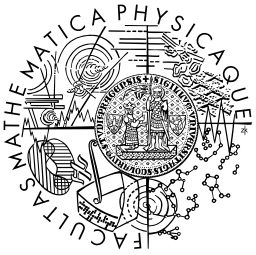

MATEMATICKO-FYZIKÁLNÍ FAKULTA
Univerzita Karlova

BAKALÁŘSKÁ PRÁCE

Dora Todorovová

Malé kořeny celočíselných polynomů více proměnných

Katedra Algebry

Vedoucí bakalářské práce: doc. Mgr. Pavel Příhoda, Ph.D.

Studijní program: Matematika

Studijní obor: Matematika pro informační technologie

Praha 2020

Prohlašuji, že jsem tuto bakalářskou práci vypracoval(a) samostatně a výhradně s použitím citovaných pramenů, literatury a dalších odborných zdrojů. Tato práce nebyla využita k získání jiného nebo stejného titulu.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona v platném znění, zejména skutečnost, že Univerzita Karlova má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle §60 odst. 1 autorského zákona.

V ..... dne ....

Podpis autora

Chtěla bych poděkovat vedoucímu mé práce doc. Mgr. Pavlu Příhodovi, Ph.D. za ochotu, trpělivost, čas strávený konzultacemi a za rady, které mi pomohly s psaním této práce.

Název práce: Malé kořeny celočíselných polynomů více proměnných

Autor: Dora Todorovová

Katedra: Katedra Algebry

Vedoucí bakalářské práce: doc. Mgr. Pavel Příhoda, Ph.D., Katedra algebry

Abstrakt: Tato práce se zabývá Coppersmithovou metodou na hledání kořenů celočíselných polynomů modulo N, která je založena na redukci báze mřížky. Nejprve zadefinujeme pojem mřížka a ukážeme si LLL algoritmus ve zjednodušené podobě. Dále popíšeme Coppersmithovu metodu a tvrzení, která se k ní vztahují. Následně ukážeme řešený příklad z článku od D. Boneh a G. Durfee a obecný postup z článku od E. Jochemsz a A. May, který doplníme o několik důkazů navíc. V poslední kapitole vyřešíme příklady pomocí obecného postupu.

Klíčová slova: kořeny, Coppersmith, mřížky, LLL

Title: Small roots of multivariate polynomials with integral coefficients

Author: Dora Todorovová

Department: Department of Algebra

Supervisor: doc. Mgr. Pavel Příhoda, Ph.D., Department of Algebra

Abstract: This thesis focuses on the Coppersmith method for finding roots of modular polynomials. The method is based on the base reduction of a lattice. Firstly, we define a lattice and show a simplified form of the LLL algorithm. Then we describe the Coppersmith method and related theorems. Furthermore, we introduce a solved example from the article written by D. Boneh and G. Durfee. The general form of the method from the article written by E. Jochemsz and A. May snd we add some proofs. In the last chapter we solve examples using the method in general form.

Keywords: roots, Coppersmith, lattices, LLL

Obsah

Úvod

Coppersmithova metoda nám pomáhá najít kořeny polynomů, které jsou modulo nějaké N. Je založena na redukci báze mřížky a využití můžeme například najít při útoku na RSA (to si ukážeme na příkladu z Boneh a Durfee (2000)). Budeme vycházet ze dvou článků první od Boneh a Durfee (2000) a druhý od Jochemsz a May (2006).
Článek od Boneh a Durfee (2000) se zabývá útokem na RSA a popisuje nejprve základní metodu a poté metodu pro lepší odhad. Ukážeme pouze první postup jen pro představu využití Coppersmithova algoritmu.
Článek od Jochemsz a May (2006) se zabývá obecným postupem na hledání kořenů. Vysvětluje se zde, jak volit množiny monočlenů a jak pomocí nich sestavit mřížku, abychom dostali požadovaný tvar matice mřížky a mohli s ní dále pracovat.
V první kapitole zadefinujeme pojem mřížky, ukážeme si zjednodušený LLL algoritmus (budeme vycházet z učebnice Počítačová algebra od Stanovský (2017)) a poté to vše použijeme k vysvětlení Coppersmithovy metody.
Druhá kapitola začíná příkladem z článku od Boneh a Durfee (2000), který nám dá představu o tom, jak přesněji tento algoritmus funguje. Nemáme zde úplné řešení, ale pouze zjednodušenou verzi. Poté ukážeme obecný postup z článku od Jochemsz a May (2006) a doplníme ho o několik důkazů navíc a popíšeme důkladněji závěr. Budeme se zabývat pouze základní formou, rozšířenou strategii pouze popíšeme, ale nebudeme s ní více pracovat.
Třetí kapitola se zabývá řešením příkladů. Máme zde podrobněji popsáno jak vyřešit příklad $f = x+ay+bz+c \pmod N$. Jako poslední ukážeme jak se liší odhad přes základní verzi od odhadu, který získáme rozšířenou strategií pro polynom jehož výsledek je známý.

1. Úvod do práce s mřížkami

Předtím, než se pustíme do hledání kořenů rovnic, zadefinujeme si základní pojmy a značení, které budeme používat. Celý náš proces vychází z Coppersmithovy metody, která je založena na redukci mřížky. Prvním naším krokem tedy bude vysvětlení co to je mřížka a jak se s ní počítá. Dále si také ukážeme LLL-algoritmus, na kterém stojí celé naše řešení. Základní definice a věty jsou brány z učebnice Počítačová algebra (Stanovský, 2017).

1.1 Mřížky a Gram-Schmidtův proces

Zadefinujeme pojem mřížka a ukážeme si její základní vlastnosti.

Definice 1. *Podmnožinu* $\boldsymbol{L} \subseteq \mathbb{R}^n$ *nazveme mřížkou, pokud existuje báze* $\mathbf{b_1},...,\mathbf{b_n}$ *vektorového prostoru* $\mathbb{R}^n$ *taková, že*

$$\boldsymbol{L} = \sum_{i=1}^{n} \mathbb{Z}\mathbf{b_i} = \{x_1\mathbf{b_1} + x_2\mathbf{b_2} + ... + x_n\mathbf{b_n} | x_i \in \mathbb{Z}, \forall i = 1,...,n\}.$$

Vektory $\mathbf{b_1},\mathbf{b_2},...,\mathbf{b_n} \in \mathbb{R}^n$ *nazýváme bází mřížky* $\boldsymbol{L}$*. Mřížku* $\boldsymbol{L}$ *nazveme celočíselnou, pokud* $\boldsymbol{L} \subseteq \mathbb{Z}^n$*.*

Abychom si ulehčili zápis, budeme značit matici, jejíž sloupce jsou tvořeny vektory $\mathbf{b_1},\mathbf{b_2},...,\mathbf{b_k}$ jako $(\mathbf{b_1}|\mathbf{b_2}|...|\mathbf{b_k})$.

