UNIVERSITAT DE BARCELONA

Treball final de grau

GRAU DE MATEMÀTIQUES

GRAU D'ENGINYERIA INFORMÀTICA

Facultat de Matemàtiques i Informàtica
Universitat de Barcelona

El criptosistema NTRU
Una solució a l'amenaça quàntica

Autor: Miquel Guiot Cusidó

Director: Dr. Xavier Guitart Morales
Realitzat a: Departament de Matemàtiques i Informàtica

Barcelona, 24 de gener de 2022

Abstract

The aim of this project is to study the quantum-resistant cryptosystem NTRU, both from a theoretical and practical point of view. Firstly, the mathematical results on which it is based are presented and the existing attacks against it are analyzed. Finally, a practical implementation of the cryptosystem and four of its attack is carried out, from which experimental results are extracted.

Resum

L'objetiu d'aquest treball és estudiar el criptosistema postquàntic NTRU, tant des d'una vessant teòrica com pràctica. En primer lloc, es presenten els resultats matemàtics sobre els que es fonamenta i s'analitzen els atacs existents vers ell. Finalment, es du a terme una implementació pràctica del criptosistema i de quatre dels seus atacs, a partir de la qual se n'extreuen resultats experimentals.

2020 Mathematics Subject Classification. 11H06, 94A60

Agraïments

Vull agrair al Dr. Xevi Guitart no tan sols el seu acompanyament i guiatge al llarg de tot el treball, des de la proposta del tema fins als darrers consells, sinó també l'haver-me introduït en l'apassionant món de la criptografia. Ha estat un plaer poder desenvolupar aquest treball sota la teva tutorització.

També dono les gràcies a tothom qui, d'alguna manera o altra, m'ha ajudat durant aquest extens periple universitari: familiars, amics, companys i professors.

Finalment, m'agradaria fer un agraïment especial als meus pares, a qui agraeixo tota l'estima i suport que sempre m'han mostrat, i al meu germà Quim, de qui he après la importància de la constància i l'esforç. Sense vosaltres no hagués estat possible. Moltes gràcies de tot cor.

Índex

Capítol 1

Introducció

1.1 Un viatge a la criptografia postquàntica

És ben sabut que els humans som éssers socials per naturalesa. Des de la prehistòria, hem tendit a organitzar-nos en tribus i les relacions interpersonals que se'n deriven han anat guanyant complexitat a través de la nostra evolució com a societat, donant lloc a necessitats com la privacitat i la voluntat de transmetre secrets. Com a resposta a aquesta casuística, neix l'art d'encriptar i desencriptar missatges, és a dir, la criptografia.

De fet, els experts situen l'origen de la criptografia fa uns 4000 anys a l'Antic Egipte, on els escribes es comunicaven entre ells mitjançant jeroglífics desconeguts per la resta de la població. D'aleshores ençà, han estat infinitud les tècniques emprades per tal d'encriptar missatges, des del xifratge de Cèsar propi dels romans fins a la màquina Enigma a la Segona Guerra Mundial. Malgrat que aquests mecanismes esdevenien cada vegada més complexos, tots ells disposaven del mateix factor limitant: eren criptosistemes de clau privada, és a dir, emissor i receptor necessitaven acordar prèviament una clau compartida per tal de poder encriptar i desencriptar el missatge correctament. Per exemple, en el xifratge de Cèsar ambdós actors han d'establir amb antel·lació el número referent al decalatge. Així doncs, no ha estat fins ben entrat el segle XX, amb l'aparició de la computació i dels criptositemes de clau pública, que s'ha resolt aquesta problemàtica donant lloc al que es coneix com a criptografia moderna.

En línies generals, la criptografia moderna és la ciència que estudia les comunicacions segures entre dos agents, emissor i receptor, (que en l'àmbit de la criptografia sovint són anomenats Alice i Bob), mitjançant eines pròpies de les matemàtiques i la informàtica com ara la teoria de nombres, l'estadística, la computació teòrica i l'optimització computacional. Entrant en detalls, convé puntualitzar que actualment la criptografia engloba molts altres aspectes més enllà de la comunicació xifrada, com poden ser l'autenticitat i les signatures digitals, si bé en aquest treball es tracta únicament l'enviament i recepció de missatges xifrats. En conseqüència, per tal d'assolir aquest objectiu, en el si de la criptografia moderna existeixen dos enfocaments clarament diferenciats: els criptosistemes de clau privada i els criptosistemes de clau pública.

D'una banda, tal i com s'ha esmentat amb anterioritat, els criptosistemes de clau pri-

vada disposen d'un bagatge històric considerable i el seu funcionament es basa en la compartició d'una clau simètrica entre l'Alice i en Bob que els permet encriptar i desencriptar missatges. En afegit, destaquen per ser criptosistemes molt eficients tant a nivell d'execució com d'emmagatzematge i la seva seguretat rau en el fet que aquesta clau privada únicament és coneguda per ambdós interlocutors. Per tant, esdevé un sistema molt segur sempre i quan aquesta clau no sigui coneguda per cap atacant. No obstant això, el seu principal inconvenient és, precisament, la compartició d'aquesta clau.

D'altra banda, els criptosistemes de clau pública daten de la dècada dels 70, amb la publicació l'any 1976 del protocol Diffie-Hellman [DH76] i l'any 1978 del criptosistema RSA [RSA78]. A grans trets, la idea darrere d'aquesta tipologia de criposistemes és permetre l'encriptació i desencriptació de missatges sense necessitat d'establir prèviament una clau comuna. Més concretament, l'Alice genera un parell de claus, una pública i una privada, amb la propietat que els missatges encriptats amb la clau pública es poden desencriptar amb la clau privada. Acte seguit, tal i com els seus noms indiquen, l'Alice publica la clau pública a la xarxa i es guarda tan sols per a ella la clau privada. D'aquesta manera, quan en Bob o qualsevol altre agent vulgui enviar un missatge xifrat a l'Alice, tan sols caldrà que l'encripti mitjançant la clau pública, ja que l'encriptació resultant únicament podrà ser desencriptada mitjançant la clau privada que només té l'Alice, impossibilitant així l'obtenció del missatge original per part de tercers. Com a contrapartida, convé subratllar que els criptosistemes de clau pública són força més lents que els de clau privada i la seva seguretat no és tan elevada. Això es deu, en gran part, a que la seva resistència es fonamenta en problemes matemàtics que es consideren difícils de resoldre, com per exemple la factorització d'enters en el cas del criptosistema RSA.

Malgrat que, pel vist fins ara, els criptosistemes de clau pública i clau privada puguin semblar dues visions antagòniques del mateix problema, el cert és que, a la pràctica, resulta tot el contrari. En particular, l'Alice i en Bob es comuniquen mitjançant criptosistemes de clau privada on la clau ha estat prèviament compartida gràcies a l'ús d'un criptosistema de clau pública. D'aquesta manera, ambdues tipologies de criptosistemes treballen en una espècie de simbiosi quasi perfecta, aprofitant l'alta eficiència dels criptosistemes de clau privada i resolent-ne la problemàtica referent a la compartició de la clau. Sense anar gaire lluny, la majoria de comunicacions xifrades que es duen a terme diàriament per Internet empren aquest esquema.

Ara bé, l'elecció del terme *quasi perfecta* en el paràgraf anterior no és pas casual i respon a una casuística molt concreta, la publicació l'any 1994 de l'algorisme de Shor [Sho94]. Aquest algorisme redueix la complexitat dels problemes de factorització d'enters i del logarisme discret, permetent resoldre'ls en temps polinomial. Tot i que únicament es pot implementar de forma eficient en ordinadors quàntics, els quals eren poc més que una utopia en el moment de la seva publicació, l'existència d'aquest algorisme juntament amb els avenços tecnològics en el camp de la computació quàntica posen en escac la majoria dels criptosistemes de clau pública utilitzats en l'actualitat i, en conseqüència, comprometen bona part de les comunicacions privades. De fet, els experts estimen que, en un termini de 2 o 3 dècades, l'existència d'ordinadors quàntics prou potents per poder executar l'algorisme de Shor deixarà de ser quelcom propi de la ciència ficció per esdevenir realitat.

Així doncs, no és d'estranyar que en els darrers anys molts estaments governamen-

tals i empreses tecnològiques hagin dipositat molts esforços en el desenvolupament del que es coneix com a criptografia postquàntica, és a dir, la ciència que estudia els criptosistemes resistents als atacs quàntics. En aquesta línia, l'any 2016 el NIST (de l'anglès *National Institute of Standards and Technology*) va iniciar un concurs amb l'objetiu d'establir i estandaritzar els criptosistemes postquàntics amb millors prestacions. Actualment, el concurs es troba en la seva fase final i els candidats que continuen sobre la taula es corresponen als criptosistemes Classic McEliece, CRYSTALS-KYBER, NTRU i SABER. D'aquests quatre criptosistemes, els tres darrers comparteixen el fet que estan dissenyats a partir de la teoria matemàtica de reticles, motiu pel qual es considera la criptografia basada en reticles com el camp més prometedor dins de la criptografia postquàntica.

1.2 Continguts d'aquesta memòria

En primer lloc, és important precisar que l'objectiu principal d'aquesta memòria és estudiar la criptografia basada en reticles i, en particular, el criptosistema NTRU des d'una perspectiva matemàtica. Més concretament, mitjançant eines pròpies de la teoria de nombres i de la computació teòria, es pretén presentar els resultats més rellevants en aquest àmbit i enllaçar-los amb les seves aplicacions pràctiques, en contraposició amb el que succeeix habitualment. Sovint, els escrits en aquest camp tendeixen a polaritzar-se, ja sigui descuidant-ne els fonaments matemàtics per centrar-se únicament en l'estudi del criptosistema, o bé a la inversa, excedint-se en el contingut matemàtic i deixant de banda aspectes com ara la seguretat del criptosistema. Així doncs, aquest treball parteix amb la premisa d'oferir una perspectiva general i permetre al lector comprendre com ambdues vessants, la matemàtica i la informàtica, estan més relacionades del que sembla. En afegit, convé destacar que l'estudi que es presenta no només és teòric, sinó que també disposa d'una part pràctica corresponent a la implementació del criptosistema NTRU i quatre dels seus atacs.

Enllaçant-ho amb això, la tria del criptosistema NTRU per sobre dels altres tres candidats respon a dos motius molt concrets. D'una banda, es tracta d'un criptosistema basat en reticles i per tant, tal i com s'ha esmentat prèviament, s'engloba dins del camp més prometedor de la criptografia postquàntica. D'altra banda, de les alternatives existents de criptosistemes basats en reticles, és sense cap mena de dubte el criptosistema amb més història al darrere i està estretament relacionat amb algorismes d'un interès científic considerable, com per exemple l'algorisme LLL.

A mode de síntesi, tot seguit es llisten els diversos capítols d'aquest estudi, així com també els temes que tracten.

D'entrada, el capítol que segueix presenta el concepte de reticle i n'estudia les seves propietats i definicions bàsiques. A més a més, també aprofundeix al voltant dels principals problemes presents en aquest camp i analitza alguns dels seus algorismes més destacats.

El capítol 3 fa referència als anells polinomials convolucionals de forma molt breu, ja que es tracten d'una eina matemàtica esencial per a l'estudi del criptosistema NTRU. D'aquesta manera, s'hi presenten les definicions i propietats bàsiques i alguns resultats referents a la invertibilitat polinomial.

El capítol 4 és el que es correspon a l'estudi teòric del criptosistema NTRU. En ell s'hi llisten els diversos passos a realitzar, des de l'elecció dels paràmetres públics fins la desencriptació, l'anàlisi de la seva seguretat i els atacs existents vers ell. Endemés, també s'hi detalla l'estreta relació entre el propi criptosistema i la teoria de reticles.

De manera similar, el capítol 5 engloba la part pràctica d'aquest projecte. Més concretament, explica de forma resumida l'entorn de programació, l'estructura i les funcionalitats del codi implementat.

En el capítol 6 es recull tot l'exposat al projecte en forma de conclusions.

Finalment, a l'Annex s'hi presenten els resultats experimentals obtinguts en forma de taules i gràfics, on es tenen en compte les mètriques referents al temps d'execució, els valors dels paràmetres públics i la mida de les diverses claus. A més a més, també s'hi explica de forma resumida els criptosistemes CRYSTALS-KYBER i SABER i s'hi detalla la planificació temporal del projecte.

Capítol 2

Reticles

2.1 Definicions i propietats bàsiques

Les primeres aparicions de la teoria de reticles en les matemàtiques daten de mitjans del segle XVIII. D'aleshores ençà, han estat moltes les seves aplicacions: ja siguin en camps propis de les matemàtiques pures com poden ser les àlgebres de Lie, o en les matemàtiques aplicades on, més enllà de la criptografia, pren importància en la teoria de codis. De fet, la teoria de reticles ha estat present en la tasca de matemàtics clàssics de la importància de Gauss (veure Problema del Cercle de Gauss [Hux96]) o Lagrange, així com també en publicacions de matemàtics contemporanis com László Lovász, guardonat el 2021 amb el premi Abel.

Al llarg d'aquest capítols es veuran un seguit de definicions i propietats bàsiques de la teoria de reticles que permetran, en seccions posteriors, entendre de primera mà el funcionament de la criptografia basada en reticles.

Definició 1. *Siguin $v_1, ..., v_n \in \mathbb{R}^m$ un conjunt de vectors linealment independents. Anomenem reticle L generat per $v_1, ..., v_n$ al conjunt de combinacions lineals de $v_1, ..., v_n$ amb coeficients enters, és a dir,*

$$L = \{a_1 v_1 + ... + a_n v_n \mid a_1, ...a_n \in \mathbb{Z}\}. \tag{2.1}$$

És interessant observar la similitud d'aquesta definició amb el concepte d'espai vectorial. En aquest sentit, resulta pràctic pensar els reticles com un conjunt de punts a $\mathbb{R}^n$, on cada punt representa el final de cada vector que surt des de l'origen de coordenades, tal i com es mostra a la figura 2.1.

A partir d'aquesta mateixa definició es poden introduir els conceptes de base i dimensió d'un reticle.

Definició 2. *Donat un reticle L, una base seva és qualsevol conjunt de vectors linealment independents que generin L. La dimensió del reticle L es correspon al nombre de vectors d'una base seva qualsevol.*

Tal i com succeeix amb els espais vectorials, tot reticle L disposa d'una base i aquesta no és única. De fet, en apartats posteriors es veurà com la multiplicitat i tria de les bases

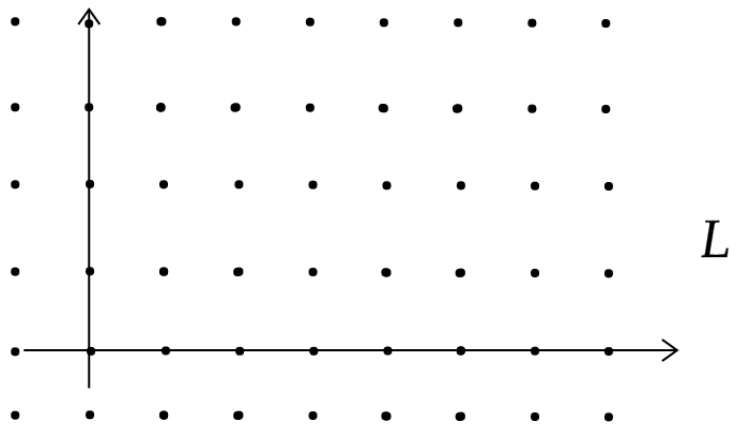

Figura 2.1: Exemple de reticle amb $n = 2$.

juga un paper cabdal en la criptografia de reticles. La proposició següent relaciona bases diferents d'un mateix reticle L.

Proposició 3. *Dues bases qualssevol d'un reticle L estan relacionades per una matriu amb tots els coeficients enters determinant igual a* ± 1.

Demostració. Siguin $\{v_1, ..., v_n\} = V$ i $\{w_1, ..., w_n\} = W$ dues bases diferents del mateix reticle L. Com que els vectors $v_1, ..., v_n$ són una base de L, és clar que es poden escriure els vectors $w_1, ..., w_n$ com a combinació lineal de $v_1, ..., v_n$ amb els coeficients a_{ij} enters,

$$w_1 = a_{11}v_1 + ... + a_{n1}v_n$$

$$\vdots$$

$$w_n = a_{1n}v_1 + ... + a_{nn}v_n.$$

En format matricial, s'obté la matriu de canvi de base següent que permet passar de W a V

$$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{pmatrix}.$$

Així doncs, s'ha vist que A disposa de coeficients enters i, per tant, el seu determinant és enter. Raonant de forma anàloga, s'obté la matriu B de canvi de base que permet passar de V a W i es prova que disposa de coeficients enters.

Per acabar, únicament s'ha de veure que els determinants d'ambdues matrius A i B són iguals a ± 1. Donat un vector qualsevol $u_W \in L$ expressat en la base W, es té $u_V = Au_W$ i, multiplicant a l'esquerra en ambdues bandes per B, s'obté que $BAu_W = Bu_V = u_W$, de tal manera que $BA = Id$. Per tant, com que s'ha vist que A i B tenen determinant enter i se sap que el determinant de $Id = 1$, s'obté que $\det(A) = \det(B) = \pm 1$ tal i com es volia veure. □

Computacionalment, sovint interessa treballar amb reticles que no només tinguin els coeficients enters, sinó que les coordenades dels vectors també ho siguin. És el que es coneix com a reticles enters. De fet, aquesta tipologia de reticles és la que empra el criptosistema NTRU.

Definició 4. *Un reticle enter és un reticle en el que tots els seus vectors disposen de totes les coordenades enteres. En altres paraules, un reticle enter és un subgrup de $\mathbb{Z}^n$ per $n \geq 1$.*

La definició i la proposició que segueixen presenten el concepte de domini fonamental.

Definició 5. *Sigui L un reticle de dimensió n i sigui $\{v_1, ..., v_n\} = V$ una base de L. S'anomena domini fonamental de L corresponent a V al conjunt*

$$\mathcal{F}(v_1, ..., v_n) \ = \ \{t_1 v_1 + ... + t_n a_n \mid 0 \leq t_1, ..., t_n < 1\} \tag{2.2}$$

A la figura 2.2 es pot veure un exemple de domini fonamental $\mathcal{F}$ quan $n = 2$. Intuïtivament, es pot pensar el domini fonamental com un paral·lelepípede.

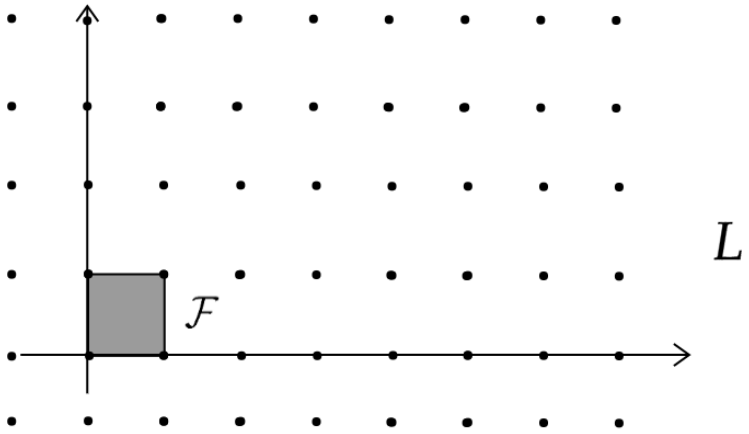

Figura 2.2: Exemple de domini fonamental amb $n = 2$.

Proposició 6. *Sigui L un reticle de dimensió n i sigui $\mathcal{F}$ un domini fonamental de L. Aleshores qualsevol vector $v \in \mathbb{R}^n$ es pot escriure de la forma $v = w + t$ per un únic $t \in \mathcal{F}$ i un únic $w \in L$. En altres paraules, la unió de les translacions del domini fonamental*

$$\mathcal{F} + w = \{t + w \mid t \in \mathcal{F}\}$$

cobreix exactament $\mathbb{R}^n$, on w recorre tots els vectors del reticle L.

Demostració. Sigui $\{w_1, ..., w_n\} = W$ una base de L que es correspon amb el domini fonamental $\mathcal{F}$. És clar que els vectors $w_1, ..., w_n$ són linealment independents a $\mathbb{R}^n$ i, per tant, en són base. Així doncs, es pot escriure qualsevol vector $v \in \mathbb{R}^n$ com a combinació lineal de la base W:

$$v = \alpha_1 w_1 + ... + \alpha_n w_n \text{ per alguns } \alpha_1, ..., \alpha_n \in \mathbb{R}.$$

Acte seguit, es pot reescriure cada α_i com a suma de la seva part entera i decimal, és a dir,

$$\alpha_i = t_i + a_i \text{ amb } 0 \leq t_i < 1 \text{ i } a_i \in \mathbb{Z}.$$

Així doncs, és clar que v pot ser escrit de la forma volguda de la manera següent:

$$v = \alpha_1 w_1 + ... + \alpha_n w_n = t_1 w_1 + ... + t_n w_n + a_1 w_1 + ... + a_n w_n.$$

Per tal de concloure la demostració, resta provar que aquesta expressió és única. Per veure-ho, es suposa que $v = t + w = t' + w'$ disposa de dues representacions com a suma d'un vector de $\mathcal{F}$ i d'un vector de L i s'arriba a la conclusió que $t = t'$ i $w = w'$.

En primer lloc, s'expressen ambdues representacions com a combinació lineal de la base W

$$(t_1 + a_1)w_1 + ... + (t_n + a_n)w_n = (t'_1 + a'_1)w_1 + ... + (t'_n + a'_n)w_n.$$

Com que els vectors $w_1, ..., w_n$ són linealment independents, és clar que $t_i + a_i = t'_i + a'_i$ per tota $i = 1, ..., n$. A continuació, aprofitant el fet que $a_i \in \mathbb{Z}$ per tota $i = 1, ..., n$, tenim que $t_i - t'_i = a_i - a'_i \in \mathbb{Z}$ per tota $i = 1, ..., n$. Però com que per construcció $0 \leq t_i, t'_i < 1$, per força ha de ser $t_i - t'_i = 0$. D'aquí és directe que $t_i = t'_i$ i $a_i = a'_i$ per tota $i = 1, ..., n$ tal i com es volia veure. □

El concepte de domini fonamental és una eina bàsica en la teoria de reticles entre d'altres coses perquè, tal i com mostra la proposició 6, permet obtenir una partició de $\mathbb{R}^n$. De fet, una altra de les seves utilitats rau en el fet que permet definir el que es coneix com a determinant del reticle L, concepte que es defineix a continuació i que al llarg d'aquest capítol es demostrarà que representa un invariant del reticle i que permet fer estimacions sobre les mides dels vectors del reticle.

Definició 7. *Sigui L un reticle de dimensió n i sigui $\mathcal{F}$ un domini fonamental de L. El volum n-dimensional de $\mathcal{F}$ s'anomena determinant o covolum de L. El determinant de L es denota com* $\det(L)$.

De la definició anterior és interessant puntualitzar que un reticle $L \subset \mathbb{R}^n$ de dimensió n, per si mateix, no disposa de volum, ja que es trata d'una col·lecció comptable de punts. Així doncs, el covolum de L es correspon al volum del grup quocient $\mathbb{R}^n / L$.

Com a conseqüència de la definició de determinant d'un reticle L, sorgeix la pregunta de com es pot calcular aquest concepte. Si bé és cert que existeixen diversos resultats interessants al respecte [Mic10], pel tema que ens ocupa es suficient amb estudiar el cas en la que la dimensió del reticle i de l'espai ambient són la mateixa i donar-ne una fita superior.

Proposició 8. *Sigui $L \subset \mathbb{R}^n$ un reticle de dimensió n, sigui $\{v_1, ..., v_n\} = V$ una base de L i sigui $\mathcal{F} = \mathcal{F}(v_1, ..., v_n)$ el domini fonamental de L associat a la base V segons la definició 5. Sigui F la matriu que té per columnes v_j, $j = 1, ..., n$, els vectors de la base V. Aleshores el volum de $\mathcal{F}$ es correspon a*

$$Vol(\mathcal{F}(v_1, ..., v_n)) = |\det(F(v_1, ..., v_n))|. \tag{2.3}$$

Demostració. Primerament, és important precisar que aquesta demostració empra conceptes bàsics del càlcul en diverses variables, els quals es poden trobar a [SH21].

Entrant en matèria, es pot calcular el volum de $\mathcal{F}$ com la integral de la funció constant amb valor 1 sobre la regió delimitada per $\mathcal{F}$, és a dir,

$$Vol(\mathcal{F}) = \int_{\mathcal{F}} dx_1 ... dx_n.$$

En base a la definició 5, es pot dur a terme un canvi de variables per expressar $x = (x_1, ..., x_n)$ en funció de $t = (t_1, ..., t_n)$ de la forma següent: $x = (x_1, ..., x_n) = t_1 v_1 + ... + t_n v_n$.

