UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE MATEMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM MATEMÁTICA APLICADA

Polinômios Multivariados: fatoração e mdc.

por

Luiz Emilio Allem

Tese submetida como requisito parcial
para a obtenção do grau de

Doutor em Matemática Aplicada

Prof. Dr. Vilmar Trevisan
Orientador

Porto Alegre, Outubro de 2010.

CIP - CATALOGAÇÃO NA PUBLICAÇÃO

Allem, Luiz Emilio

Polinômios Multivariados:
fatoração e mdc. / Luiz Emilio Allem.—Porto Alegre: PPGMAp da UFRGS, 2010.

95 p.: il.

Tese (doutorado) —Universidade Federal do Rio Grande do Sul, Programa de Pós-Graduação em Matemática Aplicada, Porto Alegre, 2010.
Orientador: Trevisan, Vilmar

Tese: Matemática Aplicada
Politopos, Teorema da Irredutibilidade de Hilbert, Fatoração de polinômios, mdc

Polinômios Multivariados: fatoração e mdc.

por

Luiz Emilio Allem

Tese submetida ao Programa de Pós-Graduação em Matemática Aplicada do Instituto de Matemática da Universidade Federal do Rio Grande do Sul, como requisito parcial para a obtenção do grau de

Doutor em Matemática Aplicada

Linha de Pesquisa: Algoritmos Numéricos e Algébricos

Orientador: Prof. Dr. Vilmar Trevisan

Banca examinadora:

Prof. Dr. Jayme Luiz Szwarcfiter
UFRJ

Profa. Dra. Sueli Irene Rodrigues da Costa
UNICAMP

Prof. Dr. Carlos Hoppen
PPGMAp/IM/UFRGS

Profa. Dra. Virgínia Maria Rodrigues
PUCRS

Tese apresentada e aprovada em
21 de Outubro de 2010.

Prof. Waldir Leite Roque, Ph.D.
Coordenador

SUMÁRIO

RESUMO

Nesta tese de doutorado estudamos polinômios multivariados. Começamos fazendo uma revisão bibliográfica sobre o teorema da irredutibilidade de Hilbert. Abordamos com detalhes as demonstrações da versão clássica feita pelo próprio Hilbert e das versões efetivas feitas por Erich Kaltofen e Shuhong Gao.

Desenvolvemos um novo algoritmo para fatoração de polinômios multivariados inteiros usando logaritmo discreto. Nosso método é baseado em novos tipos de reduções de polinômios multivariados para polinômios bivariados, as quais têm como principal característica manter a esparsidade do polinômio. Nosso método mostrou-se eficiente quando usado para fatorar polinômios multivariados que possuem apenas fatores esparsos e quando usado para extrair fatores esparsos de polinômios multivariados que têm fatores esparsos e densos.

Terminamos essa tese trabalhando com o máximo divisor comum (mdc) de polinômios. Estudamos critérios geométricos de politopos para determinar coprimalidade entre polinômios multivariados. Desenvolvemos um novo algoritmo que trabalha em tempo polinomial (sobre o número de monômios) para detectar coprimalidade entre polinômios multivariados usando seus politopos de Newton associados. Esse método geométrico tem a vantagem de determinar a coprimalidade entre famílias de polinômios, pois podemos mudar arbitrariamente os coeficientes dos polinômios desde que certos coeficientes permaneçam não nulos. Além disso, os polinômios permanecerão coprimos sobre qualquer corpo. Terminamos mostrando como construir o mdc entre dois polinômios bivariados usando seus polígonos de Newton associados.

ABSTRACT

In this dissertation we study multivariate polynomials. We begin with a bibliographical review on the Hilbert irreducibility theorem. We cover in detail the demonstrations of the classic version due to Hilbert himself and effective versions due to Erich Kaltofen and Shuhong Gao.

We developed a new algorithm for factoring multivariate integral polynomials using discrete logarithm. Our method is based on new types of reductions, from multivariate polynomias to bivariate polynomials, whose main feature is to maintain the sparsity of the polynomial. Our method has proved to be efficient when used for factoring multivariate polynomials that have only sparse factors and when used to extract sparse factors of multivariate polynomials that have sparse and dense factors.

We finish this dissertation studying the greatest common divisor (gcd) of polynomials. We study geometric criteria of polytopes to determine coprimality between multivariate polynomials. We developed a new algorithm that works in polynomial time (on the number of monomials) to detect coprimality between multivariate polynomials using their associated Newton polytopes. This geometric method has the advantage of determining the coprimality between families of polynomials, since we can arbitrarily change the polynomial coefficients as long as some coefficients remain non-zero. Moreover, the coprime polynomials shall remain coprime on any field. We ended up showing how to build the gcd between two bivariate polynomials using their associated Newton polygons.

AGRADECIMENTOS

Agradeço a minha família. Ao meu pai Luiz Costa Allem pelo companheirismo, à minha mãe Marlei Beatriz Allem que sempre me motivou e nunca me deixou desanimar, minha irmã Luciane Beatriz Allem por todos esses anos de muito amor, muito incentivo, muita compreensão. Agradeço pela paciência e ajuda em todos os sentidos. Amo vocês.

Agradeço a todos os amigos do departamento de matemática, em especial: Paulo Lino, Lucas, Vander, Julio, Antonio (Colômbia), Jorge, João, Esequia, Fábio, Guilherme, Renne, Ju, Fernando, John, Greice...e peço desculpas se esqueci de alguém.

Agradeço à Professora e amiga Maria Cristina Varriale pela motivação e pela amizade ao longo dos anos. Agradeço ao Professor e amigo Carlos Hoppen pela amizade e pela disposição entusiasmada para discutir matemática.

Agradeço ao Professor Shuhong Gao de Clemson University que me acolheu e orientou durante meu doutorado sanduíche, sempre com muita disposição e passando grande conhecimento matemático.

Agradeço ao Instituto de Matemática da UFRGS, à CAPES e ao CNPq que sempre me deram ótimas condições de estudar e auxílio financeiro durante estes anos.

Em especial quero agradecer a minha namorada Márcia pois sei que não é fácil namorar alguém cujo trabalho é estudar. Te agradeço pelo apoio, pelo carinho. E como posso te agradecer por ter ido comigo para Clemson? Márcia me encorajou a ir e me apoiou dizendo que iria comigo, com certeza se ela não fosse eu não teria ido. Tu estás no meu coração. Te amo!

Agradeço a Deus por ter me dado saúde e vontade para chegar até aqui.

DEDICATÓRIA

Eu quero dedicar esta tese de doutorado ao meu orientador e amigo professor Vilmar Trevisan que eu conheci no meu primeiro dia de aula na UFRGS. Ao longo dos anos, desde a graduação até aqui o professor Vilmar tem sido um grande incentivador que sempre me motivou e ajudou muito. O professor Vilmar nunca deixou de me apoiar e estar sempre disposto a responder minhas perguntas. Com grande alegria dedico este trabalho ao professor Vilmar e agradeço por tudo que ele tem feito por mim até hoje; sem o professor Vilmar, com certeza, eu não teria chegado a este feito.

1 INTRODUÇÃO

Nesta tese de doutorado apresentamos um estudo sobre polinômios multivariados. Mais precisamente, fazemos uma revisão bibliográfica sobre as versões do teorema da irredutibilidade de Hilbert, desenvolvemos um novo algoritmo para fatoração de polinômios multivariados baseado em um novo tipo de redução, e introduzimos um método para detectar coprimalidade entre polinômios multivariados usando seus politopos de Newton associados.

O Capítulo 2 desta tese é dedicado ao estudo do Teorema da Irredutibilidade de Hilbert. Dado um polinômio multivariado irredutível F, esta teoria aborda a manutenção da irredutibilidade de F quando algum tipo de redução é aplicada no polinômio multivariado. O trabalho que deu o nome a esta área de estudo, feito pelo próprio Hilbert [13] em 1892, garante que dado um polinômio $F(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ irredutível sobre $\mathbb{Q}$, então existem infinitos números inteiros $a_1, \ldots, a_{n-1}$ tais que o polinômio univariado reduzido $F(a_1, \ldots, a_{n-1}, x_n)$ permanece irredutível sobre $\mathbb{Q}$. Este resultado é conhecido como o Teorema da Irredutibilidade de Hilbert clássico.

Em meados de 1980, cerca de 100 anos depois, novos tipos de redução foram apresentadas por Erich Kaltofen [14, 15] e Joachin von zur Gathen [32, 34]. Estas novas substituições reduzem um polinômio multivariado para um polinômio bivariado. O diferencial destas novas técnicas é que, ao contrário do teorema da irredutibilidade de Hilbert clássico, possuem argumentos probabilísticos com respeito à manutenção da irredutibilidade. Ou seja, dado um polinômio multivariado irredutível F então F tem boa probabilidade de permanecer irredutível quando reduzido para um polinômio bivariado.

Ao longo dos anos, cotas probabilísticas para reduções similares às apresentadas por Kaltofen e von zur Gathen têm sido melhoradas por: Bajaj e seus

colaboradores [3] em 1989, Erich Kaltofen [16] em 1995, Shuhong Gao [8] em 2001 e a melhor cota conhecida pelo autor foi estabelecida em 2007 por Lecerf [20].

As primeiras implementações práticas de algoritmos para fatoração polinomial multivariada foram feitas na década de 70 por Paul Wang e Rothschild [37, 38]. Estes algoritmos são baseados no Teorema da Irredutibilidade de Hilbert clássico. Neste método o polinômio multivariado $F(x_1, \ldots, x_n)$ é reduzido para um polinômio univariado $F(a_1, \ldots, a_{n-1}, x_n)$, para certas constantes aleatórias $a_1, \ldots, a_{n-1}$, e então fatorado. Os fatores do polinômio univariado são usados para construir os fatores multivariados usando levantamento de Hensel.

Quando utilizamos o método do parágrafo anterior para a fatoração de polinômios multivariados podemos nos deparar com o problema conhecido na literatura como: **problema dos zeros ruins**. Isto acontece quando o polinômio univariado $F(0, \ldots, 0, x_n)$ tem grau menor em x_n se comparado a $F(x_1, \ldots, x_n)$ ou quando $F(0, \ldots, 0, x_n)$ não é livre de quadrados. Devido a isto, como pode ser visto em [15, 38], devemos fatorar, no lugar de $F(x_1, \ldots, x_n)$, o polinômio $F(x_1, x_2 + a_2, \ldots, x_n + a_n)$. Observe que utilizando este tipo de transformação, $F(x_1, x_2 + a_2, \ldots, x_n + a_n)$ provavelmente terá, especialmente se F for esparso, um número bem maior de termos que $F(x_1, \ldots, x_n)$. Ou seja, se considerarmos que $F_e x_1^{e_1} \cdots x_n^{e_n}$ seja um termo do polinômio multivariado $F(x_1, \ldots, x_n)$, então $F_e x_1^{e_1} \cdots x_n^{e_n}$ irá criar vários termos em $F(x_1, x_2 + a_2, \ldots, x_n + a_n)$, como ilustrado na figura abaixo.

Wang
$F_e x_1^{e_1} \cdots x_n^{e_n}$
$\downarrow$
$F_e x_1^{e_1} (x_2 - a_2)^{e_2} \cdots (x_n - a_n)^{e_n}$

Nos anos 80, von zur Gathen e Kaltofen [34] apresentaram um algoritmo para fatoração usando uma substituição diferente da de Wang. Eles reduzem o polinômio multivariado $F(x_1, \ldots, x_n)$ para o polinômio bivariado $F_0 = F(x_1, x_2, a_3 x_1 + b_3 x_2 + c_3, \ldots, a_n x_1 + b_n x_2 + c_n)$, onde a_i's, b_i's e c_i's são constantes

aleatórias e usam os fatores bivariados de F_0 para construir os fatores multivariados de F. Desde já gostaríamos de observar que esta substituição poderá aumentar muito o número de termos de F_0 se comparado a F (este assunto será abordado com cuidado no Capítulo 3). Como antes, suponhamos que $F_e x_1^{e_1} \cdots x_n^{e_n}$ seja um termo do polinômio multivariado $F(x_1, \ldots, x_n)$. E como a figura abaixo ilustra, $F_e x_1^{e_1} \cdots x_n^{e_n}$ irá gerar vários termos em F_0.

von zur Gathen
$F_e x_1^{e_1} \cdots x_n^{e_n}$
$\downarrow$
$F_e x_1^{e_1} x_2^{e_2} (a_3 x_1 + b_3 x_2 + c_3)^{e_3} \cdots (a_n x_1 + b_n x_2 + c_n)^{e_n}$

A partir do que foi explanado, podemos observar que se $F(x_1, \ldots, x_n)$ for um polinômio multivariado esparso então, provavelmente, as substituições anteriores farão com que este polinômio perca esta propriedade.

Esta foi a motivação para o nosso algoritmo de fatoração multivariada que apresentaremos no Capítulo 3. Desenvolvemos novas reduções que têm como principal característica manter a esparsidade do polinômio. Resumidamente, dado um polinômio multivariado $F(x_1, \ldots, x_n)$, reduzimos F para os polinômios bivariados

$$F(sX^{a_1}Y^{b_1}, s^d X^{a_2} Y^{b_2}, \ldots, s^{d^{n-1}} X^{a_n} Y^{b_n})$$

e

$$F(s^2 X^{a_1}Y^{b_1}, s^{2d} X^{a_2} Y^{b_2}, \ldots, s^{2d^{n-1}} X^{a_n} Y^{b_n})$$

para certas constantes s, d e constantes aleatórias a_i's e b_i's. E usamos os fatores dos polinômios bivariados para construirmos os fatores multivariados do polinômio $F(x_1, \ldots, x_n)$. Se considerarmos, novamente, $F_e x_1^{e_1} \cdots x_n^{e_n}$ um termo do polinômio multivariado $F(x_1, \ldots, x_n)$, então, como a figura abaixo ilustra, este termo irá gerar apenas um termo quando a redução for aplicada.

Nossa redução
$F_e x_1^{e_1} \cdots x_n^{e_n}$
$\downarrow$
$F_e s^{\ell(e)} X^{a \cdot e} Y^{b \cdot e}$

Na figura anterior, utilizamos $\ell(e) = e_1 + e_2 d + \cdots + e_n d^{n-1}$, $a \cdot e = a_1 e_1 + a_2 e_e + \cdots + a_n e_n$ e $b \cdot e = b_1 e_1 + b_2 e_e + \cdots + b_n e_n$.

