Universidade Estadual Paulista "Júlio de Mesquita Filho"
Instituto de Geociências e Ciências Exatas
Campus de Rio Claro

Um Estudo sobre Criptografia

Carlos Celestino Lima Souza

Dissertação apresentada ao Programa de Pós-Graduação – Mestrado Profissional em Matemática Universitária do Departamento de Matemática como requisito parcial para a obtenção do grau de Mestre

Orientadora
Profa. Dra. Carina Alves

2013

Agradecimentos

Para que esta dissertação fosse escrita, foram necessários meses de dedicação e estudo, e por isso quero agradecer primeiramente a Deus, que meu deu forças para continuar diante de tantas dificuldades apresentadas.

Agradeço aos funcionários do Programa de Pós-Graduação da UNESP, que sempre foram muito atenciosos a todas as dúvidas.

Agradeço aos professores, pela partilha do conhecimento, pela paciência e pelos ensinamentos para a vida.

Aos meus amigos, pelas horas de risos, estudos e por todo o apoio.

Quero agradecer minha à orientadora, Profª Drª Carina Alves, pela sua disponibilidade, atenção, amizade, por toda a sua ajuda e colaboração em todas as etapas da dissertação.

A minha família, que apesar das dificuldades que encontrei, sempre esteve ao meu lado me apoiando para que eu pudesse concluir essa importante etapa da minha vida.

Enfim, agradeço a todas as pessoas que contribuíram para esta realização.

Quem nunca errou nunca experimentou nada novo.

Albert Einstein

Resumo

Neste trabalho, apresentamos sistemas criptográficos clássicos, como o criptossistema Diffie-Hellman, o criptossistema RSA e o criptossistema de ElGamal. Estudamos alguns aspectos da criptografia quântica e alguns sistemas de criptografia pós-quântica, como o criptossistema Ajtai-Dwork, o criptossistema NTRU, o criptossistema de McEliece e o criptossistema de Niederreiter. Discutimos a segurança dos métodos de criptografia e possíveis soluções apresentadas para garantir a troca de informações confidencias mesmo com o avanço da computação quântica.

Palavras-chave: Criptografia, Teoria dos Números, Criptografia Quântica, Criptografia Pós-Quântica.

Abstract

In this work, we show classic cryptography systems, as the Diffie-Hellman cryptosystem, the RSA cryptosystem and the ElGamal cryptosystem. We studied some aspects of quantum cryptography and some post-quantum cryptography systems, as the Ajtai-Dwork cryptosystem, the NTRU cryptosystem, the McEliece cryptosystem and the Niederreiter cryptosystem. We discussed the classic cryptography methods security and possible solutions that are introduced to ensure the confidential information exchange even with the quantum computing advancement.

Keywords: Cryptography, Number Theory, Quantum Cryptography, Post-Quantum Cryptography.

Sumário

1 Introdução

Este trabalho visa o estudo de uma das técnicas utilizadas na história da Teoria dos Códigos: a criptografia e sua relação com a matemática. Além disso, estudamos métodos atuais de segurança e os possíveis métodos de segurança que teremos no futuro para troca de informações confidenciais.

O estudo da codificação de mensagens é antigo, tendo-se registros de mensagens codificadas em 1900 *a.C.* no Egito. Tais mensagens sempre foram muito importantes na história, pois sempre houve a necessidade de evitar que informações confidenciais caíssem nas mãos de inimigos ou se tornassem públicas.

Atualmente, a codificação de mensagens, principalmente por meio da criptografia, torna-se essencial no meio eletrônico, por exemplo, no envio de e-mails, transações bancárias, conversas online privadas, entre outros. Pensando dessa forma, é que surgiram os primeiros métodos de ocultação, ou codificação de mensagens que são, respectivamente, a *esteganografia* e a *criptografia*.

A esteganografia vem do grego, onde *steganos* significa coberto, e *graphien* significa escrever. O historiador grego Heródoto (*484 a.C. - 425 a.C.*), em sua publicação "As histórias", relata em um dos contos um exemplo de estenografia onde Histaeu, Rei de Mileto, raspou a cabeça de seu escravo mais confiável e tatuou em seu couro cabeludo um plano de revolta contra a dominação persa. Após o cabelo do escravo crescer, o rei enviou-o ao seu amigo Aristágoras, que recebeu a mensagem oculta. Métodos de ocultação de mensagem só eram eficazes se o mensageiro não era "descoberto".

Dessa forma, juntamente com o desenvolvimento da esteganografia, houve o desenvolvimento da criptografia, que vem do grego *"kriptos"*, que significa oculto. Criptografia é a ciência de escrever mensagens em *cifras* ou *códigos*, permitindo que apenas o destinatário desejado possa decifrar e compreender a mensagem. Portanto, diferentemente da esteganografia, a criptografia tem por objetivo ocultar o significado da mensagem, e não a mensagem propriamente dita.

Com o surgimento da criptografia, nasce então a criptoanálise que tem por objetivo decriptar a mensagem criptografada.

A história da criptografia, como já vimos é muito antiga. Além do registro do Egito, há também registros na Mesopotâmia, onde as peças de pedra com símbolos, conhecidas como *"intaglios"* funcionavam como certificados rudimentares.

Por volta de 600 a 500 a.C., os hebreus faziam uso de cifras que consistiam na substituição simples de uma letra por outra (substituição monoalfabética).

Outros métodos foram sendo utilizados ao longo dos anos. Mas sempre que surge um novo método de criptografia, a criptoanálise aperfeiçoava-se para decriptar o método.

No século IX, o filósofo árabe Al-Kindi escreveu um trabalho sobre a utilização da *"análise da frequência"* para decifrar códigos. Esse método consiste em analisar uma mensagem codificada observando a repetição dos símbolos, e aos que mais aparecem, atribuir as letras mais utilizadas no alfabeto de determinada língua, no nosso caso, as vogais "a", "e", "i", "o" e "u".

Com a renascença, a criptografia, juntamente com a criptoanálise, ganham força, principalmente incentivados pelos Governos. É nessa época, que destacamos nesse trabalho outro importante sistema de criptografia, que ficou conhecido como a *Cifra de Vigenère*. Essa cifra, foi criada, primeiramente por Leon Battista Alberti (1404 - 1472) que propôs o uso de dois alfabetos cifrados para codificar uma mensagem alternando-os para a codificação, o que até o momento não existia, pois as mensagens eram codificadas apenas com um alfabeto cifrado. Apesar dessa ideia de Alberti, este não conseguiu transformá-la em um sistema completo de cifragem, e coube a um grupo de intelectuais aperfeiçoarem a ideia original, Johannes Trithemius (1462 - 1516), depois o italiano Giovanni Porta (1541 - 1615), e por fim o francês Blaise de Vigenère (1523 - 1596). Vigenère ao tomar conhecimento de todos os trabalhos, mesclou-os, formando uma cifra que consiste em até 26 alfabetos para criar a mensagem cifrada.

Com o passar dos anos, novas cifras foram surgindo. Gottfried Wilhelm von Leibniz (1646-1716), apresentou um estudo inovador que criou o cálculo diferencial e integral, a máquina de calcular e descreveu minuciosamente o sistema binário.

Em 1795, Thomas Jefferson apresenta o "cilindro de Jefferson", que realiza a criptografia polialfabética de forma rápida e eficiente.

Cilindro de Jefferson

Surgiram, ainda nesse período, o código Braille e o código Morse.

Já em 1854, Charles Barbage quebra a cifra de Vigenère e apresenta a máquina de diferenças e a máquina analítica, que são as precursoras dos computadores.

Os primeiros protótipos de máquinas de criptografar, surgem a partir de 1924, com a máquina SIGABA (M-134-C), inventada por William F. Friedman, que ficou conhecido pelos americanos como o “pai da criptoanálise”.

Já no período da Segunda Guerra Mundial, os nazistas aperfeiçoaram uma máquina chamada Enigma, que era composta por um teclado com 26 letras e 26 lâmpadas.

Mas é por volta de 1970 que os computadores começam a ganhar mais força, e com isso surge um algoritmo conhecido por *DES (Data Encryption Standard)*. Esse algoritmo é um algoritmo simétrico, que consiste em criptografar usando um algoritmo que codifica a mensagem, e o mesmo algoritmo a decodifica. Muitas vezes, a chave tinha que ser transmitida ao destinatário para que este pudesse decriptar a mensagem, o que por muitas vezes era um problema. O DES foi considerado passível de ataque através da força bruta, e acabou evoluindo para o 3DES, que consistia em aplicar o DES três vezes.

Em 1977, Ronald L. Rivest, Adi Shamir e Leonard M. Adleman apresentam o algoritmo assimétrico RSA. Esse sistema é a base dos sistemas de chave pública, que consiste em uma chave que todos tem acesso, e uma chave privada, que apenas o seu dono tem acesso, sendo a chave similar a uma "senha" que deve-se conhecer para decriptar a mensagem. O algoritmo baseia-se em funções matemáticas, trabalhando com primos grandes. Veremos, com detalhes nesse trabalho a criptografia RSA e a criptografia ElGamal, que são dois tipos de criptografia que utilizam algoritmos assimétricos.

Em 1990 surge a criptografia quântica, que usa fótons únicos para transmitir um fluxo de bits chave para uma posterior cifragem. Esse tipo de criptografia também será discutida nesse trabalho.

Atualmente, existem muitos métodos para quebrar a segurança das principais criptografias, tais como, a criptografia baseada em reticulados, os criptossistemas de McEliece e Niederreiter, que são conhecidos como criptografia pós-quântica. Estes últimos são baseados na síndrome dos códigos corretores de erros. Todos esses sistemas serão discutidos ao longo desse trabalho.

A fim de proporcionar uma visão geral do nosso trabalho, apresentamos uma breve descrição dos assuntos que iremos tratar em cada um dos capítulos.

No Capítulo 2, abordamos de modo detalhado todos os pré-requisitos de teoria do números que serviram de ferramenta para o desenvolvimento do Capítulo 3. Procuramos dessa forma, destacar a aplicabilidade de teoria dos números em criptografia.

No Capítulo 3, apresentamos o algoritmo de criptografia Diffie-Hellman que foi o primeiro algoritmo de chave pública a ser inventado. Em seguida, apresentamos o algoritmo RSA que usa o sistema de chaves assimétricas e fundamenta-se na teoria dos números. Ele foi uma das grandes inovações em criptografia de chave pública. Apresentamos ainda, o algoritmo de criptografia ElGamal que também faz o uso de chaves assimétricas.

No Capítulo 4, dando continuidade ao estudo da criptografia, demos uma descrição sobre a criptografia quântica e introduzimos alguns aspectos da criptografia pós-quântica. No capítulo 5, apresentamos a conclusão final.

Devido ao grau de dificuldade e pré-requisitos necessários para um estudo mais

aprofundado destes tipos de criptografias, e ao mesmo tempo, para irmos de encontro com os ideais de formação de um matemático do Programa de Pós-Graduação em Matemática Universitária da UNESP - Rio Claro, limitamo-nos em abordar somente os aspectos teóricos destes tipos de criptografias e não exploramos os aspectos computacionais.

Além disso, o trabalho visa mostrar aplicações práticas de teoria dos números, teoria dos reticulados e códigos corretores de erros possibilitando que pesquisas futuras se beneficiem das informações contidas aqui.

2 Teoria dos números

Os principais problemas matemáticos relacionados a criptografia RSA são: como achar números primos e como fatorar um número. A área da matemática a que estes problemas pertencem é conhecida como Teoria do Números.

A Teoria dos Números é herdeira da aritmética dos gregos e a palavra aritmética é usada hoje em dia para descrever o que os gregos chamavam de logística.

Alguns dos problemas da Teoria de Números abordados pelos gregos antigos podem ser encontrados no livro os *Elementos* escrito pelo matemático Euclides, que viveu em Alexandria por volta de 300 a.C.. Neste capítulo utilizamos as referências [3], [4], [6], [12], [13], [5], [8], [9],[10], [16], e [11].

2.1 Algoritmos fundamentais

Nesta seção, apresentamos os principais algoritmos usados para estudar o sistema de criptografia RSA.

2.1.1 Algoritmo da divisão

Definição 2.1.1. Sejam a e b inteiros. Dizemos que a divide b, denotado por $a|b$, se existir um inteiro c tal que $b = ac$.

Teorema 2.1.1. Dados dois inteiros a e b, existe um único par de inteiros q e r tais que

$$a = bq + r, \text{ com } 0 \leq r < |b| \ (r = 0 \Leftrightarrow b|a).$$

Neste caso, q recebe o nome de *quociente* e r recebe o nome de *o resto não negativo* da divisão de a por b.

Demonstração. Vamos mostrar a existência de q e r. Dados $a, b \in \mathbb{Z}$, consideremos o conjunto

$$S = \{a - bx | x \in \mathbb{Z}, a - bx \geq 0\}.$$

É fácil ver que $S \subseteq \mathbb{N}$. Para $x = -|a|$, tem-se $a - bx = a - b(-|a|) = a + b|a| \geq a + |a| \geq 0$, pois $b \geq 1$. Assim $S \neq \emptyset$. Pelo princípio da boa ordem, segue que $r \in S$ é mínimo, ou seja, $r \leq y, \forall\, y \in S$. Como $r \in S$, segue que $x = q$, com $q \in \mathbb{Z}$, onde $r = a - bq$. Desta forma, $a = bq + r$. Mostremos, agora, que $0 \leq r < |b|$. Como $r \in S$, segue que $r \geq 0$. Se $r \geq |b|$, então $\exists\ s \geq 0$ tal que $r = |b| + s$, com $0 \leq s < r$. Mas isto contradiz o fato de r ser o menor elemento, já que

$$s = -|b| + r = -|b| + a - bq = a - b(q \pm 1) \in S.$$

Desta forma, não existe $r \geq b$ nessas condições. Portanto $r < b$. Verifiquemos agora, a unicidade de r e q. Suponhamos que existam r, q e r', q' inteiros tais que

$$a = bq + r = bq' + r', \text{ e } 0 \leq r, r' \leq |b|.$$

Assim, $r - r' = bq - bq' = b(q - q')$, e desta forma $|r - r'| = |b(q - q')| = b|q - q'|$. Dadas as desigualdades $0 \leq r' < b$ e $-b < -r \leq 0$, segue que $-b < r' - r < b$, ou seja, $|r' - r| < |b|$. Daí se $q \neq q'$, então

$$|b| \leq |b||q - q'| = |r' - r| < |b|,$$

o que é uma contradição pois $b|(r - r')$. Portanto, podemos concluir que $q = q'$ e $r = r'$. □

Exemplo 2.1.1. Sejam $n > m$ inteiros positivos. Se o resto da divisão de n por m é r, então o resto da divisão de $2^n - 1$ por $2^m - 1$ é $2^r - 1$. De fato, podemos escrever n como

$$n = mq + r,$$

sendo q um número inteiro positivo. Vamos mostrar que existe um inteiro q' tal que $0 \leq 2^r - 1 < 2^m - 1$ e $2^n - 1 = (2^m - 1)q' + 2^r - 1$. Note que

$$0 \leq r < m \Longrightarrow 2^0 \leq 2^r < 2^m \Longrightarrow 0 \leq 2^r - 1 < 2^m - 1.$$

Mostremos, agora, que q' é um número inteiro. Como

$$2^n - 1 = (2^m - 1)q' + 2^r - 1 \Longleftrightarrow q' = \frac{(2^n - 1) - (2^r - 1)}{2^m - 1} \Longleftrightarrow q' = \frac{2^n - 2^r}{2^m - 1},$$

segue que $2^n - 2^r = 2^r(2^{n-r} - 1) = 2^r(2^{mq} - 1)$. Mas $2^{mq} - 1 = (2^m - 1)(2^{mq-1} + \cdots + 1)$, e assim,

$$q' = \frac{2^r(2^m - 1)(2^{mq-1} + \cdots + 1)}{2^m - 1} = 2^r(2^{mq-1} + \cdots + 1).$$

Portanto, q' é um número inteiro, concluindo a nossa afirmação.

2.1.2 Algoritmo euclidiano

Antes de enunciar o algoritmo euclidiano, precisamos de uma definição e um lema, enunciados a seguir.

Definição 2.1.2. Denotamos por $mdc(a, b)$ com $a, b \in \mathbb{Z}$ o maior número inteiro que divide a e b. A esse número damos o nome de *máximo divisor comum*.

Lema 2.1.1. Sejam a e b números inteiros positivos. Se existirem q e r inteiros, tal que $a = bq + r$, então $mdc(a, b) = mdc(b, r)$.

Demonstração. Sejam $d_1 = mdc(a, b)$ e $d_2 = mdc(b, r)$. Mostremos que $d_1 \leq d_2$. Como d_1 divide a e b, segue que existem inteiros positivos u e v tais que $a = d_1u$ e $b = d_1v$. Logo,

$$a = bq + r \Rightarrow d_1u = d_1vq + r \Rightarrow r = d_1u - d_1vq \Rightarrow r = d_1(u - vq).$$

Portanto, d_1 divide r. Como d_1 também divide b, segue que d_1 é divisor comum de b e r, e assim, $d_1 \leq d_2$, pois d_2 é o maior divisor de b e r. De modo análogo, tem-se que $d_2 \leq d_1$, e assim segue o resultado. □

Teorema 2.1.2 (Algoritmo euclidiano). Sejam $r_0 = a$ e $r_1 = b$ inteiros não-negativos com $b \neq 0$. Se o algoritmo da divisão for aplicado sucessivamente para se obter

$$r_i = q_{i+1}r_{i+1} + r_{i+2}, \text{ com } 0 \leq r_{i+2} < r_{i+1},$$

para $i = 0, 1, 2, ..., n-1$ e $r_{n+1} = 0$, então $mdc(a, b) = r_n$, que é o último resto não-nulo.

Demonstração. Primeiramente, vamos dividir a por b. Assim, podemos escrever $a = bq_1 + r_2$, com q_1 e r_2 inteiros. Logo, $r_0 = qr_1 + r_2$. Em seguida, vamos dividir r_1 por r_2, e obtemos $r_1 = q_2r_2 + r_3$. Fazemos isso sucessivamente até $r_{n+1} = 0$. Com isto, obtemos a seguinte sequência de equações

$$\begin{aligned}
a &= q_1b + r_2, & 0 < r_2 < r_1,\\
r_1 &= q_2r_2 + r_3, & 0 < r_3 < r_2,\\
r_2 &= q_3r_3 + r_4, & 0 < r_4 < r_3,\\
&\vdots & \vdots\\
r_{n-2} &= q_{n-1}r_{n-1} + r_n, & 0 < r_n < r_{n-1},\\
r_{n-1} &= q_nr_n + 0.
\end{aligned}$$

Pelo lema anterior, segue que o máximo divisor comum de r_n e r_{n-1} é r_n. Aplicando o lema várias vezes, segue que

$$r_n = mdc(r_{n-1}, r_n) = mdc(r_{n-2}, r_{n-1}) = \ldots = mdc(r_1, r_2) = mdc(r_0, r_1) = mdc(a, b).$$

Portanto, $mdc(a, b) = r_n$, que é o último resto não-nulo da sequência escrita. □

Exemplo 2.1.2. Utilizando o algoritmo euclidiano, vamos determinar o $mdc(428, 934)$. Sejam $r_0 = 934$ e $r_1 = 428$. Aplicando o algoritmo da divisão sucessivas vezes, obtem-se

$$\begin{aligned} 934 &= 482 \cdot 2 + 78, \\ 428 &= 78 \cdot 5 + 38, \\ 78 &= 38 \cdot 2 + 2, \\ 38 &= 19 \cdot 2 + 0. \end{aligned}$$

Pelo algoritmo euclidiano, segue que $mdc(428, 934)$ é o último resto não nulo, ou seja, $mdc(428, 934) = 2$.