En format matricial, aquest canvi de variables s'escriu $x = Ft$. És a dir, la matriu Jacobiana d'aquest canvi de variables és la matriu F definida a l'enunciat de la proposició i el domini fonamental $\mathcal{F}$ és la imatge del cub unitat $C_n = [0,1]^n$ per la matriu F. Així doncs, aplicant aquest canvi de variables i calculant la integral s'obté

$$\begin{aligned}\text{Vol}(\mathcal{F}) = \int_{\mathcal{F}} dx_1 ... dx_n = \int_{FC_n} dx_1 ... dx_n = \\ = \int_{C_n} |\det(F)| dt_1 ... dt_n = |\det(F)| \text{Vol}(C_n) = |\det(F)|.\end{aligned}$$

□

D'aquesta darrera proposició se n'extreu el corol·lari següent, el qual prova la invariància del volum dels dominis fonamentals d'un mateix reticle.

Corol·lari 9. *Sigui $L \subset \mathbb{R}^n$ un reticle de dimensió n. Tots els dominis fonamentals de L tenen el mateix volum. En altres paraules, el* $\det(L)$ *és un invariant del reticle L.*

Demostració. Siguin $\{v_1, ..., v_n\} = V$ i $\{w_1, ..., w_n\} = W$ dues bases diferents de L i siguin $F(v_1, ..., v_n)$ i $F(w_1, ..., w_n)$ els seus dominis fonamentals associats. Per la proposició 3 existeix una matriu n per n unitària M tal que $W = MV$. Així doncs, tenint en compte això i que les matrius unitàries tenen determinant igual a ± 1, s'obté

$$\begin{aligned}\text{Vol}(\mathcal{F}(w_1, ..., w_n)) = |\det(F(w_1, ..., w_n))| = |\det(MF(v_1, ..., v_n))| = \\ = |\det(M)||\det(F(v_1, ..., v_n))| = |\det(F(v_1, ..., v_n))| = \text{Vol}(\mathcal{F}(v_1, ..., v_n))\end{aligned}$$

□

Addicionalment, si hom pensa els vectors de la base del reticle com els costats del paral·lelepípede corresponent al domini fonamental, és clar que el volum màxim s'obté quan aquests vectors són dos a dos ortogonals. És precisament aquest raonament intuïtiu el que hi ha darrere del que es coneix com desigualtat de Hadamard.

Proposició 10 (Desigualtat de Hadamard)**.** *Sigui $L \subset \mathbb{R}$ un reticle un reticle de dimensió n, sigui $\{v_1, ..., v_n\} = V$ una base de L i sigui $\mathcal{F}$ un domini fonamental de L. Aleshores*

$$\det(L) = Vol(\mathcal{F}) \leq \|v_1\| ... \|v_n\|. \tag{2.4}$$

Demostració. Sigui F la matriu que té per columnes $v_j,\ j = 1, ..., n$, els vectors de la base V. La descomposició QR és un resultat conegut d'àlgebra lineal [Yan07] que permet escriure la matriu F com

$$F = QR,$$

on Q és una matriu ortogonal que té per columnes $q_j,\ j = 1, ..., n$, els vectors de la base ortonormal obtinguda via Gram-Schmidt i R una matriu triangular superior. Així doncs, és clar que

$$v_j = \sum_{i=1}^{j} r_{ij} q_i$$

i prenent normes al quadrat queda

$$\|v_j\|^2 = \sum_{i=1}^{j} |r_{ij}|^2 \|q_j\| \geq |r_{jj}|^2,$$

on s'ha aprofitat l'ortonormalitat de les columnes q_j. Finalment, tenint en compte que Q és una matriu ortogonal, que R és una matriu triangular superior i la darrera desigualtat, s'obté

$$\det(F) \leq |\det(F)| = |\det(Q)||\det(R)| = 1 \cdot |\prod_{j=1}^{n} r_{jj}| \leq \prod_{j=1}^{n} \|v_j\|$$

tal i com es volia veure. □

Una de les principals conseqüències que es deriven de l'enunciat de la desigualtat de Hadamard és l'existència d'una fita inferior pels vectors de qualsevol base del reticle. Així doncs, aquesta desigualtat afirma que, donat un reticle L, és impossible trobar una successió de vectors de L on les seves normes es redueixin indefinidament.

2.2 Problemes i heurístiques associats als reticles

Al llarg de l'apartat anterior, s'han introduït diversos conceptes propis de la teoria de reticles que deriven en un ventall molt ampli de problemes. No obstant això, a nivell computacional i criptogràfic destaquen dos reptes per sobre de la resta: el problema del vector més curt i el problema del vector més proper. En aquest sentit, aquesta secció pretèn establir la base teòrica d'ambdós reptes de cara a les aplicacions que se'n derivaran en apartats posteriors.

2.2.1 Els problemes del vector més curt i del vector més proper

Definició 11. *S'anomena problema del vector més curt o SVP (de l'anglès* Shortest Vector Problem*) a trobar un dels vectors més curts diferents de zero d'un reticle L. En altres paraules, trobar un vector $v \in L$ diferent de zero que minimitzi la norma euclidiana $\|v\|$.*

De l'enunciat anterior és important remarcar el fet que el vector més curt d'un reticle no ha de ser necessàriament únic. Sense anar més lluny, en el reticle $L = \mathbb{Z}^2$ els 4 vectors $(1,0)$, $(-1,0)$, $(0,-1)$ i $(0,1)$ compleixen clarament la condició de ser els vectors més petits diferents de zero.

Definició 12. *S'anomena problema del vector més proper o CVP (de l'anglès* Closest Vector Problem*) a donat un vector $v \in \mathbb{R}^n$ trobar-ne el vector $w \in L$ més proper a v, és a dir, tal que minimitzi la norma euclidiana $\|v - w\|$.*

Malgrat que puguin semblar uns enunciats aparentment senzills, el cert és que ambdós problemes són computacionalment exigents a mesura que creix la dimensió del reticle. De fet, des d'una òptica de la teoria de la complexitat [For09], cal precisar que és un resultat conegut que el CVP és un problema NP-Hard, sent NP-Complete la seva versió de decisió, mentre que el SVP és NP-Hard sota reduccions aleatòries. Les demostracions, juntament amb una explicació extensa d'aquests resultats i conceptes, escapen l'abast d'aquest treball i es poden trobar amb tot detall a [MG02].

No obstant això, a l'hora de treballar en criptografia s'acostumen a utilitzar casos particulars d'aquests problemes que disposen d'alguna propietat concreta que en millori l'eficiència o la resolució. Entrant en detall, tot seguit es llisten un seguit de variants del SVP i del CVP que usarem en apartats posteriors.

Definició 13. *S'anomena problema del vector més curt aproximat o apprSVP (de l'anglès* Approximate Shortest Vector Problem*) a, donada una funció $\psi(n)$ i un reticle L de dimensió n, trobar un vector $v \in L$ diferent de zero que no sigui més de $\psi(n)$ vegades més llarg que el vector més curt de L. En altres paraules, trobar un vector $v \in L$ diferent de zero tal que $\|v\| \leq \psi(n)\,\|w\|$, on w és el vector més curt de L.*

Definició 14. *S'anomena problema del vector més proper aproximat o apprCVP (de l'anglès* Approximate Closest Vector Problem*) a, donada una funció $\psi(n)$, un reticle L de dimensió n i un vector $v \in \mathbb{R}^n$ trobar-ne un vector $w \in L$ tal que la distància entre v i w no sigui major que $\psi(n)$, és a dir, tal que $\|v - w\| \leq \psi(n)$.*

Definició 15. *S'anomena problema de la base més curta o SBP (de l'anglès* Shortest Basis Problem*) a, donat un reticle L de dimensió n trobar-ne una base que sigui la més curta respecte una condició donada, com per exemple imposar que $\sum_{i=1}^{n} \|v_i\|^2$ sigui el més petita possible, on els vectors $v_1, ..., v_n$ formen una base de L.*

2.2.2 Fites i heurístiques

En relació amb els problemes enunciats, sembla lògic preguntar-se sobre la longitud del vector més curt d'un reticle. Doncs bé, el teorema d'Hermite ofereix una cota superior per aquest vector en funció del determinant i de la dimensió del reticle.

Teorema 16 (Hermite)**.** *Tot reticle L de dimensió n conté un vector $v \in L$ diferent de zero que satisfà $\|v\| \leq \sqrt{n}\ \det(L)^{1/n}$.*

La demostració d'aquest teorema requereix d'eines externes a les vistes en aquest treball i es pot trobar a [HPS08]. Així mateix, convé assenyalar que existeixen versions

alternatives del teorema d'Hermite. Entre elles, destaca la següent que ens permet complementar la Proposició 10, corresponent a la desigualtat de Hadamard .

Teorema 17 (Versió alternativa d'Hermite)**.** *Tot reticle L de dimensió n conté una base $\{v_1, ..., v_n\} = V$ tal que $\|v_1\| ... \|v_n\| \leq \det(L) n^{n/2}$.*

A partir d'aquesta desigualtat es defineix el que es coneix com a relació de Hadamard, un valor acotat entre 0 i 1 que ens permet quantificar com d'ortogonal és una base d'un reticle.

Definició 18. *Donada una base $\{v_1, ..., v_n\} = V$ d'un reticle L la relació de Hadamard $\mathcal{H}(V)$ és*

$$\mathcal{H}(V) = \left(\frac{\det(L)}{\|v_1\| ... \|v_n\|} \right)^{1/n}. \tag{2.5}$$

Com més proper a 1 sigui aquest valor, més ortogonal és la base.

Definició 19. *Sigui n una dimensió qualsevol. La constant d'Hermite γ_n és el menor valor tal que per tot reticle L de dimensió n existeix $v \in L$ amb $\|v\|^2 \leq \gamma_n \ \det(L)^{2/n}$.*

Una altra conseqüència directa del teorema d'Hermite és que $\gamma_n \leq n$ per a tot n. De fet, en criptografia, al treballar amb reticles de dimensió considerable únicament interessa el valor de γ_n per valors de n grans. En aquests casos, es coneix que $\frac{n}{2\pi e} \leq n \leq \frac{n}{\pi e}$. Per més informació sobre aquesta cota veure [Ngu09].

Per finalitzar aquesta secció, convé puntualitzar que la cota oferta pel teorema d'Hermite es pot millorar mitjançant hiperesferes i la funció gamma Γ. És el que es coneix com a heurística Gaussiana.

Definició 20. *Donat un reticle L de dimensió n, la longitud Gaussiana mínima esperada és*

$$\sigma(L) = \sqrt{\frac{n}{2\pi e}} (\det(L))^{1/n}. \tag{2.6}$$

Més concretament, l'heurística Gaussiana indica que un dels vectors més curts diferents de zero v d'un reticle escollit aleatòriament compleix $\|v\| \approx \sigma(L)$. A més a més, també permet estimar que donat un vector $w \in \mathbb{R}^n$ qualsevol la distància al vector $u \in L$ més proper a w satisfà $\|w - u\| \approx \sigma(L)$

Intuïtivament, aquesta heurística surt de calcular el radi mínim que hauría de tenir aproximadament una hiperesfera centrada en l'origen per poder assegurar que inclou un punt del reticle. Aquests càlculs fan ús de resultats com ara el teorema de Minkowski i es poden trobar amb tot detall a [HPS08].

La seva utilitat rau en què serveix de referència per ponderar la dificultat del SVP d'un reticle concret. En aquest sentit, si s'és coneixedor de l'existència d'un vector de L molt més curt que $\sigma(L)$, vol dir que el reticle disposa d'un vector inusualment curt, de tal manera que els algorismes que resolen el SVP rendiran correctament. Per contra, si es coneix que el vector més curt de L és força més gran que $\sigma(L)$, els algorismes tindran majors dificultats per obtenir resultats acceptables.

2.3 Algorismes de reticles

Un cop presentats alguns dels resultats essencials de la teoria de reticles, en aquest apartat s'estudiaran en profunditat diversos algorismes que recullen i empren tots aquests conceptes. Aquests algorismes permeten donar resposta, ja sigui total o aproximada, a alguns dels problemes vistos anteriorment com per exemple el apprCVP. De fet, hom pot veure'ls com el nexe d'unió entre la vessant més abstracta de la teoria de reticles i les seves aplicacions en diversos criptosistemes que es detallaran en seccions posteriors.

2.3.1 La importància de les bases ortogonals

En primer lloc, donat un reticle $L \subset \mathbb{R}^n$ de dimensió n, convé notar el gran avantatge que suposa disposar d'una base $\{v_1, ..., v_n\} = V$ amb els vectors dos a dos ortogonals de cara a la resolució del SVP i del CVP.

D'una banda, per tal de solucionar el SVP en el reticle L, prenent la base ortogonal V anterior, n'hi ha prou prou amb veure que donat qualsevol vector $w \in L$ la seva longitud al quadrat es correspon a

$$\|w\|^2 = \|a_1 v_1 + ... + a_n v_n\|^2 = a_1^2 \|v_1\|^2 + ... + a_n^2 \|v_n\|^2$$

i com que $a_1, ..., a_n \in \mathbb{Z}$, és clar que el vector més curt de L es correspon amb el vector més curt del conjunt $\{\pm v_1, ..., \pm v_n\}$.

D'altra banda, donat qualsevol vector $w \in \mathbb{R}^n$, per tal de solucionar el CVP en el mateix reticle L i prenent la mateixa base ortogonal V, es pot raonar de forma similar al vist en el cas del SVP. Primerament, s'expressa w com a combinació lineal de la base V, és a dir,

$$w = k_1 v_1 + ... + k_n v_n, \text{ amb } k_1, ..., k_n \in \mathbb{R}.$$

Tot seguit, cal buscar el vector $u \in L$ més proper a w. Així doncs, s'expressa u com un vector qualssevol del reticle L,

$$u = a_1 v_1 + ... + a_n v_n, \text{ amb } a_1, ..., a_n \in \mathbb{Z}.$$

Finalment, únicament resta trobar els valors enters de $a_1, ..., a_n$ per tal que $\|w - u\|$ sigui el més petita possible. Aprofitant altre cop que els vectors de la base V són dos a dos ortogonals s'obté

$$\|w - u\|^2 = \|(k_1 - a_1)v_1 + ... + (k_n - a_n)v_n\|^2 = (k_1 - a_1)^2 \|v_1\|^2 + ... + (k_n - a_n)^2 \|v_n\|^2 .$$

Per tant, el vector $u \in L$ més proper a w s'aconsegueix prenent com a coeficients $a_1, ..., a_n$ els enters més propers a $k_1, ..., k_n$.

No obstant això, és important remarcar el fet que ambdós raonaments funcionen gràcies al fet que la base V és ortogonal, d'aquí la seva importància. A continuació, es presenta l'algorisme de Babai, el qual mecanitza el procés de resolució del CVP acabat de detallar.

Teorema 21 (Algorisme de Babai)**.** *Sigui $L \subset \mathbb{R}^n$ un reticle amb una base $\{v_1, ..., v_n\} = V$ i sigui $w \in \mathbb{R}^n$ un vector qualsevol. Si els vectors de la base V són ortogonals dos a dos, aleshores el següent algorisme resol el CVP.*

Algorisme 1 Algorisme de Babai

1: $w \leftarrow k_1 v_1 + ... + k_n v_n$ ▷ Expressar w com a combinació lineal de la base V
2: **for** $1 \leq i \leq n$ **do**
3: $\quad a_i \leftarrow \lfloor k_i \rceil$
4: **end for**
5: **return** $u \leftarrow a_1 v_1 + ... + a_n v_n$

És important notar que, malgrat que l'algorisme de Babai requereix que la base sigui totalment ortogonal per funcionar correctament, més endavant es mostrarà que es pot relaxar aquesta condició al combinar-lo amb altres algorismes. En línies generals, si els vectors de la base són força ortogonals l'algorisme de Babai dona resposta a l'apprCVP, però no garanteix que la solució coincideixi amb la del CVP. Una manera ràpida de comprovar el grau d'ortogonalitat dels vectors de la base és calcular la seva relació de Hadamard (2.5) i mirar si es tracta d'un valor proper a 1.

Pel vist fins ara, és clar que la clau per a poder resoldre tant el SVP com el CVP de qualsevol reticle és disposar d'una base adequada. En línies generals, es considera que una base d'un reticle és reduïda quan els seus vectors són curts i pràcticament ortogonals dos a dos. Si bé és cert que es tracta d'una definició qualitativa que no precisa com de curts i ortogonals han de ser els vectors entre ells, el motiu que hi ha al darrere d'aquest fet és que aquestes condicions han de ser més o menys restrictives en funció de diversos factors com ara la dimensió del reticle o el grau de precisió requerit. En aquest sentit, al llarg d'aquesta secció es detallaran la definició de base reduïda per algunes situacions específiques i un seguit d'algorismes enfocats a la reducció de reticles.

2.3.2 Algorisme de Gauss

En dimensió 2 es coneix un algorisme capaç d'obtenir una base reduïda d'un reticle partint d'una base qualsevol en temps polinomial. És el que es coneix com a algorisme de Gauss, malgrat que el primer matemàtic en presentar-lo va ser Joseph Louis Lagrange l'any 1773.

A grans trets, la idea principal de l'algorisme de Gauss és la d'anar restant múltiples del vector més curt de la base a l'altre vector fins no poder adequar-la més. Més concretament, el múltiple m del vector curt de la base a es tria de tal manera que el vector resultant de la resta sigui el més ortogonal possible al vector curt. De fet, aquest valor m es correspon amb l'arrodoniment enter de l'operador de projecció de Gram-Schmidt [Wen17]. A mode d'exemple, donada una base $\{v_1, v_2\} = V$ d'un reticle $L \subset \mathbb{R}^2$ amb $\|v_1\| < \|v_2\|$, en la primera iteració es substituiria el vector v_2 pel vector $v_2' = v_2 - mv_1$, on $m = \left\lfloor \frac{v_1 \cdot v_2}{\|v_1\|^2} \right\rceil$.

Teorema 22 (Algorisme de Gauss)**.** *Sigui $L \subset \mathbb{R}^2$ un reticle 2-dimensional i sigui $\{v_1, v_2\} = V$ una base de L. Aleshores el següent algorisme sempre finalitza en un nombre de passos finit i el vector v_1 és un dels vectors més curts de L. A més a més, l'angle θ entre els vectors v_1 i v_2 compleix $|\cos\theta| \leq \frac{\|v_1\|}{2\|v_2\|}$.*

Algorisme 2 Algorisme de Gauss

Require: $m \geq 0$
Ensure: $\{v_1, v_2\}$ base del reticle L amb $\|v_2\| < \|v_1\|$.
1: **while** $m \neq 0$ **do**
2: **if** $\|v_2\| < \|v_1\|$ **then**
3: Intercanviar v_1 i v_2.
4: **end if**
5: $m \leftarrow \left\lfloor \frac{v_1 \cdot v_2}{\|v_1\|^2} \right\rceil$
6: $v_2 \leftarrow v_2 - m v_1$
7: **end while**
8: **return** v_1, v_2

Demostració. Primerament, cal veure que l'algorisme finalitza sempre. Tal i com s'observa en l'enunciat del teorema, l´única manera de sortir del bucle i acabar amb l'execució de l'algorisme és aconseguir que $m = 0$.

Per tal de demostrar-ho, convé notar que es pot reduir el problema a veure que

$$\text{si } m \neq 0 \text{ aleshores } \left\|v_2'\right\| < \|v_2\|, \tag{2.7}$$

ja que en aquest cas quedarà provat que en cada iteració on l'algorisme no finalitza la base del reticle es fa cada vegada més petita, cosa que no pot succeir de manera indefinida degut a que la Desigualtat de Hadamard (2.4) fita inferiorment la mida de les bases del reticle. En altres paraules, si s'és capaç de veure (2.7), tard o d'hora l'algorisme de Gauss desemboca a $m = 0$, finalitzant així la seva execució.

Així doncs, suposant que $m \neq 0$ el primer que s'observa de la igualtat $m = \left\lfloor \frac{v_1 \cdot v_2}{\|v_1\|^2} \right\rceil$ és que el signe de m és el mateix que el de $v_1 \cdot v_2$. A més a més, sense perdre generalitat es pot prendre $m > 0$ i $v_1 \cdot v_2 > 0$, ja que l'argument que segueix és anàleg pel cas negatiu.

Fet aquest apunt, desenvolupant $\|v_2'\|^2$ s'obté que

$$\left\|v_2'\right\|^2 = \|v_2 - m v_1\|^2 = \|v_2\|^2 - 2m v_1 \cdot v_2 + m^2 \|v_1\|^2.$$

Per tant, per provar que $\|v_2'\| < \|v_2\|$ és suficient amb veure que

$$-2m v_1 \cdot v_2 + m^2 \|v_1\|^2 < 0,$$

i tenint present que $m > 0$, la darrera desigualtat és equivalent a

$$m \|v_1\|^2 < 2 v_1 \cdot v_2. \tag{2.8}$$

Per tal de provar (2.8) és convenient separar-ne l'estudi en dos casos:

1. Cas $m \leq \frac{v_1 \cdot v_2}{\|v_1\|^2}$. Es veu clarament que $m \|v_1\|^2 < v_1 \cdot v_2 < 2 v_1 \cdot v_2$.
2. Cas $m > \frac{v_1 \cdot v_2}{\|v_1\|^2}$. Per la definició de m es veu que $m - \frac{1}{2} < \frac{v_1 \cdot v_2}{\|v_1\|^2}$. De tal manera que multiplicant a ambdues bandes per $\|v_1\|^2$ s'obté

$$m \|v_1\|^2 < v_1 \cdot v_2 + \frac{\|v_1\|^2}{2}.$$

Per tant, només resta veure que $\frac{\|v_1\|^2}{2} < v_1 \cdot v_2$, la qual cosa és certa ja que s'ha suposat $m \neq 0$.

Així doncs, queda demostrada la condició (2.7) i, en consequència, que l'algorisme de Gauss finalitza.

D'ara en endavant, un cop vist que l'algorisme sempre finalitza, pel que resta de demostració els vectors v_1 i v_2 es correspondran als vectors resultants de l'execució de l'algorisme de Gauss.

D'aquesta manera, per provar la desigualtat $|\cos\theta| \leq \frac{\|v_1\|}{2\|v_2\|}$ n'hi ha prou amb ajuntar les dues expressions següents. D'una banda, és un resultat conegut de geometria que

$$|\cos\theta| = \frac{v_1 \cdot v_2}{\|v_1\| \, \|v_2\|}.$$

D'altra banda, com que l'algorisme de Gauss ha finalitzat, es té $m = 0$ i, per tant, és clar que

$$\frac{|v_1 \cdot v_2|}{\|v_1\|^2} \leq \frac{1}{2}. \tag{2.9}$$

D'aquesta manera, combinant ambdues expressions s'obté $|\cos\theta| \leq \frac{\|v_1\|}{2\|v_2\|}$.

Finalment, únicament resta veure que el vector v_1 és un dels vectors més curts del reticle. Altre cop, com que l'algorisme ha finalitzat es disposa de la desigualtat (2.9) i es té

$$\|v_1\| \leq \|v_2\|. \tag{2.10}$$

Sigui $v \in L$ un vector diferent de 0. Es pot expressar v com la combinació lineal de v_1 i v_2, és a dir,

$$v = a_1 v_1 + a_2 v_2 \text{ amb } a_1, a_2 \in \mathbb{Z}.$$

Desenvolupant aquesta expressió s'obté

$$\begin{aligned}
\|v\|^2 = \|a_1 v_1 + a_2 v_2\|^2 = a_1^2 \|v_1\|^2 + 2a_1 a_2 (v_1 \cdot v_2) + a_2^2 \|v_2\|^2 \geq \\
\geq a_1^2 \|v_1\|^2 - 2|a_1 a_2||v_1 \cdot v_2| + a_2^2 \|v_2\|^2 \geq \text{ per (2.9)} \\
\geq a_1^2 \|v_1\|^2 - |a_1 a_2| \|v_1\|^2 + a_2^2 \|v_2\|^2 \geq \text{ per (2.10)} \\
\geq a_1^2 \|v_1\|^2 - |a_1 a_2| \|v_1\|^2 + a_2^2 \|v_1\|^2 = (a_1^2 - |a_1||a_2| + a_2^2) \|v_1\|^2.
\end{aligned}$$

Tenint en compte que els valors $a_1, a_2 \in \mathbb{Z}$ i que almenys un d'ells és diferent de 0, és clar que el coeficient de la darrera expressió és més gran que 1. Per tant, queda demostrat que el vector v_1 és un dels vectors més curts del reticle L. □

2.3.3 Algorisme LLL

Un cop vist l'algorisme de Gauss, la pregunta que segueix de forma instintiva és si es pot generalitzar a dimensions superiors. En aquest sentit, a mesura que augmenta la dimensió del reticle L, l'algorisme de Gauss presenta diverses problemàtiques que el fan impossible de replicar com a tal. Entre elles, destaca que per a escollir la combinació lineal

de vectors que permeti realitzar la reducció cal resoldre el CVP a un subreticle de L. Una explicació més detallada d'aquestes problemàtiques es pot trobar a [Gal12].

