UNIVERSITÀ DEGLI STUDI DI ROMA TRE
FACOLTÀ DI SCIENZE M.F.N.
Corso di Laurea Magistrale in Matematica

Sintesi della tesi di Laurea

CRITTOGRAFIA A CHIAVE PUBBLICA BASATA SU RETICOLI: IL SISTEMA NTRU.

Relatore:
Prof.
FRANCESCO PAPPALARDI

Candidato:
RICCARDO CARBONE

Anno Accademico 2016-2017
Dipartimento di Matematica

Sintesi

NTRU è un crittosistema presentato dai tre matematici J. Hoffstein, J. Pipher e J.H. Silverman nell'articolo "NTRU: A Ring-Based Public Key Cryptosystem", nel 1996. Non risultano spiegazioni ufficiali da parte degli autori riguardo il significato dell'acronimo NTRU. Secondo alcune interpretazioni, sembra che sia un'abbreviazione di ***N**-th **TRU**ncated Polinomial Ring*, cioè *Anello dei polinomi troncati di grado n*, in notazione matematica $R[x]/(x^N - 1)$, ossia l'anello sottostante in cui lavora il sistema. Altre interpretazioni sono ***N**umber **T**heorist '**R** **U**s* e ***N**umber **T**heory **R**esearch **U**nited*.
Esso fa parte della categoria dei crittosistemi a chiave pubblica ed è stato profondamente studiato dalla crittografia teorica, dopo la sua pubblicazione, per via dei suoi innegabili vantaggi: il basso costo computazionale delle operazioni di cifratura e decifratura dei messaggi; la lunghezza della chiave, decisamente inferiore a quella dei crittosistemi più comuni; l'assenza in letteratura di attacchi quantistici, che lo fanno appartenere a tutti gli effetti ai cosiddetti Post Quantum Cryptosystems, l'insieme degli algoritmi resistenti agli attacchi di un computer quantistico.
Attualmente RSA è l'algoritmo a chiave pubblica più conosciuto e usato nell'ambito della sicurezza informatica; tuttavia l'algoritmo di Shor, proposto nel 1994, risolve il problema della fattorizzazione dei numeri interi in tempo polinomiale, se implementato in un computer quantistico. Rappresenta quindi una minaccia piuttosto vicina per la sicurezza di RSA e di ogni altro algoritmo basato sulla fattorizzazione dei numeri interi.
Dopo aver fornito una panoramica generale della crittografia, dagli albori fino agli anni '60, abbiamo esposto gli aspetti matematici di NTRU e valutato poi la vulnerabilità. Descritti i principali attacchi conosciuti, il primo dei quali l'algoritmo LLL, abbiamo visto come sia possibile renderlo inviolabile con un'adeguata scelta dei parametri, facendo di esso uno dei migliori candidati a sostituire RSA.

Nel Capitolo uno, dopo aver presentato la crittografia classica attraverso una breve introduzione storica, abbiamo definito il concetto di crittografia a chiave pubblica. Ne abbiamo illustrato i concetti principali, e abbiamo mostrato i più comuni esempi, dando maggiore rilevanza all'algoritmo RSA.

Nel Capitolo due abbiamo introdotto la teoria dei reticoli.
Nel primo paragrafo abbiamo descritto un semplice crittosistema, la cui struttura matematica è basata su un reticolo di dimensione 2. Poi abbiamo richiamato alcuni concetti fondamentali della geometria e dell'algebra lineare. Infine abbiamo esposto e dimostrato l'algoritmo di Gram-Schmidt, necessario per la dimostrazione e la comprensione dell'algoritmo LLL, descritto nel Capitolo quattro. Abbiamo successivamente descritto le nozioni principali che ci sono servite per vedere il crittosistema NTRU come un reticolo e evidenziato come la sua sicurezza sia fondata sulla ricerca del vettore minimo su un reticolo, analogamente alla fattorizzazione degli interi in RSA. Queste nozioni ci sono servite inoltre per descrivere l'algoritmo LLL (Lenstra-Lenstra-Lovasz) che, concepito inizialmente per fattorizzare polinomi a coefficienti razionali in tempo polinomiale, trova applicazioni in teoria dei numeri e in crittoanalisi, e rappresenta attualmente, sotto certe condizioni, il principale attacco a NTRU.