2.1.3 Algoritmo euclidiano estendido

O algoritmo euclidiano estendido consiste em encontrar α e β inteiros, tal que

$$\alpha a + \beta b = mdc(a, b), \text{ com } a, b \in \mathbb{Z}.$$

Vamos supor que após o cálculo do $mdc(a, b)$, obtemos a seguinte sequência de divisões:

$$\begin{array}{ccc} a = bq_1 + r_1 & \text{e} & r_1 = ax_1 + by_1, \\ b = r_1q_2 + r_2 & \text{e} & r_2 = ax_2 + by_2, \\ r_1 = r_2q_3 + r_3 & \text{e} & r_3 = ax_3 + by_3, \\ r_2 = r_3q_4 + r_4 & \text{e} & r_4 = ax_4 + by_4, \\ r_3 = r_4q_5 + r_5 & \text{e} & r_5 = ax_5 + by_5, \\ \vdots & \text{e} & \vdots \\ r_{n-3} = r_{n-2}qn - 1 + r_{n-1} & \text{e} & r_{n-1} = ax_{n-1} + by_{n-1}, \\ r_{n-2} = r_{n-1}q_n & \text{e} & r_n = 0. \end{array}$$

Os números $x_1, \ldots, x_{n-1}$ e $y_1, \ldots, y_{n-1}$ são inteiros a determinar. Seja a seguinte tabela com essas informações:

Resto	Quociente	x	y
b	$*$	x_{-1}	y_{-1}
a	$*$	x_0	y_0
r_1	q_1	x_1	y_1
r_2	q_2	x_2	y_2
$\vdots$	$\vdots$	$\vdots$	$\vdots$
r_{n-2}	q_{n-2}	x_{n-2}	y_{n-2}
r_{n-1}	q_{n-1}	x_{n-1}	y_{n-1}

Consideremos a $j-$ésima linha, e vamos dividir r_{j-2} por r_{j-1}, para encontrarmos r_j e q_j de forma que

$$r_{j-2} = r_{j-1}q_j + r_j \qquad \text{e} \qquad 0 \leq r_j < r_j - 1.$$

Assim,

$$r_j = r_{j-2} - r_{j-1}q_j.$$

Observando os valores de x_{j-2}, x_{j-1}, y_{j-2} e y_{j-1} nas linhas $j-1$ e $j-2$, podemos escrever

$$r_{j-2} = ax_{j-2} + by_{j-2} \qquad \text{e} \qquad r_{j-1} = ax_{j-1} + by_{j-1}.$$

Desta forma,

$$\begin{aligned} r_j &= (ax_{j-2} + by_{j-2}) - (ax_{j-1} + by_{j-1})q_j \\ &= a(x_{j-2} - q_jx_{j-1}) + b(y_{j-2} - q_jy_{j-1}). \end{aligned}$$

Portanto,

$$x_j = x_{j-2} - q_jx_{j-1} \text{ e } y_j = y_{j-2} - q_jy_{j-1}.$$

Para calcularmos os valores de x_j e y_j da tabela, basta sabermos os valores das duas linhas imediatamente acima, através de um processo recursivo. Para determinarmos x e y iniciais, de modo análogo ao feito acima, devemos ter

$$a = ax_{-1} + by_{-1} \text{ e } a = ax_0 + by_0.$$

Com isto, podemos afirmar que $x_{-1} = 1, y_{-1} = 0$ e $x_0 = 1, y_0 = 0$. Assim, damos início a recursão. Como esse algoritmo nos fornece o máximo divisor comum correspondente ao resto $n-1$, segue que

$$mdc(a,b) = r_{n-1} = ax_{n-1} + by_{n-1},$$

ou seja,

$$\alpha = x_{n-1} \text{ e } \beta = y_{n-1}.$$

Exemplo 2.1.3. Vamos determinar $mdc(345, 92)$ e, em seguida, determinar α e β inteiros que satisfazem:

$$345\alpha + 92\beta = mdc(345, 92).$$

Para isso, utilizamos o *algoritmo euclidiano estendido*. Primeiramente, montamos a tabela, completando-a.

Resto	Quociente	x	y
345	$*$	1	0
92	$*$	0	1
69	3	$1 - 0 \cdot 3 = 1$	$0 - 1 \cdot 3 = -3$
23	1	$0 - 1 \cdot 1 = -1$	$1 - 1 \cdot (-3) = 4$
0	3	$*$	$*$

Portanto, $mdc(345, 92) = 23$, $\alpha = -1$, $\beta = 4$ e

$$345 \cdot (-1) + 92 \cdot 4 = 23.$$

2.2 Aritmética modular

Nesta seção, veremos como escrever um inteiro como um produto de números primos. Achar a decomposição de um inteiro em primos, pode ser um processo muito lento e difícil.

2.2.1 Congruência

A seguir, veremos alguns resultados que serão utilizados em algumas criptografias.

Definição 2.2.1. Sejam a, b e n números inteiros com $n > 0$. Dizemos que a é côngruo a b módulo n, se tiverem o mesmo resto da divisão por n. Notação, $a \equiv b(mod\ n)$.

Definição 2.2.2. Sejam a, b e n inteiros tal que $a \equiv b\ (mod\ n)$. Dizemos que b é o resíduo de a módulo n.

Proposição 2.2.1. $a \equiv b(mod\ n)$ se, e somente se, $n|(a-b)$.

Demonstração. Se $a \equiv b(mod\ n)$, então a e b possuem o mesmo resto na divisão por n. Vamos supor que $a = r + qn$ e $b = r + pn$, sendo que $p, q \in \mathbb{Z}$. Desta forma, $a-b = (r+qn)-(r+pn) = qn-pn = n(q-p)$. Portanto, $n|(a-b)$. Reciprocamente, suponhamos que $a = qn + r_1$ e $b = pn + r_2$, sendo q e p inteiros, $0 \leq r_1 < n$ e $0 \leq r_2 < n$. Mostraremos que $r_1 = r_2$. Se $r_2 < r_1$, então $a-b = (qn+r_1)-(pn+r_2) = (qn-pn)+(r_1-r_2) = (q-p)n+(r_1-r_2)$, e $0 < r_1 - r_2 < n$. Mas isto vai contra a hipótese de que $n|(a-b)$, pois desta forma, o resto da divisão de $a-b$ por n não é zero. Como o caso $r_1 < r_2$ é análogo, segue que $r_1 = r_2$, e portanto, $a \equiv b(mod\ n)$. □

Teorema 2.2.1 (Bézout)**.** Se a e b são inteiros não nulos e $d = mdc(a, b)$ (máximo divisor comum), então existem inteiros x e y tal que $d = ax+by$. Se a e b são positivos, então podemos escolher $x > 0$ e $y < 0$, ou vice-versa.

Demonstração. Seja $C = \{ax + by$, com $ax + by > 0$ e $x, y \in \mathbb{Z}\}$. O conjunto C é não vazio pois a e b são inteiros positivos. Uma vez que $a^2 + b^2 > 0$, segue que $a^2 + b^2 \in C$. Seja m o menor elemento de C. É fácil ver que $d|m$. Como $m, d \in C$, segue que $m > 0$ e $d > 0$. Seja $a = qm + r$, com $q \in \mathbb{Z}$ e $0 \leq r < m$. Desta forma,

$$0 \leq r = a - q(ax + by) = a - aqx - bqy = a(1 - qx) - bqy \in C.$$

Como $r < m$, segue que $r = 0$. Analogamente, tem-se que $m|b$. Portanto, $m|mdc(a, b) = d$. Vamos mostrar, agora, que se existe c inteiro, tal que $c|a$ e $c|b$, então $c \leq d$. De fato, $c|ax + by = d$ e $c \leq |c| \leq |d| = d$. □

Proposição 2.2.2. Se $a \equiv b(mod\ n)$ e $c \equiv d(mod\ n)$, então:
(i) $a + c \equiv b + d(mod\ n)$; (ii) $ac \equiv cd(mod\ n)$.

Demonstração. (i) Como $a \equiv b(mod\ n)$ e $c \equiv d(mod\ n)$, segue que $n|(a-b)$ e $n|(c-d)$. Logo, $n|(a-b)+(c-d) \Rightarrow n|(a+c)-(b+d)$. Portanto, $a+c \equiv b+d(mod\ n)$.

(ii) Por hipótese, segue que $a = b+xn$ e $c = d+yn$, sendo x, y números inteiros positivos. Desta forma,

$$ac = (b+xn)(d+yn) = bd+byn+xnd+xyn^2 = bd+n(by+xd+xyn),$$

e $z = by+xd+xyn$, então $ac = bd+zn$. Logo, $ac = bd(mod\ n)$. □

Definição 2.2.3 (Inverso modular)**.** O inverso de $a\ (mod\ n)$ é um inteiro a' tal que

$$aa' \equiv a'a \equiv 1\ (mod\ n).$$

Teorema 2.2.2. Se $mdc(a,n) = 1$, então existe um inteiro x tal que $ax \equiv 1\ (mod\ n)$.

Demonstração. Pelo *teorema de Bézout*, se $mdc(a,n) = 1$, então existem x_1 e y inteiros tal que $ax_1+ny = 1$, ou seja, $ax_1 \equiv 1\ (mod\ n)$. Vamos supor que exista x_2 tal que $ax_2+ny = 1$. Desta forma, $ax_2 \equiv 1\ (mod\ n)$, e assim, $ax_1 \equiv 1 \equiv ax_2\ (mod\ n) \Rightarrow a(x_1-x_2) \equiv 0\ (mod\ n)$. Mas, $mdc(a,n) = 1 \Rightarrow m|(x_1-x_2) \Rightarrow x_1 \equiv x_2(mod\ n)$. Portanto, mostramos que se o máximo divisor comum de dois inteiros a e n existe e é igual a 1, então o inverso de $a(mod\ n)$ existe e é único. □

Exemplo 2.2.1. Vamos determinar o inverso de 51 em $\mathbb{Z}_{3865}$. Pela definição anterior, segue que o inverso de um inteiro a é um inteiro a' tal que $aa' \equiv a'a \equiv 1\ (mod\ n)$. Desta forma, para determinarmos a', primeiramente, vamos aplicar o algoritmo euclidiano estendido em 3865 e 51:

Resto	Quociente	x	y
3865	$*$	1	0
51	$*$	0	1
40	75	$1-0\cdot 75 = 1$	$0-1\cdot 75 = -75$
11	1	$0-1\cdot 1 = -1$	$1+75\cdot 1 = 76$
7	3	$1+1\cdot 3 = 4$	$-75-76\cdot 3 = -303$
4	1	$-1-4\cdot 1 = -5$	$76+303\cdot 1 = 379$
3	1	$4+5\cdot 1 = 9$	$-303-379\cdot 1 = -682$
1	1	$-5-9\cdot 1 = -14$	$379+682\cdot 1 = 1061$
0	3	$*$	$*$

Logo, o $mdc(3865, 51) = 1$, $\alpha = -14$ e $\beta = 1061$. Portanto,

$$3865\cdot(-14)+51\cdot 1061 = 1.$$

Aplicando a congruência módulo 3865 na igualdade acima, segue que $51\cdot 1061 \equiv 1(mod\ 3865)$, e pela Definição 2.2.3, segue que o inverso de 51 é 1061 em $\mathbb{Z}_{3865}$.

Definição 2.2.4. Um inteiro a é dito *resíduo quadrático* módulo n se a congruência $x^2 \equiv a \ (mod\, n)$ tiver solução, caso contrário, dizemos que a é um resíduo não quadrático módulo n.

Exemplo 2.2.2. $4^2 \equiv 2 \ (mod\ 7)$, pois $7|(16-2)$, segue que 2 é um resíduo quadrático módulo 7.

Definição 2.2.5. Seja p é um primo e x é um inteiro não divisível por p, o *Símbolo de Legendre* é definido como:

$$\left(\frac{x}{p}\right) = \begin{cases} 1, \text{ se } x \text{ é um resíduo quadrático de } p. \\ -1, \text{ se } x \text{ não é um resíduo quadrático de } p. \end{cases}$$

Exemplo 2.2.3. $\left(\frac{2}{7}\right) = \left(\frac{3}{7}\right) = 1$, pois $x^2 \equiv 2(mod\ 7)$ e $x^2 \equiv 3 \ (mod\ 7)$ tem solução.

2.2.2 Equações diofantinas

As equações diofantinas apresentam um importante papel na criptografia RSA.

Definição 2.2.6. Uma equação nas variáveis inteiras x e y do tipo

$$ax + by = c, \text{ com } a, b, c \in \mathbb{Z}$$

diz-se *equação diofantina.*

Teorema 2.2.3. Sejam a e b inteiros positivos, $c \in \mathbb{Z}$ e $mdc(a, b) = d$. A equação

$$ax + by = c, \text{ com } x, y \in \mathbb{Z}$$

tem solução se, e somente se, d divide c. Além disso, se x_0, y_0 são tais que $ax_0 + by_0 = c$, então a solução geral da equação $ax + by = c$ é dada por

$$x = x_0 + \left(\frac{b}{d}\right) k,$$
$$y = y_0 - \left(\frac{a}{d}\right) k,$$

onde $k \in \mathbb{Z}$.

Demonstração. Se $a = 0$ ou $b = 0$, então o resultado é imediato. Vejamos o caso em que $a \neq 0$ e $b \neq 0$. Nesse caso, a demonstração do teorema será dividida em duas partes, a primeira com relação a equação possuir solução se d divide c, e a segunda com relação a solução geral.

Primeira parte do teorema: Supondo que a equação $ax + by = c$ tenha solução em $\mathbb{Z}$, e $mdc(a, b) = d$, então $d|ax$ e $d|by$ e, portanto, $d|ax + by \Longrightarrow d|c$. Reciprocamente, supondo que $d|c$, então existe $t \in \mathbb{Z}$ tal que $c = dt$. Pelo algoritmo euclidiano estendido existem $\alpha, \beta \in \mathbb{Z}$ tal que $a\alpha + b\beta = d$. Multiplicando essa última igualdade por t

tem-se que $a(t\alpha)+b(t\beta)=dt=c$ e, portanto $x=\alpha t$ e $y=\beta t$ são soluções da equação $ax+by=c$.

Segunda parte do teorema: Sejam x_0 e y_0 tal que $ax_0+by_0=c$. Assim, se $k\in\mathbb{Z}$, então

$$a\left(x_0+\frac{b}{d}k\right)+b\left(y_0-\frac{a}{d}k\right)=ax_0+\frac{ab}{d}k+by_0-\frac{ab}{d}k=ax_0+by_0=c,$$

o que mostra que $x=x_0+\left(\frac{b}{d}\right)k$, $y=y_0-\left(\frac{a}{d}\right)k$ é uma solução da equação $ax+by=c$. Vamos mostrar agora, que se $x,y\in\mathbb{Z}$ são tais que $ax+by=c$, então x,y são da forma dada acima. Para isso, vamos supor que o par (x',y') seja outra solução da equação. Assim,

$$ax'+by'=c=ax_0+by_0,$$

mas isto é equivalente a

$$a(x'-x_0)=b(y_0-y').$$

Como $mdc(a,b)=d$, segue que $d|a$ e $d|b$. Assim existem m e n inteiros tal que $a=md$ e $b=nd$, com $mdc(m,n)=1$. Desta forma,

$$m(x'-x_0)=n(y_0-y'),$$

e assim $m|n(y_0-y')$. Como $mdc(m,n)=1$, segue que m deve dividir y_0-y', isto é, $y_0-y'=mk$, para algum $k\in\mathbb{Z}$. Portanto,

$$y'=y_0-mk=y_0-\frac{a}{d}k.$$

Assim, $m(x'-x_0)=n(y_0-y')=nmk$, e portanto, $x'-x_0=nk$, ou equivalentemente,

$$x'=x_0+nk=x_0+\frac{b}{d}k,$$

o que prova o resultado. □

2.3 Grupos

Nesta seção estudaremos alguns resultados importantes com relação a teoria dos grupos, pois estes serão importantes ao longo do trabalho.

2.3.1 Grupos e subgrupos

Definição 2.3.1. Um conjunto G onde está definida uma operação $*$ é um *grupo* se a operação satisfaz as seguintes propriedades:

(1) **Associatividade:** dados a,b, e $c\in G$ tem-se que

$$a*(b*c)=(a*b)*c.$$

(2) **Elemento Neutro:** existe um elemento $e \in G$ tal que para todo $a \in G$ tem-se que

$$a * e = e * a = a.$$

(3) **Elemento inverso:** dado um elemento $a \in G$ qualquer, existe um elemento $a' \in G$ (o inverso de a) tal que

$$a * a' = a' * a = e.$$

Definição 2.3.2. Seja $(G, *)$ um grupo. Um subconjunto não vazio H de G é um *subgrupo* de G quando, com a operação de G, H é um grupo.

Para facilidade de notação, a partir de agora usamos que a operação do grupo é o produto.

Definição 2.3.3. Um grupo multiplicativo G se denomina *grupo cíclico* se existe um elemento $a \in G$ de maneira que $G = \{a^m;\ m \in \mathbb{Z}\}$. Ou seja, um grupo diz-se cíclico se for gerado por um único elemento (Notação: $G = [a]$. O elemento a é dito gerador de G).

Exemplo 2.3.1. O grupo multiplicativo $[i] = \{i^1 = i, i^2 = -1, i^3 = -i, i^4 = 1\} = \{1, -1, i, -i\}$ é um grupo cíclico.

Definição 2.3.4. Seja R uma relação de equivalência sobre E. Dado a, com $a \in E$, chama-se *classe de equivalência* por a, módulo R, o subconjunto $\overline{a}$ de E constituído pelos elementos x tais que xRa. Em símbolos:

$$\overline{a} = \{x \in E \mid xRa\}.$$

2.3.2 Classe lateral e teorema de Lagrange

Antes de apresentarmos o Teorema de Lagrange, veremos alguns resultados que serão importantes para demonstrarmos o teorema.

Definição 2.3.5. Sejam G um grupo e H um subgrupo de G. Dado $a \in G$, chamamos de *classe lateral* (à esquerda) módulo H o conjunto

$$aH = \{ah \mid h \in H\}.$$

Observação: 2.3.1. Definimos classe lateral (à direita) módulo H de maneira análoga, ou seja, $Ha = \{ha \mid h \in H\}$.

