problema LWE vsaj 2^{153}.

Kriptosistem z javnim ključem, osnovan na problemu LWE

Priljubljenost problema LWE izhaja iz dejstva, da omogoča konstrukcijo novih kriptografskih shem, katerih varnost temelji na težavnosti reševanja odločitvenega problema LWE. Prvo tako shemo je opisal že Regev v članku [4].

Naj bo $n > 1$ parameter, ki določa varnost, p praštevilo, da velja $n^2 \leq p \leq 2n^2$, $m = (1+\epsilon)(n+1)\log(p)$ za neko malo konstanto $\epsilon > 0$ (izbrano tako, da je $m \in \mathbb{N}$) in naj bo $\chi = \bar{D}_\sigma$, torej diskretna Gaussova porazdelitev nad $\mathbb{Z}_p$, kjer je $\sigma = p/(\sqrt{n}\log^2(n))$. Izbrani parametri ustrezajo pogojem izreka 8, tako da bi rešitev odločitvenega problema $\mathrm{LWE}_{n,m,p,\chi}$ pomenila velik preboj na področju reševanja problemov na rešetkah.

Opišimo kriptosistem z javnim ključem, ki temelji na problemu LWE. Vse aritmetične operacije v shemi opravimo po modulu p, torej v $\mathbb{Z}_p$:

- **Generiranje ključev:** Naj bo $s \in \mathbb{Z}_p^n$ enakomerno naključen vektor in $A \in \mathbb{Z}_p^{m \times n}$ enakomerno naključna matrika. Naj bo $e \in \mathbb{Z}^m$ vektor, katerega komponente so izbrane naključno iz porazdelitve $\bar{D}_\sigma$. Izračunajmo $b = As + e$. Potem je vektor s skrivni ključ in par (A, b) javni ključ kriptosistema.

- **Šifriranje:** Naj bo $M \in \{0, 1\}$ sporočilo, ki ga želimo šifrirati. Če poznamo javni ključ (A, b), lahko izberemo vektor $r \in \{0, 1\}^m$ enakomerno naključno in izračunamo

$$C = (c_0, c_1) = (r^T A, r^T b + M \cdot \lfloor p/2 \rfloor),$$

  ki ga obravnavamo kot šifriran M.

- **Dešifriranje:** S pomočjo skrivnega ključa s izračunamo $d = c_1 - c_0 s$, kar dešifriramo kot vrednost 0, če je d bližje 0, in 1, če je bližje $\lfloor p/2 \rfloor$.

Zgoraj opisani kriptosistem omogoča šifriranje sporočila $M \in \{0, 1\}$, torej enega bita. Če želimo poslati daljše sporočilo (256-bitno sporočilo je v praksi pogost primer), potem preprosto pošljemo več različnih sporočil, pri čemer za vsako izberemo nov naključen r. Slednje je računsko zahtevnejše kot šifriranje s pomočjo standardnih kriptosistemov z javnim ključem, vendar praktično izvedljivo s sodobnimi računalniki.

Vsak kriptosistem mora izpolnjevati dve zahtevi: po pravilnosti in varnosti. Za kriptosistem pravimo, da *deluje pravilno*, če je dešifrirano sporočilo enako prvotnemu sporočilu. Za določitev varnosti kriptosistema obstaja več matematičnih definicij. Najmanjša zahteva je, da napadalec samo iz kriptograma in javnih parametrov ne sme biti sposoben dešifrirati sporočila. Vendar to za sodobne standarde varnosti ne zadošča več: želimo, da napadalec iz kriptograma in javnih parametrov ne more izvedeti (skoraj) ničesar. Temu se približa naslednja definicija. Za kriptosistem pravimo, da je *IND-CPA* varen (ang. *indistinguishability of ciphertext*), če napadalec, ki ne pozna skrivnega ključa, na podlagi javnih parametrov in kriptograma C ne more razlikovati med dvema besediloma enake dolžine, ki sta bili šifrirani (v našem primeru, ali je bil šifriran bit $M = 0$ ali $M = 1$).