No obstant això, flexibilitzant les condicions referents a la base reduïda sí que es pot recollir la idea present a l'algorisme de Gauss i adaptar-la per obtenir resultats interessants. Precisament això és el que van fer els matemàtics Lenstra, Lenstra i Lovász amb la publicació l'any 1982 de l'algorisme LLL, el qual permet obtenir una base reduïda en temps polinomial i juga un paper cabdal en la criptografia de reticles.

Abans d'analitzar detalladament l'algorisme LLL, convé presentar alguns conceptes i resultats sobre els que es sustenta i que permeten demostrar-ne el correcte funcionament. El primer de tots ells fa referència al procés d'ortogonalització de Gramm-Schmidt.

Teorema 23 (Ortogonalització de Gramm-Schmidt)**.** *Sigui $\{v_1, ..., v_n\} = V$ una base d'un espai vectorial $E \subset \mathbb{R}^m$, aleshores el següent procés genera una base ortogonal $v'_1, .., v'_n$ de E amb la propietat que $< v'_1, .., v'_i >=< v_1, .., v_i >$ per tota $i = 1, ..., n$.*

Algorisme 3 Ortogonalització de Gramm-Schmidt

Ensure: $\{v_1, ..., v_n\}$ base del reticle L.

$v'_1 \leftarrow v_1$
for $2 \leq i \leq n$ **do**
  for $1 \leq j < i$ **do**
    $s_{ij} \leftarrow \frac{v_i \cdot v'_j}{\left\|v'_j\right\|^2}$
  end for
  $v'_i \leftarrow v_i - \sum_{j=1}^{i-1} s_{ij} v'_j$
end for
return $v'_1, ..., v'_n$ ▷ Base ortogonal

Demostració. Per veure l'ortogonalitat de la nova base es fa ús del mètode d'inducció. Per construcció, és clar que v'_2 és ortogonal a v'_1. Acte seguit, suposant que els vectors $v'_1, ..., v'_{i-1}$ són dos a dos ortogonals, cal veure que el vector v_i també és ortogonal amb tots ells. Sigui $k < i$ qualsevol,

$$v'_i \cdot v'_k = \left(v_i - \sum_{j=1}^{i-1} s_{ij} v'_j \right) \cdot v'_k = v_i \cdot v'_k - s_{ik} \left\|v'_k\right\|^2 = 0.$$

On a la segona igualtat s'empra el fet que $v'_j \cdot v'_k = 0$ per $j \neq k$ i la darrera igualtat és certa per construcció de s_{ik}.

Finalment, resta provar la igualtat referent als subespais d'ambdues bases. D'una banda, la inclusió $< v_1, .., v_i > \subset < v'_1, .., v'_i >$ és clara per construcció dels v'_i. Per tant, únicament cal veure que $< v'_1, .., v'_i > \subset < v_1, .., v_i >$. Fent ús d'inducció, el cas per $i = 1$ és clar. Així doncs, suposant que $< v'_1, .., v'_{i-1} > \subset < v_1, .., v_{i-1} >$, de la definició de v'_i se'n deriva que $v'_i \in < v'_1, .., v'_{i-1}, v_i >$ i, per tant, clarament $< v'_1, .., v'_i > \subset < v_1, .., v_i >$ com volíem veure. □

És important precisar que el teorema anterior fa referència a espais vectorials i no a reticles. Així doncs, com a norma general el reticle generat per una base $\{v_1, ..., v_n\} = V$

no coincideix amb el reticle generat per la base que deriva de V pel procés de Gramm-Schmidt, ja que els coeficients s_{ij} no solen ser enters. Ara bé, la proposició següent demostra que ambdós reticles tenen el mateix determinant.

Proposició 24. *Sigui $\{v_1, ..., v_n\} = V$ una base d'un reticle L de dimensió n i sigui $\{v'_1, ..., v'_n\} = V'$ la seva base de Gramm-Schmidt associada. Aleshores* $\det(L) = \prod_{i=1}^{n} \|v'_i\|$.

Demostració. Sigui A la matriu que té per columnes les coordenades dels vectors de la base V. La Proposició 8 demostra que $\det(L) = |\det(A)|$. Sigui A' la matriu que té per columnes les coordenades dels vectors de la base V'. Seguint el procés d'ortogonalització de Gramm-Schmidt s'obté la següent relació matricial

$$A = A'M,$$

on M es correspon amb la següent matriu

$$M = \begin{pmatrix} 1 & s_{2,1} & s_{3,1} & \dots & s_{n-1,1} & s_{n,1} \\ 0 & 1 & s_{3,2} & \dots & s_{n-1,2} & s_{n,2} \\ 0 & 0 & 1 & \dots & s_{n-1,3} & s_{n,3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & 1 & s_{n,n-1} \\ 0 & 0 & 0 & \dots & 0 & 1 \end{pmatrix}.$$

És clar que $\det(M) = 1$ per ser matriu triangular amb la diagonal plena de 1. Per tant,

$$\det(L) = |\det(A)| = |\det(A'M)| = |\det(A')\det(M)| = |\det(A')| = \prod_{i=1}^{n} \|v'_i\|,$$

on a la darrera igualtat s'empra que les columnes de A' són dos a dos ortogonals. □

Acte seguit, convé formalitzar el concepte de base reduïda de cara a l'algorisme LLL.

Definició 25. *Sigui $\delta \in (\frac{1}{4}, 1)$, sigui $\{v_1, ..., v_n\} = V$ una base d'un reticle L i sigui $\{v'_1, ..., v'_n\} = V'$ la base de Gramm-Schmidt construïda a partir de V seguint el teorema (23). La base V es diu que és una base δ-LLL reduïda si satisfà les dues condicions següents:*

- *Condició de mida:* $|s_{ij}| = \frac{|v_i \cdot v'_j|}{\|v'_j\|^2} \leq \frac{1}{2}$ *per tot* $1 \leq j < i \leq n$.
- *Condició de Lovász:* $\|v'_i\|^2 \geq (\delta - s_{ii-1}^2) \|v'_{i-1}\|^2$ *per tot* $1 < i \leq n$.

D'una banda, és clar que la condició de mida és anàloga al còmput de m que s'usa en l'algorisme de Gauss 22 i que, tal i com succeeix en dimensió 2, permet controlar la mida i ortogonalitat dels vectors de la base. D'altra banda, la condició de Lovász és la generalització n-dimensional de la desigualtat $\|v_2\| \geq \|v_1\|$ també present l'algorisme de Gauss 22 i s'encarrega de controlar l'ordenació creixent dels vectors de la base.

Per tal de donar una breu intuició sobre la condició de Lovász, convé notar que la desigualtat $\|v_2\| \geq \|v_1\|$ és equivalent a $\|v'_2\|^2 \geq (1 - s_{21}) \|v'_1\|^2$, de tal manera que al canviar el 1 per una constant $\delta \in (\frac{1}{4}, 1)$ s'aconsegueix relaxar la condició. De fet, geomètricament

parlant, la condició de Lòvasz comprova que la norma de la projecció del vector v_{i-1} a l'espai generat pels vectors $v'_1, ..., v'_{i-2}$ sigui δ vegades menor o igual que la norma de la projecció del vector v_i a aquest mateix espai. És a dir,

$$\left\|\text{projecció de } v_i \text{ a } < v'_1, ..., v'_{i-2} >\right\| \geq \delta \left\|\text{projecció de } v_{i-1} \text{ a } < v'_1, ..., v'_{i-2} >\right\|.$$

La següent proposició mostra algunes propietats interessants de les bases δ-LLL reduïdes. En particular, la quarta desigualtat prova que permeten resoldre el apprSVP amb un factor de $\left(\frac{1}{\delta-\frac{1}{4}}\right)^{(n-1)/2}$.

Teorema 26. *Sigui L un reticle de dimensió n. Qualsevol base δ-LLL reduïda $\{v_1, ..., v_n\} = V$ amb la seva base de Gramm-Schmidt associada $\{v'_1, ..., v'_n\} = V'$ té les següents propietats:*

1. $\prod_{i=1}^{n} \|v_i\| \leq \left(\frac{1}{\delta-\frac{1}{4}}\right)^{(n-1)/4} \det(L)$.
2. $\|v_j\| \leq \left(\frac{1}{\delta-\frac{1}{4}}\right)^{(i-1)/2} \|v'_i\|$ *per tot* $1 \leq j \leq i \leq n$.
3. $\|v_1\| \leq \left(\frac{1}{\delta-\frac{1}{4}}\right)^{(n-1)/4} \det(L)^{1/n}$.
4. $\|v_1\| \leq \left(\frac{1}{\delta-\frac{1}{4}}\right)^{(n-1)/2} \min_{0 \neq v \in L} \|v\|$.

Demostració. Tenint en compte les condicions de mida i de Lovász de la base V, és clar que

$$\|v'_i\|^2 \geq \left(\delta - s^2_{ii-1}\right) \|v'_{i-1}\|^2 \geq \left(\delta - \frac{1}{4}\right) \|v'_{i-1}\|^2$$

i aplicant successivament aquest mateix argument es conclou que

$$\left(\frac{1}{\delta - \frac{1}{4}}\right)^{i-j} \|v'_i\|^2 \geq \left\|v'_j\right\|^2. \tag{2.11}$$

D'ara en endavant, amb l'objectiu de simplificar la notació es pren $\alpha = \left(\frac{1}{\delta-\frac{1}{4}}\right)$. Acte seguit, cal buscar una fita superior per $\|v_i\|^2$. Desenvolupant l'expressió, aprofitant l'ortogonalitat de la base V', la condició de mida, la desigualtat acabada de provar (2.11) i tenint present que $4(\alpha - 1) \geq \alpha$ s'obté

$$\begin{aligned} \|v_i\|^2 = \left\|v'_i + \sum_{j=1}^{i-1} s_{ij} v'_j\right\|^2 = \|v'_i\|^2 + \sum_{j=1}^{i-1} s^2_{ij} \left\|v'_j\right\|^2 \leq \\ \leq \|v'_i\|^2 + \sum_{j=1}^{i-1} \frac{1}{4} \left\|v'_j\right\|^2 \leq \|v'_i\|^2 + \sum_{j=1}^{i-1} \frac{1}{4} \alpha^{i-j} \|v'_i\|^2 \leq \\ \leq \left(1 + \frac{1}{4} \sum_{k=1}^{i-1} \alpha^k\right) \|v'_i\|^2 \leq \alpha^{i-1} \|v'_i\|^2. \end{aligned} \tag{2.12}$$

Multiplicant aquesta darrera desigualtat per tot $i = 1, ..., n$ i tenint present la Proposició 24 s'aconsegueix

$$\prod_{i=1}^{n} \|v_i\|^2 \leq \prod_{i=1}^{n} \alpha^{i-1} \left\|v_i'\right\|^2 = \alpha^{n(n-1)/2} \prod_{i=1}^{n} \left\|v_i'\right\|^2 = \alpha^{n(n-1)/2} \det(L)^2,$$

d'on prenent l'arrel quadrada es demostra la propietat 1.

Per tal de demostrar la propietat 2, n'hi ha prou amb partir altre cop de la desigualtat (2.12), considerar $j \leq i$ i usar la desigualtat (2.11) per obtenir

$$\left\|v_j\right\|^2 \leq \alpha^{j-1} \left\|v_j'\right\|^2 \leq \alpha^{i-1} \left\|v_i'\right\|^2,$$

d'on prenent arrels quadrades de nou queda demostrada la propietat 2.

La propietat 3 es demostra de forma molt similar a la propietat 1, únicament cal partir de la desigualtat de la propietat 2 fixant $j = 1$ i procedir de forma anàloga a la desigualtat (2.12), amb l'única diferència que enlloc d'arrels quadrades al final cal prendre arrels enèssimes.

Finalment, cal demostrar la propietat 4. En aquest cas, s'expressa un vector $w \in L$ qualsevol com a combinació lineal de les bases V i V', és a dir, $w = \sum_{j=1}^{i} a_j v_j = \sum_{j=1}^{i} a_j' v_j'$ de tal manera que a_i sigui el darrer coeficient de w diferent de 0. Per construcció de les bases V i V' s'obté que $a_i = a_i'$ i com que $a_i \neq 0$ és enter, resulta que $|a_i| \geq 1$. Per tant, tenint això present, fent ús de la propietat 2 amb $j = 1$ i desenvolupant l'expressió de $\|w\|^2$ s'obté

$$\|w\|^2 = \sum_{j=1}^{i} a_j'^2 \left\|v_j'\right\|^2 \geq a_i'^2 \left\|v_i'\right\|^2 \geq \left\|v_i'\right\|^2 \geq \alpha^{-(i-1)} \left\|v_1'\right\|^2 \geq \alpha^{-(n-1)} \left\|v_1'\right\|^2,$$

d'on prenent arrels quadrades de nou queda demostrada la propietat 4. □

En referència a la proposició anterior, convé parar-se a reflexionar breument al voltant del factor d'aproximació que ofereix de cara a la resolució de l'apprCVP. Tenint en compte que $\left(\frac{1}{\delta - \frac{1}{4}}\right) \geq \frac{4}{3}$, és clar que es tracta d'un factor exponencial que creix a mesura que augmenta la dimensió del reticle n. Per tant, en reticles de dimensions molt grans la solució de l'apprCVP obtinguda gràcies a aquesta proposició sovint passa a ser trivial i sense valor pràctic.

D'ara en endavant, es fixarà el valor de $\delta = \frac{3}{4}$ degut a que és la xifra més comuna en l'àmbit de la criptografia i per tal de simplificar els resultats que se'n deriven. De fet, $\frac{3}{4}$ és precisament el valor present en la primera aparició de l'algorisme LLL l'any 1982 a [LLL82], motiu pel qual s'acostuma a prendre $\frac{3}{4}$ com a valor estàndard de δ. Endemés, és interessant puntualitzar que a dia d'avui encara es desconeix si per $\delta = 1$ l'algorisme LLL finalitza en temps polinomial.

Teorema 27 (Algorisme LLL)**.** *Sigui $\{v_1, ..., v_n\} = V$ una base d'un reticle L amb la seva base de Gramm-Schmidt associada $\{v_1', ..., v_n'\} = V'$. Aleshores el següent algorisme finalitza en un nombre finit de passos i retorna una base $\frac{3}{4}$-LLL reduïda de L. Més concretament, prenent $B = \max_{1 \leq i \leq n} \|v_i\|$, l'algorisme executa el bucle principal (passos 2-14) un màxim de $\mathcal{O}(n^2 \log n + n^2 \log B)$ vegades. En particular, l'algorisme LLL és un algorisme de temps polinomial.*

Algorisme 4 Algorisme LLL

Ensure: $\{v_1, ..., v_n\}$ base del reticle L.
Require: $k = 2$
1: $v'_1 \leftarrow v_1$
2: **while** $k \leq n$ **do**
3: **for** $1 \leq j < k$ **do**
4: $v_k \leftarrow v_k - \left\lfloor s_{kj} \right\rceil v'_j$ ▷ Reducció de mida
5: Recalcular la base V'.
6: **end for**
7: **if** $\|v'_k\|^2 \geq \left(\frac{3}{4} - s^2_{kk-1}\right) \|v'_{k-1}\|^2$ **then** ▷ Condició de Lovász
8: $k \leftarrow k + 1$
9: **else**
10: Intercanviar v_{k-1} i v_k.
11: Recalcular la base V'.
12: $k \leftarrow \max(k - 1, 2)$
13: **end if**
14: **end while**
15: **return** $v'_1, ..., v'_n$ ▷ Base $\frac{3}{4}$-LLL reduïda

En línies generals, hom pot considerar l'algorisme LLL una generalització n-dimensional de l'Algorisme de Gauss 22, ja que la idea de fons és la mateixa: reduïr, ortogonalitzar i reordenar una base V donada. En aquest cas, en el bucle del pas 3 l'algorisme LLL procedeix de forma anàloga a l'algorisme de Gauss per tal de reduir i ortogonalitzar els vectors de V, si bé cal tenir present que en aquest cas la dimensió pot ser molt superior. Pel que fa a la correcta ordenació de la base V, aquesta característica es comprova mitjançant la condició de Lovász al pas 7 explicada prèviament.

Un altre aspecte a tenir en compte referent a l'algorisme LLL és la necessitat d'actualitzar els vectors de la base de Gramm-Schmidt associada V' sempre que es modifiquin les coordenades o l'ordre d'algun vector de la base V, tal i com s'observa als passos 5 i 11. No obstant això, el cert és que aquests càlculs no intervenen en la demostració del Teorema 27.

Abans d'entrar en la demostració del Teorema 27, és interessant presentar breument dues de les variants proposades de l'algorisme LLL d'ençà la seva aparició l'any 1982. D'una banda, existeix el que es coneix com *deep insertion method*, que es correspon a una modificació en la manera d'intercanviar els vectors de la base. Mentre que a l'algorisme LLL presentat al Teorema 27 únicament intercanvia els vectors v_{k-1} i v_k, aquesta modificació de l'algorisme afegeix un vector v_i entre els vectors v_{k-1} i v_k de tal manera que es maximitzi la reducció de la base al duur a terme l'intercanvi. D'altra banda, existeix la coneguda com a variant BKZ-LLL (de l'anglès *Block Korkin Zolotarev*), la qual es fonamenta en el concepte de base KZ-reduïda, una alternativa a les bases δ-LLL reduïdes que destaca per les seves propietats i que és obtinguda al finalitzar l'execució de l'algorisme BKZ-LLL. Entre les seves propietats, destaca el fet que el vector de menor mida d'una base KZ-reduïda sempre es correspon amb la solució del SVP. Per veure una explicació

detallada d'aquestes i altres variants és recomanable consultar [SE94] i [Sch09].

Demostració. En primer lloc, convé precisar que per motius de practicitat aquesta demostració es restringeix al cas en el que el reticle L és enter, és a dir, $L \subset \mathbb{Z}^n$.

En primer lloc, és clar que en cas que l'algorisme LLL finalitzi, la base V resultant serà una base $\frac{3}{4}$-LLL reduïda degut a que els passos 4 i 7 garanteixen que es compleixin les condicions inherents a les bases $\frac{3}{4}$-LLL reduïdes.

A continuació, cal demostrar que l'algorisme finalitza, cosa que no és trivial degut a que l'increment del comptador del bucle principal al pas 8 es veu contrarrestat per la reducció del mateix comptador al pas 12. Així doncs, és suficient amb veure que el pas 12 s'executa un nombre finit de cops per provar que l'algorisme finalitza, ja que d'aquesta manera es garanteix que a la llarga el valor del comptador sobrepassi n i s'acabi el bucle principal.

Sigui $\{v_1, ..., v_n\} = V$ una base del reticle L amb la seva base de Gramm-Schmidt associada $\{v'_1, ..., v'_n\} = V'$ i sigui L_h el subreticle generat per $\{v_1, ..., v_h\} = V_h$ amb $h = 1, ..., n$. Es defineixen les quantitats següents:

$$d_h = \prod_{i=1}^{h} \|v'_i\|^2 \text{ i } D = \prod_{h=1}^{n} d_h = \prod_{i=1}^{n} \|v'_i\|^{2(n+1-i)}.$$

Tot seguit, convé notar que el valor de D es veu alterat en el pas 10, ja que a l'intercanviar els vectors v_{k-1} i v_k la quantitat d_{k-1} es veu modificada degut a que cal substituir al productori el valor de $\|v'_{k-1}\|^2$ per $\|v'_k\|^2$. Entrant en detall, es pot quantificar aquesta modificació del valor d_{k-1} tenint present que, quan es duu a terme, la condició de Lovász no es compleix, de tal manera que

$$\|v'_k\|^2 < \left(\frac{3}{4} - s^2_{kk-1}\right) \|v'_{k-1}\|^2.$$

Així doncs, desenvolupant l'expressió s'obté

$$d^{\text{nou}}_{k-1} = \|v'_1\|^2 \cdot ... \|v'_{k-2}\|^2 \cdot \|v'_k\|^2 = \\ = \|v'_1\|^2 \cdot ... \|v'_{k-2}\|^2 \cdot \|v'_{k-1}\|^2 \cdot \frac{\|v'_k\|^2}{\left\|v'_{k-1}\right\|^2} = d^{\text{anitc}}_{k-1} \cdot \frac{\|v'_k\|^2}{\left\|v'_{k-1}\right\|^2} \leq \frac{3}{4} d^{\text{anitc}}_{k-1}.$$

Per tant, cada vegada que s'executa el pas 10 el valor de D es redueix per un factor de $\frac{3}{4}$. En altres paraules, quan el número de vegades que el pas 10 s'executa tendeix a infinit, el valor de D tendeix cap a 0. D'aquesta manera, si s'aconsegueix obtenir una fita inferior pel valor de D, és clar que s'haurà demostrat que l'algorisme finalitza, ja que voldrà dir que els passos 10 i 12 s'executen un nombre finit de vegades.

Per tal de trobar aquesta fita inferior per D convé notar que, al ser L un reticle enter, tot vector pertanyent a L diferent de zero té norma més gran o igual a 1 i fent ús del Teorema d'Hermite 16 als subreticles L_h s'obté

$$1 \leq \min_{0 \neq w \in L_h} \|v\| \leq \sqrt{h} \det(L_h)^{1/h}. \tag{2.13}$$

Seguint la Proposició 24 es pot veure que $d_h = \det(L_h)^2$, de tal manera que ajuntant-ho amb la desigualtat (2.13) i aïllant s'obté $d_h \geq h^{-h}$.

A continuació, com que es disposa d'una fita inferior per d_h, únicament resta tenir present la definició de D per obtenir

$$D = \prod_{h=1}^{n} d_h \geq \prod_{h=1}^{n} h^{-h} \geq \prod_{h=1}^{n} h^{-n} = (n!)^{-n} \geq n^{-n^2}$$

tal i com es volia.

Un cop vist que l'algorisme LLL finalitza en temps finit, per tal d'acabar de demostrar el teorema cal provar la fita superior referent al nombre d'execucions del bucle principal.

En primer lloc, és necessari establir alguns paràmetres que jugaran un paper clau en la demostració. Prenent D_{inicial} com el valor de D per la base V entrada per paràmetre a l'algorisme i D_{final} com el valor de D per la base resultant, pel vist abans en la demostració es té

$$n^{-n^2} \leq D_{\text{final}} \leq \left(\frac{3}{4}\right)^N D_{\text{inicial}}, \tag{2.14}$$

on N es correspon amb el nombre de vegades que s'executa el pas 10. A més, tenint en compte que el bucle principal s'executa un total de $2N + n$ vegades, per establir el nombre màxim de vegades que es recorre el bucle principal en tenim prou amb donar una cota superior per N.

Prenent logaritmes a la desigualtat (2.14) s'aconsegueix

$$N = \mathcal{O}(n^2 \log n + \log D_{\text{inicial}}).$$

Així doncs, només cal fer una estimació del valor de D_{inicial} per tal de finalitzar la demostració. Tenint en compte que $\|v_i'\| \leq \|v_i\|$ gràcies a com està definit l'Algorisme de Gramm-Schmidt 23, s'aconsegueix

$$D_{\text{inicial}} = \prod_{i=1}^{n} \|v_i'\|^{n+1-i} \leq \prod_{i=1}^{n} \|v_i\|^{n+1-i} \leq (\max_{1\leq i\leq n} \|v_i\|)^{2(1+2+\ldots+n)} = B^{n^2+n}.$$

Per tant, $\log(D_{\text{inicial}}) = \mathcal{O}(n^2 \log B)$ i $N = \mathcal{O}(n^2 \log n + n^2 \log B)$ tal i com es volia veure. □

Acte seguit, un cop analitzat amb detall l'algorisme LLL, és interessant relacionar-lo amb el SVP i el CVP per tal de fer-se una idea del seu potencial i importància. Al principi d'aquest apartat, s'ha fet èmfasi en el gran avantatge que suposa disposar d'una base adequada de cara a afrontar aquests problemes. Així doncs, tenint en compte que l'algorisme LLL retorna una base δ-LLL reduïda, no hauria de sorprendre que jugui un paper destacat a l'hora d'afrontar tant el SVP com el CVP.

Pel que fa al SVP, en base a les propietats vistes en la Proposició 26, és clar que l'algorisme LLL permet resoldre l'apprSVP amb un factor de $\left(\frac{1}{\delta-\frac{1}{4}}\right)^{(n-1)/2}$ sense necessitat de cap tipus de modificació o mètode auxiliar.

Per contra, per tal d'obtenir resultats útils en relació al CVP és necessari emprar l'algorisme LLL juntament amb l'Algorisme de Babai 21, el qual és capaç de resoldre el CVP

a partir de bases ortogonals. Així doncs, l'estratègia consisteix en primerament obtenir una base δ-LLL reduïda mitjançant l'algorisme LLL i, posteriorment, fer servir aquesta base per tal d'obtenir el vector solució gràcies a l'algorisme de Babai. Ara bé, és important precisar que les bases δ-LLL reduïdes no són plenament ortogonals, per la qual cosa l'algorisme de Babai no és capaç de resoldre plenament el CVP a partir d'aquest tipus de bases. De fet, aquest procediment acabat d'exposar permet obtenir solucions de l'apprCVP, tal i com enuncia el teorema següent.

