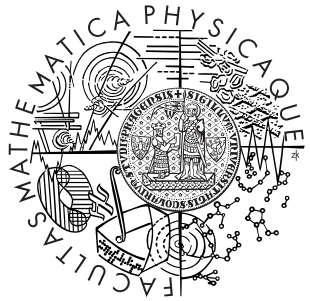
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Lukáš Perůtka

Hledání optimálních strategií číselného síta

Katedra algebry

Vedoucí diplomové práce: Prof. RNDr. Aleš Drápal, CSc., DSc.

Studijní program: Matematika, Matematické medoty informační bezpečnosti

Děkuji vedoucímu mé diplomové práce Prof. Drápalovi za jeho vedení a cenné rady. Dále bych chtěl poděkovat celému kolektivu autorů softwarové implementace algoritmu číselného síta, kterou jsem použil ke své práci, a která mi pomohla lépe pochopit detaily algoritmu.

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce a jejím zveřejňováním.

V Praze dne 1. dubna 2009 Lukáš Perůtka

Obsah

Název práce: Hledání optimálních strategií číselného síta
Autor: Lukáš Perůtka
Katedra (ústav): Katedra algebry
Vedoucí diplomové práce: Prof. RNDr. Aleš Drápal, CSc., DSc.
e-mail vedoucího: drapal@karlin.mff.cuni.cz

Abstrakt: V předložené práci studujeme algoritmus číselného síta. Zaměřujeme se především na jeho teoretickou podstatu s vyložením všech důležitých tvrzení potřebných k pochopení fungování algoritmu. Dále popisujeme několik nejpoužívanějších realizací jednotlivých částí algoritmu s vysvětlením, pro jaké situace jsou nejvýhodnější. Na závěr uvádíme výsledky měření efektivnosti prosívání dvou základních metod s pomocí implementace algoritmu vzniklém na katedře algebry.

Klíčová slova: NFS, GNFS, číselné těleso

Title: Searching optimal strategies for the number field sieve
Author: Lukáš Perůtka
Department: Department of Algebra
Supervisor: Doc. RNDr. Aleš Drápal, CSc., DSc.
Supervisor's e-mail address: drapal@karlin.mff.cuni.cz

Abstract: In this work we study the number field sieve algorithm. Our main focus is on its theoretical background. We present all important theorems which are needed for a full understanding of the algorithm. We also describe the most widely used implementation of the parts of the algorithm and we discuss in which situation they should be used. At the end we show results from measurements of sieving phase on the implementation which was written for our Department of Algebra.

Keywords: NFS, GNFS, number fields

Kapitola 1

Úvod

1.1 Faktorizace

Úkol rozložit zadané číslo na prvočísla patří mezi nejstarší matematické problémy. Ani v dnešní době nedokážeme v rozumném čase faktorizovat čísla s řádem větším než 200 (tato skutečnost je jedním z pilířů bezpečnosti některých asymetrických šifer). Hledání rychlého algoritmu, který by dokázal v krátké době rozložit velká čísla, se dostává do popředí zájmů kryptografů, matematiků i informatiků. V polovině dvacátého století bylo za úspěch považováno rozložení dvaceticiferného čísla. V roce 1975 byl Michaelem A. Morrisonem a Johnem Brillhartem vymyšlen faktorizační algoritmus CFRAC (the Continued Fraction Factoring Algorithm), který je vhodný pro faktorizování čísel přibližně s padesáti ciframi. O pár let později přišel Carl Pomerance se základní verzí algoritmu QS (the Quadratic Sieve) viz [30], který byl následně vylepšen R. D. Silvermanem tzv. MPQS (the Multiple Polynomial Quadratic Sieve) a dále pak nezávisle na sobě R. Peraltanem a W. Alfordem spolu s C. Pomerancem tzv. SIQS (the Self Initializing Quadratic Sieve). Algoritmus QS dokáže v přijatelném čase faktorizovat čísla s řádem okolo sta. Avšak v dnešní době nejrychlejší faktorizační algoritmus pro čísla s řádem větším než sto je GNFS (the General Number Field Sieve) odvozený od algoritmu navrhnutého J. M. Pollardem. Tímto algoritmem se tato práce zabývá.

V druhé kapitole se zaměříme na teorii, na které je algoritmus číselného síta založen. Budeme se soustředit pouze na věci, které jsou zásadní k pochopení hlavních principů algoritmu. Našim cílem tedy není popsat celou obecnou teorii číselných těles, ale pouze vybrat stěžejní věty. V této kapitole jdeme za rámec látky vyučované v magisterských kurzech. Jde zejména o popis duálních bází (věty 2.25, 2.26), které využijeme v závěrečné fázi algoritmu, a celá dlouhá část 2.3, ve které se vysvětluje, jak se algoritmicky v číselných tělesech počítají rozklady ideálů na prvoideály. K tomu budeme potřebovat nalézt prvoideály nad speciálními prvočísly. Je pozoruhodné, že i když o daném prvku číselného tělesa víme, že je druhou mocninou, tak zdaleka není jednoduché takový prvek nalézt. Část 2.5 je věnována teoretickým úvahám potřebným pro zvládnutí algorit-

mického řešení tohoto problému. V této kapitole se od čtenáře očekává znalost základů lineární algebry, teorie čísel, Galoisovy teorie, vlastností konečně generovaných abelovských grup a teorie konečných okruhů.

Popis algoritmu číselného síta je uveden v třetí kapitole. Zvolen byl zcela obecný popis algoritmu, který umožňuje vysvětlit kromě standardní verze s jedním lineárním polynomem také variantu s dvěma polynomy vyššího stupně. Pomocí tohoto popisu je dále možné jednoduše rozšířit algoritmus na použití více než dvou polynomů vyšších stupňů. V první části rozebereme i stručnou historii a vývoj algoritmu číselného síta. Následující části jsou pak věnovány jednotlivým fázím algoritmu. Algoritmus číselného síta je velmi robustní a pro jednotlivé fáze algoritmu existuje několik přístupů, jak je řešit. Našim cílem je uvést nejpoužívanější způsoby. Přitom uvedeme, ve kterých situacích je vhodné jednotlivá řešení použít. Zmíníme i možnosti paralelizace jednotlivých částí.

Na závěr využijeme skutečnosti, že na naší fakultě máme implementaci číselného síta, na jejímž vývoji se podílím, a provedeme několik měření. Cílem měření bude na konkrétním čísle ukázat rozdíl v efektivnosti mezi klasickým řádkovým prosíváním a mřížovým prosíváním. Přitom se zaměříme na nejlepší volbu meze sloužící k identifikaci kandidátů na faktorizaci a navrhneme způsob, jak určit mez u klasického prosívání. Následně při zkušební faktorizaci testovaného čísla ověříme, zda navržený způsob výpočtu meze vede ke zrychlení prosívací fáze.

Kapitola 2

Teoretický základ algoritmu

2.1 Číselná tělesa

Nejdříve uvedeme definici číselného tělesa.

Definice 2.1. *Číselné těleso* K je podtěleso komplexních čísel, které má konečný stupeň nad $\mathbb{Q}$.

Jinými slovy, jedná se o algebraické rozšíření $\mathbb{Q}$ konečného stupně, proto $K = \mathbb{Q}[\alpha]$ pro nějaké algebraické číslo $\alpha \in \mathbb{C}$ nad $\mathbb{Q}$. Je-li f ireducibilní polynom nad $\mathbb{Q}$ stupně n a $f(\alpha) = 0$, pak můžeme psát:

$$K = \mathbb{Q}[\alpha] = \{a_0 + a_1\alpha + \ldots + a_{n-1}\alpha^{n-1};\, a_i \in \mathbb{Q},\, i = 0, \ldots, n-1\}$$

Množina $\{1, \alpha, \ldots, \alpha^{n-1}\}$ tvoří bázi K nad $\mathbb{Q}$ jakožto vektorového prostoru. V dalším textu se budeme striktně držet označení K, L, M pro číselná tělesa. Některé pojmy budeme uvádět pouze pro číselná tělesa, i když je možné je definovat mnohem obecněji.

Definice 2.2. Komplexní číslo ϑ nazýváme *algebraické celé číslo*, právě když je kořenem nějakého monického polynomu nad $\mathbb{Z}$. Množinu všech algebraických celých čísel označíme $\mathbb{A}$.

Nezapomínejme, že libovolné algebraické číslo α (nad $\mathbb{Q}$) je obecně kořenem nemonického polynomu nad $\mathbb{Z}$. Stačí vzít minimalní polynom α nad $\mathbb{Q}$ a vynásobit ho společným jmenovatelem koeficientů. Pomocí Gaussova lemmatu o ireducibilních polynomech nad $\mathbb{Q}$ snadno odvodíme, že pro každé celé algebraické číslo existuje monický ireducibilní polynom nad $\mathbb{Q}$ s koeficienty v $\mathbb{Z}$.

Nyní uvedeme obecné lemma, které využijeme v několika důkazech.

Lemma 2.3. *Nechť R je komutativní okruh a O je konečně generovaný věrný R-modul s množinou generátorů $\{\alpha_1, \ldots, \alpha_n\}$. Pak pro libovolný nenulový prvek ρ takový, že $\rho\alpha_i \in O$ pro všechna $i = 1, \ldots, n$, existuje monický polynom nad R s kořenem ρ.*

Důkaz. Prvky $\rho\alpha_i$ můžeme vyjádřit jako R-lineární kombinace generátorů

$$\begin{pmatrix} \rho\alpha_1 \\ \vdots \\ \rho\alpha_n \end{pmatrix} = A \cdot \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix},$$

kde $A \in R^{n\times n}$. Elementárními úpravami dostáváme

$$(\rho I - A) \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}.$$

Tedy $\det(\rho I - A) = 0$, takže ρ je kořenem polynomu nad R, který vznikne rozvinutím determinantu matice $xI - A$.

□

Další vlastnost celých algebraických čísel popisuje následující lemma.

Lemma 2.4. *Nechť ϑ a μ jsou algebraická celá čísla. Pak i $\vartheta + \mu$ a $\vartheta\mu$ jsou algebraická celá čísla.*

Důkaz. Pro pevně zvolené ϑ a μ potřebujeme najít monické polynomy nad $\mathbb{Z}$, které mají za kořeny $\vartheta + \mu$ a $\vartheta\mu$. Pak se podle definice 2.2 bude jednat o celá algebraická čísla. Nechť m, resp. n, je stupeň monického polynomu nad $\mathbb{Z}$ jehož je ϑ (resp. μ) kořenem. Okruh $\mathbb{Z}[\vartheta, \mu]$ je jistě generovaný množinou $M = \{\vartheta^i\mu^j;\, 0 \le i < m,\, 0 \le j < n\} = \{\xi_1, \dots \xi_r\}$, $r = mn$. Uvažujme $\rho \in \{\vartheta + \mu, \vartheta\mu\}$. Potom podle předchozího lemmatu existuje polynom nad $\mathbb{Z}$ s kořenem ρ.

□

Přímým důsledkem je následující důležitá věta.

Věta 2.5. *Množina $\mathbb{A}$ celých algebraických čísel v $\mathbb{C}$ tvoří okruh.*

Nyní můžeme definovat obdobu celých čísel (chápaných v $\mathbb{Q}$) pro číselné těleso K:

Definice 2.6. *Okruhem celých algebraických čísel* (*obor celých čísel*, *celistvý obor*) číselného tělesa K nazýváme okruh $\mathbb{A} \cap K$. Budeme ho značit O_K.

Pro ilustraci uveďme, že pro $K = \mathbb{Q}[\alpha]$ je jistě $\mathbb{Z}[\alpha] \subseteq O_K$, ale obecně rovnost nenastává. Přejďeme k dalším potřebným definicím.

Definice 2.7. Nechť K je číselné těleso, $[K : \mathbb{Q}] = n$, a ať $\sigma_1, \dots, \sigma_n$ jsou po dvou různá vnoření K do $\mathbb{C}$ s vlastností, že restrikce $\sigma_i \mid \mathbb{Q} = \mathrm{id}_{\mathbb{Q}}$ pro $i = 1, \dots, n$. Pak definujeme zobrazení $T : K \to \mathbb{C}$ (*stopa*) a $N : K \to \mathbb{C}$ (*norma*) předpisem

$$T(\alpha) = \sigma_1(\alpha) + \ldots + \sigma_n(\alpha), \text{ a}$$
$$N(\alpha) = \sigma_1(\alpha) \cdot \ldots \cdot \sigma_n(\alpha).$$

Pro zpřesnění někdy píšeme $T^K_{\mathbb{Q}}$ a $N^K_{\mathbb{Q}}$.

Přímo z definice plyne, že stopa je lineární zobrazení a norma je multiplikativní zobrazení. Dále je zřejmé, že pro $a \in \mathbb{Q}$ máme $T(a) = na$ a $N(a) = a^n$.

Lemma 2.8. *Nechť K je číselné těleso s oborem celých algebraických čísel O_K. Pak pro $\alpha \in K$ leží $T(\alpha)$ a $N(\alpha)$ v $\mathbb{Q}$ a pro $\alpha \in O_K$ leží $T(\alpha)$ a $N(\alpha)$ v $\mathbb{Z}$.*

Důkaz. Pro libovolné $\alpha \in K$ položme $d = [\mathbb{Q}[\alpha] : \mathbb{Q}]$. Pak $[K : \mathbb{Q}[\alpha]] = \frac{n}{d}$ a pouze d vnoření z K do $\mathbb{C}$ je různých na $\mathbb{Q}[\alpha]$. Přitom tato vnoření permutují kořeny monického ireducibilního polynomu příslušného α. Proto až na znaménko je $T_{\mathbb{Q}}^{\mathbb{Q}[\alpha]}(\alpha)$ druhý koeficient tohoto polynomu a $N_{\mathbb{Q}}^{\mathbb{Q}[\alpha]}(\alpha)$ poslední koeficient. Tyto koeficienty jsou racionální. Dostáváme $T_{\mathbb{Q}}^{K}(\alpha) = \frac{n}{d} \cdot T_{\mathbb{Q}}^{\mathbb{Q}[\alpha]}(\alpha)$, podobně pro normu. Protože algebraická celá čísla mají příslušný monický polynom nad $\mathbb{Z}$, je norma a stopa celé číslo.

□

Pojem normy a stopy lze snadno rozšířit. V definici 2.7 můžeme místo $\mathbb{Q}$ uvažovat libovolné číselné podtěleso tělesa K. Analogicky platí i předchozí lemma a další zmíněné vlastnosti. Navíc dostáváme tranzitivitu v následujícím smyslu:

Věta 2.9. *Nechť K, L, M jsou číselná tělesa s vlastností $K \subseteq L \subseteq M$. Pak pro každé $\alpha \in M$ platí:*

$$T_K^L(T_L^M(\alpha)) = T_K^M(\alpha)$$
$$N_K^L(N_L^M(\alpha)) = N_K^M(\alpha)$$

Důkaz. Označme $\sigma_1, \ldots, \sigma_n$ vnoření tělesa L do $\mathbb{C}$, pro která platí, že restrikce $\sigma_i \mid K = \mathrm{id}_K$, a $\rho_1, \ldots, \rho_m$ vnoření tělesa M do $\mathbb{C}$, pro která platí, že restrikce $\rho_j \mid L = \mathrm{id}_L$. Nejprve potřebujeme normální rozšíření N tělesa $\mathbb{Q}$ takové, že $M \subseteq N$. Potom všechna vnoření σ_i a ρ_j mohou být rozšířena na automorfismy N; budeme je značit s pruhem. Nyní můžeme tato vnoření skládat a dostáváme

$$T_K^L(T_L^M(\alpha)) = \sum_{i=1}^{n} \overline{\sigma}(\sum_{j=1}^{m} \overline{\rho}_j(\alpha)) = \sum_{i,j=1}^{n,m} \overline{\sigma}_i\overline{\rho}_j(\alpha)$$

$$N_K^L(N_L^M(\alpha)) = \prod_{i=1}^{n} \overline{\sigma}(\prod_{j=1}^{m} \overline{\rho}_j(\alpha)) = \prod_{i,j=1}^{n,m} \overline{\sigma}_i\overline{\rho}_j(\alpha)$$

Přitom zřejmě $\overline{\sigma}_i\overline{\rho}_j$ po restrikci na M jsou vnoření do $\mathbb{C}$, pro která platí $\overline{\sigma}_i\overline{\rho}_j(K) = K$, a jsou všechna různá. □

S pomocí multiplikativity normy snadno dokážeme následující lemma.

Lemma 2.10. *Nechť K je číselné těleso s oborem celých algebraických čísel O_K a nechť $\vartheta \in O_K$. Pak platí následující:*

(i) ϑ je jednotka v O_K, právě když $N(\vartheta) = \pm 1$;

(ii) pokud $N(\vartheta) = \pm p$, p prvočíslo, pak ϑ je ireducibilní v O_K.

Důkaz. Lemma postupně dokážeme.

(i) Zřejmě $N(1) = 1$. Máme-li $\vartheta\mu = 1$, pak $1 = N(1) = N(\vartheta\mu) = N(\vartheta)N(\mu)$. Přitom $N(\vartheta)$, $N(\mu) \in \mathbb{Z}$, proto $N(\vartheta) = \pm 1$.

Naopak nechť $n = [K : \mathbb{Q}]$ a $\sigma_1, \ldots, \sigma_n$ jsou všechna různá vnoření K do $\mathbb{C}$ a nechť $\sigma_1 = \mathrm{id}_K$. Pak $1 = N(\vartheta) = \prod_{i=1}^{n} \sigma_i(\vartheta) = \vartheta \cdot \prod_{i=2}^{n} \sigma_i(\vartheta) = \vartheta \cdot \mu$, přitom μ zřejmě leží v O_K.

(ii) Pokud $\vartheta = \mu_1 \cdot \mu_2$, pak $N(\vartheta) = N(\mu_1)N(\mu_2)$ v $\mathbb{Z}$ a protože $N(\vartheta)$ je prvočíslo, musí nutně $N(\mu_1)$ nebo $N(\mu_2)$ být ± 1 a tedy jednotkou v O_K podle bodu (i).

□

Pro algoritmus číselného síta bude potřeba umět rychle počítat normy prvků tvaru $a + b\alpha$, kde $a, b \in \mathbb{Z}$ jsou nesoudělná a $K = \mathbb{Q}[\alpha]$. Následující lemma nám dává jednoduchý prostředek k určení normy těchto prvků.

Lemma 2.11. *Nechť $K = \mathbb{Q}[\alpha]$ je číselné těleso a nechť $f(x)$ je minimální polynom α nad $\mathbb{Q}$ stupně n. Pak pro prvek $a + b\alpha$, kde $a, b \in \mathbb{Q}$, platí*

$$N(a + b\alpha) = F(a, -b),$$

kde $F(x, y) = y^n f(\frac{x}{y})$ je homogenizovaný polynom $f(x)$.

Důkaz. Důkaz provedeme přímo výpočtem. Nechť $f(x) = \sum_{i=0}^{n} c_i x^i$ a nechť $\sigma_1, \ldots, \sigma_n$ jsou vnoření K do $\mathbb{C}$ s vlastností, že restrikce $\sigma_i \mid \mathbb{Q} = \mathrm{id}_{\mathbb{Q}}$ pro $i = 1, \ldots, n$. Pak podle definice normy je

$$N(a + b\alpha) = \sigma_1(a + b\alpha) \cdot \ldots \cdot \sigma_n(a + b\alpha) = (a + b\alpha_1) \cdot \ldots \cdot (a + b\alpha_n),$$

kde $\alpha_i \in \mathbb{C}$ jsou všechny kořeny polynomu $f(x)$ v $\mathbb{C}$. Pokud výraz napravo roznásobíme a rozdělíme na části podle toho, v kolikáté mocnině obsahuje (například) číslo b, dostaneme obecně

$$a^{n-i} b^i \left(\sum_{1 \le j_1 < \ldots < j_i \le n} \alpha_{j_1} \ldots \alpha_{j_i} \right),$$

kde $I = \{1, \ldots, n\}$. Výraz v závorce je zřejmě roven $(-1)^i c_{n-i}$ a tedy

$$N(a + b\alpha) = (a + b\alpha_1) \cdot \ldots \cdot (a + b\alpha_n) = \sum_{i=0}^{n} c_{n-i} a^{n-i} (-b)^i = F(a, -b).$$

□

Další důležitý pojem v teorii číselných těles je diskriminant. Uvedeme několik základních vlastností diskriminantu a jeho význam pro číselné těleso a pro obor celých algebraických čísel.

Definice 2.12. Nechť K a L jsou číselná tělesa, $L \supseteq K$, $[L : K] = n$ a $\sigma_1, \ldots, \sigma_n$ jsou vnoření L do $\mathbb{C}$, pro která platí, že restrikce $\sigma_i \mid K = \mathrm{id}_K$. Pro libovolnou n-tici prvků $\alpha_1, \ldots, \alpha_n \in L$ definujeme *diskriminant* předpisem:

$$\mathrm{disc}(\alpha_1, \ldots, \alpha_n) = |(\sigma_i(\alpha_j))_{i,j=1}^n|^2$$

(diskriminant je tedy druhá mocnina determinantu matice, která má na i-tém řádku v j-tém sloupci hodnotu $\sigma_i(\alpha_j)$).

Diskriminant je možné jednodušeji vyjádřit pomocí stopy.

Lemma 2.13. *Nechť K a L jsou číselná tělesa, $L \supseteq K$, $[L : K] = n$. Pak pro $\alpha_1, \ldots, \alpha_n \in L$ dostáváme*

$$\mathrm{disc}(\alpha_1, \ldots, \alpha_n) = |(T_K^L(\alpha_i \alpha_j))_{i,j=1}^n|.$$

Důkaz. Pro libovolnou čtvercovou matici M platí $(\det(M))^2 = \det(MM) = \det(M^T M)$. Stačí tedy dokázat, že když vynásobíme i-tý sloupec s j-tým sloupcem matice $(\sigma_i(\alpha_j))_{i,j=1}^n$, dostaneme prvek z matice $(T_K^L(\alpha_i\alpha_j))_{i,j=1}^n$ na pozici (i, j). Ovšem

$$(\sigma_1(\alpha_i), \ldots, \sigma_n(\alpha_i)) \cdot (\sigma_1(\alpha_j), \ldots, \sigma_n(\alpha_j))^T =$$
$$= \sigma_1(\alpha_i)\sigma_1(\alpha_j) + \ldots + \sigma_n(\alpha_i)\sigma_n(\alpha_j) = \sigma_1(\alpha_i\alpha_j) + \ldots + \sigma_n(\alpha_i\alpha_j) = T_K^L(\alpha_i\alpha_j).$$

□

Důsledek 2.14. *Nechť K je číselné těleso, $n = [K : \mathbb{Q}]$ a $\alpha_1, \ldots, \alpha_n \in K$. Pak* $\mathrm{disc}(\alpha_1, \ldots, \alpha_n) \in \mathbb{Q}$. *Je-li $\alpha_1, \ldots, \alpha_n \in O_K$, pak* $\mathrm{disc}(\alpha_1, \ldots, \alpha_n) \in \mathbb{Z}$.

Víme, že číselné těleso K tvoří vektorový prostor nad $\mathbb{Q}$, proto můžeme uvažovat o lineární nezávislosti či závislosti prvků z K. Pojem diskriminantu nám umožňuje určit, kdy je n-tice prvků z K $\mathbb{Q}$-lineárně nezávislá, tedy tvoří bázi K nad $\mathbb{Q}$.

Věta 2.15. *Nechť K je číselné těleso, $n = [K : \mathbb{Q}]$. Potom $\alpha_1, \ldots, \alpha_n \in K$ jsou lineárně závislé nad $\mathbb{Q}$, právě když* $\mathrm{disc}(\alpha_1, \ldots, \alpha_n) = 0$.

Důkaz. Pokud $\alpha_1, \ldots, \alpha_n \in K$ jsou lineárně závislé nad $\mathbb{Q}$, pak existují čísla $a_i \in \mathbb{Q}$ ne všechny nulová tak, že $\sum_{i=1}^n a_i\alpha_i = 0$. Nechť $\sigma_1, \ldots, \sigma_n$ jsou po dvou různá vnoření K do $\mathbb{C}$, pro která platí, že restrikce $\sigma_i \mid \mathbb{Q} = \mathrm{id}_{\mathbb{Q}}$. Zřejmě $\sum_{i=1}^n a_i\sigma_j(\alpha_i) = 0$ pro $j = 1, \ldots, n$. Pak jsou lineárně závislé i sloupcové vektory matice $(\sigma_i(\alpha_j))_{i,\,j=1}^n$, opět můžeme použít koeficienty a_i. Tedy $\mathrm{disc}(\alpha_1, \ldots, \alpha_n) = 0$.

Naopak budeme postupovat sporem. Nechť tedy $\alpha_1, \ldots, \alpha_n \in K$ jsou lineárně nezávislé nad $\mathbb{Q}$. Je-li $\operatorname{disc}(\alpha_1, \ldots, \alpha_n) = 0$, pak jsou lineárně závislé řádky χ_i matice $(T(\alpha_i\alpha_j))_{i,j=1}^n$ a můžeme nalézt racionální čísla $a_1, \ldots, a_n$, která nejsou všechna nulová, že $a_1\chi_1 + \ldots + a_n\chi_n$ je nulový vektor. Z lineární nezávislosti $\alpha_1, \ldots, \alpha_n$ dostáváme, že prvek $\alpha = a_1\alpha_1 + \ldots + a_n\alpha_n \neq 0$. Přitom ale pro každé j dostáváme $T(\alpha\alpha_j) = \sum_{i=1}^n a_i T(\alpha_i\alpha_j) = 0$. Dále zřejmě $\alpha_1, \ldots, \alpha_n$ tvoří bázi K nad $\mathbb{Q}$ a tedy i prvky $\alpha\alpha_1, \ldots, \alpha\alpha_n$ tvoří bázi. Pak ovšem pro libovolné $\beta \in K$ máme

$$T(\beta) = T(\sum_{i=1}^n b_i\alpha\alpha_i) = \sum_{i=1}^n b_i T(\alpha\alpha_i) = 0, \;\; b_i \in \mathbb{Q}.$$

Ale to je zřejmě ve sporu s tím, že například pro 1 máme $T(1) = n$.

□

Nyní s pomocí diskriminantu dokážeme důležitou větu popisující strukturu oboru celých algebraických čísel.

Věta 2.16. *Nechť K je číselné těleso s oborem celých algebraických čísel O_K, $[K : \mathbb{Q}] = n$. Pak existují $\vartheta_1, \ldots, \vartheta_n \in O_K$, že $O_K = \{\sum_{i=1}^n z_i \cdot \vartheta_i;\ z_i \in \mathbb{Z}\}$. Neboli O_K je volná abelovská grupa hodnosti n.*

Důkaz. Ze všech možných lineárně nezávislých n-tic $\vartheta_1, \ldots, \vartheta_n \in O_K$ vybereme takovou, pro kterou $0 < |\operatorname{disc}(\vartheta_1, \ldots, \vartheta_n)| \in \mathbb{Z}$ je nejmenší a ukážeme, že se jedná o hledanou volnou bázi (protože je diskriminant nenulový zřejmě se jedná o bázi K nad $\mathbb{Q}$). Takové n-tice jistě existují, například $1, \vartheta, \ldots, \vartheta^{n-1}$, kde $K = \mathbb{Q}[\vartheta]$. Postupujeme sporem. Nechť tedy například máme $\mu \in O_K$, $\mu = a_1\vartheta_1 + \ldots + a_n\vartheta_n$, $a_i \in \mathbb{Q}$, taková, že $a_1 \notin \mathbb{Z}$. Označme $f = \lfloor a_1 \rfloor \in \mathbb{Z}$ a $1 > \theta = a_1 - f > 0$. Položme $\xi_i = \vartheta_i$, $i = 2, \ldots, n$, a $\xi_1 = \theta\alpha_1 + a_2\alpha_2 + \ldots + a_n\alpha_n = \mu - f\alpha_1 \in O_K$. Vyjádříme-li ξ_i pomocí ϑ_i, dostaneme

$$\begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \vdots \\ \xi_n \end{pmatrix} = \begin{pmatrix} \theta & a_2 & \cdots & \cdots & a_n \\ 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 1 \end{pmatrix} \begin{pmatrix} \vartheta_1 \\ \vartheta_2 \\ \vdots \\ \vdots \\ \vartheta_n \end{pmatrix}.$$

Pokud na tuto rovnici aplikujeme σ_i (vnoření K do $\mathbb{C}$, pro která platí, že restrikce $\sigma_i \mid \mathbb{Q} = \mathrm{id}_{\mathbb{Q}}$, celkem jich máme n různých), zůstane prostřední matice nezměněna. Můžeme tedy sestavit maticovou rovnici

$$\begin{pmatrix} \sigma_1(\xi_1) & \cdots & \sigma_1(\xi_n) \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \sigma_n(\xi_1) & \cdots & \sigma_n(\xi_n) \end{pmatrix} = \begin{pmatrix} \theta & a_2 & \cdots & a_n \\ 0 & 1 & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & \cdots & 1 \end{pmatrix} \begin{pmatrix} \sigma_1(\vartheta_1) & \cdots & \sigma_1(\vartheta_n) \\ \vdots & & \vdots \\ \vdots & & \vdots \\ \sigma_n(\vartheta_1) & \cdots & \sigma_n(\vartheta_n) \end{pmatrix}.$$

Vezmeme-li druhou mocninu determinantů dostáváme

$$\operatorname{disc}(\xi_1, \ldots, \xi_n) = \theta^2 \operatorname{disc}(\vartheta_1, \ldots, \vartheta_n).$$

Ovšem $0 < \theta^2 < 1$ a dostáváme spor s volbou $\vartheta_1, \ldots, \vartheta_n$, protože máme $0 < |\operatorname{disc}(\xi_1, \ldots, \xi_n)| < |\operatorname{disc}(\vartheta_1, \ldots, \vartheta_n)|$. □

Definice 2.17. Množinu $\{\vartheta_1, \ldots, \vartheta_n\}$ nazveme *celočíselnou (celistvou) bází*, pokud $O_K = \{\sum_{i=1}^{n} z_i \cdot \vartheta_i;\ z_i \in \mathbb{Z}\}$. Existence plyne z věty 2.16.

Obecně existuje více různých celočíselných bází, ale následující věta ukazuje, že všechny mají stejný diskriminant.

Proto můžeme diskriminant použít jako invariant oboru celých algebraických čísel tělesa K, případně přímo číselného tělesa K. Označuje se $\operatorname{disc}(O_K)$, případně $\operatorname{disc}(K)$.

Věta 2.18. *Nechť K je číselné těleso s oborem celých algebraických čísel O_K. Nechť $\{\vartheta_1, \ldots, \vartheta_n\}$ a $\{\mu_1, \ldots, \mu_n\}$ jsou celočíselné báze O_K. Pak*

$$\operatorname{disc}(\vartheta_1, \ldots, \vartheta_n) = \operatorname{disc}(\mu_1, \ldots, \mu_n).$$

Důkaz. Protože $\{\mu_1, \ldots, \mu_n\}$ je báze, můžeme pomocí ní vyjádřit prvky druhé báze $\{\vartheta_1, \ldots, \vartheta_n\}$. Dostáváme

$$\begin{pmatrix} \vartheta_1 \\ \vdots \\ \vartheta_n \end{pmatrix} = A \cdot \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_n \end{pmatrix},$$

kde $A \in \mathbb{Z}^{n \times n}$. Pokud na tuto rovnici aplikujeme σ_i (vnoření K do $\mathbb{C}$, celkem jich máme n různých), zůstane matice A nezměněna. Můžeme tedy podobně jako v důkazu tvrzení 2.16 sestavit maticovou rovnici

$$\begin{pmatrix} \sigma_1(\vartheta_1) & \cdots & \sigma_n(\vartheta_1) \\ \vdots & \ddots & \vdots \\ \sigma_1(\vartheta_n) & \cdots & \sigma_n(\vartheta_n) \end{pmatrix} = A \cdot \begin{pmatrix} \sigma_1(\mu_1) & \cdots & \sigma_n(\mu_1) \\ \vdots & \ddots & \vdots \\ \sigma_1(\mu_n) & \cdots & \sigma_n(\mu_n) \end{pmatrix}.$$

Vezmeme-li druhou mocninu determinantů dostáváme

$$\operatorname{disc}(\vartheta_1, \ldots, \vartheta_n) = \det(A)^2 \operatorname{disc}(\mu_1, \ldots, \mu_n)$$

Zřejmě $\det(A)$ a diskriminanty leží v $\mathbb{Z}$. Stejný postup můžeme provést i naopak. Dostáváme, že $\operatorname{disc}(\mu_1, \ldots, \mu_n)$ a $\operatorname{disc}(\vartheta_1, \ldots, \vartheta_n)$ jsou asociované a zřejmě mají i stejné znaménka, proto se rovnají. □

Pokud $K = \mathbb{Q}[\vartheta]$ pro nějaké $\vartheta \in O_K$, bylo by vhodné mít celočíselnou bázi tvaru $\{1, \vartheta, \ldots, \vartheta^{n-1}\}$, ale takováto báze nemusí vždy existovat. Avšak můžeme alespoň najít vztah mezi $\operatorname{disc}(K)$ a $\operatorname{disc}(1, \vartheta, \ldots, \vartheta^{n-1})$.

Věta 2.19. *Nechť K je číselné těleso s oborem celých algebraických čísel O_K, $K = \mathbb{Q}[\vartheta]$, $\vartheta \in O_K$ a $n = [K : \mathbb{Q}]$. Potom platí*

$$\operatorname{disc}(1, \vartheta, \ldots, \vartheta^{n-1}) = \operatorname{disc}(K) \cdot [O_K : \mathbb{Z}[\vartheta]]^2.$$

Důkaz. Dokážeme obecnější případ. Nechť $R' \subseteq R$ jsou konečně generované $\mathbb{Z}$-moduly v K hodnosti n. Jedná se tedy o volné abelovské grupy hodnosti n. A proto existuje volná báze $\{\vartheta_1, \ldots, \vartheta_n\}$ $\mathbb{Z}$-modulu R taková, že pro vhodná $r_1, \ldots, r_n \in \mathbb{Z}$ je $\{r_1 \cdot \vartheta_1, \ldots, r_n \cdot \vartheta_n\} = \{\vartheta'_1, \ldots, \vartheta'_n\}$ báze podmodulu R'. Pak

$$\begin{pmatrix} \vartheta'_1 \\ \vartheta'_2 \\ \vdots \\ \vdots \\ \vartheta'_n \end{pmatrix} = \begin{pmatrix} r_1 & 0 & \cdots & \cdots & 0 \\ 0 & r_2 & 0 & \cdots & 0 \\ \vdots & & \ddots & & \vdots \\ \vdots & & & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & r_n \end{pmatrix} \begin{pmatrix} \vartheta_1 \\ \vartheta_2 \\ \vdots \\ \vdots \\ \vartheta_n \end{pmatrix}.$$

Podobně jako v předchozí větě dostáváme

$$\operatorname{disc}(\vartheta'_1, \ldots, \vartheta'_n) = (r_1 \cdot \ldots \cdot r_n)^2 \cdot \operatorname{disc}(\vartheta_1, \ldots, \vartheta_n) = [R : R']^2 \cdot \operatorname{disc}(R).$$

K důkazu věty stačí volit $R = O_K$ a $R' = \mathbb{Z}[\vartheta]$.

□

Další pojem, který definujeme, využijeme pouze okrajově.

Definice 2.20. Nechť R je komutativní obor integrity a nechť $f = \sum_{i=0}^{n} c_i x^i$ je polynom nad R stupně $n \geq 1$ s kořeny $\alpha_1, \ldots, \alpha_n$. Pak

$$\operatorname{disc}(f) = c_n^{2n-2} \prod_{i<j} (\alpha_i - \alpha_j)^2,$$

kde c_n je vedoucí koeficient, nazýváme *diskriminant polynomu f.*

Pojem diskriminantu polynomu jsou definovali z následujícího důvodu.

Věta 2.21. *Nechť K je číselné těleso s oborem celých algebraických čísel O_K, $K = \mathbb{Q}[\vartheta]$, $\vartheta \in O_K$ a f je minimální monický polynom ϑ nad $\mathbb{Z}$ stupně d. Potom platí*

$$\operatorname{disc}(1, \vartheta, \ldots, \vartheta^{n-1}) = \operatorname{disc}(f).$$

Důkaz. Označme $\sigma_1, \ldots, \sigma_n$ různá vnoření K do $\mathbb{C}$, restrikce $\sigma_i \mid \mathbb{Q} = \mathrm{id}_{\mathbb{Q}}$. Nejprve spočítáme determinant matice

$$\begin{vmatrix} \sigma_1(1) & \sigma_1(\vartheta) & \cdots & \sigma_1(\vartheta^{n-1}) \\ \vdots & \vdots & & \vdots \\ \sigma_n(1) & \sigma_n(\vartheta) & \cdots & \sigma_n(\vartheta^{n-1}) \end{vmatrix} = \begin{vmatrix} 1 & \sigma_1(\vartheta) & \cdots & \sigma_1(\vartheta)^{n-1} \\ \vdots & \vdots & & \vdots \\ 1 & \sigma_n(\vartheta) & \cdots & \sigma_n(\vartheta)^{n-1} \end{vmatrix} =$$

$$= \prod_{1 \le i < j \le n} (\sigma_j(\vartheta) - \sigma_i(\vartheta)).$$

Zřejmě se jedná o vandermondův determinant. Podle definice diskriminantu máme

$$\mathrm{disc}(1, \vartheta, \ldots, \vartheta^{n-1}) = |(\sigma_i(\alpha_j))_{i,j=1}^n|^2 = \prod_{1 \le i < j \le n} (\sigma_j(\vartheta) - \sigma_i(\vartheta))^2.$$

Protože $\sigma_i(\vartheta)$, $i = 1, \ldots, n$, jsou všechny různé kořeny polynomu f a vedoucí koeficient je jedna, dostáváme dokazovanou rovnost.

□

Nyní ještě uveďme několik vět, které využijeme v závěrečné fázi algoritmu. Přitom vyjdeme z [17].

Definice 2.22. Nechť K je číselné těleso, $K = \mathbb{Q}[\vartheta]$. Nechť B je aditivní podgrupa K. Definujeme *doplněk* B relativní ke stopě jako množinu všech $\alpha \in K$ takových, že $T(\alpha B) \subseteq \mathbb{Z}$, označujeme ho B'.

Je zřejmé, že B' je opět aditivní grupa. Dále pokud B a C jsou aditivní podgrupy a $B \subseteq C$, pak $C' \subseteq B'$.

Nechť V je vektorový prostor konečné dimenze nad tělesem T. Dále nechť $b : V \times V \to T$ je bilineární forma a definujme $(\Phi^b(x))(y) = b(x, y)$. Pak z lineární algebry víme, že homomorfismus Φ^b je izomorfismem vektorového prostoru V a jeho duálního prostoru V^*, právě když je bilineární forma b nedegenerovaná (matice bilineární formy je regulární).

Věta 2.23. *Nechť K je číselné těleso. Pak $b(x, y) = T(xy)$ je nedegenerovaná bilineární forma.*

Důkaz. Protože stopa je lineární forma a K je těleso, je zřejmé, že $b(x, y)$ je bilineární forma. Nechť $\mathbf{B}$ je matice formy b vzhledem k bázi $M = \{\alpha_1, \ldots, \alpha_n\}$ tělesa K nad $\mathbb{Q}$. Pak $\det(\mathbf{B}) = \det((T(\alpha_i \alpha_j)_{i,j=1}^n) = \mathrm{disc}(\alpha_1, \ldots, \alpha_n) \neq 0$ podle lemmatu 2.13 a věty 2.15. Přitom z lineární algebry víme, že pokud M' je jiná báze K nad $\mathbb{Q}$ a $\mathbf{P} \in \mathbb{Q}^{n \times n}$ je matice přechodu od báze M k bázi M', tak $\mathbf{C} = \mathbf{P}^T \mathbf{B} \mathbf{P}$ je matice bilineární formy b vzhledem k bázi M'. □

Nechť $\alpha \in K$, $\alpha \neq 0$. Pak podle poznámky výše a věty 2.23 je zobrazení $T(\alpha x)$ z K do $\mathbb{Q}$ prvkem duálního prostoru k prostoru K. Navíc indukuje izomorfismus K a jeho duálního prostoru. Proto můžeme duální bázi k nějaké bázi číselného tělesa K nad $\mathbb{Q}$ reprezentovat prvky ležícími v K. Tento poznatek využijeme v následujících větách, kde budeme duální bází myslet reprezentaci duální báze v tělese K.