O conjunto das classes laterais à direita (esquerda), módulo H, determina uma *partição* em G, ou seja:

i) se $a \in G$, então $aH \neq \emptyset$;

ii) se $a, b \in G$, então $aH = bH$ ou $aH \cap bH = \emptyset$;

iii) a união de todas as classes laterais é igual a G.

Proposição 2.3.1. Seja H um subgrupo de G. Então duas classes laterais quaisquer módulo H são subconjuntos de G que tem a mesma cardinalidade.

Demonstração. Dadas duas classes laterais aH e bH, temos de mostrar que é possível construir uma aplicação bijetora $f : aH \to bH$. Lembrando a forma geral dos elementos dessas classes, é natural definir f da seguinte maneira: $f(ah) = bh$, para qualquer $h \in H$. Sem maiores difculdades, prova-se que f é injetora e sobrejetora. De fato:

• (injetora) Se h, $h_1 \in H$ e $f(ah) = f(ah_1)$, então $bh = bh_1$; como, porém, todo elemento de G é regular, então $h = h_1$.

• (sobrejetora) Seja $y \in bH$. Então $y = bh$, para algum $h \in bH$. tomando-se $x = ah \in aH$, então $f(x) = f(ah) = bh = y$. □

Definição 2.3.6. Seja G um grupo finito. Se $a \in G$ então a *ordem* de a é o menor inteiro $\alpha > 0$ tal que $a^\alpha = e$. A ordem de G, $|G|$, é o número de elementos de G.

Notação: $o(a) = \alpha$.

Teorema 2.3.1 (Teorema de Lagrange)**.** Em um grupo finito, a ordem de qualquer subgrupo divide a ordem do grupo.

Demonstração. Sejam G um grupo finito e H um subgrupo de G. Pelo item (iii) da Observação 2.3.1, segue que existem $g_1, g_2, \cdots, g_n \in G$ com $g_1 = e_G$, de forma que a união

$$G = g_1H \cup g_2H \cup \cdots \cup g_nH,$$

é disjunta. Desta forma, o número de elementos de G é igual a soma do número de elementos de cada classe lateral da união acima, ou seja,

$$|G| = |g_1H| + |g_2H| + \cdots + |g_nH|.$$

Pela Proposição 2.3.1, segue que $|g_iH| = |H|$, para $i = 1, \ldots, n$. Assim,

$$|G| = |g_1H| + |g_2H| + \cdots + |g_nH| = \underbrace{|H| + |H| + \cdots + |H|}_{n} = n|H|.$$

Como $|G| = n|H|$, segue que $|H|$ divide $|G|$. □

O lema a seguir, será utilzado mais adiante, no teste de Lucas

Lema 2.3.1. Seja G um grupo finito e seja $a \in G$. Um inteiro positivo t satisfaz $a^t = e$ se, e somente se, t é divisível pela ordem de a.

Demonstração. Seja s a ordem de a. Se s divide t então $t = sr$, para algum inteiro positivo r, e

$$a^t = (a^s)^r = e.$$

Reciprocamente, suponhamos que $a^t = e$. Como a ordem é o menor inteiro positivo s tal que $a^s = e$, então $s \leq t$. Dividindo t por s temos:

$$t = sq + r \text{ onde } 0 \leq r < s.$$

Assim,

$$e = a^t = (a^s)^q a^r = a^r,$$

uma vez que $a^s = e$. Como $r < s$, isto só pode acontecer se $r = 0$, ou teríamos uma contradição com o fato de s ser a ordem de a. □

2.3.3 A função de Euler

Definição 2.3.7. Dado um inteiro positivo n, denota-se por $U(n)$ o conjunto de todos os elementos inversíveis de $\mathbb{Z}_n$. Ou seja,

$$U(n) = \{\overline{a} \in \mathbb{Z}_n : mdc(a, n) = 1\}.$$

Este conjunto é um grupo para a operação de multiplicação de classes de $\mathbb{Z}_n$. O número de elementos do conjunto $U(n)$ é igual a

$$\phi(n) = \#\{1 \leq r \leq n, mdc(r, n) = 1\},$$

onde ϕ é a função de Euler, e $\#$ indica a quantidade de elementos do conjunto.

No caso do cálculo de $U(p)$, sendo p primo, tem-se que $mdc(a, p) = 1$, ou seja, p não divide a. Se caso p divide a, então $\overline{a} = \overline{0}$. Portanto, se p é primo, então todas a classes diferentes de $\overline{0}$ tem inverso. Assim,

$$U(p) = \mathbb{Z}_p \backslash \{\overline{0}\}.$$

Isto só é válido quando p é primo, pois por exemplo, $U(4) = \{\overline{1}, \overline{3}\}$.

Em termos de congruência, dizemos que a tem ordem α módulo n se $a^\alpha \equiv 1 \ (mod\, n)$, onde $a, n \in \mathbb{Z}$, $mdc(a, n) = 1$ e α está nas condições do da Definição 2.3.6. Se $\alpha = \phi(n)$ então dizemos que a é uma *raiz primitiva* de n (ou módulo n).

Apresentamos agora o teorema da raiz primitiva, o qual sera usado mais adiante no Teste de Lucas e sua demonstração pode ser encontrada na referência [4].

Teorema 2.3.2 (Teorema da raiz primitiva)**.** Se p é um primo, então o grupo $U(p)$ é cíclico.

Teorema 2.3.3. Se p é um primo e a um inteiro positivo, então

$$\phi(p^a) = p^a - p^{a-1},$$

onde ϕ é a função de Euler.

Demonstração. Pela definição de $\phi(n)$, segue que $\phi(p^a)$ é o número de inteiros positivos não-superiores a p^a e relativamente primos com p^a. Mas os únicos não primos com p^a e menores ou iguais a p^a, são os divisíveis por p. Como os múltiplos de p não superiores a p^a são p^{a-1}, segue que $\phi(p^a) = p^a - p^{a-1}$. □

A seguinte proposição cuja demonstração pode ser encontrada em [4], será usada no próximo capítulo, que trata da criptografia RSA.

Proposição 2.3.2. Se $mdc(m,n) = 1$, então $\phi(mn) = \phi(m)\phi(n)$.

2.4 Números primos e testes de primalidade

A descoberta dos números primos é imprescindível na Matemática, pois eles intitulam o princípio central na teoria dos números, consistindo no Teorema Fundamental da Aritmética. Além deste importante teorema, veremos nesta seção, alguns testes para a determinação de números primos. Estes serão importantes para o desenvolvimento do próximo capítulo, onde abordaremos a criptografia RSA, e esta é baseada em números primos.

Existem muitas demonstrações quanto a infinidade dos números primos. A seguir, apresentaremos uma delas.

Teorema 2.4.1. Existem infinitos números primos.

Demonstração. Sejam $C = \{p_1, p_2, \cdots, p_n\}$ um conjunto finito com todos os números primos e $P = p_1 \cdot p_2 \cdots p_n$ o produto desses primos. Seja $q = P + 1$. Se q for primo, então $q = p_i$ para algum $i = 1, 2, \cdots, n$. Assim, $q|1$, o que não ocorre. Se q não for primo, então pelo teorema fundamental da aritmética, segue que q possui um fator primo p, que deve ser um dos $p_1, p_2, \cdots, p_n$, e consequentemente, divide o produto $p_1 \cdot p_2 \cdots p_n$. Mas como $p|q$ e $p|p_1 \cdot p_2 \cdots p_n$, segue que $p|1$, o que é um absurdo. □

Uma das demonstrações mais importantes é a que foi dada por Euler em 1737 [4].

2.4.1 Teorema fundamental da aritmética

A seguir, apresentamos um dos mais importantes teoremas da aritmética que diz que todo número inteiro maior ou igual a 2 pode ser escrito como produto de números primos. Por exemplo, 50 é escrito de maneira única, a menos da ordem dos fatores, como $2^1 5^2$. A ordem dos fatores, pela propriedade comutativa da multiplicação é irrelevante.

Teorema 2.4.2. Dado um inteiro $n \neq -1, 0, 1$ podemos escrevê-lo, de modo único, na forma

$$n = \pm p_1^{e_1} \dots p_k^{e_k},$$

onde $1 < p_1 < p_2 < p_3 < \cdots < p_k$ são primos ao passo que $e_1, \cdots, e_k$ são inteiros positivos.

Demonstração. Existência da fatoração. Seja $n \geq 2$ inteiro. Vamos dividir n por todos os inteiros de 2 até $n-1$. Se algum destes inteiros dividir n, então teremos p_1 um fator de n que é primo. Logo $2 \leq p_1 \leq n-1$. Supondo que p_1 seja o menor fator de n, e que p_1' é um fator maior do que 1, segue que, existem inteiros a e b tais que

$$n = p_1 a \text{ e } p_1 = p_1' b.$$

Assim, $n = p_1' ab$. Portanto, p_1' também é um fator de n. Com p_1 é o menor fator de n, segue que $p_1 \leq p_1'$. Mas, como p_1' é fator de p_1, segue que $p_1' \leq p_1$. Destas duas desigualdades, obtemos que $p_1 = p_1'$. Como existe um único fator p_1 maior que 1, segue que p_1 é primo. Ao repetirmos o processo descrito acima para $m_1 = \dfrac{n}{p_1}$, obtemos um fator p_2 de m_1. Em seguida, tomamos $m_2 = \dfrac{m_1}{p_2}$ e repetimos o processo descrito acima. Após um número i de etapas, encontramos $m_i = p_i$. Assim, $n = p_1 p_2 \dots p_i$. Ao agruparmos os termos $p_j's$ iguais, teremos

$$n = p_1^{e_1} \dots p^{e_k},$$

onde $e_1 \dots e_k$ são inteiros positivos. □

2.4.2 Teste de primalidade de Fermat

O Teorema de Fermat, que originou o Teste de primalidade de Fermat, oferece um teste simples e eficiente para ignorar números não-primos. Qualquer número que falhe no teste não é primo, o que não significa que os números que satisfazem o teste são primos.

Teorema 2.4.3 (Pequeno Teorema de Fermat)**.** Se p é um primo, então p divide $a^p - a$, $\forall a \in \mathbb{Z}$, ou seja,

$$a^{p-1} \equiv 1 \pmod{p}.$$

Demonstração. Primeiramente, vamos colocar os possíveis resíduos módulo p:

$$1,\ 2,\ 3, \dots,\ p-1.$$

Multiplicamos esses resíduos por a:

$$a1,\ a2,\ a3, \dots,\ a(p-1).$$

Seja r_1 o resíduo de $a1$, r_2 o resíduo de $a2$, e assim, sucessivamente. Desta forma:

$$\begin{aligned} r_1 &\equiv a1 \pmod{p}, \\ r_2 &\equiv a2 \pmod{p}, \\ &\vdots \\ r_{p-1} &\equiv a(p-1) \pmod{p}. \end{aligned}$$

Multiplicando os termos, obtemos

$$r_1 r_2 r_3 \dots r_{p-1} \equiv (a1)(a2)(a2)\dots(a(p-1)) \ (mod\ p).$$

Como

$$(a1)(a2)(a3)\dots(a(p-1)) = a^{p-1}(123(p-1)),$$

segue que

$$r_1 r_2 r_3 \dots r_{p-1} \equiv a^{p-1}(123\dots(p-1)) \ \ (mod\ p).$$

□

Exemplo 2.4.1. Neste exemplo vamos calcular $2^{5432675}$ $(mod\ 13)$. Pelo teorema acima, basta calcular o resto da divisão de 5432975 por 12, resultando em 11. Desta forma

$$2^{5432675} \equiv 2^{11} \equiv 7\ (mod\ 13).$$

Existe uma classe famosa conhecida como *números pseudoprimos de Fermat* que são os números compostos que satisfazem o *pequeno teorema de Fermat*.

Exemplo 2.4.2. O número 341 é um pseudoprimo de Fermat para base 2. De fato, temos que $2^{341} \equiv 1\ (mod\ 341)$, mas $341 = 11 \cdot 31$. Logo, 341 é composto e atende as condições do teorema, portanto, é um pseudoprimo.

2.4.3 Fórmulas polinomiais

Um dos métodos mais simples que os matemáticos utilizaram para tentar encontrar números primos foi através da fórmula polinomial. Com isto nos referimos a um polinômio

$$f(x) = a_n x^n + a_{n-1}x^{n-1} + \cdots + a_1 x + a_0,$$

onde $a_n, a_{n-1}, \cdots, a_1, a_0$ são números inteiros que satisfazem a seguinte condição:

$$f(m) \text{ é primo para todo } m \text{ inteiro positivo.}$$

Exemplo 2.4.3. $f(x) = x^2 + 1$. É fácil ver que este polinômio possui vários valores que não são primos, como podemos ver na tabela abaixo:

x	$f(x)$
1	2
2	5
3	10
4	17
5	26
6	37
7	50
8	65

Observe que se x for ímpar, então $f(x)$ é par. Assim, quando $x \neq 1$ e x é ímpar, o valor de $f(x)$ é sempre composto (e múltiplo de 2).

Portanto, quando $x > 1$, temos que $f(x)$ é sempre primo se x for par. Não podemos considerar apenas os valores pares para x, uma vez que $f(8) = 65$ é composto. Vemos, então, que este polinômio não nos dá uma fórmula para primos.

Teorema 2.4.4. Se $f(x)$ é um polinômio com coeficientes inteiros, então existe uma infinidade de inteiros positivos n tais que $f(n)$ é composto.

Demonstração. Apesar de o teorema ser geral, vamos demonstrá-lo para o caso em que f tem grau 2, pois as ideias da demonstração são semelhantes para a demonstração do caso geral. Seja $f(x) = ax^2 + bx + c$, com a, b, $c \in \mathbb{Z}$. Vamos supor que $a > 0$. Se $a < 0$ o raciocínio seria análogo com $-f(x)$. Seja n um inteiro, tal que $f(n)$ seja um número primo p, e consideremos um inteiro positivo q tal que $apq > -b - 2an$. Desta forma,

$$\begin{aligned} f(n+pq) = a(n+pq)^2 + b(n+pq) + c = (an^2 + bn + c) + p(apq^2 + bq + 2anq) = \\ = p + p(apq^2 + bn + 2anq) = p(1 + apq^2 + bn + 2anq). \end{aligned}$$

Sendo $q > 0$ e $apq + b + 2an > 0$, tem-se que $apq^2 + bn + 2anq > 0$, e assim, $1 + apq^2 + bn + 2anq > 1$. Portanto, $f(n+pq)$ é um produto de um número primo por um número maior que 1, ou seja, $f(n + pq)$ é sempre composto quando

$$q > \frac{-b - 2an}{ap}.$$

Como existem infinitos valores que q pode assumir, segue que existem infinitos valores positivos de n tal que $f(n)$ é composto. □

2.4.4 Fórmulas fatoriais

Vamos supor p um número primo positivo. Vamos construir uma função semelhante a fatorial, mas apenas os números primos são multiplicados.

Seja p' essa função. Por exemplo, $2' = 2$, e $5' = 2 \cdot 3 \cdot 5 = 30$. Estamos interessados nos números da forma $p' + 1$. Para melhor entendimento, observe a tabela abaixo.

p	p'	$p' + 1$
2	2	3
3	6	7
5	30	31
7	210	211
11	2310	2311
13	30030	30031

Este não é um bom método, pois como podemos ver na tabela acima, para $p = 13$, temos $p' + 1 = 13' + 1 = 30031 = 59 \cdot 509$, que é composto.

São conhecidos somente 16 primos da forma $p' + 1$, sendo que, o maior deles é $p = 24029$.

2.4.5 Divisão por tentativa

É o método mais trivial para determinar se um número é primo ou não, baseando-se no teste sucessivo de divisões, de 2 até o número testado n, procurando se há outro divisor diferente de n.

Exemplo 2.4.4. Vejamos, por meio do método de *divisão por tentativa*, se o número 11 é primo. Para fazer a verificação, é feita a divisão de 11 por 2. Como 11 não é divisível por 2, é feita a divisão de 11 por 3, e assim sucessivamente até chegarmos ao número 11. Como o único número que divide 11 é ele mesmo, concluímos que 11 é primo.

Apesar desse método ser efetivo com relação a resultados, o tempo de execução de um algoritmo que faça esse tipo de divisão não é viável quando aplicado a números com muitas casas decimais.

2.4.6 ρ-Método de Pollard

O ρ-Método de Pollard torna-se eficiente para determinar o expoente de decriptação do código RSA, a partir de uma chave pública, quando esse código possuir menos de 160 bits. O código RSA será discutido com mais detalhes posteriormente neste trabalho.

O algoritmo de ρ-Pollard consiste em encontrarmos um fator de um inteiro maior que 10^5, limitando-se a encontrar fatores de um número que não seja superior a 10^{10}. A ideia do algoritmo é a seguinte: É escolhido um polinômio irredutível $f(x)$ com coeficientes inteiros e um valor inicial x_0. Seja d um fator não trivial de n. Recursivamente, geramos a sequência:

$$x_i \equiv f(x_{i-1}) \ (mod\ n).$$

Se $y_i \equiv x_i \ (mod\ d)$, então $y_i \equiv f(y_{i-1}) \ (mod\ d)$. Note que, para algum par (i, j), $y_i = y_j$ pois existe um número finito de classes de congruência módulo d. Desta forma

$$y_{i+t} = y_{j+t}, \forall t > 0.$$

Se $y_i = y_j$, então $x_i \equiv x_j \ (mod\ d)$. Desta forma, tem-se que $d|(x_i - x_j)$, e deste modo $mdc(x_i - x_j, n) > 1$ é um divisor de n.

Como d é desconhecido, segue que também não sabemos os valores de y_i. Sabemos que existem infinitos pares (i, j) tal que $y_i = y_j$. Tomando l como o número correspondente ao comprimento do ciclo, então para qualquer par (i, j) para o qual $l|(j - i)$, tem-se que

$$j \equiv i \ (mod\ l) \Longrightarrow y_i \equiv x_i.$$

Podemos repetir esse processo escolhendo pares (i, j), e dessa forma, calculamos o $mdc(x_i - x_j, n)$. Uma maneira de encontrar vários pares (i, j) é tomar $2^\alpha < i \leq 2^{\alpha+1}$ para algum α natural, e testar se o $mdc(x_i - x_{2^\alpha}, n) \neq 1$. Se o máximo divisor comum não for 1, então será tomado um t sendo o menor inteiro tal que $j + t = 2^\alpha$ e $2^\alpha < i + t \leq 2^{\alpha+1}$. Com relação ao polinômio, devemos escolher f tal que a sequência (x_i) seja parecida com uma sequência aleatória.