Dále si připomeneme Gram-Schmidtův ortogonalizační proces. Mějme vektory $\mathbf{b_1},...,\mathbf{b_k}$ pak $\mathbf{b_1^*},...,\mathbf{b_k^*}$ je posloupnost, kterou dostaneme použitím Gram-Schmidtova algoritmu. A tedy platí

$$\mathbf{b_i^*} = \mathbf{b_i} - \sum_{j=1}^{i-1} \mu_{ij}\mathbf{b_j^*} \quad \text{a} \quad \mu_{ij} = \frac{\mathbf{b_i}^\top \cdot \mathbf{b_j^*}}{\mathbf{b_j^*}^\top \cdot \mathbf{b_j^*}}.$$

Definice 2. *Pro mřížku* $\boldsymbol{L}$ *s bází* $\mathbf{u_1},\mathbf{u_2},...,\mathbf{u_n}$ *definujeme* $u^*_{min} := min_i\|\mathbf{u_i^*}\|$*.*

Tvrzení 1. *Pro* u^*_{min} *z definice 2 platí, že to je dolní odhad na délku nejkratšího nenulového vektoru v* $\boldsymbol{L}$*.*

Důkaz. Označme $\mathbf{U}$ bázi a $\mathbf{U}^*$ výstup Gram-Schmidtova procesu na $\mathbf{U}$. Nahlédneme, co platí pro obecný vektor $\mathbf{Ux}$ mřížky $\boldsymbol{L}$, kde $\mathbf{x} \in \mathbb{Z^n} \setminus \{\mathbf{0}\}$ a k je největší takové, že $x_k \neq 0$. Chceme dokázat

$$\|\mathbf{Ux}\| \geq \|\mathbf{u_k^*}\| \geq min_i\|\mathbf{u_i^*}\|.$$

Vezmeme skalární součin $\mathbf{Ux}$ a $\mathbf{u_k^*}$ a dostaneme

$$\langle \mathbf{Ux}, \mathbf{u_k^*} \rangle = \sum_{i \leq k} \langle \mathbf{u_i}x_i, \mathbf{u_k^*} \rangle = x_k \langle \mathbf{u_k}, \mathbf{u_k^*} \rangle = x_k\|\mathbf{u_k^*}\|^2.$$

Využili jsme zde ortogonalitu $\mathbf{u_k^*}$ a $\mathbf{u_i}$ pro $i < k$.
Dále použijeme Cauchyho-Schwartzovu větu a dostaneme

$$\|\mathbf{Ux}\| \cdot \|\mathbf{u_k^*}\| \geq |\langle \mathbf{Ux}, \mathbf{u_k^*} \rangle| \geq |x_k|\|\mathbf{u_k^*}\|^2.$$

Využitím $|x_k| \geq 1$ a vydělením $\|\mathbf{u_k^*}\|$ dostaneme $\|\mathbf{Ux}\| \geq \|\mathbf{u_k^*}\|$.

□

Definice 3. *Mějme mřížku* $\boldsymbol{L}$ *tvořenou bází* $\mathbf{b_1},...,\mathbf{b_n}$. *Jejím determinantem rozumíme*

$$det(\boldsymbol{L}) := |det(\mathbf{b_1}|\mathbf{b_2}|...|\mathbf{b_n})|.$$

Vidíme, že determinant udává objem rovnoběžnostěnu, který je určen bázovými vektory. Následující Tvrzení říká, že objem tohoto objektu není větší než kvádru o stejně dlouhých hranách.

Tvrzení 2. *Determinant mřížky nezávisí na volbě báze a platí*

$$det(\boldsymbol{L}) = \|\mathbf{b_1^*}\|\|\mathbf{b_2^*}\|\cdots\|\mathbf{b_n^*}\| \leq \|\mathbf{b_1}\|\|\mathbf{b_2}\|\cdots\|\mathbf{b_n}\|.$$

Důkaz. Viz. učebnice Počítačová algebra (Stanovský, 2017).

□

1.2 LLL algoritmus

Definice 4. *Báze* $\mathbf{b_1},...,\mathbf{b_n}$ *mřížky* $\boldsymbol{L}$ *se nazývá LLL-redukovaná, pokud*

(1) $|\mu_{ij}| \leq \frac{1}{2}$ *pro všechna* $1 \leq j < i \leq n$

(2) $\|\mathbf{b_i^*}\|^2 \geq (\frac{3}{4} - \mu_{ii-1}^2)\|\mathbf{b_{i-1}^*}\|^2$ *pro všechna* $1 < i \leq n$.

Poznámka. Z (1) a (2) vidíme, že $\|\mathbf{b_i^*}\|^2 \geq \frac{1}{2}\|\mathbf{b_{i-1}^*}\|^2$ z čehož indukcí dostaneme, že $\|\mathbf{b_j^*}\|^2 \geq 2^{i-j}\|\mathbf{b_i^*}\|^2$ pro všechna $1 \leq i \leq j \leq n$.

Vstupem LLL algoritmu je báze $\mathbf{b_1},...,\mathbf{b_n}$ mřížky $\boldsymbol{L} \subseteq \mathbb{Z}^n$. Na výstupu dostaneme LLL-redukovanou bázi mřížky. Zde máme zjednodušenou verzi, která se dá dále optimalizovat.

LLL algoritmus

1: Gram-Schmidtovou ortogonalizací spočti $\mathbf{b_1^*},...,\mathbf{b_n^*}$ a μ_{ij}, $1 \leq j < i \leq n$
2: **for** $i = 2,...,n$ **do**
3: **for** $j = i-1,...,1$ **do**
4: $x := \lfloor \mu_{ij} \rceil$
5: $\mathbf{b_i} := \mathbf{b_i} - x\mathbf{b_j}$
6: $\mu_{ij} := \mu_{ij} - x$
7: **for** $l = 1,...,j-1$ **do** $\mu_{il} := \mu_{il} - x\mu_{jl}$
8: **for** $i = 2,...,n$ **do**
9: **if** $\|\mathbf{b_i^*}\|^2 < (\frac{3}{4} - \mu_{i\,i-1}^2)\|\mathbf{b_{i-1}^*}\|^2$ **then**
10: prohoď $\mathbf{b_i}$ a $\mathbf{b_{i-1}}$
11: **goto 1.**
12: **return** $\mathbf{b_1},\mathbf{b_2},...,\mathbf{b_n}$

Časová složitost LLL algoritmu v této základní podobě je polynomiální vzhledem k velikosti vstupu. Provádí se zde polynomiálně mnoho aritmetických operací s racionálními čísly (jejichž jmenovatelé a čitatelé jsou polynomiálně omezeny - viz učebnice Počítačová algebra (Stanovský, 2017)).
Pokud bychom algoritmus optimalizovali dostali bychom se k časové složitosti $\mathcal{O}(n^6(logB)^3))$ (resp.$\mathcal{O}(n^{5+\epsilon}(logB)^{2+\epsilon})$ pokud použijeme rychlé násobení), kde B je horní odhad norem vstupních vektorů.

Tvrzení 3. *Pro libovolnou LLL-redukovanou bázi* $\mathbf{b_1},...,\mathbf{b_n}$ *mřížky* $\boldsymbol{L}$ *a každé* $1 \leq i \leq j \leq n$ *platí*

$$\|\mathbf{b_i}\|^2 \leq 2^{j-1}\|\mathbf{b_j^*}\|^2 \quad a \quad \|\mathbf{b_1}\| \leq 2^{\frac{n-1}{4}} \sqrt[n]{det(\mathbf{L})}.$$

Důkaz. První nerovnost vychází z toho, že

$$\mathbf{b_i} = \mathbf{b_i^*} + \sum_{k=1}^{i-1} \mu_{i,k}\mathbf{b_k^*}$$

a jelikož jsou na sebe vektory $\mathbf{b_1^*},...,\mathbf{b_k^*}$ kolmé, dostaneme

$$\|\mathbf{b_i}\|^2 = \|\mathbf{b_i^*}\|^2 + \sum_{k=1}^{i-1} \mu_{i,k}^2\|\mathbf{b_k^*}\|^2.$$

Dále pomocí (1) z definice a poznámky výše dostaneme

$$\begin{aligned}\|\mathbf{b_i}\|^2 &\leq \|\mathbf{b_i^*}\|^2 + \sum_{k=1}^{i-1} \frac{1}{4}2^{i-k}\|\mathbf{b_i^*}\|^2 = \|\mathbf{b_i^*}\|^2(1 + \frac{1}{4}(2^i - 2)) \\ &= \|\mathbf{b_i^*}\|^2(\frac{1}{2} + 2^{i-2}) \leq 2^{i-1}\|\mathbf{b_i^*}\|^2 \leq 2^{i-1}2^{j-i}\|\mathbf{b_j^*}\|^2 = 2^{j-1}\|\mathbf{b_j^*}\|^2.\end{aligned}$$

Druhá nerovnost nám vychází z první, neboli vezmeme $\|\mathbf{b_1}\|^2 \leq 2^{i-1}\|\mathbf{b_i^*}\|^2$ a vynásobíme ji přes všechna $1 \leq i \leq n$ a dostaneme

$$\|\mathbf{b_1}\|^{2n} \leq \prod_{i=1}^{n} 2^{i-1}\|\mathbf{b_i^*}\|^2 = 2^{\frac{n(n-1)}{2}} det(\mathbf{L})^2.$$

Požadovaný výsledek dostaneme odmocněním.