Věta 2.24. *Nechť K je číselné těleso s bází $\{\alpha_1, \ldots, \alpha_n\}$ nad $\mathbb{Q}$ a nechť*

$$B = \alpha_1\mathbb{Z} + \ldots + \alpha_n\mathbb{Z}.$$

Pak

$$B' = \alpha_1'\mathbb{Z} + \ldots + \alpha_n'\mathbb{Z},$$

kde $\{\alpha_1', \ldots, \alpha_n'\} \subseteq K$ je duální báze k bázi $\{\alpha_1, \ldots, \alpha_n\}$.

Důkaz. Nechť $\alpha \in B'$ a $\alpha = a_1\alpha_1' + \ldots + a_n\alpha_n'$, kde $a_i \in \mathbb{Q}$. Pak $T(\alpha\alpha_i) = a_i$ a tedy podle definice doplňku $a_i \in \mathbb{Z}$ pro všechna $i \leq n$. Z toho plyne, že $B' \subseteq \alpha_1'\mathbb{Z} + \ldots + \alpha_n'\mathbb{Z}$. Naopak pro $r \in \mathbb{Z}$ platí

$$T(r\alpha_i'B) = rT(\alpha_i'B) \subseteq \mathbb{Z}.$$

□

Věta 2.25. *Nechť $K = \mathbb{Q}[\alpha]$ je číselné těleso a nechť $f(x)$ je minimální polynom α nad $\mathbb{Q}$ stupně n. Označme $f'(x)$ formální derivaci polynomu $f(x)$ a*

$$\frac{f(x)}{x - \alpha} = \beta_0 + \beta_1 x + \ldots + \beta_{n-1}x^{n-1}.$$

Pak duální báze k bázi $\{1, \alpha, \ldots, \alpha^{n-1}\}$ je

$$\left\{\frac{\beta_0}{f'(\alpha)}, \ldots, \frac{\beta_{n-1}}{f'(\alpha)}\right\}.$$

Důkaz. Nechť $\alpha_1, \ldots, \alpha_n \in \mathbb{C}$ jsou různé kořeny polynomu $f(x)$. Dokážeme, že

$$\sum_{i=1}^{n} \frac{f(x)}{x - \alpha_i} \frac{\alpha_i^r}{f'(\alpha_i)} = x^r, \;\; 0 \leq r \leq n-1.$$

Označme rozdíl pravé a levé strany rovnosti polynomem $g(x)$. Polynom $g(x)$ má stupěn $n - 1$, přitom ale

$$g(\alpha_i) = \left(\prod_{j \neq i}(\alpha_i - \alpha_j)\right) \frac{\alpha_i^r}{f'(\alpha_i)} - \alpha_i^r = f'(\alpha_i)\frac{\alpha_i^r}{f'(\alpha_i)} - \alpha_i^r = 0.$$

Tedy $g(x)$ má alespoň n kořenů a musí být proto identicky roven nule. Nechť σ_j jsou vnoření K do $\mathbb{C}$, pro která platí, že restrikce $\sigma_j \mid \mathbb{Q} = \mathrm{id}_{\mathbb{Q}}$, $j = 1, \ldots, n$. Pro libovolný polynom $h(x) = \sum_{i=1}^{k} \gamma_i x^i \in K[x]$ definujme $\sigma_i(h(x)) = \sum_{i=1}^{k} \sigma_j(\gamma_i)x^i$. Pokud definujeme $T(h(x)) = \sum_{j=1}^{n} \sigma_j(h(x))$, pak

$$T\left(\frac{f(x)}{x-\alpha}\frac{\alpha^r}{f'(\alpha)}\right) = T\left((\beta_0+\beta_1 x+\ldots+\beta_{n-1}x^{n-1})\frac{\alpha^r}{f'(\alpha)}\right) =$$
$$= \sum_{j=1}^{n}\frac{f(x)}{x-\alpha_j}\frac{\alpha_j^r}{f'(\alpha_j)} = x^r.$$

Pokud porovnáme koeficienty u stejných mocnin x, dostaneme

$$T\left(\alpha^j\frac{\beta_i}{f'(\alpha)}\right) = \delta_{ij}.$$

Tedy $\{\beta_0/f'(\alpha),\ldots,\beta_{n-1}/f'(\alpha)\}$ je duální báze.

□

Na závěr dostáváme požadovanou větu.

Věta 2.26. *Nechť $K=\mathbb{Q}[\vartheta]$ je číselné těleso s oborem celých algebraických čísel O_K. Nechť $f(x)=\sum_{i=0}^n a_i x^i$ je minimální polynom ϑ nad $\mathbb{Z}$. Pak*

$$f'(\vartheta)O_K \subseteq \mathbb{Z}[\vartheta].$$

Důkaz. Protože $T(O_K)\subseteq\mathbb{Z}$ podle lemmatu 2.8, dostáváme $O_K\subseteq O_K'$. Dále víme, že $\mathbb{Z}[\vartheta]\subseteq O_K$, tedy $O_K'\subseteq\mathbb{Z}[\vartheta]'$. Nyní ukážeme, že $\mathbb{Z}[\vartheta]'=(1/f'(\vartheta))\mathbb{Z}[\vartheta]$ z toho již věta snadno plyne. Z předcházející věty 2.25 víme, že $\mathbb{Z}[\vartheta]'$ má bázi $\{\beta_0/f'(\vartheta),\ldots,\beta_{n-1}/f'(\vartheta)\}$, kde

$$\frac{f(x)}{x-\vartheta} = \beta_0+\beta_1 x+\ldots+\beta_{n-1}x^{n-1}.$$

Vidíme tedy, že $a_i=\beta_{i-1}+\vartheta\beta_i$. Rekurzí dostáváme

$$\begin{aligned}\beta_{n-1} &= a_n = 1\\ \beta_{n-2}-\vartheta\beta_{n-1} &= a_{n-1}\\ &\vdots\end{aligned}$$

Z toho je patrné, že β_i lze získat pomocí celočíselné kombinace mocnin ϑ. Podobně z těchto vztahů plyne, že mocniny ϑ lze získat z celočíselných kombinací β_i. Například máme

$$\begin{aligned}\beta_{n-2}-a_{n-1} &= \vartheta\\ \beta_{n-3}-\vartheta\beta_{n-2} &= a_{n-2}\\ \beta_{n-3}-\vartheta(a_{n-1}+\vartheta) &= a_{n-2}\\ \beta_{n-3}-a_{n-1}(\beta_{n-2}-a_{n-1})-a_{n-2} &= \vartheta^2\end{aligned}$$

Tedy modul generovaný prvky $1, \vartheta, \ldots, \vartheta^{n-1}$ je stejný jako modul generovaný prvky $\beta_0, \beta_1, \ldots, \beta_{n-1}$.

□

Následující lemma využijeme v některých důkazech.

Lemma 2.27. *Nechť I je nenulový ideál oboru celých algebraických čísel O_K číselného tělesa K. Pak okruh O_K/I je konečný.*

Důkaz. O_K je volná abelovská grupa hodnosti $n = [K : \mathbb{Q}]$. Nechť $\{\vartheta_1, \ldots, \vartheta_n\}$ je celočíselná báze O_K. Pokud nalezneme $m \in \mathbb{Z}$ takové, že $m \in I$, pak $(m) \subseteq I$ a tedy $O_K/I \subseteq O_K/(m)$. Přitom $O_K/(m) = \{\sum_{i=1}^{n} a_i\vartheta;\ a_i \in \mathbb{Z}_m\}$. Stačí proto ukázat, že každý nenulový ideál I obsahuje nějaké přirozené číslo. Nechť $\vartheta \in I \subseteq O_K$, $\vartheta \neq 0$. Pak existují navzájem nesoudělné koeficienty $b_i \in \mathbb{Z}$ takové, že $\vartheta^k + b_{k-1}\vartheta^{k-1} + \ldots + b_0 = 0$ a k je nejmenší možné. Přitom $|b_0| = |N(\vartheta)| \neq 0$, protože se jedná o nenulový prvek a tedy $|b_0| \in I$ je hledané přirozené číslo. □

2.2 Dedekindovy obory

Definice 2.28. Obor integrity R nazveme *Dedekindův*, pokud jsou splněny následující podmínky:

(i) R je noetherovské;

(ii) každý nenulový prvoideál je maximální;

(iii) R je celistvě uzavřené ve svém podílovém tělese T.

Podmínka (iii) znamená, že pokud $a \in T$ a zároveň existuje monický polynom $f \in R[x]$, $f(a) = 0$, pak $a \in R$. Podmínka (i) je ekvivaletní každé z následujících podmínek:

- každý ideál je konečně generovaný;
- každá neprázdná množina ideálů má maximální prvek.

Dedekindovy obory jsou pro nás velmi důležité. Postupně ukážeme, že každý ideál v R je možné jednoznačně (až na pořadí činitelů) vyjádřit jako součin prvoideálů. Toto jistě platí například v $\mathbb{Z}$, kde každý ideál je hlavní a rozklad jeho generujícího prvku na součin prvočísel nám poskytuje i rozklad tohoto ideálu na součin příslušných prvoideálů. Můžeme tedy chápat rozklad ideálů na součin prvoideálů jako jisté zobecnění faktorizace. Důvod, proč se zabýváme Dedekindovými obory, je následující:

Věta 2.29. *Obor celých algebraických čísel je Dedekindův.*

Důkaz. Potřebujeme dokázat, že obor celých algebraických čísel O_K číselného tělesa K splňuje všechny tři podmínky z definice Dedekindova oboru.

(i) Obor O_K je volná abelovská grupa konečné hodnosti, proto každá podgrupa O_K je také volná s konečnou hodností, tedy každý ideál $I \subseteq O_K$ má aditivní strukturu rovnu volné abelovské grupě konečné hodnosti. Z toho plyne, že ideál I je konečně generovaný i jako ideál.

(ii) Nechť P je libovolný nenulový prvoideál z O_K. Stačí ukázat, že O_K/P je těleso. Podle lemmatu 2.27 víme, že O_K/P je konečný okruh, a protože P je prvoideál, je tento okruh oborem integrity. Avšak konečný obor integrity je těleso (pro $a \in (O_K/P)^*$ definujme $\varphi_a : (O_K/P)^* \to (O_K/P)^*, \varphi_a(b) = ab$. Toto zobrazení je prosté a díky konečnosti i na, tedy existuje $c \in O_K/P$, že $ac = 1$).

(iii) Nechť T je podílové těleso O_K, $\alpha \in T$, a zároveň existuje monický polynom $f = \sum_{i=1}^{k} \vartheta_i x^i \in O_K[x]$, $f(\alpha) = 0$. Potřebujeme ukázat, že $\alpha \in O_K$. K tomu stačí nalézt monický polynom nad $\mathbb{Z}$, který má α jako kořen. Zřejmě $O_K[\vartheta_1, \ldots, \vartheta_k, \alpha]$ je konečně generovaný okruh nad O_K a tedy i konečně generovaný nad $\mathbb{Z}$. Můžeme tedy použít lemma 2.3 k nalezení monického polynomu nad $\mathbb{Z}$ s kořenem α.

□

V oboru celých algebraických čísel nemůžeme jednoznačně rozložit jednotlivé prvky, ale dokážeme jednoznačně rozložit ideály, které generují. Tato vlastnost nám umožní sestrojit algoritmus číselného síta.

K dokázání nastíněného tvrzení potřebujeme několik pomocných lemmat.

Lemma 2.30. *Každý nenulový ideál v Dedekindově oboru R obsahuje součin prvoideálů.*

Důkaz. Budeme postupovat sporem. Díky vlastnosti (i) Dedekindova oboru R má množina ideálů, které neobsahují součin prvoideálů, maximální prvek M. Zřejmě se nejedná o prvoideál, proto existují prvky $a, b \in R \setminus M$ takové, že $ab \in M$. Ideály $M+(a)$, $M+(b)$ jsou ostře větší než M, proto již obsahují součin prvoideálů. Ale potom součin prvoideálů obsahuje i ideál $(M + (a))(M + (b))$, který je částí M. A to je hledaný spor. □

Lemma 2.31. *Nechť I je vlastní ideál v Dedekindově oboru R. Pak existuje $\alpha \in T \setminus R$ s vlastností $\alpha I \subset R$, kde T je podílové těleso R.*

Důkaz. Nechť a je nenulový prvek I. Podle lemmatu 2.30 obsahuje hlavní ideál (a) součin prvoideálů, označme je $P_1, \ldots, P_k$, přitom požadujeme, aby k bylo minimální. Ideál I je obsažen v nějakém maximálním ideálu P, který je zároveň i prvoideál, proto P obsahuje některý z prvoideálů $P_1, \ldots, P_k$, nechť je to P_1. Díky vlastnosti 2 Dedekindova oboru je nutně $P = P_1$. Protože hlavní ideál (a) nemůže obsahovat součin méně než k prvoideálů, existuje $b \in P_2 \cdot \ldots \cdot P_k \setminus (a)$. Z $b \in P_2 \cdot \ldots \cdot P_k$ a $I \subseteq P_1$ dostáváme $bI \subseteq P_1 \cdot \ldots \cdot P_k \subseteq (a)$, tedy $\frac{b}{a} I \subseteq R$ a $\frac{b}{a} \notin R$. □

Věta 2.32. *Nechť I je nenulový ideál v Dedekindově oboru R. Pak existuje ideál J takový, že IJ je hlavní ideál v R.*

Důkaz. Nechť a je nenulový prvek I. Definujme $J = \{b \in R; bI \subseteq (a)\}$. Zřejmě J je nenulový ideál v R a platí $IJ \subseteq (a)$. Ukážeme, že nastává rovnost. Uvažujme množinu $M = a^{-1}IJ \subseteq R$. Snadno se ukáže, že tato množina je ideál. Pokud $M = R$, pak $IJ = (a)$, jinak je M vlastní ideál R a podle lemmatu 2.31 existuje $\alpha \in T \setminus R$, $\alpha M \subseteq R$, T podílové těleso R. Využijeme podmínky (iii), podle které je Dedekindův obor celistvě uzavřený ve svém podílovém tělese, a ukážeme, že $\alpha \in R$, což bude spor. Protože $a \in I$, máme $a^{-1}aJ \subseteq M$, tedy $J \subseteq M$ a $\alpha J \subseteq \alpha M \subseteq R$. Dále z $\alpha a^{-1}IJ \subseteq R$ dostáváme $\alpha IJ \subseteq (a)$. Použijeme-li definici J, vidíme, že $\alpha J \subseteq J$. Podle podmínky (i) můžeme najít konečnou množinu generátorů $\{a_1, \dots a_k\}$ ideálu J a opět použijeme lemma 2.3 k nalezení monického polynomu nad R s kořenem α. To je hledaný spor. □

Definice 2.33. Nechť A a B jsou ideály v Dedekindově oboru R. Řekneme, že ideál A dělí ideál B, pokud $B = AC$ pro nějaký ideál $C \subseteq R$. Budeme značit $A|B$.

Jako důsledky věty 2.32 dostáváme následující lemmata:

Lemma 2.34. *Nechť A, B a C jsou ideály v Dedekindově oboru R. Pokud $AB = AC$, pak $B = C$.*

Důkaz. Podle věty 2.32 existuje ideál D takový, že $AD = (a)$, $a \in R$. Z $DAB = DAC$ plyne $aB = aC$. Protože zřejmě existuje bijekce mezi B a aB, a zároveň mezi aC a C, dostáváme $B = C$. □

Lemma 2.35. *Nechť A a B jsou ideály v Dedekindově oboru R. Potom $B \subseteq A$, právě když $A|B$.*

Důkaz. Pokud $A|B$, pak triviálně máme $B \subseteq A$. Naopak nechť $B \subseteq A$. Pro ideál A podle věty 2.32 existuje ideál D takový, že $AD = (a)$, $a \in R$. Máme $(a) = AD \supseteq BD$, proto množina $C = a^{-1}DB \subseteq R$. Navíc se jedná o ideál a zřejmě $AC = B$. □

Nyní již snadno dokážeme požadovanou větu.

Věta 2.36. *Každý ideál v Dedekindově oboru R lze jednoznačně rozložit na součin prvoideálů.*

Důkaz. Nejprve dokážeme existenci rozkladu. Budeme postupovat sporem. Množina všech ideálů, které nelze rozložit na součin prvoideálů v R, má podle podmínky (i) maximální prvek M. Ideál M je obsažen v nějakém maximálním ideálu P, který je i prvoideálem. Tedy podle lemmatu 2.35 existuje ideál I takový, že $M = IP$, a podle téhož musí M být částí I. Navíc M je vlastní podmnožina I (neboť podle 2.34 máme $PM = PI \Rightarrow PM = M \Rightarrow PM =$

$RM \Rightarrow P = R$, spor). Tedy I je ostře větší než M a je ho možné rozložit na součin prvoideálů. Ale pak lze rozložit i M, což je spor.

Nyní dokážeme jednoznačnost. Nechť $I = P_1 \cdot \ldots \cdot P_k = Q_1 \cdot \ldots \cdot Q_l$, kde P_i, Q_i jsou prvoideály. Potom $Q_1 \cdot \ldots \cdot Q_l \subseteq P_1$, a proto pro nějaké j, $1 \leq j \leq l$, máme $Q_j \subseteq P_1$. Předpokládejme, že $j = 1$. Podle podmínky (ii) musí být $P_1 = Q_1$ a použijeme-li lemma 2.34, dostaneme $P_2 \cdot \ldots \cdot P_k = Q_2 \cdot \ldots \cdot Q_l$. Takto můžeme postupně krátit, až dostaneme, že $k = l$ a $P_i = Q_i$, $1 \leq i \leq k$. □

Předchozí věta nám umožňuje definovat na množině všech ideálů v Dedekindově oboru R pojmy největší společný dělitel (označme gcd) a nejmenší společný násobek (označme lcm) tak, jak je známe z celých čísel. Přitom je zřejmé, že vzhledem k relaci inkluze jsou pojmy největší a nejmenší myšleny opačně. Dostáváme tedy následující vzorečky pro dva ideály I, J z R:

$$\begin{aligned} \gcd(I, J) &= I + J, \\ \operatorname{lcm}(I, J) &= I \cap J. \end{aligned}$$

Pomocí největšího společného dělitele můžeme dokázat následující větu.

Věta 2.37. *Nechť I je ideál v Dedekindově oboru R a nechť a je nenulový prvek I. Pak existuje $b \in I$ takové, že $I = (a, b)$.*

Důkaz. Využijeme pojmu největší společný dělitel a nalezneme $b \in R$, že $I = \gcd((a), (b)) = (a) + (b)$, tedy zřejmě $b \in I$. Nechť $I = P_1^{e_1} \cdot \ldots \cdot P_k^{e_k}$, kde P_i jsou prvoideály. Pak jistě hlavní ideál (a) je dělitelný všemi P_i, $1 \leq i \leq k$, dále je dělitelný i dalšími prvoideály. Z nich vybereme jen ty, které jsou různé od P_i, označme je $Q_1, \ldots, Q_l$. Musíme najít prvek b tak, aby neležel v žádném z prvoideálů $Q_1, \ldots, Q_l$ (pak jimi nebude hlavní ideál (b) dělitelný) a hlavní ideál (b) byl také dělitelný $P_i^{e_i}$, $1 \leq i \leq k$, ale ne ve vyšší mocnině. Dostáváme

$$b \in \Big(\bigcap_{i=1}^{k} (P_i^{e_i} \setminus P_i^{e_{i+1}})\Big) \cap \Big(\bigcap_{i=1}^{l} (R \setminus Q_i)\Big).$$

Takové b můžeme nalézt pomocí Čínské věty o zbytku. Abychom ji mohli použít, potřebujeme vědět, že mocniny prvoideálů P_i, $1 \leq i \leq k$, a prvoideály Q_j, $1 \leq j \leq l$, jsou po dvou komaximální. To je zřejmé, pokud si uvědomíme, že součet ideálů je jejich největší společný dělitel, a že jako jednotku chápeme R. Díky jednoznačnosti rozkladu na prvoideály musí být $P_i^{e_i} \setminus P_i^{e_{i+1}}$ neprázdné, můžeme tedy vybrat nějaké $b_i \in P_i^{e_i} \setminus P_i^{e_{i+1}}$, $1 \leq i \leq k$. Pro b tedy máme

$$\begin{aligned} b &\equiv b_i \mod P_i^{e_i+1}, \quad i = 1, \ldots, k, \\ b &\equiv 1 \mod Q_j, \quad j = 1, \ldots, l. \end{aligned}$$

□

2.3 Rozklady na prvoideály

Nyní máme dva cíle. Zaprvé, jak nalézt prvoideály v nějakém oboru celých algebraických čísel. Zadruhé, jak lze určit rozklad ideálu na prvoideály. V následujícím textu budou K, L (popřípadě M) označovat číselná tělesa, R, S (T) jejich obory celých algebraických čísel a P, Q (U) jejich prvoideály. Pro zjednodušení zápisu budeme v této části pojmem prvoideál označovat pouze nenulový prvoideál. Nejprve uvažujme následující situaci. Mějme prvoideál P v R a L číselné nadtěleso K. Jak se rozkládá ideál PS v S? Postupně nalezneme odpověď na tuto otázku a tím splníme první cíl. Pro prvoideály P v R a Q v S dostáváme následující snadnou ekvivalenci:

$$Q|PS \Leftrightarrow PS \subseteq Q \Leftrightarrow P \subseteq Q \Leftrightarrow P = Q \cap R \Leftrightarrow P = Q \cap K.$$

Pomocí této ekvivalence již není těžké dokázat zajímavou větu:

Věta 2.38. *Pro každý prvoideál P v R položme $M_P = \{Q;\ Q$ prvoideál v S, $P \subseteq Q\ \}$. Pokud P probíhá množinu všech prvoideálů R, tak množiny M_P poskytují rozklad množiny všech prvoideálů S na neprázdné, disjunktní a konečné podmnožiny.*

Důkaz. Nejprve ukážeme, že M_P je neprázdné. Budeme postupovat sporem. Nechť tedy pro nějaký prvoideál P je M_P prázdná množina. Pak musí $PS = S$. Podle lemmatu 2.31 existuje $\alpha \in T \setminus R$ (T je podílové těleso R), že $\alpha P \subseteq R$. Dostáváme $\alpha PS \subseteq RS = S$, a protože $1 \in PS$, musí být $\alpha \in S$. Ovšem to znamená, že α je algebraické celé číslo a tedy $\alpha \in R$. Což je hledaný spor.

Z ekvivalence výše plyne, že prvoideál $Q \subseteq S$ leží v M_P určené prvoideálem $P = R \cap Q$. Tento prvoideál je nenulový, neboť můžeme zvolit nenulové $\alpha \in Q \subseteq S$ o němž víme, že $N_K^L(\alpha) \in R$. Dále $N_K^L(\alpha) = \prod_\sigma \sigma(\alpha) = \alpha \prod_{\sigma \neq id_L} \sigma(\alpha) = \alpha\beta$, kde $\beta \in S$, a proto $N_K^L(\alpha) \in Q$. Přitom tato norma je nenulová. Současně $P \neq R$, neboť bychom dostali $1 \in Q$. Z tohoto plyne disjunktnost rozkladu.

Konečnost M_P pro prvoideál P je zřejmá z toho, že S je Dedekindův obor a množina M_P obsahuje všechny prvoideálové dělitele PS. □

Přejděme nyní k definicím, které popisují rozklad ideálu PS.

Definice 2.39. *Ramifikačním (štěpícím, větvícím) indexem* nazýváme největšího mocnitele e prvoideálu Q v S, pro kterého Q^e dělí PS, kde P je prvoideál v R. Označujeme: $\mathrm{e}(Q|P)$.
Řekneme, že P je *ramifikované v S*, pokud $\mathrm{e}(Q|P) > 1$ pro nějaké Q.

Když $Q|PS$, tak je běžné říkat, že P *leží pod* Q nebo Q *leží nad* P. Víme, že P jakožto nenulový prvoideál je zároveň i maximálním ideálem. Proto R/P je těleso, navíc konečné podle lemmatu 2.27 (podobně pro S/Q). Dále je možné těleso $R/P = R/(P \cap Q) \cong (R + Q)/Q$ vnořit do tělesa S/Q a tedy dimenze S/Q nad R/P je konečná. Pozorování shrnuje definice 2.40.

Definice 2.40. *Stupněm setrvačnosti (nehybnosti) f* nazýváme dimenzi S/Q nad R/P. Označujeme: $\mathrm{f}(Q|P)$.

Uveďme pár jednoduchých pozorování. Pokud $K = \mathbb{Q}$, $R = \mathbb{Z}$ a $P = p\mathbb{Z}$, pak dostáváme pro $L \supseteq K$, $S = O_L$, $Q \supseteq P$, že $|S/Q| = p^f$, $f = \mathrm{f}(Q|P)$. Přitom zřejmě $f \leq n = [L : K]$.

Dále uvažujme prvoideály $P \subseteq Q \subseteq U$ a po řadě jejich obory celých algebraických čísel $R \subseteq S \subseteq T$. Pak platí:

$$\mathrm{e}(U|P) = \mathrm{e}(U|Q) \cdot \mathrm{e}(Q|P),$$
$$\mathrm{f}(U|P) = \mathrm{f}(U|Q) \cdot \mathrm{f}(Q|P).$$

Definice 2.41. *Norma ideálu I* v R je $\mathcal{N}(I) = |R/I|$. Někdy se můžeme setkat s označením $\| I \|$, případně pro rozlišení oboru $\mathcal{N}_R(I)$.

Norma ideálu má pro nás velký význam, protože s její pomocí dokážeme určit dělitele I. Hlavní vlastnosti normy jsou, jak očekáváme, následující:

Věta 2.42. *Nechť K a L jsou číselná tělesa, $L \supseteq K$, $n = [L : K]$, R a S jejich obory celých algebraických čísel.*

(i) Nechť I a J jsou ideály v R. Pak $\mathcal{N}(IJ) = \mathcal{N}(I) \cdot \mathcal{N}(J)$.

(ii) Nechť I je ideál v R. Pak pro ideál IS v S platí: $\mathcal{N}_S(IS) = (\mathcal{N}_R(I))^n$.

Důkaz. (i) Důkaz rozdělíme na dvě části. Nejprve ukážeme, že tvrzení platí pro nesoudělné ideály a potom dokážeme, že $\mathcal{N}(P^m) = (\mathcal{N}(P))^m$ pro každé $m \geq 1$. Dostaneme

$$\mathcal{N}(P_1^{m_1} \cdot \ldots \cdot P_r^{m_r}) = \mathcal{N}(P_1)^{m_1} \cdot \ldots \cdot \mathcal{N}(P_r)^{m_r}.$$

Z rozkladu I a J na prvoideály již tvrzení okamžitě plyne.

Předpokládejme, že I a J jsou nesoudělné. Pak zřejmě platí $I + J = R$ a $I \cap J = IJ$. Pomocí Čínské věty o zbytku dostáváme izomorfismus

$$R/IJ \cong R/I \times R/J.$$

Z toho již plyne $\mathcal{N}(IJ) = \mathcal{N}(I) \cdot \mathcal{N}(J)$.

Nyní uvažujme prvoideál $P \in R$. Ukážeme, že $\mathcal{N}(P^m) = (\mathcal{N}(P))^m$ pro $m \geq 1$. Mocniny prvoideálu P tvoří posloupnost ideálů $R \supset P \supset P^2 \supset \ldots \supset P^m$. Pokud dokážeme, že $\mathcal{N}(P) = |P^k/P^{k+1}|$, jsme hotovi, neboť $\mathcal{N}(P^m) = |R/P^m| = |R/P| \cdot |P/P^2| \cdot \ldots \cdot |P^{m-1}/P^m| = (\mathcal{N}(P))^m$. Ideály P^k uvažujeme dále pouze jako aditivní grupy. Pokud vybereme prvek $\vartheta \in P^k \setminus P^{k+1}$, dostáváme grupový izomorfismus

$$R/P \cong \vartheta R/\vartheta P.$$

Protože $\vartheta R = (\vartheta) \subset P^k$ můžeme uvažovat homomorfismus

$$(\vartheta) \to P^k/P^{k+1},$$

kde jádro je zřejmě $(\vartheta) \cap P^{k+1}$ a obraz je $((\vartheta) + P^{k+1})/P^{k+1}$. Prvek ϑ byl zvolen tak, aby P^k byla největší mocnina P dělící (ϑ). Proto

$$(\vartheta) \cap P^{k+1} = \operatorname{lcm}((\vartheta), P^{k+1}) = \vartheta P,$$
$$(\vartheta) + P^{k+1} = \gcd((\vartheta), P^{k+1}) = P^k.$$

Celkově dostáváme grupové izomorfismy

$$R/P \cong \vartheta R/\vartheta P \cong P^k/P^{k+1}$$

a tedy $\mathcal{N}(P) = |P^k/P^{k+1}|$.

(ii) S využitím předchozího bodu bude stačit dokázat tvrzení pro libovolný prvoideál $P \subseteq R$. Zřejmě S/PS je vektorový prostor nad R/P. Chceme dokázat, že jeho dimenze je n. Nejprve ukážeme, že dimenze není větší než n. K tomu stačí ukázat, že libovolných $n+1$ prvků je lineárně závislých nad R/P. Označme tyto prvky $\mu_1, \ldots, \mu_{n+1} \in S$. Tyto prvky jsou jistě lineárně závislé nad K a tedy i nad R (stačí vynásobit dostatečně velkým číslem). Dostáváme $\vartheta_1 \cdot \mu_1 + \ldots + \vartheta_{n+1} \cdot \mu_{n+1} = 0$, kde $\vartheta_i \in R$, ne všechny nulové. Potřebujeme ukázat, že alespoň jedno neleží v P, pak dostaneme lineární závislost i nad R/P. Postupujme sporem. Nechť jsou všechny takovéto možné $(n+1)$-tice $\{\vartheta_1, \ldots, \vartheta_{n+1}\} \subseteq P$. Vyberme libovolnou z nich a uvažujme ideál $(\vartheta_1, \ldots, \vartheta_{n+1}) \subseteq P$. Podle věty 2.32 existuje ideál I takový, že $(\vartheta_1, \ldots, \vartheta_{n+1}) \cdot I$ je hlavní ideál. Označme ho (ϑ). Pak ovšem $(\vartheta_1, \ldots, \vartheta_{n+1}) \cdot I = \vartheta R \nsubseteq \vartheta P$, protože $P \subsetneq R$. Zvolme $\xi \in I$ tak, aby $\xi(\vartheta_1, \ldots, \vartheta_{n+1}) \nsubseteq \vartheta P$. Dostaneme $\frac{\xi}{\vartheta}(\vartheta_1, \ldots, \vartheta_{n+1}) \nsubseteq P$ a přitom $\frac{\xi}{\vartheta}(\vartheta_1, \ldots, \vartheta_{n+1}) \subseteq R$. To je hledaný spor. Dimenze je tedy nejvýše n.

Nechť $p\mathbb{Z} = P \cap \mathbb{Z}$ a nechť P_i, $i = 1, \ldots, r$, jsou všechny prvoideály z R nad prvočíslem p. Víme, že S/P_iS je vektorový prostor nad R/P_i dimenze $n_i \leq n$. Ukážeme, že rovnost platí pro všechna i, tedy i pro P. Označme $e_i = \mathrm{e}(P_i|(p))$ a $f_i = \mathrm{f}(P_i|(p))$. Nechť m je dimenze K nad $\mathbb{Q}$. Z věty 2.16 plyne, že $\mathcal{N}(pR) = p^m$. Zřejmě

$$p^m = \mathcal{N}(pR) = \mathcal{N}(P_1^{e_1} \cdot \ldots \cdot P_r^{e_r}) = \prod_{i=1}^{r} \mathcal{N}(P_i)^{e_i} = \prod_{i=1}^{r} p^{f_i e_i}.$$

Protože $pR = \prod_{i=1}^{r} P_i^{e_i}$, máme $pS = \prod_{i=1}^{r} (P_iS)^{e_i}$ a podle předchozího dostáváme

$$\mathcal{N}(pS) = \prod_{i=1}^{r} \mathcal{N}(P_i S)^{e_i} = \prod_{i=1}^{r} \mathcal{N}(P_i)^{n_i e_i} = \prod_{i=1}^{r} (p^{f_i})^{n_i e_i}.$$

Také platí $\mathcal{N}(pS) = p^{mn}$, takže celkově $mn = \sum_{i=1}^{r} f_i n_i e_i$. A protože $n_i \leq n$ a $\sum_{i=1}^{r} f_i e_i = m$, dostáváme $n_i = n$ pro všechna i.

□

Věta 2.42 nám za předpokladu, že známe normu ideálu I v S, umožňuje zúžit okruh možných dělitelů I. Pokud nějaký prvoideál Q v S dělí I, pak nutně tento prvoideál musí ležet nad prvočíslem p, které dělí normu ideálu I. Zatím ještě nevíme, jaký prvoideál ležící nad p to je, ani jak jednoduše zjistíme normu nějakého ideálu v S. Následující věta nám však dává jiný způsob výpočtu normy alespoň u hlavních prvoideálů.

Věta 2.43. *Nechť R je obor celých algebraických čísel číselného tělesa K, $\vartheta \in R$, $\vartheta \neq 0$. Pak pro hlavní ideál (ϑ) platí:*

$$\mathcal{N}((\vartheta)) = |N_{\mathbb{Q}}^{K}(\vartheta)|.$$

Důkaz. Uvažujme normální rozšíření M číselného tělesa K. Nechť T je obor celých algebraických čísel tělesa M. Označme σ libovolné vnoření K do $\mathbb{C}$. Pokud σ rozšíříme na automorfismus M, zřejmě platí $\sigma(T) = T$, a tedy dostáváme

$$\mathcal{N}(\sigma(\vartheta)T) = \mathcal{N}(\vartheta T).$$

Označme $a = N_{\mathbb{Q}}^{K}(\vartheta) = \prod_{i=1}^{n} \sigma_i(\vartheta) \in \mathbb{Z}$. Použijeme-li větu 2.42, dostáváme

$$\mathcal{N}(aT) = \prod_{i=1}^{n} \mathcal{N}(\sigma_i(\vartheta)T) = \mathcal{N}(\vartheta T)^n.$$

Je-li $m = [M : K]$, pak $\mathcal{N}(aT) = |a|^{mn}$ a podle věty 2.42 je $\mathcal{N}(\vartheta T) = \mathcal{N}(\vartheta R)^m$. Dostáváme $|a| = \mathcal{N}(\vartheta R)$.

□

Obecně norma prvku v číselném tělese také není snadno spočitatelná, ale pro určitá čísla ji podle lemmatu 2.11 umíme velmi rychle spočítat.

Následující věta je označována jako fundamentální identita pro číselná tělesa.

Věta 2.44. *Nechť po řadě R, S jsou obory celých algebraických čísel číselných těles K, L, $[L : K] = n$. Nechť P je nenulový prvoideál v R a*

$$PS = Q_1^{e_1} \cdot \ldots \cdot Q_k^{e_k},$$

kde Q_i jsou prvoideály v S a $f_i = \mathrm{f}(Q_i|P)$. Pak platí:

$$n = \sum_{i=1}^{k} e_i f_i.$$

Důkaz. K důkazu použijeme větu 2.42. Víme, že $\mathcal{N}(PS) = \mathcal{N}(P)^n$ a zároveň

$$\mathcal{N}(PS) = \mathcal{N}(\prod_{i=1}^{k} Q_i^{e_i}) = \prod_{i=1}^{k} \mathcal{N}(Q_i)^{e_i} = \prod_{i=1}^{k} \mathcal{N}(P)^{f_i e_i}.$$

Tedy $n = \sum_{i=1}^{k} e_i f_i$.

□

Věta 2.45. *Nechť $K = \mathbb{Q}$, $R = \mathbb{Z}$ a L je číselné těleso s oborem celých algebraických čísel S. Je-li $p \in \mathbb{Z}$ prvočíslo, pak hlavní ideál $p\mathbb{Z} \subseteq \mathbb{Z}$ je ramifikovaný v S, právě když $p|\operatorname{disc}(S)$.*

Důkaz. Dokážeme pouze, že pokud $p\mathbb{Z}$ je ramifikované v S, pak p dělí $\operatorname{disc}(S)$. Opačná implikace je složitější, jeden z možných důkazů lze nalézt v [22] kapitola 4.

Nechť Q je prvoideál v S nad $p\mathbb{Z}$ s $\mathrm{e}(Q|p\mathbb{Z}) > 1$, nechť I je ideál v S, pro který platí $pS = QI$, tedy I je dělitelný všemi prvoideály v S nad $p\mathbb{Z}$. Označme $\{\vartheta_1, \ldots, \vartheta_n\}$ celistvou bázi S. Vybereme libovolné $\vartheta \in I \setminus pS$. O ϑ víme, že leží v každém prvoideálu v S nad p, ale neleží v pS. Je-li $\vartheta = a_1\vartheta_1 + \ldots + a_n\vartheta_n$, pak ne všechny a_i jsou dělitelné p. Předpokládejme, že a_1 není dělitelné p. Nyní vyjádříme diskriminant báze $\{\vartheta, \vartheta_2 \ldots, \vartheta_n\}$ pomocí původní báze. Využijeme-li přechodové matice a podobný postup s vnořeními L do $\mathbb{C}$ jako v předchozích důkazech, dostaneme

$$\operatorname{disc}(\vartheta, \vartheta_2, \ldots, \vartheta_n) = a_1^2 \cdot \operatorname{disc}(\vartheta_1, \ldots, \vartheta_n).$$

Protože p nedělí a_1, stačí ukázat, že p dělí $\operatorname{disc}(\vartheta, \vartheta_2, \ldots, \vartheta_n)$. Nechť M je rozšíření L, které je normální nad $\mathbb{Q}$ s oborem celých algebraických čísel T a nechť $\sigma_1, \ldots, \sigma_n$ jsou různá vnoření L do $\mathbb{C}$, která rozšíříme na M. Protože ϑ leží v každém prvoideálu v S nad p, musí také ležet v každém prvoideálu v T nad p. Nechť U je jeden z těchto prvoideálů v T. Chceme ukázat, že dokonce $\sigma_i(\vartheta) \in U$, pro každé i. To je snadné, neboť víme, že automorfimus σ_i^{-1} pouze permutuje jednotlivé prvoideály nad p v T, tedy $\vartheta \in \sigma_i^{-1}(U)$ (ϑ leží ve všech prvoideálech nad p v T) a dostáváme $\sigma_i(\vartheta) \in U$. Protože diskriminant $\operatorname{disc}(\vartheta, \vartheta_2, \ldots, \vartheta_n)$ je druhá mocnina determinantu matice se všemi prvky v U, musí být i tento diskriminant v U, ovšem také se jedná o celistvou bázi, takže leží i v $\mathbb{Z}$. Tedy $\operatorname{disc}(\vartheta, \vartheta_2, \ldots, \vartheta_n) \in U \cap \mathbb{Z} = p\mathbb{Z}$.

□

Podle věty 2.45 existuje jen konečně mnoho ramifikovaných prvoideálů $P = p\mathbb{Z}$. Jinými slovy, až na konečně mnoho případů je PS součinem prvoideálů v první mocnině. Věta 2.44 pro změnu říká kolik těchto prvoideálů může být.