Exemplo 2.4.5. Neste exemplo, vamos encontrar um divisor não trivial de $n = 28$ utilizando o ρ-método de Pollard, com valor inicial $x_0 = 2$ e polinômio $f(x) = x^2 + 1$. Assim, para determinarmos o valor não trivial, utilizaremos o ρ-método de Pollard para encontrar as divisões de $f(x)$. Em seguida aplicamos esses valores no algoritmo de Euclides. Assim, como $f(x_{s+1}) \equiv f(x_s)(mod\ n)$ tem-se que

$f(x_{0+1}) \equiv f(x_0)(mod\ n)$
$f(x_1) \equiv 5\ (mod\ 28)$
$f(x_1) \equiv 5;$
$f(x_2) \equiv (5^2 + 1)(mod\ 28)$
$f(x_2) \equiv 26\ (mod\ 28)$
$f(x_2) \equiv 26;$
$f(x_3) \equiv f(x_2)(mod\ n)$
$f(x_3) \equiv (26^2 + 1)(mod\ 28)$
$f(x_3) \equiv 677\ (mod\ 28)$
$f(x_3) \equiv 5;$
$f(x_4) \equiv f(x_3)(mod\ 28)$
$f(x_4) \equiv (5^2 + 1)(mod\ 28)$
$f(x_4) \equiv 26\ (mod\ 28)$
$f(x_4) \equiv 26.$

Calculamos apenas até $f(x_4)$ pois esses valores são suficientes para a resolução do exemplo. Aplicando o algoritmo de Euclides, tem-se que

$$mdc(x_2 - x_1, 28) = mdc(26 - 5, 28) = mdc(21, 28) = 1.$$

$$mdc(x_4 - x_2, 28) = mdc(26 - 26, 28) = mdc(0, 28) = 28.$$

Portanto, um fator de 28 é o próprio 28.

2.4.7 Crivo de Erastóteles

Este é um método simples para a determinação de números primos até um número máximo desejado.

Teorema 2.4.5. Se n não é um primo, então n possui, necessariamente, um fator primo menor ou igual a $\sqrt{n}$.

Demonstração. Como n não é primo, segue que n é composto, e dessa forma, pode ser escrito como $n = ab$, onde $0 < a < n$ e $0 < b < n$. Suponhamos que $a \leq b$, sendo o caso $b \leq a$ análogo. Vamos supor ainda que, por absurdo, que $a > \sqrt{n}$. Assim, podemos afirmar, que $n = \sqrt{n}\sqrt{n} \leq aa \leq ab = n$, o que resulta em $n < n$, o que é um absurdo. Portanto, $a \leq \sqrt{n}$, e pelo Teorema 2.4.1, segue que a possui algum fator primo que deve ser menor que ele mesmo e, assim, menor que $\sqrt{n}$. Deste modo, concluímos que n possui um fator primo menor que $\sqrt{n}$. □

Exemplo 2.4.6. O número 36 possui algum primo menor ou igual a $\sqrt{36}$. Temos que 36 pode ser escrito como $36 = 2^2 \cdot 3^2$. Como $\sqrt{36} = 6$, segue que os fatores primos de 36 são menores que $\sqrt{36}$.

2.4.8 Crivo quadrático

O crivo quadrático é um dos mais importantes métodos de fatoração já desenvolvidos. Este método baseia-se no fato de que se existirem números x e y que satisfaçam a condição

$$x^2 \equiv y^2 \ (mod\ n),$$

então

$$x^2 - y^2 \equiv 0 \ (mod\ n),$$

ou seja,

$$(x + y)(x - y) \equiv 0 \ (mod\ n).$$

Desta forma,

$$n|(x + y)(x - y).$$

Sejam d e f dois números tais que $d = mdc(x + y, n)$ e $f = mdc(x - y, n)$. Estes números poderão ser fatores não triviais de n. Em outras palavras, a ideia básica deste método é encontrar congruências da forma $x_i^2 \equiv y_i \ (mod\ n)$, onde $\prod y_i = y^2$ é um quadrado perfeito. Se $\prod x_i = x^2$, então $x^2 \equiv y^2 (mod\ n)$. Para encontrarmos x e y, devemos primeiramente encontrar uma base de fatores, que é um conjunto de números primos e, em seguida, determinar um conjunto de números completamente fatorados sobre uma base de fatores.

A base de fatores é formada por números primos p_i, com $i = 1, \cdots, m$, tal que $p_i \leq B$, sendo B o limite dado, e para cada primo p, o número n deve ser um resíduo quadrático módulo p. Desta forma, serão considerados os números primos menores do que B tais que:

$$n^{(p-1)/2} (mod\ p) \equiv 1 \quad \Leftrightarrow \quad \left(\frac{n}{p}\right) = 1.$$

De acordo com Crandall e Pomerance (2005), o valor para o limite B é calculado a partir de experimentação, não há uma fórmula exata. Se dois valores de B tiverem resultados similares, obviamente é vantajoso escolher o menor. A seguinte fórmula $B = (e^{\sqrt{ln(n)ln(ln(n))}})^{1/2}$ pode encontrar o valor aproximado de B.

Segundo Landquist (2001), a fórmula $B = (e^{\sqrt{ln(n)ln(ln(n))}})^{\sqrt{2}/4}$ também pode encontrar o valor aproximado de B.

Exemplo 2.4.7. Seja $n = 10186$. Vamos verificar se o primo 3 pertence a base de fatores. Tem-se que $10186^{(3-1)/2}\ (mod\ 3) = 10186^1\ (mod\ 3) \equiv 1$. Portanto 3 é incluído na base de fatores. Caso contrário, 3 seria descartado e o próximo primo seria examinado, e assim sucessivamente, até $p_i \leq B$.

Após a base de fatores ser encontrada, devemos determinar os $f(x_i)'s$ que sejam completamente fatorados pela base de fatores por meio da expressão: $f(x_i) = x_i^2 - n$, onde n é o número a ser fatorado e x_i são números próximos de $\lfloor\sqrt{n}\rfloor$, onde $\lfloor\sqrt{n}\rfloor$ é o maior inteiro menor ou igual a $\sqrt{n}$. É aconselhável que -1 esteja contido na base de fatores, mesmo não sendo primo, para o caso de algum $f(x_i) < 0$.

Exemplo 2.4.8. Vamos aplicar o método do crivo quadrático para $n = 9487$, com $B = 30$. Primeiramente, vamos encontrar uma base de fatores. Como $B = 30$, vamos verificar que para cada primo $p < 30$, n é um resíduo quadrático módulo p. Para isso, utilizamos o teste $n^{(p-1)/2}\ (mod\ p)$. Após a aplicação do teste para os primos $p < 30$, obtemos a base de fatores $\{-1, 2, 3, 7, 11, 13, 17, 19, 29\}$. Note que o -1 foi incluído na base para o caso de obtermos $f(x_i) < 0$. Em seguida, vamos calcular os $f(x_i)'s$ completamente fatorados pela base encontrada, para x_i próximo de $\sqrt{9487}$. Por exemplo, se $x_i = 98$, então

$$f(x_i) = x_i^2 - n \Longrightarrow f(98) = 98^2 - 9487 = 117 = 3^2 \times 13,$$

ou seja, 117 é completamente fatorado pela base de fatores. Em seguida, para cada valor de $f(x_i)$ encontrado, é associado um vetor de 9 dígitos binários, com cada coluna correspondendo a um dos primos da base. Se o número for negativo, então o primeiro dígito será 1, caso contrário, será 0. Se o primo correspondente tem potência par, então o dígito será 0, caso contrário, será 1. A seguir, veremos exemplos de $x_i's$ e $f(x_i)'s$ e seus dígitos binários correspondentes.

x_i	$f(x_i)$	-1	2	3	7	11	13	17	19	29
81	-2926	1	1	0	1	1	0	0	1	0
84	-2431	1	0	0	0	1	1	1	0	0
85	-2262	1	1	1	0	0	1	0	0	1
89	-1566	1	1	1	0	0	0	0	0	1
95	-462	1	1	1	1	1	0	0	0	0
97	-78	1	1	1	0	0	1	0	0	0
98	117	0	0	1	0	0	0	0	1	0
100	513	0	0	1	0	0	0	0	1	0
101	714	0	1	1	1	0	0	1	0	0
103	1122	0	1	1	0	1	0	1	0	0

Calculando $v(81) + v(95) + v(100)$, obtemos o vetor $(0, 0, 0, 0, 0, 0, 0, 0, 0)$, que corresponde ao quadrado perfeito $(81 \times 95 \times 100)^2 = (2 \times 3^2 \times 7 \times 11 \times 19)^2 (mod\ 9487)$. Assim,

$$x = 81 \times 95 \times 100 (mod\ 9487) \equiv 1053$$

e

$$y = 2 \times 3^2 \times 7 \times 11 \times 19 (mod\ 9487) \equiv 7360.$$

Calculando $d = mdc(1053 + 7360, 9487) = 179$, obtemos um fator não trivial de n. O segundo fator é dado por $9487 \div 179 = 53$.

A seguir, presentamos métodos engenhosos para achar primos.

2.4.9 Números de Carmichael

Definição 2.4.1. Seja n um inteiro *composto* ímpar tal que $b^n \equiv b\ (mod\ n)$ para todo $1 < b < n - 1$. Esse número recebe o nome de *número de Carmichael*.

Apesar de a definição ser simples, a determinação de números de Carmichael nem sempre é rápida, mesmo utilizando processos computacionais. Desta forma, para determinar se um número é de Carmichael, temos o seguinte teorema:

Teorema 2.4.6 (Teorema de Korselt)**.** Um inteiro positivo ímpar n é um número de Carmichael se, e somente se, cada fator primo p de n satisfaz as duas condições seguintes:

(1) p^2 não divide n;

(2) $p - 1$ divide $n - 1$.

A demonstração desse teorema foge do escopo desse trabalho. Para maiores detalhes consultar a referência [4].

Exemplo 2.4.9. Para verificar que o número 561 é um número de Carmichael, temos que mostrar:

$$b^{561} \equiv b \ (mod\ 561)$$

para todo $b = 2, \cdots, 559$. Se fatorarmos 561, então $561 = 3 \cdot 11 \cdot 17$. Logo, basta mostramos que $b^{561} - b$ é divisível por 3, 11 e 17. Desta forma teremos que $561|(b^{561} - b)$.

- Verificando que $b^{561} \equiv b \ (mod\ 3)$:

Se $3|b$, então $b^{561} \equiv b \equiv 0 \ (mod\ 3)$;

Se 3 não divide b, então $b^2 \equiv 1 \ (mod\ 3)$. Logo $b^{561} \equiv (b^2)^{280} b \equiv b \ (mod\ 3)$.

- Verificando que $b^{561} \equiv b \ (mod\ 11)$:

Se $11|b$, então $b^{561} \equiv b \equiv 0 \ (mod\ 11)$;

Se 11 não divide b, então $b^{10} \equiv 1 \ (mod\ 11)$. Logo $b^{561} \equiv (b^{11})^{56} b \equiv b \ (mod\ 11)$.

- Verificando que $b^{561} \equiv b \ (mod\ 17)$:

Se $17|b$, então $b^{561} \equiv b \equiv 0 \ (mod\ 17)$;

Se 17 não divide b, então $b^{16} \equiv 1 \ (mod\ 17)$. Logo $b^{561} \equiv (b^{16})^{35} b \equiv b \ (mod\ 17)$.

2.4.10 Teste de Lucas

Teorema 2.4.7. Seja n um inteiro positivo ímpar e b um inteiro tal que $2 \leq b \leq n-1$. Se

$$b^{n-1} \equiv 1 \ (mod\ n) \text{ e } b^{(n-1)/p} \not\equiv 1 \ (mod\ n),$$

para cada fator primo p de $n-1$, então n é primo.

Demonstração. Seja σ a ordem de $\overline{b}$ em $U(n)$, e $b^{(-n-1)} = \overline{1}$. Pelo Teorema 2.3.2 (teorema da raiz primitiva), segue que $\sigma|(n-1)$. Desta forma, podemos escrever $n - 1 = \sigma m$. Para atender as condições do teorema, temos que mostrar que $m = 1$. Vamos supor, por absurdo que $m > 1$. Assim, m é divisível por algum primo p. Note que, se $p|m$, então $p|(n-1)$. Como o resultado de ambas as divisões são inteiros, segue que

$$\frac{n-1}{p} = \frac{\sigma m}{p} \Rightarrow \frac{\frac{n-1}{p}}{\sigma} = \frac{m}{p}.$$

Desta forma, σ divide $\dfrac{n-1}{p}$. Pelo lema 2.3.1, segue que

$$\overline{b}^{\frac{n-1}{p}} = \overline{1}.$$

Portanto, $m = 1$, e assim, $\sigma = n - 1$. Pelo teorema de Lagrange, segue que σ tem que dividir a ordem de $U(n)$, ou seja, $(n-1)|\sigma(n)$, mas como $\sigma(n) \leq n-1$, segue que $\sigma(n) = n - 1$, e n tem que ser primo. □

Exemplo 2.4.10. Para $n = 7$, temos: $n - 1 = 6, b = 3$ tal que $2 \cdot 3 \cdot 6$. Verificando a congruência: $b^{n-1} \equiv 1 \ (mod\ n)$, tem-se que $3^6 \equiv 1 \ (mod\ n)$, pois $729 \div 7 = 104$ e resto igual a 1. Decompondo 6, segue que $6 = 2 \cdot 3$. Logo $b^{(n-1)/p} \not\equiv 1 \ (mod\ n)$, para $p = 2$ e $p = 3$. Como $3^3 \not\equiv 1 \ (mod\ 7)$, pois $27 \div 7 = 3$ e resto igual a 6 e $3^2 \not\equiv 1 \ (mod\ 7)$, pois $9 \div 7 = 1$ e resto igual a 2, segue que 7 é primo.

2.4.11 Teste de Miller

Seja j o menor expoente tal que $b^{2^j q} \equiv 1 (mod\ n)$. Se $j \geq 1$, então podemos escrever

$$b^{2^j q} - 1 = (b^{2^{j-1} q} - 1)(b^{2^{j-1} q} + 1).$$

Se n é primo e divide $b^{2^j q} - 1$, então, pela minimalidade de j, segue que n divide $(b^{2^{j-1} q} + 1)$. Logo,

$$b^{2^{j-1} q} \equiv -1 \ (mod\ n).$$

Isto quer dizer que uma das potências $b^q, b^{2q}, \cdots, b^{2^k q}$ deve ser congruente a -1 módulo n. Se isto não acontecer, n deve ser composto.

Método prático do teste de Miller

(i) Divida $n - 1$ por 2 até encontrar q ímpar e k tal que $n - 1 = 2^k q$.
(ii) Faça $i = 0$ e $r =$ resto da divisão de b^q por n.
(iii) Se $i = 0$ e $r = 1$ ou $i \geq 0$ e $r = n - 1$ então o teste é inconclusivo.
(iv) Faça $i = i + 1$ e $r = r_2$, onde r_2 é o resto da divisão de r^2 por n.
(v) Se $i < k$ volte à etapa 3; senão: n é composto.

Exemplo 2.4.11. O número 561 é composto. Pelo método prático, temos que $n-1 = 560$, que pode ser escrito como $560 = 2^4 \cdot 35$. Calculando as potências de 2 módulo 561 para os expoentes $35, 2 \cdot 35, 2^2 \cdot 35$ e $2^3 \cdot 35$, segue que

$$\begin{gathered} 2^q \ (mod\ n) \\ 2^{35} \ (mod\ 561) = 263 \\ 2^{2 \cdot 35} \ (mod\ 561) = 166 \\ 2^{2^2 \cdot 35} \ (mod\ 561) = 67 \\ 2^{2^3 \cdot 35} \ (mod\ 561) = 1. \end{gathered}$$

Logo, 561 é composto.

2.4.12 Teste AKS

O teste AKS recebe esse nome devido aos seus criadores, Manindra Agrawal, Neeraj Kayal e Nitin Saxena. Este teste é baseado no conceito de equivalência e através dele, podemos verificar se um dado número é primo ou composto. O teste é baseado no próximo teorema.

Teorema 2.4.8. Se $a \in \mathbb{Z}_p^*$, então $p > 1$ é primo se:

$$(x-a)^p \equiv x^p - a \ (mod\ p).$$

Pelo teorema binomial de Newton, segue que:

$$(x-a)^p = \sum_{i=0}^{p} \binom{p}{i} (-1)^{p-i} x^i a^i.$$

Note que quando $i \neq 1$ e $i \neq p$, o coeficiente binomial $\binom{p}{i}$ é divisível por p. Assim, todos os termos intermediários desta expansão são divisíveis por p, ou seja, são congruentes a zero módulo p.

Como o objetivo do AKS é deixar o teste binomial rápido, há uma mudança com relação ao *mod* p, em vez disso, é trabalhado com o polinômio $x^r - 1$, sendo r um primo razoavelmente pequeno. Isto quer dizer que no lugar de ser calculado $(x-a)^p$, é calculado o resto da divisão de $(x-a)^p$ por (x^r-1), que é feito usando o mesmo método em Álgebra para dividir um polinômio por outro. Assim, tem-se no máximo $r-1$ termos, enquanto que na expansão de $(x-a)^p$ teríamos $p-1$ termos.

O AKS considera a seguinte congruência

$$(x-a)^p \equiv (x^p - a)\ mod(x^r - 1, p),$$

onde $mod(x^r-1,p)$ representa aplicar $mod(x^r-1$ e ao resultado aplicar *mod* p, então tem-se que a congruência do teorema também é satisfeita sempre que p for primo pois, uma vez que, ao dividir essas duas expressões por (x^r-1), estas duas terão restos iguais. Note que quando p é composto, teremos $(x-a)^p \not\equiv x^p - a\ (mod\ p)$.

Exemplo 2.4.12. Vamos verificar que 7 é primo e 4 é composto utilizando o teste AKS. Primeiramente, verifiquemos que 7 é primo. Seja $p = 7, a = 2$ e $r = 3$. Substituindo os valores na equação $(x-a)^p \equiv (x^p - a)\ mod(x^r-1,p)$, segue que

$$(x-2)^7 \equiv (x^7 - 2)\ mod(x^3 - 1, 7).$$

O quociente da divisão de (x^7-2) por (x^3-1) é igual a x^4+x, e o resto é igual a $x-2$. Calculando, $(x-2)$ $(mod\ 7)$, segue que o quociente é -1 e o é resto $x+5$. Agora, tomando-se $(x^p-a)\ mod(x^r-1,p) = (x^7-2)\ mod(x^3-1)$, pela expansão binomial segue que $(-128 + 448x - 672x^2 + 560x^3 - 280x^4 + 84x^5 - 14x^6 + x^7)\ mod\ (x^3-1)$, com quociente igual a $x^4 - 14x^3 + 84x^2 - 279x + 546$ e resto igual a $418 + 169x - 588x^2$. Calculando $418 + 169x - 588x^2$ $(mod\ 7)$, segue que o quociente é $-84x^2 + 24x + 59$ e o resto é $x+5$. Como $(x-2)^7 \equiv (x^7-2)\ mod(x^3-1,7)$, r é primo e r não divide p, segue que

$$p^2 \equiv 1\ (mod\ r) \Rightarrow 7^2 \equiv 1\ (mod\ 3).$$

Portanto, a equação $p^2 \equiv 1\ (mod\ r)$ é válida e 7 é primo.