Skicirajmo, zakaj je kriptosistem, osnovan na problemu LWE, varen. Več podrobnosti o dokazu lahko bralec najde v [4].

Izrek 9. *Zgoraj opisan kriptosistem za dovolj velik n deluje pravilno – torej dešifrirana vrednost ustreza šifrirani – ter je IND-CPA varen pri predpostavki LWE s parametri n, m, p, χ.*

Skica dokaza. Dokažimo, da sistem pravilno dešifrira vrednosti:

$$d = c_1 - c_0 s = r^T b + M \cdot \lfloor p/2 \rfloor - r^T A s = r^T (As + e) + M \cdot \lfloor p/2 \rfloor - r^T A s$$
$$= M \cdot \lfloor p/2 \rfloor + r^T e.$$

Omejimo vrednost $r^T e$. Komponente e so izbrane iz porazdelitve $\bar{D}_\sigma$, kjer je $\sigma = p/(\sqrt{n}\log^2(n))$. Naj bodo $i_1, \ldots, i_k$, $k \leq m$ indeksi komponent vektorja r, ki so neničelni. Potem je $r^T e = \sum_{j=1}^{k} e_{i_j}$. Slednje je vsota neodvisnih diskretnih Gaussovih porazdelitev, kar ima porazdelitev računsko neločljivo diskretni Gaussovi porazdelitvi s standardno deviacijo $\sqrt{k}\sigma \leq \sqrt{m}\sigma \leq \sqrt{2n\log(2n^2)}p/(\sqrt{n}\log^2(n)) = \alpha p$, kjer je α v $O(1/\log(n))$. Za dovolj velik n je verjetnost, da je vrednost iz porazdelitve $\bar{D}_{p/\log(n)}$ večja od $p/4$, zanemarljivo majhna. Torej je $d = M \cdot \lfloor p/2 \rfloor + r^T e$ bližje $\lfloor p/2 \rfloor$ kot 0, če je $M = 1$, in bližje 0, če je $M = 0$. Sledi, da je dešifriranje pravilno.

Skicirajmo še, zakaj je kriptosistem varen. Pokazati moramo, da napadalec, ki ne pozna skrivnega ključa, na podlagi javnih parametrov in kriptograma ne more razlikovati, ali je bil šifriran bit $M = 0$ ali $M = 1$. Oglejmo si kriptogram $C = (r^T A, r^T b + M \cdot \lfloor p/2 \rfloor)$. Če velja predpostavka LWE, je za napadalca vrednost b neločljiva od enakomerno naključnega vektorja iz $\mathbb{Z}_p^m$. Tako imenovana »leftover hash lemma« [4] pravi, da je potem porazdelitev $r^T b$ za enakomerno naključen $r \in \{0,1\}^m$ statistično zelo blizu enakomerne porazdelitve. Enako velja za $r^T A$; napadalec torej ne more razločiti vrednosti $(r^T A, r^T b)$ od enakomerno naključno izbranih vrednosti. Potem je tudi porazdelitev $(r^T A, r^T b + M \cdot \lfloor p/2 \rfloor)$ za napadalca neločljiva od enakomerno naključne in ne more razločiti, ali je šifriran bit $M = 0$ ali $M = 1$. ∎

Homomorfno šifriranje

V prejšnjem razdelku smo spoznali, kako sestaviti kvantno varen kriptosistem z javnim ključem s pomočjo problema LWE. V tem razdelku pa bomo pokazali, da se tak kriptosistem z nekaj truda da nadgraditi v t. i. sistem homomorfnega šifriranja. Osnovna naloga klasičnih kriptosistemov je, da omogočajo varen prenos oziroma hrambo podatkov. Homomorfno šifriranje poleg slednjega omogoča še varno računanje na šifriranih podatkih.