Nyní uvedeme větu, která nám umožňuje určit rozklad ideálu PS pro skoro všechna prvočísla.

Věta 2.46. *Nechť $K \subseteq L$ jsou číselná tělesa s obory celých algebraických čísel R a S, $[L : K] = n$, $L = K[\vartheta]$ pro nějaké $\vartheta \in S$ s ireducibilním monickým*

polynomem f nad K. Nechť P je prvoideál v R ležící nad prvočíslem p, které nedělí $|S/R[\vartheta]|$. Dále nechť

$$f \equiv f_1^{e_1} \cdot \ldots \cdot f_k^{e_k} \bmod P[x].$$

Potom platí

$$PS = Q_1^{e_1} \cdot \ldots \cdot Q_k^{e_k},$$

kde $Q_i = PS + f_i(\vartheta)S = (P, f_i(\vartheta))$. Navíc $\mathrm{f}(Q_i|P) = \deg f_i$.

Důkaz. Důkaz rozdělíme do čtyř kroků. Víme, že můžeme koeficienty polynomu f vyjádřit pomocí jeho kořenů, které jsou také nutně celá algebraická čísla. Proto $f \in R[x]$. S polynomy tedy pracujeme nad okruhem R i nad konečným tělesem R/P charakteristiky p. Pro rozlišení budeme pro odpovídající polynomy v $(R/P)[x]$ používat pruh. Nechť $d_i = \deg f_i$.

(i) Pro každé i platí, že buď $Q_i = S$, nebo že S/Q_i je těleso řádu $|R/P|^{d_i}$.

Vidíme, že vhodně velké těleso, které by mohlo být použito k důkazu, je $F_i = (R/P)[x]/(\overline{f_i})$. K nalezení izomorfismu mezi F_i a S/Q_i použijeme následující homomorfismy. Nejprve uvažujme homomorfismus z $R[x]$ do F_i, který definujeme obvyklým způsobem jako redukování koeficientů modulu P a poté modulo $(\overline{f_i})$. Zřejmě se jedná o epimorfismus, jehož jádro je ideál $P[x] + (f_i) = (P, f_i)$. Dostáváme izomorfimus

$$R[x]/(P, f_i) \cong F_i,$$

navíc musí být (P, f_i) maximální ideál.

Zobrazení $R[x]$ do S definované pomocí $x \mapsto \vartheta$ indukuje homomorfismus $R[x]$ do S/Q_i. Protože $PS \subseteq Q_i$ a zároveň $f_i(\vartheta) \in Q_i$, musí maximální ideál (P, f_i) v $R[x]$ být v jádru homomorfismu. Tedy jádro je buď celé $R[x]$, nebo (P, f_i). Pokud bude tento homomorfismus na, jsme hotovi. Abychom dokázali, že je na, potřebujeme ukázat, že $S = R[\vartheta] + Q_i$. Z $p \in P \subset Q_i$ plyne $pS \subset Q_i$ a bude tedy stačit dokázat, že $S = R[\vartheta]+pS$. Index $R[\vartheta] + pS$ v S je ovšem společný dělitel $|S/R[\vartheta]|$ a $|S/pS|$, ale ty jsou nesoudělné, protože podle předpokladu p nedělí $|S/R[\vartheta]|$, zatímco $|S/pS|$ je mocninou p.

(ii) $Q_i + Q_j = S$, pro $i \neq j$

Protože $\overline{f_i}$ jsou ireducibilní polynomy v oboru hlavních ideálů $(R/P)[x]$, pro dané $i \neq j$ existují polynomy $\overline{g}$, $\overline{h}$ tak, že $\overline{f_i}\overline{g} + \overline{f_j h} = 1$ v $(R/P)[x]$. A tedy

$$f_i g + f_j h \equiv 1 \mod P[x].$$

Využijeme-li předchozího zobrazení $R[x]$ do S záměnou x za ϑ dostaneme kongruenci

$$f_i(\vartheta)g(\vartheta) + f_j(\vartheta)h(\vartheta) \equiv 1 \mod PS,$$

pak ovšem $1 \in (P, f_i(\vartheta), f_j(\vartheta)) = Q_i + Q_j$.

(iii) $PS|Q_1^{e_1} \cdot \ldots \cdot Q_k^{e_k}$

Označme $f_i(\vartheta) = \gamma_i$. Zřejmě součin ideálů $Q_1^{e_1} \cdot \ldots \cdot Q_k^{e_k}$ je obsažen a tedy i dělitelný ideálem $(P, \prod_{i=1}^k \gamma_i)$. Stačí ukázat, že $\prod_{i=1}^k \gamma_i$ leží v PS. Podle předpokladů máme

$$f \equiv f_1^{e_1} \cdot \ldots \cdot f_k^{e_k} \mod P[x].$$

Pokud opět použijeme zobrazení $R[x]$ do S dostaneme

$$f(\vartheta) \equiv \gamma_1^{e_1} \cdot \ldots \cdot \gamma_k^{e_k} \mod PS,$$

přitom $f(\vartheta) = 0$ a tedy $\prod_{i=1}^k \gamma_i \in PS$.

(iv) $PS = Q_1^{e_1} \cdot \ldots \cdot Q_k^{e_k}$

Nyní již můžeme vše zkombinovat a tvrzení dokázat. Rozdělme ideály Q_i tak, aby $Q_1, \ldots, Q_m \neq S$ a $Q_{m+1}, \ldots, Q_k = S$. Podle (i) jsou $Q_1, \ldots, Q_m$ prvoideály (jsou maximální) a navíc zřejmě leží nad P, navíc $\mathrm{f}(Q_i|P) = d_i = \deg f_i$ pro $i \leq m$. Z (ii) plyne, že jsou všechny prvoideály $Q_1, \ldots, Q_m$ různé. A nakonec víme, že $PS|Q_1^{e_1} \cdot \ldots \cdot Q_m^{e_m}$, ostatní jsou celé S. Potom ale musí být $PS = Q_1^{s_1} \cdot \ldots \cdot Q_m^{s_m}$, kde $0 \leq s_i \leq e_i$. Podle tvrzení 2.44 platí $n = d_1 s_1 + \ldots + d_m s_m$, přitom ale zřejmě $n = d_1 e_1 + \ldots + d_k e_k$ (z rozkladu polynomu f). Tedy $m = k$, $s_i = e_i$ pro všechna i.

□

Prvočísla, která dělí $|S/R[\vartheta]|$, označujeme jako *speciální*. Rozložit ideál PS nad speciálním prvočíslem je mnohem komplikovanější. Než uvedeme kompletní algoritmus, pomocí něhož můžeme rozložit speciální ideály, potřebujeme ještě několik pomocných definic a vět. Našim cílem bude vytvořit nadokruh $R[\vartheta]$, ve kterém leží ideál, který je součinem všech prvoideálů nad daným speciálním prvočíslem. S jeho pomocí původní ideál PS rozložíme.

Definice 2.47. Nechť K je číselné těleso s oborem celých algebraických čísel R, podokruh $O \subseteq R$ nazveme *plnoobor*, pokud je i $\mathbb{Z}$-modulem hodnosti $n = [K : \mathbb{Q}]$. Dále nechť p je prvočíslo. Řekneme, že O *je p-maximální*, pokud p nedělí $[R : O]$.

Obor celých algebraických čísel je zřejmě i plnooborem (vzhledem k inkluzi největší) a také p-maximální pro všechna prvočísla p, proto se nazývá *maximální plnoobor*.

Definice 2.48. Nechť O je plnoobor číselného tělesa K. Pro prvočíslo p definujme *p-radikál* I_p předpisem:

$$I_p = \{\vartheta \in O;\, \exists m \geq 1 : \vartheta^m \in pO\}.$$

Následující lemma uvádí základní vlastnosti p-radikálu.

Lemma 2.49. *Nechť O je plnoobor číselného tělesa K a p libovolné prvočíslo.*

(i) p-radikál I_p je ideál oboru O.

(ii) Nechť $P_1, \ldots, P_k$ jsou všechny prvoideály v O nad prvočíslem p, pak pro I_p platí

$$I_p = P_1 \cdot \ldots \cdot P_k.$$

(iii) Existuje $m \in \mathbb{N}$ takové, že $I_p^m \subseteq pO$.

Důkaz. Lemma postupně dokážeme.

(i) Stačí ukázat, že pro ξ , $\zeta \in I_p$ a $\vartheta \in O$ platí $\xi\vartheta \in I_p$ a $\xi + \zeta \in I_p$. Nechť m, n jsou takové, že $\xi^m \in pO$ a $\zeta^n \in pO$. Pak zřejmě $(\xi + \zeta)^{m+n} \in pO$ a $(\vartheta\xi)^{\max(m,n)} \in pO$.

(ii) Nejprve dokážeme, že $I_p \subseteq P_1 \cdot \ldots \cdot P_k$. Protože P_i leží nad p, máme $pO \subseteq P_i$ pro každé i. Zvolme libovolné $\xi \in I_p$. Podle definice p-radikálu existuje m, že $\xi^m \in pO \subseteq P_i$ a z vlastnosti prvoideálů plyne, že $\xi \in P_i$ pro každé i. Tedy $\xi \in \bigcap_{i=1}^k P_i = \prod_{i=1}^k P_i$, protože P_i jsou po dvou komaximální. Komaximalita plyne z toho, že O/P_i je konečný obor integrity, neboli těleso, a tedy P_i jsou maximální ideály.

Naopak uvažujme $\xi \in \prod_{i=1}^k P_i$. Mezi množinou ideálů v O obsahujících pO a množinou ideálů ve faktorokruhu O/pO existuje přirozená korespondence. Protože O/pO je konečný okruh a tedy obsahuje pouze konečně ideálů, musí být i konečný počet ideálů v O obsahujících pO. Speciálně v O/pO je pouze konečný počet ideálů tvaru $[\xi]^e O/pO$ ($[\xi]$ označuje třídu v O/pO obsahující ξ). Musí tedy existovat e tak, že $[\xi]^e O/pO = [\xi]^{e+1} O/pO$. Dostáváme $[\xi]^e(1 - [\xi][\zeta]) = 0$ pro nějaké $[\zeta] \in O/pO$. Podle předpokladu $[\xi]$ leží ve všech maximálních ideálech $P_i/pO \subset O/pO$. Pak ovšem $1 - [\xi][\zeta]$ nemůže patřit do žádného z nich, jinak by 1 patřila také (to není možné, protože uvažujeme netriviální prvoideály). To znamená, že $(1 - [\xi][\zeta])O/pO = O/pO$, tedy $1 - [\xi][\zeta]$ je invertibilní, a proto $[\xi]^e = 0 \in O/pO$. Dostáváme, že $\xi^e \in pO$, a podle definice p-radikálu $\xi \in I_p$.

(iii) Protože I_p je ideál v plnooboru, který je $\mathbb{Z}$-modulem konečné hodnosti, musí mít konečnou bázi $\{\xi_1, \ldots, \xi_n\}$. Existují tedy m_i, $1 \leq i \leq n$, takové, že $\xi_i^{m_i} \in pO$. Stačí volit $m = \sum_{i=1}^n m_i$.

□

Ještě dodejme, že p-radikál I_p není obecně součinem všech prvoideálů nad p, které leží v R, ale pouze těch, které leží v O. Ale je zřejmé, že pokud je O p-maximální, potom obsahuje již všechny prvoideály nad p z R (jednoduchými úpravami dostáváme $O/pO \cong O/(O \cap pR) \cong (O + pR)/pR \cong R/pR$). Nechť O je plnoobor, který není p-maximální pro nějaké prvočíslo p. Popíšeme způsob (známý jako Pohst-Zassenhausův), kterým zvětšíme tento plnoobor na p-maximální.

Věta 2.50. *Nechť O je plnoobor číselného tělesa K, $n = [K : \mathbb{Q}]$ a nechť p je prvočíslo. Označme*

$$O' = \{\alpha \in K;\, \alpha I_p \subseteq I_p\}.$$

Potom buď $O' = O$ a O je p-maximální, nebo $O \subset O'$ a $p \,|\, [O' : O] \,|\, p^n$.

Důkaz. Zřejmě O' je okruh, který obsahuje O. Protože $p \in I_p$, dostáváme pro $\vartheta \in O'$ vztah $\vartheta p \in I_p \subset O$ a tedy $O \subset O' \subset \frac{1}{p}O$. To znamená, že O' má hodnost n a je plnooborem. Navíc $[O' : O]$ dělí $[\frac{1}{p}O : O] = [O : pO] = p^n$.

Předpokládejme, že $O' = O$. Definujme

$$O_p = \{\vartheta \in R;\, \exists j \geq 1, p^j\vartheta \in O\}.$$

Zřejmě $O \subset O_p$ a O_p je plnoobor, který je již p-maximální. Kdyby $p|[R : O_p]$, pak by existovalo $\vartheta \in R$ tak, že $\vartheta \notin O_p$ a $p\vartheta \in O_p$. To ale není podle definice možné.

Ukážeme, že O se v tomto případě rovná O_p. Protože O_p je plnoobor, má nějakou bázi $\{\vartheta_1, \ldots, \vartheta_n\}$. Pro každé ϑ_i existuje $r_i \in \mathbb{Z}$ tak, že $p^{r_i}\vartheta_i \in O$. Volíme-li $r = \max(r_i)$, dostáváme $p^r O_p \subset O$. Podle lemmatu 2.49 existuje $m \in \mathbb{N}$, že $I_p^m \subset pO$. Tedy $O_p I_p^{mr} \subset O$. Dále budeme pokračovat sporem. Nechť $O_p \setminus O \neq \emptyset$. Označme $k < mr$ největší mocninu, že $O_p I_p^k \setminus O \neq \emptyset$. Nechť $\mu \in O_p I_p^k \setminus O$. Protože $O_p I_p^{k+1} \subset O$, máme $\mu I_p \subset O$. Dále platí $O_p I_p^{k+m+1} \subset I_p^m \subset pO$. Zvolíme-li $\xi \in I_p$, potom $(\mu\xi)^{k+m+1} \in pO$ a podle definice p-radikálu I_p leží $\mu\xi$ v radikálu I_p, tedy $\mu I_p \subseteq I_p$. Tím jsme ukázali, že μ leží v O', ale podle volby neleží v O. A to je spor, protože předpokládáme, že $O = O'$. □

Ukazuje se, že je výpočetně velmi náročné a mnohdy zbytečné zkoušet pro každé speciální prvočíslo zvětšovat zadaný plnoobor pomocí Pohst-Zassenhausova algoritmu, protože je často již p-maximální. Takzvané Dedekindovo kritérium, které je popsané níže, nám umožní efektivněji dosáhnout požadovaného výsledku a pokud je již obor p-maximální vyhnout se náročným výpočtům. Nejprve uvedeme dvě pomocná lemmata.

Pro následující část nechť $K = \mathbb{Q}[\vartheta]$ je číselné těleso, $m_\vartheta \in \mathbb{Z}[x]$ je monický minimální polynom ϑ nad $\mathbb{Z}$ a p je prvočíslo. Dále budeme $\bar{}$ značit redukci modulo p. Pokud definujeme nejprve nějaké $\overline{f}(x) \in \mathbb{Z}_p[x]$, tak $f(x) \in \mathbb{Z}[x]$ bude mít všechny koeficienty v $\{0, 1, \ldots, p-1\}$. Nechť

$$\overline{m}_\vartheta(x) = \prod_{i=1}^{k} \overline{m}_i^{e_i}(x)$$

je faktorizace $\overline{m}_\vartheta$ v $\mathbb{Z}_p[x]$. Označme

$$g(x) = \prod_{i=1}^{k} m_i(x),$$

kde $m_i(x) \in \mathbb{Z}[x]$.

Lemma 2.51. *Nechť K, m_ϑ, m_i a $g(x) \in \mathbb{Z}[x]$ jsou definována jako výše. Pak pro p-radikál I_p v $\mathbb{Z}[\vartheta]$ platí*

$$I_p = p\mathbb{Z}[\vartheta] + g(\vartheta)\mathbb{Z}[\vartheta].$$

Tedy $\xi = u(\vartheta) \in I_p$, kde $u(x) \in \mathbb{Z}[x]$, právě když $\overline{g}|\overline{u}$.

Důkaz. Z definice I_p víme, že $p \in I_p$. Zřejmě $e_i \leq n = [K : \mathbb{Q}] = \deg m_\vartheta$, tedy $\overline{m}_\vartheta | \overline{g}^n$ modulo p, a dostáváme, že $g^n(\vartheta) \equiv 0 \mod p\mathbb{Z}[\vartheta]$. Pak ovšem $g(\vartheta)$ leží v I_p a tedy $I_p \supseteq p\mathbb{Z}[\vartheta] + g(\vartheta)\mathbb{Z}[\vartheta]$.

Naopak nechť $\xi \in I_p$. Pak podle definice p-radikálu existuje $m \in \mathbb{N}$ tak, že $\xi^m \in p\mathbb{Z}[\vartheta]$. Protože I_p je ideál v $\mathbb{Z}[\vartheta]$, existuje $u(x) \in \mathbb{Z}[x]$, že $u(\vartheta) = \xi$. Pak i $u^m(\vartheta) \in p\mathbb{Z}[\vartheta]$. Tedy existuje $v(x) \in \mathbb{Z}[x]$, že $u^m - pv$ je dělitelné minimálním polynomem m_ϑ. Pak ovšem $\overline{m_\vartheta}$ dělí $\overline{u}^n$. Dostáváme, že $\overline{m}_i | \overline{u}^m$, a protože $\overline{m}_i$ jsou ireducibilní v $\mathbb{Z}_p[x]$, máme $\overline{m}_i | \overline{u}$. Dále víme, že $\overline{m}_i$ jsou navzájem nesoudělné, tedy $\overline{g}|\overline{u}$. To znamená, že ξ leží v $p\mathbb{Z}[\vartheta] + g(\vartheta)\mathbb{Z}[\vartheta]$. □

Lemma 2.52. *Nechť K, m_ϑ, m_i a $g(x) \in \mathbb{Z}[x]$ jsou definována jako výše. Nechť $\overline{h} = \overline{m}_\vartheta(x)/\overline{g}(x)$. Položme*

$$f(x) = (g(x)h(x) - m_\vartheta(x))/p \in \mathbb{Z}[x].$$

Dále nechť $\xi = u(\vartheta)/p$, kde $u(x) \in \mathbb{Z}[x]$. Pak platí

(i) $p\xi \in I_p$, právě když $\overline{g}|\overline{u}$.

(ii) Nechť $\overline{w} = \overline{g}/\gcd(\overline{f}, \overline{g})$. Potom $\xi g(\vartheta) \in I_p$, právě když $\overline{w}\overline{h}|\overline{u}$.

Důkaz. Lemma postupně dokážeme.

(i) Toto je zřejmě důsledkem lemmatu 2.51.

(ii) Podle lemmatu 2.51 je $I_p = p\mathbb{Z}[\vartheta] + g(\vartheta)\mathbb{Z}[\vartheta]$. Tedy $\xi g(\vartheta)$ leží v I_p, právě když existují polynomy $u_1(x)$, $u_2(x) \in \mathbb{Z}[x]$ tak, že

$$\xi g(\vartheta) = u(\vartheta)g(\vartheta)/p = pu_1(\vartheta) + g(\vartheta)u_2(\vartheta).$$

Protože m_ϑ je minimální polynom ϑ, dostáváme, že $\xi g(\vartheta) \in I_p$, právě když existuje polynom $u_3(x) \in \mathbb{Z}[x]$ tak, že

$$u(x)g(x) = p^2u_1(x) + pg(x)u_2(x) + m_\vartheta(x)u_3(x) \text{ v } \mathbb{Z}[x].$$

Když provedeme redukci modulo p, dostaneme $\overline{u} = \overline{u}_3\overline{h}$. Tedy

$$u(x) = h(x)u_3(x) + pu_4(x), \;\; u_4(x) \in \mathbb{Z}[x].$$

Máme $\xi g(\vartheta) \in I_p$, právě když existují polynomy $u_i(x) \in \mathbb{Z}[x]$ tak, že

$$(h(x)g(x) - m_\vartheta(x))u_3(x) = p^2u_1(x) + pg(x)(u_2(x) - u_4(x))$$

a současně platí $u(x) = h(x)u_3(x) + pu_4(x)$. Tedy právě když $u(x) = h(x)u_3(x) + pu_4(x)$ a

$$f(x)u_3(x) = pu_1(x) + g(x)u_5(x).$$

Opět použijeme redukci modulo p a vidíme, že poslední vztah je ekvivaletní $\overline{g}|\overline{f}\overline{u}_3$. Tedy $\overline{w}|\overline{u}_3$, kde $\overline{w} = \overline{g}/\gcd(\overline{f}, \overline{g})$. Když přejdeme zpět do $\mathbb{Z}[x]$, tak $\overline{w}|\overline{u}_3$ je ekvivalentní existenci polynomů $u_6(x), u_7(x) \in \mathbb{Z}[x]$ tak, že

$$u_3(x) = w(x)u_6(x) + pu_7(x).$$

Nyní můžeme dát vše dohromady. Pro $\xi = u(\vartheta)/p$ platí, že $\xi g(\vartheta) \in I_p$, právě když existují polynomy $u_4(x), u_6(x), u_7(x) \in \mathbb{Z}[x]$ tak, že

$$u(x) = h(x)w(x)u_6(x) + p(h(x)u_7(x) + u_4(x)).$$

A to je zřejmě ekvivalentní tomu, že $\overline{h}\overline{w}|\overline{u}$.

□

Nyní již můžeme dokázat samotné Dedekindovo kritérium.

Věta 2.53. *Nechť $K = \mathbb{Q}[\vartheta]$ je číselné těleso, $m_\vartheta \in \mathbb{Z}[x]$ je monický minimální polynom ϑ nad $\mathbb{Z}$ a nechť p je prvočíslo. Dále budeme $^-$ značit redukci modulo p. Nechť*

$$\overline{m}_\vartheta(x) = \prod_{i=1}^{k} \overline{m}_i^{e_i}(x)$$

je faktorizace $\overline{m}_\vartheta$ v $\mathbb{Z}_p[x]$. Označme

$$g(x) = \prod_{i=1}^{k} m_i(x),$$

kde $m_i(x) \in \mathbb{Z}[x]$. *Nechť* $\overline{h} = \overline{m}_\vartheta(x)/\overline{g}(x)$. *Položme*

$$f(x) = (g(x)h(x) - m_\vartheta(x))/p \in \mathbb{Z}[x].$$

(i) Plnoobor $\mathbb{Z}[\vartheta]$ *je* p*-maximální, právě když*

$$\gcd(\overline{f}, \overline{g}, \overline{h}) = 1 \ \ v \ \ \mathbb{Z}_p[x].$$

(ii) Obecněji, nechť O' *je plnoobor, který dostaneme použitím tvrzení 2.50, když začneme s* $O = \mathbb{Z}[\vartheta]$. *Nechť* $\overline{v} = \overline{m}_\vartheta / \gcd(\overline{f}, \overline{g}, \overline{h})$. *Pak*

$$O' = \mathbb{Z}[\vartheta] + \frac{v(\vartheta)}{p}\mathbb{Z}[\vartheta].$$

Pokud $r = \deg(\gcd(\overline{f}, \overline{g}, \overline{h}))$, *pak* $[O' : \mathbb{Z}[\vartheta]] = p^r$. *Pro diskriminant plnooboru* O' *platí, že* $\operatorname{disc}(O') = \operatorname{disc}(m_\vartheta)/p^{2r}$.

Důkaz. Tvrzení postupně dokážeme.

(i) Plyne z druhé části, $r = \deg(\gcd(\overline{f}, \overline{g}, \overline{h})) = 0$.

(ii) Připomeňme, že $O' = \{\alpha \in K;\ \alpha I_p \subseteq I_p\}$. Podle 2.51 je $\xi \in O'$, právě když $p\xi \in I_p$ a $g(\vartheta)\xi \in I_p$. Protože $I_p \subseteq \mathbb{Z}[\vartheta]$, tak z $p\xi \in I_p$ plyne

$$\xi = u(\vartheta)/p, \ \ u(x) \in \mathbb{Z}[x].$$

Podle 2.52 máme $\xi \in O'$, právě když jak $\overline{g}$ tak $\overline{w}\overline{h}$ dělí $\overline{u}$ v $\mathbb{Z}_p[x]$ ($\overline{w} = \overline{g}/\gcd(\overline{f}, \overline{g})$). Víme, že $\mathbb{Z}_p[x]$ je obor hlavních ideálů, tedy podmínka je ekvivalentní tomu, že nejmenší společný násobek $\overline{g}$ a $\overline{w}\overline{h}$ dělí $\overline{u}$. Navíc v oboru hlavních ideálů zřejmě platí $\operatorname{lcm}(a,b) = ab/\gcd(a,b)$ a také $\operatorname{lcm}(ca, cb) = c\operatorname{lcm}(a,b)$. Proto dostáváme

$$\begin{aligned} \operatorname{lcm}(\overline{g}, \overline{w}\overline{h}) &= \overline{w}\operatorname{lcm}(\gcd(\overline{f}, \overline{g}), \overline{h}) = \frac{\overline{g}}{\gcd(\overline{f}, \overline{g})}\frac{\overline{h}\gcd(\overline{f}, \overline{g})}{\gcd(\overline{f}, \overline{g}, \overline{h})} = \\ &= \frac{\overline{m}_\vartheta}{\gcd(\overline{f}, \overline{g}, \overline{h})} = \overline{v}. \end{aligned}$$

Tedy libovolné $\xi = u(\vartheta)/p \in O'$, právě když $\overline{u}(x)$ je dělitelné $\overline{v}(x)$, neboli $u(x) = u_1(x)v(x) + pu_2(x)$ pro nějaké $u_1(x),\ u_2(x) \in \mathbb{Z}[x]$. Proto $O' = \mathbb{Z}[\vartheta] + (v(\vartheta)/p)\mathbb{Z}[\vartheta]$. Také je zřejmé, že reprezentanti tříd v $O'/\mathbb{Z}[\vartheta]$

jsou $u(\vartheta)v(\vartheta)/p$, kde $\overline{u}$ volíme přes všechny polynomy v $\mathbb{Z}_p[x]$, pro které $\deg(\overline{u}) < \deg(\overline{m}_\vartheta) - \deg(\overline{v}) = \deg(\gcd(\overline{f}, \overline{g}, \overline{h})) = r$. Dostáváme tedy $[O' : \mathbb{Z}[\vartheta]] = p^r$.

Z 2.21 víme, že $\mathrm{disc}(\mathbb{Z}[\vartheta]) = \mathrm{disc}(m_\vartheta)$. Podle tvrzení 2.19, které je dokázáno i obecněji, máme $\mathrm{disc}(\mathbb{Z}[\vartheta]) = \mathrm{disc}(O')[O' : \mathbb{Z}[\vartheta]]^2$, z toho již tvrzení snadno plyne.

□

Kdybychom Dedekindovo kritérium mohli použít pouze na $\mathbb{Z}[\vartheta]$, využili bychom ho pouze jednou. Ale protože důkaz je závislý na prvočísle p, je možné nahradit $\mathbb{Z}[\vartheta]$ za libovolný plnoobor O takový, že $[O : \mathbb{Z}[\vartheta]]$ není dělitelný p. V tomto se skrývá jeho užitečnost.

Nyní popíšeme algoritmus na hledání rozkladu ideálu pO v p-maximálním plnooboru O, kde p je speciální prvočíslo. Nechť I_p je p-radikál O. Z lemmatu 2.49 víme, že $I_p = P_1 \cdot \ldots \cdot P_k$, kde P_i jsou všechny prvoideály v O nad p. Nechť dále $pO = P_1^{e_1} \cdot \ldots \cdot P_k^{e_k}$. Našim cílem je nálézt generátory prvoideálů P_i a čísla e_i. Ideály I_p a pO známe (výpočet I_p viz [5], I_p/pO je nilradikál okruhu O/pO). Nejprve budeme chtít vyjádřit pomocné ideály

$$H_j = \prod_{i;\, e_i = j} P_i\,, \;\; j \geq 0.$$

Je zřejmé, že ideály H_j jsou navzájem nesoudělné (ve smyslu násobení ideálů) a jsou násobkem různých prvoideálů. Označme $e = \max(e_i;\, i = 1, \ldots, k)$, nyní můžeme vyjádřit pO pomocí H_j

$$pO = \prod_{j=1}^{e} H_j^j.$$

K tomu, abychom určili rozklad pO na prvoideály, zjevně stačí najít rozklad na prvoideály jednotlivých ideálů H_j. Dále víme, že prvoideály dělící H_j mají ramifikační index roven j.

Výpočet ideálů H_j vypadá následovně. Označme

$$K_j = I_p^j + pO\,, \;\; j \geq 0.$$

Díky $O/pO \subseteq O_K/pO_K$ víme, že K_j můžeme vyjádřit pomocí prvoideálů P_i.

$$K_j = I_p^j + pO = \prod_{i=1}^{k} P_i^j + \prod_{i=1}^{k} P_i^{e_i} = \prod_{i=1}^{k} P_i^{\min(j,e_i)}.$$

Z toho již snadno plyne $K_j \subseteq K_{j-1}$ a umožňuje nám definovat pomocné ideály J_j předpisem

$$J_j = K_j \cdot (K_{j-1})^{-1} = \prod_{i;\, e_i \geq j} P_i\,, \;\; j \geq 1,$$

navíc $J_j \subseteq J_{j+1}$. Nakonec dostáváme ideály H_j

$$H_j = J_j \cdot (J_{j+1})^{-1} = \prod_{i;\, e_i = j} P_i\,, \;\; j \geq 0.$$

K rozkladu H_j budeme potřebovat ještě jednu pomocnou definici.

Definice 2.54. Nechť R je komutativní okruh. Pak řekneme, že prvek $e \in R$ je *idempotent*, pokud platí $e \cdot e = e$. Nechť e, e' jsou dva různé idempotenty v R, řekneme, že jsou navzájem *ortogonální*, pokud $e \cdot e' = 0$.

Protože O je plnoobor, neboli volná abelovská grupa hodnosti n, můžeme psát $O/pO = \{\sum_{i=1}^{n} a_i \vartheta_i;\, a_i \in \mathbb{Z}_p\}$, pro nějakou volnou bázi $\{\vartheta_1, \ldots, \vartheta_n\}$ plnooboru O. Každý ideál $I \cdot pO$ v O/pO tedy odpovídá nějakému vektorovému podprostoru a můžeme ho popsat vhodnou bází, tím také dostáváme reprezentaci odpovídajícího ideálu $I \supseteq pO$ v O. Budeme proto pracovat s faktor okruhem O/pO jako s $\mathbb{Z}_p$-algebrou, kterou označíme A. Pro ideál I v O označme $\overline{I}$ odpovídající ideál v A. Pro jednotlivé ideály H_j dostáváme pomocí Čínské věty o zbytku, že $A/\overline{H}_j \cong A/\overline{P}_{i_1} \times \ldots \times A/\overline{P}_{i_k} \cong F_1 \times \ldots \times F_k$, kde F_i jsou algebraická rozšíření tělesa $\mathbb{F}_p$. Věta 2.55 nám umožní rozhodnout, zdali $A/\overline{H}_j$ je již těleso, nebo nám pomůže najít prvek e z A, pomocí něhož rozložíme $A/\overline{H}_j$ na součin ideálů následovně. Nechť $e \in A$ takový, že $e + \overline{H}_j$ je netriviální idempotent v $A/\overline{H}_j$. Pak $A/\overline{H}_j \cong A/(\overline{H}_j + eA) \times A/(\overline{H}_j + (1-e)A)$. Podařilo se nám tedy rozložit H_j na součin dvou různých větších ideálů a postupně můžeme určovat jednotlivé prvoideály, ze kterých se H_j skládá. Pokud nalezneme více vzájemně ortogonálních idempotentů, můžeme ihned rozložit H_j na více ideálů.

Věta 2.55. *Nechť $B = F_1 \times \ldots \times F_k$, kde F_i jsou algebraická rozšíření komutativního tělesa F. Pro $\alpha = (\alpha_1, \ldots, \alpha_k)$ uvažujme dosazovací homomorfismus $j_\alpha : F[x] \to B$, $j_\alpha(a) = (a, \ldots, a)$ pro $a \in F$, $j_\alpha(x) = (\alpha_1, \ldots, \alpha_k)$. Pak existuje jediný monický polynom $m_\alpha \in F[x]$ takový, že* $\operatorname{Ker} j_\alpha = m_\alpha F[x]$. *Přitom m_α je rovno nejmenšímu společnému násobku minimálních polynomů $m_{\alpha_1}, \ldots, m_{\alpha_k}$.*

Důkaz. Zřejmě i-tá složka zobrazení j_α má v jádru minimální polynom α_i. Ten je ovšem ireducibilní, takže ideál generovaný tímto polynomem je maximální, a proto je celým jádrem i-té složky. Pak jistě jádro zobrazení j_α obsahuje nejmenší společný násobek minimálních polynomů $m_{\alpha_1}, \ldots, m_{\alpha_k}$. Naopak pokud polynom f leží v jádru zobrazení j_α musí ležet i v jádrech jednotlivých složek a tedy musí být dělitelný odpovídajícím minimálním polynomem. Protože jsme v oboru hlavních ideálů je jádro j_α generováno tímto nejmenším společným násobkem. □

Nyní můžeme popsat hledání idempotentu. Nechť $B = F_1 \times \ldots \times F_k$, kde F_i jsou konečná algebraická rozšíření tělesa $\mathbb{F}_p$. Vystačíme s nalezením prvku $\alpha \in \mathbb{F}_p \times \ldots \times \mathbb{F}_p$, $k-$krát. Takový prvek jistě leží v jádru lineárního zobrazení $x \mapsto x^p - x$, proto není těžké alespoň jeden najít. Dále najdeme jeho minimální

polynom (ten zkonstruujeme hledáním závislostí mezi $1, \alpha, \alpha^2, \ldots$). Protože $\alpha = (\alpha_1, \ldots, \alpha_k)$, kde $\alpha_i \in \mathbb{F}_p$, jsou minimální polynomy $m_{\alpha_1}, \ldots, m_{\alpha_k}$ lineární a polynom m_α se v $\mathbb{F}_p[x]$ rozkládá na kořenové činitele, které lze snadno nalézt. Když známe rozklad $m_\alpha = \prod_{i=1}^{l} m_i$, $l \leq k$, můžeme uvažovat polynomy $\overline{m}_1 = m_\alpha/m_1, \ldots, \overline{m}_l = m_\alpha/m_l$, které jsou nesoudělné (m_α je nejmenší společný násobek všech minimálních polynomů). Dostáváme

$$1 = f_1\overline{m}_1 + \ldots + f_l\overline{m}_l$$

pro vhodná $f_i \in \mathbb{F}_p[x]$. Tedy $f_i(\alpha)\overline{m}_i(\alpha)$, $i = 1, \ldots l$, jsou idempotenty (navíc jsou zřejmě ortogonální), neboť například

$$\begin{aligned}(f_1(\alpha)\overline{m}_1(\alpha))^2 &= (f_1(\alpha)\overline{m}_1(\alpha)) \cdot (1 - f_2(\alpha)\overline{m}_2(\alpha) - \ldots - f_l(\alpha)\overline{m}_l(\alpha)) = \\ &= f_1(\alpha)\overline{m}_1(\alpha) - g_2(\alpha)m_\alpha(\alpha) - \ldots - g_l(\alpha)m_\alpha(\alpha) = \\ &= f_1(\alpha)\overline{m}_1(\alpha),\end{aligned}$$

protože $m_\alpha(\alpha) = 0$, $g_i \in \mathbb{F}_p[x]$, $i = 2, \ldots, l$. Pomocí netriviálního idempotentu podle výše napsaného můžeme rozložit $\overline{H}_j$ na dva větší ideály. Že se jedná již o prvoideál poznáme podle toho, že polynom m_α je prvního stupně. Stupeň setrvačnosti tohoto prvoideálu P určíme podle dimenze A/P nad $\mathbb{Z}_p$, neboť známe odpovídající vektorový podprostor. Tímto způsobem dokážeme najít požadovaný rozklad ideálu pO nad speciálním prvočíslem.

Nyní můžeme popsal celý postup. Uvažujme číselné těleso $K = \mathbb{Q}[\vartheta]$, $\vartheta \in O_K$ stupně n nad $\mathbb{Q}$. Kompletní algoritmus na rozložení ideálu pO_k, p prvočíslo, vypadá následovně. Začneme s plnooborem $O = \mathbb{Z}[\vartheta]$ a vypočítáme $\mathrm{disc}(1, \vartheta, \ldots, \vartheta^{n-1})$. Podle věty 2.19 každé prvočíslo, které v druhé mocnině dělí tento diskriminant, může být speciální, proto pomocí Pohst-Zassenhausova algoritmu s prvním krokem podle Dedekindova kritéria (nebo jeho vylepšenou modifikací) zvětšíme plnoobor O na p-maximální pro každé takové prvočíslo. Dostaneme tedy plnoobor O, který je již p-maximální pro všechna speciální prvočísla. Pokud použijeme všechna prvočísla dostáváme maximální plnoobor O_K, ale většinou nás zajímá jen rozklad ideálů nad prvočísly do určité pevně dané velikosti. Když máme k dispozici dostatečně velký plnoobor O, rozhodneme se podle typu prvočísla jaký algoritmus pro rozložení použijeme. Pro nespeciální prvočísla použijeme postup popsaný ve větě 2.46, pokud se jedná o speciální prvočíslo použijeme tzv. Buchmann-Lenstrův algoritmus popsaný výše. Další detaily, včetně podrobného popisu kroků algoritmu, lze nalézt v [5] (6. kapitola).

Nakonec uveďme několik vět, které využijeme v popisu číselného síta.

Věta 2.56. *Nechť $K = \mathbb{Q}[\vartheta]$, $\vartheta \in O_K$, je číselné těleso, P je prvoideál v O_K nad nespeciálním prvočíslem p a nechť a,$b \in \mathbb{Z}$ jsou nesoudělná. Pokud prvek $a + b\vartheta \in P$, pak P je stupně setrvačnosti 1.*

Důkaz. Protože $a + b\vartheta \in P$, nemůže být b dělitelné p. Jinak bychom dostali, že a leží v P a tedy a by bylo také dělitelné p. Ale a, b jsou nesoudělná. Když b není dělitelné p, pak existuje inverze b modulo p. Označme ji c. Máme

$$b\vartheta \equiv -a \mod P.$$

Dále vynásobíme kongruenci číslem c

$$\vartheta \equiv -ac \mod P.$$

Tedy $O_K = P + \mathbb{Z}$. Podle definice stupně setrvačnosti 2.40 máme

$$[O_K/P : \mathbb{Z}/p\mathbb{Z}] = [O_K/P : (P + \mathbb{Z})/P] = [O_K/P : O_K/P] = 1.$$

První rovnost plyne z 3. věty o izomorfismu.