□

Následující tvrzení a jeho důsledek jsou brány z článku Jochemsz a May (2006).

Tvrzení 4. *Nechť* $\boldsymbol{L}$ *je mřížka s bází* $\mathbf{u_1},...,\mathbf{u_n}$ *a nechť* $\mathbf{b_1},...,\mathbf{b_n}$ *je výstup LLL algoritmu pro vstup* $\mathbf{u_1},...,\mathbf{u_n}$ *a nechť* $u_{min}^* \geq 1$. *Pak pro* $1 \leq i \leq n$ *platí*

$$\|\mathbf{b_i}\| \leq 2^{\frac{n(n-1)}{4(n+1-i)}} det(L)^{\frac{1}{n+1-i}}$$

Důkaz. Použijeme část důkazu z Tvrzení 3 neboli platí nám

$$\|\mathbf{b_i}\|^2 \leq 2^{j-1}\|\mathbf{b_j^*}\|^2.$$

Tak vezmeme tuto nerovnost přes všechna $i \leq j \leq n$ a dostaneme

$$\|\mathbf{b_i}\|^{2(n+1-i)} \leq \prod_{j=i}^{n} 2^{j-1}\|\mathbf{b_j^*}\|^2 \leq 2^{\frac{n(n-1)}{2}} det(\mathbf{L})^2.$$

Odmocněním dostaneme požadovaný vzorec.

□

Důsledek. Nechť L je mřížka tvořená $\mathbf{u_1},...,\mathbf{u_n}$ a nechť $\mathbf{b_1},...,\mathbf{b_n}$ je výstup LLL algoritmu pro vstup $\mathbf{u_1},...,\mathbf{u_n}$ a nechť $u_{min}^* \geq 1$. Pak

$$\|\mathbf{b_2}\| \leq 2^{\frac{n}{4}} det(L)^{\frac{1}{n-1}}.$$

1.3 Coppersmithova metoda

Nejprve vysvětlíme základní myšlenku Coppersmithovy metody a pak dokážeme Tvrzení, která budeme potřebovat k našemu výpočtu.

Jde vlastně o postup jak najít malé kořeny celočíselných polynomů, které jsou modulo nějaké $N \in \mathbb{N}$. Například mějme polynom $F(x)$ a chceme najít $x_0 \in \mathbb{Z}$ tak, že $F(x_0) \equiv 0 \ (mod\ N)$. Myšlenka je taková, že najdeme polynom $f(x)$, který je nějakým způsobem odvozený z $F(x)$, má stejný kořen $x_0 \ (mod\ N)$ a zároveň má malé koeficienty. Pokud takový polynom najdeme, může se stát, že bude platit $f(x_0) = 0$ i nad celými čísly. To nám poté usnadní hledání x_0, jelikož budeme moci využít nějaký standartní algoritmus pro hledání kořenů celočíselných polynomů.

My ale budeme řešit polynomy více proměnných, a tedy budeme hledat x_0 a y_0 pro nějaký polynom $F(x,y)$

Definice 5. *Mějme polynom* $f(x,y) = \sum_{i,j} a_{i,j}x^i y^j$*, pak definujeme jeho normu jako* $\|f(x,y)\| := \sqrt{\sum_{i,j} |a_{i,j}^2|}$.

K řešení hledání kořenů budeme potřebovat následující Tvrzení, které nám říká, že pokud polynom $f(x,y)$ má malou normu, potom jeho každý malý kořen modulo nějaké $N \in \mathbb{N}$ je i jeho kořenem nad celými čísly. Je to Tvrzení, se kterým přišel Nick Howgrave-Graham v roce 1998.

Tvrzení 5 (HG98)**.** *Nechť* $f(x,y) \in \mathbb{Z}[x,y]$ *je polynom, který je součtem nejvýše* w *termů případně lineární kombinace maximálně* w *mnohočlenů. A nechť existují* X,Y *reálná tak, že:*

(1) $f(x_0,y_0) = 0 \ (mod\ N^m)$ *pro* $m \in \mathbb{N}$ *, kde* $|x_0| < X, |y_0| < Y$

(2) $\|f(xX,yY)\| < N^m/\sqrt{w}$.

Pak $f(x_0,y_0) = 0$ *platí nad celými čísly.*

Důkaz. Všimněme si, že

$$\begin{aligned} |f(x_0,y_0)| = |\sum a_{i,j}x_0^i y_0^j| = |\sum a_{i,j}X^iY^j(\frac{x_0}{X})^i(\frac{y_0}{Y})^j| \leq \\ \leq \sum |a_{i,j}X^iY^j(\frac{x_0}{X})^i(\frac{y_0}{Y})^j| \leq \sum |a_{i,j}X^iY^j| \leq \\ \leq \sqrt{w}\|f(xX,yY)\| < N^m \end{aligned}$$

Ale jelikož $f(x_0,y_0) \equiv 0 \ (mod\ N^m)$ dostaneme, že $f(x_0,y_0) = 0$

□

Definice 6. *Mějme polynom* $f(x_1,...,x_n) = \sum_{i_1,...,i_n} a_{i_1,...,i_n}x_1^{i_1} \cdots x_n^{i_n}$*, pak definujeme jeho normu jako* $\|f(x_1,...,x_n)\| := \sqrt{\sum_{i_1,...,i_n} |a_{i_1,...,i_n}^2|}$.

Tvrzení 6 (HG98 - pro více proměnných)**.** *Nechť* $f(x_1,...,x_n) \in \mathbb{Z}[x_1,...,x_n]$ *je polynom, který je součtem nejvýše* w *termů případně lineární kombinace maximálně* w *mnohočlenů. A nechť existují* $X_1,...,X_n$ *reálná tak, že:*

(1) $f(x_1^{(0)},...,x_n^{(0)}) \equiv 0 \ (mod\ N^m)$ *pro* $m \in \mathbb{N}$ *a* $|x_1^{(0)}| < X_1,...,x_n^{(0)} < X_n$ *a*
(2)$\|f(x_1X_1,...,x_nX_n)\| < N^m/\sqrt{w}$.

Pak $f(x_1^{(0)},...,x_n^{(0)}) = 0$ *platí nad celými čísly.*

Náš postup podle Coppersmithovy metody tedy bude takový, že nagenerujeme více polynomů, které budou mít stejný kořen jako náš $F(x,y)$ a vytvoříme z nich mřížku. Dále z ní určíme LLL redukovanou bázi a její nejkratší vektor, který bude naším hledaným $f(x,y)$ a pro který bude platit $f(x_0,y_0) = 0$ nad celými čísly. Poté najdeme další polynomy, abychom mohli vypočítat rezultanty a získat hledaný kořen.

2. Hledání malých kořenů

Nejprve si ukážeme postup na konkrétním příkladu a poté ho ukážeme v obecnější formě. Budeme postupovat Coppersmithovou metodou a tedy používat redukci báze mřížky.