V naslednjih podrazdelkih bomo natančneje spoznali homomorfno šifriranje in njegove uporabe. V podrazdelku Aditivno šifriranje bomo dokazali, da že kriptosistem, opisan v razdelku Kriptosistem z javnim ključem, osnovan na problemu LWE, vsebuje nekatere lastnosti homomorfnega šifriranja. Podrazdelka Generator psevdonaključnih vektorjev s stranskimi vrati in Dvojiško kodiranje sta tehnične narave, saj predstavita vse potrebno za nadgradnjo v kriptosistem za homomorfno šifriranje, ki je opisan v podrazdelku Homomorfno šifriranje.

Aditivno šifriranje

Kriptosistem, opisan v prejšnjem razdelku, ima zanimivo lastnost. Recimo, da z istim javnim ključem (A, b) šifriramo dve vrednosti: $m_1, m_2 \in \{0, 1\}$, tj.

$$(c_0^1, c_1^1) = (r_1^T A, r_1^T b + m_1 \cdot \lfloor p/2 \rfloor), \quad (c_0^2, c_1^2) = (r_2^T A, r_2^T b + m_2 \cdot \lfloor p/2 \rfloor).$$

Potem lahko brez poznavanja skrivnega ključa izračunamo:

$$(c_0, c_1) = (c_0^1, c_1^1) + (c_0^2, c_1^2) = ((r_1 + r_2)^T A,\ (r_1 + r_2)^T b + (m_1 + m_2) \cdot \lfloor p/2 \rfloor).$$

Opazimo, da lahko (c_0, c_1) obravnavamo kot šifro in jo poskusimo dešifrirati s pomočjo skrivnega ključa s. Dobimo:

$$\begin{aligned} d = c_0 s - c_1 &= (r_1 + r_2)^T (As + e) + (m_1 + m_2) \cdot \lfloor p/2 \rfloor) - (r_1 + r_2)^T As \\ &= (r_1 + r_2)^T e + (m_1 + m_2) \cdot \lfloor p/2 \rfloor). \end{aligned}$$

Če so vrednosti e dovolj majhne v primerjavi s p (kar pri kriptosistemu zahtevamo), potem je vrednost d bližje 0 kot $\lfloor p/2 \rfloor$, natanko tedaj, ko je $m_1 + m_2 = 0$ ali pa $m_1 + m_2 = 2$. Torej, par (c_0, c_1) ustreza kodiranemu sporočilu $m_1 + m_2 \mod 2$. Ali z drugimi besedami, dobimo šifrirano XOR-operacijo bitov m_1 in m_2.

Recimo, da imamo podatke, podane z biti $m_1, \ldots, m_k$, in bi želeli izračunati neko funkcijo $f(m_1, \ldots, m_k)$. Omejimo se na funkcije, ki se jih da opisati z zaporedno uporabo operacije XOR. Potem lahko podatke šifriramo in jih pošljemo ponudniku računanja v oblaku, ki nam pretvori šifrirane podatke v šifrirano vrednost $f(m_1, \ldots, m_k)$, ne da bi kakorkoli poznal podatke. Tako lahko računanje funkcij prenesemo na ponudnika takih storitev, brez ogrožanja osebnih podatkov.

Seveda so funkcije, ki jih lahko opišemo samo z uporabo vrat XOR, zelo omejene, pravzaprav neuporabne. Želeli bi kriptosistem, ki omogoča računanje poljubnih funkcij na šifriranih podatkih. Osnovna teorija računanja nam zagotavlja, da lahko poljubno funkcijo izrazimo z zaporedno uporabo NAND-vrat (znana tudi kot Shefferjev veznik). V naslednjih razdelkih bomo predstavili kriptosistem, ki omogoča seštevanje in množenje šifriranih podatkov in s tem uporabo NAND-vrat.