Como veremos nesta tese (ver tabela abaixo), nosso método de fatoração mostra-se eficiente quando usado para fatorar polinômios multivariados que possuem apenas fatores esparsos, ou quando usado para extrair fatores esparsos de polinômios que possuem fatores densos e esparsos. Gostaríamos de observar que um polinômio é dito esparso quando possui poucos termos. Uma definição mais concreta de esparsidade é apresentada no Cápitulo 3.

Na tabela a seguir fatoramos polinômios em $\mathbb{Z}[x_1, \ldots, x_5]$ que possuem apenas fatores esparsos. Note que a célula localizada na terceira coluna e quarta linha está vazia. Isto significa que durante nossos experimentos o programa Maple não conseguiu fatorar tais polinômios. A coluna Esparsos contém o número de fatores esparsos do polinômio fatorado.

Grau total de F	Esparsos	tempo(Maple)	tempo(Algoritmo 3.5.2)
72	6	5.87 min	2.93 min
83	7	25.91 min	5.09 min
95	8	23.31 min	12,79 min
106	9	-	31.87 min

No capítulo final estudamos a conexão entre polinômios multivariados e politopos de Newton. Um dos primeiros resultados conhecidos ligando estes dois assuntos foi feito por Ostrowski em 1921 [22]. Ele associou a cada polinômio multivariado F um politopo, dito politopo de Newton. Observou que, se o polinômio

multivariado fosse fatorável, digamos $F = G \cdot H$, então o politopo de Newton associado a F seria decomposto como a soma do politopo de Newton associado a G mais o politopo de Newton associado a H, sendo esta decomposição em relação à soma de Minkowski. Recentemente Shuhong Gao [7, 9] usou propriedades geométricas de politopos para construir famílias de polinômios absolutamente irredutíveis, e, em [25, 26], Shuhong Gao e seus colaboradores acabaram efetivamente fatorando polinômios bivariados usando seus polígonos de Newton associados.

Um resultado probabilístico bem conhecido garante que dados dois polinômios multivariados sobre um domínio integral, eles são quase sempre primos entre si (veja na literatura, por exemplo [17, 33] ou na seção 4.5). Então, quando precisamos calcular o máximo divisor comum entre dois polinômios multivariados, é de grande utilidade aplicarmos um teste preliminar de coprimalidade antes de efetivamente procurarmos o mdc. Essa é nossa principal contribuição no Capítulo 4. Apresentamos um algoritmo que testa coprimalidade entre polinômios multivariados usando propriedades de seus politopos de Newton associados. Nosso teste preliminar mostrou-se realmente eficiente quando usado em polinômios multivariados esparsos com grau grande (veja tabela abaixo). Para polinômios bivariados acabamos apresentando um algoritmo que usa seus polígonos de Newton associados para efetivamente calcular o mdc. Na figura abaixo os coeficientes dos polinômios inteiros usados têm módulo menor que 100.

Polinômios esparsos com 6 variáveis.

$(\text{grau}(f), \text{grau}(g))$	Maple	Teste preliminar	$\text{mdc}(f, g)$
(1000,1000)	0.0057	0.0188	1
(10000,10000)	0.211	0.0204	1
(100000,100000)	23.9003	0.0192	1

Algoritmos recentes para fatoração polinomial: iremos citar alguns dos últimos avanços obtidos nesta década na área de fatoração.

- Shuhong Gao [8]: desenvolveu um novo método para fatoração polinomial bivariada sobre qualquer corpo de característica zero ou característica relativamente grande. Nesta técnica usa-se uma equação diferencial parcial que leva a um sistema de equações lineares. A dimensão do espaço solução do sistema linear é igual ao número de fatores absolutamente irredutíveis do polinômio e qualquer base para o espaço solução leva a uma fatoração completa atráves do cálculo de mdc's e fatoração de polinômios univariados.

- Grégoire Lecerf [19, 20, 21]: resolve o problema combinatorial do levantamento de Hensel. Quando queremos fatorar o polinômio bivariado $F \in \mathbb{K}[x, y]$, primeiro especializamos uma variável. Digamos $x = 0$ e então fatoramos $F(0, y)$. Os fatores univariados são então levantados no anel das séries de potências $\mathbb{K}[[x]][y]$ usando levantamento de Hensel. E então, usando os fatores bivariados levantados em $\mathbb{K}[[x]][y]$ formamos subconjuntos com estes fatores e verificamos quais levam a fatores de $F(x, y)$ em $\mathbb{K}[x, y]$. Este último estágio é a parte combinatorial e Lecerf resolve este problema atráves da resolução de um sistema linear.

- Mark van Hoeij [31]: desenvolve um método similar ao de Lecerf. Mas desta vez para polinômio univariados.

Nossa tese está organizada da seguinte maneira. No Capítulo 2 apresentamos uma demonstração detalhada do Teorema da Irredutibilidade de Hilbert clássico [13], e duas versões efetivas do teorema. Uma feita por Erich Kaltofen [14] e outra por Shuhong Gao [8]. No Capítulo 3 apresentamos um novo algoritmo para fatoração multivariada. E no Capítulo 4 apresentamos um algoritmo que testa coprimalidade entre polinômios multivariados usando propriedades de seus politopos de Newtom associados e para polinômios bivariados acabamos apresentando um algoritmo que usa seus polígonos de Newton associados para efetivamente calcular o mdc.

Gostaríamos de observar que cada capítulo de nossa tese é auto-contido, ou seja, toda teoria e notação usada em cada capítulo está presente no mesmo. Deste modo, cada capítulo pode ser lido separadamente.

2 TEOREMA DA IRREDUTIBILIDADE DE HILBERT

Este capítulo é dedicado ao estudo do Teorema da Irredutibilidade de Hilbert. Esta teoria trata da redução do problema de fatorar polinômios multivariados para o problema de fatorar polinômios bivariados ou univariados. Neste capítulo apresentaremos as versões clássica e efetiva do teorema, detalhando as demonstrações já existentes. Começaremos com a versão clássica feita por Hilbert em [13], a qual nos diz que, dado um polinômio bivariado $f(t, x)$ irredutível sobre $\mathbb{Q}$, existem infinitos inteiros t_0 para os quais o polinômio univariado reduzido $f(t_0, x)$ permanece irredutível sobre $\mathbb{Q}$. Depois estudaremos duas versões efetivas do teorema, uma feita por Erich Kaltofen [14] e outra por Shuhong Gao [8], onde os autores usam reduções distintas. Desta vez, um polinômio multivariado é reduzido para um polinômio bivariado e estas reduções possuem argumentos probabilísticos garantindo que, dado um polinômio multivariado irredutível, este tem alta probabilidade de permanecer irredutível quando reduzido para um polinômio bivariado.

2.1 Introdução

No intuito de fatorar um polinômio multivariado a idéia básica que tem sido usada é a seguinte: primeiro reduzimos o polinômio multivariado $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ para um polinômio bivariado usando a transformação $f_0 = f(a_1x + b_1y + c_1, \ldots, a_nx + b_ny + c_n) \in \mathbb{F}[x, y]$ ou a transformação $f_0 = f(x_1 + c_1, a_2x_1 + c_2, \ldots, a_{n-1}x_1 + c_{n-1}, x_n) \in \mathbb{F}[x_1, x_n]$ ou para um polinômio univariado usando a transformação $f_0 = f(a_1, \ldots, a_{n-1}, x_n) \in \mathbb{F}[x_n]$ com $a_1, b_1, c_1, \ldots, a_n, b_n, c_n \in \mathbb{F}$. E então, usando os fatores de f_0 podemos recuperar os fatores de f usando levantamento de Hensel (ver por exemplo [10, 34, 38]).

Quando reduzimos um polinômio multivariado para um polinômio bivariado ou univariado usando alguma das transformações citadas, estamos interes-

sados em saber se tal redução irá manter a irredutibilidade do polinômio, ou seja, se f é irredutível então f_0 será irredutível ? O estudo de tal propriedade é o que denominamos Teorema da Irredutibilidade de Hilbert.

O teorema da Irredutibilidade de Hilbert clássico encontra-se em seu artigo [13] publicado em 1892. Nesta versão, Hilbert mostra que, dado um polinômio bivariado $f(t, x)$ irredutível sobre $\mathbb{Q}$, existem infinitos inteiros t_0 para os quais o polinômio univariado reduzido $f(t_0, x)$ permanece irredutível sobre $\mathbb{Q}$. Baseado neste resultado, em 1975, um dos primeiros algoritmos efetivos para fatoração de polinômios multivariados foi feito por Wang [38]. Neste algoritmo, um polinômio multivariado $f(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ é reduzido para um univariado, substituindo-se $(n-1)$ de suas variáveis por números inteiros $a_1, \ldots, a_{n-1}$ e então fatorando $f(a_1, \ldots, a_{n-1}, x_n)$ sobre os $\mathbb{Z}$. Com os fatores do polinômio univariado é feito levantamento de Hensel para construirmos os fatores do polinômio multivariado $f(x_1, \ldots, x_n)$. Porém, este tipo de método não possui argumentos probabilísticos que garantam que se o um polinômio multivariado dado for irredutível, este permanecerá irredutível quando reduzido para um polinômio univariado.

Na década de 80, novos tipos de reduções foram apresentadas por Erich Kaltofen e Joachin von zur Gathen. Porém, um polinômio multivariado é reduzido para um polinômio bivariado. O diferencial destas técnicas é que possuem argumentos probabilísticos garantindo que um polinômio multivariado irredutível tem boa probabilidade de permanecer irredutível. Kaltofen transforma um polinômio multivariado $f(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ em um polinômio bivariado $f(x_1 + c_1, a_2x_1 + c_2, \ldots, a_{n-1}x_1 + c_{n-1}, x_n) \in \mathbb{Z}[x_1, x_n]$ com $a_2, \ldots, a_{n-1}$ e $c_1, \ldots, c_{n-1}$ inteiros. Em seu artigo [15], ele mostra que para certos inteiros $a_2, \ldots, a_{n-1}$ e $c_1, \ldots, c_{n-1}$ satisfazendo algumas propriedades e cotas, um polinômio irá manter a irredutibilidade, ou seja, se o polinômio multivariado $f(x_1, \ldots, x_n)$ for irredutível sobre $\mathbb{Z}$, então o polinômio bivariado $f(x_1 + c_1, a_2x_1 + c_2, \ldots, a_{n-1}x_1 + c_{n-1}, x_n)$ será irredutível sobre $\mathbb{Z}$.

Supondo que escolhamos aleatoriamente $a_1, b_1, c_1, \ldots, a_n, b_n, c_n$ de um conjunto finito $S \subset \mathbb{F}$, queremos saber qual a probabilidade de que todos os fatores irredutíveis de um polinômio multivariado $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ de grau total d permaneçam irredutíveis após alguma das substituições apresentadas ter sido feita, ou seja, qual a probabilidade de f e f_0 terem a mesma fatoração padrão. A seguir apresentaremos um histórico sobre estas cotas probabilísticas:

- 1983: von zur Gathen [32] mostra que a probabilidade de $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ e $f(x_1, x_2, a_3x_1 + b_3x_2 + c_3, \ldots, a_nx_1 + b_nx_2 + c_n) \in \mathbb{F}[x, y]$ terem a mesma fatoração padrão é de no mínimo
$$1 - \frac{9^{d^2}}{|S|},$$
onde $\mathbb{F}$ é um corpo arbitrário.

- 1985: Kaltofen [14] mostra que a probabilidade de $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ e $f(x_1 + c_1, a_2x_1 + c_2, \ldots, a_{n-1}x_1 + c_{n-1}, x_n) \in \mathbb{F}[x_1, x_n]$ terem a mesma fatoração padrão é de no mínimo
$$1 - \frac{4d2^d}{|S|},$$
onde $\mathbb{F}$ é um corpo arbitrário.

- 1989: Bajaj e seus colaboradores [3] mostram que a probabilidade de $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ e $f(a_1x + b_1y + c_1, \ldots, a_nx + b_ny + c_n) \in \mathbb{F}[x, y]$ terem a mesma fatoração padrão é de no mínimo
$$1 - \frac{d^4 - 2d^3 + d^2 + d + 1}{|S|},$$
onde $\mathbb{F} = \mathbb{C}$.

- 1995: Kaltofen [16] mostra que a probabilidade de $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ e $f(x + b_1, a_2x + b_2y + c_2, \ldots, a_nx + b_ny + c_n) \in \mathbb{F}[x, y]$ terem a mesma fatoração padrão é de no mínimo
$$1 - \frac{2d^4}{|S|},$$
onde $\mathbb{F}$ é um corpo perfeito.

- 2001: Gao [8] mostra que a probabilidade de $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ e $f(a_1x+b_1y+c_1, \ldots, a_nx+b_ny+c_n) \in \mathbb{F}[x, y]$ terem a mesma fatoração padrão é de no mínimo
$$1 - \frac{2d^3}{|S|},$$
onde $\mathbb{F}$ é um corpo de característica 0 ou maior que $2d^2$.

- 2007: Lecerf [20] mostra que a probabilidade de $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ e $f(a_1x+b_1y+c_1, \ldots, a_nx+b_ny+c_n) \in \mathbb{F}[x, y]$ terem a mesma fatoração padrão é de no mínimo
$$1 - \frac{3d^2}{|S|},$$
onde $\mathbb{F}$ é um corpo de característica 0 ou maior que $d(d-1)+1$.

Por completude gostaríamos de observar que o leitor interessado em saber como é feito o levantamento das tranformações apresentadas pode consultar o artigo [34]. Este capítulo está organizado da seguinte maneira: na Seção 2.2 apresentaremos uma demonstração detalhada do teorema da irredutibilidade de Hilbert clássico baseada principalmente nos livros [18] e [24]. Na Seção 2.3 apresentaremos uma versão efetiva deste teorema, ou seja, com argumentos probabilísticos, feita por Erich Kaltofen em [14]. Terminamos apresentando uma outra versão efetiva feita por Shuhong Gao em [8], cuja demonstração decorre de seu novo método de fatoração polinomial bivariada apresentado no mesmo artigo. Gostaríamos de observar que nas Seções 2.3 e 2.4 deste capítulo mantivemos as notações usadas nos artigos originais.

2.2 Teorema da Irredutibilidade de Hilbert Clássico

Nesta seção apresentaremos uma demonstração detalhada do teorema da irredutibilidade de Hilbert [13], o qual nos diz, que dado um polinômio bivariado $f(t, x)$ irredutível, o polinômio reduzido univariado $f(t_0, x)$ permanecerá irredutível para infinitos inteiros t_0.