2.1 Řešený příklad

Následující příklad je útok na RSA z článku Boneh a Durfee (2000). Motivace je taková, že máme dvojici veřejných klíčů $\langle N,e\rangle$ tak, že $N = pq$ a předpokládáme $NSD(p-1,q-1) = 2$ a $e = N^{\alpha}$, kde α je velmi blízko 1. Poté pro soukromý klíč d platí $e \cdot d \equiv 1 \ (mod \ \frac{\varphi(N)}{2})$, kde $\varphi(N) = N - p - q + 1$. Existuje zde tedy nějaké k tak, že

$$ed + k\Big(\frac{N+1}{2} - \frac{p+q}{2}\Big) = 1.$$

Označíme $s = -\frac{p+q}{2}$, $A = \frac{N+1}{2}$ z čehož dostaneme:

$$k(A+s) \equiv 1 \ (mod \ e).$$

V RSA zpravidla platí,že p i q jsou menší než $2\sqrt{N}$. Odhadneme tedy $|s|$ jako:

$$|s| = \frac{p+q}{2} < \frac{4\sqrt{N}}{2} = 2\sqrt{N} = 2e^{1/(2\alpha)}$$

Pokud vezmeme $\alpha \approx 1$ a budeme ignorovat malé konstanty dostaneme odhad $|s| < e^{0,5}$. My tento příklad ale budeme řešit obecně bez spojitosti k RSA. Přeznačíme tedy proměnné a modula tak, aby se nám s tím lépe pracovalo. Máme rovnici $F(x,y) = x(A+y) - 1$ pro nějaké $A \in \mathbb{Z}$ a chceme najít $x_0,y_0 \in \mathbb{Z}$ tak, aby platilo:

$$F(x_0,y_0) \equiv 0 \ (mod \ N) \text{ pro dané } N \in \mathbb{N}$$

a zároveň $|x_0| < N^{\delta}$ a $|y_0| < N^{0,5}$.
Ukážeme si algoritmus, který funguje pro $\delta < \frac{7}{3} - \frac{1}{3}\sqrt{7} \approx 0{,}284$.
Pro jednoduchost mějme horní odhady našich kořenů jako $X = N^{\delta}$ a $Y = N^{0,5}$. Máme tedy $F(x,y) \in \mathbb{Z}[x,y]$ a Tvrzení 5 nám říká, že bychom měli hledat další polynom s malou normou, který má (x_0,y_0) jako kořeny modulo N^m. Abychom to mohli udělat, zadefinujeme si polynomy:

$$g_{i,k}(x,y) := x^i F^k(x,y) N^{m-k} \text{ a } h_{j,k}(x,y) := y^j F^k(x,y) N^{m-k}.$$

Říkáme, že $g_{i,k}$ polynomy jsou $x-$posunutí a $h_{j,k}$ polynomy jsou $y-$posunutí. Snadno nahlédneme, že (x_0,y_0) jsou kořeny všech těchto polynomů modulo N^m pro $k = 0,...,m$. Chceme najít celočíselnou lineární kombinaci $g_{i,k}(xX,yY)$ a $h_{j,k}(xX,yY)$ s malou normou.
Abychom toho dosáhli, vytvoříme si mřížku tvořenou odpovídajícími vektory koeficientů, abychom mohli použít LLL algoritmus. Pro toto provedení musíme ukázat, že má malý determinant (podle Tvrzení 3).

Mřížku vytvoříme pomocí daného m a vektorů koeficientů polynomů pro $k = 0,...,m$. Pro každé k použijeme $g_{i,k}(xX,yY)$ pro $i = 0,...,m-k$ a $h_{j,k}(xX,yY)$ pro $j = 1,...,t$, kde t je parametr, který určíme později. Sestavíme ji tak, že si sloupce rozdělíme do bloků, podle x^i pro $i = 0,...m$, ke kterému budeme přidávat mocniny y. Pro y posunutí budeme mít bloky $y^j, xy^{j+1}, ..., x^m y^{m+j}$, kde $j = 1,...,t$. Řádky uspořádáme podle vedoucího monočlenu (viz 8) v $g_{i,k}$, který je roven $x^{i+k}y^k$. Tedy pro $m = 3$ to bude $g_{0,0}, g_{1,0}, g_{0,1}, g_{2,0}, g_{1,1}, g_{0,2}$ atd. Díky tomuto a uspořádání sloupců podle velikosti monočlenů dostaneme dolní trojúhelníkovou matici. Na diagonále budou právě tyto monočleny krát nějaká mocnina N. Pro y posunutí to uděláme analogicky s polynomem $h_{j,k}(x,y) := y^j F^k(x,y) N^{m-k}$ s tím, že tentokrát je vedoucí člen roven $x^k y^{j+k}$, a tedy pro $m = 3, t = 1$ to bude $h_{1,0}, h_{1,1}, h_{1,2}, h_{1,3}$. Koukneme se, jak tato mřížka bude vypadat například pro $m = 3, t = 1$:

	1	x	xy	x^2	x^2y	x^2y^2	x^3	x^3y	x^3y^2	x^3y^3	y	xy^2	x^2y^3	x^3y^4
N^3	N^3													
xN^3		N^3X												
FN^2	-	-	N^2XY											
x^2N^3				N^3X^2										
xFN^2		-		-	N^2X^2Y									
F^2N	-	-	-	-	-	NX^2Y^2								
x^3N^3							N^3X^3							
x^2FN^2				-			-	N^2X^3Y						
xF^2N		-		-	-		-	-	NX^3Y^2					
F^3	-	-	-	-	-	-	-	-	-	X^3Y^3				
yN^3											N^3Y			
yFN^2			-								-	N^2XY^2		
yF^2N			-		-	-					-	-	NX^2Y^3	
yF^3			-			-		-	-	-	-	-	-	X^3Y^4

Vidíme, že je tvořena dolní trojúhelníkovou maticí, kde "-"značí nenulové hodnoty, které nás ale nezajímají, jelikož determinant nám závisí pouze na prvcích na diagonále, které snadno zjistíme. Každý blok řádků (tak jak jsou rozdělené výše) odpovídá určité mocnině x. Poslední blok je výsledkem $y-$posunutí. V naší vzorové mřížce je pro $t = 1$ máme pouze jedno posunutí o y. Později uvidíme, že právě tato $y-$posunutí jsou hlavním důvodem našeho lepšího výsledku.

Nyní si spočteme determinant naší mřížky L. Začneme determinantem podmatice tvořené x- posunutími. Jelikož máme dolní trojúhelníkovou matici, bude to součin prvků na diagonále. Vidíme, že zde jsou pouze mocniny X,Y a N. Máme to rozdělené do bloků podle řádků. Pro X platí, že každá mocnina pro $i = 1,...,m$ je zde $i+1$ krát. Tedy dostaneme, že X zde máme v mocnině

$$\sum_{i=1}^{m} i(i+1) = \frac{m(m^2+3m+2)}{3} = \frac{m(m+1)(m+2)}{3}.$$

N je zde stejně jako X. Zbývá Y, které vidíme má o polovinu méně výskytů. A tedy:

$$det_x = N^{m(m+1)(m+2)/3} \cdot X^{m(m+1)(m+2)/3} \cdot Y^{m(m+1)(m+2)/6}.$$

Vynechali jsme $y-$posunutí neboli vzali jsme $t = 0$.
Vidíme, že například pro naši mřížku, tedy pro $m = 3$ a s vynecháním posledního bloku dostaneme determinant $N^{20}X^{20}Y^{10}$. A dosadíme-li $X = N^{\delta}, Y = N^{0,5}$ tak dostaneme:

$$det_x = N^{m(m+1)(m+2)(5+4\delta)/12} = N^{\frac{5+4\delta}{12}m^3+o(m^3)}.$$

Všimněme si, že dimenze naší podmatice je $w = (m+1)(m+2)/2$, tedy w-tá odmocnina našeho determinantu je $D_x = N^{m(5+4\delta)/6}$. Abychom mohli použít Tvrzení 5. musíme mít $D_x < N^m$. Z toho dostaneme

$$\begin{aligned} \frac{m(5+4\delta)}{6} &< m \\ m(5+4\delta) &< 6m \\ 5+4\delta &< 6 \\ 4\delta &< 1 \Rightarrow \delta < 0{,}25. \end{aligned}$$