□

Věta 2.57. *Nechť $K = \mathbb{Q}[\vartheta]$, $\vartheta \in O_K$, je číselné těleso a nechť a,$b \in \mathbb{Z}$ jsou nesoudělná. Pak pro každý prvoideál P v O_K nad nespeciálním prvočíslem p stupně setrvačnosti 1 existuje jednoznačně určené číslo $c_p \in \mathbb{Z}_p$ takové, že $a+b\vartheta \in P$, právě když $a+bc_p \equiv 0 \mod p$. Přitom c_p je kořenem minimálního polynomu ϑ nad $\mathbb{Z}$ modulo p.*

Důkaz. Prvoideál P je stupně setrvačnosti 1, tedy $\vartheta \equiv u \mod P$, kde $u \in \mathbb{Z}_p$ ($\vartheta - u \in P$). Dále víme (věta 2.46), že tento prvoideál je generovaný dvojicí $(p, \vartheta - v)$, pro nějaké $v \in \mathbb{Z}$. Polynom $x - v$ je dělitel minimálního polynomu ϑ nad $\mathbb{Z}$ modulo p. Zřejmě $u \equiv v \equiv c_p \mod p$. Pokud $a + b\vartheta \in P$, pak $-ab^{-1} \equiv \vartheta \equiv c_p \mod P$ a tedy $a + bc_p \equiv 0 \mod p$. Naopak podobně. □

Věta 2.56 říká, že hlavní ideál $(a + b\vartheta)$, $a, b \in \mathbb{Z}$ nesoudělná, je dělitelný pouze prvoideály, které jsou stupně setrvačnosti 1 (uvažujeme pouze prvoideály nad nespeciálními prvočísly). Věta 2.57 navíc tyto prvoideály přesně popisuje, protože podle věty 2.46 víme, že každý prvoideál nad nespeciálním prvočíslem p stupně setrvačnosti 1 můžeme vyjádřit ve tvaru $P = (p, \vartheta - c_p) = pO_K + (\vartheta - c_p)O_K$. Dále je zřejmé, že tento prvoideál je jen jeden (nad uvažovaným prvočíslem p dělícím hlavní ideál $(a + b\vartheta)$). Podle věty 2.42 je norma ideálu $(a + b\vartheta)$ dělitelná normou prvoideálu P, která je rovna p. A protože norma hlavního ideálu $(a + b\vartheta)$ je rovna absolutní hodnotě normy čísla $a + b\vartheta$ podle 2.43. Tedy maximální mocnina, ve které prvočíslo p dělí normu čísla $a + b\vartheta$, je zároveň i maximální mocnina, ve které prvoideál P dělí hlavní ideál $(a + b\vartheta)$.

Pokud uvažujeme prvoideály nad speciálními prvočísly, je situace podobná. Přesněji nechť $Q \subseteq O_K$ je prvoideál nad speciálním prvočíslem q. Máme-li $a + b\vartheta \in Q$, pak jistě $a + b\vartheta \in Q \cap \mathbb{Z}[\vartheta]$, a proto $a + b\vartheta$ leží v prvoideálu $\overline{Q} = Q \cap \mathbb{Z}[\vartheta] \subseteq \mathbb{Z}[\vartheta]$. Navíc nemůže q dělit b, jinak dostáváme, že a leží v Q a tedy q dělí a, ale a, b jsou nesoudělná. Protože q nedělí b, můžeme najít jeho inverz modulo q, označme ho u, $ub \equiv 1 \mod q$. Pak ovšem prvek

$au + \vartheta$ leží v prvoideálu $\overline{Q} \subseteq \mathbb{Z}[\vartheta]$ a tedy $\vartheta \equiv -au \mod \overline{Q}$. Z tohoto důvodu je libovolný prvek $\mu \in \mathbb{Z}[\vartheta]$ kongruentní modulo $\overline{Q}$ nějakému celému číslu. Máme $\mathbb{Z}[\vartheta] = \mathbb{Z} + \overline{Q} = \mathbb{Z} + Q \cap \mathbb{Z}[\vartheta]$, a protože $\mathbb{Z} \cap Q \cap \mathbb{Z}[\vartheta] = q\mathbb{Z} \cap \mathbb{Z}[\vartheta] = q\mathbb{Z}$, dostáváme podle 3. věty o izomorfismu, že $\mathbb{Z}[\vartheta]/(Q \cap \mathbb{Z}[\vartheta]) \cong \mathbb{Z}/q\mathbb{Z}$. Prvoideály v $\mathbb{Z}[\vartheta]$ nad q dostaneme s pomocí izomorfismu $\mathbb{Z}[\vartheta]/q\mathbb{Z}[\vartheta] \cong \mathbb{Z}_q[x]/\overline{m}_\vartheta \mathbb{Z}_q[x]$, kde m_ϑ je minimální polynom ϑ nad $\mathbb{Z}$ a $\overline{m}_\vartheta$ je jeho projekce modulo q (víme, že $\mathbb{Z}[\vartheta] \cong \mathbb{Z}[x]/m_\vartheta \mathbb{Z}[x]$). Opět rozložíme polynom $\overline{m}_\vartheta = \prod_{i=1}^{k} \overline{m}_i$ v $\mathbb{Z}_q[x]$ a všechny prvoideály nad q v $\mathbb{Z}[\vartheta]$ jsou rovny $(q, m_i(\vartheta))$, $i = 1, \dots k$. Tedy pro každý kořen c polynomu $\overline{m}_\vartheta$ v $\mathbb{Z}_q[x]$ existuje prvoideál $Q \subseteq O_K$ takový, že $a + b\vartheta$, $a, b \in \mathbb{Z}$ nesoudělná, leží v Q, právě když q dělí $a + bc$. Na rozdíl od prvoideálů nad nespeciálními prvočísly není prvoideál Q určen jednoznačně pomocí q a kořene c. Také prvoideál Q nemusí být nutně stupně setrvačnosti 1, potom je nutné znát tento stupeň setrvačnosti, aby bylo možné zjistit, v kolikáté mocnině dělí prvoideál Q hlavní ideál $(a + b\vartheta)$.

2.4 Třídová grupa

V této části zavedeme zobecnění ideálu v oboru celých algebraických čísel a seznámíme se s důležitým pojmem - třídová grupa.

Lemma 2.58. *Nechť K je číselné těleso s oborem celých algebraických čísel R a nechť T je podílové těleso R. Potom $A \subseteq T$ je konečně generovaný R-modul, právě když $A = \vartheta^{-1} I$ pro nějaké $\vartheta \in R$, $\vartheta \neq 0$, I ideál R.*

Důkaz. Nechť A je generované prvky $\alpha_1, \dots, \alpha_k$, $\alpha_i = \frac{\mu_i}{\vartheta_i}$, $\mu_i, \vartheta_i \in R$. Pak pro $\vartheta = \prod_{i=1}^{k} \vartheta_i$ je $\vartheta A \subseteq R$, tedy ϑA je ideál. Označme ho I, potom $A = \vartheta^{-1} I$.

Naopak, je-li $I \subseteq R$ ideál, je I i konečně generovaný R-modul. Proto je $\vartheta^{-1} I \subseteq T$ konečně generovaný R-modul pro každé $\vartheta \in R$, $\vartheta \neq 0$.

□

Definice 2.59. Nechť K je číselné těleso s oborem celých algebraických čísel R. Nechť T je podílové těleso R. Potom $A \subseteq T$, které je jako R-modul konečně generované, nazveme *lomený ideál*. Pokud $A = \alpha R$, $\alpha \in T$, $\alpha \neq 0$, jde o *hlavní lomený ideál*. Množinu všech lomených ideálů pro obor R budeme značit $\mathcal{I}_R$, množinu všech hlavních lomenných ideálů označíme $\mathcal{P}r_R$.

Lomené ideály můžeme chápat jako zobecnění klasických ideálů, které potom označujeme jako *celistvé*. Speciálně v Dedekindově oboru dostáváme několik důležitých vět o lomených ideálech, které budeme potřebovat pro algoritmus číselného síta. Na množině všech lomených ideálů $\mathcal{I}_R$ můžeme přirozeně definovat operaci násobení s neutrálním prvkem R. Tato operace je zřejmě komutativní a existenci inverzního prvku k lomenému ideálu A také není těžké dokázat. Stačí volit inverzi k A jako $\{\alpha \in T; \alpha A \subseteq R\}$. Dostáváme tedy grupu. Podobně ukážeme, že $\mathcal{P}r_R$ je podgrupa $\mathcal{I}_R$ (díky komutativitě normální). Můžeme proto faktorizovat.

Definice 2.60. Nechť $\mathcal{I}_R$ je grupa lomených ideálů oboru celých algebraických čísel R číselného tělesa K a nechť $\mathcal{P}r_R \subseteq \mathcal{I}_R$ je podgrupa hlavních lomených ideálů. Potom faktorgrupu $\mathcal{I}_R/\mathcal{P}r_R$ nazýváme *třídová grupa* (anglicky *class group*) a značíme $\mathcal{C}l_R$.

Zobecněním 2.36 dostáváme následující větu.

Věta 2.61. *Nechť $\mathcal{I}_R$ je grupa lomených ideálů oboru celých algebraických čísel R číselného tělesa K. Potom každý lomený ideál $A \in \mathcal{I}_R$ lze jednoznačně vyjádřit ve tvaru*

$$A = P_1^{r_1} \cdot \ldots \cdot P_k^{r_k},$$

kde P_i jsou prvoideály R a $r_i \in \mathbb{Z}$, $i = 1, \ldots k$.

Důkaz. Lomený ideál A vyjádříme podle lemmatu 2.58 jako

$$A = \vartheta^{-1} I = (\vartheta R)^{-1} I = Q_1^{-e_1} \cdot \ldots \cdot Q_n^{-e_n} \cdot \overline{Q}_1^{f_1} \cdot \ldots \cdot \overline{Q}_m^{f_m},$$

kde $e_i, f_i, n, m \in \mathbb{N}$. Pokud ideály zkombinujeme dostaneme požadované vyjádření. Kdyby bylo nejednoznačné, dostali bychom

$$P_1^{r_1} \cdot \ldots \cdot P_k^{r_k} = \overline{P}_1^{\overline{r}_1} \cdot \ldots \cdot \overline{P}_{k'}^{\overline{r}_{k'}}.$$

Nechť prvních $l \leq k$ mocnin r_i a prvních $l' \leq k'$ mocnin $\overline{r}_i$ je záporných. Po úpravě máme

$$\overline{P}_1^{\overline{r}_1} \cdot \ldots \cdot \overline{P}_{l'}^{\overline{e}_{l'}} \cdot P_{l+1}^{r_{l+1}} \cdot \ldots \cdot P_k^{r_k} = P_1^{r_1} \cdot \ldots \cdot P_l^{r_l} \cdot \overline{P}_{l'+1}^{\overline{r}_{l'+1}} \cdot \ldots \cdot \overline{P}_{k'}^{\overline{r}_{k'}}.$$

Tedy dvojí vyjádření celistvého ideálu v Dedekindově oboru R, ale to nelze. □

Triviálním důsledkem předchozí věty je

Důsledek 2.62. *Grupa lomených ideálů $\mathcal{I}_R$ číselného tělesa K je volná abelovská grupa generovaná množinou všech prvoideálů oboru celých algebraických čísel R.*

Uveďme ještě definici normy pro lomený ideál.

Definice 2.63. Nechť $A = P_1^{e_1} \cdot \ldots \cdot P_k^{e_k} \cdot Q_1^{-f_1} \cdot \ldots \cdot Q_l^{f_l} \in \mathcal{I}_R$ číselného tělesa K, kde $e_i, f_i \in \mathbb{N}$. Pak definujeme *normu lomeného ideálu A* jako

$$\mathcal{N}(A) = \frac{\prod_{i=1}^{k} \mathcal{N}(P_i)^{e_i}}{\prod_{i=1}^{l} \mathcal{N}(Q_i)^{f_i}}.$$

Nyní se zaměříme na třídovou grupu a ukážeme, že je konečná. Pro lepší představu o třídové grupě uveďme, že se podle definice skládá z tříd ekvivalence relace $\sim$ definované následovně:

$$A \sim B \iff \exists\, \alpha, \beta \in T, \alpha \neq 0, \beta \neq 0 : \alpha A = \beta B,$$

kde T je podílové těleso oboru celých algebraických čísel R a A, B jsou lomené ideály R. Dokázat konečnost třídové grupy je snadné. Využijeme skutečnost, že v každé třídě ekvivalence leží alespoň jeden celistvý ideál, a následující větu.

Věta 2.64. *Nechť K je číselné těleso s oborem celých algebraických čísel R. Potom existuje kladné reálné číslo λ závislé pouze na volbě K s vlastností, že každý nenulový ideál I oboru R obsahuje prvek ϑ, pro který platí*

$$|N_{\mathbb{Q}}^{K}(\vartheta)| \leq \lambda \cdot \mathcal{N}(I).$$

Důkaz. Zvolme nějakou celočíselnou bázi $\{\vartheta_1, \ldots, \vartheta_n\}$ oboru R. Dále označme $\sigma_1, \ldots, \sigma_n$ vnoření K do $\mathbb{C}$ s vlastností, že restrikce $\sigma_i \mid \mathbb{Q} = \mathrm{id}_{\mathbb{Q}}$ pro $i = 1, \ldots, n$. Ukážeme, že λ může být zvoleno jako

$$\lambda = \prod_{i=1}^{n} \sum_{j=1}^{n} |\sigma_i(\vartheta_j)|.$$

Pro ideál I z R zvolíme m tak, aby platilo $m^n \leq \mathcal{N}(I) < m^{n+1}$. Uvažujme $(m+1)^n$ prvků R tvaru

$$\sum_{j=1}^{n} a_j \vartheta_j, \;\; a_j \in \mathbb{Z},\, 0 \leq a_j \leq m.$$

Protože těchto prvků je více než je norma ideálu I, neboli počet prvků faktor okruhu R/I, musí mezi nimi existovat dva, které jsou kongruentní modulo I. Tedy jejich rozdíl leží v ideálu I a přitom víme, že je nenulový. Označíme ho ϑ. Platí

$$\vartheta = \sum_{j=1}^{n} b_j \vartheta_j, \;\; b_j \in \mathbb{Z},\, |b_j| \leq m.$$

Pro normu prvku ϑ dostáváme

$$|N_{\mathbb{Q}}^{K}(\vartheta)| = \prod_{i=1}^{n} |\sigma_i(\vartheta)| \leq \prod_{i=1}^{n} \Big(\sum_{j=1}^{n} |b_j| \cdot |\sigma_i(\vartheta_j)| \Big) \leq \lambda \cdot m^n \leq \lambda \cdot \mathcal{N}(I).$$

□

Důsledek 2.65. *Nechť K je číselné těleso s oborem celých algebraických čísel R. Nechť $\lambda \in \mathbb{R}$ jako v předešlé větě. Pak každá třída ekvivalence v třídové grupě oboru R obsahuje ideál I s normou $\mathcal{N}(I) \leq \lambda$.*

Důkaz. Uvažujme $C \in \mathcal{C}l_R$. Vybereme libovolný celistvý ideál $I \in C^{-1}$. Podle věty 2.64 najdeme $\vartheta \in I$. Zřejmě $(\vartheta) \subseteq I$ a tedy existuje ideál J takový, že $IJ = (\vartheta)$. Přitom jistě $J \in C$. Podle věty 2.43 dostáváme

$$\mathcal{N}(I)\mathcal{N}(J) = \mathcal{N}((\vartheta)) = |N_{\mathbb{Q}}^K(\vartheta)| \leq \lambda \mathcal{N}(I).$$

□

Důsledek 2.66. *Nechť $\mathcal{C}l_R$ je třídová grupa číselného tělesa K s oborem celých algebraických čísel R. Pak $\mathcal{C}l_R$ je konečná.*

Důkaz. Ukážeme, že v R existuje pouze konečně mnoho ideálů, pro které platí $\mathcal{N}(I) \leq \lambda$ (λ jako ve větě 2.64). Nechť $I = \prod_{i=1}^k P_i^{e_i}$, kde P_i jsou prvoideály v R. Pak dostáváme $\mathcal{N}(I) = \prod_{i=1}^k (\mathcal{N}(P_i))^{e_i} \leq \lambda$. Tedy $\mathcal{N}(P_i) = p_i^{f_i} \leq \lambda$, kde $f_i = \mathrm{f}(P_i|p_i\mathbb{Z})$. Ke konstrukci ideálů I můžeme tedy využít jen konečně mnoho prvoideálů z R a navíc jen do omezených mocnin. Proto existuje jen konečně mnoho tříd ideálů podle důsledku 2.65. □

2.5 Čtverce

Nechť K je číselné těleso s okruhem celých algebraických čísel R. Nyní ukážeme postup, jak s určitou pravděpodobností rozhodnout, zda nějaký prvek z R je možné vyjádřit jako druhou mocninu jiného prvku z R.

Označme $S = \{\alpha \in K^*;\ \alpha R = A^2,\ A \in \mathcal{I}_R\}$. Protože každý lomený ideál lze jednoznačně vyjádřit pomocí prvoideálů, existuje pro každé $\alpha \in S$ právě jeden lomený ideál A takový, že $\alpha R = A^2$. Budeme ho označovat A_α. Dále nechť U_R je množina invertibilních prvků oboru R.

Lemma 2.67. *Nechť K je číselné těleso. Pak množina S je podgrupou K^* a zobrazení $\varphi : S \to \mathcal{I}_R$, $\varphi(\alpha) = A_\alpha$, $\alpha \in S$ je homomorfismus grup.*

Důkaz. Nejprve ukážeme, že S je grupa. Nechť $\alpha, \beta \in S$ a $A_\alpha, A_\beta \in \mathcal{I}_R$ jsou odpovídající ideály. Pak zřejmě $\alpha\beta R = (A_\alpha A_\beta)^2$ a $\alpha^{-1}R = (A_\alpha^{-1})^2$. K dokázání, že φ je homomorfismus grup, stačí ukázat, že $\varphi(\alpha\beta) = \varphi(\alpha)\varphi(\beta)$. Jak už jsme naznačili, $\varphi(\alpha\beta) = A_{\alpha\beta} = A_\alpha A_\beta = \varphi(\alpha)\varphi(\beta)$. □

Nechť $\Omega_2(G) = \{x \in G;\ 2x = 0\}$ je podgrupa abelovské grupy G. Pokud je abelovská grupa G konečná, dostáváme $G/2G \cong \Omega_2(G)$. Pro lomený ideál $A \in \mathcal{I}_R$ označme $[A] \in \mathcal{C}l_R$ třídu lomených ideálů obsahující A.

Věta 2.68. *Nechť K je číselné těleso s oborem celých algebraických čísel R. Zobrazení $\varphi : S/(K^*)^2 \to \Omega_2(\mathcal{C}l_R)$, $\varphi(\alpha \cdot (K^*)^2) = [A_\alpha]$, $\alpha \in S$, je surjektivní homomorfismus grup s jádrem $U_R/(U_R)^2 \cong U_R(K^*)^2/(K^*)^2$.*

Důkaz. Nejdříve potřebujeme ukázat, že $[A_\alpha]^2 = [0]$. Podle definice množiny S je $A_\alpha^2 = \alpha R$, tedy hlavní lomený ideál, a proto $[A_\alpha]^2 \in \mathcal{P}r_R$. Pro libovolnou třídu $[A] \in \Omega_2(\mathcal{C}l_R)$ máme $A^2\mathcal{P}r_R = \mathcal{P}r_R$, tedy $A^2 \in \mathcal{P}r_R$, a proto $A^2 = \alpha R$,

z tohoto plyne surjektivita homomorfismu. Nyní budeme chtít dokázat, že jádro je $U_R(K^*)^2/(K^*)^2$. Nechť $\gamma \in U_R$. Pak dostáváme $\gamma R = R = R^2 = (1 \cdot R)^2$, tedy $\varphi(\gamma \cdot (K^*)^2) = [1 \cdot R]$. To je zřejmě hlavní lomený ideál, a proto $\gamma \in \operatorname{Ker} \varphi$. Naopak nechť $\alpha \in \operatorname{Ker} \varphi$. Pak existuje $\beta \in R$ tak, že $A_\alpha = \beta R$. Dostáváme $\alpha R = (\beta R)^2 = \beta^2 R$. Tedy $\alpha = \beta^2 \gamma$, kde $\gamma \in U_R$. Tím jsme ukázali, že α leží v $U_R(K^*)^2$. Pomocí 3. věty o izomorfismu máme

$$U_R(K^*)^2/(K^*)^2 \cong U_R/(U_R \cap (K^*)^2) \cong U_R/(U_R)^2.$$

K poslednímu izomorfismu potřebujeme ukázat, že $U_R \cap (K^*)^2 = (U_R)^2$. Zřejmě $U_R \cap (K^*)^2 \supseteq (U_R)^2$. Opačnou inkluzi dokážeme následovně. Nechť $\alpha \in U_R \cap (K^*)^2$. Protože α leží v $(K^*)^2$, existuje $\beta \in R$ tak, že $\alpha = \beta^2$ ($\alpha \in R$ je kořenem nějakého monického polynomu $f(x)$ nad $\mathbb{Z}$ a tedy β bude kořenem monického polynomu $f(x^2)$). Dostáváme $\beta R \supseteq \beta^2 R = \alpha R = R$, protože α leží i v U_R. Pak ovšem $\beta \in U_R$ a tedy $\alpha \in (U_R)^2$. □

Na $U_R/(U_R)^2$ a $\Omega_2(\mathcal{C}l_R)$ se můžeme dívat jako na vektorové prostory nad $\mathbb{Z}_2$. Protože třídová grupa $\mathcal{C}l_R$ je konečná, je jistě konečná i grupa $\Omega_2(\mathcal{C}l_R)$. Dále lze ukázat, že grupa $U_R/(U_R)^2$ je také konečná (s využitím Dirichletovy věty o jednotkách viz [22]). Podle věty 2.68 máme $S/(K^*)^2 \cong U_R/(U_R)^2 \oplus \Omega_2(\mathcal{C}l_R)$ izomorfismus konečných vektorových prostorů nad $\mathbb{Z}_2$. Dimenzi $U_R/(U_R)^2$ umíme určit, ale nedokážeme vhodně omezit dimenzi $\Omega_2(\mathcal{C}l_R)$, proto se omezíme pouze na konstatování, že je konečná.

Uvažujme prvek $\alpha \in S$ a s pomocí věty 2.68 zkusme zjistit, zda leží v $(K^*)^2$. Označme $\dim S/(K^*)^2 = k$. Zřejmě by stačilo nalézt k lineárně nezávislých forem $\psi_1, \ldots, \psi_k$ z $S/(K^*)^2$ do $\mathbb{Z}_2$ a zkontrolovat, jestli pro každé i platí $\psi_i(\alpha \cdot (K^*)^2) = 0$. Pak bychom dostali, že $\alpha \cdot (K^*)^2 = (K^*)^2$, tedy $\alpha \in (K^*)^2$. Jako lineární formy použijeme multiplikativní zobrazení z K^* do $\mathbb{Z}_2$, jejichž jádro obsahuje $(K^*)^2$ (samozřejmě v jádru nesmí být celé S, jinak se pro naše účely bude jednat o triviální formu). Multiplikativním zobrazením do těles se obecně říká charaktery, v našem případě kvadratické charaktery. Jak tyto kvadratické charaktery budeme generovat říká následující definice.

Definice 2.69. Nechť K je číselné těleso s oborem celých algebraických čísel R. Nechť P je prvoideál nad lichým prvočíslem p s $\mathrm{f}(P|p\mathbb{Z}) = f$. Pak pro $\vartheta \in R$ definujme *zobecněný Legenderův symbol* předpisem

$$\left(\frac{\vartheta}{P}\right) = \vartheta^{\frac{p^f-1}{2}} \bmod P.$$

Správnost definice plyne z toho, že R/P je těleso s p^f prvky. Tedy jeho cyklická grupa $(R/P)^*$ má $p^f - 1$ prvků. Zřejmě $(\frac{\vartheta\mu}{P}) = (\frac{\vartheta}{P})(\frac{\mu}{P}), \vartheta, \mu \in R$. Pokud $\vartheta \in R \setminus P$, pak $(\frac{\vartheta}{P}) \in \{-1, 1\}$ a prvky z $(K^*)^2$ se jistě zobrazují na 1. Dostáváme, že zobecněný Legenderův symbol poskytuje hledané kvadratické charaktery. Avšak zjistit lineární nezávislost generované lineární formy je obtížné, a proto raději využijeme následující lemma.

Lemma 2.70. *Nechť V je vektorový prostor nad $\mathbb{Z}_2$ dimenze n. Pak $n + r$, $r \in \mathbb{N}$, náhodně zvolených vektorů z V generuje celý prostor s pravděpodobností větší než $1 - 2^{-r}$.*

Důkaz. Nechť W je nadrovina V. Protože polovina vektorů z V leží ve W, je pravděpodobnost, že $n + r$ vektorů bude současně ležet v nadrovině W 2^{-n-r}. Každý nenulový vektor z V určuje právě jednu nadrovinu (ortogonální doplněk). Nenulových vektorů je $2^n - 1$. Pravděpodobnost, že $n + r$ náhodně zvolených vektorů bude ležet v jedné nadrovině je $2^{-n-r} \cdot (2^n - 1) = 2^{-r} - 2^{-n-r} < 2^{-r}$. □

Shrňme naše poznatky. Vidíme, že k rozhodnutí, zda nějaké $\vartheta \in R$ je kvadrátem jiného prvku z R, stačí použít dostatečný počet prvoideálů P_i (náhodně vybraných), ve kterých ϑ neleží. A zkontrolovat, jestli $(\frac{\vartheta}{P_i}) = 1$ pro tyto prvoideály. Čím více prvoideálů použijeme, tím větší je pravděpodobnost, že se skutečně jedná o kvadrát.

Kapitola 3

Číselné síto

3.1 Popis algoritmu

Než přejdeme k stručné historii a popisu čísleného síta, zavedeme neprve pojmy, které bude používat.

Definice 3.1. Nechť n, B jsou přirozená čísla. Řekneme, že n je *B-hladké*, pokud je dělitelné pouze prvočísly menšími než B. Pokud přesně i prvočísel z rozkladu n je větších než B, řekneme, že n je *i-částečně B-hladké*. Pro zpřehlednění se mez B neuvádí, pokud je z kontextu zřejmá.

Algoritmus číselného síta můžeme zařadit mezi moderní faktorizační algoritmy, které vychází z takzvané Kraitchikovy metody. Mějme číslo N, které chceme rozložit. Pokud bychom jednoduše uměli generovat dvojice x_i, y_i (netriviální v následujícím smyslu) tak, že

$$x_i^2 \equiv y_i^2 \mod N,$$

pak zřejmě s pravděpodobností alespoň $1/2$ je $\gcd(N, x_i - y_i)$ netriviální dělitel N. Při dostatečném počtu těchto dvojic jistě číslo N rozložíme.

Algoritmus číselného síta přebírá hlavní myšlenku z dalšího velmi významného faktorizačního algoritmu kvadratické síto, proto uvedeme vzájemnou podobnost. V algoritmech typu QS, případně CFRAC, se hledají kongruence tvaru

$$x_i^2 \equiv p_0^{e_0} p_1^{e_1} \cdot \ldots \cdot p_s^{e_s} \mod N, \tag{3.1}$$

kde $\{p_0, p_1, \ldots, p_s\}$ je předem zvolená množina prvočísel s určitými vlastnostmi (tato množina obsahuje i číslo -1). Těmto kongruencím říkáme v souladu s definicí 3.1 hladké relace. Tento pojem použijeme i v číselném sítu a budeme ho přesněji definovat později. Pokud se těchto hladkých relací nalezne dostatečný počet, je možné z nich sestavit dvojice x_i, y_i.

V algoritmu kvadratického síta se pomocí metody prosívání hledají vhodné relace z funkčních hodnot kvadratických polynomů, ale při faktorizaci velkých

čísel (nad 120 decimálních cifer) je pravděpodobnost nalezení kongruence tvaru 3.1 velmi malá, a proto tento algoritmus již není vhodný na rozložení takto velkých čísel. Problém spočívá ve velikosti funkčních hodnot polynomů a ani různá vylepšení algoritmu, kdy využíváme i částečně hladké relace, ze kterých sestavíme hladké relace, nejsou dostatečná. Proto se začal hledat nový způsob využívající podobným přístup k faktorizaci velkých čísel. Ještě dodojme, že později byly zkoušeny i 3-částečně hladké relace v kvadratickém sítu, ale ukázalo se, že jejich pozitivní přínos je až pro velmi velká čísla, která v té době nastupující číselné síto dokázalo faktorizovat ve zlomku času potřebného kvadratickým sítem.

Předchůdcem číselného síta byla faktorizace čísla N tvaru $r^3 + 2$ od J. M. Pollarda (viz [28]). Zde se poprvé objevuje myšlenka použít číselné těleso k hledání relací částečně podobných 3.1. Postupné zlepšování a upřesňování zvoleného postupu přináší algoritmus nyní označovaný jako speciální číselné síto (Special Number Field Sieve - SNFS, mezi první popisy patří například [21]), které umí faktorizovat čísla tvaru

$$N = r^e - s,$$

kde r, $|s|$ jsou relativně malá. Odhadovaná složitost tohoto algoritmu je pro $N \to \infty$:

$$L_N\left[\frac{1}{3}, \left(\frac{32}{9}\right)^{1/3}\right],$$

kde tzv. *L-funkci* $L_N[u, v]$ definujeme jako

$$L_N[u, v] = \exp((v + o(1))(\log N)^u(\log\log N)^{1-u}), \;\; u,\, v,\, N \in \mathbb{R}.$$

Zřejmě $L_N[1, v] = N^{v+o(1)}$ (odpovídá exponenciální složitosti) a $L_N[0, v] = (\log N)^{v+o(1)}$ (odpovídá polynomiální složitosti). Jedná se tedy o určitou interpolaci mezi exponenciální a polynomiální funkcí v $\log N$. Algoritmy číselného síta mají vždy $u = \frac{1}{3}$, to je velké teoretické zrychlení o proti $u = \frac{1}{2}$ pro kvadratické síto. Speciální číselné síto je dodnes nejrychlejší faktorizační algoritmus ale se zmíněným omezením. Poslední faktorizační rekord je číslo $2^{1039} - 1$ s 313 decimálními ciframi ([1]).

Přibližně na počátku devadesátých let J. P. Buhler, H. W. Lenstra jr. a Carl Pomerance přišli s upravenou verzí algoritmu tak, aby bylo možné faktorizovat libovolné číslo N (viz [3]). Byly vyřešeny problémy s hledáním vhodných polynomů, správné kombinování nalezených relací i výpočet odmocniny v oboru celých algebraických čísel. Tímto algoritmem (General Number Field Sieve - GNFS) se budeme v této práci zabývat podrobněji. Nejprve popišme hlavní myšlenky.

Nechť N je číslo, které chceme faktorizovat. Předpokládejme, že máme dva monické ireducibilní polynomy $f_1,\, f_2 \in \mathbb{Z}[x]$ stupně $d_1,\, d_2$ $(d_1 + d_2 > 2)$ a číslo $m \in \mathbb{Z}$, pro které platí

$$f_i(m) \equiv 0 \mod N.$$

Postup, jak nalézt polynomy f_1 a f_2, popíšeme později. Na základě stupňů polynomů pak mluvíme o metodě typu (d_1, d_2). Nechť ϑ_i je kořenem polynomu f_i, $i = 1, 2$. Potom zřejmě f_i je minimálním polynomem ϑ_i a $K_i = \mathbb{Q}[\vartheta_i]$ je číselné těleso. Uvažujme homomorfismy φ_i

$$\varphi_i : \mathbb{Z}[\vartheta_i] \to \mathbb{Z}_N, \;\; \varphi_i(\vartheta_i) = m, \; \varphi_i(a) \equiv a \mod N, \; a \in \mathbb{Z}.$$

Tyto homomorfismy jsou jistě dobře definovány. Našim cílem bude nalézt čísla $\alpha_i \in \mathbb{Z}[\vartheta_i]$ tak, aby

$$\varphi_1(\alpha_1^2) \equiv \varphi_2(\alpha_2^2) \mod N. \tag{3.2}$$

Pak pomocí vlastností homomorfismu dostáváme hledaný pár x_i, y_i

$$(\varphi_1(\alpha_1))^2 \equiv (\varphi_2(\alpha_2))^2 \mod N$$

a můžeme se pokusit faktorizovat N.

Abychom nalezli $\alpha_i \in \mathbb{Z}[\vartheta_i]$ s vlastností 3.2, využijeme hlavní ideály tvaru $(a + b\vartheta_i)$, kde $a, b \in \mathbb{Z}$ a $\gcd(a, b) = 1$. Budeme s nimi pracovat v Dedekindově oboru O_{K_i}, ale také zřejmě platí $a + b\vartheta_i \in \mathbb{Z}[\vartheta_i]$. Podle věty 2.36 víme, že každý ideál v O_{K_i} se jednoznačně rozkládá na součin prvoideálů. Nechť $FB_i = \{P_1, \ldots, P_{k_i}\}$ je množina všech prvoideálů v O_{K_i} nad prvočísly menšími než předem pevně daná mez, kterou nalezneme postupem popsaným v části 2.3 (přesněji řečeno nepracujeme vždy v O_{K_i}, ale v nějakém dostatečně velkém plnooboru O_{K_i}, do kterého již všechny tyto prvoideály patří a leží v něm tedy i všechny ideály těmito prvoideály generované). Tyto množiny nazýváme faktorizační báze. Podle tvrzení 2.57 a poznámky pod ním snadno určíme faktorizaci hlavního ideálu tvaru $(a + b\vartheta_i)$ v O_{K_i} podle jeho normy. Normu můžeme vypočítat pomocí homogenního polynomu $F_i(x, y) = y^{d_i} f(\frac{x}{y}) \in \mathbb{Z}[x, y]$, jak bylo dokázáno v lemmatu 2.11 (podrobněji se tímto zabývá část 3.3). Na základě podobnosti s relacemi v kvadratickém sítu zavádíme následující definici.

Definice 3.2. Nechť pro nesoudělná $a, b \in \mathbb{Z}$ platí, že $N(a+b\vartheta_i)$ je B_i-hladká. Pak řekneme, že dvojice (a, b) je *hladká relace*. Pokud je $N(a+b\vartheta_i)$ j_i-částečně B_i-hladká, pak řekneme, že (a, b) je j_1, j_2-*částečně hladká relace*.

Pokud nalezneme dostatečně velký počet hladkých relací můžeme, podobně jako v algoritmu kvadratického síta, vybrat podmnožinu indexů M tak, abychom dostali

$$\begin{aligned}
\prod_{j \in M} ((a_j + b_j\vartheta_1)O_{K_1}) &= P_1^{2e_1} P_2^{2e_2} \ldots P_{k_1}^{2e_{k_1}} = I^2, \\
\prod_{j \in M} ((a_j + b_j\vartheta_2)O_{K_2}) &= Q_1^{2f_1} Q_2^{2f_2} \ldots Q_{k_2}^{2f_{k_2}} = J^2.
\end{aligned}$$

Nyní ale na rozdíl od oboru celých čísel neplatí, že by obecně ideály I a J byly také hlavní. Pokud bychom uměli zařídit, aby I, J byly hlavní ideály, tedy $I = \alpha_1^2 O_{K_1}$, $J = \alpha_2^2 O_{K_2}$, pak ale opět obecně neplatí požadované

$$\begin{aligned}\prod_{j\in M}(a_j + b_j\vartheta_1) &= \alpha_1^2,\\ \prod_{j\in M}(a_j + b_j\vartheta_2) &= \alpha_2^2.\end{aligned}$$

Ovšem podle části 2.5, kde jsme se zabývali hledáním kvadrátů v O_{K_i}, zřejmě $\prod_{j\in M}(a_j + b_j\vartheta_i) \in S_i = \{\gamma \in K_i^*;\, \gamma O_{K_i} = A^2,\, A \in \mathcal{I}_{O_{K_i}}\}$. Pokud bychom pomocí postupu s kvadratickými charaktery s dostatečně velkou pravděpodobností dokázali, že $\prod_{j\in M}(a_j + b_j\vartheta_i) \in (K_i^*)^2$, pak zřejmě $\prod_{j\in M}(a_j + b_j\vartheta_i) = \alpha_i^2$. A uvažovaný ideál I, resp. J, je hlavní. Tedy dostáváme existenci hledaných α_i. Netriviální je jejich následné získání z $\prod_{j\in M}(a_j + b_j\vartheta_i)$, protože potřebujeme odmocnit v O_{K_i}. To není snadné již z důvodu, že pouhé roznásobení výrazu by mohlo být časově nejnáročnější z celého algoritmu číselného síta. V části 3.6 popíšeme několik algoritmů, jak odmocninu takovéhoto výrazu rychle spočítat a navíc dosáhnout toho, aby α_i ležely v $\mathbb{Z}[\vartheta_i]$.

Shrňme tedy postup, jak získat vhodnou podmnožinu z množiny nalezených faktorizovaných ideálů $\{(a_1 + b_1\vartheta_i), \ldots, (a_r + b_r\vartheta_i)\}$. Nechť jednotlivé faktorizace jsou

$$\begin{aligned}(a_1 + b_1\vartheta_1) &= P_1^{e_{1\,1}} P_2^{e_{1\,2}} \ldots P_{k_1}^{e_{1\,k_1}}\\ &\vdots \qquad\qquad \vdots\\ (a_r + b_r\vartheta_1) &= P_1^{e_{r\,1}} P_2^{e_{r\,2}} \ldots P_{k_1}^{e_{r\,k_1}}\\ (a_1 + b_1\vartheta_2) &= Q_1^{f_{1\,1}} Q_2^{f_{1\,2}} \ldots Q_{k_2}^{f_{1\,k_2}}\\ &\vdots \qquad\qquad \vdots\\ (a_r + b_r\vartheta_2) &= Q_1^{f_{r\,1}} Q_2^{f_{r\,2}} \ldots Q_{k_2}^{f_{r\,k_2}}\end{aligned}$$

Dále uvažujme prvoideály $\{U_{i\,1}, \ldots, U_{i\,l_i}\}$ v O_{K_i} nad nespeciálními prvočísly, která jsou ostře větší než zvolené meze pro faktorizační báze. Prvoideály použijeme jako kvadratické charaktery. Protože prvky $a_j + b_j\vartheta_i$ v nich nemohou ležet, nikdy nedostaneme $(\frac{a_j + b_j\vartheta_i}{U_{i\,k}}) = 0$. Definujme $g_{j\,k} = 1$, pokud $(\frac{a_j + b_j\vartheta_1}{U_{1\,k}}) = -1$, jinak $g_{j\,k} = 0$, a $h_{j\,k} = 1$, pokud $(\frac{a_j + b_j\vartheta_2}{U_{2\,k}}) = 1$, jinak $h_{j\,k} = 0$, kde $j = 1, \ldots, r$, $k = 1, \ldots, l_i$. Sestavme matici $\mathbf{B}$ takto

$$\mathbf{B}^T = \begin{pmatrix} e_{1\,1} & \cdots & e_{1\,k_1} & f_{1\,1} & \cdots & f_{1\,k_2} & g_{1\,1} & \cdots & g_{1\,l_1} & h_{1\,1} & \cdots & h_{1\,l_2}\\ \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots & \vdots & & \vdots\\ e_{r\,1} & \cdots & e_{r\,k_1} & f_{r\,1} & \cdots & f_{r\,k_2} & g_{r\,1} & \cdots & g_{r\,l_1} & h_{r\,1} & \cdots & h_{r\,l_2}\end{pmatrix}.$$

Tedy jeden sloupec matice $\mathbf{B}$ představuje nejprve rozklad ideálu $(a_j + b_j\vartheta_1)$ ve faktorizační bázi v oboru celých algebraických čísel O_{K_1} (mocniny jednotlivých prvoideálů), pak rozklad ideálu $(a_j + b_j\vartheta_2)$ v O_{K_2}, dále převedené kvadratické charaktery z prvku $a_j + b_j\vartheta_1$ v O_{K_1} (charaktery vznikly ze zvolených prvoideálů) a nakonec převedené charaktery z prvku $a_j + b_j\vartheta_2$ v O_{K_2}. Řešení $\mathbf{w} = (w_1, \ldots, w_r)$ rovnice

$$\mathbf{Bw} = \mathbf{0}$$

nad $\mathbb{Z}_2$ určuje hledanou podmnožinu indexů $M = \{i \in \mathbb{N} : w_i = 1\}$. Matice $\mathbf{B}$ je v praxi příliš velká (milióny řádků) pro použití klasických metod řešení, a proto se používají speciální algoritmy, které jsou popsány v části 3.5, navíc tyto algoritmy poskytují více řešení, tedy dostaneme několik výsledných párů x_i, y_i, které můžeme použít k faktorizaci čísla N.