Agora, verifiquemos que 4 não é primo. Seja $p = 4, a = 2$ e $r = 3$. Pela equação $(x-a)^p \equiv (x^p - a)\ mod(x^r - 1, p)$, segue que o quociente de $(x^4 - 2)\ mod(x^3 - 1)$ é igual a x, e o resto é igual a $x - 2$. Calculando $x - 2\ (mod\ 4)$, segue que o quociente é -1 e o resto é $x + 2$. Pela expansão binomial seque que $(x^4 - 8x^3 + 24x^2 - 32x + 16)\ mod(x^3 - 1)$, resultando no quociente $x - 8$ e resto $24x^2 - 31x + 16$. Calculando $(24x^2 - 31x + 16)\ (mod\ 4)$, segue que o quociente é $6x^2 - 7x + 2$ e o resto é $-3x$. Portanto, $(x-2)^4 \not\equiv (x^4 - 2)\ mod(x^3 - 1, 4)$, e 4 é composto.

Conclusão

Nesse capítulo construímos o conhecimento necessário para estudar com mais detalhes a criptografia RSA e ElGamal, que será vista no próximo capítulo.

Também vimos os principais testes de primalidade, sendo que apesar das várias técnicas desenvolvidas por matemáticos ao longo dos séculos, não existe uma função que gera todos os primos, o que é de suma importância para a segurança do algoritmo RSA, por exemplo. É importante entender esses métodos para que se entenda a ideia de segurança que o RSA proporciona.

3 Criptografia

A chave pública consiste em um algoritmo que todos os envolvidos tem acesso, é a responsável pela codificação dos dados. Essa chave deve possuir funções de difícil inversão, uma vez que é isso que garante o sigilo de informações, pois dessa forma, outras pessoas que não estão autorizadas, não conseguem obter a chave pública, que é confidencial, e assim ter acesso as informações criptografadas. Neste capítulo utilizamos as referências [4], [17], [18], [19] e [20].

A seguir, uma ilustração de como funciona a criptografia de chave pública:

3.1 Conceitos e terminologia básica

A seguir, apresentamos uma lista de termos e conceitos básicos usados ao longo dessa dissertação.

- Denotamos por A um conjunto finito chamado de *alfabeto de definição*. Por exemplo, $A = \{0, 1\}$, o *alfabeto binário*, é um exemplo de alfabeto de definição frequentemente usado. Note que qualquer alfabeto pode ser codificado em termos do alfabeto binário.

- M denota um conjunto chamado de *espaço mensagem*. M consiste de strings de símbolos de um alfabeto de definição. Um elemento de M é chamado de mensagem de texto original. Por exemplo, M pode consistir de strings binárias, códigos de computadores, etc.

- C denota um conjunto chamado *espaço do texto cifrado*. C consiste de strings de símbolos de um alfabeto de definição, que pode ser diferente do alfabeto de definição para M. Um elemento de C é chamado de texto cifrado.

Transformações de encriptação e decriptação

• K denota um conjunto chamado de *espaço chave*. Um elemento de K é chamado uma *chave*.

• Cada elemento $e \in K$ determina unicamente uma bijeção de M a C, denotado por E_e. E_e é chamada uma *função de encriptação*, ou uma *transformação de encriptação*. Note que E_e precisa ser uma bijeção se o processo precisa ser revertido e uma única mensagem original é recuperada para cada texto cifrado distinto.

• Para cada $d \in K$, D_d denota uma bijeção de C para M (i.e. $D_d : C \to M$). D_d é chamada uma *função de decriptação* ou *transformação de decriptação*.

• O processo de aplicação da transformação E_e em uma mensagem $m \in M$ é normalmente referido como *encriptando* m ou *encriptação de* m.

• O processo de aplicação da transformação D_d em um texto cifrado c é normalmente referido como *decriptando* m ou *decriptação de* m.

• Um *esquema de encriptação* consiste de um conjunto $\{E_e : e \in K\}$ de transformações de encriptação e um conjunto correspondente $\{D_d : d \in K\}$ de transformações de decriptação com a propriedade que para cada $e \in K$ existe uma única chave $d \in K$ tal que $D_d = E_e^{-1}$; isto é, $D_d(E_e(m)) = m$ para todo $m \in M$. Um esquema de encriptação é alguma vezes referido como uma cifra.

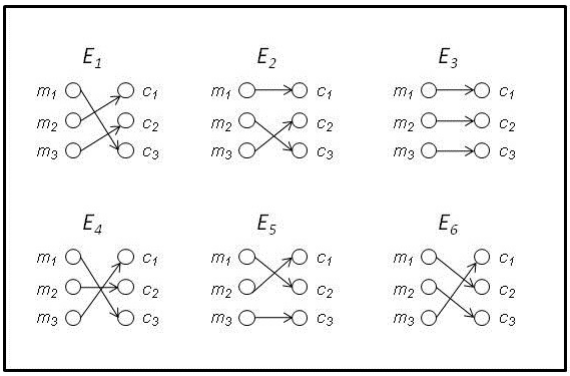
• As chaves e e d citadas anteriormente são referidas como um par de chaves e algumas vezes denotada por (e, d). Note que e e d podem ser iguais.

• *Construir* um esquema de encriptação requer selecionar um espaço de mensagem M, um espaço de texto cifrado C, um espaço de chave K, um conjunto de transformações de encriptação $\{E_e : e \in K\}$, e um conjunto de transformações de decriptação correspondente $\{D_d : d \in K\}$.

Um esquema de encriptação pode ser usado com o propósito de alcançar confiabilidade. Por exemplo, suponha que Alice e Bob primeiramente escolhem secretamente um par de chaves (e, d). Se Alice quiser enviar uma mensagem $m \in M$ para Bob, ela

calcula $c = E_e(m)$ e transmite para Bob. Ao receber c, Bob calcula $D_d(c) = m$ e por isso recupera a mensagem original m.

Exemplo 3.1.1. (Esquema de encriptação) Seja $M = \{m_1, m_2, m_3\}$ e $C = \{c_1, c_2, c_3\}$. Existem exatamente $3! = 6$ bijeções de M para C. A chave $K = \{1, 2, 3, 4, 5, 6\}$ possui 6 elementos, cada um especificando uma das transformações. A figura abaixo ilustra os seis elementos de encriptação que foram denotados por E_i, com $1 \leq i \leq 6$.



Alice e Bob concordam na transmissão, por exemplo E_1. Para encriptar a mensagem m_1, Alice calcula $E_1(m_1) = c_3$ e envia c_3 para Bob. Bob decripta c_3 revertendo as flechas do diagrama para E_1 e observa que c_3 aponta para m_1.

Criptografia simétrica e assimétrica

Criptografia simétrica: possui apenas uma chave e essa é compartilhada entre as pessoas da conexão. Em geral não são usadas em autenticações porém, em sua maioria são usadas para confidencialidade, ou seja, usadas para transportar informações em sigilo. O seu funcionamento é bem mais rápido do que um método assimétrico se comparado com um do mesmo nível. Em geral a chave é única para cifrar e decifrar.
Passos:

1. Depois de ambos conhecerem a chave K, acontece o seguinte;
2. Alice encripta sua mensagem $M1$, $Y = E(M1)$;
3. Alice envia Y para Bob;
4. Bob decripta Y e encontra $M1$, $M1 = D(Y)$;

   Mas isso só acontece porque K é conhecida por ambos.

Criptografia Assimétrica: é implementada com duas chaves para cada usuário, uma chave pública e uma chave privada. A chave pública é conhecida por todos na conexão, a chave privada é de conhecimento apenas pelo dono. Vale salientar que computacionalmente é lenta se comparada com um método simétrico de mesmo nível. A vantagem é que pode ser usada em autenticação e confidencialidade.
Passos:

1. Cada usuário cria suas chaves tanto a pública quanto a privada;
2. Alice envia a sua chave pública para Bob;
3. Bob criptografa a mensagem com a chave pública de Alice, ela recebe a mensagem encriptada e decripta com sua chave privada.

3.2 Diffie-Hellman

Nesta seção, apresentamos o funcionamento do protocolo Diffie-Hellman, que propôs uma maneira de transmissão de mensagens confidenciais através de chaves assimétricas, dessa forma esse protocolo apresenta um nível de segurança maior comparado a um protocolo com criptografia de chave simétrica.

O protocolo Diffie-Hellman foi criado em 1975 por Whitfield Diffie e Martin Hellman, sendo publicado em 1976, e consiste em um protocolo para troca de chaves usando meios inseguros.

A seguir, exibiremos os passos para a aplicação desse protocolo.

• Dado p primo e a uma raiz primitiva módulo p, ambos conhecidos pelos participantes da conexão, nesse caso Alice e Bob;

• Alice e Bob geram números aleatórios X_a e X_b, respectivamente, sendo que X_a e X_b são menores que p. Esses números gerados são as chaves privadas;

• Alice e Bob calculam as senhas públicas $Y_a \equiv a^{X_a}(mod\ p)$ e $Y_b \equiv a^{X_b}(mod\ p)$, respectivamente;

• Alice e Bob trocam as senhas (números) públicas;

• Alice calcula:

$$k \equiv Y_b^{X_a}(mod\ p) \Longrightarrow k \equiv a^{X_b X_a}(mod\ p).$$

• Bob calcula:

$$k \equiv Y_a^{X_b}(mod\ p) \Longrightarrow k \equiv a^{X_a X_b}(mod\ p).$$

• Desta forma eles possuem a mesma chave secreta k.

Exemplo 3.2.1. Alice e Bob encriptarão uma mensagem utilizando o protocolo Diffie-Hellman:

• Se $p = 97$ então 5 será uma raiz primitiva de 97, pois $5^{96} \equiv 1\ (mod\ 97)$.

• Alice e Bob geram seus números secretos $X_a = 36$ e $X_b = 58$, respectivamente.

• Alice e Bob calculam os números públicos Y_a e Y_b, respectivamente.

• Alice calcula:

$$Y_a \equiv a^{X_a}(mod\ p) \Longrightarrow Y_a \equiv 5^{36}(mod\ 97),$$

e obtém $Y_a = 50$, isso acontece pelo fato de que o algoritmo sempre usa o menor Y_a positivo, ou seja, o resto da divisão de 5^{36} por 97.

- Bob calcula:

$$Y_b \equiv a^{X_b}(mod\ p) \Longrightarrow Y_b \equiv 5^{58}(mod\ 97),$$

e obtém $Y_b = 44$.

- Bob envia Y_b para Alice, e Alice envia Y_a para Bob.
- Alice calcula:

$$k \equiv Y_b^{X_a}(mod\ p) \Longrightarrow k \equiv 44^{36}(mod\ 97),$$

e obtém $k = 75$.

- Bob calcula:

$$k \equiv Y_a^{X_b}(mod\ p) \Longrightarrow k \equiv 50^{58}(mod\ 97),$$

e obtém $k = 75$.

- Logo, a chave simétrica deles será $k = 75$.

A versão original do protocolo é vulnerável a um ataque conhecido como *"man-in-the-middle"*, que consiste em uma técnica onde o invasor se interpõe entre os computadores que se comunicam sem que ambos saibam da invasão. Para evitar esse tipo de ataque, em 1992 o protocolo *STS* foi desenvolvido por Diffie, Oorschot e Wiener para não sofrer mais o ataque *"man-in-the-middle"*. Isto possibilitou que ambas as partes, remetente e receptor, autenticassem suas mensagens a partir de assinaturas digitais, que veremos nesse trabalho, especialmente para o caso *RSA*. A ideia é que antes de iniciar o protocolo, remetente e destinatário obtenham um par de chaves públicas e privadas, e durante o protocolo é calculado a assinatura em algumas mensagens.

3.3 Criptografia RSA

RSA é um algoritmo de criptografia de dados, que foi inventado em 1978 por Ronald Rivest, Adi Shamir e Leonard Adleman, que na época trabalhavam no *Massachussets Institute of Technology* (MIT).

O algoritmo RSA é do tipo assimétrico, ou seja, possui uma chave pública e uma privada, e o que garante sua segurança é o fato desse algoritmo ser baseado em primos com números de algarismos acima de 100. Como já visto, não temos um algoritmo eficiente para a fatoração de inteiros em primos, o que torna o tempo de decriptação de uma mensagem criptografada utilizando o algoritmo RSA, na maioria das vezes, inviável.

3.3.1 Texto cifrado

Para utilizarmos o método de criptografia RSA devemos converter uma mensagem em uma sequência de números.

Para convertermos a mensagem em símbolos ou números (sendo a conversão comumente feita para números), devemos escolher uma boa chave para a conversão a maneira como é escolhida a chave será vista a seguir. Como último passo, o número formado com a conversão deve ser separado em blocos de números menores que n.

Observações

• A maneira de se escolher os blocos não precisa ser homogênea (cada bloco pode possuir um número diferente de dígitos).

• Geralmente a tabela de conversão utilizada possui para cada algarismo da mensagem dois ou mais dígitos (por exemplo, a letra A é representada pelo número 10, a letra B, pelo número 11 e assim sucessivamente) para que se evite possíveis ambiguidades.

• Não podemos começar a sequência com um bloco de zeros, pois isso traria problemas na hora da decriptação.

Exemplo 3.3.1. Vamos utilizar a tabela de conversão abaixo (não diferenciando letras maiúsculas de minúsculas), para cifrar a frase: *Ana come biscoito.*

a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t
10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29

u	v	w	x	y	z	
30	31	32	33	34	35	36

Assim temos o texto cifrado:

$$10231036122422143611182812241829 24.$$

3.3.2 Geração das chaves

A seguir, vamos apresentar todos os passos do processo de geração de chaves.

(i) Escolhemos dois números primos distintos p e q.

(ii) Calculamos $n = pq$.

Exemplo 3.3.2. Os números primos utilizados para esse exemplo são considerados pequenos, o que deixa a segurança do RSA quase nula, portanto é utilizado nesse trabalho apenas para fins de exemplificação. Seja $p = 5$ e $q = 11$, portanto $n = 55$.

3.3.3 Encriptação

Para encriptarmos uma mensagem, usamos o seguinte Teorema 2.3.3 e a Proposição 2.3.2

(iii) Iniciamos com o cálculo da função de Euler para n. Sejam:

$$\phi(p) = p^k - p^{k-1},$$
$$\phi(q) = q^k - q^{k-1},$$
$$\phi(n) = \phi(pq) = (p^k - p^{k-1})(q^k - q^{k-1}).$$

Como p e q são números primos, segue que as equações podem ser simplificadas por:

$$\phi(p) = p^1 - p^0 = p - 1,$$
$$\phi(q) = q^1 - q^0 = q - 1,$$
$$\phi(n) = \phi(pq) = (p^1 - p^0)(q^1 - q^0) = (p-1)(q-1).$$

Desta forma, calculamos $\phi(n) = (p-1)(q-1)$;

(iv) Escolhemos um inteiro positivo e tal que $1 < e < \phi(n)$ e $mdc(\phi(n), e) = 1$. Denominamos e como *componente de enciframento*;

(v) A *chave pública de encriptação do sistema RSA* é dada pelo par de números (n, e);

(vi) Em seguida é encriptado cada bloco obtido do texto cifrado.

(vii) A encriptação de um bloco b será denotada por $C(b)$, que é dada por:

$$C(b) \equiv b^e \ (mod\ n).$$

Exemplo 3.3.3. Vamos encriptar a mensagem do Exemplo 3.3.1. Primeiramente, vamos obter a função $\phi(n)$, ou seja,

$$\phi(n) = (p-1)(q-1) \Longrightarrow \phi(n) = 4 \cdot 10 = 40.$$

Como conhecemos o valor de $\phi(n)$, segue que podemos escolher um componente de enciframento e tal que $1 < e < 40$ sendo que $mdc(40, e) = 1$. Seja $e = 3$. Desta forma, a chave pública de encriptação é dada por $(55, 3)$.

Sendo assim, podemos dividir a mensagem em blocos:

$$10 - 23 - 10 - 36 - 12 - 24 - 22 - 14 - 36 - 11 - 18 - 28 - 12 - 24 - 18 - 29 - 24.$$

Lembrando que os blocos não precisam ser divididos com o mesmo número de caracteres. A seguir, vamos encriptar cada bloco:

$$
\begin{aligned}
\text{Bloco } 10 &\longrightarrow (10)^3 \equiv 10 \ (mod\ 55)\\
\text{Bloco } 23 &\longrightarrow (23)^3 \equiv 12 \ (mod\ 55)\\
\text{Bloco } 10 &\longrightarrow (10)^3 \equiv 10 \ (mod\ 55)\\
\text{Bloco } 36 &\longrightarrow (36)^3 \equiv 16 \ (mod\ 55)\\
\text{Bloco } 12 &\longrightarrow (12)^3 \equiv 23 \ (mod\ 55)\\
\text{Bloco } 24 &\longrightarrow (24)^3 \equiv 19 \ (mod\ 55)\\
\text{Bloco } 22 &\longrightarrow (22)^3 \equiv 33 \ (mod\ 55)\\
\text{Bloco } 14 &\longrightarrow (14)^3 \equiv 49 \ (mod\ 55)\\
\text{Bloco } 36 &\longrightarrow (36)^3 \equiv 16 \ (mod\ 55)\\
\text{Bloco } 11 &\longrightarrow (11)^3 \equiv 11 \ (mod\ 55)\\
\text{Bloco } 18 &\longrightarrow (18)^3 \equiv 2 \ (mod\ 55)\\
\text{Bloco } 28 &\longrightarrow (28)^3 \equiv 7 \ (mod\ 55)\\
\text{Bloco } 12 &\longrightarrow (12)^3 \equiv 23 \ (mod\ 55)\\
\text{Bloco } 24 &\longrightarrow (24)^3 \equiv 19 \ (mod\ 55)\\
\text{Bloco } 18 &\longrightarrow (18)^3 \equiv 2 \ (mod\ 55)\\
\text{Bloco } 29 &\longrightarrow (29)^3 \equiv 24 \ (mod\ 55)\\
\text{Bloco } 24 &\longrightarrow (24)^3 \equiv 19 \ (mod\ 55)
\end{aligned}
$$

O resultado é a mensagem encriptada:

$$10121016231933491611272319224 19.$$

3.3.4 Decriptação

Para decriptarmos a mensagem, necessitamos do número n e do inverso de e módulo $\phi(n)$, lembrando que ϕ é a função de Euler. Chamaremos esse número de d. Assim,

$$ed \equiv 1 \ (mod\ \phi(n)).$$

Para determinar d, basta resolver a equação $ed + x\phi(n) = 1$ através do *algoritmo euclidiano estendido* 2.1.3.