Generator psevdonaključnih vektorjev s stranskimi vrati

Problem LWE nam omogoča sestaviti nepričakovano orodje, ki ga bomo uporabili v shemi za homomorfno šifriranje. Tako kot prej naj bo $m \geq n \geq 1$, $A \in \mathbb{Z}_p^{m \times n}$ enakomerno naključno izbrana matrika, $x \in \mathbb{Z}_p^n$ enakomerno naključno izbran vektor in $e \in \mathbb{Z}^n$ vektor z majhnimi vrednostimi, naključno

izbran iz porazdelitve χ. Naj bo $b = Ax + e$ in naj bo $\beta = \lfloor p/2 \rfloor^{-1}$, torej inverz $\lfloor p/2 \rfloor$ v $\mathbb{Z}_p$. definirajmo matriko $B \in \mathbb{Z}_p^{m \times (n+1)}$, katere prvi stolpec je $-\beta b$, ostali stolpci pa ustrezajo A. Prav tako definirajmo $s \in \mathbb{Z}_p^{n+1}$, katerega prva komponenta je $\lfloor p/2 \rfloor$, ostale komponente pa ustrezajo x.

Predpostavimo, da je odločitveni problem $\mathrm{LWE}_{n,m,p,\chi}$ težek; npr. izberimo vrednosti $n, m, p\chi$ tako, da je problem vsaj tako težek kot problem na rešetkah, opisan v izreku 8. Potem nihče, ki ne pozna s, ne more razlikovati matrike B od enakomerno naključno generirane matrike, saj je b (in zato tudi $-\beta b$) neločljiv od enakomerno naključnega vektorja. Po drugi strani, če poznamo s, lahko izračunamo

$$Bs = -\lfloor p/2 \rfloor \beta b + Ax = -(Ax + e) + Ax = -e.$$

Po izreku 8 lahko parametre izberemo tako, da je $|e_i| < \sqrt{p}$ za vsako koordinato e_i vektorja e, saj lahko izberemo porazdelitev $\chi = D_\sigma$ z dovolj majhno standardno deviacijo. Torej je $-e$ vektor z majhnimi vrednostmi. Če poznamo s, lahko določimo, ali je B resnično enakomerna naključna matrika ali pa je bila generirana, kot je opisano zgoraj, saj lahko preprosto preverimo, če je vsaka komponenta Bs od 0 oddaljena za največ $\sqrt{p}$.

Zgornji postopek nam ob pravilni izbiri parametrov n, m, p, χ omogoča naslednje. Če naključno izberemo s, potem lahko generiramo $n+1$-dimenzionalne vektorje b_i (vrstice matrike B), ki jih nihče, ki ne pozna s, ne more razlikovati od enakomerno naključnih. Vendar porazdelitev teh vektorjev ni enakomerno naključna (je samo psevdonaključna), saj za njih velja, da je skalarni produkt b_i in s manjši od $\sqrt{p}$. Psevdonaključni vektorji imajo torej t. i. stranska vrata: čeprav se zdijo naključno izbrani, imajo posebne lastnosti, ki jih lahko izkoristimo.

Definicija 10. *LWE-generator psevdonaključnih vektorjev s stranskimi vrati* je par (G_s, s), kjer je G_s algoritem, ki generira naključne vektorje $b_i \in \mathbb{Z}_p^n$, katerih porazdelitev je brez poznavanja s računsko neločljiva od enakomerno naključnih vrednosti, ter $s \in \mathbb{Z}_p^n$ s prvo komponento $s_1 = \lfloor p/2 \rfloor$ tak, da je $|b_i^T s| < \sqrt{p}$ za vsak b_i generiran z G_s.

Zgoraj smo opisali, kako sestaviti LWE-generator psevdonaključnih vektorjev s stranskimi vrati, ki lahko generira vsaj m psevdonaključnih vektorjev pri predpostavki $\mathrm{LWE}_{n,m,p,\chi}$. Rekli bomo, da tak generator izberemo naključno, če naključno izberemo s, e in A iz ustreznih porazdelitev.

Dvojiško kodiranje

Preden spoznamo kriptosistem, ki omogoča homomorfno šifriranje, potrebujemo še zadnje orodje. Kodiranje je injektivna funkcija, ki vhodne podatke

pretvori v vektorje izbrane oblike. Za razliko od šifriranja kodiranje ne skrije podatkov, ampak jih samo preoblikuje. Da sestavimo kriptosistem s homomorfnim šifriranjem, potrebujemo orodje, ki nam bo vektorje nad $\mathbb{Z}_p$ pretvorilo v (daljše) vektorje z majhnimi vrednostmi.