Což je přesně výsledek, s kterým přišel Wiener (1990). Vidíme, že mřížka složená pouze z $x-$posunutí nám nepomůže zlepšit tento výsledek. Abychom ho mohli vylepšit, musíme zapojit i $y-$posunutí. Koukneme se jak vypadá determinant podmatice $y-$posunutí. Tedy chceme součin prvků na diagonále posledního bloku. X i N zde máme v mocnině

$$t \cdot \sum_{i=1}^{m} i = \frac{tm(m+1)}{2}.$$

Y má mocninu

$$\sum_{j=1}^{t} j + (j+1) + ...(j+m) = \sum_{j=1}^{t}\sum_{i=0}^{m}(j+i) = \frac{t(m+1)(m+t+1)}{2}.$$

Z toho dostáváme:

$$det_y = N^{tm(m+1)/2} \cdot X^{tm(m+1)/2} \cdot Y^{t(m+1)(m+t+1)/2}.$$

Opět dosadíme za X a Y:

$$det_y = N^{tm(m+1)(1+\delta)/2 + t(m+1)(m+t+1)/4} = N^{\frac{3+2\delta}{4}tm^2 + \frac{mt^2}{4} + o(tm^2)}.$$

Z toho dostáváme, že determinant celé matice je $det(L) = det_x \cdot det_y$ a její dimenze je $w = (m+1)(m+2)/2 + t(m+1)$.
Nyní budeme chtít použít Tvrzení 5 na nejkratší vektor v LLL-redukované bázi L. Abychom to tak mohli udělat, musíme se ujistit, že norma b_1 je menší než $N^m/\sqrt{w}$. Když to zkombinujeme s Tvrzením 3 musíme to řešit pro největší hodnotu δ splňující

$$det(L) < N^{mw}/\gamma,$$

kde $\gamma = (w2^w)^{w/2}$. Jelikož dimenze w je pouze funkcí m a t, ale ne N pak pro realistické volby t a m (tj. aby doběhl LLL algoritmus) a velká N můžeme γ nahradit nějakou konstantou, kterou ve zde uvedeném nepřesném postupu zanedbáme. Následující výpočet není úplně přesný, ignoruje členy malých řádů. Podrobný postup lze nalézt v článku Boneh a Durfee (2000), přesněji v Appendixu A tohoto článku. Dostáváme:

$$w = \frac{m^2}{2} + tm + o(m^2)$$

$$det(L) = N^{\frac{5+4\delta}{12}m^3+\frac{3+2\delta}{4}tm^2+\frac{mt^2}{4}+o(m^3)}$$

Abychom měli splněnou podmínku $det(L) < N^{mw}$ musí platit:

$$\frac{5+4\delta}{12}m^3 + \frac{3+2\delta}{4}tm^2 + \frac{mt^2}{4} < \frac{1}{2}m^3 + tm^2$$

Z čehož dostaneme

$$m^2(-1+4\delta) - 3tm(1-2\delta) + 3t^2 < 0$$

Vidíme, že pro všechna m se levá strana minimalizuje pro $t = \frac{m(1-2\delta)}{2}$. Dosazením dostaneme:

$$m^2\Big[-1+4\delta-\frac{3}{2}(1-2\delta)^2+\frac{3}{4}(1-2\delta)^2\Big] < 0$$

což nám dává $-7+28\delta-12\delta^2 < 0$. Dostáváme:

$$\delta < \frac{7}{6} - \frac{1}{3}\sqrt{7} \approx 0{,}284$$

Tedy pro dostatečně velké m dostaneme $g_1 \in \mathbb{Z}[x,y]$,že $g_1(x_0,y_0) = 0$ nad celými čísly. Toto nám ale nestačí. Abychom dostali další vztah, použijeme Tvrzení 3, čímž odhadneme normu b_2.

Poznámka. Jelikož původní báze L je trojúhelníková matice, u^*_{min} je nejmenší prvek na diagonále (viz Boneh a Durfee (2000)).

Z poznámky vidíme, že tento prvek je poslední řádek $x-$posunutí tedy $u^*_{min} = X^mY^m$ což víme je větší než 1, tedy můžeme použít důsledek za Tvrzením 4.

Kombinací důsledku a Tvrzení 5 vidíme, že b_2 nám dá další polynom g_2 splňující $g_2(x_0,y_0) = 0$ pokud $det(L) < N^{m(w-1)}/\gamma'$, kde $\gamma' = (w2^w)^{\frac{w-1}{2}}$. Pro dostatečně velké m nám toto bude platit. A tedy máme $g_2 \in \mathbb{Z}[x,y]$, které je lineárně nezávislé k g_1 a splňuje $g_2(x_0,y_0) = 0$ nad celými čísly. Nyní se můžeme pokusit o vyřešení pro y_0 tím, že spočítáme rezultant $h(y) = res_x(g_1,g_2)$. Použijeme na to počítání přes Sylvesterovu matici.

Definice 7. *Buď* $f = \sum_{i=0}^{n} a_ix^i, g = \sum_{i=0}^{m} b_ix^i$ *dva nekonstantní polynomy z* $\boldsymbol{R}[x]$. *Sylvesterovou maticí* $M(f,g)$ *rozumíme čtvercovou matici velikosti* $m+n$ *nad* $\boldsymbol{R}$ *definovanou*

$$M(f,g) = \begin{pmatrix}
a_n & a_{n-1} & \cdots & \cdots & \cdots & a_0 & 0 & \cdots & \cdots & 0 \\
0 & a_n & a_{n-1} & \cdots & \cdots & \cdots & a_0 & 0 & \cdots & 0 \\
0 & 0 & a_n & a_{n-1} & \cdots & \cdots & \cdots & a_0 & \cdots & 0 \\
\vdots & & & & & & & & & \\
0 & 0 & \cdots & 0 & a_n & a_{n-1} & \cdots & \cdots & \cdots & a_0 \\
b_m & b_{m-1} & \cdots & \cdots & \cdots & b_0 & 0 & \cdots & \cdots & 0 \\
0 & b_m & b_{m-1} & \cdots & \cdots & \cdots & b_0 & 0 & \cdots & 0 \\
0 & 0 & b_m & b_{m-1} & \cdots & \cdots & \cdots & b_0 & \cdots & 0 \\
\vdots & & & & & & & & & \\
0 & 0 & \cdots & 0 & b_m & b_{m-1} & \cdots & \cdots & \cdots & b_0
\end{pmatrix}$$

Determinant této matice se nazývá rezultant polynomů f,g a značí se

$$res(f,g) = detM(f,g).$$

(Rezultant je tedy prvkem okruhu $\boldsymbol{R}$*).*

My ale máme polynomy o dvou proměnných. Budeme chtít $res_x(g_1,g_2)$ a tedy nahlédneme na ně jako na polynomy v proměnné x nad $(\mathbb{Z}[y])[x]$ a tedy dostaneme, že $h(y) = res_x(g_1,g_2) \in \mathbb{Z}[y]$. Pak y_0 musí být kořenem $h(y)$.

Tvrzení 7 (Sylvestrovo kritérium)**.** *Buď* $\boldsymbol{R}$ *gaussovský obor,* $\boldsymbol{Q}$ *jeho podílové těleso a f,g nekonstantní polynomy z* $\boldsymbol{R}[x]$*. Pak jsou následující tvrzení ekvivalentní:*

$$(1) \textit{Polynomy } f,g \textit{ jsou soudělné v } \boldsymbol{Q}[x]$$
$$(2) res(f,g) = detM(f,g)^T = 0.$$

Důkaz. Viz. učebnice Počítačová algebra (Stanovský, 2017).

□

Polynomy $g_1(x,y_0)$ a $g_2(x,y_0)$ jsou soudělné, jelikož mají společný kořen x_0. Pokud tedy dosadíme y_0 za y do $M(g_1,g_2)$ a vznikne Sylvesterova matice obsahující nekonstantní polynomy $g_1(x,y_0), g_2(x,y_0)$, musí být y_0 kořenem $h(y)$. I přes to, že jsou g_1,g_2 lineárně nezávislé, nemusí být algebraicky nezávislé $\Rightarrow$ v obecných případech nemůžeme garantovat to, že resultant $h(x)$ není 0.