Se $a = C(b)$ é um bloco de mensagem codificada, então $D(a)$ é o resultado da decriptação, que é dada por:

$$D(a) \equiv a^d \ (mod\ n).$$

O par (n, d) é denominado *chave privada de decriptação do sistema RSA*. Ao decriptar os blocos de mensagens codificadas, é encontrada a mensagem original, ou seja, $D(C(b)) = b$. Note que para decriptarmos a mensagem, não é necessário sabermos os valores de p e q, apenas os valores de n e d.

Valor de d negativo

Quando o valor de d é negativo, é utilizado o *Teorema Geral de uma Equação Diofantina* 2.2.3 para encontrar um valor positivo para d.

Desta forma, para reverter valores negativos de d, escolhemos $t \in \mathbb{Z}$ tal que $d + \phi(n)t > 0$. Assim, $\overline{d} = d + \phi(n)t$.

Exemplo 3.3.4. Vamos decriptar a mensagem do Exemplo 3.3.3. Para decriptarmos a mensagem, precisamos encontrar d que satisfaz

$$ed \equiv 1 \ mod(\phi(n)) \rightarrow 3d \equiv 1 \ mod(40).$$

Isto é equivalente a resolvermos a seguinte equação:

$$40x + 3d = 1.$$

Pelo algoritmo euclidiano estendido, obtemos

Resto	Quociente	x	d
40	$*$	1	0
3	$*$	0	1
1	13	$1 - 13 \cdot 0 = 1$	$0 - 13 \cdot 1 = -13$
0	3	$*$	$*$

Desta forma, $40 \cdot 1 + 3 \cdot (-13) = 1 \Longrightarrow x = 1$ e $d = -13$. Como d é negativo, segue que

$$\begin{aligned} &\overline{d} = -13 + 40t > 0, \\ &\overline{d} = -13 + 80 = 67 \text{ (para } t = 2) \\ &\overline{d} = d = 67 \end{aligned}$$

Após encontrado o valor de d, positivo, continuamos o processo de decriptação. Assim, seja $a = C(b)$ um bloco da mensagem codificada. Temos que encontrar $D(a)$ tal que

$$D(a) \equiv a^d \ (mod \ n).$$

A seguir, vamos decriptar cada bloco:

$$
\begin{aligned}
\text{Bloco } 10 &\longrightarrow (10)^{67} \equiv 10 \ (mod\ 55)\\
\text{Bloco } 12 &\longrightarrow (12)^{67} \equiv 23 \ (mod\ 55)\\
\text{Bloco } 10 &\longrightarrow (10)^{67} \equiv 10 \ (mod\ 55)\\
\text{Bloco } 16 &\longrightarrow (16)^{67} \equiv 36 \ (mod\ 55)\\
\text{Bloco } 23 &\longrightarrow (23)^{67} \equiv 12 \ (mod\ 55)\\
\text{Bloco } 19 &\longrightarrow (19)^{67} \equiv 24 \ (mod\ 55)\\
\text{Bloco } 33 &\longrightarrow (33)^{67} \equiv 22 \ (mod\ 55)\\
\text{Bloco } 49 &\longrightarrow (49)^{67} \equiv 14 \ (mod\ 55)\\
\text{Bloco } 16 &\longrightarrow (16)^{67} \equiv 36 \ (mod\ 55)\\
\text{Bloco } 11 &\longrightarrow (11)^{67} \equiv 11 \ (mod\ 55)\\
\text{Bloco } 2 &\longrightarrow (2)^{67} \equiv 18 \ (mod\ 55)\\
\text{Bloco } 7 &\longrightarrow (7)^{67} \equiv 28 \ (mod\ 55)\\
\text{Bloco } 23 &\longrightarrow (23)^{67} \equiv 12 \ (mod\ 55)\\
\text{Bloco } 19 &\longrightarrow (19)^{67} \equiv 24 \ (mod\ 55)\\
\text{Bloco } 2 &\longrightarrow (2)^{67} \equiv 18 \ (mod\ 55)\\
\text{Bloco } 24 &\longrightarrow (24)^{67} \equiv 29 \ (mod\ 55)\\
\text{Bloco } 19 &\longrightarrow (19)^{67} \equiv 24 \ (mod\ 55)
\end{aligned}
$$

Logo, a mensagem decodificada fica da forma 10231036122422143611182812241829 24, que é a mensagem original, conforme pode ser visto no início do Exemplo 4.2.1.

Funcionalidade do sistema

Para mostrar que o sistema visto acima realmente funciona, precisamos verificar que se $C(b)$ é um inteiro e $1 \leq b < n$, então $D(C(b)) = b$.

Como $D(C(b))$ e b estão no intervalo de 1 até $n-1$, segue que b e $D(C(b))$ somente serão congruentes módulo n, se forem iguais. Logo, é suficiente mostrarmos que $D(C(b)) \equiv b \ (mod\ n)$. Por definição, segue que

$$D(C(b)) \equiv (b^e)^d \equiv b^{ed} \ (mod\ n).$$

Como $n = pq$, vamos calcular b^{ed} módulo p e módulo q. Calculemos b^{ed} módulo p. Sendo d o inverso de e módulo $\phi(n)$, segue que

$$ed = 1 + k\phi(n) = 1 + k(p-1)(q-1) \Rightarrow (b^e)^d \equiv b(b^{p-1})^{k(q-1)} \ (mod\ p).$$

Utilizaremos agora o *Pequeno Teorema de Fermat* 2.4.3. Para isso, vamos supor que p não divide b. Se isso acontece, então $b^{p-1} \equiv 1 \ (mod\ p)$, ou seja, $b^{ed} \equiv b \ (mod\ p)$. Assim, podemos afirmar que $b^{ed} - b$ é divisível por p, e analogamente, chega-se ao mesmo resultado para q. Como p e q são primos distintos, segue que $mdc(p, q) = 1$, e assim,

$$pq|(b^{ed} - b) \Longrightarrow b^{ed} \equiv b \ (mod\ n),$$

para qualquer b inteiro, pois $n = pq$. Logo, $D(C(b)) = b$.

3.3.5 Assinatura digital

O método RSA pode ser utilizado para a verificação de autenticidade de uma mensagem enviada, esse método consiste em uma assinatura eletrônica.

Para que seja implementada essa assinatura digital, o emissor deve possuir a chave privada, e poderá assinar uma mensagem (em blocos). Sendo b um desses blocos, para assiná-lo, o emissor utiliza a chave privada para encriptar a mensagem, e em seguida, a chave pública do receptor para encriptar a mensagem. Ao receber a mensagem, o receptor utilizará sua própria chave privada, e em seguida, ele utiliza a chave pública do emissor para decriptar a mensagem.

Exemplo 3.3.5. Antônio deseja enviar um bloco contendo informações bancárias confidenciais para um banco utilizando a internet. Para sua segurança ele deseja que a mensagem contenha uma assinatura digital. Segundo o método acima, chamamos de C_a e D_a as funções de encriptação e decriptação, respectivamente, de Antônio, e de C_b e D_b as do banco. Para que a mensagem de Antônio, digamos m, possua assinatura digital, a mensagem, que codificada seria da forma $C_a(m)$, na realidade será da forma $C_b(D_a(m))$. Note que primeiramente foi utilizada a função de decriptação de Antônio, e em seguida, a função de encriptação do banco. Para que o banco decodifique a mensagem, primeiro utiliza sua função de decriptação, ou seja, $D_b(C_b(D_a(m)))$, resultando $D_a(m)$, e após isso basta utilizar a função de encriptação de Antônio, que é pública para decriptar a mensagem, isto é, $C_a(D_a(m)) = m$.

3.3.6 Segurança do RSA

A segurança do RSA está baseada no fato de ser extremamente difícil a fatoração de n, se n for grande, pois para "quebrar" a segurança do RSA, é necessário determinar d a partir de n e e, pois para encontrar d é utilizado $\phi(n)$ e e, mas para determinar $\phi(n)$ é necessário ter os valores de p e q, ou seja, n fatorado.

Outra forma de se tentar quebrar a segurança do RSA é tentar determinar o valor de b a partir de $C(b) \equiv b^{ed}\ (mod\ n)$, o que é praticamente impossível sem saber o valor de d.

Observação: 3.3.1. Deve-se tomar alguns cuidados com relação a escolha de p e q, pois se estes forem relativamente pequenos, torna-se fácil determiná-los, ou ainda, se $|p - q|$ é um valor pequeno, também torna-se fácil determinar p e q, tornando-se fatorável pelo algoritmo de Fermat.

Outro método para a tentativa de "quebra de segurança" do algoritmo é analisando, o tempo que um servidor demora para reconhecer se um usuário é legítimo ou não. Esse tipo de resposta diz muito sobre o tamanho de n. Em geral, esse método só tem sucesso se a programação da assinatura digital possuir falhas.

3.4 A Criptografia de ElGamal

A criptografia de ElGamal é um sistema que utiliza chaves assimétricas que foi criado por *Taher ElGamal* em 1984. Sua segurança é baseada na dificuldade de resolução do problema do logaritmo discreto. É frequentemente utilizado em assinaturas digitais.

A geração das Chaves

Para gerar a chaves da criptografia ElGamal, são necessários:

(i) Escolher um número primo (preferencialmente grande), e um gerador α do grupo multiplicativo $\mathbb{Z}_p^*$;

(ii) Selecionar aleatoriamente um número natural $n < p-1$ e calcular $\alpha^n \ (mod\ p)$;

(iii) A chave pública é dada por (p, α, α^n) e a chave privada é dada por n.

Encriptando uma mensagem

Nessa etapa, o emissor A deve:

(iv) Obter a chave pública (p, α, α^n) do receptor B.

(v) Converter as letras, números e símbolos da mensagem em números m, com m entre 0 e $p-1$.

(vi) Escolher aleatoriamente um número natural a tal que $a < p-1$.

(vii) Para cada m, calcular

$$\beta \equiv \alpha^a \ (mod\ p) \quad \text{e} \quad \gamma \equiv m(\alpha^n)^a \ (mod\ p).$$

(viii) Enviar a encriptação $c = (\beta, \gamma)$ para B.

Decriptação

Ao receber a mensagem codificada c, o receptor B deve:

(ix) Utilizar a chave privada para calcular

$$\beta^{p-1-n} \ (mod\ p);$$

(x) decriptar m calculando $\beta^{-n}\gamma \ (mod\ p)$;

(xi) Devido ao teorema de Fermat, segue que

$$\beta^{-n}\gamma \equiv \alpha^{-an} m \alpha^{an} \equiv m \ (mod\ p).$$

Exemplo 3.4.1. Utilizando a tabela do Exemplo 3.3.1, vamos encriptar e decriptar a palavra "oi". Primeiramente, vamos escolher um número primo $p = 29$ (lembrando que esse número considerado pequeno foi escolhido apenas para fins de exemplo de como o algoritmo funciona). Escolheremos agora um gerador $\alpha = 2$ do grupo multiplicativo $\mathbb{Z}_{29}^*$. Em seguida, o destinatário deve escolher a chave privada $n = 8$ e calculamos $2^8 \ (mod\ 29) = 256 \ (mod\ 29)$, resultando que $\alpha^n = 24$. Com isto, obtemos a chave pública $(p, \alpha, \alpha^n) = (29, 2, 24)$.

Encriptando a mensagem

Para encriptarmos a mensagem, o emissor deve obter a chave pública, no caso (29, 2, 24), do receptor B. O primeiro passo é converter as letras em símbolos. Utilizando a tabela do Exemplo 3.3.1, obtemos 2418. Vamos separar esse número em $m_1 = 24$ e $m_2 = 18$, lembrando que cada um desses m's deve estar entre 0 e $p-1$. Em seguida, vamos escolher um número natural $a = 5$, e para cada m_i calcular:

$$\begin{aligned} \beta \equiv \alpha^a \ (mod\ p) = 24^5 \ (mod\ 29) &\implies \beta = 7 \\ \gamma_1 \equiv m_1(\alpha^n)^a \ (mod\ p) = 24(24)^5 \ (mod\ 29) &\implies \gamma_1 = 23 \\ \gamma_2 \equiv m_2(\alpha^n)^a \ (mod\ p) = 18(24)^5 \ (mod\ 29) &\implies \gamma_2 = 10 \end{aligned}$$

Desta forma, a mensagem está codificada, sendo que $c_1 = (7, 23)$ e $c_2 = (7, 10)$ serão enviados para o receptor.

Decriptando a mensagem

Para decriptar a mensagem, o receptor utiliza a chave privada para calcular $\beta^{p-1-n} \ (mod\ p) = 7^{29-1-8} \ (mod\ 29) = 7^{20} \ (mod\ 29)$. Logo, $\beta^{p-1-n} = 25$. Em seguida, calcula

$$\begin{aligned} m_1 = \beta^{-n}\gamma \ (mod\ p) = 2523 \ (mod\ 29) = 24 \\ m_2 = \beta^{-n}\gamma \ (mod\ p) = 2510 \ (mod\ 29) = 18 \end{aligned}$$

resultando na mensagem enviada 2418.

Conclusão

Nesse capítulo apresentamos alguns dos mais utilizados algoritmos de criptografia da atualidade. Desde transações bancárias até o envio de emails, são utilizados esses, ou algoritmos similares.

O método de criptografia, conhecido como o método das chaves assimétricas, revolucionou a criptografia, e como vimos, até hoje ele é utilizado, pois apesar dos esforços da criptoanálise, essas criptografias ainda não foram quebradas de maneira efetiva, devido ao problema da fatoração de primos, no caso da RSA, e na fatoração do logaritmo discreto, no caso da ElGamal.

4 Criptografia quântica e pós-quântica

A segurança dos métodos atuais se baseia em problemas computacionalmente difíceis, proporcionando um bom nível de segurança. Todavia, caso haja um grande crescimento do poder computacional, onde destacamos pesquisas em computadores quânticos, poderão ocorrer brechas de segurança na correspondência criptografada pelos métodos usados na atualidade. Através dessa constatação, diversos centros de pesquisa vem estudando métodos alternativos que possam, num futuro próximo, substituir as técnicas existentes do poder de computação disponível. Uma técnica muito promissora é a criptografia quântica, que daremos uma breve descrição nesse capítulo.

A criptografia pós-quântica é um ramo da criptografia que estuda classes de algoritmos resistentes a criptografia quântica. Alguns exemplos dessas classes de algoritmos são os baseados em reticulados e em códigos corretores de erros, que serão abordados nesse capítulo. Para o desenvolvimento deste capítulo, as referências utilizadas foram [21], [26] [22] e [23].

4.1 Criptografia quântica

A criptografia quântica utiliza técnicas de segurança que se devem a fenômenos quânticos. Baseando-se nos princípios da Mecânica Quântica, a grande vantagem deste método em relação aos outros reside na segurança incondicional, ou seja, não apresenta falhas como os métodos criptográficos atuais, e não pode ser quebrado, mesmo com poderosos computadores.

A distribuição de chaves quânticas tem o objetivo de gerar e distribuir chaves secretas, normalmente para que duas pessoas as usem. A comunicação durante o processo de criação da chave é dividido em duas partes, sendo que parte dessa comunicação é feita através de um canal clássico, e a outra através de um canal quântico. As mensagens enviadas através do último canal mencionado tem vantagem com relação as mensagens enviadas através do canal clássico, pois é impossível medir integralmente ou copiar uma função de onda. Desta forma, a criptografia quântica torna viável o uso de protocolos onde é impossível a interceptação de uma mensagem sem que o intruso seja percebido,

diferentemente da criptografia clássica.

Dentre suas aplicações, a mais utilizada é em *distribuição de chaves quânticas* [25], mas há outros tipos como o *"cara ou coroa"* quântico, que abordaremos a seguir.

4.1.1 "Cara ou coroa" quântico

Imagine que duas pessoas pessoas vão jogar "cara ou coroa" à distância e, é necessário garantir, sem a ajuda de uma terceira pessoa, que ambas não vão trapacear. Para isto utilizamos um protocolo quântico que simula esse "jogo" da seguinte forma:

Sejam Alice e Bob as pessoas que vão jogar. Desta forma, Alice escolhe uma base de polarização (retilínea ou diagonal) aleatoriamente e uma sequência de bits. Em seguida, envia para Bob fótons codificando esses bits na base que foi escolhida. Ao receber os fótons codificados, Bob escolherá, de maneira aleatória, uma base para fazer a leitura de cada fóton que Alice enviou, e ao final da decriptação, Bob dirá, aleatoriamente, qual base que Alice escolheu. Alice irá dizer se os palpites de Bob estão corretos ou não, e para garantir que não haverá qualquer tipo de trapaça de sua parte, Alice envia, através de um canal público clássico, a sequência de bits escolhida. Bob, por sua vez, verifica se os bits lidos na base enviada por Alice foram iguais aos enviados no canal clássico para garantir que a sequência de bits enviada não foi alterada.

Caso Alice ainda tentasse trapacear enviando cada bit em uma base aleatória, não teria como responder para Bob uma base e uma sequência de bits de forma que todos os bits detectados por Bob no canal quântico sejam iguais aos bits enviados no canal clássico nessa mesma base. Bob, para trapacear, teria que opinar qual base Alice usa, e para isso teria que ter uma chance maior que 50% de acerto, o que é impossível.

Podemos aplicar esse protocolo em certificações de cartas de e-mail, e na solução do uso da criptografia para garantir, sem a necessidade de uma terceira pessoa, que em um jogo de pôquer a distância, nenhum dos jogadores trapaceie.

A preocupação com a quebra de segurança dos criptossistemas atuais aumenta pelo fato de que podemos, em breve, ter computadores quânticos produzidos em larga escala, uma vez que os estudos nessa área estão cada vez mais avançados.

4.1.2 Dificuldades da utilização da criptografia quântica

Apesar de já ser possível utilizar a criptografia quântica, a tecnologia disponível não consegue fornecer canais quânticos com comprimento suficiente para fazer qualquer tipo de conexão (o máximo que se atinge são algumas centenas de quilômetros). A solução encontrada seria conseguir fazer algum tipo de transmissor quântico que retransmitiria o sinal em intervalos de distâncias pré-definidos. No entanto não se conseguiu fazer isso ainda.

Desde que se descobriu o potencial de criptoanálise do computador quântico, especialmente com o algoritmo de Shor, criou-se a ideia de que a criptografia moderna se

tornaria inútil a partir do momento que esse tipo de computador começasse a ser desenvolvido em larga escala. Logo, todas as mensagens trocadas com o uso de algoritmos criptográficos atuais seriam facilmente interceptáveis.

Em resposta a isso, surgiu a criptografia pós-quântica, ramo da criptografia que estuda classes de algoritmos criptográficos resistentes à criptoanálise quântica, e que será abordada na próxima seção.

4.2 Criptografia pós-quântica

Nesta seção vamos estudar os sistemas criptográficos de McEliece e Niederreiter baseados em códigos corretores de erros e os sistemas criptográficos de Ajtai-Dwork e NTRU, baseados em reticulados.