Dvojiško kodiranje je funkcija, ki naravna števila pretvori v vektorje ničel in enk, t. i dvojiški zapis. Omejimo se na naravna števila, manjša od nekega $p \in \mathbb{N}$, torej števila iz $\mathbb{Z}_p$. Za vsak $x \in Z_p$ lahko enolično določimo vrednosti $y_i \in \{0,1\}$ za $0 \leq i < \lceil \log_2(p) \rceil$, da velja

$$x = \sum_{i=0}^{\lceil \log_2(p) \rceil - 1} 2^i y_i.$$

Preslikavo $x \mapsto \widehat{x}$, kjer je $\widehat{x} = (y_0, y_1, \ldots, y_{\lceil \log_2(p) \rceil - 1}) \in \{0,1\}^{\lceil \log_2(p) \rceil}$ tak, da velja zgornja enačba, imenujemo *dvojiško kodiranje* z $\lceil \log_2(p) \rceil$ biti. Inverzno preslikavo imenujemo dekodiranje in je po zgornji enačbi linearna preslikava, tj. x dobimo iz $\widehat{x}$ kot skalarni produkt $x = q \cdot \widehat{x}$, kjer je $q = (1, 2, 4, \ldots, 2^{\lceil \log_2(p) \rceil - 1})$.

Poleg elementov $\mathbb{Z}_p$ bi želeli dvojiško kodirati tudi vektorje in matrike nad $\mathbb{Z}_p$. Za vektorje nad $\mathbb{Z}_p^n$ dvojiško kodiranje definiramo kot preslikavo iz $\mathbb{Z}_p^n$ v $\{0,1\}^{n\lceil \log_2(p) \rceil}$. Dvojiško kodiranje vektorja $x \in \mathbb{Z}_p^n$ označimo z $\widehat{x}$ in ga definiramo kot vektor dolžine $n\lceil \log_2(p) \rceil$ s komponentami iz množice $\{0,1\}$, katerega bloki dolžine $\lceil \log_2(p) \rceil$ predstavljajo dvojiški zapis komponent x:

$$x = (x_1, x_2, \ldots, x_n) \mapsto (\widehat{x_1}, \widehat{x_2}, \ldots, \widehat{x_n}) = \widehat{x}.$$

Za tako definirano dvojiško kodiranje vektorjev je operacija dekodiranja linearna. Zato lahko s Q označimo matriko dimenzij $n \times n\lceil \log_2(p) \rceil$, da velja $Q\widehat{x} = x$ za vsak $x \in \mathbb{Z}_p^n$. Iz zgoraj opisanega je razvidno, da lahko matriko Q zapišemo kot

$$Q = \begin{bmatrix} 1 & 2 & 4 & \ldots & 2^{\lceil \log_2(p) \rceil - 1} & 0 & 0 & 0 & \ldots & 0 & \ldots & 0 & 0 & 0 & \ldots & 0 \\ 0 & 0 & 0 & \ldots & 0 & 1 & 2 & 4 & \ldots & 2^{\lceil \log_2(p) \rceil - 1} & \ldots & 0 & 0 & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \ldots & 0 & 0 & 0 & 0 & \ldots & 0 & \ldots & 1 & 2 & 4 & \ldots & 2^{\lceil \log_2(p) \rceil - 1} \end{bmatrix}.$$

Podobno lahko dvojiško kodiramo tudi matrike. Za matriko $C \in \mathbb{Z}_p^{m \times n}$ označimo s $\widehat{C}$ matriko dimenzij $m \times n\lceil \log_2(p) \rceil$, ki predstavlja dvojiško kodiranje vrstic c_i, za $1 \leq i \leq m$, v C:

$$C = \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_m \end{bmatrix} \mapsto \begin{bmatrix} \widehat{c_1} \\ \widehat{c_2} \\ \vdots \\ \widehat{c_m} \end{bmatrix} = \widehat{C}.$$

Po definiciji velja $\widehat{C}Q^T = C$ za vsako matriko $C \in \mathbb{Z}_p^{k\times n}$, kjer je $k > 0$ poljuben.