2.2 Obecný postup

Nyní si popíšeme obecný postup řešení našeho problému. Vycházíme z článku Jochemsz a May (2006).

Poznámka. Polynom f, který řešíme modulo nějaké N budeme značit f_N.

Chceme najít malý kořen $(x_1^{(0)},...,x_n^{(0)})$ polynomu f_N, kde N známe, ale neznáme jeho faktorizaci. Předpokládejme, že máme horní odhad pro náš kořen, neboli platí $|x_j^{(0)}| < X_j$ pro $j = 1,...,n$. Nechť l je vedoucí monočlen f_N s koeficientem $a_l \Rightarrow$ není zde žádný jiný takový, že by byl dělitelný l. Vidíme, že $NSD(a_l,N) = 1$ jelikož pokud by měli společného dělitele, najdeme faktorizaci N. Můžeme proto použít $f'_N = a_l^{-1} f_N \ (mod\ N)$. Ukážeme základní strategii a poté rozšířenou, ale nejprve zadefinujeme pojem vedoucí monočlen a lexikografické uspořádání na množině monočlenů.

Definice 8. *Mějme monočleny* $x_1^{u_1} \cdot ... \cdot x_k^{u_k}$ *a* $x_1^{v_1} \cdot ... \cdot x_k^{v_k}$*. Lexikografické uspořádání znamená, že*

$$x_1^{u_1} \cdot ... \cdot x_k^{u_k} < x_1^{v_1} \cdot ... \cdot x_k^{v_k} \Leftrightarrow \exists i : u_1 = v_1,...,u_i < v_i.$$

O pořadí rozhoduje exponent u x_1*, v případě rovnosti pak u* x_2 *atd.*
Vedoucí monočlen je ten, který je nejvyšší vzhledem k lexikografickému uspořádání.

Příklad. Pro x_1,x_2,x_3 platí: $x_1 > x_2^{100} > x_2x_3^{100} > x_2x_3 > x_3^{100}$

2.2.1 Základní strategie

Pevně zvolíme nějaké $m \in \mathbb{Z}$, jehož volba závisí na analýze každého jednotlivého případu. Pro $k \in \{0,...,m+1\}$ zadefinujeme množiny monočlenů M_k jako:

$$M_k := \{x_1^{i_1}x_2^{i_2}\dots x_n^{i_n} \mid x_1^{i_1}x_2^{i_2}\dots x_n^{i_n} \text{ je monočlen v } f_N^m \text{ a } \frac{x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}}{l^k} \text{ je monočlen v } f_N^{m-k}\}.$$

Budeme předpokládat, že všechny monočleny, které jsou v f_N,...,f_N^{m-1} jsou také v f_N^m a $M_{k+1} \subseteq M_k$. Platí nám to jestliže máme m a koeficienty polynomu kladné s nenulovým konstantním členem a jsou malé oproti N. Pokud by tomu tak nebylo, změnili bychom definici M_k tak, že by tato množina obsahovala všechny monočleny $x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}$ z f_N^j pro $j \in \{1,...,m\}$,pro které je $\frac{x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}}{l^k}$ v f_N^i pro nějaké $i \in \{0,...,m-k\}$.
Podle naší definice vidíme, že množina M_0 obsahuje všechny monočleny v f_N^m a $M_{m+1} = \emptyset$.

Příklad. Mějme polynom $f_N(x,y) = x^2y + xy^2 + 1$ a $m = 2$. Vedoucí monočlen je $l = x^2y$. Vypíšeme si, jak vypadají jednotlivá M_k pro $k \in \{0,...,3\}$:

$$\begin{aligned} M_3 &= \emptyset \\ M_2 &= \{x^4y^2\} \\ M_1 &= \{x^2y, x^3y^3, x^4y^2\} \\ M_0 &= \{1,xy^2,x^2y^4,x^2y, x^3y^3, x^4y^2\} \end{aligned}$$

Jako další zadefinujeme polynom posunutí:

$$g_{i_1,\dots,i_n} := \frac{x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}}{l^k} f'_N(x_1,...,x_n)^k N^{m-k}$$

pro $k = 0,...,m$ a $x_1^{i_1}x_2^{i_2}\dots x_n^{i_n} \in M_k \setminus M_{k+1}$.

Poznámka. Polynomy g mají $(x_1^{(0)},...,x_n^{(0)})$ jako kořen modulo N^m.

Příklad. Vezměme si polynom jako výše a koukneme se jak nám dopadnou posunutí:

$$\begin{aligned} &\text{pro } 1 \in M_0 \setminus M_1 : g(x,y) = N^2 \\ &\text{pro } xy^2 \in M_0 \setminus M_1 : g(x,y) = xy^2N^2 \\ &\text{pro } x^2y^4 \in M_0 \setminus M_1 : g(x,y) = x^2y^4N^2 \\ &\text{pro } x^2y \in M_1 \setminus M_2 : g(x,y) = f_N N \\ &\text{pro } x^3y^3 \in M_1 \setminus M_2 : g(x,y) = xy^2 f_N N \\ &\text{pro } x^4y^2 \in M_2 \setminus M_3 : g(x,y) = f_N^2 \end{aligned}$$

Píšeme zde f_N jelikož pro náš příklad platí $f_N = f'_N$

Nyní zadefinujeme mřížku $\mathbf{L}$ tak, že za její bázi vezmeme vektory koeficientů $g(x_1X_1,...,x_nX_n)$. Můžeme sestavit matici popisující $\mathbf{L}$ tak, aby byla dolní trojúhelníková. Postup shrneme v následujícím Lemmatu.

Lemma 8. *Nechť* $\boldsymbol{L}$ *je matice, která má v řádcích koeficienty polynomů* $g_{i_1,\dots,i_n}$ *a sloupce určují pořadí těchto koeficientů. Pokud bázové sloupce seřadíme vzestupně zleva doprava a polynomy vzestupně podle vedoucích monočlenů, pak dostaneme dolní trojúhelníkovou matici. Platí, že sloupec odpovídající monočlenu*

$$x_1^{i_1}x_2^{i_2}\dots x_n^{i_n} \in M_k \setminus M_{k+1}$$

má menší řád než ten korespondující s

$$x_1^{j_1}x_2^{j_2}\dots x_n^{j_n} \in M_{k'} \setminus M_{k'+1}$$

pokud $k < k'$ *(je-li* $k = k'$*, použijeme lexikografické pořadí monočlenů),*

Důkaz. Ve sloupcích máme pouze vedoucí monočleny polynomů, které dáváme do řádků. Jelikož platí, že $g_{i_1,\dots,i_n}$ má jako vedoucí monočlen $x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}$. Tyto polynomy mají pokaždé jiný tento monočlen. Pokud je seřadíme také vzestupně podle těchto hodnot, dostaneme nenulové prvky na diagonále a možná nějaké vlevo od diagonály, jelikož zde máme koeficienty těchto polynomů, a tedy vlevo od vědoucího členu se mohou vyskytnout nějaké další nenulové prvky. Napravo žádné být nemohou díky lexikografickému uspořádání sloupců. Tím pádem dostaneme dolní trojúhelníkovou matici. S vedoucími členy polynomů na diagonále. □

Na diagonále naší matice jsou prvky odpovídající monočlenu l^k v $(f'_N)^k$ pro každý řádek. Neboli jsou to $X_1^{i_1}X_2^{i_2}\dots X_n^{i_n}N^{m-k}$ pro danou kombinaci k a i_j.