Teorema 2.2.1. *Seja $f(t,x)$ um polinômio irredutível sobre $\mathbb{Q}$. Então existem infinitos inteiros t_0 para os quais $f(t_0,x)$ é irredutível sobre $\mathbb{Q}$.*

Dem.: Seja $f(t,x)$ um polinômio irredutível sobre $\mathbb{Q}[t,x]$. Então, pelo Lema de Gauss, este é irredutível sobre $\mathbb{Q}(t)[x]$, onde $\mathbb{Q}(t)$ é o corpo das funções racionais. Sendo assim, vamos representá-lo na forma

$$f(t,x) = a_n(t)x^n + \cdots + a_0(t), \text{ com } a_i(t) \in \mathbb{Q}(t).$$

Seja $\overline{\mathbb{Q}(t)}$ o fecho algébrico de $\mathbb{Q}(t)$. Então fatorando $f(t,x)$ temos

$$f(t,x) = a_n(t)(x-\alpha_1(t))\cdots(x-\alpha_n(t)),$$

onde $\alpha_i(t) \in \overline{\mathbb{Q}(t)}$. Agora, escolha $t_0 \in \mathbb{Z}$ tal que $a_n(t_0) \neq 0$.

Neste momento vamos supor que $f(t_0,x)$ é redutível sobre $\mathbb{Q}$. Então

$f(t_0,x) = a_n(t_0)\cdot g_0(x)\cdot h_0(x)$, com $g_0(x), h_0(x) \in \mathbb{Q}[x]$ e grau$(g_0(x))$, grau$(h_0(x)) \geq 1$.

Deste modo, podemos considerar que, no fecho algébrico $\overline{\mathbb{Q}}$ de $\mathbb{Q}$ temos:

$$g_0(x) = (x-\alpha_1(t_0))\cdots(x-\alpha_s(t_0))$$

e

$$h_0(x) = (x-\alpha_{s+1}(t_0))\cdots(x-\alpha_n(t_0)),$$

com $\alpha_i(t_0) \in \overline{\mathbb{Q}}$. Agora, vamos fazer a seguinte correspondência. Sejam

$$g(t,x) = (x-\alpha_1(t))\cdots(x-\alpha_s(t))$$

e

$$h(t,x) = (x-\alpha_{s+1}(t))\cdots(x-\alpha_n(t)).$$

Então,

$$f(t,x) = a_n(t)\cdot g(t,x)\cdot h(t,x), \text{ com } g(t,x), h(t,x) \in \overline{\mathbb{Q}(t)}[x].$$

Como estamos considerando $f(t,x)$ irredutível sobre $\mathbb{Q}(t)[x]$, então $g(t,x)$ deve ter pelo menos um coeficiente $y(t) \in \overline{\mathbb{Q}(t)}\backslash\mathbb{Q}(t)$. Note que, como $y(t)$ está no fecho

algébrico de $\mathbb{Q}(t)$, então este é algébrico sobre $\mathbb{Q}(t)$. Isto é, existem funções racionais $b_m(t)$, $b_{m-1}(t), \ldots, b_0(t) \in \mathbb{Q}(t)$ tais que

$$b_m(t)y(t)^m + b_{m-1}(t)y(t)^{m-1} + \cdots + b_0(t) = 0. \tag{2.1}$$

Observe que a equação (2.1) representa uma curva C com coeficientes racionais no plano (t, y). Note que o coeficiente $y(t)$ de $g(t, x)$, quando substituído em $t = t_0$, nos leva a um coeficiente $y(t_0)$ de $g_0(x)$. E, como $g_0(x) \in \mathbb{Q}[x]$, temos que $y(t_0) \in \mathbb{Q}$. Concluímos que o ponto racional $(t_0, y(t_0))$ satisfaz

$$b_m(t_0)y(t_0)^m + b_{m-1}(t_0)y(t_0)^{m-1} + \cdots + b_0(t_0) = 0.$$

Então a curva C tem um ponto racional garantido pela redutibilidade de $f(t_0, x)$.

Chave do teorema: Considere todos os polinômios $g_i(t, x)$ formados por combinações dos polinômios $(x - \alpha_1(t)), \ldots, (x - \alpha_n(t))$, onde $\alpha_i(t) \in \overline{\mathbb{Q}(t)}$, tal que $g_i(t, x)$ seja fator de $f(t, x)$ em $\overline{\mathbb{Q}(t)}[x]$. Como $f(t, x)$ é irredutível sobre $\mathbb{Q}(t)[x]$, todo polinômio $g_i(t, x)$ sempre terá coeficientes em $\overline{\mathbb{Q}(t)} \backslash \mathbb{Q}(t)$. Agora, considere todos os coeficientes deste tipo, e como eles são algébricos sobre $\mathbb{Q}(t)$, associamos a estes coeficientes $y_1(t), \ldots, y_l(t)$ curvas algébricas planas $C_1, \ldots, C_l$ com coeficientes racionais. Se escolhermos $t = t_0 \in \mathbb{Z}$ tal que $y_i(t_0) \notin \mathbb{Q}$, ou seja, $(t_0, y_i(t_0))$ não é ponto racional da curva C_i para $1 \leq i \leq l$, então, o polinômio $f(t_0, x)$ será irredutível em $\mathbb{Q}[x]$, pois todo fator de $f(t_0, x)$ sempre terá um coeficiente $y_i(t_0) \notin \mathbb{Q}$.

A partir de agora, iremos mostrar que cada vez que aumentarmos o intervalo de escolha de t_0 nos inteiros, temos menos chance de encontrarmos um inteiro $t = t_0$ que nos leva a um ponto $(t_0, y(t_0))$ racional de uma curva. E assim garantindo que, dado um polinômio $f(t, x)$ irredutível sobre os $\mathbb{Q}$, quase todas as substituições $t = t_0 \in \mathbb{Z}$ deixam $f(t_0, x)$ irredutível sobre $\mathbb{Q}$.

Começamos estudando pontos racionais da curva algébrica plana dada pela equação

$$b_m(t)y(t)^m + b_{m-1}(t)y(t)^{m-1} + \cdots + b_0(t) = 0, \text{ com } b_i(t) \in \mathbb{Q}(t).$$

Multiplicando por um polinômio adequado de $\mathbb{Z}[t]$, podemos considerar que

$$b_m(t)y(t)^m + b_{m-1}(t)y(t)^{m-1} + \cdots + b_0(t) = 0, \text{ com } b_i(t) \in \mathbb{Z}[t]. \tag{2.2}$$

Fazendo a mudança de variável $\widetilde{y}(t) = b_m(t)y(t)$ temos:

$$\widetilde{y}(t)^m + b_{m-1}(t)\widetilde{y}(t)^{m-1} + \cdots + b_1(t)(b_m(t))^{m-2}\widetilde{y} + b_0(t)(b_m(t))^{m-1} = 0. \tag{2.3}$$

Então, podemos estudar pontos racionais sobre a curva de equação

$$y(t)^m + b_{m-1}(t)y(t)^{m-1} + \cdots + b_0(t) = 0, \text{ com } b_i(t) \in \mathbb{Z}[t]. \tag{2.4}$$

Começamos demonstrando a seguinte propriedade relacionada a pontos racionais sobre a curva (2.4).

Propriedade 2.2.1. *Seja $(t_0, y(t_0))$ um ponto racional sobre a curva (2.4). Se $t_0 \in \mathbb{Z}$ então $y(t_0) \in \mathbb{Z}$.*

Dem.: Seja $y(t_0) = \frac{p}{q}$ com $mdc(p, q) = 1$. Então

$$\frac{p^m}{q^m} + b_{m-1}(t_0)\frac{p^{m-1}}{q^{m-1}} + \cdots + b_0(t_0) = 0$$

$$\frac{p^m}{q^m} + \frac{\overbrace{b_{m-1}(t_0)p^{m-1} + \cdots + b_0(t_0)q^{m-1}}^{=c\in\mathbb{Z}}}{q^{m-1}} = 0,$$

logo

$$\frac{p^m}{q^m} = \frac{-c}{q^{m-1}}, \text{ assim } p^m = -cq.$$

Portanto, $q|p^m$, e como $mdc(p, q) = 1$, devemos ter $q = 1$. □

A seguinte propriedade mostra porque podemos estudar apenas pontos inteiros sobre a curva (2.4).

Propriedade 2.2.2. *Seja $(t_0, y(t_0))$ ponto sobre a curva (2.4) tal que $t_0 \in \mathbb{Z}$ e $y(t_0) \notin \mathbb{Z}$. Então $y(t_0) \notin \mathbb{Q}$.*

Dem.: Suponha que $y(t_0) \in \mathbb{Q}$, então $(t_0, y(t_0))$ é ponto racional sobre a curva (2.4). E, como $t_0 \in \mathbb{Z}$, pela Propriedade 2.2.1 temos que $y(t_0) \in \mathbb{Z}$, absurdo. Logo, $y(t_0) \notin \mathbb{Q}$. □

A última propriedade mostra que, dado $t_0 \in \mathbb{Z}$ tal que $(t_0, y(t_0))$ não é ponto inteiro da curva (2.4), então $(t_0, y(t_0))$ também não é ponto racional da curva. Devido a isso, iremos estudar no restante da seção a ocorrência de pontos inteiros sobre curvas algébricas planas. Mais especificamente, a partir de agora iremos estudar a quantidade de pontos inteiros sobre a curva de equação

$$y(t)^m + b_{m-1}(t)y(t)^{m-1} + \cdots + b_0(t) = 0, \text{ com } b_i(t) \in \mathbb{Z}[t]. \tag{2.5}$$

Lembre que estamos considerando $y(t)$ coeficiente de algum fator $g(t,x)$ de $f(t,x)$ com $y(t) \in \overline{\mathbb{Q}(t)} \setminus \mathbb{Q}(t)$, pois $f(t,x)$ é irredutível sobre $\mathbb{Q}(t)[x]$. No intuito de procurarmos pontos inteiros sobre a curva (2.5), estudaremos o comportamento da função algébrica $y(t)$ na vizinhança do ponto $t = \infty$. Pois, conforme pode ser visto em [30], em uma vizinhança do ponto $t = \infty$ a equação (2.5) determina a função algébrica $y(t)$ como uma série de potências do tipo

$$y(t) = a(\sqrt[k]{t})^n + \cdots + b + c(\sqrt[k]{t})^{-1} + \cdots \tag{2.6}$$

com coeficientes complexos, $k \in \mathbb{N}$ e $n \in \mathbb{Z}$. Tal expansão em séries de potência é conhecida como série de Puiseux.

Note que estamos interessados no caso em que existe uma seqüência infinita de inteiros positivos t_i para os quais $y(t_i)$ é um número real, ou melhor $y(t_i)$ é um número inteiro. Pois, se isto não acontecer haverá apenas um número finito de inteiros t_i para os quais $y(t_i)$ é real (de preferência inteiro) e talvez $f(t_i, x)$ redutível.

Para o caso em que exista uma seqüência infinita de inteiros positivos t_i para os quais $y(t_i)$ é um número real mostraremos a seguir que todos os coeficientes da série (2.6) são reais.

Propriedade 2.2.3. *Se existe uma seqüência infinita de inteiros positivos para os quais $y(t_i)$ é real, então os coeficientes da série (2.6) são todos reais.*

Dem.: Vamos supor que nem todos os coeficientes são reais. Considere $dt^{\frac{s}{k}}$ o termo de maior grau $\frac{s}{k}$ tal que $d = d_1 + d_2 i$ com $d_2 \neq 0$. Então para t tendendo ao infinito, os termos complexos de menor grau são pequenos quando comparados a $d_2 t^{\frac{s}{k}} i$ e não

o anulariam. E, assim, para $t \to \infty$ os valores $y(t)$ seriam complexos, contradizendo a hipótese. □

Teorema 2.2.2. *Seja*

$$\varphi(t) = a(\sqrt[k]{t})^n + \cdots + b + c(\sqrt[k]{t})^{-1} + \cdots,$$

onde t é real e a série é real e converge para $t \geq R$. Suponha que $\varphi(t)$ não é um polinômio. Então existem constantes $C > 0$ e $\varepsilon \in (0,1)$ tais que o número de inteiros positivos $t \leq N$ para o qual $\varphi(t) \in \mathbb{Z}$ não excede CN^{ε}.

Dem.: Primeiro, observe que a expansão da m-ésima derivada de $\varphi(t)$ não contém termos do tipo t^{ν} com $\nu > \frac{n}{k} - m$. Então podemos escolher um inteiro positivo $m \geq 1$ tal que

$$\varphi^m(t) \sim ct^{-\mu}, \tag{2.7}$$

com $c \neq 0$ e $\mu > 0$ ao $t \to \infty$. Para terminarmos a demonstração do teorema 2.2.2 precisaremos do seguinte lema.