Teorema 28 (aprrCVP amb algorismes LLL i de Babai)**.** *Existeix una constant C tal que per qualsevol reticle L de dimensió n donat per una base $\{v_1, ..., v_n\} = V$ el següent algorisme soluciona l'apprCVP amb un factor de C^n.*

Algorisme 5 aprrCVP amb algorismes LLL i de Babai

Ensure: $\{v_1, ..., v_n\}$ base del reticle L.
1: Aplicar l'algorisme LLL.
2: Aplicar l'algorisme de Babai.
3: **return** w ▷ Vector solució de l'apprCVP

Demostració. La demostració d'aquest teorema requereix d'eines fora de l'abast d'aquest treball i és d'una extensió considerable. Es pot trobar a [Bab86]. □

Com a apunt final, convé notar que els algorismes presentats donen solució a les versions aproximades del CVP i del SVP i en cap cas solucions exactes. El motiu darrere aquest fet rau en el fet que, en l'actualitat, els algorismes existents que resolen de manera exacta el SVP i el CVP són, en el millor dels casos, de complexitat exponencial. Així doncs, tot i que el seu resultat sigui molt precís, l'alt nivell computacional que requereixen els fa impracticables quan s'empren reticles de dimensions considerables. Bona part d'aquests algorismes empren tècniques de cribatge, un bon exemple n'és [AKS01].

Capítol 3

Anells polinomials convolucionals

3.1 Definicions i propietats bàsiques

Un cop vista amb detall la teoria de reticles, de cara a la plena comprensió del criptosistema NTRU és necessari presentar breument un altre concepte matemàtic que hi juga un paper cabdal: els anells polinomials convolucionals. Tot i tractar-se d'una branca molt extensa i potent dins de l'àlgebra abstracta, en aquest apartat únicament es descriuen de forma breu i concisa algunes de les seves propietats bàsiques. En cas que es vulgui aprofundir al voltant d'aquest tema és recomanable consultar [Jud19] o similars.

Definició 29. *Sigui N un enter positiu. S'anomena anell polinomial convolucional de rang N a l'anell quocient $R = \frac{\mathbb{Z}[x]}{(x^N-1)}$.*

Definició 30. *Siguin N i q dos enters positius. S'anomena anell polinomial convolucional de rang N mòdul q a l'anell quocient $R_q = \frac{(\mathbb{Z}/q\mathbb{Z})[x]}{(x^N-1)}$.*

A grans trets, al prendre el quocient pel polinomi $x^N - 1$ el que s'estableix és la igualtat $x^N = 1$. D'aquesta manera, es garanteix que els polinomis pertanyents a R i R_q disposin sempre de com a màxim grau $N - 1$.

A nivell de notació, per tal de simplificar la representació dels elements $p(x)$ de R i R_q, s'escriu $p(x) = (p_0, ..., p_{n-1})$, on les coordenades del vector es corresponen amb els coeficients dels termes del polinomi en sentit ascendent.

Acte seguit, convé presentar les operacions de suma i multiplicació a R i R_q. D'una banda, la suma es duu a terme de forma anàloga a la suma de polinomis habitual tant a R com a R_q, tenint present que a R_q cal sempre reduir els coeficients mòdul q. D'altra banda, la multiplicació a R i R_q ve donada per la proposició següent.

Proposició 31. *El producte de dos polinomis $a(x)$ i $b(x)$ pertanyents a R ve donat per la fòrmula $a(x) \cdot b(x) = p(x)$, amb $p_k = \sum_{i+j \equiv k \pmod N} a_i b_{k-1}$. De la mateixa manera, el producte de dos polinomis $a(x)$ i $b(x)$ pertanyents a R_q ve donat per la mateixa fòrmula a excepció del fet que els coeficients p_k cal reduir-los mòdul q.*

Demostració. Per tal de demostrar aquesta proposició únicament cal desenvolupar la multiplicació habitual de polinomis $a(x) \cdot b(x)$, fer ús de la relació $x^N = 1$ existent a R i R_q i, finalment, agrupar termes. □

Seguidament, un altre aspecte a considerar és la relació existent entre els anells R i R_q. Per la propia definició d'ambdós anells, és clar que existeix un homomorfisme $s_q : R \to R_q$ definit per $s_q(p(x)) = p(x) \pmod q$. En altres paraules, l'existència d'aquest homomorfisme és equivalent a demostrar les següents igualtats per qualssevol $a(x)$ i $b(x)$ pertanyents a R:

- $(a(x) + b(x)) \pmod q = (a(x) \pmod q) + (b(x) \pmod q)$
- $(a(x) \cdot b(x)) \pmod q = (a(x) \pmod q) \cdot (b(x) \pmod q)$

Clarament, es pot veure que s_q no és un isomorfisme degut a que el pas al quocient $\pi : \mathbb{Z} \to \mathbb{Z}/q\mathbb{Z}$ no és injectiu. Per tant, s_q no disposa d'un morfisme invers. No obstant això, per tal d'establir una espècie de camí invers que vagi de R_q a R n'hi ha prou amb prendre el morfisme $l_q : R_q \to R$ definit per $l_q(p(x)) = p'(x)$, on $p'(x)$ es correspon amb l'únic polinomi pertanyent a R tal que $p'(x) \pmod q = p(x)$ i tots els seus coeficients pertanyin a l'interval $\left(\frac{-q}{2}, \frac{q}{2}\right]$. Així doncs, d'ara en endavant s'estableixen com a representants distingits de R_q aquells polinomis que disposin de tots els coeficients en l'interval $\left(\frac{-q}{2}, \frac{q}{2}\right]$.

3.2 Invertibilitat polinomial

A continuació, es llisten un seguit de definicions i proposicions que seran d'utilitat a l'hora d'abordar el criptosistema NTRU, fent especial èmfasi en l'estudi de la invertibilitat dels elements de R i R_q.

En primer lloc, cal definir els conceptes de polinomis inversos a R i R_q.

Definició 32. *Sigui $a(x) \in R$. Es diu que $a(x)$ és invertible a R si existeix $b(x) \in R$ tal que $a(x) \cdot b(x) = 1$ a R. En aquest cas, $b(x)$ s'anomena l'invers de $a(x)$ a R.*

Definició 33. *Sigui $a(x) \in R_q$. Es diu que $a(x)$ és invertible a R_q si existeix $b(x) \in R_q$ tal que $a(x) \cdot b(x) = 1$ a R_q. En aquest cas, $b(x)$ s'anomena l'invers de $a(x)$ a R_q.*

Acte seguit, convé notar que que no sempre tots els elements d'un anell són invertibles. De fet, els anells que disposen d'inversos per tots els seus elements s'anomenen cossos. Enllaçant-ho amb això, la proposició següent aporta informació útil de cara a la detecció de polinomis invertibles a R_q.

Proposició 34. *Sigui $a(x) \in R$ tal que $a(1) \equiv 0 \pmod q$, aleshores $a(x)$ no és invertible a R_q.*

Demostració. Aquesta demostració és per contrarrecíproc. Així doncs, prenent $a(x) \in R$ invertible mòdul q és suficient veure que $a(1) \not\equiv 0 \pmod q$. En aquest sentit, com que $a(x)$ és invertible mòdul q, existeix $a(x)^{-1} \in R$ amb $a(x) \cdot a(x)^{-1} \equiv 1 \pmod q$. D'altra banda, aplicant el morfisme m: $R_q \to \mathbb{Z}/q\mathbb{Z}$ definit per avaluar el polinomi quan $x = 1$ s'obté que $a(1) \cdot a(1)^{-1} \equiv 1 \pmod q$. En particular, és clar que $a(1) \not\equiv 0 \pmod q$. □

Un cop vistos amb detall els conceptes de polinomis invertibles i d'inversos tant a R com a R_q, és lògic preguntar-se com es calculen aquests inversos. Doncs bé, quan l'anell en qüestió és un cos, s'empra el que es coneix com a algorisme d'Euclides ampliat, el qual es pot trobar a [Mag18]. D'altra banda, tal i com es veurà en apartats posteriors, de cara a la implementació del criptosistema NTRU és necessari calcular inversos a R_q amb q potència de 2. Així doncs, la proposició que segueix permet obtenir aquests inversos de manera senzilla.

Proposició 35. *Sigui q primer, $r \in \mathbb{N}$ i $a(x) \in R$. Donada la seqüència de polinomis $\{b_i(x)\}_i$ amb $i \geq 2$ definida per*

1. $a(x) \cdot b_1(x) \equiv 1 \pmod{q}$
2. $b_i(x) \equiv 2b_{i-1}(x) - a(x)b_{i-1}^2(x) \pmod{q^{2^i}}$

aleshores $a(x) \cdot b(x)_{\lceil \log_2(r) \rceil} \equiv 1 \pmod{q^r}$.

Demostració. Primerament, convé notar que és suficient amb veure que $a(x) \cdot b(x)_i \equiv 1 \pmod{q^{2^i}}$, ja que aleshores, prenent $i = \lceil \log_2(r) \rceil$, és clar que $a(x) \cdot b(x)_{\lceil \log_2(r) \rceil} \equiv 1 \pmod{q^r}$ per ser $q^{2^{\lceil \log_2(r) \rceil}}$ un múltiple de q^r. Així doncs, per tal de demostrar aquesta darrera equivalència s'empra el mètode d'inducció.

D'una banda, pel cas base $i = 2$ n'hi ha prou amb tenir presents les dues condicions que defineixen la seqüència $\{b_i(x)\}_i$ i desenvolupar el polinomi $a(x) \cdot b_2(x) - 1$ de la manera següent:

$$\begin{aligned} a(x) \cdot b_2(x) - 1 &\equiv a(x) \cdot (2b_1(x) - a(x) \cdot b_1^2(x)) - 1 \equiv \\ &\equiv 2a(x) \cdot b_1(x) - a(x)^2 \cdot b_1(x)^2 \equiv (a(x) \cdot b_1(x) - 1)^2 \equiv 0 \pmod{q^2}, \end{aligned}$$

on al darrer pas s'ha fet servir que $a(x) \cdot b_1(x) \equiv 1 \pmod{p}$.

D'altra banda, el cas general és anàleg al cas $i = 2$ amb l'excepció que en la darrera equivalència enlloc de fer servir la hipòtesi del cas $i = 1$ cal fer servir la hipòtesi d'inducció. □

Capítol 4

Criptografia basada en reticles

4.1 Introducció

En primer lloc, és important destacar que la criptografia de reticles és una àrea força nova en l'àmbit de les matemàtiques i la informàtica. De fet, no va ser fins l'any 1997 amb la publicació de [AD97] per part d'Ajtai i Dwork quan es va idear el primer criptosistema que emprava eines provinents de la teoria de reticles. Tot i que ràpidament va ser descartat per la seva impossibilitat pràctica, el seu descobriment va suposar el punt de partida d'un nou àmbit de recerca que perdura fins el dia d'avui.

Originàriament, la motivació darrere de l'estudi dels criptosistemes de reticles provenia de la voluntat d'oferir una alternativa als cripstosistemes basats en la factorització d'enters i en el logaritme discret, així com també en pro de la seva velocitat i eficiència degut al menor nombre d'operacions requerides. Ara bé, el que els hi ha otorgat una gran rellevància fins al dia d'avui ha estat un factor desconegut aleshores i és la seva aparent seguretat davant de la computació quàntica.

Enllaçant-ho amb això, la gran majoria d'aquests criptosistemes es fonamenten en la gran dificultat de resolució del CVP i del SVP. En capítols anteriors, s'ha vist de primera ma la impossibilitat pràctica de resoldre'ls més enllà de les seves versions d'aproximació. Més concretament, el desconeixement, a dia d'avui, d'algorismes que resolguin el CVP i el SVP en temps polinomial permet considerar als criptosistemes de reticles com a segurs en el si de la comunitat científica. Nogensmenys, tal i com es veurà al llarg d'aquest capítol, és cert que mitjançant l'algorisme LLL i algunes de les seves variants es poden duur a terme atacs a aquests criptosistemes amb més o menys èxit.

En referència a la seva resistència quàntica, la gran connexió entre ambdós mons data de l'any 2004 a càrrec de Regev [Reg04] i relaciona la resolució del SVP amb l'existència d'una solució al problema del subgrup amagat en el cas del grup diedral. D'aleshores ençà, no s'ha donat a conèixer cap algorisme quàntic capaç de resoldre aquest problema ni tampoc cap que millori substancialment els resultats obtinguts pels algorismes clàssics de cara a la resolució del SVP i del CVP. De fet, en base al coneixement actual de la computació quàntica, res fa pensar que en un període raonable de temps existeixin algorismes quàntics capaços de donar resposta a aquests problemes i és precisament aquesta creença

el que permet considerar la criptografia de reticles com una alternativa segura als atacs quàntics. No obstant això, és necessari tenir present que els coneixements actuals sobre computació quàntica encara disten molt dels que es disposa sobre computació clàssica, per la qual cosa cal ser prudents a l'hora d'abordar aquests temes.

Entre aquests criptosistemes destaca el conegut com a NTRU, el qual serà objecte d'estudi al llarg d'aquest apartat tant des d'una vessant teòrica com pràctica.

4.2 NTRU

L'origen del criptosistema de clau pública NTRU (de l'anglès *Nth Degree Truncated polynomial Ring Units*) data de l'any 1998 quan va ser presentat en societat per Hoffstein, Pipher i Silverman a [HPS98]. Tal i com el seu nom indica, malgrat que és concebut com un criptosistema basat en anells polinomials convolucionals, el cert és que el problema matemàtic sobre el que es sustenta es pot formular en termes de la teoria de reticles, més concretament com el SVP (de cara a la recuperació de la clau) o el CVP (de cara a la recuperació del missatge). És per aquest motiu que es considera un criptosistema de reticles. A més, és important precisar que NTRU disposa de dos algorismes independents: NTRUEncrypt (usat per encriptar i desencriptar missatges) i NTRUSign (usat en signatures digitals), si bé en aquest treball s'estudia únicament l'algorisme NTRUEncrypt.

A nivell computacional, NTRU destaca per ser un criptosistema eficient i pràctic degut a que requereix aproximadament d'$\mathcal{O}(n^2)$ operacions per tal d'encriptar i desencriptar. A tall de comparació, per dur a terme les mateixes tasques, criptosistemes tant comuns com RSA o ECC empren $\mathcal{O}(n^3)$ operacions. A més a més, és interessant puntualitzar que NTRU es tracta també d'un criptosistema probabilístic, ja que a l'hora d'encriptar el missatge utilitza un element aleatori. Tal i com es veurà més endavant, això permet disposar de diverses encriptacions possibles pel mateix missatge, però també comporta possibles errors en el procés de desencriptació.

Pel que fa a la seguretat, seguint la línia de l'esmentat en la introducció d'aquest capítol, el criptosistema NTRU no és una excepció i la seva seguretat es basa en la impossibilitat de resoldre el SVP i el CVP. Així doncs, no és d'estranyar que la majoria d'atacs coneguts cap a ell provinguin de tècniques sobre reducció de reticles. En referència a la seva resistència vers els ordinadors quàntics, és interessant consultar [FS15], on es s'estudien alguns dels atacs quàntics coneguts cap a NTRU.

Finalment, abans d'entrar en matèria, convé puntualitzar que avui en dia existeixen múltiples implementacions d'aquest criptosistema, les quals difereixen en petits detalls com ara la tria dels paràmetres o l'ús de màscares per millorar-ne la seguretat. En pro de la divulgació, en aquest treball es presenta la versió original publicada a [HPS98], ja que sintetitza perfectament l'essència del seu funcionament.

4.2.1 Notació i paràmetres públics

Un cop presentat el criptosistema NTRU i per tal de descriure'n el seu funcionament, cal establir la notació i els paràmetres públics que es faran servir. Així doncs, es pren un enter $N \geq 1$ i dos enters p i q sobre els que es faran mòduls. En particular, cal que N

sigui primer i que $\gcd(p,q) = \gcd(N,q) = 1$ per motius de seguretat que s'expliquen a la secció 4.5.1 i, de cara a millorar l'eficiència computacional del criptosistema, és interessant prendre q com a potència de 2 per tal d'agilitzar els càlculs mòdul q.

Tot seguit, en base a aquests tres valors, es defineixen els anells polinomials convolucionals

$$R = \frac{\mathbb{Z}[x]}{(x^N-1)},\ R_p = \frac{(\mathbb{Z}/p\mathbb{Z})[x]}{(x^N-1)} \text{ i } R_q = \frac{(\mathbb{Z}/q\mathbb{Z})[x]}{(x^N-1)},$$

d'on surten els diversos polinomis que utilitza NTRU. De fet, per passar de R a R_p o R_q i viceversa es fa ús dels morfismes s i l vistos al capítol 3.

El darrer paràmetre que cal establir és d pertanyent als naturals tal que

$$d \leq \frac{N-1}{2}, \tag{4.1}$$

el qual juga un paper important en la tria dels polinomis corresponents a la clau privada. Per raons de seguretat que es detallen en seccions següents s'acostuma a prendre $\approx N/3$.

A més a més, és recomanable que els paràmetres escollits compleixin la desigualtat

$$q > (6d+1)p, \tag{4.2}$$

de cara a la correcta desencriptació del missatge tal i com es provarà més endavant.

Finalment, cal precisar que la tria dels quatre paràmetres públics (N,p,q,d) li correspon al receptor del missatge que es vol encriptar, el qual acostuma a delegar aquesta tasca a una autoritat de confiança.

4.2.2 Creació de les claus públiques i privades

Publicada la quaterna (N,p,q,d), en Bob, el receptor del missatge, escull com a claus privades dos polinomis $f(x)$ i $g(x)$ pertanyents a R tals que

1. $f(x)$ disposi de $d+1$ coeficients iguals a 1, d coeficients iguals a -1 i la resta de coeficients iguals a 0.

2. $g(x)$ disposi de d coeficients iguals a 1, d coeficients iguals a -1 i la resta de coeficients iguals a 0.

El motiu darrere d'aquestes dues condicions és la millora en eficiència computacional que suposa l'emprar polinomis amb coeficients reduïts. A més, convé notar que aquesta tria de polinomis sempre existeix gràcies a haver escollit d complint (4.1).

A continuació, en Bob calcula els inversos de $f(x)$ mòdul p i q anomenats $f(x)_p^{-1}$ i $f(x)_q^{-1}$ respectivament. Més concretament, en Bob computa $f(x)_p^{-1} \in R_p$ i $f(x)_q^{-1} \in R_q$ tals que $s_p(f(x)) \cdot f(x)_p^{-1} \equiv 1 \pmod{p}$ i $s_q(f(x)) \cdot f(x)_q^{-1} \equiv 1 \pmod{q}$.

En cas que $f(x)$ no disposi d'inversos mòdul p o q, en Bob descartaria $f(x)$ i en triaria un altre d'entre els possibles. De fet, s'imposa que $f(x)$ compleixi la primera condició i no pas la segona degut a que la Proposició 34 prova que els polinomis que compleixen la segona condició no disposen d'inversos a R_q.

Enllaçant-ho amb això, és important precisar que d'ara en endavant s'emprarà l'abús de notació següent per tal de no sobrecarregar el contingut i facilitar-ne la lectura. Quan

es produeixin multiplicacions de polinomis a R_p o a R_q on un dels dos factors provingui de R, $a(x) \in R$ per exemple, enlloc d'escriure $s_p(a(x)) \cdot b(x) \pmod p$ o $s_q(a(x)) \cdot b(x) \pmod q$, s'escriurà directament $a(x) \cdot b(x) \pmod p$ o $a(x) \cdot b(x) \pmod q$.

Un cop fet aquest apunt notacional i seguint amb el procés de generació de claus, en Bob obté $h(x) \in R_q$ a partir del producte

$$h(x) = f(x)_q^{-1} \cdot g(x) \pmod q \tag{4.3}$$

i estableix $h(x)$ com a clau pública per encriptar els missatges que se li enviïn.

En resum, les claus privades són els polinomis $f(x)$ i $g(x)$ i la clau pública el polinomi $h(x)$.

Abans de prosseguir amb els següents passos del criptosistema NTRU, convé analitzar breument la mida de les claus públiques i privades, ja que és un factor essencial de cara al rendiment i eficiència de qualsevol criptosistema. D'una banda, tenint en compte que la clau pública $h(x)$ és correspon a un polinomi de grau N amb coeficients a $\mathbb{Z}_q$, és clar que el número de bits que calen per identificar-la de forma unequívoca és $N \log_2 q$. D'altra banda, les claus privades $f(x)$ i $g(x)$ també són polinomis de grau N però en aquest cas disposen de coeficients a $\{-1, 0, 1\}$, per la qual cosa calen $\lceil 2N \log_2 3 \rceil$ bits per a guardar-les conjuntament.

4.2.3 Encriptació

Posteriorment a la publicació dels paràmetres (N, p, q, d) i de la clau pública $h(x)$, en cas que l'Alice li vulgui enviar un missatge a en Bob, durà a terme el procés següent.

En primer lloc, el missatge es correspon amb un polinomi $m(x) \in R$ amb tots els seus coeficients en l'interval $\left(\frac{-p}{2}, \frac{p}{2}\right]$.

Seguidament, l'Alice escull de forma aleatòria un polinomi $r(x) \in R$ tal que $r(x)$ disposi de d coeficients iguals a 1, d coeficients iguals a -1 i la resta de coeficients iguals a 0. Precisament és la tria d'aquest polinomi el que afegeix la component probabilística al criptosistema NTRU.

Finalment, l'Alice obté $e(x) \in R_q$ corresponent a l'encriptació del missatge $m(x)$ mitjançant el càlcul

$$e(x) \equiv (ph(x) \cdot r(x) + m(x)) \pmod q \tag{4.4}$$

i envia el resultat $e(x)$ a en Bob.

4.2.4 Desencriptació

Quan en Bob rep el missatge encriptat $e(x)$ provinent de l'Alice, per tal de desencriptar-lo ha de realitzar diversos càlculs. Primerament calcula

$$a(x) \equiv f(x) \cdot e(x) \pmod q, \tag{4.5}$$

tot seguit aplica el morfisme l_q a $a(x)$ per tal de convertir-lo en un element de R i, per acabar, calcula $b(x) \in R_p$ a partir del càlcul

$$b(x) \equiv f(x)_p^{-1} \cdot a(x) \pmod p. \tag{4.6}$$

D'aquesta manera, la proposició següent garanteix que, amb una bona tria dels paràmetres, $m(x)$ i $b(x)$ són iguals. Entrant en detalls, si bé és cert que $m(x) \in R$ i $b(x) \in R_p$, com que els coeficients de $m(x)$ pertanyen a $\left(\frac{-p}{2}, \frac{p}{2}\right]$, a l'aplicar el morfisme s_p a $m(x)$ el valor dels coeficients resultants coincideix amb els de $b(x)$ i d'aquí surt la comparativa entre $m(x)$ i $b(x)$.

Proposició 36. *Si els paràmetres públics* (N, p, q, d) *satisfan* (4.2) *aleshores el polinomi* $b(x)$ *calculat a* (4.6) *és igual al text* $m(x)$.

Demostració. En primer lloc, cal estudiar la forma del polinomi $a(x)$ sorgit de (4.5). Desenvolupant s'obté

$$\begin{aligned} a(x) \equiv f(x) \cdot e(x) \pmod q &\equiv f(x) \cdot (ph(x) \cdot r(x) + m(x)) \pmod q \equiv \\ &\equiv pf(x) \cdot f(x)_q^{-1} \cdot g(x) \cdot r(x) + f(x) \cdot m(x) \pmod q \equiv \\ &\equiv pg(x) \cdot r(x) + f(x) \cdot m(x) \pmod q \end{aligned}$$

Seguidament, interessa analitzar quin és el valor més gran que poden prendre els coeficients de $pg(x) \cdot r(x) + f(x) \cdot m(x)$ a R per tal de veure com afecta a $a(x)$ la reducció mòdul q i l'aplicació del morfisme l_q posterior. Així doncs, tenint en compte que tant el polinomi $g(x)$ com el polinomi $r(x)$ disposen únicament de d coeficients iguals a 1, d coeficients iguals a -1 i la resta de coeficients iguals a 0, és clar que el producte $g(x) \cdot r(x)$ aconsegueix el coeficient més gran possible quan s'emparellen tots els 1 i els -1 obtenint el valor de $2d$. Procedint de forma anàloga, tenint en compte que el polinomi $f(x)$ disposa de $d+1$ coeficients iguals a 1, d coeficients iguals a -1 i la resta de coeficients iguals a 0 i que els coeficients del polinomi $m(x)$ pertanyent tots a l'interval $\left(\frac{-p}{2}, \frac{p}{2}\right]$, s'arriba a la conclusió que el coeficient més gran possible del producte $f(x) \cdot m(x)$ és $(2d+1)\frac{1}{2}p$. Per tant, unint ambdós resultats s'arriba a la conclusió que el coeficient més gran possible de $pg(x) \cdot r(x) + f(x) \cdot m(x)$ és

$$p2d + (2d+1)\frac{1}{2}p = (3d + \frac{1}{2})p.$$

Aquesta darrera igualtat, juntament amb la desigualtat (4.2), prova que tot coeficient de $pg(x) \cdot r(x) + f(x) \cdot m(x)$ està en l'interval $\left(\frac{-q}{2}, \frac{q}{2}\right]$. Aleshores és clar que al reduir mòdul q i posteriorment aplicar el morfisme l_q el seu valor es manté. És a dir, la igualtat $a(x) = pg(x) \cdot r(x) + f(x) \cdot m(x)$ es manté a R i no només a R_q.