Příklad. Pro náš polynom nám vyjde matice mřížky $\mathbf{L}$ jako:

	1	xy^2	x^2y^4	x^2y	x^3y^3	x^4y^2
N^2	N^2					
xy^2N^2		XY^2N^2				
$x^2y^4N^2$			$X^2Y^4N^2$			
f_NN	N	XY^2N		X^2YN		
xy^2f_NN		XY^2N	X^2Y^4N		X^3Y^3N	
f_N^2	1	$2XY^2$	X^2Y^4	$2X^2Y$	$2X^3Y^3$	X^4Y^2

Vysvětlení za zvolení množin M_k je takové, že chceme dostat matici s malým determinantem (Tvrzení 3). Abychom udrželi prvky na diagonále korespondující s monočleny $x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}$ polynomu f_N^m co nejmenší, použijeme nejvyšší možnou mocninu f_N v posunutích daných polynomem g. Podmínka, že $\frac{x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}}{l^k}$ je v f_N^{m-k} nám zajistí, že se nám zde neobjeví žádný monočlen, který není z f_N^m. V zásadě zjistíme, že náš odhad $det(L) < N^{m(w+1-n)}$, který dostaneme z Tvrzení 4 a Tvrzení 5 se např. po zanedbání malých členů zredukuje na:

$$\prod_{j=1}^{n} X_j^{s_j} < N^{s_N} \quad \begin{cases} s_j = \sum_{x_1^{i_1}x_2^{i_2}\dots x_n^{i_n}\in M_0} i_j \\ s_N = \sum\limits_{k=0}^{m} k(|M_k| - |M_{k+1}|) = \sum\limits_{k=1}^{m} |M_k| \end{cases} \tag{2.1}$$

Lemma 9 (Odvození vzorce 2.1.)**.** *Z odhadu* $det(L) < N^{m(w+1-n)}$ *matice dostaneme vzorec 2.1.*

Důkaz. Vycházíme z nerovnosti $det(L) < N^{m(w+1-n)}$. Koukneme se, jak vypadá determinant matice. Jelikož je to dolní trojúhelníková, je roven součinu prvků na diagonále což víme, že jsou prvky $X_1^{i_1} X_2^{i_2} \dots X_n^{i_n} N^{m-k}$ pro danou kombinaci k a i_j. Z toho hned vidíme, že

$$det(L) = \prod_{j=1}^{n} X_j^{s_j} N^{\sum_{k=0}^{m} (m-k)(|M_k|-|M_{k+1}|)},$$

kde $s_j = \sum_{x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \in M_0} i_j$.
Koukneme se na mocninu N. Pokud dosadíme do odhadu pro determinant, dostaneme:

$$\prod_{j=1}^{n} X_j^{s_j} N^{\sum_{k=0}^{m} (m-k)(|M_k|-|M_{k+1}|)} < N^{m(w+1-n)}$$

$$\prod_{j=1}^{n} X_j^{s_j} < N^{mw+m-mn-\sum_{k=0}^{m} (m-k)(|M_k|-|M_{k+1}|)}$$

Platí, že

$$\sum_{k=0}^{m} m(|M_k| - |M_{k+1}|) = m(|M_0| - |M_1| + |M_1| - \dots + |M_m| - |M_{m+1}|) = mw$$

$\Rightarrow$ dostáváme

$$\prod_{j=1}^{n} X_j^{s_j} < N^{m-mn+\sum_{k=0}^{m} k(|M_k|-|M_{k+1}|)}$$

$$\prod_{j=1}^{n} X_j^{s_j} < N^{m-mn+\sum_{k=1}^{m} |M_k|}$$

Vzorec 2.1 zanedbává $m - mn$, a tedy dostáváme požadovaný výsledek.

□

Pokud požijeme pro dané f_N tento postup, dá nám vzorec 2.1 horní odhad na kořen, který chceme najít. Pro X_j a N, která splňují 2.1 dostaneme n polynomů h_i tak, že $h_i(x_1^{(0)},\dots,x_n^{(0)}) = 0$. První polynom najdeme tak, že použijeme LLL algoritmus abychom našli krátký vektor mřížky. Tento vektor poté bude odpovídat koeficientům našeho polynomu s kořenem $(x_1^{(0)},\dots,x_n^{(0)})$. Další dostaneme též z LLL redukované báze, ale odhady na tyto vektory budou čím dál tím horší. Pokud jsou algebraicky nezávislé (tedy nesdílejí netriviální NSD), dostaneme z výpočtu resultantu kořen což nás navede k hledanému $(x_1^{(0)},\dots,x_n^{(0)})$. Ale pouze za předpokladu, že výpočtem resultantu pro h_i dostaneme nenulové polynomy.

Příklad. Ukážeme, jak by tento postup fungoval pro $n = 3$.
Algoritmem dostaneme nějaké tři polynom $h_1,h_2,h_3 \in \mathbb{Z}[x_1,x_2,x_3]$. Postupujeme tak, že spočítáme $res_{x_3}(h_1,h_2) = r$ a $res_{x_3}(h_1,h_3) = s$, neboli nahlédneme na ně jako na polynomy v proměnných x_1,x_2 nad $(\mathbb{Z}[x_1,x_2])[x_3]$. Platí $r,s \in \mathbb{Z}[x_1,x_2]$ a mají společný kořen $(x_1^{(0)},x_2^{(0)})$, čímž dostaneme stejnou situaci jako v minulé kapitole.

Všechny metody pro $n \geq 2$ mají podobný předpoklad, že jsou polynomy h_i nezávislé. A tedy se vždy musí provést nějaký experiment, abychom ověřili správnost našeho předpokladu.

2.2.2 Rozšířená strategie

Jak již jsme si ukazovali v příkladu výše, dostaneme lepší výsledky, pokud máme ještě jiná posunutí než ty základní. A tedy tato strategie se liší pouze ve volbě M_k a to tak, že definujeme

$$M_k := \bigcup_{0 \leq j \leq t} \{x_1^{i_1+j} x_2^{i_2} \dots x_n^{i_n} \mid x_1^{i_1} x_2^{i_2} \dots x_n^{i_n} \text{ je monočlen v } f_N^m \text{ a } \frac{x_1^{i_1} x_2^{i_2} \dots x_n^{i_n}}{l^k} \text{ je monočlen v } f_N^{m-k}\}.$$

Neboli používáme extra posunutí pro x_1. Samozřejmě můžeme použít i posunutí více proměnných, ale to opět záleží na typu příkladu. Vždy volíme tak, abychom dostali optimální mez pro náš kořen.
Zbytek strategie je úplně stejný jako v Základní a nebudeme se tím více zabývat. V další sekci si spočteme příklad pomocí základní verze a koukneme jak se nám výsledky liší od rozšířené.

3. Příklady

V této kapitole budeme řešit příklady pomocí našeho základního postupu. Jako první budeme řešit polynom $f_N = x + ay + bz + c$ a jako druhý budeme mít příklad $f_N = axy + bx + c$, u kterého známe odhad pomocí rozšířené verze a koukneme se tedy jak moc se nám to bude lišit.

3.1 $f_N = x + ay + bz + c$

Začneme tím, že na příklad koukneme přes obecné m a poté zkusíme ukázat jak by to vypadalo pro $m = 2$. Vedoucí monočlen je $l = x$ a vidíme, že $a_l = 1$, a proto nemusíme rozlišovat f_N a f'_N.
Chceme najít množiny M_0,...,M_{m+1}. Víme, že M_0 obsahuje všechny monočleny z f_N^m. Dostaneme něco jako:

$$M_0 = \{x^m, x^{(m-1)}y, x^{(m-1)}z, ..., z, 1\}$$

V M_i pro $i = 1,...,m$ jsou členy, které jsou v M_0 a jsou dělitelné x^i:

$$M_i = \{x^{j_1}y^{j_2}z^{j_3} \mid x^{j_1}y^{j_2}z^{j_3} \in M_0 \text{ a } j_1 \geq i\}$$

Poté vytvoříme polynomy posunutí jako

$$g_{i_1,i_2,i_3} := \frac{x^{i_1}y^{i_2}z^{i_3}}{x^k} f_N(x,y,z)^k N^{m-k} \text{ pro } x^{i_1}y^{i_2}z^{i_3} \in M_k \setminus M_{k+1}$$