4.2.1 Criptografia pós-quântica e teoria dos códigos

Apresentaremos aqui os sistemas criptográficos de McEliece e Niederreiter, que surgiram no final da década de 70, um ano depois do algoritmo RSA. Ambos os criptossistemas são baseados na síndrome dos códigos corretores de erros que visa corrigir ou recuperar informações que foram perdidas ou alteradas em uma transmissão. Mesmo supondo a utilização de computadores quânticos, apenas em 2011, foi provado que estes sistemas são resistentes a ataques que quebram os sistemas atuais.

Códigos corretores de erros

Os códigos corretores de erros apresentam um importante papel nas comunicações, uma vez que praticamente todos os sistemas de envio de informações possui algum tipo de código corretor de erro, como por exemplo, a telefonia digital, transmissão de dados via satélite, a digitalização de música e a segurança de sistemas criptográficos, como veremos a seguir.

Definição 4.2.1. Um (n, k)-**código linear** sobre um corpo finito $\mathbf{F}_q$ é um subespaço vetorial k-dimensional de um espaço linear n-dimensional $\mathbf{F}_q^n$. Dizemos que n é o comprimento do código, e que k é sua dimensão.

Dependendo da estrutura algébrica do corpo utilizado para sua construção, cada código possuirá uma determinada capacidade de correção de erros.

Definição 4.2.2. Seja C um (n, k)-código linear sobre $\mathbf{F}_q$. Uma matriz G cujo espaço gerado pelas linhas seja igual a C é chamada de **matriz geradora** de C. Reciprocamente, se G é uma matriz com valores em $\mathbf{F}_q$, o espaço gerado por suas linhas é chamado de **código gerado** por G.

Definição 4.2.3. Uma **palavra-código** de um (n, k)-código linear C é um vetor n-dimensional pertencente a C.

Exemplo 4.2.1. Seja C um $(7,4)$ - código linear com a seguinte matriz geradora:

$$G = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{pmatrix}$$

Este código é chamado de *Código de Hamming*.

Um (n,k) - código linear possui q^k palavras-código, com isso existem q^k mensagens distintas. A encriptação de uma k-upla $u = (u_1, \cdots, u_k)$ consiste no mapeamento através da combinação linear das linhas da matriz G geradora do código, da seguinte forma:

$$u \to uG. \tag{4.1}$$

Assim, utilizando o código do Exemplo 4.2.1, é possível mapear vetores quaisquer de dimensão 4 em vetores de dimensão 7 que pertencem ao código, através da multiplicação de vetor por uma matriz da forma descrita em (4.1).

A recuperação da mensagem original através da palavra-código é feita através da utilização de uma matriz de paridade associada ao código. Para um determinado (n,k) - código linear C, uma verificação de paridade é uma equação da forma

$$a_1x_1 + \cdots + a_nx_n = 0$$

que é satisfeita para todo $x = (x_1, \cdots, x_n) \in C$. O conjunto de todos os vetores $a = (a_1, \cdots, a_n)$ para os quais a equação acima é satisfeita é também um subespaço vetorial de $\mathbf{F}_q^n$, chamado *código dual* de C e denotado por $C^\perp$. É possível mostrar que $C^\perp$ é um $(n, n-k)$ - código linear sobre $\mathbf{F}_q$, de forma que uma matriz de paridade para C pode ser definida como uma matriz geradora de $C^\perp$. De forma equivalente, temos a definição:

Definição 4.2.4. Seja C um (n,k) - código linear sobre $\mathbf{F}_q$. A matriz H com a propriedade de que $Hx^T = 0$ se, e somente se, $x \in C$ é chamada de **matriz de verificação de paridade para** C.

Supondo que os alfabetos das mensagens enviadas e recebidas por um certo canal ruidoso sejam os mesmos, se uma certa palavra-código $x = (x_1, \cdots, x_n) \in F_q^n$ é enviada, então a palavra recebida $y = (y_1, \cdots, y_n)$ também pertence a $\mathbf{F}_q^n$. A diferença $z = y - x$ é chamada de *padrão de erros* introduzido pelo canal na transmissão da mensagem, de forma que se $z_i \neq 0$, então dizemos que ocorreu um erro na posição i da palavra-código.

Definição 4.2.5. O **peso** de um vetor x (ou peso de Hamming) é o número de coordenadas não nulas de x, denotado por $wt(x)$.

Definição 4.2.6. O **peso mínimo** d de um código C é o menor peso existente em alguma palavra não totalmente nula (diferente de zero em alguma componente) desse código.

Exemplo 4.2.2. O vetor (1 0 0) tem peso 1, e vetor (1 1 1) tem peso 3.

Teorema 4.2.1 (Voloch 1987)**.** Um código linear binário, representado por (n, k, d), sendo n seu comprimento, k sua dimensão e d seu peso mínimo, pode corrigir até $t = \left\lfloor \frac{d-1}{2} \right\rfloor$ erros.

Exemplo 4.2.3. Seja um código de Hamming $(5, 3, 2)$. Neste caso, $n = 5, k = 3$ e $d = 2$. Logo, a mensagem codificada tem tamanho 5, as mensagens originais tem tamanho 3 e este código corrige $t = \lfloor (2-1)/2 \rfloor = 0$ erros.

Definição 4.2.7. O polinômio mônico $g \in F_{2^m}[X]$ de grau t, definido por: $g(X) = \sum_{i=0}^{t} g_i X^i$, com $g_t = 1$, é chamado de *polinômio de Goppa*.

Definição 4.2.8. Seja $L = (\gamma_1, \cdots, \gamma_{n-1}) \in F_{2^m}^n$ tal que nenhum dos γ_i seja raiz do polinômio de Goppa g, ou seja, $g(\gamma_i) \neq 0, \forall 0 \leq i < n$. Então, este conjunto é chamado *suporte de código*.

Definição 4.2.9. Para qualquer vetor $c = (c_0, \cdots, c_{n-1}) \in F_2^n$, definimos a *síndrome* de c como:

$$S_c(X) = -\sum_{i=0}^{n-1} \frac{c_i}{g(\gamma_i)} \frac{g(X) - g(\gamma_i)}{X - \gamma_i} \; mod \; g(X).$$

Definição 4.2.10. O *código binário de Goppa*, denotado por $\Gamma(L, g(X))$ sobre F_2 é o conjunto de todos os vetores $c = (c_0, \cdots, c_{n-1}) \in F_2^n$ tais que $S_c(X) = 0$ seja assegurado em $F_{2^m}[X]$. Se g for irredutível sobre F_{2^m}, então Γ é chamado *código de Goppa binário irredutível*, e como $g(\gamma) \neq 0, \forall \gamma \in F_{2^m}$, então L contém todos os elementos de F_{2^m}.

Teorema 4.2.2 (Singleton Bound - Voloch 1987)**.** Se C é um (n, k)-código linear de peso mínimo d sobre $\mathbf{F}_2$, então $d + k \leq n + 1$.

Os códigos de Goppa são bastante indicados para fins criptográficos, porém, ainda restaria decidir por seus parâmetros n, k e t, que tem as escolhas limitadas pelo Teorema 4.2.2.

4.2.2 O criptossistema de McEliece

Veremos, agora, o criptossistema de McEliece, que é conhecido como um criptossistema pós-quântico, pois sua segurança está baseada em camuflar um código de um algoritmo de decriptação eficiente na forma de um código linear genérico. Os algoritmos de encriptação e decriptação desse sistema apresentam um consumo de tempo muito menor se comparados aos algoritmos da criptografia RSA, porém as matrizes

que representam as chaves são muito maiores que as do RSA para um mesmo nível de segurança, o que torna a utilização desse criptossistema inviável em larga escala.

O criptossistema de McEliece [27] utiliza um (n, k) - código linear binário C construído selecionando-se aleatoriamente um polinômio irredutível de grau t sobre $\mathbf{F}_{2^m}$. A possibilidade de construção de um sistema criptográfico está baseada no fato de que para cada polinômio deste tipo existe um código de comprimento 2^m e dimensão $k \geq n - tm$ eficientemente decodificável e capaz de corrigir um padrão de erros com peso menor ou igual a t.

Sejam G uma matriz $k \times n$ geradora do código C, S uma matriz $k \times k$ não singular e P uma matriz de permutação $n \times n$. O criptossistema de McEliece é definido da seguinte forma:

Geração das Chaves

Chave privada: G, S e P, como descritas acima.
Chave pública: $G' = SGP$ e t.

Exemplo 4.2.4. Sejam G, S e P as matrizes que geram a chave privada:

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix} \quad S = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix} \quad P = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

Portanto, a chave pública é dada por:

$$G' = SGP = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{bmatrix}.$$

- **Mensagens**: vetores x de k bits sobre $\mathbf{F}_2$.

Encriptação

O texto cifrado é $y = xG' + e$, sendo que e é um vetor de erros aleatório sobre $\mathbf{F}_q^n$ com $wt(e) = t$.

Exemplo 4.2.5. Seja a mensagem $x = (1011)$;

- Escolha um vetor erro e aleatoriamente: $e = (0, 1, 0, 0, 0, 0, 0)$;
- Calcule a mensagem codificada: $y = xG' + e = (0, 0, 0, 1, 1, 0, 0)$.

4.2.3 Decriptação

Para decriptarmos o texto, segue-se os seguintes passos:

• Calcule $y' = yP^{-1} \Rightarrow y' = xSG + e'$, sendo $e' = eP^{-1}$;

• Utilizando o algoritmo de decriptação para corrigir os erros, obtém-se a palavra-código x';

• Calcule x'' tal que $x' = x''G$;

• Calcule $x = x''S^{-1}$.

Exemplo 4.2.6. Vamos recuperar a mensagem original:

• Calculando $y' = yP^{-1} \Rightarrow y' = xSG + e' = (0, 0, 1, 0, 0, 0, 1)$ (lembrando que $e' = eP^{-1}$);

• Utilizando o algoritmo de decriptação para corrigir os erros, obtemos a palavra-código $x' = (0, 0, 1, 0, 0, 1, 1)$;

• Calculamos x'' de forma que $x''G = x' \Rightarrow x'' = (0, 0, 1, 0)$;

• Por fim, encontramos a mensagem original $x = x''S^{-1} = (1011)$.

A seguir, apresentaremos o criptossistema de Niederreiter, que utiliza a ideia de matriz de verificação de paridade, também baseada em códigos corretores de erro (nesse caso, o mais comum é o uso do código de Goppa). Este código será definido sobre o corpo finito $\mathbf{F}_q$.

Criptossistema de Niederreiter

Sejam H uma $(n-k) \times n$ matriz de verificação de paridade de um código, M uma $(n-k) \times (n-k)$ matriz não singular e P uma $n \times n$ matriz de permutação, todas definidas sobre $\mathbf{F}_q$. Utilizando esses parâmetros, definimos o criptossistema de Niederreiter.

Geração das Chaves

• **Chave privada**: H, M e P, como descritos acima.

• **Chave Pública**: $H^* = MHP$ e t.

As mensagens serão dadas como vetores y sobre $\mathbf{F}_q$ com peso t.

Encriptação

• Texto cifrado: $x = y(H^*)^T$.

Decriptação

• $x = y(H^*)^T = y(MHP)^T \Rightarrow x(M^T)^{-1} = (yP^T)H^T$.

Utilizamos um algoritmo de decriptação para o código para encontrar yP^T, e desta forma, encontramos y.

Conclusão

Os sistemas criptográficos estudados nesse capítulo não receberam muita atenção na época que foram apresentados, mas agora, com o desenvolvimento da criptografia pós-quântica, tem recebido maior atenção, pois se mostraram resistentes a alguns tipo de ataques quânticos.

Porém, estes sistemas apresentam certa vulnerabilidade a alguns tipos de ataques, por exemplo, um tipo de ataque chamado de *Chosen Plaintex Attacks*, onde caso o atacante tenha acesso a um texto cifrado e saiba ver a criptografia de uma ou duas mensagens, podendo calcular o peso da mensagem e verificar se esse peso (t) é igual ao do valor público, e assim tem alta probabilidade de quebrar a segurança do sistema.

Para evitar esses tipos de vulnerabilidades, são feitas implementações nos criptossistemas originais, como por exemplo, nesse caso, utiliza-se o código de Goppa.

4.2.4 Criptografia pós-quântica e reticulados

Apresentaremos a seguir, dois sistemas criptográficos baseados em reticulados.

Reticulados

Os reticulados têm se mostrado bastante úteis em aplicações em telecomunicações e em criptografia pós-quântica. Nos estudos envolvendo reticulados, o que se busca em geral é, fixada uma dimensão, qual é o melhor reticulado nesta dimensão em relação à uma certa propriedade. Intuitivamente, um reticulado no $\mathbb{R}^n$ é um conjunto infinito de pontos dispostos de forma regular.

Definição 4.2.11. Seja $\{v_1, \cdots, v_m\}$ um conjunto de vetores linearmente independentes do $\mathbb{R}^n$ com $m \leq n$. Chama-se *reticulado* o conjunto de pontos da forma

$$\Lambda = \left\{ x = \sum_{i=1}^{m} a_i v_i, \text{ com } a_i \in \mathbb{Z} \right\}$$

e $\{v_1, \cdots, v_m\}$ é chamada uma *base* do reticulado.

Definição 4.2.12. Seja $\{v_1, \ldots, v_m\}$ uma base de Λ. Se $v_i = (v_{i1}, \ldots, v_{in})$, para $i = 1, \cdots, m$, a matriz

$$M = \begin{pmatrix} v_{11} & v_{12} & \ldots & v_{1n} \\ v_{21} & v_{22} & \ldots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{m1} & v_{m2} & \ldots & v_{mn} \end{pmatrix}$$

é chamada uma *matriz geradora* para o reticulado Λ. A matriz $G = MM^t$ é chamada *matriz de Gram* para o reticulado, onde t denota a transposição.

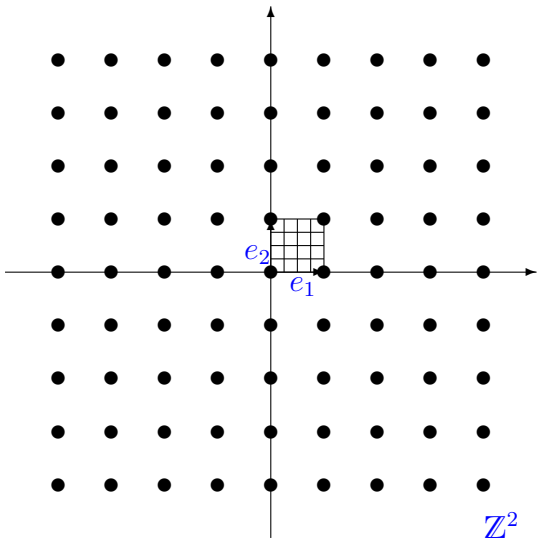
Definição 4.2.13. Chamamos de *posto* de um reticulado Λ o número de vetores de uma base de Λ, isto é, a dimensão do subespaço gerado por Λ em $\mathbb{R}^n$.

Definição 4.2.14. Seja $\Lambda \subset \mathbb{R}^n$ um reticulado, com $\mathbb{Z}$-base $\beta = \{v_1, \cdots, v_n\}$. O conjunto

$$P_\beta = \left\{ x \in \mathbb{R}^n : x = \sum_{i=1}^{n} \lambda_i v_i,\ 0 \leq \lambda_i < 1 \right\},$$

é chamado de *região fundamental* de Λ com relação a base β.

Exemplo 4.1. $\Lambda = \mathbb{Z}^2$ é um reticulado gerado pelos vetores $e_1 = (1, 0)$ e $e_2 = (0, 1)$ com região fundamental descrita na figura abaixo.



Definição 4.2.15. **a)** Um *empacotamento esférico*, ou simplesmente um empacotamento no $\mathbb{R}^n$, é uma distribuição de esferas de mesmo raio no $\mathbb{R}^n$ de forma que a intersecção de quaisquer duas esferas tenha no máximo um ponto. Pode-se descrever um empacotamento indicando apenas o conjunto dos centros das esferas e o raio.
b) Um *empacotamento reticulado* é um empacotamento em que o conjunto dos centros das esferas formam um reticulado Λ de $\mathbb{R}^n$.
c) Dado um empacotamento no $\mathbb{R}^n$, associado a um reticulado Λ, com $\{v_1, \cdots, v_n\}$ uma $\mathbb{Z}$-base, a sua *densidade de empacotamento* é definida como sendo a proporção do espaço $\mathbb{R}^n$ coberta pela união das esferas.

A forma de dispor essas esferas de modo a cobrir a maior parte do espaço, sempre foi um desafio para os matemáticos. Ao estudar a densidade de empacotamento, um dos principais problemas é a obtenção de reticulados com alta densidade de empacotamento e que sejam ao mesmo tempo manipuláveis.

Definição 4.2.16. Dizemos que um reticulado $\Lambda \in \mathbb{R}^n$ é *racional (integral)* se uma (e portanto todas) de suas matrizes de Gram possuir todas as entradas racionais (inteiras).

Observação: Dado um reticulado racional Λ sempre existe um reticulado integral Λ^* tal que $\Lambda^* = k\Lambda$ para algum $k \in \mathbb{N}$. Assim, sem perda de generalidade, os problemas envolvendo reticulados serão definidos em termos de reticulado integral.

4.2.5 Problemas clássicos envolvendo reticulados

Problema 1

Problema do vetor mais curto (Shortest Vector Problem - SPV): Dada uma base $B \in \mathbb{Z}^{m \times n}$, encontrar um vetor Bx (com $x \in \mathbb{Z}^n$) não-nulo no reticulado gerado por B tal que $||Bx|| \leq ||By||,\ \forall y \in \mathbb{Z}^n \backslash \{0\}$.

Note que o vetor mais curto em um reticulado não é único. De fato, se u é um vetor mais curto não nulo, então $-u$ também será. É possível termos vários vetores mais curtos linearmente independentes.

Problema 2

Problema do vetor mais próximo (Approximate Closest Vector Problem - CVP): Dados uma base $B \in \mathbb{Z}^{m \times n}$ para um reticulado e um vetor-alvo $t \in \mathbb{Z}^m$, encontrar um vetor Bx (com $x \in \mathbb{Z}^n$) mais próximo de t no reticulado gerado por B, isto é, $||Bx - t|| \leq ||By - t||,\ \forall y \in \mathbb{Z}^n$.

Método de Babai para CVP

Métodos de redução de reticulados fornecem uma base de vetores razoavelmente curtos e, portanto, podem ser usados para aproximar o problema do vetor mais curto. Entretanto, também é possível usar os resultados da redução de reticulados para criar algoritmos que aproximem o problema do vetor mais curto. Babai publicou dois métodos para aproximar CVP em 1986 [29], o método de arredondamento e o algoritmo do plano mais próximo. Ambos aproximam algoritmos para CVP com fatores de aproximação demonstráveis graças ao algoritmo LLL [21]. O primeiro passo nos dois algoritmos é aplicar o LLL à base de dados de modo a reduzir o comprimento dos vetores da base. No entanto, os métodos subjacentes podem ser aplicados utilizando qualquer base para o reticulado. O algoritmo do plano mais próximo fornece um melhor fator de aproximação que o método de arredondamento, mas este último é mais simples. Descreveremos aqui apenas o método de arredondamento.