Finalment, únicament queda fer ús d'aquesta igualtat i desenvolupar el producte $f_p^{-1} \cdot a(x)$ de la manera següent

$$\begin{aligned} b(x) \equiv f_p^{-1} \cdot a(x) = f_p^{-1} \cdot (pg(x) \cdot r(x) + f(x) \cdot m(x)) \equiv \\ \equiv f_p^{-1} \cdot f(x) \cdot m(x) \pmod p \equiv m(x) \pmod p. \end{aligned}$$

Per tant, $m(x)$ i $b(x)$ són iguals mòdul p. A més, com que els coeficients del missatge $m(x)$ originals pertanyen a l'interval $\left(\frac{-p}{2}, \frac{p}{2}\right]$, el valor de $m(x) \pmod p$ coincideix amb el valor de $m(x)$ a R. □

Una conclusió interessant que s'extreu de la demostració anterior és que, tot i que garanteix la correcta desencriptació del missatge quan es compleix la desigualtat (4.2), per valors de q propers a complir la desigualtat la desencriptació també funcionarà amb una alta probabilitat, ja que és molt poc probable que en ambdós productes $g(x) \cdot r(x)$ i $f(x) \cdot m(x)$ s'emparellin tots els coeficients positius i negatius i, per tant, l'expressió de $a(x)$ no sigui igual a R i R_q. No obstant això, si es pren un valor de q significativament menor a $(6d+1)p$ la probabilitat d'error al desencriptar augmenta considerablement. Per motius de seguretat, es recomana disposar d'una probabilitat d'error com a màxim de 2^{-80} degut a que aquesta tipologia d'errors poden revelar informació sobre la clau privada.

A mode de resum, la taula següent ofereix un esquema del funcionament del criptosistema NTRU.

Paràmetres públics	
Enters (N, p, q, d)	
Alice	**Bob**
Creació de les claus	
	Claus privades: $f(x), g(x) \in R$
	Clau pública: $h(x) = f(x)_q^{-1} \cdot g(x) \pmod q$
	Publica la clau pública $h(x)$.
Encriptació	
Missatge: $m(x) \in R$	
Clau efímera: $r(x) \in R$	
Encriptació: $e(x) = (ph(x) \cdot r(x) + m(x)) \pmod q$	
Envia l'encriptació $e(x)$ a en Bob.	
Desencriptació	
	Modificació de l'encriptació: $a(x) = f(x) \cdot e(x) \pmod q$
	Recuperació del missatge: $m(x) = f(x)_p^{-1} \cdot a(x) \pmod p$

Taula 4.1: Taula que mostra el funcionament del criptosistema NTRU.

4.3 Problemes matemàtics associats a NTRU

Com la gran majoria de criptosistemes de clau pública, NTRU basa la seva seguretat en la gran dificultat que suposa per agents externs obtenir la clau privada o el missatge desencriptat. Aquests dos problemes s'anomenen problema de la recuperació de la clau o KRP (de l'anglès *Key Recovery Problem*) i problema de la recuperació del missatge o MRP (de l'anglès *Message Recovery Problem*) respectivament.

4.3.1 Problema de la recuperació de la clau

En línies generals, tal i com el seu nom indica, el problema de la recuperació de la clau consisteix en obtenir les claus privades del criptosistema a partir de tota la informació publica de la que es disposa. Entrant en detalls, donada una clau pública $h(x)$ d'una instància del criptosistema NTRU, l'atacant està interessat en emprar-la per tal de recuperar les claus privades $f(x)$ i $g(x)$ i poder així desencriptar tots els missatges que s'enviïn.

D'aquesta manera, partint de la generació de la clau pública $h(x)$ a l'equació (4.3) s'obté l'equivalència

$$f(x) \cdot h(x) \equiv g(x) \pmod{q}, \tag{4.7}$$

que permet establir una relació directa entre ambdues tipologies de claus. Addicionalment, més enllà d'aquesta relació, l'atacant també és coneixedor del fet que tant el polinomi $f(x)$ com $g(x)$ únicament disposen de coeficients per valor de $-1,\ 0$ i 1. Així doncs, agrupant ambdues casuístiques es pot formular el KRP en termes matemàtics de la manera següent.

Definició 37 (Problema de recuperació de clau)**.** *Donat un polinomi $h(x)$ es volen trobar polinomis $f(x)$ i $g(x)$ amb coeficients $-1,\ 0$ i 1 tals que $f(x) \cdot h(x) \equiv g(x) \pmod{q}$.*

En referència al KRP, és interessant puntualitzar que no disposa de solució única, ja que en cas que una parella de polinomis $(f(x), g(x))$ fos solució, aleshores totes les parelles $(x^i f(x), x^i g(x))$ amb $i = 0, .., N-1$ també serien solució. Aquest tipus de polinomis s'anomenen rotacions.

4.3.2 Problema de la recuperació del missatge

La idea general darrere del problema de la recuperació del missatge consisteix en obtenir el missatge original enviat a través del criptosistema sense disposar de la clau privada per desencriptar-lo. En conseqüència, de la mateixa manera que succeeix amb el KRP, l'atacant mira d'aprofitar tota informació pública per assolir el seu objectiu.

Així doncs, en base al càlcul (4.4) s'aconsegueix la congruència

$$e(x) - (ph(x) \cdot r(x)) \equiv m(x) \pmod{q},$$

la qual relaciona el missatge que es vol aconseguir $m(x)$ amb un seguit de termes coneguts a excepció del polinomi aleatori $r(x)$. A més a més, una altra informació pública és que el polinomi original $m(x)$ disposa de tots els seus coeficients a l'interval $\left(\frac{-p}{2}, \frac{p}{2}\right]$ i que el polinomi $r(x)$ únicament disposa de coeficients per valor de $-1,\ 0$ i 1. D'aquesta manera, procedint de forma anàloga al KRP, es pot formular el MRP en termes matemàtics de la manera següent.

Definició 38 (Problema de recuperació de missatge)**.** *Donat un enter p i uns polinomis $e(x)$ i $h(x)$ es vol trobar un polinomi $r(x)$ amb coeficients $-1,\ 0$ i 1 i un polinomi $m(x)$ amb coeficients pertanyents a $\left(\frac{-p}{2}, \frac{p}{2}\right]$ tals que $e(x) - (ph(x) \cdot r(x)) \equiv m(x) \pmod{p}$.*

Finalment, és important remarcar que la seguretat del criptosistema NTRU es basa, en gran part, en l'enorme dificultat matemàtica que suposa resoldre aquests dos problemes per uns paràmetres donats. Així doncs, tot i que a simple vista pugui semblar complicat estimar-ne el grau de dificultat únicament amb el seu enunciat, el cert és que en l'apartat que segueix es prova l'equivalència entre el KRP i el MRP amb el SVP i el CVP respectivament, els quals ja s'ha vist que són problemes d'alta complexitat.

4.4 NTRU com a criptosistema de reticles

Al llarg dels apartats anteriors s'ha presentat el criptosistema NTRU juntament amb el procés d'encriptació i desencriptació de missatges i els seus problemes matemàtics associats, però en tot aquest procés no s'ha emprat cap concepte provinent de la teoria de reticles. Així doncs, en aquesta secció es pretèn connectar ambdós camps i analitzar quins avantatges teòrics i pràctics comporta aquesta nova perspectiva del criptosistema NTRU, fent especial èmfasi en la formulació del KRP i del MRP com a instàncies del SVP i del CVP.

En primer lloc, cal definir el reticle sobre el qual es vol treballar. Per fer-ho, és clar que convé generar-lo a partir d'alguns dels elements propis del criptosistema NTRU per tal d'establir-hi relació.

Definició 39. *Donada una instància del criptosistema NTRU amb els paràmetres (N, p, q, d) i la clau pública $h(x) = h_{N-1}x^{N-1} + \cdots + h_1 x + h_0$, s'anomena reticle NTRU associat a $h(x)$ al reticle enter L_h generat per les columnes de la matriu de dimensió $2N \times 2N$*

$$M_h = \left(\begin{array}{cccc|cccc} 1 & 0 & \dots & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & 0 & 0 & \dots & 0 \\ \hline h_0 & h_{N-1} & \dots & h_1 & q & 0 & \dots & 0 \\ h_1 & h_0 & \dots & h_2 & 0 & q & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ h_{N-1} & h_{N-2} & \dots & h_0 & 0 & 0 & \dots & q \end{array}\right). \tag{4.8}$$

De fet, la matriu M_h està formada per quatre submatrius NxN, on el bloc superior esquerra es correspon amb Id, el bloc superior dret es correspon amb la matriu nul·la, el bloc inferior esquerre són les permutacions cícliques dels coeficients de $h(x)$ i el bloc inferior dret és qId.

Un aspecte interessant sobre la matriu M_h és que malgrat ser una matriu de dimensió $2N$, computacionalment el seu emmagatzematge és molt poc costós. Això es deu a que, com que el bloc inferior esquerre són les permutacions cícliques dels coeficients de $h(x)$, només cal guardar els N coeficients de $h(x)$ una vegada. A més, les 3 submatrius restants no requereixen emmagatzematge per a poder ser reproduïdes llevat del valor q. En resum, la matriu M_h es pot representar guardant únicament $N + 1$ valors diferents.

D'altra banda, tal i com s'ha vist en el capítol referent a la teoria de reticles, un dels conceptes més útils a l'hora de treballar amb aquests objectes és el del determinant del reticle. Per tant, convé calcular el valor de $\det(L_h)$.

Proposició 40. *Donada una instància del criptosistema NTRU amb els paràmetres (N, p, q, d), la clau pública $h(x)$ i matriu M_h, el reticle NTRU associat L_h té volum q^N.*

Demostració. Gràcies a la Proposició 8 i a que la matriu M_h és triangular superior és clar que $\det(L_h) = q^N$. □

Acte seguit, un cop definit el reticle NTRU associat L_h és interessant veure quin paper juga en el KRP i el MRP i com ens permet relacionar-los amb el SVP i el CVP respectivament.

4.4.1 KRP com a instància del SVP

A grans trets, en aquest subapartat es pretén veure que el vector (f,g) corresponent a les dues claus privades del criptosistema NTRU pertany a L_h i, a més a més, que es tracta d'un dels vectors més curts de L_h. D'aquesta manera, la connexió entre el KRP i el SVP es fa evident.

La proposició següent mostra que el vector (f,g) pertany a L_h.

Proposició 41. *Donada una instància del criptosistema NTRU amb els paràmetres (N,p,q,d), la clau pública $h(x) = h_{N-1}x^{N-1} + \cdots + h_1x + h_0$, matriu M_h i reticle NTRU associat L_h, assumint* (4.7) *i prenent $u(x)$ pertanyent a R com el polinomi que satisfà la igualtat $f(x)\cdot h(x) = g(x) + qu(x)$, aleshores*

$$M_h \begin{pmatrix} f \\ -u \end{pmatrix} = \begin{pmatrix} f \\ g \end{pmatrix}. \tag{4.9}$$

Per tant, el vector (f,g) pertany a L_h.

Demostració. D'una banda, és clar que les primeres N coordenades del producte (4.9) són iguals a $f(x)$, degut a que es duu a terme el producte dels blocs Id i la matriu nul·la amb les coordenades de $f(x)$ i $-u(x)$ respectivament.

D'altra banda, per analitzar les N darreres coordenades del producte (4.9), convé estudiar-ho fila per fila. Així doncs, al multiplicar la fila $N+1+i$ amb $i = 0,...,N-1$ de la matriu M_h amb el vector $(f,-u)$ s'obté l'expressió

$$h_if_0 + h_{i-1}f_1 + \cdots + h_{i+1}f_{N-1} - qu_i,$$

que clarament coincideix amb el coeficient i-èssim del polinomi $g(x) = f(x)\cdot h(x) - qu(x)$. Per tant, és clar que les darreres N coordenades del producte (4.9) són iguals a $g(x)$. □

Un cop vist que el vector (f,g) pertany a L_h, resta veure que té una longitud considerablement curta. Precisament això és el que prova la proposició que segueix.

Proposició 42. *Donada una instància del criptosistema NTRU amb els paràmetres (N,p,q,d), la clau pública $h(x)$, matriu M_h, reticle NTRU associat L_h i suposant que $d \approx N/3$ i $q \approx 6d \approx 2N$. Aleshores es compleixen els següents enunciats:*

1. $\|(f,g)\| \approx \sqrt{4d} \approx \sqrt{\frac{4N}{3}} \approx 1,155\sqrt{N}$.
2. *L'heurística Gaussiana prediu que el vector més curt diferent de zero de L_h té llargada*

$$\sigma(L_h) \approx \sqrt{\frac{Nq}{\pi e}} \approx 0,484N.$$

Així doncs, quan N és prou gran hi ha una alta probabilitat que el vector més curt diferent de zero de L_h sigui (f, g) i les seves rotacions. Addicionalment,

$$\frac{\|(f,g)\|}{\sigma(L_h)} \approx \frac{2,386}{\sqrt{N}},$$

és a dir, el vector (f, g) és un factor

$$\mathcal{O}(\frac{1}{\sqrt{N}})$$

més petit que el predit per l'heurística Gaussiana.

Demostració. Es demostraran els dos enunciats de forma consecutiva.

1. De la definició de les claus privades $f(x)$ i $g(x)$ se n'extreu que ambdós poliomis tenen aproximadament d coeficients iguals a 1 i d coeficients iguals a -1. D'aquí i emprant les estimacions considerades com a hipòtesi s'aconsegueixen les aproximacions buscades.

2. Únicament cal tenir present que $\dim(L_h) = 2N$, $\det(L_h) = q^N$ i aplicar la fórmula de l'heurística Gaussiana (2.6).

□

4.4.2 MRP com a instància del CVP

De forma similar al que succeeix amb el KRP, en aquest subapartat es vol mostrar que resoldre el CVP pel vector $(0, e)$ a L_h permet resoldre el MRP automàticament. En aquest cas, per tal d'aconseguir-ho cal provar que el vector $(pr, e - m)$ pertany a L_h i veure que es tracta, molt probablement, del vector de L_h més proper a $(0, e)$. D'aquesta manera, resolent el CVP pel vector $(0, e)$ s'obté $(pr, e - m)$ i únicament cal calcular $(0, e) - (pr, e - m) = (-pr, m)$ per aconseguir el missatge original.

La proposició següent mostra que el vector $(pr, e - m)$ pertany a L_h.

Proposició 43. *Donada una instància del criptosistema NTRU amb els paràmetres (N, p, q, d), la clau pública $h(x) = h_{N-1}x^{N-1} + \cdots + h_1x + h_0$, matriu M_h i reticle NTRU associat L_h, assumint que $e(x) \equiv pr(x) \cdot h(x) + m(x) \pmod q$ i prenent $u(x)$ pertanyent a R com el polinomi que satisfà la igualtat $e(x) = pr(x) \cdot h(x) + m(x) + qu(x)$, aleshores*

$$M_h \begin{pmatrix} pr \\ u \end{pmatrix} = \begin{pmatrix} pr \\ e - m \end{pmatrix}. \tag{4.10}$$

Per tant, el vector $(pr, e - m)$ pertany a L_h.

Demostració. Aquesta demostració segueix un curs pràcticament idèntic a la de la Proposició 41. D'una banda, és clar que les primeres N coordenades del producte (4.10) són iguals a $pr(x)$, degut a que es duu a terme el producte dels blocs Id i la matriu nul·la amb les coordenades de $pr(x)$ i $u(x)$ respectivament.

D'altra banda, per analitzar les N darreres coordenades del producte (4.10), convé estudiar-ho fila per fila. Així doncs, al multiplicar la fila $N+1+i$ amb $i = 0, ..., N-1$ de la matriu M_h amb el vector $(f, -u)$ s'obté l'expressió

$$h_i pr_0 + h_{i-1} pr_1 + \cdots + h_{i+1} pr_{N-1} - qu_i,$$

que clarament coincideix amb el coeficient i-èssim del polinomi $e(x) - m(x) = pr(x) \cdot h(x) - qu(x)$. Per tant, és clar que les darreres N coordenades del producte (4.10) són iguals a $e(x) - m(x)$. □

Un cop vist que el vector $(pr, e-m)$ pertany a L_h, resta provar que és molt probable que sigui el vector de L_h més proper a $(0, e)$. Per veure-ho, es fa ús de la proposició que es mostra a continuació.

Proposició 44. *Donada una instància del criptosistema NTRU amb els paràmetres (N, p, q, d), la clau pública $h(x)$, matriu M_h, reticle NTRU associat L_h i suposant que $p = 2$, $d \approx \frac{N}{3}$ i $q \approx 6d \approx 2N$. Aleshores es compleixen els següents enunciats:*

1. $\|(pr, e-m) - (0, e)\| \approx 2,646\sqrt{N}$.
2. *L'heurística Gaussiana prediu que el vector més proper a $(0, e)$ de L_h està a una distància*

$$\sigma(L_h) \approx \sqrt{\frac{Nq}{\pi e}} \approx 0,484N.$$

Així doncs, quan N és prou gran hi ha una alta probabilitat que el vector més proper a $(0, e)$ de L_h sigui $(pr, e-m)$. Addicionalment,

$$\frac{\|(pr, e-m) - (0, e)\|}{\sigma(L_h)} \approx \frac{5,467}{\sqrt{N}},$$

és a dir, la distància entre els vectors $(pr, e-m)$ i $(0, e)$ és un factor $\mathcal{O}(\frac{1}{\sqrt{N}})$ més petita que la predida per l'heurística Gaussiana.

Demostració. Es demostraran els dos enunciats de forma consecutiva.

1. De la definició del polinomi aleatori $r(x)$ se n'extreu que té d coeficients iguals a 1 i d coeficients iguals a -1. A més, com que $p = 2$, els coeficients de $m(x)$ valen 0 o 1. Per tant, emprant les estimacions considerades com a hipòtesi i desenvolupant s'obté

$$\|(pr, e-m) - (0, e)\| = \|(pr, -m)\| \approx \sqrt{p^2 2d + N} \approx$$
$$\approx \sqrt{\frac{p^2 2N}{3} + N} = \sqrt{(\frac{p^2 2}{3} + 1)N} \approx 2,646\sqrt{N}.$$

2. Únicament cal tenir present que $\dim(L_h) = 2N$, $\det(L_h) = q^N$ i aplicar la fórmula de l'heurística Gaussiana (2.6).

□

4.5 Seguretat del criptosistema NTRU

Al final de la secció anterior ha quedat palesa la relació entre la seguretat del criptosistema NTRU i la resolució de problemes associats als reticles. Més concretament, el fet d'expressar el KRP i el MRP com a instàncies del SVP i del CVP respectivament implica que la dificultat de trencar el criptosistema NTRU és, com a molt, la dificultat de resoldre el SVP o el CVP. De fet, si bé aquesta sembla una via interessant a explorar de cara a comprovar la integritat del criptosistema NTRU, el cert és que existeixen molts altres atacs que parteixen de premises completament diferents. Així doncs, a continuació s'estudien un seguit d'atacs i la seva eficàcia vers el criptosistema NTRU amb l'objectiu d'obtenir una visió global del seu grau de seguretat.

4.5.1 Atacs a la vulnerabilitat dels paràmetres

Des d'un principi, a l'hora d'establir els paràmetres (N, p, q, d) del criptosistema NTRU s'ha imposat que N sigui primer i que p i q siguin coprimers entre ells per motius de seguretat. Més concretament, si alguna d'aquestes dues condicions no es compleix es poden dur a terme els atacs que es llisten a continuació.

D'una banda, si $\gcd(p, q) = c$ amb $c \neq 1$, tenint present que

$$e(x) \equiv (ph(x) \cdot r(x) + m(x)) \pmod{q},$$

al prendre mòdul c s'obté

$$e(x) \equiv m(x) \pmod{c}.$$

Per tant, com major sigui el divisor c més informació obté l'atacant referent al missatge original. En particular, si $p \mid q$ és clar que aquest atac permet recuperar el missatge integrament.

D'altra banda, en cas que N sigui un nombre compost es pot reduir el KRP a un reticle de dimensió inferior a $2N$, fet que en facilita considerablement la resolució. Per motius de concreció, a continuació únicament es detalla aquest procés quan N és parell. Per tal de veure'n la generalització per N compost es recomana consultar [Gen01].

Prenent N parell es pot factoritzar el polinomi $x^N - 1$ com

$$x^N - 1 = (x^{N/2} + 1)(x^{N/2} - 1),$$

on ambdós polinomis $x^{N/2} + 1$ i $x^{N/2} - 1$ són coprimers entre ells per no tenir arrels comunes. Acte seguit, aplicant els morfismes quocient

$$\pi_1 : R \to \frac{\mathbb{Z}[x]}{(x^{N/2} + 1)} \text{ i } \pi_2 : R \to \frac{\mathbb{Z}[x]}{(x^{N/2} - 1)}$$

a la clau pública $h(x)$ s'obté $\pi_1(h)$ i $\pi_2(h)$. El pas que segueix consisteix a definir els reticles NTRU de dimensió N $L_{\pi_1(h)}$ i $L_{\pi_2(h)}$ associats a $\pi_1(h)$ i $\pi_2(h)$ respectivament i resoldre'n el KRP per obtenir les claus privades $f_1(x)$, $g_1(x)$, $f_2(x)$ i $g_2(x)$. D'aquesta manera, es redueix el problema inicial de resoldre el KRP a reticle de dimensió $2N$ a resoldre dues vegades el KRP a reticles de dimensió N. Finalment, per tal d'obtenir les claus privades $f(x)$ i $g(x)$ originals, únicament cal aplicar el Teorema Xinès del Residu

[Chi09] als polinomis $f_1(x)$, $g_1(x)$, $f_2(x)$ i $g_2(x)$. Evidentment, si el valor $N/2$ continua sent parell, es pot tornar a aplicar aquest procés de forma recursiva reduïnt encara més la dimensió del reticle.

4.5.2 Atacs per força bruta

Tal i com el seu nom indica, aquests atacs no es basen en explotar cap vulnerabilitat tècnica del criptosistema ni en cap ocurrència de l'atacant, sinó que senzillament consisteixen en tirar pel dret provant totes les possibles combinacions. Així doncs, no és d'estranyar que aquesta tipologia d'atacs requereixin d'una alta exigència computacional i gairebé mai donin resultats. Malgrat això, és necessari tenir-los presents i assegurar-se que el criptosistema en qüestió no hi sigui vulnerable, ja que el seu ús és força comú.

En el cas que ens ocupa, l'atac per força bruta consisteix en trobar les claus privades $f(x)$ i $g(x)$. Per fer-ho, l'atacant té present l'equivalència (4.7), que el polinomi $f(x)$ té $d+1$ coeficients que valen 1, d coeficients que valen -1 i la resta 0, i que el polinomi $g(x)$ té d coeficients que valen 1, d coeficients que valen -1 i la resta 0. D'aquesta manera, l'atacant considera tots els polinomis $a(x)$ que compleixen les mateixes condicions que la clau privada $f(x)$ i en va calculant

$$a(x) \cdot h(x) \pmod{q} \tag{4.11}$$

fins a obtenir un polinomi resultat $b(x)$ que complexi les condicions de la clau privada $g(x)$. Tal i com es veurà a continuació, és molt probable que, en cas que compleixin les condicions de les claus privades, els polinomis $a(x)$ i $b(x)$ es corresponguin amb les claus privades $f(x)$ i $g(x)$.

Abans però, és interessant estimar la quantitat ingent d'intents que es requereixen per tal de trobar un polinomi $a(x)$ que serveixi a l'atacant. Tenint en compte que les N rotacions del polinomi $f(x)$ també són claus privades vàlides, el nombre esperat d'intents que s'han de dur a terme per encertar amb l'elecció de la clau $a(x)$ és

$$\frac{\binom{N}{d+1}\binom{N-d-1}{d}}{N} = \frac{(N-1)!}{(d+1)!\,d!\,(N-2d-1)!},$$

que pren el seu valor màxim quan $d = N/3$. Evidentment, a mesura que N augmenta, el nombre de casos creix considerablement i les opcions de trencar el criptosistema NTRU mitjançant aquest atac disminueixen fins a ser irrisòries. A tall d'exemple, prenent $N = 79$ i $d = 26$ s'haurien de realitzar al voltant de 2^{112} intents.

Finalment, per tal de justificar que els polinomis $a(x)$ i $b(x)$ segurament es corresponen amb les claus privades $f(x)$ i $g(x)$, n'hi ha prou amb estimar el nombre de polinomis $b(x)$ complint les condicions de $g(x)$ que s'obtenen a partir de l'operació $a(x) \cdot h(x) \pmod{q}$, on $a(x)$ compleix les condicions de $f(x)$, i veure que es tracta d'un valor molt proper a 0. D'aquesta manera, tenint en compte que el vist a l'apartat 4.4.1 assegura l'obtenció de les N claus vàlides mitjançant el producte (4.11), com que la probabilitat d'obtenir un polinomi complint les condicions de $g(x)$ és ínfima, és raonable afirmar que, en cas de trobar-ne un, es correspondrà amb una de les N claus vàlides.