Matice mřížky bude mít na diagonále prvky $X^{i_1}Y^{i_2}Z^{i_3}N^{m-k}$ v závislosti na i_j a k. Dimenze mřížky w je rovna velikosti $|M_0|$ což máme ze vzorce (kniha od Chen a Koh (1992)) pro počet členů :

$$w = \frac{(m+1)(m+2)(m+3)}{6}.$$

Jelikož máme obecné m, nebudeme mřížku vypisovat, ale spočítáme si odhad z vzorce 2.1.
Z tvaru našeho polynomu $x + ay + bz + c$ vidíme, že s_j je stejné pro všechny tři proměnné, a tedy budeme počítat pouze pro x.
Pokud dáme f_N^m bude zde x v mocnině $(m-i)$ právě $1/2(i+1)(i+2)$ krát a tedy dostaneme:

$$s_j = \sum_{i=0}^{m}(m-i)\frac{1}{2}(i+1)(i+2) = \frac{1}{24}m(1+m)(2+m)(3+m).$$

Dále máme:

$$s_N = \sum_{k=1}^{m} |M_k|.$$

Nahlédneme tedy, co obsahují M_k pro $k = 1,...,m$. V M_1 máme všechny členy s x až x^m. V M_2 máme členy s x^2 až x^m. Z toho vidíme, že každé x^i je v i množinách. A tedy $s_N = s_j$ z čehož dostáváme:

$$X^{s_j}Y^{s_j}Z^{s_j} < N^{s_j} \Rightarrow |X||Y||Z| < |N|$$

Nyní ukážeme případ s $m = 2$. Vedoucí monočlen je Nyní tedy chceme zjistit jak vypadá M_0,...,M_3. Víme, že M_0 obsahuje všechny monočleny z f_N^2:

$$M_0 = \{x^2,xy,xz,x,y^2,yz,y,z^2,z,1\}$$

M_1 obsahuje monočleny z f_N^2 tak, že pokud je vydělíme x, budou náležet do f_N. M_2 a M_3 jsou jasné:

$$\begin{aligned} M_1 &= \{x^2,xy,xz,x\} \\ M_2 &= \{x^2\} \\ M_3 &= \emptyset \end{aligned}$$

Množiny máme vypsané a koukneme na posunutí. Pro přehlednost označíme:

$$g(x,y,z) = x^{i_1}y^{i_2}z^{i_3}f_N^k N^{m-k} \text{ pro } x^{i_1}y^{i_2}z^{i_3} \in M_k \setminus M_{k+1}$$

Z toho dostaneme tyto polynomy:

pro $1 \in M_0 \setminus M_1 : g(x,y) = N^2$	pro $y^2 \in M_0 \setminus M_1 : g(x,y) = y^2N^2$
pro $z \in M_0 \setminus M_1 : g(x,y) = zN^2$	pro $x \in M_1 \setminus M_2 : g(x,y) = f_N N$
pro $z^2 \in M_0 \setminus M_1 : g(x,y) = z^2N^2$	pro $xz \in M_1 \setminus M_2 : g(x,y) = zf_N N$
pro $y \in M_0 \setminus M_1 : g(x,y) = yN^2$	pro $xy \in M_1 \setminus M_2 : g(x,y) = yf_N N$
pro $yz \in M_0 \setminus M_1 : g(x,y) = yzN^2$	pro $x^2 \in M_2 \setminus M_3 : g(x,y) = f_N^2$

Matice mřížky bude vypadat takto:

$$L = \begin{pmatrix} N^2 & & & & & & & & & \\ & ZN^2 & & & & & & & & \\ & & Z^2N^2 & & & & & & & \\ & & & YN^2 & & & & & & \\ & & & & YZN^2 & & & & & \\ & & & & & Y^2N^2 & & & & \\ - & - & & - & & & XN & & & \\ & - & - & & - & & & XZN & & \\ & & & - & - & - & & & XYN & \\ - & - & - & - & - & - & - & - & - & X^2 \end{pmatrix}$$

"-"nám opět značí nenulové hodnoty, které nás nezajímají. Podíváme se, jak nám dopadl determinant:

$$det(L) = N^{15}X^5Y^5Z^5$$

Podle vzorce 2.1 dostaneme, že $X^5Y^5Z^5 < N^5 \Rightarrow |X||Y||Z| < |N|$

3.2 $f_N = axy + bx + c$

V článku Jochemsz a May (2006) používali rozšířenou strategii, ale my použijeme pouze tu základní a uvidíme, jak moc se nám bude lišit výsledek.

Zvolíme $m = 2$. Vedoucí monočlen $l = xy$ a $a_l = a$, tedy budeme mít $f'_N = a^{-1} f_N$. Dostaneme množiny:

$$M_0 = \{x^2y^2, x^2y, x^2, xy, x, 1\}$$
$$M_1 = \{x^2y^2, x^2y, xy\}$$
$$M_2 = \{x^2y^2\}$$
$$M_3 = \emptyset$$

Dále máme posunutí

pro $1 \in M_0 \setminus M_1 : g(x,y) = N^2$
pro $x \in M_0 \setminus M_1 : g(x,y) = xN^2$
pro $x^2 \in M_0 \setminus M_1 : g(x,y) = x^2N^2$
pro $xy \in M_1 \setminus M_2 : g(x,y) = f'_N N$
pro $x^2y \in M_1 \setminus M_2 : g(x,y) = xf'_N N$
pro $x^2y^2 \in M_2 \setminus M_3 : g(x,y) = f'^2_N$

Matice mřížky:

$$L = \begin{pmatrix} N^2 & & & & & \\ & XN^2 & & & & \\ & & X^2N^2 & & & \\ - & - & & XYN & & \\ & - & - & & X^2YN & \\ - & - & - & - & - & X^2Y^2 \end{pmatrix}$$

Opět "-"značí nenulové hodnoty. Z vzorce 2.1 dostaneme $X^8Y^4 < N^4 \Rightarrow X^2Y < N$. V článku Jochemsz a May (2006) mají výsledek pomocí rozšířené strategie $X^{1+4\tau}Y^{1+3\tau+3\tau^2} < N^{1+3\tau}$, kde $t = \tau m$ pro nějaké $\tau > 0$ a používají $x^{i_1}y^{i_2} \in M_k \Leftrightarrow i_1 = k,...,m; i_2 = k,...,i_1 + t$.

Závěr

V této práci jsme nejprve ukázali jak pracovat s mřížkami, LLL algoritmem a Coppersmithovu metodu na hledání kořenů polynomů více proměnných. Tento algoritmus je založen na redukci báze mřížky, proto jsme popsali několik tvrzení, která nám odhadují délky vektorů LLL-redukované báze. Coppersmithovu metodu jsme poté ukázali na složitějším příkladu od Boneh a Durfee (2000). Tento příklad zde máme pouze v základní verzi. Lepší odhady a poté použití rozšířené strategie na tento příklad se dají najít v článku.
Dále jsme popsali obecnější formu tohoto algoritmu z článku od Jochemsz a May (2006), kterou jsme poté použili na řešení dvou jednoduchých příkladů.
Motivací k využití této metody je například útok na RSA, což je v podstatě příklad od Boneh a Durfee (2000) v druhé kapitole.

Seznam použité literatury

Boneh, D. a Durfee, G. (2000). Cryptanalysis of RSA with private key less than $N^{0.292}$. *IEEE Trans. on Information theory*, **46(4)**, 1339 – 1349.

Chen, C. a Koh, K. (1992). *Principles and Techniques in Combinatorics*. WSPC.

Jochemsz, E. a May, A. (2006). A strategy for finding roots of multivariatepolynomials with new applicationsin attacking RSA variants. *ASIACRYPT 2006, LNCS 4284*, pages 267–282.

Stanovský, D. a Barto, L. (2017). *Počítačová algebra*. Druhé upravené vydání. Matfyzpress, Praha. ISBN 978-80-7378-167-5.

Wiener, M. (1990). Cryptanalysis of short RSA secret exponents. *IEEE Transactions on Information theory*, **36**, 553–558.