Lema 2.2.1. *Existem constantes positivas c_1 e α tais que, se T é suficientemente grande, então o intervalo $[T, T + c_1T^{\alpha}]$ não contém mais que m inteiros positivos t para os quais $\varphi(t) \in \mathbb{Z}$.*

Dem.: Sejam $t_1 < t_2 < \cdots < t_{m+1}$ inteiros. E, considere o polinômio interpolador de Lagrange

$$f(t) = \sum_{i=1}^{m+1} \varphi(t_i) \frac{(t-t_1)\cdots(t-t_{i-1})(t-t_{i+1})\cdots(t-t_{m+1})}{(t_i-t_1)\cdots(t_i-t_{i-1})(t_i-t_{i+1})\cdots(t_i-t_{m+1})}.$$

Note que a função $\varphi - f$ se anula nos pontos $t_1, \ldots, t_{m+1}$, e então pelo Teorema de Rolle existe $\xi \in (t_1, t_{m+1})$ tal que

$$\varphi^m(\xi) = f^m(\xi) = m! \sum_{1 \leq i \leq m+1} \frac{\varphi(t_i)}{(t_i-t_1)\cdots(t_i-t_{i-1})(t_i-t_{i+1})\cdots(t_i-t_{m+1})}.$$

Deste modo, podemos observar que $\varphi^m(\xi)$ é um número racional cujo denominador não excede

$$\prod_{1 \leq i < j \leq m+1} (t_i - t_j) < (t_{m+1} - t_1)^{C_2^m} = (t_{m+1} - t_1)^{\frac{m(m+1)}{2}}.$$

Então

$$|\varphi^m(\xi)| = \frac{p}{q} \geq p(t_{m+1} - t_1)^{-\frac{m(m+1)}{2}}. \tag{2.8}$$

Como foi observado anteriormente na equação (2.7), se t_1 for suficientemente grande, podemos considerar que

$$0 < |\varphi^m(\xi)| \leq ct_1^{-\mu}. \tag{2.9}$$

Agora, seja $\Delta T = t_{m+1} - t_1$. Então de (2.8) e (2.9) temos que

$$(\frac{|p|}{c})^{\frac{2}{m(m+1)}} t_1^{\frac{2\mu}{m(m+1)}} \leq \Delta T.$$

Escolhendo $c_1 = (\frac{|p|}{c})^{\frac{2}{m(m+1)}}$ e $\alpha = \frac{2\mu}{m(m+1)}$. Temos que

$$\Delta T \geq c_1 T^{\alpha}.$$

□

Retomando à prova do Teorema 2.2.2. Escolha $\varepsilon = \frac{1}{1+\alpha}$. Então $\varepsilon \in (0,1)$. Vamos separar o intervalo $[1, N]$ em dois subintervalos $[1, N^{\varepsilon}]$ e $[N^{\varepsilon}, N]$. Seja N suficientemente grande, tal que de acordo com o Lema 2.2.1 cada subintervalo de comprimento $c_1(N^{\varepsilon})^{\alpha}$ de $[N^{\epsilon}, N]$ não contenha mais que m inteiros positivos para os quais $\varphi(t) \in \mathbb{Z}$. Então, podemos observar que o número total de tais inteiros positivos no internalo $[1, N]$ não excede

$$N^{\varepsilon} + m\left(\frac{N - N^{\varepsilon}}{c_1 N^{\varepsilon\alpha}}\right) \leq N^{\varepsilon}\left(1 + \frac{m}{c_1}\right).$$

Agora, basta escolher $C = 1 + \frac{m}{c_1}$. □

Para terminarmos a demonstração do teorema da irredutibilidade de Hilbert, vamos considerar $B(N)$ o número total de inteiros positivos $t \leq N$ para os quais $\varphi(t) \in \mathbb{Z}$. Note que:

$$B(N) \leq CN^{\varepsilon}$$

e

$$\frac{B(N)}{N} \leq \frac{CN^{\varepsilon}}{N} \to 0 \text{ ao } N \to \infty.$$

Este resultado mostra que para quase todo $t_0 \in \mathbb{N}$, temos que $\varphi(t) \notin \mathbb{Z}$, ou seja, pela Propriedade 2.2.2 temos que $\varphi(t) \notin \mathbb{Q}$. Mantendo $f(t_0, x)$ irredutível em $\mathbb{Q}[x]$. □

2.2.1 Um x, vários t's

Nesta seção iremos observar que, dado um polinômio $f(t_1, \ldots, t_n, x) \in \mathbb{Q}[t_1, \ldots, t_n, x]$ irredutível sobre $\mathbb{Q}$, existem infinitos inteiros $a_1, \ldots, a_n$ tais que $f(a_1, \ldots, a_n, x)$ é irredutível em $\mathbb{Q}[x]$. Tal resultado motivou o algoritmo feito por Paul Wang em [38].

Seja $f(t_1, \ldots, t_n, x) \in \mathbb{Q}[t_1, \ldots, t_n, x]$ irredutível sobre $\mathbb{Q}$. Mostraremos que existem infinitos $a \in \mathbb{Z}$ tal que $f(a, t_2, \ldots, t_n, x)$ é irredutível sobre $\mathbb{Q}$. Lembre que, pelo lema de Gauss, temos que $f(t_1, \ldots, t_n, x)$ é irredutível sobre $\mathbb{Q}(t_1)[t_2, \ldots, t_n, x]$.

Fatorando $f(t_1, \ldots, t_n, x)$ sobre $\overline{\mathbb{Q}(t_1)}[t_2, \ldots, t_n, x]$, temos

$$f(t_1, \ldots, t_n, x) = f_1(t_1, \ldots, t_n, x) \cdots f_m(t_1, \ldots, t_n, x)$$

com $f_i(t_1, \ldots, t_n, x) \in \overline{\mathbb{Q}(t_1)}[t_2, \ldots, t_n, x]$ para $1 \leq i \leq m$. Agora, seja $t_1 = a \in \mathbb{Z}$. Vamos considerar que $f(a, t_2, \ldots, t_n, x)$ é redutível. Então

$$f(a, t_2, \ldots, t_n, x) = g_a(t_2, \ldots, t_n, x) \cdot h_a(t_2, \ldots, t_n, x)$$

com g_a, $h_a \in \mathbb{Q}[t_2. \ldots, t_n, x]$.

Podemos considerar que em $\overline{\mathbb{Q}}[t_2, \ldots, t_n, x]$ temos

$$g_a(t_2, \ldots, t_n, x) = f_1(a, t_2, \ldots, t_n, x) \cdots f_s(a, t_2, \ldots, t_n, x)$$

e

$$h_a(t_2, \ldots, t_n, x) = f_{s+1}(a, t_2, \ldots, t_n, x) \cdots f_m(a, t_2, \ldots, t_n, x).$$

Vamos considerar a seguinte correspondência:

$$g(t_1, \ldots, t_n, x) = f_1(t_1, \ldots, t_n, x) \cdots f_s(t_1, \ldots, t_n, x)$$

e

$$h(t_1, \ldots, t_n, x) = f_{s+1}(t_1, \ldots, t_n, x) \cdots f_m(t_1, \ldots, t_n, x).$$

Então

$$f(t_1, \ldots, t_n, x) = g(t_1, \ldots, t_n, x) \cdot h(t_1, \ldots, t_n, x)$$

com $g(t_1, \ldots, t_n, x)$ e $h(t_1, \ldots, t_n, x) \in \overline{\mathbb{Q}(t_1)}[t_2, \ldots, t_n, x]$.

Sem perda de generalidade, notemos que $g(t_1, \ldots, t_n, x) \in \overline{\mathbb{Q}(t_1)}[t_2, \ldots, t_n, x]$ possui um coeficiente $y(t_1) \in \overline{\mathbb{Q}(t_1)} \backslash \mathbb{Q}(t_1)$ devido a irredutibilidade de $f(t_1, \ldots, t_n, x)$. Como feito na seção 2.2 associamos a $y(t_1)$ uma curva algébrica plana que possui um ponto racional garantido pela redutibilidade de $f(a, t_2, \ldots, t_n, x)$. A partir daqui a demonstração segue como feito anteriormente no teorema da irredutibilidade de Hilbert clássico. Deste modo, podemos indutivamente passar de um polinômio de $n+1$ variáveis para um polinômio univariado.

2.3 Teorema da Irredutibilidade de Hilbert Efetivo: Erich Kaltofen

Nesta seção apresentaremos argumentos probabilísticos devidos a Erich Kaltofen [14] garantindo que um polinômio multivariado irredutível $f(x_1, \ldots, x_\nu)$ quando reduzido para um polinômio bivariado $f(x_1 + w_1, c_2 x_1 + w_2, \ldots, c_{\nu-1} x_1 + w_{\nu-1}, x_\nu)$ permanece irredutível com alta probabilidade. Ou seja, este é um bom teste de irredutibilidade a ser aplicado a um polinômio multivariado antes de tentarmos fatorá-lo efetivamente.

Notação: $\mathbb{Z}$ denota inteiros, $\mathbb{Q}$ os racionais e $\mathbb{C}$ os complexos. D denota um domínio integral. $D[x_1, \ldots, x_\nu]$ denota os polinômios em $x_1, \ldots, x_\nu$ sobre D, $D(x_1, \ldots, x_\nu)$ o correspondente corpo quociente. $\text{grau}_{x_1}(f)$ o maior grau de x_1 em $f \in D[x_1, \ldots, x_\nu]$, $\text{grau}_{x_1,x_2}(f)$ o maior grau total de f nas variáveis x_1 e x_2, e $\text{grau}(f) = \text{grau}_{x_1,\ldots,x_\nu}(f)$ o grau total de f. O coeficiente da maior potência de x_ν em f é chamado coeficiente líder de f em x_ν e denotado por $\text{ldcf}_{x_\nu}(f)$. Diremos que f é mônico em x_ν se $\text{ldcf}_{x_\nu}(f)$ é um elemento invertível de D. Um resultado conhecido nos diz que $D[x_1, \ldots, x_\nu]$ é um domínio de fatoração única (DFU) se D é um DFU. Neste caso o conteúdo de $f \in D[x_1, \ldots, x_\nu]$ em x_ν, $\text{cont}_{x_\nu}(f)$, é o máximo divisor comum (mdc) de todos os coeficientes de $f(x_\nu)$ como elementos de $D[x_1, \ldots, x_{\nu-1}]$. A parte primitiva de f em x_ν é definida como

$$\mathrm{pp}_{x_\nu}(f) = \frac{1}{\mathrm{cont}_{x_\nu(f)}} f$$

e diremos que f é primitiva em x_ν se $f = \mathrm{pp}_{x_\nu}(f)$. Também observamos que o grau total de um fator de f com respeito a qualquer conjunto de variáveis é menor que ou igual ao grau total da f naquele conjunto de variáveis.

O lema a seguir mostra que a probabilidade de um conjunto de pontos ser um zero de um polinômio multivariado é muito pequena.

Lema 2.3.1. *Assuma que $t(y_1, \ldots, y_\nu) \in D[y_1, \ldots, y_\nu]$ é um polinômio não nulo de grau total d e seja $S \subseteq D$. Então a probabilidade*

$$P(t(c_1, \ldots, c_\nu) = 0 \mid c_i \in S,\ 1 \leq i \leq \nu) \leq \frac{d}{card(S)}. \tag{2.10}$$

Dem.: A prova será feita por indução sobre o número de variáveis ν. Para $\nu = 1$, como $\mathrm{grau}(t(y_1)) = d$ então t possui no máximo d raízes em D. logo

$$P(t(c_1) = 0 \mid c_1 \in S) \leq \frac{d}{card(S)}.$$

Agora, assuma que a afirmação é verdadeira para $\nu - 1$ variáveis. Seja $l(y_1, \ldots, y_{\nu-1}) = ldcf_{y_\nu}(t)$, $n = \mathrm{grau}_{y_\nu}(t)$. Então $\mathrm{grau}(l) + n \leq d$, assim $\mathrm{grau}(l) \leq d - n$. E, como l tem $\nu - 1$ variáveis, por hipótese de indução temos

$$P(l(c_1, \ldots, c_{\nu-1}) = 0 \mid c_i \in S, 1 \leq i \leq \nu - 1) \leq \frac{d-n}{card(S)}.$$

Agora, se $l(c_1, \ldots, c_{\nu-1}) \neq 0$ então $\mathrm{grau}(t(c_1, \ldots, c_{\nu-1}, y_\nu)) = n$ e assim $t(c_1, \ldots, c_{\nu-1}, y_\nu)$ possui no máximo n raízes em D. Logo

$$P(t(c_1, \ldots, c_{\nu-1}, c_\nu) = 0 \mid c_\nu \in S) \leq \frac{n}{card(S)}.$$

Podemos concluir que

$$P(t(c_1, \ldots, c_\nu) = 0) = \underbrace{P(t = 0 \mid l = 0)}_{\leq 1} \cdot \underbrace{P(l = 0)}_{\leq \frac{d-n}{card(S)}} + \underbrace{P(t = 0 \mid l \neq 0)}_{\leq \frac{n}{card(S)}} \cdot \underbrace{P(l \neq 0)}_{\leq 1}$$

$$P(t(c_1, \ldots, c_\nu) = 0) \leq \frac{d-n}{card(S)} + \frac{n}{card(S)} = \frac{d}{card(S)}.$$

□

No próximo lema apresentaremos uma probabilidade em relação à preservação do grau da variável principal e a propriedade de manter-se livre de quadrados após a redução.

Lema 2.3.2. *Seja* $f(y_1, \ldots, y_\nu, x) \in F[y_1, \ldots, y_\nu, x]$ *irredutível em* $F(y_1, \ldots, y_\nu)[x]$, *F um corpo, e assumamos, além disso que* $\frac{\partial f}{\partial x} \neq 0$. *Seja* $n = grau_x(f)$, $d = grau_{y_1,\ldots,y_\nu}(f)$ *e* $a_n(y_1, \ldots, y_\nu) = ldcf_x(f)$. *Selecionemos* $w_1, \ldots, w_\nu$ *aleatoriamente de um subconjunto* $S \subseteq F$. *Então a probabilidade*

$$P(a_n(w_1, \ldots, w_\nu) = 0 \text{ ou } f(w_1, \ldots, w_\nu, x) \text{ não ser livre de quadrados}) \leq \frac{(2n+1)d}{card(S)}. \tag{2.11}$$

Dem.: Primeiro, observe que $mdc(f, g) = 1$ pois f é irredutível e $\frac{\partial f}{\partial x} \neq 0$. Então $res_x(f, \frac{\partial f}{\partial x}) \neq 0$. Seja $\Delta_f(y_1, \ldots, y_\nu) = res_x(f, \frac{\partial f}{\partial x}) \in F[y_1, \ldots, y_\nu]$, e note que grau$(\Delta_f) \leq (2n-1)d$. Agora, selecione aleatoriamente $w_1, \ldots, w_\nu \in S \subset F$ tal que $\Delta_f(w_1, \ldots, w_\nu) \neq 0$, e como $\Delta_f(w_1, \ldots, w_\nu) = res(f(w_1, \ldots, w_\nu, x), \frac{\partial}{\partial x}(f(w_1, \ldots, w_\nu, x))) \neq 0$, então $f(w_1, \ldots, w_\nu, x)$ é livre de quadrados. Considere também $\frac{\partial f}{\partial x} = ka_k x^{k-1} + \cdots + a_1$, com $ka_k \neq 0$ e $a_i \in F[y_1, \ldots, y_\nu]$ para $1 \leq i \leq k$. Podemos concluir que

$$P(a_n(w_1, \ldots, w_\nu) \neq 0 \text{ e } f(w_1, \ldots, w_\nu, x) \text{ ser livre de quadrados}) =$$

$$P((a_n a_k \Delta_f)(w_1, \ldots, w_\nu) \neq 0).$$

Note que

$$P((a_n a_k \Delta_f)(w_1, \ldots, w_\nu) \neq 0) = 1 - P((a_n a_k \Delta_f)(w_1, \ldots, w_\nu) = 0).$$

E por 3.5.1 temos

$$P((a_n a_k \Delta_f)(w_1, \ldots, w_\nu) = 0) \leq \frac{(2n+1)d}{card(S)}.$$

Então

$$P(a_n(w_1, \ldots, w_\nu) \neq 0 \text{ e } f(w_1, \ldots, w_\nu, x) \text{ ser livre de quadrados}) \geq 1 - \frac{(2n+1)d}{card(S)}.$$

□

O próximo resultado, junto com o Lema 2.3.1, dá a probabilidade de um polinômio primitivo manter-se primitivo após a redução.