Així doncs, considerant la proporció de polinomis a R_q que compleixen les condicions de $g(x)$ i tenint present la quantitat de polinomis que compleixen les condicions de $f(x)$ s'obté

$$\frac{\binom{N}{d}\binom{N-d}{d}}{q^N}\binom{N}{d+1}\binom{N-d-1}{d} = \frac{N!^2}{q^N(d+1)!\,d!^3\,(N-2d-1)!\,(N-2d)!},$$

que clarament es tracta d'un nombre molt pròxim a 0. Prenent altre cop $N = 79$ i $d = 26$ s'obté un resultat de l'ordre de 2^{-474}.

4.5.3 Atacs per transmissió múltiple

Els atacs per transmissió múltiple consisteixen en atacar els criptosistemes a través de l'enviament de diversos missatges idèntics o amb algunes propietats similars entre ells. Així doncs, es recomana no utilitzar el mateix polinomi aleatori per encriptar dos missatges consecutius ni tampoc enviar el mateix missatge emprant dos polinomis aleatoris diferents degut a que els dos atacs que es detallen a continuació aprofiten les vulnerabilitats inherents a aquests fets.

D'una banda, en cas que l'Alice enviï k vegades el mateix missatge $m(x)$ encriptant-lo usant polinomis aleatòris $r_i(x)$ distints amb $i = 1, ..., k$, aleshores l'atacant pot emprar l'equivalència (4.4) per obtenir informació del polinomi aleatori $r_1(x)$. Entrant en detalls, tenint present la construció de e_i, prenent

$$c_i(x) \equiv (e_i(x) - e_1(x)) \cdot h^{-1}(x) \pmod{q}, \tag{4.12}$$

i desenvolupant-ho s'obté

$$\begin{aligned} c_i(x) &\equiv (e_i(x) - e_1(x)) \cdot h^{-1}(x) \equiv e_i(x) \cdot h^{-1}(x) - e_1(x) \cdot h^{-1}(x) \equiv \\ &\equiv r_i(x) + m(x) \cdot h^{-1}(x) - r_1(x) + m(x) \cdot h^{-1(x)} \equiv r_i(x) - r_1(x) \pmod{q} \end{aligned}$$

per tot $i = 1, ..., k$.

Així doncs, considerant la desigualtat (4.2) i que els coeficients de $c_i(x)$ pertanyen a $\{-2, -1, 0, 1, 2\}$, és clar que $c_i(x)$ és exactament $r_i(x) - r_1(x)$ més enllà de prendre mòdul q. D'aquesta manera, quan un coeficient de $c_i(x)$ valgui 2 (o -2), voldrà dir que ambdós coeficients de $r_1(x)$ i $r_i(x)$ valen 1 (o -1) obtenint així informació sobre els coeficients de $r_1(x)$. A més a més, convé notar que la probabilitat que ambdós coeficients valguin ± 1 és de $2 \cdot \frac{1}{3} \cdot \frac{1}{3} = \frac{2}{9}$, és a dir, que calculant un sol $c_i(x)$ s'estima que l'atacant podrà recuperar al voltant de $\frac{2}{9}$ dels coeficients de $r_1(x)$.

Per acabar, cal precisar que, tal i com s'extreu de (4.12), aquest atac requereix de l'existència d'un invers mòdul q de la clau pública $h(x)$ per poder ser executat i això és quelcom que no està garantit.

D'altra banda, suposant que l'Alice enviï diversos missatges m_i amb $i = 1, ..., k$ emprant el mateix polinomi aleatori $r(x)$, l'atacant pot obtenir informació del missatge $m_1(x)$ seguint un procediment pràcticament idèntic a l'acabat d'explicar. En aquest cas, desen-

volupant la resta $e_1(x) - e_i(x) \pmod q$ s'obté

$$c_i \equiv e_1(x) - e_i(x) \equiv r(x) \cdot h(x) + m_1(x) - (r(x) \cdot h(x) + m_i(x)) \equiv \\ \equiv m_1(x) - m_i(x) \pmod q.$$

Com que tots els coeficients de c_i pertanyen a $\{-p+1, -p+2, ..., p-1\}$ i altre cop gràcies a la desigualtat (4.2), s'observa que $c_i(x)$ és exactament $m_1(x) - m_i(x)$ més enllà de prendre mòdul q. Finalment, en funció del valor fixat de p l'atacant pot deduir més o menys coeficients del missatge $m_1(x)$ fent servir el mateix raonament que abans. De fet, quan $p = 3$ els coeficients de c_i pertanyen a $\{-2, -1, 0, 1, 2\}$ i, per tant, es dona exactament la mateixa situació relatada amb anterioritat.

4.5.4 Atacs per trobada a mig camí

Els atacs per trobada a mig camí o MITM (de l'anglès *Meet In The Middle*) es fonamenten en l'ajust del compromís entre memòria i temps necessaris per atacar un criptosistema. Més concretament, la idea darrera aquesta espècie d'atacs és la de millorar els atacs per força bruta mitjançant heurístiques que en permetin reduir el nombre de comprovacions a realitzar a canvi d'augmentar-ne la memòria necessària.

En referència al criptosistema NTRU, l'atac MITM existent va ser proposat originàriament per Odlyzko i s'explica amb tot detall a [HGSW03]. A grans trets, l'objectiu d'aquest atac és el d'obtenir les claus privades $f(x)$ i $g(x)$ aprofitant l'equivalència (4.7) i un algorisme de col·lisió.

Partint del conjunt de polinomis que compleixen les condicions de la clau privada $f(x)$, l'atacant separa cadascun d'aquests polinomis $a(x)$ en dos polinomis $a_1(x)$ i $a_2(x)$ on $a_1(x)$ es correspon als $\lceil N/2 \rceil$ coeficients més grans de $a(x)$ i $a_2(x)$ es correspon als $\lfloor N/2 \rfloor$ coeficients més petits de $a(x)$. Tot seguit, es comprova si tant $a_1(x)$ com $a_2(x)$ disposen de la meitat dels coeficients que tenen valor igual a ± 1 i, en cas que no sigui així, el polinomi $a(x)$ queda descartat. No obstant això, és clar que alguna rotació del polinomi $a(x)$ sí que compleix aquesta propietat, de tal manera que tot polinomi $a(x)$ és tingut en compte indirectament.

Seguidament, es fixa un enter k tal que $2^k > \binom{N/2}{d/2}\binom{N-d/2}{d/2}$ i es classifiquen els polinomis $a_1(x)$ en 2^k calaixos etiquetats amb els nombres des de 0 a $2^k - 1$ escrits en binari seguint els passos següents:

1. Es prenen els primers k coeficients de $a_1(x) \cdot h(x) \pmod q$.
2. Es tradueixen aquests k coeficients a binari.
3. S'emmagatzema el polinomi $a_1(x)$ al calaix etiquetat amb el nombre resultant de concatenar el bit més significatiu de cadascun dels k números binaris obtinguts al pas 2.

Després de col·locar els polinomis $a_1(x)$ en els diferents calaixos, es classifiquen els polinomis $a_2(x)$ en base a la política que segueix.

1. Es prenen els primers k coeficients de $-a_2(x) \cdot h(x) \pmod{q}$.
2. Es tradueixen aquests k coeficients a binari.
3. Es suma 1 a tots els valors binaris calculats al pas 2.
4. Es tradueixen aquests k coeficients a binari.
5. Es resta 1 a tots els valors binaris calculats al pas 2.
6. Es tradueixen aquests k coeficients a binari.
7. S'emmagatzema el polinomi $a_2(x)$ al calaix etiquetat amb el nombre resultant de concatenar el bit més significatiu de cadascun dels k números binaris obtinguts al pas 2.
8. També s'emmagatzema el polinomi $a_2(x)$ als calaixos etiquetats amb els nombres corresponents a totes les combinacions resultants de substituir qualssevol dels bits concatenats al pas anterior pels seus homòlegs dels k números binaris obtinguts als passos 4 i 6.

Com a darrer pas, únicament es calcula $a(x) \cdot h(x) \pmod{q}$ amb aquells $a(x) = a_1(x) + a_2(x)$ tals que $a_1(x)$ i $a_2(x)$ comparteixin calaix. En cas que el resultat esdevingui un polinomi $b(x)$ que compleixi les condicions de la clau privada $g(x)$, pel mateix argument vist a la secció referent als atacs per força bruta, l'atacant pot usar la parella $(a(x), b(x))$ com a claus privades.

A partir de resultats experimentals, ha quedat palès que aquest atac redueix el nombre de comprovacions a realitzar en un factor d'arrel quadrada respecte l'atac de força bruta. En altres paraules, es requereixen al voltant de

$$\sqrt{\frac{(N-1)!}{(d+1)!\ d!\ (N-2d-1)!}}$$

intents per tal d'obtenir una parella de polinomis $(a(x), b(x))$ que serveixi com a claus privades. En aquest sentit, tot i que és clar que aquest atac redueix considerablement la quantitat d'intents necessaris, el cert és que el nombre de comprovacions continua sent excessivament elevat per a poder ser portat a la pràctica.

Per acabar, un cop presentat l'atac, és interessant donar una breu intuició sobre perquè funciona, fent especial èmfasi en l'algorisme de col·lisió. Separant $f(x)$ en $f_1(x)$ i $f_2(x)$ complint les condicions esmentades i partint de l'equivalència (4.7) s'obté

$$f_1(x) \cdot h(x) \equiv g(x) - f_2(x) \cdot h(x) \pmod{q}.$$

I tenint present que tots els coeficients de $g(x)$ pertanyen a $\{-1, 0, 1\}$, es pot escriure

$$f_1(x) \cdot h(x) \equiv \{-1, 0, 1\} - f_2(x) \cdot h(x) \pmod{q}.$$

Així doncs, els passos 3 i 5 de l'algorisme de col·lisió pretenen considerar els casos en els que apareixen els 1 en la darrera equivalència.

4.5.5 Atacs basats en reticles

En l'actualitat, els atacs basats en reticles són l'arma més efectiva que es coneix contra el criptosistema NTRU. De fet, precisament els dos més populars es corresponen amb els presentats, de forma indirecta, als apartats 4.4.1 i 4.4.2, on s'ha detallat la reformulació del KRP i del MRP en termes del SVP i del CVP respectivament. Entrant en detalls, mitjançant l'algorisme LLL (sovint la seva versió millorada BKZ-LLL) s'obté una base reduïda que, tal i com s'ha vist en apartats anteriors, permet resoldre ambdós problemes de reticles i, per extensió, el KRP i el MRP.

Així doncs, és necessari parar-se a analitzar com de complicat és, computacionalment parlant, resoldre el SVP i el CVP al reticle L_h. Tenint en compte la Proposició 42, es pot considerar que si l'atacant és capaç de resoldre l'apprSVP per un factor menor que $\sqrt{N}$, aleshores el resultat obtingut es correspon molt probablement amb les claus privades del criptosistema NTRU. Afortunadament o malaurada, en seccions anteriors s'ha vist com l'algorisme LLL permet resoldre en temps polinomial l'apprSVP amb un factor de $\left(\frac{1}{\delta-\frac{1}{4}}\right)^{(N-1)/2}$, on $\delta \in \left(\frac{1}{4}, 1\right)$, de tal manera que a mesura que augmenta N el resultat perd prou precisió com per ser vàlid. Ara bé, cal matisar que la variant BKZ-LLL millora aquesta aproximació fins a un factor de $\beta^{N/\beta}$, β es correspon a un enter major que 2, però requereix d'un temps d'execució exponencial en β. Per tant, és clar que a dia d'avui els atacs al KRP i MRP encara no suposen un impediment insalvable pel criptosistema NTRU degut a la impossibilitat de resoldre prou satisfactòriament tant el SVP com el CVP. A [SW03] es descriuen minuciosament els procediments experimentals que permeten arribar a aquestes conclusions.

Enllaçant-ho amb això, cal precisar que, més enllà del KRP i del MRP, existeixen altres atacs al criptosistema NTRU que es basen en la teoria de reticles. Un d'aquests atacs és el que es coneix com a atac de reticles que operen amb el zero o ZRL (de l'anglès *Zero-Run Lattice*). Aquest atac va ser proposat per May [May99] i, conceptualment, pretén recuperar les claus privades $f(x)$ i $g(x)$ aprofitant el fet que disposen d'un nombre considerable de zeros ($N-2d-1$ i $N-2d$ respectivament). Més concretament, l'atacant vol modificar el reticle NTRU L_h de tal manera que un dels vectors més curts del reticle també tingui moltes coordenades iguals a 0 amb l'esperança que aquest coincideixi amb les dues claus privades del criptosistema.

Per tal d'aconseguir-ho, en primer lloc l'atacant escull un enter positiu z corresponent al nombre de coordenades iguals a 0 que vol imposar a un dels vector més curts del reticle. Evidentment, $z \leq N-4d-1$, ja que en cas contrari aquest vector tindria masses coordenades nul·les i no seria d'utilitat. Acte seguit, modifica la matriu M_h corresponent al reticle NTRU L_h multiplicant z columnes diferents de M_h per un nombre molt gran α a escollir per l'atacant. D'aquesta manera, el nou reticle L'_h disposa d'una base amb z vectors d'una longitud inusualment gran i, per tant, els vectors més curts de L'_h han de tenir les z coordenades corresponents a aquests vectors de la base iguals a 0. Finalment, únicament resta resoldre el SVP mitjançant l'algorisme LLL i comprovar si el vector més curt obtingut serveix com a claus privades.

En referència al rendiment d'aquest atac, convé notar que, de la mateixa manera que passava amb el KRP i el MRP, ve marcat per l'ús de l'algorisme LLL. En conseqüència,

l'atac de reticles que operen amb el zero no suposa una millora vers els que ataquen directament el KRP o el MRP, sinó tan sols una alternativa. Ara bé, a [Sil99b] es presenta una modificació d'aquest atac que permet reduir la dimensió del reticle L'_h millorant-ne així les prestacions.

4.5.6 Atacs quàntics

Tal i com s'ha comentat a la introducció, el gran atractiu del criptosistema NTRU és, sense cap mena de dubte, la seva aparent resistència quàntica. Així doncs, és lògic preguntar-se quins atacs quàntics es coneixen a dia d'avui vers el criptosistema NTRU i fins a quin punt en comprometen la seva seguretat. En resposta a la pregunta anterior, convé notar que existeixen dues línies d'investigació clarament diferenciades que pretenen resoldre el SVP i, en conseqüència, solucionar el KRP des d'un enfocament quàntic.

D'una banda, la primera de les línies d'investigació es correspon amb l'estudi del problema del subgrup amagat, el qual disposa d'una importància cabdal dins del camp de la computació quàntica i té l'enunciat següent.

Definició 45. *Donat un grup G i una funció $f : G \to X$, s'anomena problema del subgrup amagat o HSP (de l'anglès* Hidden Subgroup Problem*) a trobar, a partir d'alguns dels valors de f, un subgrup $H \subset G$ tal que la funció f es mantingui constant per les seves classes d'equivalència per l'esquerra.*

Seguint el fil de l'esmentat a l'inici d'aquest mateix capítol, s'ha demostrat que l'existència d'una solució al HSP en el cas diedral implica de forma directa l'existència d'un algorisme quàntic capaç de resoldre en temps polinomial la versió única del SVP. Ara bé, malgrat que bona part dels atacs quàntics al criptosistema NTRU pretenen explotar aquesta relació, el cert és que a dia d'avui encara es no es coneix cap algorisme capaç de resoldre el cas diedral del HSP. Com a apunt final, convé remarcar que l'estudi del problema del subgrup amagat, així com el seguit de definicions i resultats que el relacionen amb el SVP, escapen de l'abast d'aquest treball i es poden trobar amb tot detall a [Lom04].

Per contra, l'altra línia d'investigació consisteix en millorar alguns algorismes clàssics que resolen el SVP mitjançant subrutines quàntiques, en concret l'algorisme de cerca de Grover, que en redueixin la complexitat computacional. Entrant en detalls, bona part dels algorismes existents en l'actualitat que donen una solució exacta al SVP o al CVP, com el presentat a [AKS01], empren tècniques de cribatge i saturació que requereixen de múltiples búsquedes d'elements dins de conjunts. En la computació clàssica, aquesta tipologia de búsquedes té una complexitat de $\mathcal{O}(N)$, on N es correspon a la cardinalitat del conjunt. En canvi, l'algorisme de cerca de Grover permet reduir-ne la complexitat en un factor d'arrel quadrada, deixant-la tan sols en $\mathcal{O}(\sqrt{N})$. D'aquesta manera, mitjançant l'algorisme de Grover es pot reduir la complexitat dels algorismes que usen el cribatge i la saturació, si bé continuen disposant d'una complexitat exponencial. Una explicació més detallada d'aquest procés i de les corresponents estimacions teòriques es pot trobar a [LMvdP13].

En base al presentat en aquesta secció, és clar que a dia d'avui els atacs quàntics no suposen una amenaça real de cara al criptosistema NTRU. Ara bé, tampoc existeixen

garanties que en un futur segueixi així, per la qual cosa cal estar pendent de l'evolució de la computació quàntica.

4.5.7 Nivells de seguretat

Fins ara, s'ha associat la seguretat del criptosistema NTRU a l'estudi de la viabilitat dels diversos atacs existents contra ell. Aquest enfocament, malgrat ser lògic i intuïtiu, manca del rigor esperat i en dificulta la comparativa tant amb altres criptosistemes com amb les seves pròpies instàncies. Així doncs, és interessant establir un valor numeric que d'alguna manera mesuri amb precisió el nivell de seguretat del criptosistema en funció dels paràmetres públics emprats i que sigui independent del criptosistema que s'utilitza. L'indicador encarregat de formalitzar aquests requeriments és el que es coneix com a nivell de seguretat (en anglès *security level*).

Tècnicament, el nivell de seguretat d'un criptosistema es defineix com el número associat a la quantitat de treball que es requereix per tal de trencar-lo i s'expressa en bits. Més concretament, un nivell de seguretat k fa referència a que l'atacant ha hagut d'emprar un total de 2^k operacions per trencar el criptosistema. Per tal d'estipular-lo, es mesura en funció de l'atac més potent que es conegui fins al moment, que en el cas del criptosistema NTRU es correspon amb l'atac híbrid que utilitza tècniques de reducció de reticles i de trobada a mig camí detallat a [HG07].

Tot seguit, la taula 4.2 llista els nivells de seguretat del criptosistema NTRU en base als valors dels paràmetres públics. Aquests resultats són originaris de [HPS+17] i es corresponen a les estimacions actuals.

(*N*, *p*, *q*, *d*)	**Nivell de seguretat (bits)**	**Llargada claus privades (bits)**	**Llargada clau pública (bits)**
(163, 3, 1024, 54)	35	517	1630
(199, 3, 1024, 66)	46	631	1990
(251, 3, 1024, 84)	66	796	2510
(293, 3, 2048, 98)	74	929	3223
(347, 3, 2048, 116)	99	1100	3817
(401, 3, 2048, 134)	116	1272	4411
(439, 3, 2048, 146)	133	1392	4829
(593, 3, 2048, 247)	256	1880	6523
(745, 3, 2048, 204)	354	2362	8195

Taula 4.2: Taula que relaciona els paràmetres públics amb el nivell de seguretat i la llargada de les claus públiques i privades del criptosistema NTRU.

En referència a la taula 4.2, i tenint en compte la tecnologia actual, es considera que per $N = 401$ el criptosistema NTRU ofereix un grau de seguretat més que elevat. Tot i això, degut a l'augment de la mida de les claus públiques i privades, sovint interessa reduir el nivell de seguretat en pro d'una major eficiència. En afegit, per tal de copsar la mutabilitat dels nivells de seguretat, és interessant notar que en la publicació original del criptosistema NTRU l'any 1998 [HPS98] s'assignava a $N = 167$ un nivell de seguretat de 82, un valor considerat aleshores elevat. Per contra, a dia d'avui, per $N = 163$ s'estableix un nivell de seguretat de 35, menys de la meitat. Evidentment, aquests canvis es deuen a l'aparició de nous atacs i a la millora dels ja existents, per la qual cosa es recomana

no considerar mai els nivells de seguretat com un resultat definitiu, sinó com quelcom temporal.

D'altra banda, tal i com s'ha comentat prèviament, l'altra gran utilitat d'aquesta mètrica és la comparació entre criptosistemes. En aquest sentit, el nivell de seguretat actua com punt de partida per garantir que els criptosistemes que es confronten disposen del mateix grau de resistència i, per tant, té sentit comparar-ne aspectes com ara el rendiment o l'eficiència. A tall d'exemple, a partir de les dades extretes de [LKK99], la taula que segueix compara la mida de la clau pública del criptosistema NTRU amb les anàlogues dels criptosistemes més comuns en l'actualitat: RSA i ECC.

Nivell de seguretat (bits)	Llargada clau pública NTRU (bits)	Llargada clau pública ECC (bits)	Llargada clau pública RSA (bits)
80	3817	160	1024
112	4411	224	2048
128	4829	256	3072
192	6523	384	7680
256	8195	512	15360

Taula 4.3: Taula que relaciona les mides en bits de les claus públiques dels criptosistemes NTRU, RSA i ECC en funció del seu nivell de seguretat.

En base a aquesta darrera taula, s'observa com en tots els nivells de seguretat el criptosistema ECC destaca per la mida considerablement reduïda de la seva clau pública. A més a més, també indica que el criptosistema NTRU augmenta la seva eficiència respecte al cripostistema RSA a mesura que augmenten els nivells de seguretat. No obstant això, és important tenir present que la mida de la clau pública no és l'únic factor a tenir en compte a l'hora de comparar diferents criptosistemes, ja que existeixen molts altres factors que juguen un paper important de cara al seu rendiment, com poden ser el temps d'encriptació o desencriptació entre d'altres.

Capítol 5

Implementació pràctica

Un cop vist amb detall les bases teòriques sobre les quals es fonamenten tant el criptosistema NTRU com els atacs existents vers ell, convé posar-los en pràctica per tal de veure'n de primera mà les seves prestacions. Així doncs, al llarg d'aquesta secció es relata el seu procés d'implementació i les principals decissions de disseny, mentre que a l'Annex s'hi detallen les conclusions experimentals que se n'extreuen. Més concretament, l'objectiu principal darrere d'aquest apartat no és altre que el de transformar el seguit d'enunciats i proposicions provients de la criptografia teòrica en un programa funcional que sigui capaç d'encriptar i desencriptar missatges. En aquest sentit, s'avantposa la vessant didàctica i la funcionalitat del codi a la seva optimització computacional.

En referència al programa resultant, es correspon a un *script* i un *notebook* que disposen de les següents funcionalitats:

- Instanciar el criptosistema NTRU amb els paràmetres públics a escollir i generar fitxers de text amb les claus públiques i privades que se'n deriven.
- Encriptar fitxers de text fent ús de la clau pública generada.
- Desencriptar fitxers de text fent ús de la clau privada generada.
- Atacar el criptosistema mitjançant l'atac a la vulnerabilitat dels paràmetres.
- Atacar el criptosistema mitjançant l'atac per força bruta.
- Atacar el criptosistema mitjançant l'atac per trobada a mig camí.
- Atacar el criptosistema mitjançant l'atac basat en reticles al KRP.

El codi en qüestió es troba disponible al següent repositori públic de *GitHub*: Criptosistema NTRU i atacs diversos.

5.1 Entorn de programació

En primer lloc, és important especificar que la totalitat del codi ha estat desenvolupat mitjançant el software lliure *SageMath*, un sistema algebraic computacional escrit en llenguatge *Python* i que es construeix sobre múltiples paquets com ara *NumPy*, *Maxima* o *R*.

La seva tria rau en el fet que *SageMath* destaca per tenir un ampli ventall de funcionalitats referents a l'àlgebra abstracta i, en particular, sobre els conceptes relacionats amb el criptosistema NTRU: els reticles i els anells polinomials convolucionals. De fet, en l'actualitat *SageMath* és emprat en multitud de publicacions matemàtiques, tant en el camp de la criptografia com en molts d'altres. En afegit, la seva sintaxi senzilla i intuïtiva facilita la comprensió i lectura del codi, fet que lliga perfectament amb la voluntat d'implementar un codi clar i entenedor. En cas de voler descarregar-lo o consultar-ne una guia detallada cal visitar [Dev21].

Ara bé, com a contrapartida, cal puntualitzar que a l'estar escrit sobre *Python*, *SageMath* no destaca per la seva velocitat d'execució, fet que comporta que, tal i com succeeix amb infinitat de programes, la versió més optimitzada que es coneix del criptosistema NTRU (la qual es pot trobar a la pàgina web del NIST) estigui implementada mitjançant el llenguatge *C*.