Abbiamo quindi dato le seguenti definizioni:

Definizione 0.1. Dati i vettori $v_1, \cdots, v_n \in \mathbb{R}^n$ linearmente indipendenti, si dice **reticolo** generato da $v_1, \cdots, v_n$ l'insieme dei vettori della forma $a_1 v_1 + \cdots + a_n v_n$, con $a_1, \cdots, a_n \in \mathbb{Z}$. I vettori costituiscono una **base** del reticolo.

Definizione 0.2. Sia $v = (x_1, \cdots, x_n) \in \mathbb{R}^n$; si definisce lunghezza di v la quantità $\|v\| = \sqrt{x_1^2 + \cdots + x_n^2}$. Si definisce **determinante** di un reticolo di base $v_1, \cdots, v_n$ la quantità $D = |\det(v_1, \cdots, v_n)|$.

Definizione 0.3. Sia L un reticolo di dimensione n, e sia $\mathbf{v}_1, \cdots, \mathbf{v}_n$ una base di L. Il *dominio fondamentale* di L è l'insieme

$$F(\mathbf{v}_1, \cdots, \mathbf{v}_n) = \{\ t_1\mathbf{v}_1 + \cdots + t_n\mathbf{v}_n : 0 \leq t_i < 1\} \ .$$

Abbiamo successivamente dimostrato i seguenti teoremi:

Proposizione 0.0.1. *(Disuguaglianza di Hadamard) Sia L un reticolo, sia $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n$ una base di L e sia F un dominio fondamentale di L. Allora*

$$det(L) = Vol(F) \leq \|\boldsymbol{v}_1\| \cdots \|\boldsymbol{v}_n\|.$$

Proposizione 0.0.2. *Sia $L \subset \mathbb{R}^n$ un reticolo di dimensione n, $\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n$ una base di L e sia F= $F(\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n)$ il dominio fondamentale associato. Scriviamo le coordinate dell'i-mo vettore della base come*

$$\boldsymbol{v}_i = (r_{i1}, \cdots, r_{in})$$

e usiamo le coordinate dei $\boldsymbol{v}_i$ come le righe di una matrice,

$$F = F(\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n) = \begin{pmatrix} r_{11} & \cdots & r_{1n} \\ \vdots & \ddots & \vdots \\ r_{n1} & \cdots & r_{nn} \end{pmatrix}$$

Allora il volume di F è dato dalla formula

$$Vol(F(\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n)) = |det(F(\boldsymbol{v}_1, \cdots, \boldsymbol{v}_n)|.$$

Teorema 0.0.3. *(Teorema di Minkowski)*

Sia $L \subset \mathbb{R}^n$ un reticolo di dimensione n e sia $S \subset \mathbb{R}^n$ un insieme convesso e simmetrico tale che

$$vol(S) > 2^n det(L).$$

Allora S contiene un vettore non nullo del reticolo.
Se S è chiuso, allora è sufficiente prendere $vol(S) \geq 2^n det(L)$.

Teorema 0.0.4. *(Teorema di Hermite) Sia L un reticolo di dimensione n. Allora esiste $\boldsymbol{v} \in L$ tale che*

$$\|\boldsymbol{v}\| \leq \sqrt{n}\ det(L)^{1/n}.$$

I reticoli sono stati ampiamente studiati già a partire dal diciottesimo secolo da Gauss e Lagrange. Come già accennato, si è scoperto negli ultimi venti anni una loro applicazione in crittoanalisi e in crittografia. Il problema computazionale sotteso ai sistemi crittografici basato sui reticoli è il cosiddetto SVP (Shortest Vector Problem), cioè la ricerca di un vettore di lunghezza minima nel reticolo. Quando la dimensione n è piccola si può ricorrere a un algoritmo di riduzione attraverso cui non è difficile trovare un vettore di lunghezza minima. Abbiamo esposto e dimostrato un algoritmo dovuto a Gauss che fornisce un vettore di lunghezza minima nel caso $n = 2$; poi siamo passati al caso multidimensionale, richiamando il processo di ortogonalizzazione di Gram-Schmidt, la cui idea è necessaria per la comprensione dell'algoritmo LLL.