Lema 2.3.3. *Sejam $f_1, \ldots, f_k \in F[x_1, \ldots, x_\nu]$, F um corpo, com grau$(f_i) \leq \delta$ para $1 \leq i \leq k$ e $mdc(f_1, \ldots, f_k) = 1$. Além disso, assuma que $f_1(0, \ldots, 0) \neq 0$. Então existe um polinômio $\Delta(y_2, \ldots, y_\nu) \in F[y_2, \ldots, y_\nu]$ com grau$(\Delta) \leq 2\delta^2$ tal que, para quaisquer elementos $c_2, \ldots, c_\nu \in F$ com $\Delta(c_2, \ldots, c_\nu) \neq 0$, temos*

$$mdc_{1 \leq i \leq k}(f_i(x_1, c_2x_1, \ldots, c_\nu x_1)) = 1.$$

Dem.: A igualdade $mdc_{1 \leq i \leq k}(f_i(x_1, y_2x_1, \ldots, y_\nu x_1)) = 1$ segue do fato que $x_1 \nmid f_1(x_1, y_2x_1, \ldots, y_\nu x_1)$. Além disso podemos encontrar polinômios $s_1, \ldots, s_k \in F(y_2, \ldots, y_\nu)[x_1]$ com $\text{grau}_{x_1}(s_i) < \delta$ tais que

$$1 = \sum_{i=1}^{k} s_i(y_2, \ldots, y_\nu, x_1) f_i(x_1, y_2x_1, \ldots, y_\nu x_1).$$

Esta identidade leva-nos a um sistema linear sobre $F(y_2, \ldots, y_\nu)$ em 2δ equações e $k\delta$ incógnitas de s_i. Assim podemos encontrar uma solução em $\frac{1}{\Delta(y_2, \ldots, y_\nu)} F[y_2, \ldots, y_\nu]$ onde Δ é o determinante de uma matriz $2m \times 2m$ com $m \leq \delta$, dos coeficientes das potências de x_1 em $f_i(x_1, y_2x_1, \ldots, y_\nu x_1)$. Além disso grau$(\Delta) \leq 2\delta^2$ e qualquer escolha de $c_2, \ldots, c_\nu$ com $\Delta(c_2, \ldots, c_\nu) \neq 0$ implica $mdc_{1 \leq i \leq k}(f_i(x_1, c_2x_1, \ldots, c_\nu x_1)) = 1$ já que $\sum_{i=1}^{k} s_i(c_2, \ldots, c_\nu, x_1) f_i(x_1, c_2x_1, \ldots, c_\nu x_1) = 1$. □

Vamos adotar a seguinte notação vetorial: $\underline{k} \equiv (k_1, \ldots, k_\nu)$, $\underline{0} \equiv (0, \ldots, 0)$, $\underline{y}^{\underline{k}} \equiv y_1^{k_1} \cdots y_\nu^{k_\nu}$, $\underline{k} \pm \underline{k}' \equiv (k_1 \pm k_1', \ldots, k_\nu \pm k_\nu')$, $\underline{k} \leq \underline{k}'$ se, para todo i, $k_i \leq k_i'$, e finalmente $|\underline{k}| \equiv k_1 + \cdots + k_\nu$ se $\underline{k} \geq \underline{0}$, e $-\infty$ caso contrário.

Lema 2.3.4. *Considere $f(y_1, \ldots, y_\nu, x) \in F[y_1, \ldots, y_\nu, x]$ primitivo em relação a x, com grau$_x(f)$ = grau$(f_{\underline{0}}(x) = f(0, \ldots, 0, x)) = n$. Se $f(0, \ldots, 0, x)$ é irredutível, então $f(y_1, \ldots, y_\nu, x)$ é irredutível.*

Dem.: Vamos supor por contradição que $f(y_1, \ldots, y_\nu, x)$ é redutível. Então $f(y_1, \ldots, y_\nu, x)$ = $cont_x(f) \cdot pp_x(f)$. Logo, $pp_x(f) = gh$ com $\text{grau}_x(g), \text{grau}_x(h) \geq 1$, pois $cont_x(f) =$

1. Fazendo a redução temos $f_{\underline{0}}(x) = g_{\underline{0}}(x)h_{\underline{0}}(x)$. Por hipótese $\text{grau}_x(f) = \text{grau}(f_{\underline{0}}(x) = f(0,\ldots,0,x)) = n$, então $\text{grau}_x(g) = \text{grau}(g_{\underline{0}}(x) \geq 1$ e $\text{grau}_x(h) = \text{grau}(h_{\underline{0}}) \geq 1$. Ou seja, $f(0,\ldots,0,x)$ é redutível, uma contradição. □

Teorema 2.3.1 (Lema de Hensel)**.** *Seja $f(y_1,\ldots,y_\nu,x) \in F[y_1,\ldots,y_\nu,x]$, F um corpo, de grau n em x, $l(y_1,\ldots,y_\nu) = ldcf_x(f)$ tal que $l_{\underline{0}} = l(0,\ldots,0) \neq 0$ e $f_{\underline{0}}(x) = f(0,\ldots,0,x)$ é livre de quadrados. Suponha*

$$(l_{\underline{0}}x^i + g_{\underline{0}}(x))(l_{\underline{0}}x^j + h_{\underline{0}}(x)) = l_{\underline{0}}f_{\underline{0}}(x),\ i+j=n$$

é uma fatoração não trivial de $l_{\underline{0}}f_{\underline{0}}$ em $F[x]$. Então, para todo $\underline{k}$ com $|\underline{k}| \geq 1$, existem únicos polinômios $g_{\underline{k}}(x)$, $h_{\underline{k}}(x) \in F[x]$ com $grau(g_{\underline{k}}) < i$, $grau(h_{\underline{k}}) < j$ tal que

$$l(y_1,\ldots,y_\nu)f(y_1,\ldots,y_\nu,x) = (l(y_1,\ldots,y_\nu)x^i + \sum_{\underline{k}\geq\underline{0}} g_{\underline{k}}(x)\underline{y}^{\underline{k}})\cdot(l(y_1,\ldots,y_\nu)x^j + \sum_{\underline{k}\geq\underline{0}} h_{\underline{k}}(x)\underline{y}^{\underline{k}}) \tag{2.12}$$

Dem.: Iremos fazer a demonstração por indução em $\underline{k}$. Note que para $\underline{k} = \underline{0}$ a afirmação é verdadeira por hipótese. Vamos fixar $\underline{k}'$ com $|\underline{k}'| = m$. Então, por hipótese de indução, na equação (2.12) conhecemos todos $g_{\underline{k}}(x)$ e $h_{\underline{k}}(x)$ tais que $0 \leq \underline{k} \leq \underline{k}'$ e $0 \leq |\underline{k}| < m$.

Considere $l(y_1,\ldots,y_\nu) = \sum_{\underline{k}\geq\underline{0}} l_{\underline{k}}\underline{y}^{\underline{k}}$ com $l_{\underline{k}} \in F$ e $lf - l^2x^n = \sum_{\underline{k}\geq\underline{0}} f_{\underline{k}}(x)\underline{y}^{\underline{k}}$, note que $f_{\underline{k}}(x) \in F[x]$ e $\text{grau}(f_{\underline{k}}(x)) < n$.

Iremos procurar os polinômios $g_{\underline{k}'}(x)$, $h_{\underline{k}'}(x)$ utilizando o coeficiente de $\underline{y}^{\underline{k}'}$ na equação

$$\sum_{\underline{k}\geq\underline{0}} f_{\underline{k}}(x)\underline{y}^{\underline{k}} = lf - l^2x^n.$$

Usando (2.12) na equação acima temos

$$\sum_{\underline{k}\geq\underline{0}} f_{\underline{k}}(x)\underline{y}^{\underline{k}} = l^2x^n + lx^i\sum_{\underline{k}\geq\underline{0}} h_{\underline{k}}(x)\underline{y}^{\underline{k}} + lx^j\sum_{\underline{k}\geq\underline{0}} g_{\underline{k}}(x)\underline{y}^{\underline{k}} + (\sum_{\underline{k}\geq\underline{0}} g_{\underline{k}}(x)\underline{y}^{\underline{k}})(\sum_{\underline{k}\geq\underline{0}} h_{\underline{k}}(x)\underline{y}^{\underline{k}}) - l^2x^n$$

$$\sum_{\underline{k}\geq\underline{0}} f_{\underline{k}}(x)\underline{y}^{\underline{k}} = x^i(\sum_{\underline{k}\geq\underline{0}} l_{\underline{k}}\underline{y}^{\underline{k}})(\sum_{\underline{k}\geq\underline{0}} h_{\underline{k}}(x)\underline{y}^{\underline{k}}) + x^j(\sum_{\underline{k}\geq\underline{0}} l_{\underline{k}}\underline{y}^{\underline{k}})(\sum_{\underline{k}\geq\underline{0}} g_{\underline{k}}(x)\underline{y}^{\underline{k}}) + (\sum_{\underline{k}\geq\underline{0}} g_{\underline{k}}(x)\underline{y}^{\underline{k}})(\sum_{\underline{k}\geq\underline{0}} h_{\underline{k}}(x)\underline{y}^{\underline{k}}).$$

Isolando o coeficiente de $\underline{y}^{\underline{k}'}$ na expansão acima temos

$$f_{\underline{k}'}(x) = x^i \sum_{\underline{0}\leq\underline{s}\leq\underline{k}'} l_{\underline{k}'-\underline{s}} h_{\underline{s}} x^j \sum_{\underline{0}\leq\underline{s}\leq\underline{k}'} l_{\underline{k}'-\underline{s}} g_{\underline{s}} + \sum_{\underline{0}\leq\underline{s}\leq\underline{k}'} g_{\underline{k}'-\underline{s}} h_{\underline{s}}.$$

$$f_{\underline{k}'}(x) = x^i l_{\underline{k}'} h_{\underline{0}} + x^i l_{\underline{0}} h_{\underline{k}'} + x^i \sum_{\underline{0}\leq\underline{s}\leq\underline{k}',0<|\underline{s}|<m} l_{\underline{k}'-\underline{s}} h_{\underline{s}} + x^j l_{\underline{k}'} g_{\underline{0}} + x^j l_{\underline{0}} g_{\underline{k}'}$$

$$+x^j \sum_{\underline{0}\leq\underline{s}\leq\underline{k}',0<|\underline{s}|<m} l_{\underline{k}'-\underline{s}} g_{\underline{s}} + g_{\underline{k}'} h_{\underline{0}} + g_{\underline{0}} h_{\underline{k}'} + \sum_{\underline{0}\leq\underline{s}\leq\underline{k}',0<|\underline{s}|<m} g_{\underline{k}'-\underline{s}} h_{\underline{s}}.$$

Então

$$f_{\underline{k}'}(x) = \underbrace{x^i l_{\underline{k}'} h_{\underline{0}}(x) + x^j l_{\underline{k}'} g_{\underline{0}}(x) + \sum_{\underline{0}\leq\underline{s}\leq\underline{k}',0<|\underline{s}|<m} (x^i l_{\underline{k}'-\underline{s}} h_{\underline{s}}(x) + x^j l_{\underline{k}'-\underline{s}} g_{\underline{s}}(x) + g_{\underline{k}'-\underline{s}} h_{\underline{s}}(x))}_{=S_{\underline{k}'}(x)}$$

$$+x^i l_{\underline{0}} h_{\underline{k}'}(x) + x^j l_{\underline{0}} g_{\underline{k}'}(x) + g_{\underline{k}'}(x) h_{\underline{0}} + g_{\underline{0}} h_{\underline{k}'}(x).$$

Note que o polinômio $S_{\underline{k}'}(x)$ é conhecido e é único, por hipótese de indução. Então

$$f_{\underline{k}'}(x) - S_{\underline{k}'}(x) = g_{\underline{k}'}(x)(x^j l_{\underline{0}} + h_{\underline{0}}(x)) + h_{\underline{k}'}(x)(x^i l_{\underline{0}} + g_{\underline{0}}(x)). \tag{2.13}$$

Já que $f_{\underline{0}}$ é livre de quadrados, temos

$$mdc(x^j l_{\underline{0}} + h_{\underline{0}}(x), x^i l_{\underline{0}} + g_{\underline{0}}(x)) = 1.$$

Então, pelo algoritmo de Euclides estendido, existem únicos polinômios $a(x)$, $b(x)$ com grau$(a) < j$ e grau$(b) < i$ tais que

$$a(x)(x^i l_{\underline{0}} + g_{\underline{0}}(x)) + b(x)(x^j l_{\underline{0}} + h_{\underline{0}}(x)) = 1.$$

Multiplicando a equação anterior por $f_{\underline{k}'}(x) - S_{\underline{k}'}(x)$ temos

$$(a(x)(f_{\underline{k}'}(x)-S_{\underline{k}'}(x)))(x^i l_{\underline{0}}+g_{\underline{0}}(x))+(b(x)(f_{\underline{k}'}(x)-S_{\underline{k}'}(x)))(x^j l_{\underline{0}}+h_{\underline{0}}(x)) = f_{\underline{k}'}(x)-S_{\underline{k}'}(x).$$

Logo,

$$h_{\underline{k}'}(x) = a(x)(f_{\underline{k}'}(x) - S_{\underline{k}'}(x))$$

e

$$g_{\underline{k}'}(x) = b(x)(f_{\underline{k}'}(x) - S_{\underline{k}'}(x)).$$

Então, o algoritmo de Euclides estendido garante a existência de $g_{\underline{k}'}(x)$ e $h_{\underline{k}'}(x)$. A igualdade (2.13) leva-nos a um sistema que tem solução com $\text{grau}(g_{\underline{k}'}(x)) < i$ e $\text{grau}(h_{\underline{k}'}(x)) < j$ pois $\text{grau}(f_{\underline{k}'}(x) - S_{\underline{k}'}(x)) < n$, implicando que estes polinômios são únicos. □

Teorema 2.3.2 (Teorema da irredutibilidade de Hilbert efetivo)**.** *Seja* $f(x_1, \ldots, x_\nu) \in F[x_1, \ldots, x_\nu]$, F *um corpo, de grau total* δ *e irredutível sobre* F. *Assumamos que* $\frac{\partial f}{\partial x_\nu} \neq 0$. *Seja* $S \subseteq F$ *e sejam* $c_2, \ldots, c_{\nu-1}, w_1, \ldots, w_{\nu-1}$ *elementos aleatórios de* S. *Então a probabilidade*

$$P(f(x_1+w_1, c_2x_1+w_1, \ldots, c_{\nu-1}x_1+w_{\nu-1}, x_2) \text{ tornar-se redutível em } F[x_1, x_2]) \leq \frac{4\delta 2^\delta}{card(S)}.$$