Homomorfno šifriranje

V tem razdelku bomo opisali kriptosistem za homomorfno šifriranje. Ta omogoča, da lastnik skrivnega ključa šifrira podatke, ki jih lahko nato (v šifrirani obliki) ne samo seštevamo, temveč tudi množimo. Za dešifriranje se uporabi isti ključ, tako da je tak sistem simetrične narave. Dokazali bomo, da sistem omogoča varno računanje v oblaku.

Varnost kriptosistema bo temeljila na težavnosti problema LWE, torej bo tak kriptosistem tudi kvantno varen ob pravilni izbiri parametrov:

- **Generiranje ključev:** Ključe, ki nam bodo omogočili šifriranje do k bitov, izberemo na naslednji način. Določimo $n, p, m = kn\lceil\log_2(p)\rceil$ in χ tako, da je odločitveni problem $\mathrm{LWE}_{n,m,p,\chi}$ težek – po izreku 8 to lahko storimo. Naključno izberemo LWE-generator psevdonaključnih vektorjev s stranskimi vrati (G_s, s) (torej izberemo A, e, s, kot opisano v podrazdelku Generirator psevdo-naključnih vektorjev s stranskimi vrati), ki temelji na težavnosti problema $\mathrm{LWE}_{n,m,p,\chi}$. Par (G_s, s) je skrivni ključ.

- **Šifriranje:** Da šifriramo bit $b \in \{0, 1\}$, uporabimo skrivni ključ (G_s, s), tako da s pomočjo G_s generiramo vektorje $d_1, \ldots, d_{n\lceil\log_2(p)\rceil} \in \mathbb{Z}_p^n$ in sestavimo matriko $D \in \mathbb{Z}^{n\lceil\log_2(p)\rceil\times n}$, katere vrstice so vektorji d_i. Izračunamo matriko $bQ^T + D$ in jo podamo v dvojiškem kodiranju:

$$C = b\widehat{Q^T + D}.$$

- **Dešifriranje:** S skrivnim ključem s in danim kriptogramom C izračunamo vektor $x = CQ^Ts$. Če je prva komponenta x bližje 0 kot $\lfloor p/2\rfloor$, potem vrnemo 0, sicer vrnemo 1.

Kriptosistem šifrira vsak bit b v dvojiško matriko C dimenzij $n\lceil\log_2(p)\rceil \times m\lceil\log_2(p)\rceil$. Kot bomo videli, lahko tako šifrirane podatke (matrike) seštevamo in množimo ter tako varno računamo funkcije brez poznavanja podatkov. Seveda je tako računanje časovno in prostorsko veliko zahtevnejše kot računanje na nešifriranih podatkih. Presenetljivo je, da je tak kriptosistem sploh teoretično mogoč. Še več; za ne preveč zahtevne funkcije je s sodobnimi računalniki računanje tudi praktično izvedljivo.

Analizirajmo varnost in pravilnost sheme:

Lema 11. *Če je C kriptogram, ki šifrira b, potem velja:*

$$CQ^T s = bQ^T s + e,$$

kjer je vsaka komponenta vektorja e manjša od $\sqrt{p}$.

Dokaz. Ker velja $\widehat{M}Q^T = M$ za vsako matriko M, lahko izračunamo:

$$CQ^T s = (b\widehat{Q^T + D})Q^T s = (bQ^T + D)s = bQ^T s + Ds.$$

Vrstice D so generirane z LWE-generatorjem psevdonaključnih vektorjev s stranskimi vrati (G_s, s), tako da za vsako vrstico $d_i \in \mathbb{Z}_p^{1\times n}$ velja $|d_i s| < \sqrt{p}$. ∎

Izrek 12. *Zgoraj opisan kriptosistem deluje pravilno in je IND-CPA varen.*

Dokaz. Pokažimo, da kriptosistem deluje pravilno. Ko v postopku dešifriranja izračunamo $x = CQ^T s$, dobimo vektor x, katerega prva komponenta ustreza prvi komponenti $bQ^T s + e$ po lemi 11. Spomnimo se, da je prva komponenta vektorja s enaka $\lfloor p/2 \rfloor$, medtem ko je prva vrstica Q^T enaka $[1, 0, \ldots, 0]$. Torej je prva komponenta $m_1 = b\lfloor p/2 \rfloor + e_1$, kjer je $|e_1| < \sqrt{p}$. Sledi, da če je $b = 1$, je m_1 bližje $\lfloor p/2 \rfloor$ kot 0, in obratno, če je $b = 0$.