5.2 Implementació del criptosistema NTRU

Tot seguit, es detalla breument l'estratègia duta a terme de cara a la creació del codi, el qual és idèntic en el *script* i en el *notebook*. A més a més, totes les funcionalitats explicades a continuació disposen de la possibilitat d'executar-se amb i sense lectura/escriptura de fitxers. No obstant això, per motius de concreció en aquesta memòria únicament es relaten els fluxos d'execució que treballen amb fitxers de text i prenent el *script* com a referència. En línies generals, el funcionament del codi no disposa d'excessives diferències quan no s'usen les funcions referents als fitxers de text. Per tal de veure'n una explicació més precisa, extensa i línia a línia es recomana llegir el codi comentat que es pot trobar al repositori de *GitHub* citat prèviament. A tall d'exemple, al *notebook* s'hi mostren els resultats d'una execució completa del codi.

D'entrada, el primer pas referent al desenvolupament del *script* es correspon amb la implementació del criptosistema NTRU. Per tal d'aconseguir-ho, al fitxer `ntruClass` s'hi ha creat la classe `Ntru`, que permet generar les claus públiques i privades, encriptar i desencriptar fitxers a partir dels paràmetres públics N, p, q i d escollits per l'usuari. No obstant això, per tal de garantir-ne la seguretat i un cert grau d'eficiència el *script* imposa que q sigui potència de 2, p sigui primer o potència de 2 i que es compleixin les condicions (4.1) i (4.2). A més a més, s'ha creat el fitxer `auxMethods` que, tal i com el seu nom indica, disposa d'un seguit de funcions auxiliars que són d'utilitat a l'hora de treballar amb el criptosistema NTRU. Alguns dels mètodes que s'hi poden trobar cobreixen aspectes com el càlcul de polinomis inversos o la lectura i escriptura de fitxers.

5.2.1 Generació de les claus públiques i privades

Pel que fa la generació de les claus, en línies generals el procediment és anàleg al relatat a l'apartat 4.2.2. Entrant en detalls, mitjançant l'ús d'un bucle `while` i de funcions pròpies de *SageMath* com per exemple `is_unit`, s'itera sobre el conjunt de polinomis candidats a esdevenir la clau privada $f(x)$ fins a trobar-ne un que disposi d'inversos a R_p i a R_q. En afegit, és necessari puntualitzar que a fi de calcular l'invers de $f(x)$ a R_q,

al no tractar-se d'un cos sinó tan sols d'un anell commutatiu, *SageMath* no disposa de cap mètode *built in* per calcular-lo. Ara bé, aprofitant que q és potència de 2 i seguint el mètode vist a la Proposició 35, s'ha construït l'algorisme `inv_pol_mod_power_2`, al fitxer `auxMethods`, capaç d'obtenir l'invers buscat. En cas de desitjar una explicació detallada d'aquest i d'altres mètodes alternatius és recomanable consultar [Sil99a]. Finalment, un cop generades, existeix l'opció d'emmagatzemar tant els coeficients de les claus com els paràmetres públics en fitxers de text mitjançant els mètodes `set_param_and_pub_key` i `set_param_and_priv_keys` amb la voluntat de facilitar-ne el seu posterior ús. D'aquesta manera, d'una banda, quan hom vol compartir els paràmetres i la clau pública del criptosistema NTRU a la resta d'agents per tal que en puguin encriptar missatges, únicament cal que els hi faciliti el fitxer generat per `set_param_and_pub_key` i; de l'altra, quan es pretén desencriptar un missatge, només cal fer servir les claus privades que apareixen al fitxer indicat a `set_param_and_priv_keys`. Com a apunt tècnic, és necessari precisar que a fi de facilitar-ne la posterior lectura, cada paràmetre i clau està separat de la resta per un ; dins del fitxer de text.

5.2.2 Encriptació

Seguidament, en referència al procés d'encriptació d'un fitxer de text, abans d'encriptar el missatge cal solventar dues problemàtiques més enllà de la pròpia lectura del fitxer. Primerament, cal traduir el text a un format que permeti interpretar-lo com a coeficients d'un polinomi i, acte seguit, dividir-lo de tal manera que la mida dels polinomis resultants no excedeixi la delimitada pel paràmetre públic N. Per tal d'aconseguir-ho, la decisió optada ha estat la de codificar en binari el text en el mètode `encrypt_file`, interpretar cada bit com a un coeficient diferent i separar la cadena de bits obtinguda en missatges de mida N gràcies a la funció `split_message`. Endemés, és interessant remarcar que aquest procediment ha estat dut a terme per cada línia de forma independent, ja que d'aquesta manera també es preserva l'estructura del text a l'encriptar-lo. Un cop codificat el text a binari i fragmentat en missatges de mida N, cadascun d'aquests missatges s'encripa seguint exactament els mateixos passos que figuren a la secció 4.2.3. De fet, per evitar atacs seguint l'estratègia vista a l'apartat 4.5.3, al mètode `encryption` s'imposa que el polinomi aleatori $r(x)$ sigui diferent a cada fragment que s'encripta. Per acabar, s'escriuen al nou fitxer de text els coeficients corresponents als polinomis encriptats línia a línia.

5.2.3 Desencriptació

Pel que respecta a la desencriptació, a grans trets el *script* desfà el camí dut a terme prèviament per l'encriptació. En concret, el primer pas consisteix a llegir el fitxer encriptat i interpretar-ne el text com a polinomis de mida N mitjançant el mètode `decrypt_file`. En aquest cas, aquesta tasca és més senzilla, ja que el missatge encriptat ja es correspon amb els coeficients dels polinomis, per la qual cosa no cal traduir-ne el contingut, sinó que tan sols cal fragmentar-los. A continuació, la funció `decryption` desencripta cada missatge tal i com s'indica a l'apartat 4.2.4 i es concatenen els coeficients obtinguts formant una considerable tira binària. Finalment, es decodifica el missatge binari obtenint el text inicial i s'escriu al fitxer de text indicat. Tal i com succeeix amb l'encriptació, aquest procés es

duu a terme línia a línia.

5.3 Implementació dels atacs

De la totalitat d'atacs detallats al llarg de la memòria, al *script* s'ha optat per programar-hi una selecció procurant donar-hi cabuda a totes les tècniques possibles i garantir-ne resultats totals. Així doncs, s'ha descartat implementar-hi els atacs per transmissió múltiple degut a que, tal i com s'ha vist a la secció 4.5.3, aquesta tipologia d'atacs no solen obtenir el missatge original en la seva totalitat, sinó tan sols informació parcial. De fet, anàlogament al que succeeix amb la implementació del criptosistema NTRU, la voluntat darrere la programació d'aquests atacs és facilitar-ne la comprensió i explorar les seves possibilitats. A més a més, tot i que els procediments que es detallen a continuació requereixen de l'ús de fitxers de text, el cert és que el codi també permet executar els atacs mitjançant només l'entradada de paràmetres. A nivell de disseny, s'ha decidit crear un fitxer i classe nous per cada atac per així promoure el principi de responsabilitat única i la independència entre funcionalitats.

5.3.1 Atac a la vulnerabilitat dels paràmetres

Abans de res, és necessari concretar que l'atac a la vulnerabilitat dels paràmetres programat es correspon al que s'explota quan $gcd(p, q) = c$ amb $c \neq 1$ i que el codi en qüestió es troba a la classe `Param_vulnerability` situada al fitxer `vulParamAttack`.

Pel que respecta a l'atac en sí, convé notar que el seu èxit o fracàs depèn exclusivament dels paràmetres públics amb els quals hagi estat instanciat el criptosistema NTRU. Per tant, com és lògic, el primer pas consisteix en llegir els paràmetres públics del fitxer de text corresponent a la clau pública i comprovar, al mètode `attack_file`, si p i q són coprimers. En cas que no sigui així, l'execució finalitza i s'indica per pantalla la impossibilitat d'emprar l'atac. Per contra, si l'atac es pot dur a terme, el següent pas a realitzar es basa en llegir el fitxer encriptat seguint el mateix procediment explicat per la desencriptació, és a dir, treballant línia a línia, interpretant el text com a coeficients i fragmentant-lo en polinomis de mida N. Tot seguit, a la funció `attack` s'ataca cada missatge en base al procés detallat a la secció 4.5.1. En aquest sentit, convé notar que degut a les restriccions imposades als paràmetres públics s'obté $gcd(p, q) = p$ i, per tant, l'atac desencripta íntegrament el missatge i no pas tan sols informació parcial. Finalment, tal i com succeeix al procés de desencriptació, resta descodificar els valors binaris obtinguts i escriure'n el text resultant al fitxer indicat.

Més enllà de la seva implementació, un aspecte computacional a destacar d'aquest atac és que és pràcticament igual d'eficient que la pròpia desencriptació del criptosistema, tal i com ho corroboren els resultats experimentals de l'Annex. Això és conseqüència del fet que per obtenir el missatge original tan sols cal reduir mòdul p el text encriptat enlloc de realitzar dos productes com es fa en el procés habitual de desencriptació. Nogensmenys, com a contrapartida, es tracta d'un atac extremadament simple d'evitar mitjançant una tria adequada dels paràmetres p i q.

5.3.2 Atac per força bruta

Primerament, cal indicar que aquest atac es pot trobar a la classe `Brute_force` dins del fitxer `bfAttack` i, contràriament al que succeeix amb l'atac a la vulnerabilitat dels paràmetres, per excutar l'atac no només es necessiten els valors dels paràmetres públics, sinó que també es necessita disposar de la clau pública $h(x)$. A més a més, en aquest cas el gruix de l'atac es basa en l'obtenció de les claus privades i no pas en la desencriptació del missatge per una via alternativa. Així doncs, un cop dut a terme l'atac com a tal, també s'ha de realitzar el procés de desencriptació propi del criptosistema NTRU.

Fent referència a la seva implementació, el primer pas consisteix en llegir els paràmetres i la clau pública del seu fitxer corresponent. Seguidament, s'executa l'atac com a tal dins del mètode `obtain_priv_keys` seguint meticulosament la secció 4.5.2. Entrant en detalls, a partir d'una llista amb $d+1$ valors iguals a 1, d valors iguals a -1 i $N-2d-1$ valors iguals a 0 i mitjançant la llibreria `Sympy` pròpia de *Python*, s'obtenen tots els candidats a esdevenir la clau pública $f(x)$. D'aquesta manera, fent servir iteradors dins d'un bucle es calcula el producte (4.11) pels diferents polinomis $f(x)$ fins que els mètodes `checking` i `is_ternary` determinen que un dels resultats compleix les condicions per ser l'altre clau privada $g(x)$. Finalment, es desencripta el missatge i s'emmagatzema el text resultant en un fitxer tal i com s'ha detallat en la secció referent a la desencriptació.

Computacionalment parlant, els resultats de l'Annex mostren com l'atac requereix d'un nombre considerable d'operacions, en especial dins del bucle principal. Així doncs, l'execució del *script* resulta impracticable per valors de N superiors a 31 en un portàtil convencional, fet que corrobora experimentalment l'anàlisi teòric vist a l'apartat 4.5.2.

5.3.3 Atac per trobada a mig camí

La classe `Meet_in_the_middle` del fitxer `mitmAttack` és la que disposa del codi corresponent a l'atac per trobada a mig camí. A més a més, malgrat que conceptualment sigui tan sols una optimització de l'atac per força bruta, el cert és que a nivell d'implementació esdevé força diferent.

Pel que fa al codi desenvolupat, tal i com succeeix amb l'atac per força bruta, en un inici és imprescindible llegir d'un fitxer els paràmetres i la clau pública per tal de poder obtenir, posteriorment, les claus privades. No obstant això, precisament és en el mètode `obtain_priv_keys` on apareixen les grans diferències entre ambdues implementacions, ja que en aquest cas es segueixen les indicacions donades a la secció 4.5.4. En aquest sentit, convé resaltar els mètodes `classify` i `convert_to_bits`, els quals s'encarreguen de la classificació dels diversos polinomis $a_1(x)$ i $a_2(x)$ en calaixos i de la conversió dels seus coeficients en bits respectivament. De fet, per tal d'optimitzar l'execució, durant la pròpia classificació del conjunt de polinomis $a_2(x)$ ja es realitza la comprovació final mitjançant l'ús del mètode `checking`, que s'encarrega d'obtenir el polinomi candidat a clau privada $f(x)$ prenent com a opcions inicials els polinomis $a(x)$ resultants de concatenar el polinomi $a_2(x)$ que s'està classificant amb els polinomis $a_1(x)$ pertanyents al seu mateix calaix. Un cop trobades les claus privades, tan sols resta desencriptar el missatge i emmagatzemar el text en un fitxer.

Com a apunt final, en base als resultats experimentals situats a l'Annex, convé no-

tar que el rendiment d'aquest atac és superior al de força bruta, corroborant l'esmentat a l'apartat 4.5.4. Tot i això, el seu creixement exponencial continua sent excessiu per a poder ser utilitzat a la pràctica. Més concretament, si bé és cert que es realitzen menys iteracions en el procés d'obtenció de les claus privades, l'augment de memòria que requereix la classificació prèvia dels polinomis suposa un contrapès a tenir en compte que el fa impracticable quan $N > 31$.

5.3.4 Atac al KRP mitjançant reticles

El codi referent a la implementació d'aquest atac es correspon a la classe `Krp_lattice`, dins del fitxer `krpRetAttack`. Endemés, la seva implementació destaca per la seva brevetat i senzillesa, ja que el gruix de l'execució ve donada per mètodes propis de *SageMath*.

Entrant en detalls, de la mateixa manera que en la majoria d'atacs, el primer pas consisteix en llegir el fitxer corresponent als paràmetres i a la clau pública. Acte seguit, dins del mètode `gen_M_matrix` es genera la matriu M_h, enunciada a la Definició 39, sobre la qual s'aplica l'algorisme LLL mitjançant la funció `LLL` de *SageMath*, el qual permet obtenir les claus privades $f(x)$ i $g(x)$ tal i com s'explica a la secció 4.5.5. De fet, per construir la matriu M_h primer es calculen les 4 submatrius i gràcies al mètode `block_matrix` s'ajunten per formar-la. Per concloure la implementació, únicament cal fer servir les claus privades obtingudes per desencriptar el missatge i emmagatzemar el text resultant en un fitxer.

Pel que fa al seu rendiment computacional, a partir dels valors obtinguts experimentalment citats a l'Annex, cal destacar es tracta de l'atac amb millors resultats en bona part gràcies a la complexitat polinomial de l'algorisme LLL. Així doncs, en un temps reduït (al voltant de mig minut), és capaç de trencar instàncies del criptosistema NTRU amb valors de $N \approx 70$. Ara bé, a mesura que augmenta la N tant el seu rendiment com la seva precisió decauen, tal i com s'ha explicat a la secció 2.3.3.

Capítol 6

Conclusions

Malgrat que la seva invenció data de l'any 1998, a partir del vist al llarg del treball es pot concloure que el criptosistema NTRU és una alternativa postquàntica viable de cara a esdevenir el sistema d'encriptació de clau pública de referència en les properes dècades. La mida de les seves claus (no major a 1000 bytes), juntament amb una considerable velocitat d'encriptació i desencriptació, fan que sigui un criptosistema eficient. A més, a nivell de seguretat es mostra resistent, ja que tots els atacs capaços de trencar-lo, tant clàssics com quàntics, són de complexitat exponencial.

Entrant en detalls, l'estudi del criptosistema NTRU ha permès presentar i desenvolupar alguns dels principals resultats de la teoria de reticles, com poden ser el SVP, el CVP, l'heurística Gaussiana, les bases reduides o l'algorisme LLL. Tots aquests conceptes, amb l'afegit dels anells polinomials convolucionals, han estat essencials per entendre en profunditat el funcionament i seguretat del criptosistema i tenen moltes altres aplicacions més enllà de la criptografia.

Enllaçant-ho amb això, l'anàlisi del criptosistema ha estat dividit en dues vessants molt marcades, la teòrica i la pràctica. Pel que fa a la part teòrica, s'ha demostrat el correcte funcionament dels mecanismes d'encriptació i desencriptació, s'ha definit la mida de les claus públiques i privades i s'han presentat els atacs existents fent èmfasi en la seva complexitat computacional. A nivell pràctic, s'ha implementat una versió del criptosistema i quatre dels atacs coneguts, on cadascun d'ells es correspon a una estratègia diferent. D'aquesta manera, ha estat possible obtenir els resultats experimentals presentats a l'Annex on s'observa la impossibilitat de trencar el criptosistema per valors de N relativament grans ($N \approx 100$), corroborant així la resistència del criptosistema.

Per finalitzar, es proposen un seguit de línies d'investigació que donen continuïtat a aquest treball. D'una banda, existeix la possibilitat de millorar l'eficiència del criptosistema NTRU mitjançant tècniques d'acceleració com les proposades a [BDLJ14]. D'altra banda, es pot prosseguir amb l'anàlisi i implementació d'atacs híbrids que compaginen tècniques com la trobada a mig camí i les reduccions de reticles, com l'exposat a [HG07]. Finalment, la darrera proposta es correspon amb l'estudi en profunditat d'altres criptosistemes basats en reticles, en especial els presententats breument a l'Annex d'aquest projecte: CRYSTALS-KYBER i SABER. Ambdós són també finalistes del concurs del NIST i es consideren alternatives molt prometedores dins del camp de la criptografia postquàntica.

Bibliografia

[AD97] Miklós Ajtai and Cynthia Dwork, *A public-key cryptosystem with worst-case/average-case equivalence*, Proceedings of the Twenty-Ninth Annual ACM Symposium on Theory of Computing, STOC '97, Association for Computing Machinery, 1997, p. 284–293.

[AKS01] Miklós Ajtai, Ravi Kumar, and D. Sivakumar, *A sieve algorithm for the shortest lattice vector problem*, Proceedings of the Thirty-Third Annual ACM Symposium on Theory of Computing, STOC '01, Association for Computing Machinery, 2001, p. 601–610.

[Bab86] László Babai, *On lovász' lattice reduction and the nearest lattice point problem*, Combinatorica **6** (1986), 1–13.

[BDK+18] Joppe Bos, Leo Ducas, Eike Kiltz, T. Lepoint, Vadim Lyubashevsky, John M. Schanck, Peter Schwabe, Gregor Seiler, and Damien Stehle, *Crystals - kyber: A cca-secure module-lattice-based kem*, 2018 IEEE European Symposium on Security and Privacy (EuroS P), 2018, pp. 353–367.

[BDLJ14] Tianyu Bai, Spencer Davis, Juanjuan Li, and Hai Jiang, *Analysis and acceleration of ntru lattice-based cryptographic system*, IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2014, pp. 1–6.

[BPR12] Abhishek Banerjee, Chris Peikert, and Alon Rosen, *Pseudorandom functions and lattices*, EUROCRYPT, 2012, pp. 228–245.

[Chi09] Lindsay N. Childs, *A concrete introduction to higher algebra*, pp. 364–366, Springer, 2009.

[Dev21] The Sage Developers, *Sagemath, the sage mathematics software system (version 9.2)*, `https://www.sagemath.org`, 2021.

[DH76] W. Diffie and M. Hellman, *New directions in cryptography*, IEEE Transactions on Information Theory **22** (1976), 644–654.

[DKR+18] Jan-Pieter D'Anvers, Angshuman Karmakar, Sujoy Sinha Roy, T. Lepoint, Vadim Lyubashevsky, John M. Schanck, and Frederik Vercauteren, *Saber: Module-lwr based key exchange, cpa-secure encryption and cca-secure kem*, IACR Cryptol, 2018, pp. 282–305.

[For09] Lance Fortnow, *The status of the p versus np problem*, Communications of the ACM, vol. 2, 2009, pp. 78–86.

[FS15] Scott Fluhrer and Cisco Systems, *Quantum cryptanalysis of ntru*, 2015.

[Gal12] Steven D. Galbraith, *Mathematics of public key cryptography*, ch. Lattice Basis Reduction, Cambridge University Press, 2012.

[Gen01] Craig Gentry, *Key recovery and message attacks on ntru-composite*, Advances in cryptology, 2001, pp. 182–194.

[HG07] Nick Howgrave-Graham, *A hybrid lattice-reduction and meet-in-the-middle attack against ntru*, Advances in Cryptology, Crypto, 2007, pp. 150–169.

[HGSW03] Nick Howgrave-Graham, Joseph H. Silverman, and William Whyte, *A meet-in-the-middle attack on an ntru private key*, Technical Report (2003).

[HPS98] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman, *Ntru: A ring-based public key cryptosystem*, ANTS, 1998.

[HPS08] ———, *An introduction to mathematical cryptography*, ch. Lattices and Cryptography, Springer, 2008.

[HPS+17] Jeffrey Hoffstein, Jill Pipher, John Schanck Joseph H. Silverman, William Whyte, and Zhenfei Zhang, *Choosing parameters for ntruencrypt*, Cryptographers' Track at the RSA Conference, 2017, pp. 3–18.

[Hux96] Martin N. Huxley, *Lattice points and exponential sums*, London Mathematical Society Monographs, 1996.

[Jud19] Thomas M. Judson, *Abstract algebra*, Stephen F. Austin State University, 2019.

[LKK99] Hitesh Loriya, Anunay Kulshreshta, and R. Keraliya, *Security analysis of various public key cryptosystems fro authentication and key agreement in wireless communications network*, International Journal of Advanced Research in Computer and Communication Engineering **6** (1999), 267–274.

[LLL82] Arjen K. Lenstra, Hendrik W. Lenstra, and László Miklós Lovász, *Factoring polynomials with rational coefficients*, Mathematische Annalen **261** (1982), 515–534.

[LMvdP13] Thijs Laarhoven, Michele Mosca, and Joop van de Pol, *Solving the shortest vector problem in lattices faster using quantum search*, PQCrypto 2013, Springer, 2013, pp. 83–101.

[Lom04] Chris Lomont, *The hidden subgroup problem - review and open problems*, 2004.

[Mag18] Peter Magyar, *Polynomial euclidean algorithm*, Michigan State University, 2018.

[May99] Alexander May, *Cryptanalysis of ntru*, 1999.

[MG02] Daniele Micciancio and Shafi Goldwasser, *Complexity of lattice problems: a cryptographic perspective*, The Kluwer International Series in Engineering and Computer Science, vol. 671, Kluwer Academic Publishers, 2002.

[Mic10] Daniele Micciancio, *Lattice algorithms and applications: Introduction to lattices*, University of California San Diego (2010).

[Ngu09] Phong Q. Nguyen, *The lll algorithm: Survey and applications*, ch. Hermite's Constant and Lattice Algorithms, Springer, 2009.

[Reg04] Oded Regev, *Quantum computation and lattice problems*, SIAM Journal on Computing **33** (2004), 738–760.

[Reg09] _____, *On lattices, learning with errors, random linear codes, and cryptography*, Journal of the ACM **56** (2009), no. 6.

[RSA78] R.L. Rivest, A. Shamir, and L. Adleman, *A method for obtaining digital signatures and public-key cryptosystems*, Communications of the ACM **21** (1978), 120–126.

[Sch09] Claus Schnorr, *Progress on lll and lattice reduction*, pp. 145–178, 02 2009.

[SE94] Claus Schnorr and M. Euchner, *Lattice basis reduction: Improved practical algorithms and solving subset sum problems*, Mathematical Programming **66** (1994), 181–199.

[SH21] Gilbert Strang and Edwin Herman, *Calculus: Change of variables in multiple integrals*, `https://math.libretexts.org/Bookshelves/Calculus/Book%3A_Calculus_(OpenStax)/15%3A_Multiple_Integration/15.7%3A_Change_of_Variables_in_Multiple_Integrals`, 2021, Últim acces 14 de setembre del 2021.

[Sho94] P.W. Shor, *Algorithms for quantum computation: discrete logarithms and factoring*, Proceedings 35th Annual Symposium on Foundations of Computer Science, 1994, pp. 124–134.

[Sil99a] Joseph H. Silverman, *Almost inverses and fast ntru key creation*, Technical Report (1999).

[Sil99b] _____, *Dimension-reduced lattices, zero-forced lattices and the ntru public key cryptosystem*, Technical Report (1999).

[SW03] Joseph H. Silverman and William Whyte, *Estimated breaking times for ntru lattices*, Technical Report (2003).

[Wen17] Shuyi Weng, *Orthogonal projections and the gram-schmidt process*, Northwester University (2017).

[Yan07] Igor Yanovsky, *Qr decomposition with gram-schmidt*, University of California Los Angeles (2007).

Annex A

Resultats experimentals

Aprofitant la implementació tant del criptosistema com de diversos dels seus atacs, és interessant corroborar l'estudi teòric realitzat al llarg del treball mitjançant comprovacions empíriques. Així doncs, la taula que es mostra a continuació recull el temps d'execució del procés de desencriptació d'un missatge de N enters en funció dels valors dels paràmetres públics N, p, q i d per part del criptosistema NTRU i dels diferents atacs. Tots els càlculs han estat fets en un Intel Core i5-7200U a 2,5 GHz i mesurats amb la funció de *Python* `time.time()`.