Seja $\Lambda \subset \mathbb{Z}^n$ um reticulado de posto d com base B, e seja $x \in \Lambda$ o vetor alvo. O método de arredondamento pode ser descrito sucintamente pelo cálculo de

$$u = B\lfloor B^{-1}x \rceil.$$

A ideia desse método é: se $x \in \Lambda$ e $B = \{b_1, \cdots, b_n\}$ é uma base de Λ, então x pode ser escrito como uma única combinação linear dos vetores da base B:

$$x = \sum_{i=1}^{d} \lambda_i b_i,$$

onde $\lambda_i \in \mathbb{R}$. Agora, o vetor de coordenadas x com relação a base B é dado por $B^{-1}x = (\lambda_1, \cdots, \lambda_d)$. No método de arredondamento de Babai, cada λ_i é arredondado

para o número inteiro mais próximo $\lfloor \lambda_i \rceil$. O vetor resultante u do reticulado é calculado tomando a combinação linear dos b_i's com coeficientes $\lfloor \lambda_i \rceil$:

$$u = \sum_{i=1}^{d} \lfloor \lambda_i \rceil b_i = B \lfloor B^{-1} x \rceil.$$

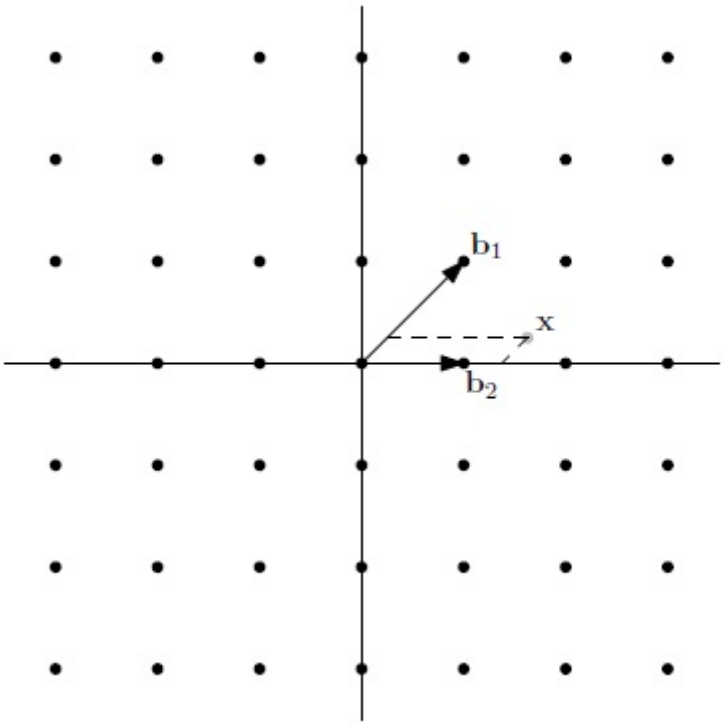
O método de arredondamento de Babai nem sempre encontra a resposta correta quando a base é ruim.

Considere o seguinte exemplo:

Exemplo 4.2.7. Seja $\Lambda = \mathbb{Z}^2$ o reticulado integral na dimensão 2. Agora considere um exemplo de CVP dado pelas bases $b_1 = (1, 1)$ e $b_2 = (1, 0)$ de Λ, e o vetor alvo $x = \left(\frac{13}{8}, \frac{1}{4}\right)$. Assim,

$$B^{-1}x = \left(\frac{1}{4}, \frac{11}{8}\right), \text{ ou equivalentemente, } \frac{1}{4}b_1 + \frac{11}{8}b_2 = x.$$

Aproximando essas coordenadas, segue que $\lfloor B^{-1}x \rceil = (0, 1)$, que leva ao vetor do reticulado $0b_1 + 1b_2 = (1, 0)$. Entretanto, o ponto do reticulado mais próximo de x é $2b_2 = (2, 0)$, como pode ser observado na figura abaixo



4.2.6 Ajtai-Dwork

Configuração e parâmetros

O sistema AD (Ajtai-Dwork) é definido sobre um espaço vetorial euclidiano, utilizando a norma euclidiana usual. O parâmetro de segurança n determina a dimensão do espaço vetorial.

Dado n, seja $m = n^3$, e $r_n = n^n$. Com esses parâmetros, denotamos um cubo n-dimensional cujos lados tem comprimento r_n por

$$B_n = \{x \in \mathbb{R}^n : |x_i| \leq r_n/2, \forall i\},$$

onde $x = (x_1, x_2, \cdots, x_n)$.

Além disso, para algum inteiro $c > 0$, denotamos a bola n-dimensional de raio n^{-c} por

$$S_n = \{x \in \mathbb{R}^n : ||x|| \leq n^{-c}\}.$$

A chave privada é um vetor u escolhido aleatoriamente da bola unitária n-dimensional. Dada essa chave privada u, a distribuição $\mathcal{H}_u$ é definida em B_n utilizando a seguinte construção:

(1) Selecione x de $\{x \in B_n : \langle x, u\rangle \in \mathbb{Z}\}$, aleatoriamente;

(2) Selecione n vetores de erro, $y_1, \cdots, y_n$ de S_n, sendo esses vetores selecionados de maneira independente e aleatória;

(3) Saída: $v = x + \sum_{i=1}^{n} y_i$.

Para obter a chave pública, os $n+m$ vetores $w_1, \cdots, w_n, v_1, \cdots, v_n$ são selecionados de $\mathcal{H}_u$, aleatoriamente. Os $w_i's$ devem estar em um paralelepípedo que os abrangem, denotado por P_β, onde $\beta = (w_1, \cdots, w_n)$, é a distância mínima de w_i ao hiperplano gerado pelos outros $w_j's$, denotado por $H_{j\neq i}$, deve ser de, no mínimo, r_n/n^2, para todo i. Se isto não ocorre, uma nova chave é gerada. Um vetor v pode ser reduzido módulo o paralelepípedo para encontrar um vetor v', tal que a diferença $v - v'$ é um vetor no reticulado gerado pelos $w_i's$.

Parâmetro	Descrição	Tipo de chave
n	Dimensão	Pública
$m = n^3$	Inteiro	Pública
$r_n = n^n$	Inteiro	Pública
u	Vetor n-dimensional	Privada
$w_1, \cdots, w_n, v_1, \cdots, v_n$	$n + m$ vetores n-dimensionais	Pública

Tabela de Parâmetros

Encriptação e decriptação

Seja um sistema com parâmetros definidos na tabela acima. A encriptação e decriptação são realizadas da seguinte maneira:

Encriptação - É realizada em cada bit, um de cada vez. Para encriptar um bit 0, tome $b_1, \cdots, b_m$, de maneira aleatória entre $\{0, 1\}$ e reduza o vetor $\sum_i b_i v_i$ módulo o paralelepípedo P_β. O resultado será um vetor n-dimensional cifrado. Para encriptar

um bit, escolha aleatoriamente um vetor n-dimensional no paralelepípedo P_β, e tome isto como um texto cifrado.

Decriptação - Para decriptar um texto cifrado c, que é um vetor n-dimensional, correspondendo a um único bit do texto simples, calcule o produto interno com a chave privada u. Se a distância $dist(< c, u >, \mathbb{Z}) \leq n^{-1}$, então c é decriptado como 0, e caso contrário, é decriptado como 1.

Conclusão

O sistema de criptografia Ajtai-Dwork é interessante teoricamente, pois a prova de segurança do sistema apresenta problemas complexos de reticulados. Entretanto, este não é um sistema muito eficiente devido à expansão de mensagem e elevado espaço de armazenamento exigido para a chave pública.

4.2.7 Criptografia NTRU

Apresentaremos nessa seção um outro sistema de criptografia pós-quântica, o sistema NTRU. Esse sistema, que é baseado em reticulados, é considerado um criptossistema mais eficiente dentre os sistemas criptográficos atuais, sendo considerado completo, ou seja, possui encriptação e assinatura digital, e pode ser facilmente implementado.

Definição 4.2.17. Seja $\mathbb{Z}_q$ o anel de inteiros módulo q. As operações do criptossistema $NTRU$ ocorrem no anel de polinômios $R = \mathbb{Z}_q[X]/(X^N - 1)$.

Usamos o símbolo $*$ para denotar a multiplicação polinomial.
O criptossistema NTRU depende de três parâmetros inteiros: N, p e q onde:

- N é primo e suficientemente grande para prevenir ataques usando reticulados.
- p e q são primos entre si.
- q é muito maior do que p.

Além dos parâmetros descritos acima, esse criptossistema também depende de quatro conjuntos de polinômios de grau $N - 1$ com coeficientes inteiros e pequenos:

- L_f é um conjunto de polinômios pequenos a partir do qual as chaves privadas são selecionadas.
- L_g é um conjunto de polinômios pequenos a partir do qual outras chaves privadas são selecionadas.
- L_m é um conjunto de polinômios $m \in \mathbb{Z}_p[X]/(X^N - 1)$ que representa mensagens encriptáveis.

- L_r é um conjunto de polinômios a partir do qual o valor escondido usado durante a encriptação é selecionado.

Geração das chaves

Para criarmos uma chave para o criptossistema NTRU, escolhemos aleatoriamente um polinômio $f \in L_f$ e um polinômio $g \in L_g$. O polinômio f tem um inverso f_p^{-1} módulo p e um inverso f_q^{-1} módulo q, isto é,

$$f * f_p^{-1} \equiv 1 \,(mod\, p), \;\; f * f_q^{-1} \equiv 1 \,(mod\, q).$$

A chave privada será f e a chave pública é o polinômio

$$h = f_q^{-1} * q \;(mod\, q).$$

Encriptação

Para encriptar uma mensagem $m \in L_m$ escolhemos aleatoriamente um polinômio $r \in L_r$. O texto encriptado é o polinômio

$$e = pr * h + m \;(mod\, q).$$

Decriptação

Para decriptar uma mensagem e usando a chave privada f, calculamos

$$a = f * e \;(mod\, q).$$

A mensagem m é então obtida reduzindo os coeficientes de

$$f_p^{-1} * a \;(mod\, p).$$

O ataque de Coppersmith e Shamir sobre o criptossistema NTRU

Coppersmith e Shamir [28] apresentaram um ataque usando reticulados sobre o criptossistema NTRU. Eles definiram um reticulado determinado pelos parâmetros N, q e h do sistema e mostraram que recuperar a chave secreta (f, g) a partir da chave pública h se restringe a encontrar um vetor mais curto do reticulado.

Seja $h = (h_0, h_1, \cdots, h_{N-1})$ a chave pública. O reticulado $NTRU$ Λ é o reticulado de dimensão $2N$ gerado pelas linhas de uma matriz da seguinte forma

$$M(\Lambda) = \begin{bmatrix} \lambda I_N & H \\ 0 & qI_N \end{bmatrix}.$$

Como $h = f_q^{-1} * g \;(mod\, q)$, então $f * h - qu = g$ para algum $u \in R$ e

$$(f, -u) * M(\Lambda) = (\lambda f, g).$$

Assim um intruso usa um algoritmo de redução de reticulados para encontrar (f, g) a partir de Λ e portanto ele pode recuperar as chaves privadas.

5 Conclusão final

Vimos a que a criptografia quântica poderá ser bastante promissora em um futuro próximo, sendo que sua segurança é melhor reforçada com relação ao algoritmo RSA, por exemplo.

Claro que com a evolução da criptografia quântica, descobriu-se também o potencial da criptoanálise quântica, e criou-se a ideia de que todos os algoritmos de criptografia se tornariam inúteis assim que computadores quânticos fossem produzidos em larga escala, pois todas as mensagens criptografadas com os algoritmos atuais seriam facilmente interceptáveis.

Mas como essa tecnologia ainda não está em uso, em larga escala, os algoritmos de criptografia atuais continuam seguros, mas como dito inicialmente, sempre criptografia e criptoanálise caminham juntas, cada algoritmo de criptografia só é eficiente até a criptoanálise encontrar um método de quebra, e assim, sucessivamente.

Boa parte da comunidade científica expressa grande ceticismo com relação ao quão rapidamente se desenvolverão computadores quânticos comerciais. Afirmam que, por causa disso, não há grande utilidade em se preocupar com tal possibilidade.

A utilização de canais quânticos de comunicação, o que já é possível atualmente, traz uma série de vantagens em segurança comparada às técnicas de criptografia da computação clássica, como impossibilidade de interceptação em alguns protocolos, e segurança da comunicação.

Vimos também que o surgimento de computadores quânticos comerciais, que não está muito distante, tem o potencial de revolucionar a computação. Eles, se capazes de executar o algoritmo de Shor, se tornarão uma ameaça a grande parte dos sistemas de segurança atuais, como a criptografia RSA e o algoritmo de Diffie-Hellman.

No entanto, há algoritmos de criptografia resistentes a ataques quânticos, como os criptossistemas de McEliece e NTRU. Podemos esperar que algoritmos desse tipo substituam o RSA em grande parte dos sistemas de segurança antes mesmo de surgirem os primeiros computadores quânticos comerciais capazes de quebrá-las. O interesse pela criptografia pós-quântica deve crescer conforme os avanços em computação quântica se tornem mais evidentes.

Referências

[1] SINGH, S. *O Livro dos Códigos*. Rio de Janeiro: Record, 2001.

[2] DIAS. J. L. Cesubra Cientista. *Revista do Centro Universitário Planalto do Distrito Federal,* Distrito Federal, n. 3, pp 749-759, 2006.

[3] SANTOS, J. P. O. *Introdução à Teoria dos Números*. Rio de Janeiro: IMPA, 2007. (Coleção Matemática Universitária)

[4] COUTINHO. S. C. *Números inteiros e Criptografia RSA*. 2 ed. Rio de Janeiro: IMPA, 2011. (Coleção Matemática e Aplicações)

[5] SOUZA, B. A. *Teoria dos números e o RSA*. 66 f. Dissertação (Mestrado em Matemática Aplicada) - Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, 2004.

[6] FREITAS H. C.; SOUSA A. S.; AUGUSTINI E. Um Enfoque Computacional da Criptografia RSA. *Revista Científica Eletrônica da Faculdade de matemática*, Uberlândia, n. 3, p. 123 - 124, set. 2004.

[7] PEDRO, L. R.; CIOLETTI, L. M. *Algoritmo Euclidiano Estendido*. Belo Horizonte. Universidade Federal de Minas Gerais, 2009.

[8] BARNABÉ, V. C. *Uma introdução ao métodos de fatoração de inteiros de Fermat e Pollard*. 33 f. Trabalho de conclusão de curso (Licenciatura em Matemática) - Universidade Estadual do Mato Grosso do Sul, Dourados, 2009.

[9] CAMPELLO, A. C.; LEAL, I. *Um algoritmo para solução de congruências do tipo* $a^{p-1} \equiv \ (mod\ p^2)$. Monografia (Graduação em Matemática Aplicada e Computacional) - Unicamp. Campinas, 2007.

[10] ZACCARON, A. Z.; MOLGORA A. B. P. *Crivo Quadrático: Um estudo da obtenção de números completamente fatorados sobre uma base de fatores*. In: ENCONTRO DE INICIAÇÃO CIENTÍFICA, Universidade Estadual de Mato Grosso do Sul, Dourados, 2011.

[11] SANCHES, A. P.; MOLGORA A. B. P. *Um estudo do método de fatoração de inteiros Crivo Quadrático.* In: Sociedade Brasileira de Matemática Aplicada e Computacional, Universidade Estadual de Mato Grosso do Sul, Dourados 2011.

[12] PROBST, R. W. *Números primos.* 2003. 53 f. Trabalho de conclusão de curso (Bacharelado em Matemática) - Universidade Regional de Blumenau, Blumenau, 2003.

[13] MOREIRA, C. G. T. A.; SALDANHA, N. C. *Primos de Mersenne (e outros primos muito grandes)* 3. ed. Rio de Janeiro: IMPA, 2008. (Publicações Matemáticas)

[14] MELO, G. S. *Uma introdução aos testes de primalidade.* 43 f. Trabalho de conclusão de curso (Licenciatura em Matemática) - Universidade Estadual de Mato Grosso do Sul, Dourados, 2009.

[15] SILVA E. F. Equações Diofantinas Lineares. *Revista da Olimpíada*, Goiânia, n. 3, p. 110-118, jan./dez. 2002.

[16] BIASE, A.G. *Introdução ao Estudo de Criptografia.* In: XII SEMINÁRIO DE INICIAÇÃO CIENTÍFICA, Universidade Federal de Uberlândia, Uberlândia 2008.

[17] FADEL, D. F. *Criptografia RSA.* Monografia (Instituto de Matemática, Estatística e Computação Científica) Campinas: Unicamp, 2007.

[18] OLIVEIRA, P. E. R.; ANDRADE, P. T. E.; D'OLIVEIRA, R. L. G. *Rsa.* Monografia. Campinas: Unicamp, 2007.

[19] POSTAL, T. *Criptografia RSA.* Monografia (Licenciatura em Matemática) - Universidade Federal do Mato Grosso, Cuiabá, 2008.

[20] BIASE, A. G. Criptografias ElGamal, Rabin e algumas técnicas de ciframento. *FAMAT em revista*, 13. Universidade Federal de Uberlândia, Uberlândia, 2009.

[21] POL, J. H. V. *Lattice-based cryptography.* 108 f. M. Sc. Thesis - Eindhoven University of Technology, Eindhoven, 2011.

[22] CONWAY, J.H.; SLOANE, N.J.A. *Sphere Packings, Lattices and Groups.* New York: Springer, 1999.

[23] SAMUEL, P. *Algebraic Theory of Numbers.* Paris: Hermana, 1967.

[24] NGUYEN, P.Q.; STERN, J. *Cryptanalysis of the Ajtai-Dwork cryptosystem.* Springer Berlin: Heidelberg, 1998.

[25] PFEIFFER G.; PAIM R.;MOTTA V. *Criptografia Quântica.* 2011. Disponível em: <http://www.gta.ufrj.br/grad/11_1/quantica/trabalho003.html>. Acesso: 28 set. 2013.

[26] MISOCZKI, R. *Criptografia Pós-Quântica com Códigos Corretores de Erros.* 41 f. Monografia (Bacharelado em Ciências da Computação) - Universidade de São Paulo, São Paulo, 2008.

[27] MCELIECE, R. J. *A public-key cryptosystem based on algebraic coding theory.* v. 42, n. 44. Jet Propulsion Laboratory DSN Progress Report, 1978.

[28] COOPERSMITH D.; SHAMIR. A. *Lattice attacks on NTRU, In Advances in cryptology-Eurocrypt.* n. 97, Springer: Berlin, 1997.

[29] BABAI, L. *On Lovász lattice reduction and the nearest lattice point problem*, Combinatorica, v. 6 (1), p. 1-13, 1986.