Varnost kriptosistema se dokaže tako, da se argumentira, da nihče, ki ne pozna skrivnosti s, ne more računsko razlikovati šifrirane vrednosti $b = 0$ od šifrirane vrednosti $b = 1$. Ker je matrika D generirana z LWE-generatorjem psevdonaključnih vektorjev, je za vsakega, ki ne pozna s, računsko neločljiva od enakomerno naključne matrike. Torej je tudi vrednost $bQ^T + D$ neločljiva od enakomerno naključne matrike. Sledi, da nihče, ki ne pozna s, ne more ločiti, katera vrednost je bila šifrirana. ∎

Pripravljeni smo, da predstavimo, kako opisan kriptosistem omogoča homomorfno šifriranje. Naj bosta C_1 in C_2 kriptograma, ki šifrirata bita b_1 in b_2:

- **Seštevanje:** definirajmo $C_1 \oplus C_2 = C_1 + C_2$ in obravnavajmo slednje kot nov kriptogram. Za dešifriranje izračunamo

  $$(C_1 \oplus C_2)Q^T s = (C_1 + C_2)Q^T s = (b_1 + b_2)Q^T s + (e_1 + e_2),$$

  kjer je $||e_i||_\infty < \sqrt{p}$, po lemi 11. Vidimo, da se $C_1 \oplus C_2$ dešifrira v vrednost $b_1 + b_2 \mod 2$, torej v XOR-operacijo bitov b_1 in b_2, le da je šum $e_1 + e_2$, ki pri tem nastane, povečan.

- **Množenje:** definirajmo $C_1 \otimes C_2 = \widehat{C_1 Q^T} C_2$. Prav tako lahko izračunamo

$$\begin{aligned}(C_1 \otimes C_2)Q^T s &= \widehat{C_1 Q^T} C_2 Q^T s = \widehat{C_1 Q^T}(b_2 Q^T s + e_2) \\ &= b_2 C_1 Q^T s + \widehat{C_1 Q^T} e_2 = b_1 b_2 Q^T s + b_2 e_1 + \widehat{C_1 Q^T} e_2,\end{aligned}$$

  kjer prva enakost velja po definiciji $C_1 \otimes C_2$, druga in četrta po lemi 11 in tretja velja, saj je $\widehat{X} Q^T = X$ za vsako matriko X po lastnostih dvojiškega kodiranja. Po lemi 11 je $||e_i||_\infty < \sqrt{p}$. Tudi tokrat lahko na zgoraj navedeno gledamo kot na dešifriranje vrednosti $b_1 b_2 \mod 2$ s šumom $b_2 e_1 + \widehat{C_1 Q^T} e_2$. Prepričajmo se, da je slednja vrednost res majhna, torej da jo res lahko obravnavamo kot šum in ne pokvari dešifriranja. Ker velja $||e_1||_\infty < \sqrt{p}$ in $b_2 \in \{0, 1\}$, je prvi člen res majhen. Po drugi strani je $\widehat{C_1 Q^T}$ dvojiško kodiranje matrike $C_1 Q^T$, torej so njene komponente iz $\{0, 1\}$. Ker ima $n\lceil \log_2(p) \rceil$ stolpcev in je $||e_2||_\infty < \sqrt{p}$, vidimo, da je vsaka komponenta drugega člena omejena z $n\lceil \log_2(p) \rceil \sqrt{p}$. Za primerno izbrane parametre je taka vrednost še vedno manjša od $p/4$, torej dešifriranje $C_1 \otimes C_2$ uspe in vrne $b_1 b_2 \mod 2$, torej AND-operacijo bitov b_1 in b_2.