Uveďme ještě odhadovanou složitost algoritmu pro $N \to \infty$:

$$L_N\left[\frac{1}{3}, \left(\frac{64}{9}\right)^{1/3} + o(1)\right].$$

Vidíme, že GNFS je pomalejší než SNFS, ale stále významně rychlejší než kvadratické síto. V současnosti se SNFS nejčastěji implementuje stejně jako GNFS, jediný rozdíl je v generování polynomů f_1, f_2. SNFS využívá speciálního tvaru N k vytvoření polynomu f_1, který obvykle nemá velký stupeň a koeficienty jsou velmi malé, a f_2 se volí jako $x - m$, případně složitěji. Jak popíšeme v části 3.2 pro GNFS existuje několik algoritmů k nalezení nejlepších kandidátů na f_1, ale ty dokáží nalézt pouze polynomy, jejichž koeficienty jsou přibližně $\sqrt[d_1]{N}$, proto je SNFS podstatně rychlejší. Opět je $f_2(x) = x - m$ ve většině implementacích, ale ukážeme i jiné varianty.

Pro zjednodušení popisu algoritmu jsme polynomy f_1 a f_2 volili jako monické, ale v praxi se mnohem častěji používají nemonické polynomy, které algoritmus zrychlují (velikost koeficientů u polynomu f_i se teoreticky pohybuje kolem $\sqrt[d_i+1]{N}$). V případě použití nemonických polynomů již neplatí, že by ϑ_i bylo celé algebraické číslo. Musíme tedy provést několik změn v algoritmu. Nechť $f_i(x) = \sum_{j=0}^{d_i} c_{i\,j} \cdot x^j, c_{i\,d_i} > 1$. Uvažujme polynomy

$$\widehat{f_i}(x) = f_i(x/c_{i\,d_i}) \cdot c_{i\,d_i}^{d_i-1}.$$

Tyto polynomy jsou již monické a jejich kořenem jsou zřejmě $\widehat{\vartheta_i} = c_{i\,d_i}\vartheta_i$. Proto $\widehat{\vartheta_i}$ jsou celá algebraická čísla a volíme $K_i = \mathbb{Q}[\widehat{\vartheta_i}] = \mathbb{Q}[\vartheta_i]$. Stále budeme hledat rozklad ideálů tvaru $(a+b\vartheta_i)$, které jsou ale nyní lomené. Normu, kterou potřebujeme k rozložení na prvoideály, spočítáme následovně

$$\begin{aligned}\mathcal{N}((a+b\vartheta_i)) &= N(a+b\vartheta_i) = N(c_{i\,d_i}^{-1}c_{i\,d_i}(a+b\vartheta_i)) = \\ &= N(c_{i\,d_i}^{-1})N(c_{i\,d_i}a+b\widehat{\vartheta_i}) = c_{i\,d_i}^{-d_i}\widehat{F_i}(c_{i\,d_i}a,-b) = \\ &= c_{i\,d_i}^{-d_i}\widehat{f_i}(c_{i\,d_i}a/(-b))(-b)^{d_i} = c_{i\,d_i}^{-d_i}f_i(a/(-b))c_{i\,d_i}^{d_i-1}(-b)^{d_i} = \\ &= F_i(a,-b)c_{i\,d_i}^{-1}.\end{aligned}$$

Máme tedy $(a+b\vartheta_i) = (c_{i\,d_i}a+b\widehat{\vartheta_i})(c_{i\,d_i})^{-1}$, přitom lomený ideál $(c_{i\,d_i})^{-1}$ dostáváme vždy. Výsledná množina M tedy musí mít sudý počet prvků, abychom dosáhli ve výsledku sudé mocniny u $(c_{i\,d_i})$. Toho docílíme tak, že do matice **B** přidáme navíc jeden řádek samých jedniček. Pokud vedoucí koeficient $c_{i\,d_i}$ je nesoudělný s $F_i(a,-b)$, znamená to, že ideál $(c_{i\,d_i}a+b\widehat{\vartheta_i})$ není dělitelný žádným prvoideálem z rozkladu $(c_{i\,d_i})^{-1}$. Proto z $F_i(a,-b)$ získáme přímo rozklad $(c_{i\,d_i}a+b\widehat{\vartheta_i})$. V opačném případě musíme rozklad dopočítat s ohledem na ideál $(c_{i\,d_i})^{-1}$. V prosívací fázi tedy hledáme rozklad ideálu $(c_{i\,d_i}a+b\widehat{\vartheta_i})$ opět pouze pomocí rozkladu $F_i(a,-b)$. Také musíme přizpůsobit odmocninovou fázi, aby správně započítala $(c_{i\,d_i})^{-1}$. Detailnější popis změn uvedeme až u jednotlivých částí.

Z přehledového popisu výše vyplývá, že můžeme algoritmus číselného síta rozdělit na pět samostatných fází. Tyto fáze podrobněji rozebereme v následujících částech, jsou to

1. Výběr polynomů
   V této fázi hledáme nejlepší dvojici polynomů f_1, f_2. Ukážeme jak generovat kandidáty na tyto polynomy a jaká kritéria používáme k určení nejlepších. Zaměříme se na typ $(d,1)$, ale popíšeme i Montgomeryho metodu $(2,2)$, která je vhodná pro menší čísla (okolo 110 cifer).

2. Prosívací fáze
   V této fázi pomocí prosívání hledáme dostatečný počet hladkých a částečně hladkých relací. Popíšeme několik možných přístupů k prosívání a zaměříme se i na implementační problémy.

3. Zpracování relací
   Jedná se o pomocnou fázi, ve které provedeme pročištění faktorizační báze a nalezených relací. Z částečných relací vytvoříme hladké relace. Dopočítáme kvadratické charaktery a vytvoříme finální matici.

4. Lineární fáze
   Nejobecnější fáze algoritmu. Pouze vyřešíme vzniklou matici. Tato matice je velmi velká a řídká, proto je potřeba používat speciální algoritmy, které zde popíšeme.

5. Odmocninová fáze
   V poslední fázi z řešení matice sestavíme prvek α_i^2 a spočítáme jeho odmocninu v $\mathbb{Z}[\vartheta_i]$.

3.2 Výběr polynomů

První fáze algoritmu číselného síta je výběr polynomů. Našim cílem je nalézt ireducibilní polynomy $f_1, f_2 \in \mathbb{Z}[x]$ a číslo $m \in \mathbb{Z}$, pro které platí

$$f_i(m) \equiv 0 \mod N. \tag{3.3}$$

Z popisu algoritmu je patrné, že našim hlavním požadavkem na tyto polynomy je, aby co nejvíce funkčních hodnot homogenních polynomů $F_i(x, y)$ bylo B_i-hladkých. Mluvíme pak o velké výtěžnosti polynomů. Hledání polynomů patří k nejméně teoreticky prozkoumaným částem algoritmu. Existují sice metody generující dobré polynomy, ale stále se nedají srovnávat s polynomy používanými ve speciálním číselném sítu. Přitom výběr polynomů má zásadní vliv na rychlost celého algoritmu a pouze z tohoto důvodu je speciální varianta číselného síta výrazně rychlejší.

Všechny současné metody hledaní polynomů se skládají ze dvou částí. Nejprve nagenerujeme velký počet dobrých kandidátů na polynomy f_1 a f_2. Potom se snažíme mezi nimi nalézt několik nejlepších. S nimi můžeme provést zkušební prosívání a určit nejlepší dvojici polynomů. Případně můžeme rovnou vybrat nejlepší pár. Jak ale poznat, který pár polynomů je dobrý, a který je nejlepší? Vyjdeme z hlavního požadavku, aby co nejvíce funkčních hodnot bylo hladkých. Zřejmě jedním ze způsobů jak tohoto dosáhnout je, aby funkční hodnoty byly co nejmenší. Proto zavádíme následující definici.

Definice 3.3. Řekneme, že polynom $f(x) = \sum_{j=0}^{d} c_j x^j \in \mathbb{Z}[x]$ s kořenem m modulo N je χ-*malý*, pokud $\chi \in \mathbb{R}$ je největší z čísel $|c_j|/m$, $j = 0, \ldots, d$.

To, že je nějaký polynom χ-malý, ještě neznamená, že funkční hodnoty homogenního polynomu budou malé na prosívací oblasti $I_a \times I_b$. Pokud ovšem volíme $I_a = [-M, M]$ a $I_b = [1, M]$ pro $M > 0$, pak definice 3.3 dokáže rozlišovat mezi „malými“ a „velkými“ funkčními hodnotami homogenních polynomů. Hlavní výhodou předchozí definice je fakt, že dokážeme velice rychle určit, zdali je polynom χ-malý pro nějaké předem zvolené χ. Je to tedy vhodný způsob pro první část, kde nám stačí pouze hrubé rozlišení mezi dobrými a špatnými polynomy. V druhé části hledání použijeme mnohem přesnější metodu odhadu velikosti (viz níže), která přibližně odpovídá námi očekávanému integrálu z $|F_i(x, y)|$ na oblasti $I_a \times I_b$.

Dlouhou dobu bylo ohodnocení polynomů podle velikosti jediným kritériem pro určení nejlepšího páru s tím, že se u nemonických polynomů volil vedoucí koeficient tak, aby byl dělitelný několika malými prvočísly. Potom pokud $p|c_{i\,d}$ (vedoucí koeficient f_i), a zároveň $p|b$, pak $p|F_i(x, b)$ pro všechna $x \in I_a$ (označují se jako *projektivní kořeny*). Dále se požadovalo, aby polynom f_i měl co nejvíce kořenů modulo několik malých prvočísel. Díky tomuto bylo více funkčních hodnot $F_i(x, y)$ dělitelných těmito prvočísly. Avšak tato poslední kritéria nebyla nijak kvantifikována, a proto nebylo možné pomocí nich polynomy porovnávat, případně určit výsledné ohodnocení polynomů

započítávající velikost a tzv. kořenové vlastnosti. Až Brian Murphy v [25] navrhuje funkci α, která by hodnotila kořenové vlastnosti, a uvádí i metodu výpočtu ohodnocení vycházející z velikosti funkčních hodnot a kořenových vlastností. Vliv funkce α na výtěžnost pak dokazuje jak teoreticky, tak i pomocí prováděných experimentů. V krátkosti funkci α popíšeme.

Nejprve zavedeme několik pomocných pojmů.

Definice 3.4. *Náhodné číslo* i_r je rovnoměrně zvolené celé číslo z intervalu $1 \leq i_r \leq r$.

Definice 3.5. Nechť $\mathrm{val}_p(r)$ je p-valuace čísla $r \in \mathbb{Z}$ (exponent největší mocniny p dělící r). Pak $\mathrm{cont}_p(S) = \mathrm{E}(\mathrm{val}_p(r))$ pro r z množiny $S \subseteq \mathbb{Z}$.

Pomocí $\mathrm{cont}_p(S)$ můžeme přibližně vyjádřit B-hladké číslo r z množiny S jako

$$\log(r) \approx \sum_{p \leq B} \mathrm{cont}_p(S) \log(p). \tag{3.4}$$

Číslu $\exp(\sum_{p \leq B} \mathrm{cont}_p(S) \log(p))$ říkáme *typická S-hodnota* (pokud S je množina funkčních hodnot polynomu $F(x, y)$ mluvíme o *typické F-hodnotě*). Pro dostatečně velké $S' \subseteq S$ můžeme $\mathrm{cont}_p(S)$ aproximovat jako

$$\mathrm{cont}_p(S) \approx \frac{\sum_{r \in S'} \mathrm{val}_p(r)}{|S'|}. \tag{3.5}$$

Pro účely ohodnocení polynomů budeme uvažovat tři různé množiny S. Pro ně uvedeme jednoduché vzorce pro výpočet $\mathrm{cont}_p(S)$ tak, jak jsou uvedeny v [25].

1. Náhodná čísla i_r: $\mathrm{cont}_p(S) = \mathrm{cont}_p(i_r) = \frac{1}{p-1}$
2. Funkční hodnoty polynomu $f(x)$: $\mathrm{cont}_p(S) = \mathrm{cont}_p(f) = \frac{q_p}{p-1}$, kde q_p je počet různých kořenů polynomu f modulo prvočíslo p.
3. Funkční hodnoty polynomu $F(x, y)$: $\mathrm{cont}_p(S) = \mathrm{cont}_p(F) = \frac{pq_p}{p^2-1}$, kde q_p je počet různých kořenů polynomu f modulo prvočíslo p.

Poslední dva vzorce platí pro nespeciální prvočísla. Pro speciální prvočísla můžeme použít odhad 3.5. Pro svou lepší přesnost je odhad 3.5 v praxi upřednostňován před uvedenými vzorci při výpočtu $\mathrm{cont}_p(S)$ pro malá prvočísla.

Uvažujem nyní náhodné číslo i_r. Z výše uvedeného vyplývá, že číslo

$$\log(i_r) - \sum_{p \leq B} \frac{\log(p)}{p-1}$$

odpovídá očekávané hodnotě logaritmu čísla i_r po vydělení všech prvočísel dělících i_r a menších než B (v maximálním možné mocnině). Jak je uvedeno v další části, odpovídá to výsledku po prosívání náhodných čísel. Podobně pro čísla $r = F(x, y)$ nebo $r = f(x)$ dostáváme

$$\log(r) - \sum_{p \leq B} \mathrm{cont}_p(r) \log(p).$$

Funkci α definujeme jako rozdíl těchto hodnot.

Definice 3.6. Nechť S je množina celých čísel a $0 < B \in \mathbb{Z}$. Pak definujeme

$$\alpha(S) = \sum_{p \leq B} \left(\frac{1}{p-1} - \mathrm{cont}_p(S) \right) \log(p).$$

Hodnota α tedy symbolizuje v našem případě rozdíl mezi náhodným zbytkem a zbytkem z funkčních hodnot polynomu $F(x, y)$. Čím je α menší, tím lepší jsou funkční hodnoty polynomu $F(x, y)$ oproti náhodným číslům stejné velikosti. Neboli funkční hodnoty mají stejné vlastnosti (co do hladkosti) jako náhodná čísla velikosti $F(x, y) \cdot e^{\alpha(F)}$. Při generování polynomů a určování nejlepšího páru proto upřednostňujeme polynomy s velmi malými hodnotami α. Metodu kombinující oba parametry k určení nejlepší dvojice uvedeme později.

V současnosti jsou upřednostňovány pouze dva typy metod generující polynomy. Nejpoužívanější a jediná, kterou lze použít na rekordní faktorizace, je metoda typu $(d, 1)$. Tedy druhý polynom je lineární a počítá se pouze s jedním číselným tělesem. Druhá používaná metoda je typu (d, d). Jediným použitelným zástupcem je Montgomeryho metoda kvadratických polynomů. Generování polynomů vyšších stupňů je prozatím příliš časově náročné, přitom se ale předpokládá, že tyto metody by mohly poskytovat mnohem lepší polynomy pro prosívání.

3.2.1 Base-m metoda

Nejjednodušší způsob, jak generovat polynomy s vlastností 3.3, je tzv. base-m metoda. Tato metoda je typu $(d, 1)$. Vhodný stupeň d polynomu f_1 je možné vypočítat na základě předpokládaných velikostí funkčních hodnot výsledných polynomů na uvažované prosívací oblasti (viz [3] a [25]). Doporučené hodnoty jsou $d = 4$ pro čísla v rozmezí 80-120 decimálních cifer (pro tyto čísla se ale doporučuje použít jinou metodu viz níže), $d = 5$ pro rozmezí 120-220 a $d = 6$ pro rozmezí 220-300. Hodnoty jsou pouze přibližné a počítají s použitím nemonických polynomů.

V základní variantě je base-m metoda jen prosté vyjádření čísla N v číselné soustavě o základu m,

$$N = \sum_{j=0}^{d} a_j^{(m)} m^j,$$

které se použije k sestavení polynomu f_1. Polynom f_2 volíme jako $x - m$. Pokud generujeme monický polynom f_1, volíme $m = O(\sqrt[d]{N})$. V opačném případě

volíme $m = O(\sqrt[d+1]{N})$. Zřejmě k vytvoření polynomu $f_1(x) = \sum_{j=0}^{d} c_{1\,j} x^j$ nepoužijeme přímo koeficienty $a_j^{(m)}$, ale provedeme následující úpravu. Pokud $a_j^{(m)} > \lfloor m/2 \rfloor$, pak $c_{1\,j} = m - a_j^{(m)}$ a $c_{1\,j+1} = c_{1\,j+1} + 1$. Jinak $c_{1\,j} = a_j^{(m)}$. Protože velikost koeficientů je přibližně $O(m)$, preferujeme nemonické polynomy. Nemonické polynomy mají také lepší kořenové vlastnosti než monické (díky projektivním kořenům). Po vytvoření velkého množství kandidátů vybere ten nejlepší na základě vlastností popsaných výše. Jeden z možných ohodnocovacích způsobů uvedeme později.

Při generování těchto polynomů vyjdeme z [25]. Nejprve ale několik pozorování. Mějme pevně zvolený vedoucí koeficient a_d. Je zřejmé, že následující koeficient $a_{d-1}^{(m)}$ bude malý (nebo $m - a_{d-1}^{(m)}$), pokud m bude blízko hodnoty, při které se vedoucí koeficient mění. Takové m vypočítáme jako

$$m = \left\lfloor \sqrt[d]{\frac{N}{a_d}} \right\rfloor . \tag{3.6}$$

Přitom následující koeficient závisí lineárně na m, protože platí

$$a_{d-1}^{(m+k)} \equiv (a_{d-1}^{(m)} - dka_d^{(m)}) \bmod (m+k). \tag{3.7}$$

Z tohoto snadno odvodíme, jaké hodnoty $k \in \mathbb{Z}$ máme použít, aby platilo $|a_{d-1}^{(m+k)}| \leq \chi m$. Podobná podmínka pro koeficient $a_{d-2}^{(m+k)}$ je již příliš složitá, aby ji bylo možné rozumně využít. Postupujeme tedy následovně. Nejprve určíme interval I, ze kterého budeme vybírat vedoucí koeficient. Dále zvolíme číslo c tak, aby bylo dělitelné velkým počtem malých prvočísel (pro zlepšení kořenových vlastností). V intervalu I nalezname všechny čísla, která jsou násobkem c. To budou naše vedoucí koeficienty. Pro každý vedoucí koeficient vypočítáme odpovídající m podle 3.6 a určíme rozsah pro k podle 3.7 tak, aby $a_{d-1}^{(m+k)}$ bylo χ-malé. Nakonec dopočítáme všechny možné rozvoje a pro ty, které jsou χ-malé vypočítáme přibližně i hodnotu α. Pokud je α dostatečně malé, označíme tento rozvoj jako vhodného kandidáta.

Když nalezneme dostatečný počet kandidátů, začneme je postupně ohodnocovat, abychom určili několik nejlepších. S těmi potom můžeme provést zkušební prosívání a určit nejlepší polynom podle skutečné výtěžnosti. Nebo se při výběru nejlepšího polynomu můžeme spolehnout pouze na ohodnocení. Jako příklad ohodnocení použijeme návrh z [25]. Polynomy f_2 jsou všechny témeř stejné, a proto se zaměříme přednostně na polynomy f_1. Odpovídající homogenní polynom $F_1(x, y)$ můžeme vyjádřit pomocí polárních souřadnic

$$F_1(x, y) = r^d F_1(\cos(\theta), \sin(\theta)).$$

Jako prosívací oblast pro tuto metodu obvykle volíme $[-M, M] \times [1, M]$ (viz následující část). Proto nám k porovnání velikosti funkčních hodnot různých polynomů na této oblasti stačí vzít několik hodnot na půlkruhu jednotkové kružnice. Definujme

$$u_{F_1}(\theta_j) = \frac{\log(|F_1(\cos(\theta_j), \sin(\theta_j))|) + \alpha(F_1)}{\log B_1},$$

kde θ_j volíme jako střed j-tého podintervalu $[0, \pi]$, $j = 1, \ldots, K$. Celkové ohodnocení polynomu F_1 je

$$\mathbb{E}(F_1) = \sum_{j=1}^{K} \rho(u_{F_1}(\theta_j)).$$

Kde ρ je Dickmanova funkce ([7]). Pokud chceme, můžeme zahrnout do ohodnocení i druhý polynom

$$\mathbb{E}(F_1, F_2) = \sum_{j=1}^{K} \rho(u_{F_1}(\theta_j))\rho(u_{F_2}(\theta_j)).$$

Čím nižší hodnota $\mathbb{E}(F_1, F_2)$, tím lepší pár polynomů F_1, F_2. Pro výpočet ohodnocení již potřebujeme přesnější hodnoty $\alpha(F_i)$. Při jejich výpočtu postupujeme tak, že pro menší prvočísla (< 100) vypočítáme $\mathrm{cont}_p(F_i)$ podle 3.5 a pro ostatní teprve použijeme aproximační vzorce.

3.2.2 Zkosené polynomy

Ukazuje se, že polynomy generované jednoduchou base-m metodou nemají tak dobré kořenové vlastnosti, jak by bylo potřeba. Proto Brian Murphy a P. L. Montgomery přisli s variantou, která generuje polynomy s mnohem lepšími kořenovými vlastnostmi. V ní se využívá tzv. *zkosení polynomů*, tedy nestejných velikostí koeficientů. To s sebou přináší i změnu rozměrů prosívací oblasti. Ke zkosení polynomů využíváme dvě operace

(i) posunutí: $f_t(x) = f(x-t)$ pro nějaké $t \in \mathbb{Z}$. Tato změna nemění kořenové vlastnosti, ale může zlepšit velikost koeficientů. K zachování 3.3 musíme posunout i m, $m_t = m + t$. Pro zjednodušení zápisu budeme posunutí o t značit indexem.

(ii) rotace: $f_P(x) = f(x) + P(x)(x - m)$ pro nějaký polynom $P(x) \in \mathbb{Z}[x]$. Stupeň tohoto polynomu by měl být nejvýše $d - 2$ (nejčastěji se používá pouze lineární polynom). Rotace mění jak kořenové vlastnosti tak velikost. Přitom je zřejmě zachována podmínka 3.3.

Nejvíce využijeme rotaci. Nejprve pomocí ní zajistíme zkosení polynomu a poté vylepšíme kořenové vlasnosti vzniklého polynomu tak, abychom nepokazili jeho velikost.

Opět začneme výběrem vedoucího koeficientu. V tomto případě ale vedoucí koeficient přímo poskládáme z násobků mocnin malých prvočísel. Z důvodů zkosení ho nevolíme příliš velký ($\approx O(\sqrt[2d]{N})$). Kořen m volíme stejně jako

u původní metody. Vypočítáme další dva koeficienty v rozvoji podle m a zkontrolujeme, jestli jsou dostatečně malé. Pokud ano, pokusíme se tento polynom zkosit. Nechť f_m je polynom odpovídající rozvoji podle m. Pro zkosení použijeme operace (i), (ii) a pomocí nich bude chtít redukovat velikost f_m nad upravenou referenční prosívací oblastí. Nechť

$$\begin{aligned} f(x) &= f_m(x_t) + (c_1 x_t + c_0)(x_t - m_t), \\ F(x,y) &= y^d f(x/y), \end{aligned}$$

kde $|x| < \sqrt{s}$, $|y| < 1/\sqrt{s}$. c_1, c_0, t a s bereme jako reálné proměnné. Snažíme se tedy minimalizovat integrál

$$\int_{-\sqrt{s}}^{\sqrt{s}} \int_{-1/\sqrt{s}}^{1/\sqrt{s}} F^2(x,y)dxdy$$

vzhledem k těmto proměnným. Při aplikaci některého z minimalizačních algoritmů (např. method of steepest descent) je nutné brát v úvahu velikost koeficientů polynomu $F^2(x,y)$, která je přibližně $O(\sqrt[d/2]{N})$. Když nalezneme optimální hodnoty c_1, c_0 a t, zaokrouhlíme je na celá čísla a znovu přepočítáme ideální s. Pro rychlý odhad velikosti funkčních hodnot použijeme

$$I(F,s) = \log\left(\sqrt{\int_{-\sqrt{s}}^{\sqrt{s}} \int_{-1/\sqrt{s}}^{1/\sqrt{s}} F^2(x,y)dxdy}\right).$$

Pokud je spočítaný odhad dostatečně malý, postupujeme k vylepšení kořenových vlastností. Pro tento krok budeme uvažovat polynomy tvaru

$$f_{j_1,j_0}(x) = f(x) + (j_1 x - j_0)(x - m),$$

kde $|j_0| < J_0$ a $|j_1| < J_1$ jsou celá čísla, $J_1 \ll J_0$. Metodou, která je velmi podobná prosívání, určíme, pro které hodnoty j_0, j_1 má polynom $f_{j_1,j_0}(x)$ nejlepší kořenové vlastnosti. Postupujeme následovně. Nejprve pevně zvolíme $j_1 \in J_1$. Pro každé uvažované malé prvočíslo p určíme největší mocninu $p^k < B_J$, pro nějakou pevně zvolenou mez B_J. Pro každé $l \in \mathbb{Z}$, $0 \leq l < p^k$, nalezneme kořeny polynomu $f_{j_1,j_0}(l)$ s neznámou j_0 modulo p^k. Pro každý kořen vypočítáme p-valuaci $f_{j_1,j_0}(l)$ a uložíme ji do pole indexovaného pomocí hodnot j_0. Po projití všech čísel l určíme podle vzorce 3.5 aproximaci $\mathrm{cont}_p(f_{j_1,j_0})$ pro všechna $j_0 \in J_0$ (sečtené hodnoty p-valuací máme uloženy v poli indexovaném pomocí j_0). Takto odhadneme $\mathrm{cont}_p(f_{j_1,j_0})$ pro všechna použitá prvočísla. Po započítání projektivních kořenů vypočítáme hodnoty $\alpha(F_{j_1,j_0})$ pro všechna $j_0 \in J_0$. Potom můžeme přejít na další j_1. Až vyčerpáme všechny možnosti, nalezneme dvojici j_0, j_1 s nejmenším α. Protože $I(F,s) \approx I(F_{j_1,j_0}, s)$, máme již vypočítaný odhad velikosti a jako přibližné ohodnocení polynomu můžeme použít

$$I(F_{j_1,j_0}, s) + \alpha(F_{j_1,j_0}).$$

Pokud je toto ohodnocení dostatečně nízké, pak vypočítáme celý polynom

$$f_1(x) = f_{j_1,j_0}(x) = f(x) + (j_1 x - j_0)(x - m).$$

Znovu přepočítáme optimální s a uložíme nalezený pár f_1, f_2.

Při závěrečném ohodnocování těchto polynomů musíme brát v úvahu zkosenou prosívací oblast. Již tedy nezkoumáme hodnoty na půlkružnici, ale na půlelipse, která má poměr mezi hlavní poloosou a vedlejší poloosou s. Dále musíme započítat oba polynomy, protože rozdíly mezi lineárními polynomy mohou být výrazné. Označme $s_1 = \sqrt{s}$, $s_2 = 1/\sqrt{s}$ a položme

$$u_{F_i}(\theta_j) = \frac{\log(|F_i(s_1 \cos(\theta_j), s_2 \sin(\theta_j))|) + \alpha(F_i)}{\log B_i},$$

kde θ_j opět volíme jako střed j-tého podintervalu $[0, \pi]$, $j = 1, \ldots, K$. Celkové ohodnocení polynomů F_1, F_2 je

$$\mathbb{E}(F_1, F_2) = \sum_{j=1}^{K} \rho(u_{F_1}(\theta_j))\rho(u_{F_2}(\theta_j)).$$

Pár s nejnižším ohodnocením je nejlepší. Podobně jako v předchozím případě můžeme vybrat několik nejlepších párů a zjistit jejich skutečnou výtěžnost.

Další vylepšení tohoto postupu pochází od Thorstena Kleinjunga. V [15] uvádí způsob, jak generovat polynomy tak, aby lineární polynom byl nemonický. Tato metoda poskytuje prozatím nejlepší polynomy pro prosívání. Hlavní myšlenka je následující. Nechť $f_2(x) = cx - m$, $\gcd(m, c) = 1$. Pak chceme nalézt polynom $f_1(x) = \sum_{j=0}^{d} c_j x^j$, aby platilo

$$f\left(\frac{m}{c}\right) c^d = N$$

a koeficienty $f_1(x)$ byly co nejmenší. Předpokládejme, že máme určený vedoucí koeficient polynomu $f_1(x)$. Pokud neplatí kongruence

$$c_d m^d \equiv N \mod c, \tag{3.8}$$

pak takový polynom $f_1(x)$ neexistuje (důkaz viz [15]). Vyjdeme tedy z kongruence 3.8 a pro pevné c_d a c nalezneme vhodné m. Potom postupujeme obdobně jako výše. Určíme další koeficienty v rozvoji podle m a zkontrolujeme jejich velikost. K dispozici máme mnohem více možností, protože můžeme volit různá c. Tento postup je ale časově náročnější, proto Kleinjung dále uvádí, jak efektivně zajistit, aby více koeficientů v rozvoji bylo dostatečně malých a tím se vyhnout složitému počítání. Také popisuje jak vhodně volit c.

3.2.3 Montgomeryho metoda kvadratických polynomů

Přestože se převážně používají metody typu $(n, 1)$, P. L. Montgomery přišel s algoritmem hledajícím dva kvadratické polynomy. Tyto polynomy mají koeficienty velikosti $O(\sqrt[4]{N})$, proto je výsledná rychlost algoritmu porovnatelná s případem, kdy k hledání polynomů použijemem metodu typu (3,1). Z tohoto také vyplývá vhodné nasazení Montgomeryho metody pro čísla s přibližně 100 až 110 decimálními ciframi. Podle měření z [9] je číselné síto s polynomy generovanými Montgomeryho metodou dokonce rychlejší než síto používající (3,1) metodu pro generování polynomů.

Pokud ztotožníme kvadratické polynomy s vektory $\mathbf{a}, \mathbf{b} \in \mathbb{Z}^3$, pak podmínka 3.3 odpovídá tomu, že vektory $\mathbf{a}, \mathbf{b}$ jsou kolmé na vektor $(1, m, m^2)$ v $\mathbb{Z}_N^3$. Předpokládejme, že $\mathbf{a} \neq k\mathbf{b}$, $k \in \mathbb{Z}$. Pak vektor $(1, m, m^2)$ generuje ortogonální prostor k prostoru s bází $\{\mathbf{a}, \mathbf{b}\}$, to zřejmě platí i v $\mathbb{Z}_N^3$. Tedy vektorový součin $\mathbf{c} = \mathbf{a} \times \mathbf{b}$ musí být násobkem vektoru $(1, m, m^2)$ v $\mathbb{Z}_N^3$. Nechť $\mathbf{c} = (c_0, c_1, c_2)$. Pak čísla c_0, c_1, c_2 tvoří geometrickou posloupnost modulo N. Pokud by tvořili geometrickou posloupnost i v $\mathbb{Z}$, pak by odpovídající polynomy nebyly ireducibilní (oba by byly dělitelné polynomem $x - m$).

Montgomeryho metoda spočívá ve vytvoření geometrické posloupnosti modulo N (ne v $\mathbb{Z}$), ze které se následně získají vektory $\mathbf{a}, \mathbf{b}$. Abychom sestrojili vektor $\mathbf{c}$, postupujeme následovně. Pro prvočíslo $p < \sqrt{N}$, pro které je N kvadratické reziduum, nalezneme c_1 tak, aby $c_1^2 \equiv N \bmod p$. Přitom chceme $|c_1 - \sqrt{N}| \leq p/2$. Dostáváme geometrickou posloupnost p, c_1, $c_2 = (c_1^2 - n)/p$ v $\mathbb{Z}_N$ s kvocientem $m = c_1/p$. Navíc všechny prvky mají velikost $O(\sqrt{N})$. Dále nechť $s \equiv c_1^{-1} \bmod p$. Pak definujeme

$$\bar{\mathbf{a}}^T = \begin{pmatrix} c_1 \\ -p \\ 0 \end{pmatrix} \quad a \quad \bar{\mathbf{b}}^T = \begin{pmatrix} (c_1(sc_2 \bmod p) - c_2)/p \\ -(sc_2 \bmod p) \\ 1 \end{pmatrix}.$$

Tyto vektory jsou voleny tak, aby byly ortogonální k vektoru $\mathbf{c} = (p, c_1, c_2)$ a platilo $\bar{\mathbf{a}} \neq k\bar{\mathbf{b}}$, $k \in \mathbb{Z}$. Pomocí Gaussova algoritmu ([5] algoritmus 1.3.14) nalezneme redukovanou bázi $\{\mathbf{a}, \mathbf{b}\}$ prostoru generovaného vektory $\bar{\mathbf{a}}, \bar{\mathbf{b}}$. O těchto zredukovaných vektorech je možné dokázat, že $\|\mathbf{a}\|\|\mathbf{b}\| = O(\sqrt{N})$. V praxi se ukazuje, že rozložení je rovnoměrné. Tedy $\|\mathbf{a}\|$ i $\|\mathbf{b}\|$ odpovídají $O(\sqrt[4]{N})$. Výsledné kvadratické polynomy jsou pro různá prvočísla p různé. Po vygenerání dostatečného počtu kandidátů nalezneme nejlepší pár podle výše uvedených kritérií. Můžeme k tomu využít zmíněný způsob ohodnocování polynomů. Navíc je možné požadovat, aby kvadratické polynomy měly reálné kořeny.

V práci [9] je popsáno vylepšení, které vychází z toho, že se bude prosívat pouze pro $a \in I_a$ s pevným $b = 1$. Abychom nalezli dostatečné množství relací, musí být interval $I_a = [-M, M]$ obrovský (M je v řádech miliónů). Protože ale koeficienty vzniklých polynomů f_i mají přibližně stejnou velikost, nebudou funkčních hodnot na intervalu I_a optimální. Tento problém se objevuje i v kvadratickém sítu, kde je také potřeba kvadratické polynomy

správně centrovat. Proto se snažíme dosáhnout toho, aby $c_{i\,2} = O(\sqrt[4]{N}/M)$, $c_{i\,1} = O(\sqrt[4]{N})$ a $c_{i\,0} = O(M\sqrt[4]{N})$, kde $f_i = \sum_{j=0}^{2} c_{i\,j}x^j$. Volíme tedy prvočíslo $p = O(\sqrt{N}/M)$. Pak $c_1 = O(\sqrt{N})$ a $c_2 = O(M\sqrt{N})$. Tím zaručíme, že koeficienty vektorů $\bar{\mathbf{a}}, \bar{\mathbf{b}}$ mají požadovaný poměr. Kdybychom ale začali vektory $\bar{\mathbf{a}} = (\bar{a}_0, \bar{a}_1, \bar{a}_2), \bar{\mathbf{b}} = (\bar{b}_0, \bar{b}_1, \bar{b}_2)$ redukovat, poměr se poruší. Proto zredukujeme vektory $(\bar{a}_0, M\bar{a}_1, M^2\bar{a}_2)$, $(\bar{b}_0, M\bar{b}_1, M^2\bar{b}_2)$. Po zredukování dostaneme vektory (a_0, Ma_1, M^2a_2), (b_0, Mb_1, M^2b_2) s koeficienty o velikosti $O(\sqrt[4]{N}/M)$, ze kterých získáme vektory $\mathbf{a}$, $\mathbf{b}$ s požadovanými vlastnostmi. Dále také dochází ke změně kvocientu, který je nyní $c_1/(pM)$.

Pokusy o nalezení metody, která by generovala polynomy vyšších stupňů, zatím selhávájí. Ukazuje se, že nalezení malé geometrické posloupnosti modulo N je výpočetně velmi náročné. Existují návrhy algoritmu pro typ (3,3) od Montgomeryho, které požadují nalezení pětiprvkové geometrické posloupnosti s jejíž pomocí lze kubické polynomy sestavit, ale náročnost algoritmu je $O(\sqrt[3]{N})$.

S příchodem Montgomeryho metody se začalo uvažovat o použití více polynomů. Kvadratické polynomy generované touto metodou mají dobré vlastnosti a i jejich jednoduché lineární kombinace, které si zachovávají vlastnost 3.3, jsou stále ještě dobré. Z [10] vyplývá, že zrychlení je zaznamenatelné při použití čtyř nebo tří polynomů oproti dvoum, ale pouze za předpokladu, že se prosívá klasickým způsobem. Pokud se použije vylepšení popsané výše, pak se plně projeví ne zcela dobré vlastnosti nakombinovaných polynomů a výsledný čas je horší. Přesné vysvětlení toho jevu neznáme, ale předpokládá se, že rozdíl v kvalitě nakombinovaných polynomů oproti původnímu páru je natolik významý, že způsobí nedostatečnou výtěžnost těchto polynomů. Přitom s každým dalším polynomem musíme najít o tolik relací více, kolik prvků má odpovídající faktorizační báze. Ukazuje se tedy, že použití dvou kvadratických polynomů a jednoho velkého prosívacího intervalu je nejlepší varianta pro faktorizaci čísel v uvažovaném rozmezí. Pro větší čísla se dá předpokládat, že nemá opodstatnění používat více polynomů různých stupňů a ani neexistují žádné práce, které by se tímto zabývaly. A protože zatím neumíme generovat efektivně dobré polynomy vyšších stupňů, natožpak více dobrých polynomů sdílejících stejný kořen modulo N, zůstává MPGNFS pouze jako teoretická možnost.

3.3 Prosívání

Prosívací fáze je časově nejnáročnější částí algoritmu. I přestože není algoritmicky náročná, její výkonnostně orientovaná implementace je esenciální. Než vysvětlíme princip prosívání a popíšeme různé modifikace, zaměříme se na sestavení faktorizační báze. Budeme se držet označení zavedenými v části 3.1. Prosívání popíšeme pro jedno těleso, protože až na závěrečnou fázi se jedná o nezávislé procesy. Pro zjednodušení zápisu budeme dvojici $(a, b) \in \mathbb{Z}^2$ ztotožňovat buď s hlavním ideálem $(a + b\vartheta)$, nebo s odpovídající relací.

Z teoretické části víme, že k rozkladu hlavního ideálu $(a+b\vartheta)$, $\gcd(a,b)=1$, jsou potřeba pouze prvoideály stupně setrvačnosti 1 a prvoideály nad speciálními prvočísly. Faktorizační bázi číselného tělesa K generovaného monických polynomem $f(x)$ (v případě nemonického polynomu použijeme $\widehat{f}(x)$) sestavíme proto následovně. Procházíme všechny prvočísla menší než předem zvolená mez B. Pro každé nespeciální prvočíslo p nalezneme kořeny $r_1,\ldots,r_k$ polynomu $f(x)$ modulo p. Z teoretické části víme, že prvoideály nad p v K stupně setrvačnosti 1 jsou $(p,\vartheta-r_1),\ldots,(p,\vartheta-r_k)$. Každý takovýto prvoideál je tedy jednoznačně určen dvojicí (p,r), $0\leq r<p$, pro které platí

$$f(r)\equiv 0 \bmod p.$$

Pokud je p speciální prvočíslo a pokud chceme použít prvoideály nad speciálními prvočísly k prosívání, musíme použít komplikovanější způsob popsaný v části 2.3 k nalezení prvoideálů nad p. Jak uvidíme dále, se speciálními prvoideály se přímo neprosívá, budeme je ale potřebovat k rozkladu hlavního ideálu $(a+b\vartheta)$.

Rozklad hlavního ideálu $(a+b\vartheta)$, kde $\gcd(a,b)=1$, určíme podle normy, která je rovna $F(a,-b)$. Pokud nespeciální prvočíslo p dělí $F(a,-b)$, pak prvoideál odpovídající dvojici $(p, ab^{-1} \bmod p)$ dělí ideál $(a+b\vartheta)$. Protože tyto prvoideály jsou stupně setrvačnosti 1, odpovídají si i valuace. Pokud je $F(a,-b)$ dělitelné i speciálním prvočíslem, musíme použít některý z obecných algoritmů pro výpočet valuace prvoideálu ([5] algoritmus 4.8.17). Přestože jsou tyto výpočty mnohem náročnější, doporučuje se prvoideály nad speciálními prvočísly používat.