Dem.: Considerando $\text{grau}_{x_\nu}(f) = n$ e $\text{grau}_{x_1,\ldots,x_{\nu-1}}(f) = d$, pelo lema 2.3.2 temos a seguinte probabilidade

$$P(f(w_1, \ldots, w_{\nu-1}, x) \text{ ser livre de quadrados e } \text{grau}(f(w_1, \ldots, w_{\nu-1}, x)) = n) \geq 1 - \frac{(2n+1)d}{card(S)}.$$

Vamos assumir que este é o caso. Isto nos leva a seguinte mudança de variável.

$$\begin{array}{cc} f(x_1, \ldots, x_{\nu-1}, x_\nu) & ldcf_{x_\nu}(x_1, \ldots, x_{\nu-1}) \\ \downarrow & \downarrow \\ f(y_1 + w_1, \ldots, y_{\nu-1} + w_{\nu-1}, x) & l(y_1, \ldots, y_{\nu-1}) = ldcf_{x_\nu}(y_1 + w_1, \ldots, y_{\nu-1} + w_{\nu-1}) \end{array}.$$

Escrevendo

$$g(y_1, \ldots, y_{\nu-1}, x) = l(y_1, \ldots, y_{\nu-1}) f(y_1 + w_1, \ldots, y_{\nu-1} + w_{\nu-1}, x),$$

pois assim $g(0, \ldots, 0, x)$ é livre de quadrados. Escolhendo aleatoriamente $c_2, \ldots, c_{\nu-1} \in S$, faremos a seguinte mudança de variável deixando o polinômio bivariado

$$\begin{array}{cc} g(y_1, \ldots, y_{\nu-1}, x) & l(y_1, \ldots, y_{\nu-1}) \\ \downarrow & \downarrow \\ g_{\underline{c}}(y_1, x) = g(y_1, c_2y_1, \ldots, c_{\nu-1}y_1, x) & l_{\underline{c}}(y_1) = l(y_1, c_2y_1, \ldots, c_{\nu-1}y_1) \end{array}.$$

No lema seguinte precisaremos de um resultado conhecido como Teorema de Puiseux (pode ser encontrado em [35]), o qual nos diz que o domínio das séries de potências formais em $y_1, \ldots, y_{\nu-1}$ sobre F denotado por $F[[y_1, \ldots, y_{\nu-1}]]$ contém o fecho algébrico de $F[y_1, \ldots, y_{\nu-1}]$.

Lema 2.3.5. *Cada fator* $\widehat{h}(y_1, x) \in F[[y_1]][x]$ *de* $g_{\underline{c}}$ *com* $ldcf_x(\widehat{h}) = l_{\underline{c}}$ *vem de um fator* $h \in F[[y_1, \ldots, y_{\nu-1}]][x]$ *de* g *com* $ldcf_x(h) = l$, *tal que*

$$\widehat{h}(y_1, x) = h(y_1, c_2y_1, \ldots, c_{\nu-1}y_1, x) = h_{\underline{c}}(y_1, x).$$

Dem.: Primeira demonstração.

Note que, trabalhando no domínio das séries de potências formais temos as seguintes fatorações

$$g(y_1, \ldots, y_{\nu-1}, x) = [l(y_1, \ldots, y_{\nu-1})]^2 \prod_{1 \leq i \leq n} (x - \alpha_i(y_1, \ldots, y_{\nu-1})), \tag{2.14}$$

com $\alpha_i(y_1, \ldots, y_{\nu-1}) \in F[[y_1, \ldots, y_{\nu-1}]]$. E

$$g_{\underline{c}}(y_1, x) = [l_{\underline{c}}(y_1)]^2 \prod_{1 \leq i \leq n} (x - \alpha_i^{'}(y_1)), \tag{2.15}$$

com $\alpha_i^{'}(y_1) \in F[[y_1]]$.

Note que

$$\alpha_i^{'}(y_1) = \alpha_i(y_1, c_2y_1, \ldots, c_{\nu-1}y_1), \text{ para } 1 \leq i \leq n.$$

De (2.14) e (2.15) podemos concluir que cada fator $\widehat{h}(y_1, x) \in F[[y_1]][x]$ de $g_{\underline{c}}$ com $ldcf_x(\widehat{h}) = l_{\underline{c}}$ vem de um fator $h \in F[[y_1, \ldots, y_{\nu-1}]][x]$ de g com $ldcf_x(h) = l$, tal que

$$\widehat{h}(y_1, x) = h(y_1, c_2y_1, \ldots, c_{\nu-1}y_1, x) = h_{\underline{c}}(y_1, x).$$

Segunda demonstração.

Seja $\widehat{h}(y_1, x) \in F[[y_1]][x]$ fator de $g_{\underline{c}}$ com $ldcf_x(\widehat{h}) = l_{\underline{c}}$. Então

$$g_{\underline{c}}(y_1, x) = \widehat{h}(y_1, x) \cdot \widehat{\widehat{h}}(y_1, x) \tag{2.16}$$

com $0 < \text{grau}_x(\widehat{h}) < n$. Fazendo $y_1 = 0$ em (2.16), obtemos

$$g_{\underline{c}}(0, x) = \widehat{h}(0, x) \cdot \widehat{\widehat{h}}(0, x),$$

ou seja, uma fatoração não trivial de $g_{\underline{c}}(0, x)$. Usando o procedimento apresentado no teorema 2.3.1, podemos encontrar $h(y_1, \ldots, y_{\nu-1}, x)$ e $\overline{h}(y_1, \ldots, y_{\nu-1}, x) \in F[[y_1, \ldots, y_{\nu-1}]][x]$ tais que

$$g = h \cdot \overline{h}$$

e

$$h(0, \ldots, 0, x) = \widehat{h}(0, x).$$

E, pela unicidade do lema de Hensel temos

$$\widehat{h}(y_1, x) = h(y_1, c_2 y_1, \ldots, c_{\nu-1} y_1, x) = h_{\underline{c}}(y_1, x).$$

Ou seja, cada fator $\widehat{h}(y_1, x) \in F[[y_1]][x]$ de $g_{\underline{c}}$ com $ldcf_x(\widehat{h}) = l_{\underline{c}}$ vem de um fator $h \in F[[y_1, \ldots, y_{\nu-1}]][x]$ de g com $ldcf_x(h) = l$. □

A partir de agora, mostraremos que para inteiros $c_2, \ldots, c_{\nu-1}$ não anulando um certo polinômio $\pi(z_2, \ldots, z_{\nu-1}) \in F[z_2, \ldots, z_{\nu-1}]$ de grau máximo $4d(2^{n-1} - 1)$ os fatores de g não nos levarão a polinômios dividindo $g_{\underline{c}}$. E como cada fator de $g_{\underline{c}}$ vem de um fator de g de acordo com o lema 2.3.5, isso garantirá a irredutibilidade de $g_{\underline{c}}$.

Lema 2.3.6. *Existe um polinômio* $\pi(z_2, \ldots, z_{\nu-1}) \in F[z_2, \ldots, z_{\nu-1}]$ *tal que se as constantes* $c_2, \ldots, c_{\nu-1}$ *satisfazem*

$$\pi(c_2, \ldots, c_{\nu-1}) \neq 0,$$

então

$$g_{\underline{c}}(y_1, x) = g(y_1, c_2 y_1, \ldots, c_{\nu-1} y_1)$$

é irredutível.

Dem.: Seja

$$h(y_1, \ldots, y_{\nu-1}, x) = \sum_{j=0}^{i} \left(\sum_{\underline{k} \geq \underline{0}} b_{\underline{k}, j} \underline{y}^{\underline{k}} \right) x^j$$

um fator de $g(y_1,\ldots,y_{\nu-1},x)$ em $F[[y_1,\ldots,y_{\nu-1}]][x]$ com $0<i<n$ e $ldcf_x(h)=f$ e seja

$$\overline{h}(y_1,\ldots,y_{\nu-1},x)=\sum_{j=0}^{n-i}\left(\sum_{\underline{k}\geq\underline{0}}\overline{b}_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j$$

seu cofator, isto é

$$g=h\overline{h}.$$

Afirmamos que existe no mínimo um $b_{\underline{k},j}$ ou $\overline{b}_{\underline{k},j}$ com $2d<|\underline{k}|\leq 4d$ e $b_{\underline{k},j}\neq 0$ ou $\overline{b}_{\underline{k},j}\neq 0$. Para ver isto assuma o contrário. Então

$$g(y_1,\ldots,y_{\nu-1},x)=\left[\sum_{j=0}^{i}\left(\sum_{0\leq|\underline{k}|\leq 2d \text{ ou } |\underline{k}|>4d}b_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j\right]\cdot\left[\sum_{j=0}^{n-i}\left(\sum_{0\leq|\underline{k}|\leq 2d \text{ ou } |\underline{k}|>4d}\overline{b}_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j\right],$$

reescrevendo:

$$g(y_1,\ldots,y_{\nu-1},x)=\left[\underbrace{\sum_{j=0}^{i}\left(\sum_{0\leq|\underline{k}|\leq 2d}b_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j}_{h_1}+\underbrace{\sum_{j=0}^{i}\left(\sum_{|\underline{k}|>4d}b_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j}_{h_2}\right]\cdot$$

$$\left[\underbrace{\sum_{j=0}^{n-i}\left(\sum_{0\leq|\underline{k}|\leq 2d}\overline{b}_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j}_{\overline{h}_1}+\underbrace{\sum_{j=0}^{i}\left(\sum_{|\underline{k}|>4d}\overline{b}_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j}_{\overline{h}_2}\right].$$

Expandindo, temos:

$$g=\underbrace{h_1\cdot\overline{h}_1}_{\text{grau}_{y_1,\ldots,y_{\nu-1}}\leq 4d}+\underbrace{h_1\cdot\overline{h}_2}_{\text{grau}_{y_1,\ldots,y_{\nu-1}}>4d}+\underbrace{h_2\cdot\overline{h}_1}_{\text{grau}_{y_1,\ldots,y_{\nu-1}}>4d}+\underbrace{h_2\cdot\overline{h}_2}_{\text{grau}_{y_1,\ldots,y_{\nu-1}}>8d}.$$

E, como $\text{grau}_{y_1,\ldots,y_{\nu-1}}(g)\leq 2d$, temos

$$g=h_1\cdot\overline{h}_1=\left[\sum_{j=0}^{i}\left(\sum_{0\leq|\underline{k}|\leq 2d}b_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j\right]\cdot\left[\sum_{j=0}^{n-i}\left(\sum_{0\leq|\underline{k}|\leq 2d}\overline{b}_{\underline{k},j}\underline{y}^{\underline{k}}\right)x^j\right]\qquad(2.17)$$

e

$$h_1\cdot\overline{h}_2+h_2\cdot\overline{h}_1+h_2\cdot\overline{h}_2=0.$$

Mas a equação (2.17) contradiz a irredutibilidade de f. Então, sem perda de generalidade, podemos assumir a existência de um vetor $\underline{p}$ e um inteiro m tal que $b_{\underline{p},m} \neq 0$ e $2d < |\underline{p}| \leq 4d$ e $0 \leq m < i$. Assim,

$$h = \cdots + \left(\cdots + \sum_{|\underline{j}|=|\underline{p}|} b_{\underline{j},m} \underline{y}^{\underline{j}} + \cdots \right) x^m + \cdots .$$

Escrevendo

$$t_{\underline{p},m}(y_1, \ldots, y_{\nu-1}) = \sum_{|\underline{j}|=|\underline{p}|} b_{\underline{j},m} \underline{y}^{\underline{j}}$$

A **chave do teorema** da irredutibilidade de Hilbert efetivo é escolher $c_2, \ldots, c_{\nu-1}$ tal que

$$t_{\underline{p},m}(y_1, \ldots, y_{\nu-1})$$
$$\downarrow$$
$$t_{\underline{p},m}(y_1, c_2 y_1, \ldots, c_{\nu-1} y_1) \neq 0,$$

e assim garantindo que $h_{\underline{c}}(y_1, x)$ terá um coeficiente não nulo de ordem $|\underline{p}| > 2d$ em y_1, implicando que $h_{\underline{c}}(y_1, x)$ não dividirá $g_{\underline{c}}(y_1, x)$, pois $\text{grau}_{y_1}(g_{\underline{c}}) \leq 2d$.

Note que

$$t_{\underline{p},m}(y_1, \ldots, y_{\nu-1}) = \sum_{|\underline{j}|=|\underline{p}|} b_{\underline{j},m} \underline{y}^{\underline{j}} = \sum_{|\underline{j}|=|\underline{p}|} b_{\underline{j},m} y_1^{j_1} y_2^{j_2} \cdots y_{\nu-1}^{j_{\nu-1}} \text{ com } j_1 + \cdots + j_{\nu-1} = |\underline{p}|.$$

Fazendo a redução:

$$t_{\underline{p},m}(y_1, c_2 y_1, \ldots, c_{\nu-1} y_1) = \sum b_{|\underline{j}|,m} c_2^{j_2} \cdots c_{\nu-1}^{j_{\nu-1}} \cdot y_1^{\overbrace{j_1 + \cdots + j_{\nu-1}}^{=|\underline{p}|}}.$$

Assim $t_{\underline{p},m}(y_1, c_2 y_1, \ldots, c_{\nu-1} y_1) \neq 0$ se e somente se $\sum b_{|\underline{j}|,m} c_2^{j_2} \cdots c_{\nu-1}^{j_{\nu-1}} \neq 0$, ou seja,

$$t_{\underline{p},m}(1, c_2, \ldots, c_{\nu-1}) \neq 0.$$

Então o polinômio $\pi(z_2, \ldots, z_{\nu-1})$ pode ser escolhido como o produto dos $t_{\underline{p},m}(1, z_2, \ldots, z_{\nu-1}) \neq 0$ sobre todos os possíveis fatores h de g. Já que existem no máximo n fatores irredutíveis de g em $F[[y_1, \ldots, y_{\nu-1}]][x]$ e não precisamos considerar os candidatos complementares, então

$$\text{grau}(\pi) \leq 4d(2^{n-1} - 1).$$

Concluímos que cada conjunto $c_2, \ldots, c_{\nu-1}$ não anulando π mantém $g_{\underline{c}}$ irredutível. Pois garante algum monômio de $h_{\underline{c}}$ com grau em y_1 maior que $2d$.

Concluímos que se escolhermos $c_2, \ldots, c_{\nu-1}$ não anulando $\pi(c_2, \ldots, c_{\nu-1})$, garantimos que os fatores derivados de g não dividirão $g_{\underline{c}}$. E de acordo com o Lema 2.16 estes são todos os possíveis fatores de $g_{\underline{c}}$. Garantindo a irredutibilidade de $g_{\underline{c}}$. □

E assim, se todas as condições forem satisfeitas, podemos garantir que o polinômio

$$f(y_1 + w_1, c_2y_2 + w_2, \ldots, c_{\nu-1}y_1 + w_{\nu-1}, x)$$

é irredutível.