(N, p, q, d)	**NTRU (s)**	**VUL (s)**	**BF (s)**	**MITM (s)**	**RET (s)**
(11, 3, 512, 2)	0,015	Impossible de realitzar	0,067	0,210	0.243
(11, 3, 512, 4)	0,017	Impossible de realitzar	0,070	0,266	0,586
(11, 8, 512, 4)	0,016	0,059	0,104	0,247	0,254
(17, 3, 512, 2)	0,016	Impossible de realitzar	1,724	0,859	0,777
(17, 3, 512, 4)	0,017	Impossible de realitzar	19,143	3,863	0,897
(17, 8, 512, 4)	0,018	0,034	15,571	3,494	0,712
(23, 3, 512, 4)	0,022	Impossible de realitzar	646,599	36,490	1.480
(23, 3, 512, 8)	0,025	Impossible de realitzar	916,586	39,377	1,695
(23, 8, 512, 8)	0,024	0,046	899,241	37,281	1,572
(31, 3, 512, 8)	0,024	Impossible de realitzar	88385,264	5921,902	3,118
(31, 3, 512, 10)	0,027	Impossible de realitzar	91880,689	6973,787	3,673
(31, 8, 512, 10)	0,025	0,058	90263,178	6413,558	3,331

Taula A.1: Taula del temps d'execució de la desencriptació d'un missatge de N enters en funció dels paràmetres públics per part del criptosistema NTRU i dels diversos atacs.

En primer lloc, convé notar que les dades de la taula anterior es corresponen, en la seva totalitat, amb valors de N extremadament petits. Això es deu a que, tal i com s'ha explicat a l'apartat 4.5, a excepció de l'atac basat en reticles, la inviabilitat de la resta d'atacs creix exponencialment amb la mida del reticle. Per tant, l'única manera de poder-ne estudiar l'augment del temps d'execució és partint de valors reduïts. En afegit, és important precisar que les combinacions escollides de paràmetres públics no

són aleatòries, sinó que es corresponen a valors que compleixen les condicions (4.1), (4.2) i $d \approx N/3$.

Un cop fetes les consideracions respecte a la tria dels paràmetres públics, analitzant-ne els valors obtinguts es corrobora la gran millora a nivell de prestacions que suposen els atacs a la vulnerabilitat dels paràmetres (quan es pot dur a terme) i mitjançant reticles respecte els de força bruta i trobada a mig camí. Més concretament, s'observa com el temps d'execució de l'atac a la vulnerabilitat dels paràmetres és pràcticament idèntic al del propi criptosistema, independentment del valor de N. En canvi, als atacs per força bruta i trobada a mig camí, tal i com era d'esperar, el procés de desencriptació augmenta exponencialment a mesura que creix el grau dels polinomis. No obstant això, el que sí que ha suposat una sorpresa ha estat observar com el rendiment de l'atac per trobada a mig camí, malgrat ser superior al de força bruta en pràcticament la totalitat dels casos, ha disposat d'un factor de millora notablement menor que l'esperat d'arrel quadrada. Molt probablement, sigui conseqüència de l'aplicació de l'algorisme de col·lisió, el qual tot i permetre reduir el nombre de comprovacions a realitzar posteriorment, també disposa d'un cost per si mateix. Precisament, l'excés de memòria que requereix aquest algorisme és el que impossibilita executar l'atac per trobada a mig camí a partir de $N > 31$. Per acabar, cal indicar que l'atac mitjançant reticles escala polinomialment respecte N.

El gràfic semilogarítmic que segueix permet visualitzar de forma més clara el creixement del temps d'execució de cadascun dels mecanismes de desencriptació anteriors.

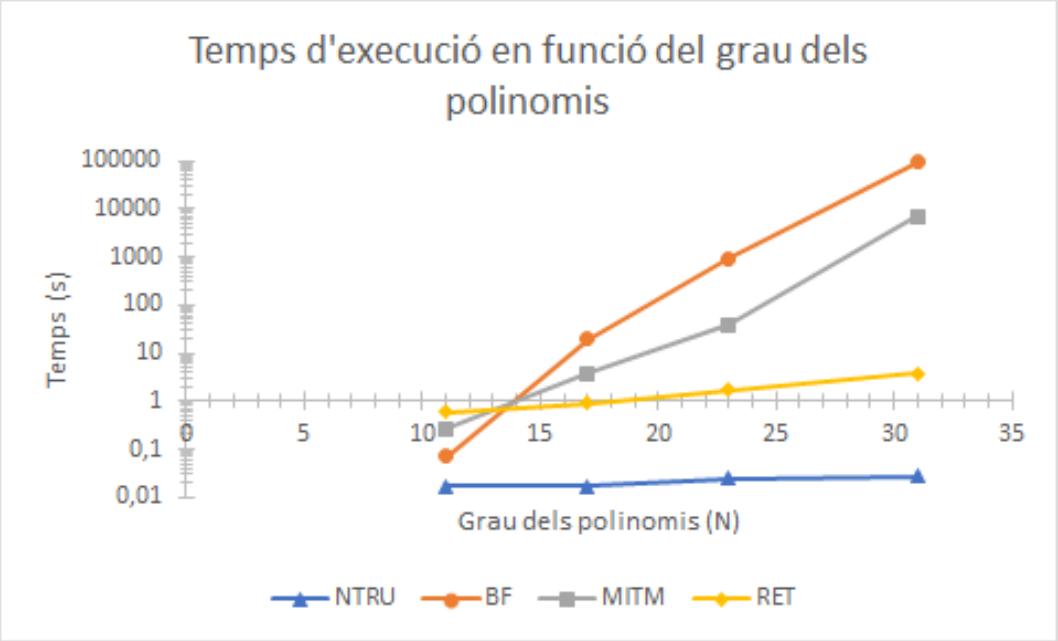

Figura A.1: Gràfic del temps d'execució en funció de N prenent els valors de la taula anterior amb $p = 3$ i $d \approx N/3$.

En base a la taula i gràfic anteriors, ha quedat palesa la gran superioritat de rendiment

de l'atac per reticles respecte a la resta per valors de N reduïts. No obstant això, cal tenir present que, a la pràctica, les implementacions emprades del criptosistema NTRU acostumen a disposar de valors de N superiors a 100. Així doncs, és interessant veure l'evolució de l'atac mitjançant reticles en funció del grau dels polinomis. Precisament això és el que llista la taula A.2.

(*N*, *p*, *q*, *d*)	**RET (s)**	**Llargada claus privades (bits)**	**Llargada clau pública (bits)**
(11, 3, 512, 4)	0,243	35	99
(17, 3, 512, 4)	0,897	54	153
(23, 3, 512, 8)	1,695	72	207
(31, 3, 512, 10)	3,673	99	279
(43, 3, 512, 14)	6,184	137	387
(53, 3, 512, 16)	9,662	169	477
(61, 3, 512, 18)	12,172	192	549
(73, 3, 512, 24)	19,112	232	657
(79, 3, 512, 26)	27,362	251	711

Taula A.2: Taula que relaciona els paràmetres públics amb el rendiment de l'atac mitjançant reticles i amb la llargada de les claus públiques i privades del criptosistema NTRU.

De fet, d'aquesta taula s'observa com l'evolució del temps d'execució de l'atac mitjançant reticles es manté en creixement polinomial en funció de N, corroborant la hipòtesi teòrica. Ara bé, a partir de $N > 79$ la reducció LLL deixa de ser d'utilitat per resoldre el SVP i per conseqüència per atacar al criptsosistema, ja que retorna un $q-$vector, és a dir, un vector amb una coordenada igual a q i la resta a 0. Això es deu a que, tal i com s'exposa en el Teorema 26, l'algorisme LLL resol l'apprSVP amb un factor de $\left(\frac{1}{\delta-\frac{1}{4}}\right)^{(n-1)/2}$, de manera que a mesura que augmenta N la precisió del resultat disminueix exponencialment. Per tal de resoldre aquesta problemàtica, es pot substituir l'algorisme LLL per l'algorisme BZZ-LLL, el qual disposa d'un factor β que millora la precisió de la resolució de l'apprSVP, però augmentant-ne la complexitat computacional a nivell exponencial en funció de β.

Annex B

CRYSTALS-KYBER

B.1 Introducció

En primer lloc, és important precisar que aquest capítol únicament pretent servir de breu introducció al criptosistema CRYSTALS-KYBER, presentant-ne les seves característiques principals per tal de coneixe'n les nocions bàsiques i en cap cas s'hi mostra un volum d'informació tan extens com en el capítol referent al NTRU. Així doncs, en cas de voler gaudir d'una explicació detallada del criptosistema es recomana consultar [BDK+18].

El criptosistema CRYSTALS-KYBER neix l'any 2017 a càrrec d'una amalgama d'investigadors de diversos centre de recerca europeus i nord-americans com a resposta al concurs de criptografia postquàntica del NIST. De la mateixa manera que succeeix amb el criptosistema NTRU, CRYSTALS-KYBER també s'engloba dins de la criptografia basada en reticles i es tracta d'un criptosistema probabilístic. Ara bé, enlloc de sustentar-se en la dificultat de resolució del SVP i del CVP, empra un altre problema matemàtic propi de la teoria de reticles conegut com a aprenentatge amb errors.

En referència a la seva seguretat i rendiment, de les conclusions publicades en la tercera ronda del concurs se n'extreu que ambdues mètriques són força similars a les del criptosistema NTRU. En afegit, si bé és cert que el nombre d'atacs que se'n coneixen és considerablement inferior al del NTRU, cal tenir en compte que es tracta d'un criptosistema de creació recent i, per tant, convé interpretar la dada amb cert esceptisime.

B.2 Aprenentatge amb errors

L'aprenentatge amb errors és un problema matemàtic presentat a la comunitat científica per part de Regev l'any 2009 [Reg09], motiu pel qual fou galardonat amb el premi Gödel el 2018. En aquest mateix article, més enllà del propi problema presentat, Regev enuncia dos altres resultats que juguen un paper important en la creació del criposistema CRYSTALS-KYBER: el disseny del primer criptosistema basat en l'aprenetatge amb errors i un resultat sobre la seva dificultat de resolució. Entrant en detalls, d'una banda s'estableix una equivalència entre la dificultat del problema d'aprenetatge amb errors i la dels pitjors casos d'alguns dels problemes clàssics de reticles, com ara la versió de decisió

del SVP; i de l'altra, el criptosistema proposat per Regev suposa la base sobre la que es s'inspira i fonamenta el criptosistema CRYSTALS-KYBER.

Un cop vist el context i la importància del problema d'aprenetage amb errors, convé enunciar-lo formalment. Precisament això és el que duu a terme la definició que segueix.

Definició 46. *Siguin $n, q \in \mathbb{N}$ amb $n \geq 1$ i $q \geq 2$ i sigui $\mathcal{X} : \mathbb{Z}_q \to \mathbb{R}^+$ una funció de distribució de probabilitats anomenada error. Donats n valors ϵ_i escollits segons $\mathcal{X}$, n vectors independents $a_i \in \mathbb{Z}_q^n$ i un vector qualsevol $s \in \mathbb{Z}_q^n$ es calcula $(a_i,\ a_i \cdot s + \epsilon_i)$ per a $i = 1, ..., n$. S'anomena problema d'aprenentatge amb errors o LWE (de l'anglès* Learning With Errors*) a trobar el vector s a partir de les parelles $(a_i,\ a_i \cdot s + \epsilon_i)$.*

Intuïtivament, hom pot pensar el LWE com la búsqueda d'una solució a un sistema d'equacions lineals mòdul q on cadascuna de les equacions disposa d'un error que ve donat per $\mathcal{X}$.

B.3 Funcionament

En línies generals, el funcionament darrere del mecanisme d'encriptació i desencriptació de CRYSTALS-KYBER és explotar la dificultat de solució del LWE. Així doncs, no és d'estranyar que la pràctica totalitat dels paràmetres públics presents al criptosistema siguin aquells que apareixen en la Definició 46, és a dir, n, q i $\mathcal{X}$. L'altre paràmetre restant és un enter d complint $d < \log_2(q)$.

Acte seguit, abans d'estudiar el funcionament del criptosistema, és necessari presentar breument dues funcions que hi juguen un paper cabdal. La funció $Com_{q,d}(x)$ envia un enter $x \in \mathbb{Z}_q$ a un element de $\{0, ..., 2^d - 1\}$ i la funció $Decom_{q,d}(x')$ actua com una espècie de pseudoinversa de la funció $Com_{q,d}(x)$, ja que a l'aplicar $Decom_{q,d}(Com_{q,d}(x))$ s'obté un element molt proper a x. Sense entrar en detalls, l'ús d'aquestes funcions permet reduir la mida dels paràmetres i desencriptar correctament el missatge.

Pel que fa a les claus, la clau pública es correspon amb la matriu $A \in \mathbb{Z}_q^{n \times n}$ que té per files els vectors $a_i \in \mathbb{Z}_q^n$ i el vector $t := Com_{q,d}(As + \epsilon)$, on $\epsilon = (\epsilon_1, ..., \epsilon_n)$ és el vector d'errors escollits segons $\mathcal{X}$ i s és un vector qualsevol, ambdós pertanyents a $\mathbb{Z}_q^n$; i la clau privada es correspon únicament al vector s, és a dir, el vector solució al LWE instanciat per la clau pública. Pel que respecta al missatge, es tracta d'un vector $m \in \{0, 1\}^n$.

En referència al mecanisme d'encriptació, l'emissor escull dos vectors aleatoris r i $e' \in \mathbb{Z}_q^n$ i un valor $e'' \in \mathbb{Z}_q$ segons $\mathcal{X}$ amb els que duu a terme els càlculs

$$u := Com_{q,d}(A^T r + e') \text{ i } v := Com_{q,d}(t^T r + e'' + \lfloor q/2 \rceil\, m)$$

i n'envia el resultat (u, v) al receptor.

Acte seguit, per tal de desencriptar el parell (u, v) i obtenir de nou el missatge m, el receptor únicament ha de realitzar l'operació

$$m' := Com_{q,d}(Decom_{q,d}(v) - s^T Decom_{q,d}(u)).$$

Annex C

SABER

C.1 Introducció

De la mateixa manera que succeeix amb l'apartat referent al criptosistema CRYSTALS-KYBER, aquesta secció de l'Annex tan sols preten presentar de forma esquemàtica el criptosistema SABER. Per tal d'obtenir una explicació amb tots els detalls es recomana consultar [DKR+18].

L'origen del criptosistema SABER es remunta a l'any 2018 quan va ser presentat per part de quatre investigadors de la univesitat belga KU Leuven, esdevenint el criptosistema més recent dels quatre finalistes al concurs del NIST i creat específicament per aquest certamen. Tal i com ja s'ha esmentat prèviament, SABER forma part de la branca de la criptografia basada en reticles, però en aquest cas la seva resistència està basada en el problema matemàtic de l'aprenentatge amb arrodoniment.

Enllaçant-ho amb això, en consonància amb els criptosistemes NTRU i CRYSTALS-KYBER, el criptosistema SABER també és probabilístic i disposa d'uns resultats similars a nivell de rendiment i seguretat. De fet, la pràctica totalitat de les decisions preses en el seu disseny, com per exemple prendre mòdul 2 en la majoria de quocients, van encaminades a garantir-ne la simplicitat i eficiència.

C.2 Aprenentatge amb arrodoniment

L'aprenentatge amb arrodoniment és un problema matemàtic proposat per Banerjee, Peikert i Rosen l'any 2011 [BPR12] i es considera una variant del problema d'aprenentatge amb arrodoniment en la que es substitueixen els errors presents als valors del LWE per un arrodoniment determinístic. En aquest sentit, es coneix l'equivalència entre la dificultat de resolució de la majoria d'instàncies del problema d'aprenentatge amb arrodoniment i la del LWE, fet que en justifica el seu ús com a base del criptosistema SABER. A més a més, aquest problema disposa d'altres utilitats pràctiques en el camp de la criptografia com poden ser la generació de funcions pseudoaleatòries. No obstant això, cal tenir present que es tracta d'un problema considerablement nou, per la qual cosa la comunitat científica encara desconeix fins a quin punt es pot extendre'n el seu ús.

La definició següent defineix de manera formal el problema d'aprenentatge amb arrodoniment.

Definició 47. *Siguin n, p, $q \in \mathbb{N}$ amb $n \geq 1$ i $q \geq p \geq 2$ i sigui $\lfloor \cdot \rfloor_{p,q} : \mathbb{Z}_q \to \mathbb{Z}_p$ la funció definida per $\lfloor x \rfloor_{p,q} = \lfloor (p/q) \cdot x \rfloor$. Donats n vectors independents $a_i \in \mathbb{Z}_q^n$ i un vector qualsevol $s \in \mathbb{Z}_q^n$ es calcula $(a_i,\ \lfloor a_i \cdot s \rfloor_{p,q})$ per a $i = 1, ..., n$. S'anomena problema d'aprenentatge amb arrodoniment o LWR (de l'anglès* Learning With Rounding*) a trobar el vector s a partir de les parelles $(a_i,\ \lfloor a_i \cdot s \rfloor_{p,q})$.*

De manera similar al que succeeix amb el LWE, es pot interpretar el LWR com la resolució d'un sistema lineal d'equacions mòdul q on el resultat ha estat arrodonit mòdul p.

C.3 Funcionament

L'elecció dels paràmetres públics està condicionada pel fet que el criptosistema SABER es fonamenta en el LWR. De fet, els valors n, p i q presents a la Definició 47 formen part del conjunt de paràmetres públics, el qual es veu completat per un vector $h \in \mathbb{Z}_q^n$ i un parell de valors $t \in \mathbb{Z}$ i $h' \in \mathbb{Z}_q$.

En referència a les claus, la clau pública consisteix en la parella (A, u), on A es correspon amb la matriu $A \in \mathbb{Z}_q^{n \times n}$ que té per files els vectors $a_i \in \mathbb{Z}_q^n$ i el vector u fa referència a $\lfloor As + h \rfloor_{p,q}$, on s és un vector qualsevol pertanyent a $\mathbb{Z}_q^n$. Per contra, la clau privada es correspon tan sols al vector s. El missatge m és un vector de $\{0,1\}^n$

Un cop establertes les claus, per tal d'encriptar el missatge m l'emissor escull un vector aleatori $s' \in \mathbb{Z}_q^n$, calcula

$$u' := \lfloor As' + h \rfloor_{p,q} \text{ i } v := \lfloor u'^T s' + h' + \frac{p}{2} m \rfloor_{2t,p}$$

i n'envia el resultat (u', v) al receptor.

Finalment, el mecanisme de desencriptació només consisteix en dur a terme el càlcul

$$m' := \lfloor u'^T s + h' + \frac{p}{2} m \rfloor_{2,p},$$

que amb una alta probabilitat coincideix amb el missatge m original.

Annex D

Planificació temporal

La taula que segueix detalla la planificació feta inicialment per aquest projecte en quinzenes.

Data	Tasca
juliol - agost	Recerca d'informació: articles, llibres, xerrades, etc.
1 - 15 setembre	Reticles: Definicions i propietats bàsiques Reticles: Problemes i heurístiques associades (1)
16 - 30 setembre	Reticles: Problemes i heurístiques associades (2) Reticles: Algorismes de reticles
1 - 15 octubre	Anells polinomials convolucionals: Definicions i propietats bàsiques Anells polinomials convolucionals: Invertibilitat polinomial Criptografia basada en reticles: Introducció
16 - 31 octubre	Criptografia basada en reticles: NTRU Criptografia basada en reticles: Problemes matemàtics associats a NTRU Implementació pràctica: Entorn de programació
1 - 15 novembre	Criptografia basada en reticles: NTRU com a criptosistema de reticles Implementació pràctica: Implementació del criptosistema NTRU
16 - 30 novembre	Criptografia basada en reticles: Seguretat del criptosistema NTRU (1) Implementació pràctica: Implementació dels atacs (1)
1 - 15 desembre	Criptografia basada en reticles: Seguretat del criptosistema NTRU (2) Implementació pràctica: Implementació dels atacs (2)
16 - 30 desembre	Introducció Conclusions Annex
1 - 15 gener	Revisió de la memòria i el codi

Taula D.1: Taula que llista la planificació inicial del projecte.

Cada tasca fa referència a tot el procés que requereix la seva realització, des de la documentació i comprensió dels continguts fins a la seva redacció a la memòria. A més a més, els números al costat dels títols de les tasques indiquen la seva partició. Pel que fa a la implementació pràctica, aquesta s'engloba dins de les tasques anomenades *Implementació*

pràctica. Així doncs, durant la primera quinzena de novembre estava previst desenvolupar el codi referent al criptosistema NTRU i entre la segona quinzena de novembre i la primera de desembre es pretenia implementar els diversos atacs.

Un cop vista la planificació inicial, convé destacar que s'ha seguit tal i com estava prevista a excepció del mes de novembre i de la primera quinzena de desembre. En aquest sentit, tal i com s'observa a la taula D.1, en un principi es pretenia desenvolupar el codi paral·lelament al seu estudi teòric, és a dir, anar realitzant la part pràctica a mesura que s'estudiaven les característiques del criptosistema i dels seus atacs. No obstant això, finalment s'ha optat per dedicar les tres primeres setmanes de novembre a la part teòrica del criptosistema i els seus atacs i les tres setmanes següents a desenvolupar tot el codi. D'aquesta manera, s'han pogut entendre amb calma tots els requeriments tècnics necessaris i acte seguit aplicar-los.

A continuació es mostra la taula corresponent al desenvolupament final d'aquest projecte.

Data	**Tasca**
juliol - agost	Recerca d'informació: articles, llibres, xerrades, etc.
1 - 15 setembre	Reticles: Definicions i propietats bàsiques Reticles: Problemes i heurístiques associades (1)
16 - 30 setembre	Reticles: Problemes i heurístiques associades (2) Reticles: Algorismes de reticles
1 - 15 octubre	Anells polinomials convolucionals: Definicions i propietats bàsiques Anells polinomials convolucionals: Invertibilitat polinomial Criptografia basada en reticles: Introducció
16 - 31 octubre	Criptografia basada en reticles: NTRU Criptografia basada en reticles: Problemes matemàtics associats a NTRU
1 - 22 novembre	Criptografia basada en reticles: NTRU com a criptosistema de reticles Criptografia basada en reticles: Seguretat del criptosistema NTRU
22 novembre - 15 desembre	Implementació pràctica: Entorn de programació Implementació pràctica: Implementació del criptosistema NTRU Implementació pràctica: Implementació dels atacs
16 - 30 desembre	Introducció Conclusions Annex
1 - 15 gener	Revisió de la memòria i el codi

Taula D.2: Taula que llista el desenvolupament final del projecte.

A mode de resum, tot seguit es presenta el diagrama de Gantt referent al desenvolupament final del treball.

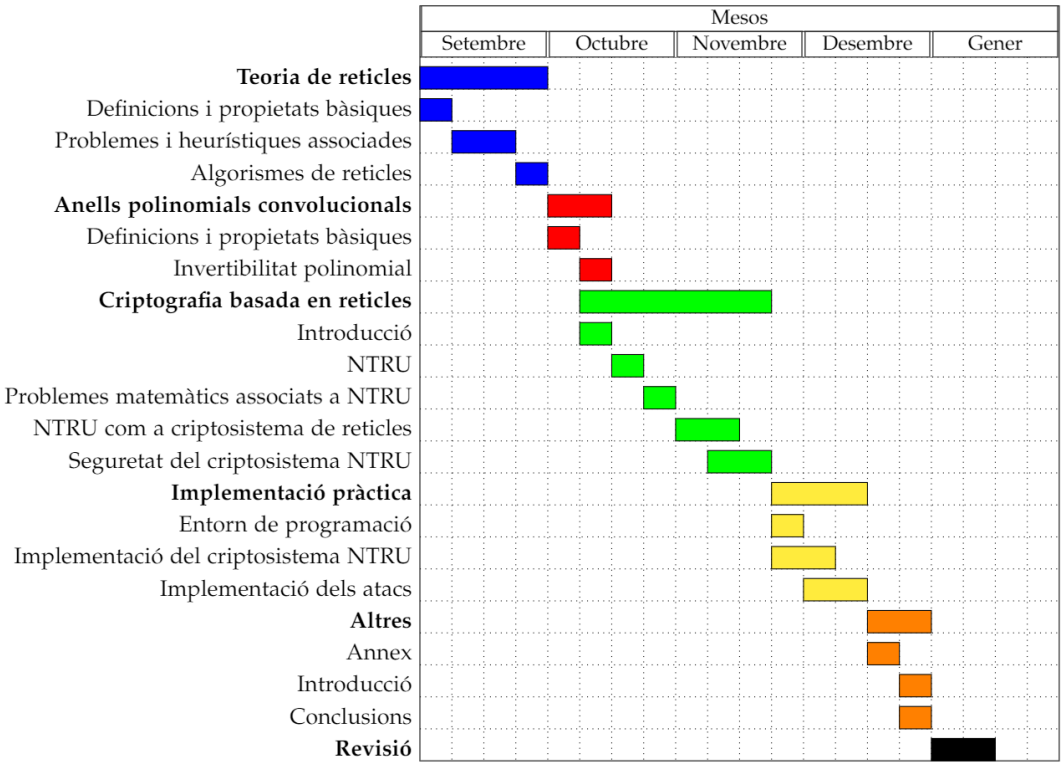

Figura D.1: Diagrama de Gant que mostra el desenvolupament final del projecte.