Jak bude patrné z popisu níže, prosívání je algoritmicky velmi jednoduché, a proto se uvažuje i o hardwarových implementacích. Nejznámější je TWIRL ([31]). Zatím se jedná pouze o teoretický návrh, ale s realistickým designem. Odhaduje se, že pomocí těchto zařízení je možné faktorizovat 1024 bitové číslo za rok při nákladu kolem 10 miliónů dolarů.

Než popíšeme prosívání uveďme jak nenáročně vygenerovat velké množství hladkých relací. Pokud se polynom f rozkládá na součin lineárních polynomů modulo prvočíslo p, pak všechny prvoideály v K ležící nad prvoideálem (p) jsou stupně setrvačnosti 1 a tedy jsou obsaženy ve faktorizační bázi. Pokud se tak stane pro oba polynomy současně, pak dostáváme hladkou relaci $(p,0)$. Těmto relacím říkáme *volné relace*. Nechť g je řád Galoisovy grupy tělesa generovaného polynomem f_1f_2. Pak pro přibližně $1/g$ prvočísel menších než mez B budou oba polynomy mít pouze lineární dělitele modulo tyto prvočísla. Tedy čím větší stupně polynomů používáme, tím méně volných relací nalezneme.

3.3.1 Klasické prosívání

Našim cílem je faktorizovat čísla $F(a,-b)$ pro dvojice $(a,b)\in I_a\times I_b$. Intervaly $I_a, I_b\subset\mathbb{Z}$ volíme tak, aby funkční hodnota $F(x,y)$ byla na vzniklé oblasti

co nejmenší. Přitom ale musí být oblast dostatečně velká, abychom nalezli dostatek relací.

Nyní popíšeme samotné prosívání. Zvolme například b pevně, $b = b_0$. Prosívání znamená, že pro každé prvočíslo $p < B$, označíme všechny hodnoty a, pro které je $F(a, -b_0)$ dělitelné prvočíslem p. Jak tyto hodnoty a nalezneme? Pro prvočíslo p označme $s_1, \ldots, s_l$ kořeny polynomu $F(x, -b_0) \bmod p$. Dostáváme

$$F(s_i + kp, b_0) \equiv 0 \bmod p,\ k \in \mathbb{Z},\ 0 \leq i < l.$$

Stačí tedy nalézt kořeny modulo p a poté označit všechny hodnoty odpovídající $s_i + kp$. Když projdeme všechny prvočísla menší než B, změníme hodnotu b. V implementacích se nejčastěji setkáme s volbou $I_a = [-M, M]$, $I_b = [1, L]$ a s prosíváním podle a (tak jak jsme popsali). V prvním kroku máme $b = 1$ a tedy hledané kořeny s_i jsou rovny r_i uvedeným výše. To znamená, že prvočíslu p s kořenem s_i odpovídá prvoideál (p, r_i) z faktorizační báze. Můžeme proto procházet všechny prvoideály z faktorizační báze a jako kořen použít příslušné r. Při změně hodnoty b na $b+1$ není těžké vypočítat odpovídající změnu kořenů a novou hodnotu uložit. Závislost je pouze lineární.

Pro každou hodnotu a z uvažovaného intervalu ukládáme prvočísla, která dělí $F(a, b_j)$. Většina implementací přičítá do odpovídajícího políčka pole reprezentujícího hodnoty a logaritmus těchto prvočísel (zaokrouhlený na celé číslo). Základ logaritmu je volen podle velikosti políčka a maximálních hodnot $F(x, y)$.

Po prosetí všech provideálů pro pevně zvolené b musíme nalézt čísla a, pro která je $F(a, -b)$ B-hladké. Provádíme to tak, že procházíme pole reprezentující hodnoty a a kontrolujeme, jestli nasčítaná hodnota logaritmů v políčku odpovídá logaritmu $F(a, b_j)$. Protože by bylo časově náročné pro každé políčko počítat příslušnou hodnotu logaritmu, vypočítá se pouze mez F, které musíme minimálně dosáhnout. Správně spočítat mez F je velmi důležité. Musí se uvažovat zaokrouhlovací chyby při počítání s logaritmy a také to, že při prosívání neurčujeme v jaké mocnině prvočíslo p dělí $F(a, -b)$. Přitom špatně zvolená mez F může znamenat, že neodhalíme spoustu hladkých relací. Naopak pokud budeme prohlašovat velké množství relací za hladké, budeme následně ztrácet čas rozkladem relativně velkého čísla $F(a, -b)$, které nakonec nemusí být hladké. Mez F se tedy blíží průměrné hodnotě funkce $F(x, -b)$ na intervalu I_a. Možný způsob volby meze F popisujeme v závěrečné kapitole. Podobně jako u kvadratického síta provede chýtrý trik pro zrychlení hledání vhodných kandidátů. Prosívací pole inicializuje číslem $-\log F$ a po ukončení prosívání hledáme políčka s kladnou hodnotou (tzv. kandidáti na faktorizaci). Protože ve většině implementací jedno políčko odpovídá jednomu bytu, můžeme zkontrolovat více polí najednou přetypováním na nativní velikost bitového slova (dnes 32 nebo 64 bitů).

Prosívání provedeme v obou číselných tělesech. Pak hledáme dvojice (a, b), které jsou v obou tělesech kandidátem na faktorizaci. Pro hledání vhodných

kandidátů se doporučuje nejprve procházet prosívací interval patřící číselnému tělesu s horším polynomem, protože bude více restriktivní. Pokud narazíme na dvojici (a, b), která je kandidátem v obou použitých číselných tělesech, pokusíme se číslo $F(a, -b)$ faktorizovat. Nejjednodušší způsob faktorizace je projít všechny prvočísla menší než B a zkusit jimi $F(a, -b)$ vydělit. K určení prvoideálové faktorizace $(a + b\vartheta)$ postupujeme podle popisu výše. Zkoušet dělitelnost $F(a, -b)$ všemi prvočísly je neefektivné. Lepší způsob je projít všechny prvoideály (p, r). Pro každý znovu zkusit, zda a odpovídá kořenu funkce $F(x, -b) \bmod p$ a pouze pokud ano, začít číslo $F(a, -b)$ dělit prvočíslem p. Pokud je $F(a, -b)$ B-hladké a v druhém čísleném tělese je také odpovídající norma hladká, nalezli jsme hladkou relaci.

Podobně jako u kvadratického síta není výskyt hladkých relací příliš velký. Abychom algoritmus urychlili používáme i částečně hladké relace, ze kterých se hladké relace dají sestavit (více v další části). Při použití částečných relací upravíme mez tak, aby zohlednila jejich používání. To znamená, že budeme nacházet mnohem více kandidátů pro následnou faktorizaci. Pokud po vyzkoušení všech prvoideálů není zbytek roven 1, nejedná se tedy o hladkou relaci, můžeme se pokusit zbytek rozložit pomocí nějakého jednoduššího faktorizačního algoritmu. Mezi používané algoritmy patří Pollardova $p - 1$ metoda (pro menší faktorizace), metoda eliptických křivek ([18]) nebo Shanksův SQUFOF ([13]). Pokud je zbytek menší než B^2 jedná se zřejmě o prvočíslo, proto použití maximálně 1,1-částečných relací prosívání příliš nezpomaluje. Máme jen více kandidátů. Obvykle se jako mez pro velké provideály volí číslo mnohem menší než B^2. Při použití až 2,2-částečně hladkých relací již faktorizace může prosívání výrazně zpomalit. Je potřeba vhodně volit použitou faktorizační metodu. Avšak jak popíšeme v následující části, stále se vyplácí 2,2-částečně hladké relace hledat. Požadujeme ale, aby oba velké prvoideály byly přibližně stejně velké. Dostáváme tedy další dvě meze. B_V pro maximální velikost velkého prvoideálu a B_M pro maximální velikost zbytku ($B_M \approx B_V^2$).

Protože prosívání je zdaleka časově nejnáročnější částí algoritmu je důležité jednotlivé kroky efektivně naprogramovat. Většina operací během prosívání jsou pouze přístupy do pole a provedení několika jednoduchých výpočtů s jeho prvky. Prosívací pole je ale většinou velmi velké, a proto ho nelze celé uložit do datové cache počítače (viz [32]). To výrazně zpomaluje práci s polem, neboť dochází k neustálému dočítání pole do cache, protože při prosívání s jedním prvočíslem procházíme celé pole. Řešením je rozdělit prosívací interval na části tak, aby se pohodlně vešly do datové cache. Prosíváme potom všemi prvoideály jen nad jednou částí. Tato změna vyžaduje přepočítávat skoky prvočísel při přechodu na další část, ale jejich náročnost je minimální.

Pokud je prosívací polynom nemonický, musíme provést několik úprav. Předně se snažíme určit faktorizaci hlavního ideálu $(c_d a + b\vartheta)$, $\gcd(a, b) = 1$. Pro jehož normu platí

$$N((c_d a + b\vartheta)) = \widehat{F}(c_d a, -b) = F(a, -b) c_d^{d-1}.$$

Stále tedy rozkládáme hodnoty $F(x, y)$, ze kterých pro prvočísla nesoudělná s vedoucím koeficientem c_d získáme provideálovou faktorizaci. S prvoideály nad prvočísly, které dělí vedoucí koeficient polynomu f, zacházíme odlišně, protože pro tyto prvočísla se zřejmě valuace normy hlavního ideálu $(c_d a + b\widehat{\vartheta})$ líší od valuace $F(a, -b)$. Dále pokud prvočíslo p dělí vedoucí koeficient, a zároveň i b, pak dostáváme

$$(c_d a + b\widehat{\vartheta}) = (p)^e (c'a + b'\widehat{\vartheta}),\ e > 0.$$

Musíme tedy do faktorizační báze přidat hlavní ideály (p), kde p je prvočíslo dělící vedoucí koeficient. Dále protože prvoideály jsou určeny polynomem $\widehat{f}$, musíme před prvním krokem přepočítat vhodně pomocné prosívací kořeny.

Prosívání je velmi dobře paralelizovatelné. Jednotlivé výpočetní stanice mohou dostat jen část intervalu I_b a provádět prosívání pouze pro b z tohoto intervalu. Další možností je rozdělit i interval I_a. Vzájemná komunikace mezi uzly nemusí být častá, některé implementace dokonce využívají emailovou komunikaci k posílání výsledků. Pomocí masivní paralelizaci je již v dnešní době teoreticky možné faktorizovat 1024 bitové číslo. Podle odhadů v [14] je ale zapotřebí kolem 12 miliónů počítačů a rok prosívání.

3.3.2 Mřížové prosívání

Významné vylepšení prosívací metody pochází od J. M. Pollarda [29], dále zpracované v [12]. Hlavní myšlenka spočívá v použití mříže k předvýběru dvojic (a, b), které budou s větší pravděpodobností hladké.

Označme FB_i faktorizační bázi příslušnou tělesu K_i, $i = 1, 2$. Nad faktorizační bází zavedeme uspořádání následovně

$$(p, r) < (q, s) \Leftrightarrow (p < q) \vee (p = q \wedge r < s), (p, r), (q, s) \in FB_i.$$

Nechť $P = (p, r) \in FB_i$. Pak L_P značí mříž generovanou vektory $(p, 0)$, $(r, 1)$ v $\mathbb{Z}^2$, tedy $L_P = \{c(p, 0) + e(r, 1) : c, e \in \mathbb{Z}\}$. Pro každý bod $(a, b) \in L_P$, kde $P \in FB_j$, platí, že $P|(a + b\vartheta_j)$. Dále nechť $Q \in FB_i$. Označme L_{PQ} podmříž mříže L_P definovanou jako $L_{PQ} = L_P \cap L_Q$.

Nechť K_1 je číselné těleso s horším polynomem. Faktorizační bázi FB_1 rozdělíme na množiny S a M. Množina S obsahuje prvoideály nad prvočíslem menším než mez B_S a množina M zbývající prvoideály. Podíl B/B_S volíme přibližně 0.1 až 0.5. Postupně procházíme prvoideály z množiny M a pro každý prvoideál $P = (p, r) \in M$ sestavíme mříž L_P. Pro tuto mříž nalezneme redukovanou bázi $V_1 = (v_1, w_1)$, $V_2 = (v_2, w_2)$. Našim cílem je prosívat přes dvojice $(c, e) \in I_c \times I_e$ reprezentující body $(a, b) = (cv_1 + ev_2, cw_1 + ew_2)$, které mají normu v číselném tělese K_1 dělitelnou prvočíslem p. K prosívání použijeme podmříže. Procházíme tedy všechny prvoideály $Q = (q, s)$ menší než P a sestavujeme podmříže L_{PQ}. Podmříž L_{PQ} je zřejmě generovaná vektory

$$U_1' = (q, 0),\ U_2' = \left(\frac{v_2 - sw_2}{sw_1 - v_1} \bmod q, 1\right).$$

V případě, že vektor U_2' není dobře definovaný nebo $U_2' = (0, 1)$, použijeme prosívání po řádcích. Tedy klasické prosívaní popsané výše (prosíváme ale přes dvojice (c, e)). Jinak vypočítáme redukovanou bázi U_1, U_2 a prosíváme pomocí všech lineárních kombinací (c, e) bázových vektorů, pro které $(c, e) \in I_c \times I_e$.

Prosívání obvykle neprobíhá přes celou oblast $I_c \times I_e$. Místo toho interval I_e rozdělíme na menší bloky. A opět jako u klasického prosívání můžeme z důvodu optimalizace přístupu do cache rozdělit i interval I_c. Když prosejeme všemi prvoideály $Q \in FB_1$ menšími než prvoideál P jednu část z $I_c \times I_e$. Začneme tuto část prosívat v druhém číselném tělese. Zde již pomocí podmříže prosíváme se všemi prvoideály z faktorizační báze.

Po prosetí se všemi uvažovanými prvoideály podobně jako v klasickém prosívání hledáme dvojice (c, e), které jsou vhodnými kandidáty na faktorizaci. Když nějakou dvojici nalezneme, zkusíme odpovídající normy faktorizovat. Opět postupujeme stejně jako u klasického prosívání. V práci [12] autoři navrhují zjednodušit faktorizaci dalším prosíváním, při kterém se zaznamenávají jednotlivá prvočísla (nebo přímo prvoideály), které normy dělí. Rychlost tohoto postupu závisí na vhodném hashování dvojic (c, e), které se mají faktorizovat. Jen pro tyto dvojice ukládáme výsledky prosívání. Přitom není nutné prosívat se všemi prvočísly, pro zbylé se použije triviální dělení. Pokud hledáme i částečně hladké relace a zbytek normy po dělení všemi prvočísly není 1, použijeme nějakou ze zmíněných faktorizačních metod k rozkladu zbytku.

Mřížové prosívání nenalezne všechny relace, které bychom mohli odhalit klasickým prosíváním, protože vždy požaduje, aby relace byla v prvním číselném tělese dělitelná alespoň jedním prvoideálem z množiny M. To ovšem při dobré implementaci nepředstavuje problém, jak je uvedeno v [29], protože pro nalezení většiny hladkých relací provádíme pouze zlomek práce, kterou vyžaduje klasické prosívání. Celkově je tedy mřížové prosívání mnohem rychlejší než klasické. U mřížového prosívání s použitím částečných relací může dojít k duplicitě relací, proto je potřeba v následující fázi provést pročištění výsledků. K duplicitě relací z principu nemůže dojít u klasického prosívání.

Paralelizace u mřížového prosívání je stejně snadná jako u klasického. Dokonce máme více možností, protože můžeme stanicím přiřazovat i jednotlivé prvoideály P určující prvotní mříž. To navíc znamená, že stanice nemusí znát celou faktorizační bázi FB_1, ale jen její část.

3.4 Zpracování relací

Fáze na zpracování relací má převážně technický ráz. Hlavním úkolem je z nalezených relací sestavit matici $\mathbf{B}$ nad $\mathbb{Z}_2$. Při sestavovaní matice bude naší prioritou zajistit, aby byla co nejmenší (počet řádků a sloupců), a také aby byla

řídká. Na začátku máme k dispozici velké množství hladkých relací, případně i částečně hladkých relací.

3.4.1 Zpracování částečně hladkých relací

Nejprve se budeme zabývat částečně hladkými relacemi. Mezi první práce zabývajícími se využitím více částečně hladkých relací patří [19]. Důvod proč hledáme i částečně hladké relace je ten, že z nich můžeme sestavit relace hladké. Pokud nalezneme několik částečně hladkých relací tak, aby po jejich vynásobení byly všechny velké prvoideály v sudé mocnině, dostáváme hladkou relaci, se kterou můžeme pracovat jako s ostatními hladkými relacemi. Pokud bychom používali pouze hladké relace, brzy bychom se dostali do podobné situace jako v případě kvadratického síta. Frekvence výskytu hladkých relací by postupně klesala, což by vedlo k velké časové náročnosti jejich hledání. Jak jsme uvedli v prosívací části, určení až 2,2-částečně hladkých relací není příliš časově náročné. To je další rozdíl mezi číselným a kvadratickým sítem, kde odpovídající použití až 4-částečně hladkých relací je výpočetně a tedy i časově velmi náročné a může být kontraproduktivní. Dále podle experimentů v [8] vyplývá, že u větších faktorizacích, kde byly použity i 2,2-částečně hladké relace, dochází po nalezení velkého množství částečných relací přímo k explozi hladkých relací, které se z nich dají nakombinovat. Tento jev nebyl doposud teoreticky vysvětlen, ale je úspěšně využíván u všech současných rekordních faktorizací. Ovšem pokud použijeme maximálně 1,1-částečně hladké relace, nemůžeme očekávat stejnou výnosnost z kombinování částečných relací jako u kvadratického síta, kde bychom použili 2-částečně hladké relace. To je způsobeno tím, že v číselném sítu musíme rozlišovat mezi prvoideály patřícími do různých číselných těles.

První krok při zpracování částečných relací je odstranit všechny duplicity, které mohly vzniknout při prosívání (např. při znovuspuštění nebo při mřížkovém prosívání). Protože každá relace je jednoznačně určena čísly a, b, můžeme pomocí vhodně zkonstruované hash tabulky nalézt duplicity. Jako příklad elegantní hashovací funkce uvedeme funkci používanou v [4]. Definujme $\Pi = \lfloor \pi \cdot 10^7 \rfloor$ a $E = \lfloor e \cdot 10^7 \rfloor$. Zřejmě $\gcd(\Pi, E) = 1$. Jako hashovací funkci volíme

$$h(a, b) = a \cdot \Pi + b \cdot E \bmod 2^{64}.$$

Druhým krokem je nalézt a odstranit všechny částečně hladké relace, které obsahují alespoň jeden velký prvoideál v liché mocnině, který se nenachází v žádné jiné částečné relaci v liché mocnině. Takovým to relacím říkáme *singletony*. Singletony jsou pro nás zřejmě bezvýznamné, neboť výsledná kombinace relací nikdy nemůže obsahovat příslušný velký prvoideál v sudé mocnině. Nejjednodušším způsobem jak singletony nalézt, je projít všechny velké prvoideály ve všech částečných relacích a do hash tabulky ukládat jejich výskyt. Při dalším projití částečných relací vymažeme ty, které obsahují velký prvoideál,

který jsme nalezli pouze jednou. Tento postup musíme opakovat tak dlouho, dokud již žádnou částečnou relaci neodstraníme. Těmto částečným relacím říkáme *užitečné částečné relace*.

Nyní chceme z nalezených užitečných relací sestavit relace hladké. Nejjednodušší situaci máme, pokud používáme maximálně 1,1-částečně hladké relace. V tomto případě můžeme podobně jako v kvadratickém sítu použít grafovou reprezentaci částečných relací. Nejprve do grafu přidáme vrchol 1. Dále každý prvoideál v částečné relaci představuje vrchol grafu, přitom musíme odlišit prvoideály z ruzných číselných těles. Tento vrchol buď spojíme hranou s vrcholem 1, pokud se jedná o 1-částečnou relaci, nebo mezi s sebou spojíme dva vrcholy z 1,1-částečně hladké relace. Nezávislé kružnice pak zřejmě představují hladké relace. Z teorie grafů víme, že počet nezávislých kružnic (cyklomatické číslo grafu) snadno spočítáme podle vzorce

$$C(G) = e + v - c,$$

kde e je počet hran, v je počet vrcholů a c je počet komponent grafu G. Z vhodně zvolené datové reprezentace grafu pak snadno můžeme hladké relace získat.

O něco komplikovanější je situace, kdy používáme i částečné relace s více velkými prvoideály. V současné době neexistuje žádná vhodná reprezentace jako v předešlém případě, která by umožnila snadno zjistit, kolik hladkých relací můžeme získat a pomoci při jejich sestavování. Většina implementací se snaží zkombinovat částečné relace obsahující velký prvoideál v liché mocnině, který se vyskytuje maximálně k-krát (většinou se jako k volí 3, případně až 8). Tímto snížíme počet částečných relací o jednu a také dosáhneme sudých mocnin tohoto prvoideálu u nově vzniklých relací (stále mohou být částečné). Musíme ale dávát pozor na vzrůstající počet prvoideálů v liché mocnině ve vzniklých relacích, protože to může vést k hustší výsledné matici $\mathbf{B}$. Zbývající částečné relace prohlásíme za hladké, tedy přidáme do faktorizačních bází všechny odpovídající velké prvoideály, které se nacházejí v lichých mocninách. To vede k velmi výraznému zvětšení výsledné matice, proto je potřeba co největší množství velkých prvoideálů odstranit zkombinováním. Výsledná matice má po tomto kroku přibližně několik milionů řádků, což je stále ještě upočítatelné množství. Podrobnější popis jak nejvhodněji kombinovat relace, abychom zamezili přílišné hustotě matice, lze nalézt v [4]. Abychom v průběhu prosívání určili kolik hladkých relací můžeme přibližně sestavit, stačí odečíst počet různých velkých prvoideálů od počtu nalezených částečně hladkých relací.

Při použití 2,2-částečně hladkých relací je pro nás výhodnější velké prvoideály přidat do bází a tím zvětšit výslednou matici, než se snažit o jejich zkombinování do hladkých relací. Z práce [8] vyplývá, že počet částečných relací potřebných k sestavení hladké relace těsně před explozí velmi vzroste. Podle jejich měření se počet potřebných relací pohyboval od 500 do 1000. To znamená že velmi mnoho částečných relací potřebujeme k získání mnohem

menšího množství hladkých relací. Sestavovat z těchto částečných relací nějaké další nové hladké relace není algoritmicky snadné. V případě, kdy používáme pouze 1,1-částečné relace, je situace odlišná. Pomocí grafové reprezentace jednoduše hladké relace nalezname a navíc počet částečných relací potřebných na sestavení hladké relace je ve většině případů nejvýše 3.

Pro výpočet výsledné odmocniny budeme potřebovat znát algebraickou reprezentaci každé relace. Původní hladké a částečně hladké relace můžeme reprezentovat polynomem $bx + a$. Pro relaci sestavenou s částečně hladkých relací dostaneme reprezentaci vynásobením polynomů jednotlivých částečných relací modulo polynom f (polynom generující číselné těleso). Tedy pro každé číselné těleso potřebujeme vlastní reprezentaci.

Při použití nemonických polynomů je pro nás důležité vědet, z kolika relací jsme výslednou hladkou, případně částečně hladkou relaci sestavili, protože v odmocninové fázi budeme s každou relací spojovat lomenný ideál $(c_d)^{-1}$, kde c_d je vedoucí koeficient odpovídajícího nemonického polynomu. Dále budeme potřebovat znát algebraickou reprezentaci každé relace, která je v tomto případě $bx + c_d a$. Pro sestavenou relaci pouze k modulování použijeme odpovídající monický polynom $\widehat{f}$.

3.4.2 Zpracování hladkých relací

Zpracování hladkých relací je téměř stejné jako částečně hladký. Nejprve odstraníme duplicity a poté singletony, přitom volíme stejný postup. Opět se můžeme pokusit o zmenšení matice seskupením relací, které v rozkladu obsahují prvoideály, které se vyskytují v rozkladech malého počtu relací. Stále je potřeba hlídat hustotu matice, aby nebyla příliš velká. Pokud máme k dispozici více hladkých relací než potřebujeme, můžeme odstranit ty, které mají příliš mnoho prvoideálů v lichých mocninách. Tím snížíme hustotu matice.

3.4.3 Vytvoření matice

Nyní máme k dispozici pročištěné hladké a částečně hladké relace. Pomocí nich určíme, které prvky z faktorizačních bází jsou v nich použity a pouze ty použijeme na sestavení matice. Pokud jsme tak již neučinili, musíme do odpovídajících faktorizačních bází přidat i všechny velké prvoideály, které se v nějaké částečné relaci vyskytují v liché mocnině.

K vytvoření matice budeme dále potřebovat najít vhodné provideály pro výpočet kvadratických charakterů. Postup popíšeme pro první číslené těleso, u druhého postupujeme stejně. Nejprve zaznamenáme všechny velké prvoideály, které se vyskytují v rozkladu použitých relacích v tomto tělese. Následně začneme postupně zkoušet prvočísla, která jsou větší než mez faktorizační báze a přitom se nevyskytují v žádném rozkladu relací. Pro každé takové prvočíslo zjistíme, jestli není speciální a zda existuje prvoideál nad tímto prvočíslem stupně setrvačnosti 1. K tomu stačí najít alespoň jeden kořen polynomu f,

resp. $\widehat{f}$, modulo uvažované prvočíslo. Doporučuje se použít alespoň 64 charakterů (při rekordních faktorizacích i 256).

Když máme připravené kvadratické charaktery můžeme začít sestavovat matici $\mathbf{B}$. Pro zjednodušení budeme sestavovat matici $\mathbf{B}^T$. Jak ukážeme v lineární části, není podstatné jestli použijeme $\mathbf{B}$, nebo $\mathbf{B}^T$. Procházíme tedy všechny hladké a užitečné částečné relace a sestavujeme řádek matice $\mathbf{B}^T$ následovně. Z popisu algoritmu víme, že sloupce představují prvky z první a druhé báze a kvadratické charaktery. Tedy jak již bylo uvedeno zapisujeme mocniny modulo 2 z rozkladu relace do odpovídajících sloupců. Potom přiřadíme do sloupců s charaktery hodnoty podle předpisu uvedeného v části 3.1.

Při použití nemonických polynomů musíme ještě do posledního sloupce zaznamenat z kolika relací se zpracovávaná relace skládá. Pro původní hladkou relaci je to 1, ale pokud jsme hladkou, případně částečně hladkou, relaci získali zkombinováním více částečně hladkých relací, musíme přiřadit odpovídající číslo.

Způsob uložení a reprezentování matice musíme volit s ohledem na její ohromnou velikost. Při rekordních faktorizacích se velikost matice pochybuje v řádech milionů sloupců.

3.5 Lineární fáze

Lineární fáze číselného síta řeší zcela obecný problém, kdy chceme nalézt řešení soustavy rovnic nad $\mathbb{Z}_2$ s nulovým vektorem pravých stran (řešení homogenní soustavy lineárních rovnic, nulový prostor matice). Lineární fáze je tedy stejná jako v případě kvadratického síta nebo CFRACu. Z předchozí fáze máme matici $\mathbf{B} \in \mathbb{Z}_2^{n \times n}$, která je velmi řídká (výskyt nenulových prvků je minimální) a chceme nalézt řešení rovnice tvaru

$$\mathbf{Bx} = \mathbf{0},$$

kde $\mathbf{0} \in (\mathbb{Z}_2)^n$ značí nulový vektor. Potřebu řešit takovouto soustavu můžeme nalézt v mnoha jiných oborech a existuje mnoho různých algoritmů pro její výpočet. Ale pro moderní faktorizační algoritmy, založené na principu prosívání, se ukazují tyto metody jako nedostatečné, protože při faktorizaci velkých čísel dosahuje matice $\mathbf{B}$ velikostí kolem $10^6 \times 10^6$. Nastávají tedy problémy s reprezentací a s počítáním u těchto obrovských matic. Nadruhou stranu matice ve faktorizačních algoritmech jsou velmi řídké a reprezentují se nejčastěji jako výčet nenulových prvků v jednotlivých řádcích, tímto se ušetří velké množství paměti počítače, neboť při standardní reprezentaci se může jednat o několik gigabytů. Klasické metody řešení jako je například Gaussova eliminační metoda nejsou v tomto případě vhodné již z důvodu, že poruší řídkost matice $\mathbf{B}$, a proto opět můžeme čelit problému s nedostatkem paměti. K řešení se tedy používají speciálně navržené iterační algoritmy, které neporuší řídkost matice,

například Lanczošova bloková metoda, Wiedemannova bloková metoda nebo metoda konjugovaných gradientů. Tyto metody jsou pouze pravděpodobnostní, ale v praxi se ukazuje, že k selhání téměř nedochází. V této části se podrobněji zaměříme na nejpoužívanější metodu, kterou je Lanczošova bloková metoda. Složitost této metody je $O(mn^2)$, kde n je počet řádků matice a m je průměrný počet nenulových prvků v řádku, pro srovnání uveďme, že složitost Gaussovy eliminační metody je $O(n^3)$.

I přes znatelné zrychlení oproti původním metodám, lineární fáze algoritmu může stále být problematická při rekordních faktorizacích, a proto je potřeba hledat nové možnosti, například dokonalejší využití paralelizace. Z tohoto důvodu se do popředí zájmu v posledních letech dostává Wiedemannova bloková metoda, protože ji lze lépe paralelizovat.

3.5.1 Lanczošova bloková metoda

Nejprve uvedeme původní Lanczošův algoritmus na řešení soustavy lineárních rovnic (viz [16]) a postupně jej přizpůsobíme použití v NFS. Lanczošův iterační algoritmus řeší obecný případ, kdy máme

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

kde $\mathbf{A} \in \mathbb{R}^{n\times n}$ je symetrická pozitivně definitní matice a $\mathbf{b} \in \mathbb{R}^n$ libovolný nenulový vektor. V prvním kroku algoritmu volíme

$$\mathbf{w}_0 = \mathbf{b}.$$

Pro i-tý krok máme

$$\mathbf{w}_i = \mathbf{A}\mathbf{w}_{i-1} - \sum_{j=0}^{i-1} c_{i\,j}\mathbf{w}_j,$$

kde

$$c_{i\,j} = \frac{\mathbf{w}_j^T\mathbf{A}^2\mathbf{w}_{i-1}}{\mathbf{w}_j^T\mathbf{A}\mathbf{w}_j}.$$

Protože matice $\mathbf{A}$ je pozitivně definitní, má výraz vpravo smysl. Po konečném počtu iterací dostaneme $\mathbf{w}_m = \mathbf{0}$. Pak řešení rovnice je

$$\mathbf{x} = \sum_{j=0}^{m-1} \frac{\mathbf{w}_j \cdot \mathbf{b}}{\mathbf{w}_j^T\mathbf{A}\mathbf{w}_j}\mathbf{w}_j.$$

Správnost algoritmu dokazuje následující tvrzení.

Věta 3.7. *Lanczošův iterační algoritmus nalezne maximálně po $n \in \mathbb{N}$ krocích řešení maticové rovnice*

$$\mathbf{A}\mathbf{x} = \mathbf{b},$$

kde $\mathbf{A} \in \mathbb{R}^{n\times n}$ *je symetrická pozitivně definitní matice.*

Důkaz. Nejprve ukážeme, že pro vektory $\mathbf{w}_i$ platí

$$\mathbf{w}_i^T \mathbf{A}\mathbf{w}_j = 0, \;\; i \neq j.$$

Řekneme, že jsou $\mathbf{A}$-ortogonální. Budeme postupovat indukcí podle $\max(i,j)$. Pro $\max(i,j) = 0$ to zřejmě platí. Nechť jsou tedy $\mathbf{w}_i$ $\mathbf{A}$-ortogonální pro $\max(i,j) = z - 1$. Dokážeme platnost i pro $\max(i,j) = z$. Nechť $i = z$ a $j < z$. Podle definice $\mathbf{w}_i$ máme

$$\mathbf{w}_i^T \mathbf{A}\mathbf{w}_j = (\mathbf{A}\mathbf{w}_{i-1} - \sum_{k=0}^{i-1} c_{i\,k}\mathbf{w}_k)^T \mathbf{A}\mathbf{w}_j = \mathbf{w}_{i-1}^T \mathbf{A}^T \mathbf{A}\mathbf{w}_j - \sum_{k=0}^{i-1} c_{i\,k}\mathbf{w}_k^T \mathbf{A}\mathbf{w}_j.$$

Z indukčního předpokladu dostáváme $\mathbf{w}_k^T \mathbf{A}\mathbf{w}_j = 0$ pro $k \neq j$. Dále z definice $c_{i\,j}$ a symetrie $\mathbf{A}$ plyne

$$\mathbf{w}_i^T \mathbf{A}\mathbf{w}_j = \mathbf{w}_{i-1}^T \mathbf{A}^T \mathbf{A}\mathbf{w}_j - c_{i\,j}\mathbf{w}_j^T \mathbf{A}\mathbf{w}_j = 0.$$

Protože je $\mathbf{A}$ pozitivně definitní, dostáváme z $\mathbf{A}$-ortogonálnosti vektorů $\mathbf{w}_0, \ldots, \mathbf{w}_{m-1}$ jejich lineární nezávislost. Proto se musí algoritmus zastavit maximálně po n krocích. Navíc pokud $i > j + 2$ máme

$$\begin{aligned} c_{i\,j}\mathbf{w}_j^T \mathbf{A}\mathbf{w}_j &= \mathbf{w}_j^T \mathbf{A}^2 \mathbf{w}_{i-1} = \mathbf{w}_j^T \mathbf{A}^T \mathbf{A}\mathbf{w}_{i-1} = (\mathbf{A}\mathbf{w}_j)^T \mathbf{A}\mathbf{w}_{i-1} = \\ &= (\mathbf{w}_{j+1} + \sum_{k=0}^{j} c_{j+1\,k}\mathbf{w}_k)^T \mathbf{A}\mathbf{w}_{i-1} = 0. \end{aligned}$$

Tedy $c_{i\,j} = 0$. Tím se zjednoduší definice $\mathbf{w}_i$ pro $i \geq 2$ na

$$\mathbf{w}_i = \mathbf{A}\mathbf{w}_{i-1} - c_{i\,i-1}\mathbf{w}_{i-1} - c_{i\,i-2}\mathbf{w}_{i-2}.$$

Nyní dokážeme správnost řešení. Máme

$$\mathbf{A}\mathbf{x} = \sum_{j=0}^{m-1} \frac{\mathbf{w}_j \cdot \mathbf{b}}{\mathbf{w}_j^T \mathbf{A}\mathbf{w}_j} \mathbf{A}\mathbf{w}_j.$$

Pro $k \in 0, \ldots, m-1$ dostáváme z $\mathbf{A}$-ortogonálnosti vztah

$$\mathbf{w}_k^T \mathbf{A}\mathbf{x} = \sum_{j=0}^{m-1} \frac{\mathbf{w}_j \cdot \mathbf{b}}{\mathbf{w}_j^T \mathbf{A}\mathbf{w}_j} \mathbf{w}_k^T \mathbf{A}\mathbf{w}_j = \mathbf{w}_k^T \mathbf{b}.$$

Po úpravě máme

$$\mathbf{w}_k^T(\mathbf{A}\mathbf{x}-\mathbf{b})=\mathbf{0}. \tag{3.9}$$

Nechť $\langle M\rangle$ značí lineární obal množiny M. Pomocí upravené definice $\mathbf{w}_i$ můžeme vyjádřit jednotlivé vektory následovně

$$\begin{aligned}
\mathbf{w}_0 &= \mathbf{b},\\
\mathbf{w}_1 &= \mathbf{A}\mathbf{w}_0 - c_{1\,0}\mathbf{w}_0,\\
\mathbf{w}_2 &= \mathbf{A}\mathbf{w}_1 - c_{2\,0}\mathbf{w}_0 - c_{2\,1}\mathbf{w}_1,\\
&\vdots\\
\mathbf{w}_{m-1} &= \mathbf{A}\mathbf{w}_{m-2} - c_{m-1\,0}\mathbf{w}_0 - \ldots - c_{m-1\,m-2}\mathbf{w}_{m-2},\\
\mathbf{w}_m &= \mathbf{A}\mathbf{w}_{m-1} - c_{m\,0}\mathbf{w}_0 - \ldots - c_{m\,m-1}\mathbf{w}_{m-1}.
\end{aligned}$$

Protože $\mathbf{w}_m=\mathbf{0}$, dostáváme, že vektory $\mathbf{A}\mathbf{w}_0,\ldots,\mathbf{A}\mathbf{w}_{m-1}$ a $\mathbf{b}$ leží v lineárním obalu množiny $\{\mathbf{w}_0,\ldots,\mathbf{w}_{m-1}\}$. Víme, že $\mathbf{A}\mathbf{x}\in\langle\{\mathbf{A}\mathbf{w}_0,\ldots,\mathbf{A}\mathbf{w}_{m-1}\}\rangle$, a proto $\mathbf{A}\mathbf{x}-\mathbf{b}$ leží v $\langle\{\mathbf{A}\mathbf{w}_0,\ldots,\mathbf{A}\mathbf{w}_{m-1},\mathbf{b}\}\rangle$ tedy i v lineárním obalu $\{\mathbf{w}_0,\ldots,\mathbf{w}_{m-1}\}$. Z toho plyne, že vektor $\mathbf{A}\mathbf{x}-\mathbf{b}$ je možné vyjádřit pomocí $\mathbf{w}_i$ a z rovnice 3.9 dostáváme

$$(\mathbf{A}\mathbf{x}-\mathbf{b})^T(\mathbf{A}\mathbf{x}-\mathbf{b})=0,$$

tedy $\mathbf{A}\mathbf{x}=\mathbf{b}$.

□

Lanczošovův iterační algoritmus poskytuje přesné řešení v $\mathbb{R}$, pokud nedochází k zaokrouhlovacím chybám. Našim cílem je přizpůsobit tento algoritmus použití v konečných tělesech. Zřejmě většinu operací a kroků je možné do konečného tělesa převést, ale problém nastává s podmínkou

$$\mathbf{w}_i\neq 0 \;\Rightarrow\; \mathbf{w}_i^T\mathbf{A}\mathbf{w}_i\neq 0. \tag{3.10}$$

To zřejmě platí v $\mathbb{R}$, ale v konečném tělese F to již obecně platit nemusí. Pokud by $|F|\gg n$, pak podmínka bude platit s přijatelně velkou pravděpodobností. Ale my chceme řešit matici $\mathbf{B}$ nad $\mathbb{Z}_2$, kde přibližně v polovině případů výše uvedená podmínka platit nebude. Některé dřívější modifikace využívaly algebraická rozšíření $\mathbb{Z}_2$, aby tento problém obešly. Avšak počítání v $\mathbb{Z}_2$ má v tomto případě mnoho výhod, je možné provádět různé operace najednou s více vektory, které reprezentujeme jako matici $n\times d$, kde d je bitová délka slova se kterou nativně počítač pracuje (dnes 32 nebo 64). Navíc jednotlivé operace můžeme provádět přímo jako bitové operace, tím opět dochází k významnému zrychlení. V práci [23] Peter Montgomery upravil Lanczošův algoritmus tak, aby pracoval nad $\mathbb{Z}_2$ a tedy využíval všech popsaných možností. K popisu algoritmu budeme potřebovat následující definice.