Nel Capitolo tre abbiamo descritto il crittosistema NTRU. In questo caso il problema computazionalmente difficile da affrontare è dato dalla difficoltà di fattorizzare certi polinomi in un anello di polinomi troncati, analogamente alla fattorizzazione degli interi nel caso di RSA.
Gli elementi di partenza del crittosistema NTRU sono quindi polinomi appartenenti all'anello $(R+,*)$, dove:

$$\mathrm{R}=\mathbb{Z}[\mathrm{x}]/(x^N\text{-}1)$$

e $*$ indica la convoluzione, ossia: dati F,G $\in R$, con F=$\sum_{i=0}^{N-1} F_i x^i$, G=$\sum_{i=0}^{N-1} G_i x^i$, allora:

$$\mathrm{F}*\mathrm{G}:=\sum_{i+j\equiv k(\bmod N)} F_i G_j \quad .$$

I coefficienti dei polinomi sono ridotti modulo $q \in \mathbb{Z}$, e si indica con:

$$R_{q,N} = (\mathbb{Z}/q\mathbb{Z})[\mathrm{x}]/((x^{(N)}\text{-}1)$$

Vengono in seguito scelti due moduli $p, q \in \mathbb{Z}$, tali che $q > p$, e g.c.d $(p, q) = 1$. Poi vengono scelti quattro sottoinsiemi di R: $\mathcal{L}_m$ (messaggi in chiaro costituiti dai polinomi i cui coefficienti sono compresi tra $-p/2$ e $p/2$), $\mathcal{L}_f$ (chiavi private), $\mathcal{L}_g$, $\mathcal{L}_r$. Si prende poi l'insieme $\mathcal{L}(d_1, d_2)$, costituito dai polinomi che hanno d_1 coefficienti uguali a 1, d_2 coefficienti uguali a -1 e i restanti coefficienti nulli. Si scelgono $d_f, d_g, d_r \in \mathbb{Z}$ e si pone: $\mathcal{L}_f = \mathcal{L}(d_f, d_f - 1)$, $\mathcal{L}_g = \mathcal{L}(d_g, d_g)$, $\mathcal{L}_r = \mathcal{L}(d_r, d_r)$. Una volta scelti tutti questi parametri, il crittosistema NTRU viene rappresentanto dalla sestupla (N, q, p, d_f, d_g, d_r). Vedremo nei capitoli successivi i valori tipici di questi parametri attualmente in uso in ambito crittografico e commerciale e le loro scelte ottimali.
Una volta definito il crittosistema, descriveremo i processi di cifratura e decifratura, che riportiamo in sintesi:

1. Generazione della chiave pubblica:
   (a) Bernardo sceglie $f \in \mathcal{L}_f, g \in \mathcal{L}_g$.
   (b) Calcola f_q di f in $R_{q,N}$ e l'inverso di f_p in $R_{p,N}$.
   (c) Genera la chiave pubblica: $h = pq * g \in R_{q,N}$.
2. Cifratura
   (a) Alice traduce il messaggio da inviare a Bernardo in un polinomio $m \in \mathcal{L}_m$

(b) Alice sceglie $r \in \mathcal{L}_r$.

(c) Calcola $e = r * h + m \in R_{q,N}$.

(d) Invia e a Bernardo.

3. Decifratura

(a) Bernardo riceve e.

(b) Calcola $a = f * e \in R_{q,N}$; i coefficienti di a sono rappresentati come interi in $[-q/2, q/2]$.

(c) Riduce i coefficienti di a modulo p, ottenendo un polinomio b.

(d) Calcola $c = f_p * b \in R_{p,N}$; i coefficienti di c sono rappresentati come interi in $[-p/2, p/2]$.

(e) c rappresenta il messaggio m di Alice.