Kot vidimo, nam kriptosistem omogoča operaciji seštevanja in množenja kriptogramov in s tem operaciji XOR in AND na šifriranih podatkih. Želeli bi kriptosistem, ki bi lahko izračunal vsako funkcijo. Kot smo omenili v razdelku Aditivno šifriranje, zadošča, da sistem omogoča zaporedno izvajanje NAND-vrat (Shefferjevega veznika) na šifriranih podatkih, saj lahko s tem veznikom predstavimo vsako funkcijo. Uporabimo množenje za definicijo nove operacije na šifriranih podatkih:

- **NAND:** definirajmo $C_1 \barwedge C_2 = I - C_1 \otimes C_2$. Če slednje obravnavamo kot nov kriptogram in dešifriramo, dobimo

$$(C_1 \barwedge C_2)Q^T s = (I - C_1 \otimes C_2)Q^T s = (1 - b_1 b_2)Q^T s - b_2 e_1 - \widehat{C_1 Q^T} e_2.$$

  V izračunu smo uporabili enakost, ki smo jo dobili pri operaciji množenja. Vidimo, da se $C_1 \barwedge C_2$ dešifrira v vrednost $1 - b_1 b_2 \mod 2$, torej NAND-vrednost bitov b_1 in b_2.

S takim kriptosistemom lahko torej (teoretično) izračunamo poljubno funkcijo na šifriranih podatkih, vendar pri tem šum narašča in z večkratnim ponavljanjem operacije lahko naraste do takih vrednosti, da dešifriranje ni več uspešno. Zato je treba vnaprej omejiti število zaporednih operacij, ki

jih sistem še omogoča, oziroma v primeru kompleksnejših funkcij parametre povečati tako, da še sprejmejo izbrano funkcijo.

Takemu kriptosistemu pravimo tudi *delno homomorfno šifriranje*. Gentry je v [3] dokazal, da se v nekaterih primerih da delno homomorfno šifriranje spremeniti v *polno homomorfnega* s tehniko, ki jo je imenoval *bootstraping*. Tehnika deluje na vsakem delno homomorfnem sistemu, ki zadošča tako imenovani *ciklični varnosti*. Vendar je prehod na polno homomorfen sistem računsko zelo zahteven in zato počasnejši, tako da se večina praktičnih uporab omeji na delno homomorfno šifriranje. Zainteresiranega bralca, ki bi želel izvedeti več o uporabi problema LWE v kriptografiji, usmerimo k branju [2].

V času pisanja tega članka že obstaja več implementacij delno in polno homomorfnih kriptosistemov. Veliko truda je bilo vloženega v razvoj učinkovitejšega homomorfnega šifriranja, kot je bil opisan v tem razdelku. Glavni steber trenutnega razvoja predstavljajo problem LWE in njegove izpeljanke.

LITERATURA

[1] M. R. Albrecht, R. Player in S. Scott, *On the concrete hardness of learning with errors*, Journal of Mathematical Cryptology **9**(3) (2015), 169–203.

[2] B. Barak, *An intensive introduction to cryptography*, dostopno na `intensecrypto.org/public/index.html`, ogled 3. 11. 2020.

[3] C. Gentry, *Fully homomorphic encryption using ideal lattices*, v Proceedings of the forty-first annual ACM symposium on Theory of computing, 2009, 169–178.

[4] O. Regev, *On lattices, learning with errors, random linear codes, and cryptography*, Journal of the ACM **56**(6) (2009), 1–40.

[5] R. L. Rivest, L. Adleman in M. L. Dertouzos, *On data banks and privacy homomorphisms*, Foundations of secure computation **4**(11) (1978), 169–180.

[6] P. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM review **41**(2) (1999), 303–332.

[7] *Post-Quantum Cryptography*, dostopno na `csrc.nist.gov/Projects/Post-Quantum-Cryptography`, ogled 3. 11. 2020.