Definice 3.8. Nechť $\mathcal{V}$ a $\mathcal{W}$ jsou podprostory $(\mathbb{Z}_2)^n$. Pak řekneme, že $\mathcal{V}$ a $\mathcal{W}$ jsou $\mathbf{A}$-*ortogonální podprostory*, pokud

$$\mathbf{v}^T\mathbf{A}\mathbf{w} = 0, \ \ \forall\, \mathbf{v} \in \mathcal{V}, \mathbf{w} \in \mathcal{W},$$

kde $\mathbf{A} \in (\mathbb{Z}_2)^{n\times n}$ je symetrická pozitivně definitní matice. Zapisujeme jako $\mathcal{V}^T\mathbf{A}\mathcal{W} = 0$.

Definice 3.9. Nechť $\mathcal{W}$ je podprostor $(\mathbb{Z}_2)^n$ s bází $\{\mathbf{w}_1, \ldots, \mathbf{w}_k\}$. Pak řekneme, že $\mathcal{W}$ je $\mathbf{A}$-*invertibilní*, pokud matice $\mathbf{W}^T\mathbf{A}\mathbf{W}$ je invertibilní, kde

$$\mathbf{W} = (\mathbf{w}_1, \ldots, \mathbf{w}_k) = \begin{pmatrix} w_{1\,1} & \ldots & w_{k\,1} \\ \vdots & & \vdots \\ w_{1\,n} & \ldots & w_{k\,n} \end{pmatrix}.$$

Zřejmě nezáleží na tom, jakou bázi podprostoru $\mathcal{W}$ zvolíme, protože mezi jednotlivými bázemi existují invertibilní transformace.

Montgomeryho bloková verze algoritmu používá místo vektorů $\mathbf{w_i}$ podprostory $\mathcal{W}_i$, tím téměř eliminuje problémy s podmínkou 3.10, která odpovídá tomu, že $\mathcal{W}_i$ není $\mathbf{A}$-ortogonální sám k sobě. To je ekvivaletní tomu, že $\mathcal{W}$ je $\mathbf{A}$-invertibilní. Nechť podprostory $\mathcal{W}_i$ splňují následující vlastnosti:

(i) $\mathcal{W}_i$ je $\mathbf{A}$-invertibilní,

(ii) $\mathcal{W}_i^T\mathbf{A}\mathcal{W}_j = 0$, $i \neq j$,

(iii) $\mathbf{A}\mathcal{W} = \{\mathbf{A}\mathbf{w} : \mathbf{w} \in \mathcal{W}\} \subseteq \mathcal{W}$, kde $\mathcal{W} = \sum_{j=0}^{m-1} \mathcal{W}_j$.

Nechť matice $\mathbf{W}_i$ podobně jako v definici 3.9 reprezentuje bázi podprostoru $\mathcal{W}_i$. Pak řešení rovnice $\mathbf{A}\mathbf{x} = \mathbf{b}$ nad $\mathbb{Z}_2$, kde $\mathbf{A}$ je symetrická matice a $\mathbf{b}$ je nenulový vektor, dostaneme jako

$$\mathbf{x} = \sum_{j=0}^{m-1} \mathbf{W}_j(\mathbf{W}_j^T\mathbf{A}\mathbf{W}_j)^{-1}\mathbf{W}_j^T\mathbf{b}.$$

Důkaz správnosti řešení $\mathbf{x}$ je podobný jako u původního Lanczošova algoritmu.

Nyní se zaměříme na generování matic $\mathbf{W}_i, i = 0, \ldots m-1$, aby platily předchozí vlastnosti. Nechť $\mathbf{V}_0$ je libovolně zvolená $n \times d$ matice. Původní Lanczošovy iterace nahradíme takto

$$\begin{array}{lcl} \mathbf{W}_i & = & \mathbf{V}_i\mathbf{S}_i, \\ \mathbf{C}_{i+1,\,j} & = & (\mathbf{W}_j^T\mathbf{A}\mathbf{W}_j)^{-1}\mathbf{W}_j^T\mathbf{A}(\mathbf{A}\mathbf{W}_i\mathbf{S}_i^T + \mathbf{V}_i), \\ \mathbf{V}_{i+1} & = & \mathbf{A}\mathbf{W}_i\mathbf{S}_i^T + \mathbf{V}_i - \sum_{j=0}^{i} \mathbf{W}_j\mathbf{C}_{i+1,\,j}. \end{array}$$

Algoritmus zastavíme, když $\mathbf{V}_i^T\mathbf{A}\mathbf{V}_i = \mathbf{0}$, $i = m$. $\mathbf{S}_i$ jsou $d \times d_i$ matice, které slouží k výběru sloupců z $\mathbf{V}_i$ tak, aby pro matici $\mathbf{W}_i$ platilo, že $\mathbf{W}_i^T\mathbf{A}\mathbf{W}_i$ je invertibilní. Tedy podprostor $\mathcal{W}_i$ je $\mathbf{A}$-invertibilní. Přitom chceme, aby $d_i \leq d$

bylo co největší (d_i odpovídá počtu zvolených sloupců). Matice $\mathbf{S}_i$ má v každém sloupci přesně jednu 1 a v řádcích maximálně jednu 1. Jak správně volit matice $\mathbf{S}_i$ je možné nalézt v [23].

Na volbu $\mathbf{V}_{i+1}$ se můžeme dívat následovně. Z $\mathbf{V}_i$ vybíráme sloupce, které nebyly vybrány do $\mathbf{W}_i$, a o kterých víme, že jsou $\mathbf{A}$-ortogonální ke všem $\mathbf{W}_j$, $j < i$. Volba $\mathbf{C}_{i+1,\,i}$ zajistí, že budou $\mathbf{A}$-ortogonální i k $\mathbf{W}_i$. Přitom $\mathbf{V}_i$ dále zajišťuje, že hodnost matice $\mathbf{V}_{i+1}$ nebude omezena hodností $\mathbf{A}\mathbf{W}_i\mathbf{S}_i^T$, protože potom by mohla klesnou brzy k nule.

Následující tvrzení dokazuje, že popsaný postup vede k vytvoření posloupnosti podprostorů, které splňují podmínky (i), (ii), (iii).

Věta 3.10. *Nechť matice $\mathbf{W}_j$, $j = 0, \ldots m-1$ vznikly postupem popsaným výše. Pak odpovídající podprostory $\mathcal{W}_j$ splňují podmínky (i), (ii), (iii). Navíc platí*

$$\mathbf{W}_j^T\mathbf{A}\mathbf{V}_i = \mathbf{0}, \;\; 0 \le j < i \le m. \tag{3.11}$$

Důkaz. Nejprve dokážeme rovnici 3.11. Budeme postupovat indukcí podle i. Pro $i = 0$ rovnice triviálně platí. Nechť tedy platí pro $i < m$ a nechť $0 \le j < i+1$. Pak podle indukčního předpokladu máme $\mathbf{W}_j^T\mathbf{A}\mathbf{V}_i = \mathbf{0}$, $0 \le j < i$, a protože $\mathbf{W}_i = \mathbf{V}_i\mathbf{S}_i$ dostáváme $\mathbf{W}_j^T\mathbf{A}\mathbf{W}_i = \mathbf{0}$, dále ze symetrie $\mathbf{A}$ plyne $\mathbf{W}_i^T\mathbf{A}\mathbf{W}_j = \mathbf{0}$. Přejdeme k $i+1$, máme

$$\begin{aligned}
\mathbf{W}_j^T\mathbf{A}\mathbf{V}_{i+1} &= \mathbf{W}_j^T\mathbf{A}(\mathbf{A}\mathbf{W}_i\mathbf{S}_i^T + \mathbf{V}_i) - \sum_{k=0}^{i}\mathbf{W}_j^T\mathbf{A}\mathbf{W}_k\mathbf{C}_{i+1,\,k} = \\
&= \mathbf{W}_j^T\mathbf{A}(\mathbf{A}\mathbf{W}_i\mathbf{S}_i^T + \mathbf{V}_i) - \mathbf{W}_j^T\mathbf{A}\mathbf{W}_j\mathbf{C}_{i+1,\,j} = \mathbf{0}.
\end{aligned}$$

Druhá rovnost plyne z indukčního předpokladu a poslední z definice $\mathbf{C}_{i+1,\,j}$. Dostáváme $\mathcal{W}_i^T\mathbf{A}\mathcal{W}_j = 0$, $i \neq j$. Dále volba $\mathbf{S}_i$ zajistila, že podprostory $\mathcal{W}_i$ jsou $\mathbf{A}$-ortogonální. Zbývá ukázat, že $\mathbf{A}\mathcal{W} \subseteq \mathcal{W}$. Uvažujme definici $\mathbf{V}_{i+1}$ a vynásobme ji zleva maticí $\mathbf{S}_i$. Dostáváme

$$\begin{aligned}
\mathbf{V}_{i+1}\mathbf{S}_i &= \mathbf{A}\mathbf{W}_i\mathbf{S}_i^T\mathbf{S}_i + \mathbf{V}_i\mathbf{S}_i - \sum_{j=0}^{i}\mathbf{W}_j\mathbf{C}_{i+1,\,j}\mathbf{S}_i = \\
&= \mathbf{A}\mathbf{W}_i + \mathbf{W}_i - \sum_{j=0}^{i}\mathbf{W}_j\mathbf{C}_{i+1,\,j}\mathbf{S}_i.
\end{aligned}$$

Po úpravě máme

$$\begin{aligned}
\mathbf{A}\mathbf{W}_i &= \mathbf{V}_{i+1}\mathbf{S}_i - \mathbf{W}_i + \textstyle\sum_{j=0}^{i}\mathbf{W}_j\mathbf{C}_{i+1,\,j}\mathbf{S}_i = \mathbf{V}_{i+1}\mathbf{S}_i + \mathbf{W}', \\
\mathbf{V}_i &= \mathbf{V}_{i+1} - \mathbf{A}\mathbf{W}_i\mathbf{S}_i^T + \textstyle\sum_{j=0}^{i}\mathbf{W}_j\mathbf{C}_{i+1,\,j} = \mathbf{V}_{i+1} - \mathbf{A}\mathbf{W}_i\mathbf{S}_i^T + \mathbf{W}'',
\end{aligned}$$

kde $\mathbf{W}'$ a $\mathbf{W}''$ jsou matice, jejichž sloupcové vektory leží v $\mathcal{W}$. Začneme-li od $\mathbf{V}_m = \mathbf{0} \subseteq \mathcal{W}$ (opět chápáno přes sloupcové vektory) dostaneme zpětnou indukcí, že $\mathbf{AW}_i \subseteq \mathcal{W}$ a $\mathbf{V}_i \subseteq \mathcal{W}$ pro $0 \leq i \leq m-1$. Tedy $\mathbf{A}\mathcal{W} \subseteq \mathcal{W}$.

□

Výpočet matic $\mathbf{V}_i$ je možné zjednodušit do podoby

$$\mathbf{V}_{i+1} = \mathbf{AW}_i\mathbf{S}_i^T + \mathbf{V}_i - \mathbf{W}_i\mathbf{C}_{i+1,\,i} - \mathbf{W}_{i-1}\mathbf{C}_{i+1,\,i-1} - \mathbf{W}_{i-2}\mathbf{C}_{i+1,\,i-2}.$$

Pokud zavedeme

$$\mathbf{W}_i^{inv} = \mathbf{S}_i(\mathbf{W}_i^T\mathbf{AW}_i)\mathbf{S}_i^T = \mathbf{S}_i(\mathbf{S}_i^T\mathbf{V}_i^T\mathbf{AV}_i\mathbf{S}_i)\mathbf{S}_i^T,$$

pak můžeme výpočet $\mathbf{V}_i$ ještě více zjednodušit na

$$\mathbf{V}_{i+1} = \mathbf{AV}_i\mathbf{S}_i\mathbf{S}_i^T + \mathbf{V}_i\mathbf{D}_{i+1} + \mathbf{V}_{i-1}\mathbf{E}_{i+1} + \mathbf{V}_{i-2}\mathbf{F}_{i+1}.$$

Kde

$$\begin{aligned}
\mathbf{D}_{i+1} &= \mathbf{I}_d - \mathbf{W}_i^{inv}(\mathbf{V}_i^T\mathbf{A}^2\mathbf{V}_i\mathbf{S}_i\mathbf{S}_i^T + \mathbf{V}_i^T\mathbf{AV}_i), \\
\mathbf{E}_{i+1} &= -\mathbf{W}_{i-1}^{inv}\mathbf{V}_i^T\mathbf{AV}_i\mathbf{S}_i\mathbf{S}_i^T, \\
\mathbf{E}_{i+1} &= -\mathbf{W}_{i-2}^{inv}(\mathbf{I}_d - \mathbf{V}_{i-1}^T\mathbf{AV}_{i-1}\mathbf{W}_{i-1}^{inv}), \\
&\quad (\mathbf{V}_{i-1}^T\mathbf{A}^2\mathbf{V}_{i-1}\mathbf{S}_{i-1}\mathbf{S}_{i-1}^T + \mathbf{V}_{i-1}^T\mathbf{AV}_{i-1})\mathbf{S}_i\mathbf{S}_i^T.
\end{aligned}$$

Pro $i < 0$ volíme $\mathbf{W}_i^{inv}$ a $\mathbf{V}_i$ jako $\mathbf{0}$ a $\mathbf{S}_i = \mathbf{I}_d$. Taková úprava výrazů je pro implementaci výhodná, protože matice $\mathbf{V}_i$, $\mathbf{W}_i$ reprezentujeme jako vektory integerů délky n a matice $\mathbf{S}_i$, $\mathbf{D}_i$, $\mathbf{E}_i$ a $\mathbf{F}_i$ dokonce jen délky d. Veškeré výpočty jsou proto velmi rychlé a paměťové nároky jsou malé. Dále je zřejmé, že místo vektoru $\mathbf{b}$ můžeme uvažovat matici, jejíž sloupcové vektory představují jednotlivá zadání, a tedy jedním během algoritmu najít více řešení pro různé pravé strany.

Doposud jsme se zabývali případem, kdy $\mathbf{b} \in (\mathbb{Z}_2)^n$ byl nenulový vektor. V číselném sítu ale potřebujeme vyřešit rovnici

$$\mathbf{Bx} = \mathbf{0}.$$

Přitom $\mathbf{B} \in (\mathbb{Z}_2)^{n_1 \times n_2}$ není symetrická matice, $n_2 > n_1$. Abychom splnili podmínku algoritmu, který potřebuje symetrickou matici, použijeme místo $\mathbf{B}$ matici $\mathbf{A} = \mathbf{B}^T\mathbf{B} \in (\mathbb{Z}_2)^{n \times n}$, $n = n_2$ (matici $\mathbf{A}$ explicitně nevyjadřujeme, nejprve násobíme s $\mathbf{B}$ a poté s $\mathbf{B}^T$). Zřejmě každé řešení $\mathbf{Bx} = \mathbf{0}$ je i řešením $\mathbf{Ax} = \mathbf{0}$, ale opačná implikace obecně neplatí, proto musíme z nalezených řešení pro $\mathbf{A}$ vybrat jen ty, které řeší i $\mathbf{Bx} = \mathbf{0}$. Postupujeme tedy následovně. Nejprve náhodně vygenerujeme matici $\mathbf{Y} \in (\mathbb{Z}_2)^{n \times d}$ a pomocí Lanczošova algoritmu budeme řešit rovnici

$$\mathbf{AX} = \mathbf{AY}$$

s neznámou $\mathbf{X}$. Jak bylo řečeno výše, najednou počítáme více řešení pro různé pravé strany. I když generujeme matici $\mathbf{Y}$ náhodně, musíme se vyvarovat případu, kdy nějaký sloupcový vektor $\mathbf{Y}$ je $\mathbf{A}$-ortogonální sám k sobě, protože jinak může algoritmus selhat (nalezne pouze triviální řešení). V algoritmu volíme $\mathbf{V}_0 = \mathbf{AY}$ a iterace končíme, když $\mathbf{V}_m^T\mathbf{AV}_m = \mathbf{0}$. Řešení dostaneme jako

$$\mathbf{X} = \sum_{j=0}^{m-1} \mathbf{W}_j(\mathbf{W}_j^T\mathbf{AW}_j)^{-1}\mathbf{W}_j^T\mathbf{AY} = \sum_{j=0}^{m-1} \mathbf{V}_j\mathbf{W}_j^{inv}\mathbf{V}_j^T\mathbf{V}_0.$$

Pokud $\mathbf{V}_m = \mathbf{0}$, pak

$$\mathbf{A}(\mathbf{X} - \mathbf{Y}) = \mathbf{0}.$$

A tedy sloupové vektory $\mathbf{X} - \mathbf{Y}$ jsou hledané řešení, které ale musíme ještě dále upravit, aby vyhovovaly i matici $\mathbf{B}$. Ne vždy nastane $\mathbf{V}_m = \mathbf{0}$, ale opět je možné dokázat, že platí předchozí vztah a navíc i sloupcové vektory $\mathbf{V}_m$ jsou řešením $\mathbf{Ax} = \mathbf{0}$. Dostáváme tedy přibližně $2d$ řešení (některá mohou být triviální). Nyní z nich potřebujeme zkonstruovat řešení rovnice s maticí $\mathbf{B}$. Ze sloupcových vektorů matic $\mathbf{X} - \mathbf{Y}$ a $\mathbf{V}_m$ vytvoříme matici $\mathbf{Z}$. Pomocí Gaussovy eliminační metody nalezneme bázi nulového prostoru matice $\mathbf{BZ}$, to není příliš náročné, protože tato matice má rozměry pouze $n_1 \times 2d$. Z těchto bázových vektorů sestavíme matici $\mathbf{U}$, velikost báze je maximálně $2d$, tedy matice U je opět malá (maximálně $2d \times 2d$). Máme

$$\mathbf{B}(\mathbf{ZU}) = (\mathbf{BZ})\mathbf{U} = \mathbf{0}.$$

Tedy hledaná řešení jsou lineárně nezávislé sloupce z matice $\mathbf{ZU}$, těch může být až $2d$ (předpokládáme-li $n_2 > 2d + n_1$).

Lanczošův algoritmus je poměrně rychlý a nepředstavuje problematickou paměťovou náročnost. Z popisu ale není úplně zřejmé, zda je možné tento algoritmus paralelizovat. Na rozdíl od prosívací fáze, která je snadno paralelizovatelná (častost a velikost přenášených dat u vzájemné komunikace výpočetních uzlů je zcela minimální, prodlevy mohou být až několika denní při rekordních faktorizacích), je paralelizace Lanczošova algoritmus složitější, ale existuje několik návrhů, jak algoritmus nejvhodněji paralelizovat. Nejnáročnější část algoritmu je násobení s maticí $\mathbf{B}$ (ostatní matice představují pouze integerové vektory). Při paralelizaci se tedy matice $\mathbf{B}$ vhodně rozdělí na větší počet částí, se kterými se pracuje samostatně a výsledky se poté přeposílají řídícímu programu. Protože je potřeba tyto data posílat velmi rychle, aby nedocházelo ke zdžování výpočtu, používají se víceprocesorové počítače, případně počítače propojené ve velmi výkonné síti (prodlevy musí být minimální). Jak správně rozdělovat matici $\mathbf{B}$ je v současnosti nejsložitější na celé paralelizaci. Víme,

že matice **B** má speciální vlastnosti, kdy se střídají řádky s větším výskytem jedniček (to odpovídá začátkům faktorizačních bází, kde jsou malé prvky, a kvadratickým charakterům) s velmi řídkými řádky (prostředky a konce faktorizačních bází). Dobrá strategie rozdělování matice **B** musí s tímto počítat. Několik přístupů lze nalézt v [11] a [27]. Dosahované zrychlení je přibližně 50 % za každý další procesor ale s klesající efektivností.

3.6 Odmocninová fáze

Z lineární fáze algoritmu jsme získali několik řešení rovnice $\mathbf{Bx} = \mathbf{0}$. Každé řešení určuje množiny indexů M tak, že

$$\begin{aligned}\prod_{j\in M}(a_j + b_j\vartheta_1)\mathbb{Z}[\vartheta_1] &= \prod_{j=1}^{k_1} P_j^{2e_j},\\ \prod_{j\in M}(a_j + b_j\vartheta_2)\mathbb{Z}[\vartheta_2] &= \prod_{j=1}^{k_2} Q_j^{2f_j}.\end{aligned}$$

I když jsme existenci odmocniny zajistili pomocí kvadratických charakterů, obecně neplatí, že by odmocnina z výrazu $\prod_{j\in M}(a_j + b_j\vartheta_i) \in \mathbb{Z}[\vartheta_i]$ také ležela v $\mathbb{Z}[\vartheta_i]$. Ale pokud výraz vynásobíme $(f_i'(\vartheta_i))^2$, pak již do $\mathbb{Z}[\vartheta_i]$ patří podle věty 2.26. Označme $g_i = f_i'(m)$, správně tedy máme vztah

$$g_2^2\,\varphi_1((f_1'(\vartheta_1))^2\prod_{j\in M}(a_j + b_j\vartheta_1)) \equiv g_1^2\,\varphi_2((f_2'(\vartheta_2))^2\prod_{j\in M}(a_j + b_j\vartheta_2)) \bmod N.$$

Pomocí některého z algoritmů, které později popíšeme, určíme odmocninu

$$\alpha_i = \sqrt{(f_i'(\vartheta_i))^2\prod_{j\in M}(a_j + b_j\vartheta_i)}.$$

Dostáváme

$$(f_2'(m))^2\varphi_1(\alpha_1^2) \equiv (f_1'(m))^2\varphi_2(\alpha_2^2) \bmod N,$$

tedy

$$(f_2'(m)\varphi_1(\alpha_1))^2 \equiv (f_1'(m)\varphi_2(\alpha_2))^2 \bmod N.$$

Nechť $x = f_2'(m)\varphi_1(\alpha_1)$ a $y = f_1'(m)\varphi_2(\alpha_2)$. Pak s pravděpodobností $1/2$ je $\gcd(N, x - y)$ netriviální faktor N. Pokud nalezneme pouze triviální faktor, vyzkoušíme další řešení. Při počítání s volnými relacemi nedochází k žádné úpravě, protože je uvažujeme jako speciální případ, kdy $b_j = 0$ a $a_j = p$.

Použití nemonických polynomů již ale vyžaduje několik změn, které popíšeme podrobněji u jednotlivých metod.

Nalezení odmocniny výrazu $(f_i'(\vartheta_i))^2 \prod_{j \in M}(a_j + b_j\vartheta_i)$ bylo hlavní překážkou (spolu se zajištěním její existence), která bránila v přechodu od speciálního tvaru čísla N k obecnému. Až využití kvadratických charakterů a vymyšlení alespoň jedné relativně rychlé metody k nalezení odmocniny obrovského čísla v $\mathbb{Z}[\vartheta_i]$, které je násobkem mnoha menších, umožnilo vznik obecného číselného síta. Do té doby se jako teoretická možnost uvažovalo roznásobení a použití klasického výpočtu odmocniny, ale náročnost výpočtu byla srovnatelná s prosívací fází. První úspěšnou metodou je Newtonova iterační metoda [3]. Tato metoda není zcela vhodná, protože stále musíme pracovat s příliš velkými čísly. Vylepšení Newtonovy metody pochází od Jean-Marca Couveignese [6], ale tato varianta požaduje, aby stupeň odpovídajícího minimálního polynomu byl lichý. Jiný přístup zvolil Peter Montgomery, který využívá skutečnosti, že známe faktorizaci jednotlivých ideálů, k postupné redukci velikosti výrazu a následnému použití klasických metod výpočtu odmocniny v $\mathbb{Z}[\vartheta_i]$. Všechny tyto metody popíšeme, ale nejprve připomeňme nejjednodušší případ, který žádnou z těchto metod nepotřebuje. Pokud jeden z polynomů ($i = 2$) je prvního stupně, zřejmě neopustíme obor celých čísel a rozklad, který získáváme je klasický prvočíselný rozklad v $\mathbb{Z}$. Máme tedy

$$\prod_{j \in M}(a_j + b_j m) = \prod_{j=1}^{k_2} q_j^{2f_j},$$

kde m je společný kořen polynomů f_i a q_j jsou prvočísla. Tedy podle výše uvedeného dostáváme

$$\alpha_2 = f_1'(m) \cdot \prod_{j=1}^{k_2} q_j^{f_j} \quad \in \mathbb{Z}.$$

Nyní se již můžeme zaměřit na algoritmy počítající odmocninu v $\mathbb{Z}[\vartheta_i]$. V popisu nebudeme používat index i pro rozlišení číselných těles a souvisejících objektů.

3.6.1 Newtonova iterační metoda

Tato metoda je založená na teorii okolo Newtonových iterací. K popisu algoritmu potřebujeme následující lemma.

Lemma 3.11. *Nechť R je komutativní okruh a nechť $ac_0^2 \equiv 1 \mod J$, kde J je ideál v R, a, $c_0 \in R$. Dále nechť 2 je invertibilní v R. Definujme $c_i \in R$, $0 < i \leq k$, následovně*

$$c_{i+1} \equiv \frac{c_i(3 - ac_i^2)}{2} \mod J^{2^{i+1}}.$$

Pak platí $ac_k^2 \equiv 1 \mod J^{2^k}$.

Důkaz. Budeme postupovat indukcí. Pro $i = 0$ lemma zřejmě platí. Nechť tedy lemma platí pro $j \leq i$. Dokážeme platnost pro $i + 1$. Následující kongruence počítáme modulo $(J^{2^i})^2 = J^{2^{i+1}}$

$$\begin{aligned} 1 - ac_{i+1}^2 &\equiv \frac{1}{4}(4 - a(2c_{i+1})^2) \equiv \frac{1}{4}(4 - a(c_i(3 - ac_i^2))^2) \equiv \\ &\equiv -\frac{1}{4}(a^3c_i^6 - 6a^2c_i^4 + 9ac_i^2 - 4) \equiv -\frac{1}{4}(ac_i^2 - 4)(a^2c_i^4 - 2ac_i^2 + 1) \\ &\equiv -\frac{1}{4}(ac_i^2 - 4)(ac_i^2 - 1)^2 \equiv 0 \mod J^{2^{i+1}}, \end{aligned}$$

poslední kongruence plyne z indukčního předpokladu.

□

Nechť $f(x) = \sum_{i=0}^{d} c_i x^i$ je minimální polynom prvku ϑ nad $\mathbb{Z}$ ne nutně monický. Zvolme prvočíslo p, pro které je $f \bmod p$ ireducibilní v $\mathbb{Z}_p[x]$. Existence a nalezení takového to prvočísla je nejslabším článkem algoritmu (i následujících), protože obecně nemusí existovat a ani neumíme vhodně odhadnout složitost hledání. Ale v praxi se ukazuje, že lze snadno tyto prvočísla nalézt (používá se prosté zkoušení od nejmenšího uvažovaného prvočísla). Podobně jako v části 2.3 můžeme ukázat, že

$$\mathbb{Z}[\vartheta]/p\mathbb{Z}[\vartheta] \cong \mathbb{Z}_p[x]/\bar{f}\mathbb{Z}_p[x].$$

Opět budeme značit $\bar{}$ prvky modulo odpovídající číslo. Protože $\bar{f}$ je ireducibilní, je $\mathbb{Z}_p[x]/\bar{f}\mathbb{Z}_p[x]$ těleso, které obsahuje p^d prvků (polynomy v $\mathbb{Z}_p[x]$ stupně menšího než d). Dále z ireducibility $\bar{f}$ plyne, že $f'(\vartheta) \notin p\mathbb{Z}[\vartheta]$, protože jinak by $\bar{f}$ a $\bar{f}'$ byly soudělné. I pro prvky $a_j + \vartheta b_j$ dostáváme, že neleží v $p\mathbb{Z}[\vartheta]$, protože a a b jsou nesoudělná. Celkově tedy máme, že $\gamma = (f'(\vartheta))^2 \prod_{j \in M}(a_j + b_j\vartheta)$ neleží v $p\mathbb{Z}[\vartheta]$ a můžeme součin vyjádřit jako nenulový polynom stupně menšího než d nad $\mathbb{Z}_p$ (využíváme izomorfismu, podobně můžeme výraz chápat jako prvek z $\mathbb{Z}[\vartheta]/p\mathbb{Z}[\vartheta]$). Vše již počítáme modulo p, proto roznásobení není náročné. Víme, že γ je druhou mocninou nějakého čísla α, které leží v $\mathbb{Z}[\vartheta]$ (toho jsme dosáhli přenásobením $f'(\vartheta))^2$). Tedy musí existovat $\xi_0 \in \mathbb{Z}[\vartheta]$ tak, že

$$\gamma\bar{\xi}_0^2 \equiv 1 \mod p\mathbb{Z}[\vartheta].$$

Existence plyne z toho, že γ i 1 jsou kvadratická rezidua, tedy i γ^{-1} musí být kvadratické reziduum. $\bar{\xi}_0$ nalezneme pomocí upraveného Tonelli-Shanksova algoritmu ([5] algoritmus 1.5.1), přitom je jednoznačné až na znaménko. Zřejmě $\alpha \equiv \bar{\xi}_0^{-1} \mod p\mathbb{Z}[\vartheta]$. Dále platí i následující izomorfismy

$$\mathbb{Z}[\vartheta]/p^{2^k}\mathbb{Z}[\vartheta] \cong \mathbb{Z}_{p^{2^k}}[x]/\bar{f}\mathbb{Z}_{p^{2^k}}[x],$$

kde $k \geq 0$. Nyní můžeme použít lemma 3.11 a postupně upřesňovat hledanou odmocninu α. V k-tém kroku dostaneme $\bar{\xi}_k$, kde $\alpha \equiv \bar{\xi}_k^{-1} \mod p^{2^k}\mathbb{Z}[\vartheta]$. Veškeré výpočty provádíme s polynomy podle uvedených izomorfismů, přitom

volíme koeficienty tak, aby byly v absolutní hodnotě menší než $(p^{2^k}-1)/2$. Iterace zastavíme ve chvíli, kdy pro nějaké $k \geq 0$ bude p^{2^k} dostatečně velké (v nejhorším případě dvakrát tak velké jako maximální odhadovaná velikost koeficientů u α, tento odhad uvedeme později). Máme

$$\alpha \equiv \pm\gamma\bar{\xi}_k \mod p^{2^k}\mathbb{Z}[\vartheta].$$

A α již musí být přímo rovno $\gamma\bar{\xi}_k$. Nechť polynom $g \in \mathbb{Z}[x]$ odpovídá prvku $\gamma\bar{\xi}_k$. Pak jednoduše dostáváme $\varphi(\alpha) \equiv g(m) \mod N$.

Lze dokázat ([6]), že pro maximální velikosti koeficientů polynomu $g(x)$, který odpovídá α, platí, že jsou menší než

$$d^{\frac{3}{2}}\|f\|^{d-2}(2u\|f\|)^{\frac{|M|}{2}},$$

kde $\|f\| = \sqrt{\sum_{i=0}^{d} c_i^2}$ a u splňuje $u \geq a_j$, $u \geq b_j$ pro všechna $j \in M$.

Newtonova metoda není příliš vhodná, protože v posledních iteracích se již počítá s obrovskými čísly a výpočet začíná být velmi náročný. Následující metoda se snaží tento problém obejít za cenu ztráty univerzálnosti.

3.6.2 Couveignesova metoda

Při pohledu na Newtonovu iterační metodu nás jistě napadá, jestli by nebylo možné počítat modulo více prvočísel a poté použitím Čínské věty o zbytku získat α. To ale není zcela jednoduché, protože musíme vybrat správné odmocniny (zřejmě vždy dostaneme dvě lišící se znaménkem), aby nedošlo ke konfliktu známek. K určení správné odmocniny použijeme normu, to ale samo o sobě nestačí, protože norma z $-\mu \in \mathbb{Z}[\vartheta]$ může být stejná jako z μ. Víme, že $N(-1) = (-1)^d$, pokud tedy je d liché, můžeme od sebe jednotlivé odmocniny odlišit. Tedy Couveignesův algoritmus můžeme použít pouze v případě, kdy polynom f je lichého stupně. Protože budeme pracovat s normami a budeme využívat prvoideálové rozklady, musíme pracovat ve správném oboru. Pro zahrnutí případu, kdy by byl polynom f nemonický, opět zavedeme označení se střížkami. Následující vztahy uvedeme obecně, tedy i pro $c_d = 1$. Položme

$$\widehat{\gamma} = (\widehat{f'}(\widehat{\vartheta}))^2 \prod_{j \in M}(c_d a_j + b_j\widehat{\vartheta}).$$

Zbavujeme se inverze čísla c_d. Protože $\widehat{f'}(\widehat{\vartheta}) = f'(\vartheta)c_d^{d-2}$, dostáváme vztah $\gamma = \widehat{\gamma}c_d^{-|M|-2(d-2)}$, kde γ je definována stejně jako v Newtonově metodě. Opět používáme $\widehat{f'}(\widehat{\vartheta})$ k zajištění náležení odmocniny do $\mathbb{Z}[\widehat{\vartheta}]$. Po zbytek výpočtu budeme $c_d^{-|M|+2(d-2)}$ ignorovat, vrátíme se k němu až v závěru. Tedy podobně jako výše jsme zajistili, aby $\widehat{\gamma}$ byl kvadrát a jeho odmocnina $\widehat{\alpha}$ ležela v $\mathbb{Z}[\widehat{\vartheta}]$. Z prosívací fáze známe rozklady $F(a,-b)$ na prvočísla (případně je dopočítáme). Tedy máme

$$\prod_{j\in M} F(a_j, -b_j) = \prod_{i=1}^{k} p_i^{2s_i}.$$

A protože $N(c_d a_j + b_j\widehat{\vartheta}) = c_d^{d-1}F(a,-b)$, dostáváme pro normu $\widehat{\alpha}$ vztah

$$N(\widehat{\alpha}) = \pm N(\widehat{f}'(\widehat{\vartheta}))c_d^{(d-1)|M|/2}\prod_{i=1}^{k} p_i^{s_i}. \tag{3.12}$$

Pro další výpočet místo $\pm N(\widehat{f}'(\widehat{\vartheta}))$ bereme jeho absolutní hodnotu. Nechť tedy B je maximální odhad pro velikost koeficientů polynomu $g(x)$ odpovídajícímu číslu $\widehat{\alpha}$. Našim cílem je vypočítat $g(c_d m) \bmod N$ ($c_d m$ je kořenem $\widehat{f}$). Nejprve nalezneme několik prvočísel $q_1, \ldots, q_l$, každé musí splňovat podmínky pro Newtonovu metodu, a k nim exponenty $e_1, \ldots, e_l$ tak, aby

$$Z = q_1^{2^{e_1}} \cdot \ldots \cdot q_l^{2^{e_l}} > B.$$

Přitom chceme, aby $z_i = q_i^{2^{e_i}} \approx N$. Položme $Z_i = Z/z_i$. Protože nepotřebujeme znát přímo $\widehat{\alpha}$ ale jen $\varphi(\widehat{\alpha}) \equiv g(c_d m) \mod N$, využijeme ještě následující dvě lemmata.

Lemma 3.12. *Nechť Z, Z_i a $z_i \in \mathbb{Z}$ jsou definována jako výše a nechť pro $a_i \in \mathbb{Z}$ platí $a_i Z_i \equiv 1 \bmod z_i$, $1 \le i \le l$. Pak pro $u \in \mathbb{Z}$, $|u| \le Z/2$, platí*

$$u \equiv \sum_{i=1}^{k} a_i u_i Z_i - rZ \mod N,$$

kde $u_i \equiv u \bmod z_i$ a $r \in \mathbb{Z}$ splňuje $|(\sum_{i=1}^{l} a_i u_i z_i^{-1}) - r| \le 1/2$.

Důkaz. Nechť $v = \sum_{i=1}^{l} a_i u_i Z_i$ (tedy $u \equiv v \mod Z$ podle Lagrangeova algoritmu). Pak zřejmě $v \equiv v - rZ \mod Z$. A z podmínky pro r dostáváme $|v - rZ| \le Z/2$, protože $\sum_{i=1}^{l} a_i u_i z_i^{-1} = v/Z$. Pak ovšem $u = v - rZ$ a tento vztah tedy platí i modulo N. □

Číslo r dostaneme použitím druhého lemma.

Lemma 3.13. *Při nezměněném označení. Nechť $0 < \varepsilon < \frac{1}{2Z}$, kde Z je liché, a nechť $\rho \in \mathbb{R}$ splňuje $|(\sum_{i=1}^{l} a_i u_i z_i^{-1}) - \rho| < \varepsilon$. Pak r z předchozího lemma je nejbližším celým číslem k ρ.*

Důkaz. Podle předpokladu je $|u| \le Z/2$, ale protože Z je liché, máme navíc $|u| \le (Z-1)/2$. Tedy

$$|u| \le \frac{Z-1}{2} = Z\left(\frac{1}{2} - \frac{1}{2Z}\right) < Z\left(\frac{1}{2} - \varepsilon\right).$$

Z předchozího lemma víme, že $u = v - rZ$, proto $|v/Z - r| = |u|/Z < \frac{1}{2} - \varepsilon$. Dále z vlastnosti ρ máme $|v/Z - \rho| < \varepsilon$. Dáme obě dvě nerovnosti dohromady a dostáváme

$$|\rho - r| = |\rho - v/Z + v/Z - r| \leq |v/Z - \rho| + |v/Z - r| < \varepsilon + \frac{1}{2} - \varepsilon = \frac{1}{2}.$$

□

Postupujeme tedy následovně. Pro každé prvočíslo q_i pustíme Newtonův iterační algorimus, který zastavíme po e_i krocích (počítáme v $\mathbb{Z}[\widehat{\vartheta}]$ s příslušným polynomem $\widehat{f}$). Z Newtonova algoritmu dostaneme $\bar{\xi}_{i,\,e_i} \in \mathbb{Z}[\widehat{\vartheta}]$, $1 \leq i \leq l$ tak, že

$$\widehat{\alpha} \equiv \pm\widehat{\gamma}\bar{\xi}_{i,\,e_i} \mod q_i^{2^{e_i}}\mathbb{Z}[\widehat{\vartheta}].$$

Dále vypočítáme normu $\widehat{\alpha}$ podle 3.12 modulo q_i. Protože polynom $\widehat{f}$ je modulo q_i ireducibilní, nemůže být tato norma dělitelná q_i. Vypočítáme i normu získaného $\widehat{\gamma}\bar{\xi}_{i,\,e_i}$, chápanou v konečném tělese s q_i^d prvky, podle vzorce

$$N(\zeta) \equiv \zeta^{\frac{q_i^d-1}{q_i-1}} \mod q_i.$$

Pokud se normy liší zvolíme $-\widehat{\gamma}\bar{\xi}_{i,\,e_i}$. Nechť $g_i(x) \in \mathbb{Z}[x]$ je odpovídající polynomiální reprezentace. Pak $g(c_d m) \equiv g_i(c_d m) \bmod z_i = q_i^{2^{e_i}})$. Klademe-li $u = g(c_d m)$ a $u_i = g_i(c_d m)$, pak splňujeme podmínky uvedených lemmat a tedy pomocí nich vypočítáme hledané $g(c_d m) \bmod N$.

Nyní se již můžeme vrátit k vedoucímu koeficientu polynomu f. Víme, že $\alpha = \widehat{\alpha} c_d^{-|M|/2-d+2}$, kde $c_d \in \mathbb{Z}$, a podle předpokladů je $|M|$ sudé. Tedy pokud rozšíříme homomorfismus φ dostaneme

$$\varphi(\alpha) \equiv \varphi(\widehat{\alpha})\varphi(c_d)^{-|M|/2-d+2} \mod N.$$

Máme $\varphi(c_d) = c_d$ a inverzi již počítáme modulo N. Pokud nelze najít, můžeme získat netriviálního dělitele N z $\gcd(N, c_d)$. Pro implementaci je vhodnější číslem $c_d^{|M|/2+d-2}$ vynásobit druhou odmocninu v konečné kongruenci.