Abbiamo analizzato in dettaglio i vari parametri del sistema e mostrato perché c rappresenta il messaggio originale inviato da Alice.

Abbiamo visto infine la correlazione apparentemente lontana tra il crittosistema NTRU, definito attraverso polinomi, e la Teoria dei reticoli, mostrando quindi perché la sicurezza di NTRU si fonda su SVP. Di seguito una breve sintesi:

dato $h = h_{N-1}x^{N-1} + \cdots + h_0$, consideriamo la seguente matrice:

$$\mathrm{H}=\begin{pmatrix} h_0 & h_1 \cdots & h_{N-1} \\ h_{N-1} & h_0 \cdots & h_{N-2} \\ \vdots & \ddots & \vdots \\ h_1 & h_2 \cdots & h_0 \end{pmatrix}.$$

Consideriamo i due polinomi segreti di NTRU, $f = f_{N-1}x^{N-1} + ... + f_0$ e $g = g_{N-1}x^{N-1} + ...g_0$. Rappresentandoli come vettori riga, otteniamo i due vettori $\overline{f} = (f_0, \cdots, f_{N-1})$ e $\overline{g} = (g_0, \cdots, g_{N-1})$, e si ha: $\overline{f}H = \overline{g} \pmod q$. Consideriamo ora la matrice:

$$\mathrm{M}=\begin{pmatrix} I & H \\ 0 & qI \end{pmatrix},$$

dove I rappresenta la matrice identica con N righe e N colonne. Adesso possiamo generare un reticolo L attraverso le righe di M.

Il Capitolo quattro è dedicato ai possibili attacchi al sistema NTRU.
Prima abbiamo visto l'algoritmo di Gauss, che rappresenta la via più efficace per trovare un vettore minimo in dimensione 2. In seguito abbiamo esposto l'algoritmo LLL, adatto per dimensioni superiori. Riportiamo entrambi di seguito:

RIDUZIONE GAUSSIANA DEL RETICOLO

- [1] Input: una base $\{\mathbf{v}_1, \mathbf{v}_2\}$
- [2] Loop: se $\|\mathbf{v}_1\| < \|\mathbf{v}_2\|$, scambia $\mathbf{v}_1$ e $\mathbf{v}_2$.
- [3] Calcola $m = \lfloor(\mathbf{v}_1 \cdot \mathbf{v}_2/\|\mathbf{v}_1\|^2)\rceil$.
- [4] Continua il loop.

ALGORITMO LLL

- [1] Input: una base $\{\mathbf{v}_1, \cdots, \mathbf{v}_n\}$
- [2] Poni $k = 2$
- [3] Poni $\mathbf{v}_1^* = \mathbf{v}_1$
- [4] Loop while $k \leq n$
- [5] Loop $j = 1, \cdots, k-1$
- [6] Poni $\mathbf{v}_k = \mathbf{v}_k - \lfloor\mu_{k,j}\rceil\mathbf{v}_j^*$ [Size condition]
- [7] End j loop
- [8] If $\|\mathbf{v}_k^*\|^2 \geq (\frac{3}{4} - \mu_{k,k-1}^2)\|\mathbf{v}_{k-1}^*\|^2$ [Lovasz Condition]
- [9] Poni $k = k + 1$
- [10] Else
- [11] Scambia $\mathbf{v}_k - 1$ con $\mathbf{v}_k$ [Swap test]
- [12] Poni $k =$max$(k-1, 2)$
- [13] End if

- [14]End k loop
- [15]Return: base LLL-ridotta $\{\mathbf{v}_1, \cdots, \mathbf{v}_n\}$

Infine abbiamo mostrato come sia possibile vanificare l'attacco attraverso una modifica dei parametri iniziali.

Nel Capitolo cinque abbiamo illustrato un esempio numerico: a partire da un crittosistema NTRU, abbiamo generato il relativo reticolo e mostrato il funzionamento dell'algoritmo LLL.