A partir de agora, mostraremos que isso acontece com alta probabilidade.

Começaremos tentando evitar um possível conteúdo em $F[x_1]$. Sejam $l_i(y_1, \ldots, y_{\nu-1})$ o coeficiente de x^i em $f(y_1 + w_1, \ldots, y_{\nu-1} + w_{\nu-1}, x)$, grau$(l_i) \leq d$. Note que $l_n = l$, ou seja, $l_n(0, \ldots, 0) \neq 0$. Já que f é irredutível então $mdc_{0\leq i\leq n}(l_i(y_1, \ldots, y_{\nu-1})) = 1$. Pelo Lema 2.3.3 existe um polinômio Δ com grau$(\Delta) \leq 2d^2$ tal que, se $\Delta(c_2, \ldots, c_{\nu-1}) \neq 0$ então $mdc_{0\leq i\leq n}(l_i(y_1, c_2y_1, \ldots, c_{\nu-1}y_1)) = 1$.

Podemos concluir que para a redução obter sucesso, ou seja, para que o polinômio multivariado permaneça irredutível quando reduzido para um polinômio bivariado devemos evitar os zeros do polinômio $\pi \cdot \Delta$. Lembremos também, que como mencionado no início da prova, queremos também que $f(w_1 \ldots, w_{\nu-1}, x)$ permaneça livre de quadrados (L.Q.) e que grau$(f(w_1, \ldots, w_{\nu-1}, x)) = \text{grau}_{x_\nu}(f(x_1, \ldots, x_\nu)) = n$.

$P(\pi\Delta \neq 0$ e $f(w_1 \ldots, w_{\nu-1}, x)$ permanecer L.Q. e grau$(f(w_1, \ldots, w_{\nu-1}, x)) = n) =$

$P(\pi\Delta \neq 0) \cdot P(f(w_1 \ldots, w_{\nu-1}, x)$ permanecer L.Q. e grau$(f(w_1, \ldots, w_{\nu-1}, x)) = n) \geq$

$$(1 - \frac{4d(2^{n-1} - 1) + 2d^2}{card(S)})(1 - \frac{(2n+1)d}{card(S)}) \geq 1 - \frac{4\delta 2^\delta - 3d}{card(S)} \geq 1 - \frac{4\delta 2^\delta}{card(S)}.$$

□

2.3.1 Teste de irredutibilidade probabilístico

Nesta subseção apresentaremos um algoritmo probabilístico baseado no Teorema 2.3.2 que funciona como um teste de irredutibilidade.

Lema 2.3.7. *Seja $f \in F[x_1, \ldots, x_\nu]$ primitivo em relação a x_ν. Se $\overline{f}(x_1, x_2)$ é irredutível e $grau_{x_2}(\overline{f}) = grau_{x_\nu}(f)$ então f é irredutível.*

Dem.: Suponha por contradição que f é redutível. Então $f = g \cdot h$ com $\text{grau}_{x_\nu}(g)$, $\text{grau}_{x_\nu}(h) \geq 1$, pois f é primitiva em relação a x_ν. Considere

$$f = \underbrace{(a_i x_\nu^i + \cdots)}_{g} \cdot \underbrace{(b_j x_\nu^j + \cdots)}_{h}$$

com a_k, $b_l \in F[x_1, \ldots, x_{\nu-1}]$ para todo $0 \leq k \leq i$ e $0 \leq l \leq j$. Fazendo a redução, temos

$$\overline{f}(x_1, x_2) = (\overline{a_i} x_2^i + \cdots) \cdot (\overline{b_j} x_2^j + \cdots)$$

e $\overline{a_i}$, $\overline{b_j} \neq 0$, pois $\text{grau}_{x_2}(\overline{f}) = \text{grau}_{x_\nu}(f)$. Então $\overline{f}$ é redutível. □

Algoritmo 2.3.1. *Dado um polinômio $f(x_1, \ldots, x_\nu) \in F[x_1, \ldots, x_\nu]$ primitivo em relação a x_ν, F corpo com $car(F) = 0$, este algoritmo testa a irredutibilidade de f com uma chance de erro pequeno de acordo com o teorema probabilístico 2.3.2.*

Elementos Aleatórios: *De um conjunto $S \subseteq F$ selecione elementos aleatórios $c_2, \ldots, c_{\nu-1}, w_1, \ldots, w_{\nu-1}$.*

Teste de Irredutibilidade: $\overline{f}(x_1, x_2) = f(x_1 + w_1, c_2 x_1 + w_2, \ldots, c_{\nu-1} x_1 + w_{\nu-1}, x_2)$.

1 *Se $grau_{x_2}(\overline{f}) < grau_{x_\nu}(f)$, então retorne falhou.*

2 *Se $grau_{x_2}(\overline{f}) = grau_{x_\nu}(f)$*

- **2.1** *Se $\overline{f}$ é irredutível, então retorne f é irredutível de acordo com o lema 2.3.7.*
- **2.2** *Se $\overline{f}$ é redutível, então retorne falhou.*

Dado um polinômio multivariado f. Primeiro aplicamos o teste de irredutibilidade 2.3.1, de acordo com o teorema 2.3.2 se f for irredutível temos uma boa probabilidade do algoritmo obter sucesso. Se o algoritmo falhar, há uma boa probabilidade do polinômio ser redutível, então aplicamos um método de fatoração efetivo.

2.4 Teorema da Irredutibilidade de Hilbert Efetivo: Shuhong Gao

Em [8] Shuhong Gao apresenta um novo método para fatoração de polinômios bivariados. A teoria deste algoritmo leva a um teorema de irredutibilidade de Hilbert efetivo.

Seja $\mathbb{F}$ um corpo e $\bar{\mathbb{F}}$ seu fecho algébrico. Dado um polinômio $f \in \mathbb{F}[x, y]$ e suponha que

$$f = f_1 \cdots f_r \tag{2.18}$$

onde $f_i \in \bar{\mathbb{F}}[x, y]$ são distintos e irredutíveis sobre $\bar{\mathbb{F}}$. Note que $f_x = \sum_{i=1}^{r} \frac{f}{f_i} \frac{\partial f_i}{\partial x}$ e defina

$$E_i = \frac{f}{f_i} \frac{\partial f_i}{\partial x} \in \bar{\mathbb{F}}[x, y],\ 1 \leq i \leq r. \tag{2.19}$$

Em seu novo método Gao considera a seguinte equação diferencial parcial

$$\frac{\partial}{\partial y}\left(\frac{g}{f}\right) = \frac{\partial}{\partial x}\left(\frac{h}{f}\right) \tag{2.20}$$

que pode ser reescrita como

$$f \cdot \left(\frac{\partial g}{\partial y} - \frac{\partial h}{\partial x}\right) + h \cdot \frac{\partial f}{\partial x} - g \cdot \frac{\partial f}{\partial y} = 0 \tag{2.21}$$

onde g, $h \in \bar{\mathbb{F}}[x, y]$. Em seu trabalho Gao restringe os graus de g e h

$$\text{grau duplo } g \leq (m-1, n),\ \text{grau duplo } h \leq (m, n-1), \tag{2.22}$$

onde grau duplo $g \leq (m-1, n)$ significa $\text{grau}_x(g) \leq m-1$ e $\text{grau}_y(g) \leq h$, e similarmente para h. E usando a equação (2.20) monta um sistema de equações

lineares nos coeficientes de g e h. A dimensão do espaço solução do sistema linear é igual ao número de fatores absolutamente irredutíveis do polinômio f e qualquer base para o espaço solução leva a uma fatoração completa através do cálculo de mdc's e fatoração de polinômios univariados.

Defina

$$\bar{G} = \{g \in \bar{\mathbb{F}}[x,y] : (2.20) \text{ e } (2.22) \text{ válidas para algum } h \in \bar{\mathbb{F}}[x,y]\}, \tag{2.23}$$

$$G = \{g \in \mathbb{F}[x,y] : (2.20) \text{ e } (2.22) \text{ válidas para algum } h \in \mathbb{F}[x,y]\}. \tag{2.24}$$

O próximo teorema determina as dimensões e estruturas de (2.23) e (2.24).

Teorema 2.4.1. *Seja $\mathbb{F}$ um corpo de característica p e $f \in \mathbb{F}[x,y]$ com $mdc(f, f_x) = 1$ e grau duplo (m,n), ou seja, $grau_x(f) = m$ e $grau_y(f) = n$. Suponha que f tem r fatores irredutíveis distintos em $\bar{\mathbb{F}}[x,y]$ como em (2.18) e seja G e $\bar{G}$ definidos como em (2.23) e (2.24). Se $p = 0$ ou $p > (2m-1)n$ então*

$$dim_{\mathbb{F}}(G) = dim_{\bar{\mathbb{F}}}(\bar{G}) = r \tag{2.25}$$

e cada $g \in \bar{G}$ é da forma

$$g = \sum_{i=1}^{r} \lambda_i E_i, \ \lambda_i \in \bar{\mathbb{F}}, \tag{2.26}$$

onde E_i está definido em (2.19).

A seguir mostraremos como reduzir o problema de fatorar polinômios com mais de duas varíaveis para o de fatorar polinômios bivariados. Para ser preciso, seja $f \in \mathbb{F}[x_1, \ldots, x_n]$ de grau total d. Consideremos a seguinte substituição

$$x_i = a_i x + b_i y + c_i \text{ para } 1 \leq i \leq n \tag{2.27}$$

onde a_i, b_i, $c_i \in \mathbb{F}$. Fazendo a substituição 2.27 em f obtemos o polinômio bivariado

$$f_0 = f(a_1 x + b_1 y + c_1, \ldots, a_n x + b_n y + c_n) \in \mathbb{F}[x,y]. \tag{2.28}$$

Suponha que tomemos valores aleatórios para a_1, b_1, c_1, ..., a_n, b_n, c_n de um subconjunto finito $S \subset \mathbb{F}$. Queremos saber qual a probabilidade de que todos os fatores

irredutíveis de f permaneçam irredutíveis após a substituição 2.27, isto é, f_0 e f terem a mesma fatoração padrão. O seguinte teorema fornece uma cota para tal probabilidade.

Teorema 2.4.2. *Seja $\mathbb{F}$ um corpo e S um subconjunto finito de $\mathbb{F}$. Seja $f \in \mathbb{F}[x_1, \ldots, x_n]$ de grau total d e f_0 definido a partir de f como em (2.28). Suponha que $\mathbb{F}$ tem característica zero ou característica maior que $2d^2$. Para escolhas aleatórias de a_i's, b_i's e c_i's em S, com probabilidade de no mínimo $1 - \frac{2d^3}{|S|}$ todos os fatores absolutamente irredutíveis de f permanecem fatores absolutamante irredutíveis de f_0 em $\mathbb{F}[x, y]$.*

Para provar o Teorema 2.4.2 precisamos de um resultado de Kaltofen [16]. Iremos imaginar a_1, b_1, c_1, ..., a_n, b_n, c_n como variáveis independentes sobre $\mathbb{F}$ e seja

$$L = \mathbb{F}(a_1, b_1, c_1, \ldots, a_n, b_n, c_n),$$

o corpo das funções racionais destas variáveis sobre $\mathbb{F}$. Então $f_0 \in L[x, y]$.

Lema 2.4.1 (Kaltofen [16])**.** *O polinômio bivariado f_0 em (2.28) é absolutamente irredutível sobre L se e somente se f é absolutamente irredutível sobre $\mathbb{F}$.*

Dem. do Teorema 2.4.2 : Podemos assumir que f é livre de quadrados, caso contrário poderíamos trabalhar com o produto de seus fatores irredutíveis distintos os quais teriam grau menor. Como comentado anteriormente, vamos enxergar a_1, b_1, c_1, ..., a_n, b_n, c_n como variáveis independentes sobre $\mathbb{F}$. Então, pelo lema 2.4.1 os fatores absolutamente irredutíveis de f_0 sobre L estão em correspondência 1-1 com aqueles de f. Em particular, já que f tem r fatores absolutamente irredutíveis sobre $\mathbb{F}$, f_0 também tem r fatores absolutamente irredutíveis sobre L. Considere o sistema linear (2.21) para f_0 sobre L. Seja M a matriz coeficiente do sistema. Usando o teorema 2.4.1 obtemos que $\text{posto}(M) + r = N$ onde r é a nulidade de M e N é o número de incógnitas do sistema. Note que f_0 tem grau total d, logo o polinômio g

em (2.21) tem grau total máximo $(d-1)$ de acordo com (2.26), e similarmente para h. Então g e h possuem no máximo $\frac{d(d+1)}{2}$ coeficientes, logo $N \leq d(d+1)$.

De acordo com a teoria de matrizes podemos dizer que M possui pelo menos uma submatriz M_1 de ordem $(N-r)\times(N-r)$ cujo determinante é diferente de zero e todas as outras submatrizes de ordem maior tem determinante nulo. Note que cada entrada de M é um polinômio nas variáveis independentes a_i's, b_i's e c_i's de grau máximo d. Logo $\det(M_1)$ é um polinômio nestas variáveis de grau no máximo

$$d\,(N-r) \leq dN \leq d^2\,(d+1) \leq 2d^3.$$

Para facilitar a notação, considere que f_0^*, M^* e M_1^* representam f_0, M e M_1, respectivamente, depois de substituirmos valores para os a_i's, b_i's e c_i's. Agora, suponhamos que os valores para os a_i's, b_i's e c_i's são tais que

$$\det\left(M_1^*\right) \neq 0. \tag{2.29}$$

Após a redução do polinômio multivariado f para o polinômio bivariado f_0^*, sem perda de generalidade, podemos lidar com dois casos: f_0^* terá $r+1$ fatores absolutamente irredutíveis sobre $\mathbb{F}$ ou f_0^* terá $r-1$ fatores absolutamente irredutíveis sobre $\mathbb{F}$.

Começamos supondo que f_0^* possui $r+1$ fatores absolutamente irredutiveis em $\mathbb{F}[x,y]$. Então pelo Teorema 2.4.1 temos que o posto da matriz M^* é $N-(r+1)$. Isto implica que M^* possui pelo menos uma submatriz de ordem $(N-(r+1))\times(N-(r+1))$ cujo determinante é não nulo e toda submatriz de ordem superior possui determinante nulo. Mas M_1^* é submatriz de ordem $(N-r)$ de M^*. Logo, $\det\left(M_1^*\right)=0$. Contradizendo (2.29). Então f_0^* possui no máximo r fatores absolutamente irredutíveis em $\mathbb{F}[x,y]$.