V práci [2] můžeme nalézt chytrý způsob, jak se zcela vyhnout Newtonovu iteračnímu algoritmu. Definujme

$$\widehat{\alpha}_{z_i} \equiv \widehat{\gamma}_{z_i}^{1+z_i(1+\ldots+z_i^{d-2})/2}/N_{z_i} \mod z_i,$$

pro $i = 1, \ldots l$. Kde $\widehat{\gamma}_{z_i} \equiv \widehat{\gamma} \bmod z_i = q_i^{e_i}$ a $N_{z_i} \equiv N(\widehat{\alpha}) \bmod z_i$. Tyto výpočty nejsou příliš náročné. Z rovnosti vypočítaných norem modulo z_i dostáváme

$$\widehat{\alpha}N_{z_i} \equiv \widehat{\alpha}\widehat{\alpha}^{\frac{q_i^d-1}{q_i-1}} \equiv \widehat{\gamma}^{1+z_i(1+\ldots+z_i^{d-2})/2} \equiv \widehat{\gamma}_{z_i}^{1+z_i(1+\ldots+z_i^{d-2})/2} \equiv \widehat{\alpha}_{z_i}N_{z_i} \mod z_i.$$

Tedy protože víme, že N_{z_i} není dělitelné z_i, máme $\widehat{\alpha} \equiv \widehat{\alpha}_{z_i} \bmod z_i$.

3.6.3 Montgomeryho metoda

Poslední metoda vychází z návrhů Peter L. Montgomeryho [24]. Její první implementace je uvedena v [9]. My se zaměříme na upravenou verzi pocházející od Phonga Nguyena [26]. Tato metoda využívá možné rozšíření homomorfismu $\varphi : \mathbb{Z}[\vartheta] \to \mathbb{Z}_N$ na $\mathbb{Q}[\vartheta]$ (toto rozšíření jsme využili i u Couveignesovy metody), tedy

$$\varphi : \mathbb{Q}[\vartheta] \to \mathbb{Z}_N,\ \varphi(\vartheta) \equiv m \bmod N,\ \varphi(\frac{a}{b}) \equiv ab^{-1} \bmod N,\ \frac{a}{b} \in \mathbb{Q}.$$

φ je správně definované jen pokud $\gcd(b, N) = 1$, ale v opačném případě nalezneme netriviálního dělitele N. Definujme několik potřebných pojmů.

Definice 3.14. Nechť $\mathcal{I}$ je grupa lomených ideálů číselného tělesa K a nechť

$$I = \prod_{i=0} P_i^{e_i} \prod_{i=0} Q_i^{-f_i}$$

je prvoideálový rozklad lomeného ideálu $I \in \mathcal{I}$, kde $e_i > 0$, $f_i > 0$. Pak definujme

$$\mathrm{num}(I) = \prod_{i=0} P_i^{e_i} \text{ a } \mathrm{den}(I) = \prod_{i=0} Q_i^{f_i}.$$

Označujeme jako *numenátor* a *denominátor* I. Dále definujme *složitost* I

$$\mathcal{C}(I) = \mathcal{N}(\mathrm{num}(I))\mathcal{N}(\mathrm{den}(I)).$$

Mějme dánu množinu exponentů $S = \{s_j;\ s_j \in \{-1, 1\},\ j \in M\}$ (jak ji volit popíšeme později). Pak dostáváme

$$\varphi(\prod_{j\in M} (a_j + b_j\vartheta_1)^{s_j}) \equiv \varphi(\prod_{j\in M} (a_j + b_j\vartheta_2)^{s_j}) \bmod N.$$

Položme

$$\gamma = \prod_{j\in M} (a_j + b_j\vartheta)^{s_j}.$$

Zřejmě takto zvolené γ je opět kvadrát, navíc známe prvoideálový rozklad lomeného ideálu (γ) v O_K (tento rozklad získáme z faktorizace $F(a_j, -b_j)$, přitom musíme započítat i možnou nemoničnost polynomu f), každý prvoideál v tomto případě potřebujeme reprezentovat i celočíselnou bází (prvky báze vyjadřujeme pomocí pevně zvolené celistvé báze O_K). Vhodnou volbou množiny S můžeme dosáhnout vykrácení některých prvoideálů, tím zmenšit složitost ideálu (γ) a udělat ho jednodušší pro další výpočty. Našim cílem bude najít odmocninu γ. Protože známe rozklad lomeného ideálu (γ), můžeme z něj získat prvoideálový rozklad lomeného ideálu $(\alpha) = \sqrt{(\gamma)}$. Pořád ale platí, že

roznásobení by bylo příliš náročné, proto použijeme iterační postup. V každém kroku vybereme ještě dostatečně malý ideál I_l, střídavě z numerátoru a denominátoru, a nalezneme v něm malé celé algebraické číslo δ_l. Pomocí nich dostaneme odmocninu z γ. Položme tedy

$$\begin{aligned}
\gamma_1 &= \gamma = \prod_{j\in M}(a_j + b_j\vartheta)^{s_j}, \\
v_1 &= 1 \text{ pokud } N(\gamma) > 1, \text{ jinak } -1, \\
G_1 &= \sqrt{(\gamma)}, \\
H_1 &= (1).
\end{aligned}$$

γ_l můžeme považovat za chybu aproximace α v l-tém kroku. v_l určuje zda vybíráme z numerátoru, nebo z denominátoru. G_l je lomený ideál, který aproximuje $\sqrt{(\gamma_l)}$ a H_l je ideál, který odpovídá chybě této aproximace. Pro $(l+1)$-ní krok tedy volme

$$\begin{aligned}
\gamma_{l+1} &= \gamma_l \delta_l^{-2v_l}, \\
v_{l+1} &= -v_l, \\
G_{l+1} &= G_l \left(\frac{I_l}{H_l}\right)^{-v_l}, \\
H_{l+1} &= \frac{(\delta_l)}{I_l}.
\end{aligned}$$

Dříve než popíšeme, jak získat ideál I_l a číslo δ_l, uvedeme vztahy, které indukcí plynou z definice l-tého kroku

$$\gamma = \gamma_l \left(\prod_{j=1}^{l-1} \delta_j^{v_j}\right)^2, \tag{3.13}$$

$$G_l = H_l^{v_l}\sqrt{(\gamma_l)}. \tag{3.14}$$

Tedy $\prod_{j=1}^{l-1}\delta_j^{v_j}$ je aproximace α v l-tém kroku s chybou γ_l (γ_l je zřejmě také čtverec). Postupným iterováním budeme chtít dosáhnout velmi malého γ_l, abychom mohli přímo vypočítat jeho odmocninu.

Ideál I_l vytvoříme tak, že ideál H_l postupně násobíme největším možným prvoideálem, včetně odpovídající mocniny, z $\mathrm{num}(G_l)$, nebo $\mathrm{den}(G_l)$ (závisí na znaménku v_l). Přitom chceme, aby norma I_l nebyla větší než předem zvolená konstanta L_{max}. Protože prvoideály nejsou příliš velké, tento postup funguje dobře. Zároveň můžeme rovnou upravovat prvoideálovou faktorizaci G_{l+1}. V každé iteraci tedy snižujeme $\mathcal{C}(G_l)$, a proto nakonec norma ideálu I_l bude mnohem menší než L_{max}. Dále chceme najít ve vytvořeném ideálu I_l algebraické číslo δ_l tak, aby norma $\mathcal{N}((\delta_l)/I_l)$ byla co nejmenší, k tomu použijeme známý LLL-algoritmus [20] (problém odpovídá nalezení minimálního vektoru).

Na ideál I_l se můžeme dívat i jako na mřížku nad $\mathbb{Z}$, proto můžeme vypočítat jeho LLL-redukovanou bázi (vzhledem k eukleidovské normě). V implementacích reprezentujeme ideály jejich HNF maticí (Hermitovská normální forma, [5]), kde jednotlivé sloupce představují prvky báze a matice je trojúhelníková. To je velmi výhodné pro počítání, ale taková to báze je špatně vyvážená a některé prvky mohou být příliš velké. Proto musíme nejprve použít LLL k její redukci (konstanta L_{max} omezující normu ideálu I_l je volena tak, aby byla co největší, ale přitom aby algoritmus LLL ještě stále velmi rychle vypočítal redukovanou bázi). Nechť tedy $\{\mu_1, \ldots, \mu_d\}$ je redukovaná báze I_l. Položme

$$\xi_i = \frac{c}{|\sigma_i(\gamma_l)|^{s_l/2}} \text{ , kde } c^d = \frac{L_{max}}{\mathcal{N}(I_l)} \sqrt{\frac{|N(\gamma_l)|^{s_l}}{\mathrm{disc}(K)}},$$

pro $i = 1, \ldots, d$ (σ_i jsou různá vnoření K do $\mathbb{C}$). Nyní zavedeme homomorfismus Ω, který nám umožní najít číslo δ_l.

$$\Omega : O_K \to \mathbb{R}^{2d}, \;\; \Omega(\mu) = (v_1, \ldots, v_d, \xi_1\sigma_1(\mu), \ldots, \xi_d\sigma_d(\mu)),$$

pro $\mu = \sum_{i=1}^{d} v_i\vartheta_i$, kde $\{\vartheta_1, \ldots, \vartheta_d\}$ je pevně zvolená celistvá báze O_K. Pokud by polynom f měl i komplexní kořeny, pak nahradíme konjugované páry σ_i, $\bar{\sigma_i}$ za $\sqrt{2} \cdot \Re(\sigma_i)$, $\sqrt{2} \cdot \Im(\sigma_i)$ v definici Ω. Pomocí homomorfismu Ω vytvoříme reálnou matici $2d \times d$ se sloupcovými vektory $\Omega(\mu_i)$. O této matici není těžké dokázat, že horní čtvercová matice má absolutní hodnotu determinantu rovnu $\mathcal{N}(I_l)$ a dolní rovnu L_{max}. Opět použijeme algoritmus LLL a tuto matici, kterou nejprve zaokrouhlíme na celá čísla, zredukujeme. Jako δ_l volíme algebraické číslo, jehož koeficienty v celistvé bázi $\{\vartheta_1, \ldots, \vartheta_d\}$ odpovídající prvním d prvkům v prvním sloupci výsledné redukované matice. Pomocí následujícího tvrzení dokážeme, že se proces nakonec zastaví (důkaz tvrzení lze nalézt v dodatku [26]).

Věta 3.15. *Existuje konstanta C závislá pouze na číselném tělese K taková, že pro algebraické číslo δ_l vypočítané výše uvedeným způsobem platí*

$$|N(\delta_l)| \leq C \cdot \mathcal{N}(I_l).$$

Tedy $\mathcal{N}(H_l) \leq C$.

Iterační proces zastavíme ve chvíli, kdy $\mathcal{C}(G_l) = 1$. Prošli jsme všechny prvoideály z rozkladu lomeného ideálu (γ) a můžeme ukázat, že algebraické číslo γ_l je již dostatečně malé. Tato situace musí nastat, pokud $L_{max} \gg C$. Protože na začátku požadujeme, aby $\mathrm{num}(\sqrt{(\gamma)})$ a $\mathrm{den}(\sqrt{(\gamma)})$ měli přibližně stejně velké normy, platí v každém kroku podle předchozího tvrzení, že $\mathcal{N}(I_l/H_l) \approx L_{max}/C$. Pak ale ze vztahu

$$\mathcal{C}(G_{l+1}) = (\mathcal{N}(I_l/H_l))^{-1}\mathcal{C}(G_l)$$

dostáváme $\mathcal{C}(G_l) \approx (C/L_{max})^{l-1}\mathcal{C}(G_1)$. Tedy pokud L_{max}/C je větší než prvočísla dělící $\mathcal{C}(G_1) = \mathcal{C}(\sqrt{(\gamma)})$, je možné ukázat, že po maximálně $2\lceil \log_2(\mathcal{C}(G_1)) \rceil$

iteracích dostaneme $\mathcal{C}(G_l) = 1$. Pokud by v poslední iteraci bylo $v_l = 1$, pak provedeme ještě jeden krok, ve kterém klademe $I_{l+1} = H_l$.

Po ukončení iterací tedy máme $\mathcal{C}(G_L) = 1$ a $v_L = -1$. Podle vztahu 3.13 nám k nalezení α stačí vypočítat odmocninu z γ_L. Z 3.14 dostáváme $H_L = \sqrt{(\gamma_L)}$ a tedy podle předchozího tvrzení je C^2 horní odhad pro normu algebraického čísla γ_L. Z tohoto odhadu normy ale neplyne, že by celočíselné koeficienty γ_L musely být také malé, avšak v praxi jsou. Proto můžeme odmocninu z γ_L vypočítat klasickým způsobem ([5] algoritmus 3.6.4). Na závěr známe jak δ_j, $1 \leq j \leq L-1$, tak i $\sqrt{\gamma_L}$ a hledanou odmocninu α dostáváme jako

$$\alpha = \sqrt{\gamma_L} \prod_{j=1}^{L-1} \delta_j^{v_j}.$$

Přitom rovnou můžeme využít homomorfismus φ a některé výpočty přesunout do Z_N.

Nyní když již víme, jaké požadavky má splňovat množina exponentů S (vykrácení maximálního počtu prvoideálů, $\mathcal{N}(\mathrm{den}((\gamma))) \approx \mathcal{N}(\mathrm{num}((\gamma)))$ uvedeme možné postupy jak ji získat. Zřejmě jedna z možností je množinu S vybrat náhodně. Další způsob je podobný tomu, jak jsme volili ideál I_l (greedy strategy). Nejprve zvolíme S náhodně a určíme normu denominátoru a numerátoru. Poté postupně procházíme jednotlivé ideály $(a_j + b_j\vartheta)$ a rozhodujeme se, jestli by nebylo vhodnější ideál přesunout (porovnáváme změnu norem denominátoru a numerátoru po přehození). Jako nejlepší způsob volby množiny S se ukazuje metoda založená na algoritmu simulated annealing, podrobnosti lze nalézt v [26].

Složitost Montgomeryho algoritmu je podobně jako všech předchozích velmi špatně teoreticky spočitatelná, ale odhaduje se a v praxi potvrzuje, že tento algoritmus má lineární složitost v $|M|$.

Kapitola 4

Měření

Cílem měření je porovnat efektivnost uvedených prosívacích metod (část 3.3). Budeme se snažit odhalit kolik dobrých relací existuje, kolik se jich skutečně podaří nalézt, případně kolik je jich špatně označeno jako dobré. Kromě typu prosívání má na úspěšnost velký vliv i volba prosívací meze. Podrobněji rozebereme způsoby jejího určení. Z měření také vyplyne, že je vhodné používat u klasického prosívání více mezí pro jeden řádek a jak tyto meze nejlépe volit, abychom dosáhli lepších výsledků než při použití jedné hodnoty.

4.1 Postup měření

K měření efektivnosti mřížového a klasického řádkového prosívání byla použita implementace číselného síta naprogramována na katedře algebry. Zvoleno bylo 100 ciferné číslo

$$N = 298904656703697855014774242300569400616783231539\backslash$$
$$6893622924826984640414223954632312857062757719936727.$$

Výběr řádu byl volen s předpokladem, že se již projeví rozdíly mezi prosívacími metodami. Pomocí první fáze algoritmu byl vybrán polynom

$$\begin{aligned} f(x) \;=\; & 168870730448535948x^4 - 150062462444309572x^3 \\ & -5530126081182449077x^2 - 248446423232885294x \\ & +14749077260448645 75 \end{aligned}$$

s kořenem $m = 364749390441717315494$. Použita byla m-base metoda. Velikosti faktorizačních bází byly voleny pro obě prosívací metody stejné a to $B_I = 1000000$ pro lineární polynom (dále integrální část/báze) a $B_A = 5000000$ pro polynom $f(x)$ (dále algebraická část/báze). Tyto hodnoty se ukázaly být dostatečné. Při použití menších velikostí by bylo potřeba více měření k nalezení relevantního počtu dobrých relací. Prosívací interval pro klasické prosívání

byl $[-I, I]$, kde $I = 5000000$. Volba byla provedena s ohledem na mřížové prosívání, kde prosívací oblast byla $S = [-2^{13}, 2^{13}] \times [1, 2^{13}]$.

Pomocí obou metod jsem kompletně prosel a faktorizoval určitou část uvažované prosívací oblasti. U klasického prosívání jsem zvolil několik reprezentativních hodnot $b \in [1, 50000000]$ (viz tabulka 4.1). Použitá implementace zohledňuje při prosívání velikost cache. Velikost najednou prosívaného bloku je 65536. Tedy jeden řádek u klasického prosívání se rozdělí na 154 bloků. Proto jsem u mřížového prosívání pro zvolená speciální prvočísla Q_k provedl zmíněné prosívání pouze pro bloky ve zvoleném rozmezí 1-154 a 1001-1154 (z celkového počtu 2048). Domnívám se, že uvedené restrikce nemají zásadní vliv na přesnost měření, kromě v další části uvedených a očekáváných případů.

Metoda prosívání	Označení	Hodnoty
klasická	B1	$b = 1, 2$
klasická	B50	$b = 50000, 50001$
klasická	B100	$b = 100000, 100001$
klasická	B250	$b = 250000, 250001$
klasická	B500	$b = 500000, 500001$
klasická	B1000	$b = 1000000, 1000001$
mřížová	Q1	$Q_1 = 5000011$
mřížová	Q2	$Q_2 = 5001071$
mřížová	Q3	$Q_3 = 5154823$

Tabulka 4.1: Použité hodnoty pro prosívání

4.2 Výsledky měření

V prvním případě se zaměříme na celkové počty existujících dobrých relací. Jako dobrou relaci uvažujeme až 2,2-částečně hladkou relaci. Tedy započítáváme relace, pro které zbytek integrální a algebraické normy po vydělení všemi prvočísly z dané faktorizační báze je menší než odpovídající mez. Tyto meze jsou $B_{MI} = (128 \cdot B_I)^2$ a $B_{MA} = (128 \cdot B_A)^2$. Z časových důvodů neprovádíme kompletní faktorizaci zbytků norem. Tady se dopouštíme možné chyby v určení dobré relace, protože může nastat případ, kdy zbytek normy je prvočíslo, nebo jeden z dělitelů je přílíš velký (potom se nejedná o dobrou relaci). Protože toto nastává pro oba případy se stejnou pravděpodobností, můžeme to zanedbat. Dále uvažujeme pouze dvojice (a, b), kde $\gcd(a, b) = 1$. To je důvod, proč u klasického prosívání používáme dva řádky. Pro sudé řádky je až 60 % dvojic soudělných. V tabulce 4.2 vidíme, že mřížkové prosívání poskytuje přibližně dvakrát více dobrých relací než klasické prosívání. To vyplývá z toho, že i když jsou průměrné hodnoty $F(a, b)$ u mřížového prosívání větší než průměrné hodnoty u klasického prosívání, po vydělení speciálním prvočíslem

Q_k jsou již daleko menší. Na tento rozdíl má vliv velikost intervalu u klasického prosívání, pokud by byl mnohem menší, pak by i průměrné hodnoty byly menší. Ale pokud použijeme příliš malý interval, pak hrozí, že vyčerpáme všechny možnosti dříve než nalezneme dostatek relací. Jak bude patrné níže, efektivnost klasického prosívání se snižuje s rostoucím b. Při vyzkoušení menšího intervalu $I = 1000000$ bylo patrné zlepšení efektivnosti ale i její následný propad. Celkově se ale menší interval neukázal jako výhodnější. Výraznou výjimku představuje případ B1, přesněji když $b = 1$. V této situaci jsou všechny dvojice (a, b) nesoudělné a to spolu s malou hodnotou b přispívá k tak výraznému rozdílu. Lze očekávat, že několik prvních řádků bude mít velmi dobré vlastnosti.

Případ	Počet dobrých relací	Výskyt dobrých relací $[10^{-4}]$
B1	47317	23,44
B50	10443	5,17
B100	10265	5,09
B250	9667	4,79
B500	7070	3,50
B1000	5788	2,87
Q1	20050	9,93
Q2	27047	13,40
Q3	25794	12,78

Tabulka 4.2: Celkové počet dobrých relací

Více existujích dobrých relací u mřížového prosívání je jistě velmi dobrá zpráva, ale dokážeme je efektivně identifikovat? Porovnáme tedy efektivnost klasického a mřížového prosívání, přitom se zaměříme na několik věcí:

(i) Kolik dobrých relací identifikujeme?

(ii) Kolik procent z dobrých relací neodhalíme (Nenalezeno)?

(iii) Kolik procent z kandidátů není dobrou relací (Omyl)?

Jak jsme řekli v části 3.3, ke správnému identifikování kandidátů potřebujeme vhodně zvolit prosívací mez. Začneme tedy s odhadem průměrných funkčních hodnot nad prosívacími oblastmi. Nejprve ale zavedeme několik označení pro zjednodušení zápisu. Pro zbytek kapitoly budeme uvažovat pouze logaritmus o základu 4. Nechť

$$H_k(i,j) = F(iv_1^{(k)} + jv_2^{(k)}, -iw_1^{(k)} - jw_2^{(k)}),$$

kde $\{(v_1^{(k)}, w_1^{(k)}), (v_2^{(k)}, w_2^{(k)})\}$ je redukovaná báze mřížky odpovídající speciálnímu prvočíslu Q_k. Nechť $I_{Q_k} = [-2^{13}\sqrt{Q_k}, 2^{13}\sqrt{Q_k}] \times [1, 2^{13}\sqrt{Q_k}]$.

První možností jak spočítat průměrnou hodnotu $F(a,b)$ nad prosívacími oblastmi je použít integrál. Protože funkce $F(x,y)$ může nabývat i záporných hodnot, vypočítáme raději

$$\sqrt{\int_{-I}^{I}(F(x,-b))^2 dx/(2I)}$$

pro klasické prosívání a

$$\sqrt{\iint_{I_S}(H(i,j))^2 didj/(|S|)}$$

pro mřížové prosívání. Jak se ukázalo, tyto odhady nejsou zcela přesné, pokud jako nejlepší variantu považujeme průměrnou hodnotu (modus se ukázal jako méně výhodný). Pro klasické prosívání jsou výsledné logaritmy pro uvažované hodnoty b téměř stejné, ale až pro $b = 1000000$ odpovídá logaritmus naměřeným hodnotám. U mřížkového prosívání by bylo časově nevýhodné vyjádřit funkci $H(i,j)$ a vypočítat uvedený integrál při každé změně speciálního prvočísla. Podobně nevýhodné by bylo počítat tento integrál s funkcí $F(x,y)$ na oblasti I_{Q_k} (hodnoty se moc neliší). Protože různá prvočísla Q_k mají minimální vliv na výsledný integrál je lepší použít jen jedno. Tedy opět by stačilo pouze jednou vypočítat vhodný integrál. V tomto případě je již výsledná hodnota lepší (alespoň v mnou uvažovaných případech) a odchylka je minimální.

Další možností je zkusmo vybrat několik párů (a,b) a vypočítat průměr jejich norem. Tento přístup je v tomto případě zcela přesný jak dokazuje tabulka 4.3. A dá se předpokládat, že pro velká faktorizovaná čísla N bude také úspěšný. U mřížového prosívání se počítá s polynomy $H_k(i,j)$ a výpočet se tedy musí provést při každé změně speciálního prvočísla, to ale není náročné (k uvedeným výpočtům stačilo použít 128 párů).

Případ	Naměřené hodnoty	Naměřený rozptyl	Integrál	Zkusmé výpočty
B1	70	3,00	72	70
B50	70	2,85	72	70
B100	70	2,70	72	70
B250	70	2,27	72	70
B500	71	1,45	72	71
B1000	72	1,13	72	72
Q1	78	2,25	78	78
Q2	76	1,85	78	76
Q3	77	1,78	78	77

Tabulka 4.3: Odhady funkčních hodnot - v logaritmech

Pro integrální část je situace v obou případech mnohem jednodušší a lze

velmi snadno najít dobrý odhad funkčních hodnot, navíc velikost integrální faktorizační báze bývá velmi často nadhodnocena.

Při hodnocení efektivnosti prosívání se zaměříme na několik možností.

a) Známe průměrnou hodnotu logaritmu funkčních hodnot nad prosívacím intervalem.

b) Dále známe i očekávanou chybu při prosívání.

c) Pro každý blok známe průměrnou hodnotu logaritmu funkčních hodnot a očekávanou chybu.

d) Použijeme vypočítanou průměrnou hodnotu logaritmu funkčních hodnot a očekávanou chybu.

Protože naše implementace neprosívá s malými prvočísly a jejich mocninami (prvočísla menší než 128), dochází při prosívání k chybě. Tuto chybu lze jednoduše odhadnout. Pro integrální část dostáváme

$$e_I = \sum_{p<128} \frac{\log(p)}{p-1} = 3,16$$

a pokud r_p je počet kořenů polynomu $f(x)$ modulo p, pak dostáváme pro algebraickou část

$$e_A = \sum_{p<128} \frac{r_p \log(p)}{p-1} = 4,35.$$

Je tedy vhodné při určování meze s touto chybou počítat. V první možnosti používáme přesný odhad průměrných hodnot funkčních hodnot, ale nezapočítáme prosívací chybu. Jak ukazují tabulky s výsledky, při této variantě máme velice malou úspěšnost, protože jsme příliš restriktivní, proto se také nedopustíme mnoha chybných označení. V druhé možnosti již započítáme i chybu a vidíme, že dostáváme nejlepší výsledky pro mřížové prosívání. Třetí možnost počítá ještě s jemnějším určováním průměrných funkčních hodnot. V popisu algoritmu jsme uvedli, že se pro prosívací řádek volí pouze jedna pevná mez. Toto omezení souvisí hlavně s implementací algoritmu, protože je mnohem rychlejší inicializovat prosívací pole jednou hodnotou, než několika různými. Ale pokud máme prosívací interval rozdělený na bloky, tak není nutné používat pouze jednu mez. Avšak jak rychle vhodnou mez vypočítat? Z měření vyplývá, že logaritmus funkční hodnoty v bloku je téměř vždy stejný. Existuje několik výjimek, například když dochází k pozvolné změně v logaritmu, nebo v okolí kořenů funkce. Ale takových bloků je minimum. Tedy jako dobrá volba se jeví logaritmus funkční hodnoty uprostřed bloku. Aby nedošlo k velké chybě v případě bloku s kořenem funkce, je vhodné příliš velké výkyvy hlídat a korigovat. Nadruhou stranu není pravda, že by přesná znalost logaritmu

funkční hodnoty vedla ke stoprocentní úspěšnosti kandidátů. Stále se projevuje chyba prosívání a možné zaokrouhlovací chyby. Tyto chyby můžeme eliminovat pouze v průměru. Z tabulky 4.4 je zřejmé, že tato možnost je nejlepší pro klasické prosívání. Ale ani v tomto případě není celkově úspěšnější než mřížové prosívání. Přitom dosahuje lepších dílčích výsledků. Je lepší v hledání a nedopouští se tolika omylů v označování kandidátů. Pro mřížové prosívání je třetí možnost stejná jako druhá, nedochází k žádným rozdílům. Poslední možnost používá pomocí integrálu vypočítanou průměrnou funkční hodnotu a započítává do meze i chybu prosívání.

Případ	Možnost	Počet kandidátů	Počet dobrých kandidátů	Výskyt dobrých kandidátů $[10^{-4}]$	Nenalezeno [%]	Omyl [%]
B1	a)	8025	7736	3,83	83,65	3,60
B1	b)	37850	23165	11,48	51,04	38,80
B1	c)	42996	29903	14,81	36,80	30,45
B1	d)	24021	17646	8,74	62,71	26,54
B50	a)	1809	1732	0,86	83,41	4,26
B50	b)	8591	5056	2,50	51,58	41,15
B50	c)	9465	6240	3,09	40,25	34,07
B50	d)	5576	3904	1,93	62,62	29,99
B100	a)	2006	1895	0,99	81,54	5,53
B100	b)	10348	5624	2,79	45,21	45,65
B100	c)	10743	6620	3,28	35,51	38,38
B100	d)	6599	4444	2,20	56,71	32,66
B250	a)	1718	1662	0,82	82,81	3,26
B250	b)	9449	5408	2,68	44,06	42,77
B250	c)	9065	5763	2,86	40,38	36,43
B250	d)	6097	4258	2,11	55,95	30,16
B500	a)	1204	1173	0,58	83,42	2,62
B500	b)	7665	4190	2,08	40,74	45,34
B500	c)	8120	4512	2,24	36,19	44,44
B500	d)	6034	3699	1,83	47,67	38,69
B1000	a)	907	901	0,45	84,43	0,66
B1000	b)	5135	3227	1,60	44,25	37,16
B1000	c)	5240	3332	1,65	42,43	36,41
B1000	d)	5135	3227	1,60	44,25	37,16

Tabulka 4.4: Výsledky pro klasické prosívání

Z tabulky 4.4 je patrný pozvolný pokles výskytu dobrých kandidátů, tento pokles koresponduje s poklesem existujících dobrých relací. Přestože prvních

několik (možná několik tisíc) řádků je velmi dobrých, celkově je klasické prosívání výrazně horší než mřížové prosívání, u kterého vidíme pouze větší odchyklu u případu Q1. Avšak klasické prosívání je možné drobnými úpravami, které mají jen minimální vliv na rychlost prosívání, částečně zlepšit jemnější volbou meze.

Případ	Možnost	Počet kandidátů	Počet dobrých kandidátů	Výskyt dobrých kandidátů $[10^{-4}]$	Nenalezeno [%]	Omyl [%]
Q1	a)	2702	2527	1,25	87,40	6,48
Q1	b)	14813	8487	4,20	57,67	42,71
Q1	d)	13266	7926	3,93	60,47	40,25
Q2	a)	4369	4150	2,06	84,66	5,01
Q2	b)	22436	13113	6,50	51,52	41,55
Q2	d)	13186	9241	4,58	65,83	29,92
Q3	a)	3415	3322	1,65	87,12	2,72
Q3	b)	17049	11265	5,58	56,33	33,93
Q3	d)	11126	8294	4,11	67,85	25,45

Tabulka 4.5: Výsledky pro mřížové prosívání

Při reálných výpočtech je ale mnohem důležitější kolik dobrých relací opravdu nalezneme za určitý čas než jejich výskyt. Klasické prosívání díky své rychlosti a jednoduchosti nemusí zkoušet mnoho relací, když nemá zaručeno, že budou dobré. Mnohem lepší strategií je projít velké množství relací a být restriktivnější při určování kandidátů. To platí i pro mřížové prosívání. Abychom dosáhli nejlepších poměrů v odhalování dobrých relací snažíme se, aby mez byla co nejblíže logaritmu funkční hodnoty. Ale pokud se zaměříme na rychlost, chceme mít jistotu, že když začneme náročnou faktorizaci kandidátů, tak bude úspěšná. Výhoda mřížového prosívání oproti klasickému je pouze ve větším výskytu dobrých relací, jinak je mřížové prosívání pomalejší a náročnější. Naše implementace číselného síta optimalizuje hodnotu meze podle nalezených relací. Pokud se snažíme faktorizovat příliš mnoho kandidátů, kteří se ukáží být špatnými, tak je mez zvýšena. Naopak pokud testuje jen pár kadidátů a přitom téměř všechni jsou dobří, tak se snažíme jich nalézt více, tedy mez snížíme. Provedl jsem několik částečných prosívání, abych zjistil reálnou rychlost nalézání dobrých relací. Tabulka 4.6 shrnuje tyto měření. Kromě uvedeného m-base polynomu byla použita i varianta se dvěma kvadratickými polynomy, která by podle části 3.2 mohla být již rychlejší. Ale jak ukazují naměřené hodnoty, tak v tomto případě tomu tak není. Pravděpodobně je potřeba větší číslo N. Pokud bychom hodnoty brali tak, jak jsou uvedeny v tabulce 4.6, odvozovali bychom, že v tomto případě je klasické prosívání rychlejší než mřížové. Ale mřížové prosívání má tuto rychlost téměř konstatní (s měnícím se speciálním

prvočíslem), zatímco u klasického prosívání klesá s rostoucím b v souvislosti s tím, jak klesá výskyt dobrých relací. Po pár desítkách tisíc řádků bude průměrná hodnota podobná jako u mřížového prosívání. Dále se ukazuje, že volit mez pro každý blok u klasického prosívání zvlášť je výhodné. Menší efektivita mřížového prosívání je také způsobena rezervami v naší implementaci mřížového prosívání. Úspěšnost kandidátů, neboli kolik procent z označených kandidátů je skutečně dobrou relací, je překvapivě větší u klasického prosívání. Jak již bylo řečeno, z pohledu celkové rychlosti nalézání relací nemusí být tato úspěšnost příliš velká, jde o vybalancování náročnosti faktorizace a testovaných kandidátů. Proto rozdíl dvou procent nemusí být kritický.

Metoda výběru polynomu	Metoda prosívání	Prosetých polí [10^6/s]	Dobrých relací [rel/s]	Úspěšnost kandidátů	Poznámka
m-base	klasická	46,12	26,5	6,65%	optimalizovaná mez
m-base	klasická	41,36	36,1	6,75%	optimalizovaná mez pro každý blok
m-base	mřížová	34,06	22,8	4,41%	
Montgomery	klasická	39,48	17,4	1,05%	mez pro každý blok

Tabulka 4.6: Průměrné rychlosti hledání relací

4.3 Závěr měření

Z výsledků měření vyplývá, že pro uvažované 100 ciferné číslo je v naší implementaci číselného síta výhodnější použít klasické prosívání. Z první části našeho měření jsme vyvodili (viz tabulka 4.4), že pokud u klasického prosívání volíme mez pro každý prosívací blok zvlášť, pak dosahujeme vyšší efektivnosti. Tento závěr jsme následně potvrdili v dalším měření, kdy byly v implementaci číselného síta provedeny příslušné úpravy. Zjištěné zrychlení bylo v tomto případě až 30% oproti použití jedné hodnoty pro celý řádek.

Naše implementace mřížového prosívání zatím nedosahuje očekávaných výsledků, ale pomocí naměřených dat byla odhalena slabá místa. Zatímco u klasického prosívání je 40 % času potřeba na změny prosívacích bloků, u mřížového prosívání je to až 60 %. Tento čas je využit jen ke správě paměti a přípravě, tedy nedochází k žádné efektivní práci. Obě tyto hodnoty by měly být zredukovány, ale zároveň vysvětlují horší výsledky mřížového prosívání.

Dále jsem pro použité 100 ciferné číslo nepotvrdil, že by pro prosívání bylo výhodnější použít polynomy nalezené Montgomeryho metodou dvou kvadratických polynomů (viz část 3.2.3). Avšak podle naměřených dat jsou již nyní normy určené podle kvadratických polynomů výrazně menší než v případě nelineárního polynomu u m-base metody. Je tedy pravděpodobné, že pro větší

faktorizované číslo N, kdy se tyto rozdíly ještě zvětší, může být použití kvadratických polynomů výhodnější. Také použití efektivního mřížového prosívání by mohlo tuto variantu urychlit, protože bychom dosáhli snížení jedné normy téměř na úroveň normy lineární části u m-base metody. Takové měření jsem neprovedl vzhledem ke zmíněným nedostatkům mřížového prosívání.

Literatura

[1] Aoki K., Franke J., Kleinjung T., Lenstra A. K., Osvik D. A. (2007): *A Kilobit Special Number Field Sieve Factorization*. In Advances in Cryptology – ASIACRYPT'07.

[2] Bernstein D. J., Lenstra A. K., Pomerance C. (1993): *A General Number Field Sieve Implementation* (The Development of the NFS). Springer-Verlar, New York.

[3] Buhler J. P., Lenstra H. W., Pomerance C. (1993): *Factoring Integers with the NFS* (The Development of the NFS). Springer-Verlar, New York.

[4] Cavallar S. (2000): *Strategies for Filtering in the Number Field Sieve*.

[5] Cohen H. (1996): *A Course in Computational Algebraic Number Theory*. Springer-Verlag, New York.

[6] Couveignes J.-M. (1993): *Computing a Square Root for the Number Field Sieve* (The Development of the NFS). Springer-Verlar, New York.

[7] Dickman K. (1930): *On the Frequency of Numbers Containing Prime Factors of a Certain Relative Magnitude*. Ark. Mat., Astronomi och Fysik 22A, 1-14.

[8] Dodson B., Lenstra A. K. (1995): *NFS with Four Large Primes: An Explosive Experiment*. Advances in Cryptology — CRYPTO' 95. Springer-Verlar, New York.

[9] Elkenbracht-Huizing M. (1996): *An Implementation of the Number Field Sieve*. Experimental Mathematics 5, **3**, 231-253.

[10] Elkenbracht-Huizing M. (1996): *A Multiple Polynomial General Number Field Sieve*. Algorithmic Number Theory, 99-114. Springer-Verlar, New York.

[11] Flesch I. (2002): *A New Parallel Approach to the Block Lanczos Algorithm for Finding Nullspaces over GF(2)*. Master's Thesis, Utrecht University.

[12] Golliver R., Lenstra A. K., McCurley K. (1994): *Lattice Sieving and Trial Division*. ANTS'94, Lecture Notes in Comput. Sci., **877**, 18-27.

[13] Gower J., Wagstaff S. S. (2008): *Square Form Factorization*. Mathematics of Computation, **77**, 551-588.

[14] Kleinjung T. (2006): *Cofactorisation Strategies for the Number Field Sieve and an Estimate for the Sieving Step for Factoring 1024-bit Integers*. SHARCS 2006.

[15] Kleinjung T. (2006): *On Polynomial Selection for the General Number Field Sieve*, Mathematics of Computation, **75**, 2037–2047.

[16] Lanczos C. (1950): *An Iteration Method for the Solution of the Eigenvalue Problem of Linear Differential and Integral Operators*. Journal of Research of the National Bureau of Standards, **45**, 255-282.

[17] Lang S. (1994): *Algebraic Number Theory*. Springer-Verlag, New York.

[18] Lenstra H. W. (1987): *Factoring integers with elliptic curves*. Annals of Mathematics, **126**, 649-673.

[19] Lenstra A. K., Manasse M. S. (1991): *Factoring with two Large Primes*. EUROCRYPT '90, **473**, 72-82.

[20] Lenstra A. K., Lenstra H. W., Lovász L. (1982): *Factoring Polynomials with Rational Coefficients*. Mathematische Annalen, **261**, 515-534.

[21] Lenstra A. K., Lenstra H. W., Manasse M. S., Pollard J. M. (1993): *The Number Field Sieve* (The Development of the NFS). Springer-Verlar, New York.

[22] Marcus D. A. (1995): *Number Fields*. Springer-Verlag, New York.

[23] Montgomery P. L. (1995): *A Block Lanczos Algorithm for Finding Dependencies over GF(2)*. EUROCRYPT '95, **921**, 106-120.

[24] Montgomery P. L. (1995): *Square Roots of Products of Algebraic Numbers*. `ftp://ftp.cwi.nl/pub/pmontgom/sqrt.ps.gz`

[25] Murphy B. (1999): *Polynomial Selection for the Number Field Sieve Integer Factorisation Algorithm*. PhD Thesis, The Australian National University.

[26] Nguyen P. (1998): *A Montgomery-Like Root for the Number Field Sieve*. Algorithmic Number Theory - ANTS III, LNCS 1443, 151-168.

[27] Peterson M. (2006): *Parallel Block Lanczos for Solving Large Binary Systems*. Master's thesis.

[28] Pollard J. M. (1988): *Factoring with Cubic Integers* (The Development of the NFS). Springer-Verlar, New York.

[29] Pollard J. M. (1991): *The Lattice Sieve* (The Development of the NFS). Springer-Verlar, New York.

[30] Pomerance C. (1985): *The Quadratic Sieve Factoring Algorithm*, Advance in Cryptology: Proceedings of EUROCRYPT 84 Springer-Verlag, 169-182.

[31] Shamir A., Tromer E. (2003): *Factoring Large Numbers with the TWIRL Device*. Crypto 2003, LNCS 2729, Springer-Verlag, New York.

[32] `http://en.wikipedia.org/wiki/CPU_cache`