Abbiamo considerato un sistema NTRU con parametri pubblici

$$(N, p, q, d) = (7, 3, 41, 2).$$

Abbiamo che

$$41 = q > (6d + 1)p = 39,$$

quindi i parametri sono scelti bene. Alice sceglie

$$\mathbf{f}(x) = x^6 - x^4 + x^3 + x^2 - 1 \in T(3, 2)$$

$$\mathbf{g}(x) = x^6 + x^4 - x^2 - x \in T(2, 2).$$

Calcola gli inversi

$$\mathbf{F}_q(x) = \mathbf{f}(x)^{-1} \pmod{q} = 8x^6 + 26x^5 + 31x^4 + 21x^3 + 40x^2 + 2x + 37 \in R_q$$

e

$$\mathbf{F}_p(x) = \mathbf{f}(x)^{-1} \pmod{p} = x^6 + 2x^5 + x^3 + x^2 + x + 1 \in R_p.$$

Sceglie $(\mathbf{f}(x), \mathbf{F}_p(x))$ come chiave privata ed espone la chiave pubblica

$$\mathbf{h}(x) = \mathbf{F}_q * \mathbf{g}(x) = 20x^6 + 40x^5 + 2x^4 + 38x^3 + 8x^2 + 26x + 30 \in R_q.$$

Bernardo invia ad Alice il messaggio

$$\mathbf{m}(x) = -5 + x^3 + x^2 - x + 1$$

usando la chiave effimera

$$\mathbf{r}(x) = x^6 - x^5 + x - 1.$$

Bernardo calcola e manda ad Alice il testo cifrato

$$\mathbf{e}(x) \equiv p\mathbf{r}(x) * \mathbf{h}(x) + \mathbf{m}(x) \equiv 31x^6 + 19x^5 + 4x^4 + 2x^3 + 40x^2 + 3x + 25 \pmod{q}.$$

Ora Alice decifra il messaggio, calcolando prima

$$\mathbf{f}(x) * \mathbf{e}(x) \equiv x^6 + 10x^5 + 33x^4 + 40x^3 + 40x^2 + x + 40 \pmod{q}$$

e poi sollevando modulo q, ottenendo

$$\mathbf{a}(x) = x^6 + 10x^5 - 8x^4 - x^3 - x^2 + x - 1 \in R.$$

Infine riduce $\mathbf{a}(x)$ modulo q e calcola

$$\mathbf{F}_p(x) * \mathbf{a}(x) \equiv 2x^5 + x^3 + x^2 + 2x + 1 \pmod{p}.$$

Sollevando l'ultima uguaglianza modulo p, ritroviamo il testo in chiaro $\mathbf{m}(x) = -x^5 + x^3 + x^2 - x + 1$.

Il reticolo associato è generato dalle righe della matrice

$$M_{\mathbf{h}}^{NTRU} = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 30 & 26 & 8 & 38 & 2 & 40 & 20 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 20 & 30 & 26 & 8 & 38 & 2 & 40 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 40 & 20 & 30 & 26 & 8 & 38 & 2 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 40 & 20 & 30 & 26 & 8 & 38 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 38 & 2 & 40 & 20 & 30 & 26 & 8 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 8 & 38 & 2 & 40 & 20 & 30 & 26 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 26 & 8 & 38 & 2 & 40 & 20 & 30 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 41 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 41 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 41 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 41 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 41 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 41 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 41
\end{pmatrix}.$$

Eva applica la riduzione LLL a $M_{\mathbf{h}}^{NTRU}$. L'algoritmo esegue 96 cambi e restituisce la matrice LLL-ridotta

$$M_{rid}^{NTRU} = \begin{pmatrix} 1 & 0 & -1 & 1 & 0 & -1 & -1 & -1 & 0 & -1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & -1 & 0 & 1 & -1 & -1 & -1 & 0 & 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & -1 & -1 & 1 & 0 & -1 & 0 & 1 & 1 & 0 & -1 & 0 \\ -1 & -1 & 1 & 0 & -1 & 1 & 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 \\ -1 & 1 & 0 & -1 & 1 & 0 & -1 & 0 & -1 & 0 & -1 & 0 & 1 & 1 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & -1 & 1 & -1 & -1 & 0 & 0 & 2 & 0 & 0 \\ -8 & -1 & 0 & 9 & 0 & -1 & 0 & -4 & 2 & 6 & 0 & -4 & 7 & -7 \\ 8 & 1 & 0 & 0 & -8 & -1 & 2 & 0 & -5 & 8 & -7 & -3 & 1 & 6 \\ 0 & -9 & -2 & 1 & 9 & -1 & 0 & -6 & -3 & 2 & 5 & 0 & -5 & 7 \\ 0 & 8 & 0 & -9 & -1 & -8 & 8 & 2 & 7 & -11 & 3 & -5 & 2 & 2 \\ 1 & 0 & 0 & 9 & 2 & -1 & -9 & 5 & -7 & 6 & 3 & -2 & -5 & 0 \\ -2 & 1 & 9 & -1 & 0 & 0 & -9 & 2 & 5 & 0 & -5 & 7 & -6 & -3 \\ 3 & 2 & 3 & 3 & -6 & 2 & -6 & 11 & 6 & 8 & -9 & 5 & 2 & \end{pmatrix}.$$

Il vettore minimo nella base ridotta è la prima riga della matrice ridotta, cioè

$$(1, 0, -1, 1, 0, -1, -1, -1, 0, -1, 0, 1, 1, 0).$$

Spezzando questo vettore in due pezzi, otteniamo i due polinomi

$$\mathbf{f}'(x) = 1 - x^2 + x^3 - x^5 - x^6 \qquad \mathbf{g}'(x) = -1 - x^2 + x^4 + x^5.$$

Notiamo che $\mathbf{f}'(x)$ e $\mathbf{g}'(x)$ non sono gli stessi polinomi $\mathbf{f}(x)$ e $\mathbf{g}(x)$ della chiave privata di Alice. Essi, però, sono loro semplici rotazioni; infatti

$$\mathbf{f}'(x) = -x^3 * \mathbf{f}(x) \qquad \text{e} \qquad \mathbf{g}'(x) = -x^3 * \mathbf{g}(x).$$

Allora Eva può usare $\mathbf{f}'(x)$ e $\mathbf{g}'(x)$ per decifrare il messaggio.

Indice

Bibliografia

- [1] J. Hoffstein, J. Pipher, J. H. Silverman, *An introduction to mathematical cryptography,* Springer Science, New York, 2008.
- [2] J. Hoffstein, J. Pipher, J. H. Silverman, *Ntru: A ring based public key cryptosystem,* Proceedings of the third International Symposium on Algorithmic Number theory, 267-268, 1998.
- [3] W. Diffie, M. E. Hellman, *New direction in cryptography,* IEEE Transaction on Information Theory, 22,1999.
- [4] D. Khan, *The code-breakers,* Mcmillan, 1973.
- [5] National Boureau of Standards, *Data encryption standard,* FIPS, 1977.
- [6] R.A. Mollin, *RSA and public-key cryptography,* CRC Press, 2003.
- [7] W. M. Boldoni, C. Ciriberto, G. M. Piacentini Cattaneo, *Aritmetica, crittografia e codici,* Springer, 2006.
- [8] P. W. Shor, *Algorithms for quantum computation: discrete logarithms and factoring,* IEEE Comp. Soc. Press., 1994.
- [9] J. Hoffstein, J. Pipher, J. H. Silverman, *NTRU: A new speed public-key cryptosystem,*Lecture notes in Computer Science, 1995.
- [10] E. Sernesi, *Geometria 1,* Bollati Boringhieri, 2000.
- [11] M. Kaib, C. P. Schnorr, *The generalized Gauss reduction algorithm,* Journal of algorithms, 1996.
- [12] O. Regev, *Lattice-based cryptography,*, Lecture notes in Computer Science, 2006.

[13] A. K. Lenstra, J. W. Lenstra, L. Lovasz, *Factoring polinomials with rationals coefficients,* Math. Ann., 1982.

[14] D. Coppersmith, A. Shamir, *Lattice attacks on NTRU,* Lecture notes in Computer Science, 1997.