Agora suponha que f_0^* possui $r-1$ fatores absolutamente irredutiveis em $\mathbb{F}[x,y]$. Deste modo $\text{posto}(M^*) = N-r+1$ e isto implica que M^* possui pelo menos uma submatriz C de ordem $(N-r+1)\times(N-r+1)$ tal que $\det(C)\neq 0$. Mas

posto$(M) = N - r$, então toda submatriz de M de ordem $(N - r + 1) \times (N - r + 1)$ tem determinante nulo. Logo, toda submatriz de M^* de ordem $(N - r + 1) \times (N - r + 1)$ tem determinante nulo. Concluímos, assim, que f_0^* tem pelo menos r fatores absolutamente irredutíveis sobre $\mathbb{F}$.

Podemos concluir que

$$\text{prob}\,(f \text{ e } f_0^* \text{ terem a mesma fatoração padrão}) = \text{prob}\,(\det\,(M_1^*) \neq 0).$$

Usando um resultado de Schwartz [28] e Zippel [39], para valores aleatórios de a_i's, b_i's e c_i's de um conjunto S, a probabilidade de $\det\,(M_1^*) \neq 0$ é no mínimo $1 - \frac{2d^3}{|S|}$. □

2.5 Conclusão

Neste capítulo fizemos uma revisão bibliográfica sobre as versões do teorema da irredutibilidade de Hilbert. Acreditamos que isso consiste em uma contribuição didática, pois não existe um texto que detalhe as demonstrações apresentadas pelos autores.

3 FATORANDO POLINÔMIOS MULTIVARIADOS INTEIROS USANDO LOGARITMO DISCRETO

Neste capítulo apresentamos um novo algoritmo para fatoração polinomial multivariada. Desenvolvemos novos tipos de substituições, as quais reduzem o polinômio multivariado para polinômios bivariados, e mostramos que nosso método é realmente eficiente quando usado para fatorar polinômios multivariados que têm apenas fatores esparsos ou para extrairmos fatores esparsos de polinômios multivariados que possuem fatores densos e esparsos. A técnica desenvolvida neste capítulo também pode ser encontrada em [1].

3.1 Introdução

As primeiras implementações práticas de algoritmos para fatoração polinomial multivariada foram feitas por Paul Wang e Rothschild em [37, 38] nos anos 70. A efetividade destes algoritmos está fundamentada no teorema da irredutibilidade de Hilbert clássico, o qual mostra que, dado um polinômio multivariado irredutível $F(x_1, \ldots, x_n)$ sobre $\mathbb{Q}$, então, para infinitos números inteiros $a_1, \cdots, a_{n-1}$, o polinômio univariado $F(a_1, \ldots, a_{n-1}, x_n)$ permanece irredutível sobre $\mathbb{Q}$. Neste método, primeiro, o polinômio multivariado $F(x_1, \ldots, x_n)$ é reduzido para um polinômio univariado $F(a_1, \ldots, a_{n-1}, x_n)$ e então fatorado. Os fatores do polinômio univariado são usados para construir os fatores multivariados irredutíveis usando levantamento de Hensel. Gostaríamos de frisar que, embora esta substituição não possua argumentos probabilísticos garantindo que todos os fatores irredutíveis de $F(x_1, \ldots, x_n)$ permanecem irredutíveis após a redução, ou seja, $F(x_1, \ldots, x_n)$ e $F(a_1, \ldots, a_{n-1}, x_n)$ têm a mesma fatoração padrão, estas são as técnicas usadas atualmente para fatoração multivariada em programas de computador como Maple e Macsyma.

Quando queremos fatorar um polinômio multivariado $F(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ usando o método abordado no páragrafo anterior, podemos nos deparar com o problema conhecido como: **problema dos zeros ruins**. Isto acontece quando o polinômio univariado $F(x_1, 0, \ldots, 0)$ tem grau menor em x_1 comparado a $F(x_1, \ldots, x_n)$ ou, ao contrário de $F(x_1, \ldots, x_n)$, não é livre de quadrados. Quando isto ocorre, somos forçados a considerar, no lugar de $F(x_1, \ldots, x_n)$, o polinômio

$$F(x_1, x_2 + a_2, \ldots, x_n + a_n) \tag{3.1}$$

com $a_2, \ldots, a_n \in \mathbb{F}$. A desvantagem desta transformação é que, se $G(x_1, \ldots, x_n)$ for um fator esparso de $F(x_1, \ldots, x_n)$, então $G(x_1, x_2 + a_2, \ldots, x_n + a_n)$ certamente perderá sua esparsidade.

Durante os anos 80 novos tipos de reduções foram apresentados por Erich Kaltofen [14] e Joachin von zur Gathen [32], sendo que, desta vez o polinômio multivariado é reduzido para um polinômio bivariado. Kaltofen reduz o polinômio multivariado $F(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ para o polinômio bivariado

$$F(x_1 + c_1, a_2x_1 + c_2, \ldots, a_{n-1}x_1 + c_{n-1}, x_n) \tag{3.2}$$

e von zur Gathen reduz para o polinômio bivariado

$$F(x_1, x_2, a_3x_1 + b_3x_2 + c_3, \ldots, a_nx_1 + b_nx_2 + c_n) \tag{3.3}$$

com a_i's, b_i's e c_i's $\in \mathbb{F}$. O diferencial destas novas substituições está no fato de que possuem argumentos probabilísticos garantindo que a fatoração padrão é mantida com boa probabilidade. Mas, como na redução 3.1, se $G(x_1, \ldots, x_n)$ for um fator esparso de $F(x_1, \ldots, x_n)$ então as reduções 3.2 e 3.3 provavelmente farão G deixar de ser esparso.

Neste capítulo apresentaremos duas novas reduções, as quais têm como principal característica manter a esparsidade do polinômio. Dado um polinômio multivariado $F(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$, uma de nossas reduções substitui cada variável x_i por $s^{d^{i-1}}X^{a_i}Y^{b_i} \bmod p$, e a outra substitui cada x_i por $s^{2d^{i-1}}X^{a_i}Y^{b_i} \bmod p$, para certas constantes s, d, p e números inteiros aleatórios não-negativos a_i's e b_i's.

Mostraremos que estas reduções mantêm o número de termos com boa probabilidade se o polinômio for esparso. Isto é, se $F(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$ for esparso, então

$$F(sX^{a_1}Y^{b_1}, s^d X^{a_2}Y^{b_2}, \ldots, s^{d^{n-1}} X^{a_1}Y^{b_1}) \in \mathbb{F}_p[X, Y] \tag{3.4}$$

e

$$F(s^2X^{a_1}Y^{b_1}, s^{2d} X^{a_2}Y^{b_2}, \ldots, s^{2d^{n-1}} X^{a_1}Y^{b_1}) \in \mathbb{F}_p[X, Y] \tag{3.5}$$

terão o mesmo número de termos que $F(x_1, \ldots, x_n)$ com boa probabilidade.

Nas figuras abaixo comparamos as reduções e como elas transformam um termo $F_e x_1^{e_1} \cdots x_n^{e_n}$ de $F(x_1, \ldots, x_n)$.

Wang	Kaltofen
$F_e x_1^{e_1} \cdots x_n^{e_n}$	$F_e x_1^{e_1} \cdots x_n^{e_n}$
$\downarrow$	$\downarrow$
$F_e x_1^{e_1} (x_2 - a_2)^{e_2} \cdots (x_n - a_n)^{e_n}$	$F_e (x_1 + c_1)^{e_1} (a_2 x_1 + c_2)^{e_2} \cdots (a_{n-1} x_1 + c_{n-1})^{e_{n-1}} x_n^{e_n}$

e

von zur Gathen	Nossa redução
$F_e x_1^{e_1} \cdots x_n^{e_n}$	$F_e x_1^{e_1} \cdots x_n^{e_n}$
$\downarrow$	$\downarrow$
$F_e x_1^{e_1} x_2^{e_2} (a_3 x_1 + b_3 x_2 + c_3)^{e_3} \cdots (a_n x_1 + b_n x_2 + c_n)^{e_n}$	$F_e s^{\ell(e)} X^{a \cdot e} Y^{b \cdot e}$

Nas figuras acima, consideramos $\ell(e) = e_1 + e_2 d + \cdots + e_n d^{n-1}$, $a \cdot e = a_1 e_1 + a_2 e_e + \cdots + a_n e_n$ e $b \cdot e = b_1 e_1 + b_2 e_e + \cdots + b_n e_n$.

Acabamos de observar que, se um polinômio multivariado for esparso, então as Reduções 3.1, 3.2 e 3.3 farão com que este polinômio perca esta propriedade. Então nosso método pode ser usado para fatorar completamente um polinômio multivariado se este possuir apenas fatores esparsos. E se um polinômio multivariado tiver fatores esparsos e densos, um método híbrido pode ser aplicado. Primeiro, ex-

traímos os fatores esparsos usando nosso algoritmo e depois encontramos os fatores densos usando um algoritmo baseado no levantamento de Hensel.

Este capítulo está organizado da seguinte maneira: na Seção 3.2 apresentamos algumas notações e definições básicas. Na Seção 3.3 introduzimos as reduções e a ordem monomial que são usadas em nossos algoritmos. Nas Seções 3.4 e 3.5 desenvolvemos os algoritmos. Por completude, na seção 3.6, estudamos o algoritmo de Garner generalizado. Na Seção 3.7 calculamos as cotas teóricas necessárias em cada algoritmo. Na Seção 3.8 apresentamos a análise de operações de nossos algoritmos. Terminamos, na Seção 3.9, apresentando os dados comparativos entre nosso algoritmo de fatoração multivariada e o programa de computador Maple.

3.2 Preliminares matemáticas

Gostaríamos de observar que, durante este capítulo, sempre estaremos trabalhando com um polinômio multivariado $F \in \mathbb{Z}[x_1, \ldots, x_n]$ livre de quadrados. Ou seja, podemos supor

$$F = G_1 \cdots G_r \tag{3.6}$$

onde G_i's $\in \mathbb{Z}[x_1, \ldots, x_n]$ têm grau positivo, são distintos e irredutíveis sobre $\mathbb{Z}$. Se um polinômio multivariado $F \in \mathbb{Z}[x_1, \ldots, x_n]$ não é livre de quadrados, ou seja, possui fatores repetidos, sempre podemos reduzir o problema de fatorar F para o caso em que F é livre de quadrados (ver [8, 38]).

Ordem monomial

Seja $\mathbb{F}$ um corpo e $\mathbb{F}[x_1, \ldots, x_n]$ o anel polinomial em n variáveis sobre $\mathbb{F}$. No caso univariado ordenar monômios é simples: a única ordem monomial é a ordem do grau dado por

$$\cdots > x^{n+1} > x^n > \cdots > x^2 > x > 1.$$

No caso multivariado ordenar monômios também é necessário, em particular, para divisão polinomial com resto, e existem infinitos modos de ordenar monômios.

Escrevemos um monômio nas variáveis $x_1, \ldots, x_n$ em $\mathbb{F}[x_1, \ldots, x_n]$ como

$$x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n},$$

onde $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n$ é o expoente vetorial ($\mathbb{N}$ denota o conjunto $\{0, 1, 2, \ldots\}$ de inteiros não negativos).

Definição 3.2.1. *Uma ordem monomial em $\mathbb{F}[x_1, \ldots, x_n]$ é uma relação de ordem parcial $>$ sobre o conjunto dos monômios $x^\alpha \in \mathbb{F}[x_1, \ldots, x_n]$ ou, equivalentemente, sobre os expoentes vetoriais $\alpha \in \mathbb{N}^n$, tal que*

1. *$>$ é uma ordem total: $\forall \alpha, \beta \in \mathbb{N}^n$, $\alpha = \beta$ ou $\alpha > \beta$ ou $\beta > \alpha$,*
2. *se $\alpha > \beta$ então $\alpha + \gamma > \beta + \gamma$, $\forall\ \alpha, \beta, \gamma \in \mathbb{N}^n$, e*
3. *$>$ é uma boa-ordem: todo subconjunto não vazio de $\mathbb{N}^n$ tem um menor elemento sob $>$.*

Deste modo, para quaisquer dois monômios x^α e x^β em $\mathbb{F}[x_1, \ldots, x_n]$, escrevemos $x^\alpha > x^\beta$, $x^\alpha = x^\beta$ ou $x^\beta > x^\alpha$, se $\alpha > \beta$, $\alpha = \beta$ ou $\beta > \alpha$, respectivamente.

Seja $\alpha = (\alpha_1, \ldots, \alpha_n)$ e $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n$.

Definição 3.2.2. *Seja $\omega = (\omega_1, \ldots, \omega_n) \in \mathbb{R}^n$ e fixe uma ordem monomial $>$ em $\mathbb{F}[x_1, \ldots, x_n]$. A ordem dos graus com peso (ou ordem peso) determinada por ω e $>$ é definida por*

$$x^\alpha >_\omega x^\beta \iff \omega \cdot \alpha > \omega \cdot \beta, \text{ ou } \omega \cdot \alpha = \omega \cdot \beta \text{ e } \alpha > \beta,$$

onde "$\cdot$" denota o produto interno usual em $\mathbb{R}^n$: $\omega \cdot \alpha = \omega_1\alpha_1 + \cdots + \omega_n\alpha_n \in \mathbb{R}$, dito o peso do grau de x^α.

Notação

Sejam $>$ uma ordem monomial em $\mathbb{F}[x_1, \ldots, x_n]$ e $F = \sum_{\alpha \in \mathbb{N}^n} F_\alpha x^\alpha \in \mathbb{F}[x_1, \ldots, x_n]$, onde os coeficientes $F_\alpha \in \mathbb{F}$ são não-nulos. Cada $F_\alpha x^\alpha$ é um *termo* de F. O *termo líder* de F é $\mathrm{lt}(F) = \max_>\{F_\alpha x^\alpha\}$. Se $\mathrm{lt}(F) = F_\omega x^\omega$ dizemos que ω é o *multigrau* de F, denotado por $\mathrm{mgrau}(F)$. Também, F_ω é dito o *coeficiente líder* de F, denotado por $\mathrm{lc}(F)$, e x^ω é dito o *monômio líder* de F, denotado por $\mathrm{lm}(f)$.

Dado um polinômio multivariado $F \in \mathbb{F}[x_1, \ldots, x_n]$, denotaremos seu número de termos por $\mathrm{nops}(F)$.

Um polinômio de grau limitado por M em cada variável em $\mathbb{F}[x_1, \ldots, x_n]$ terá no máximo $(M+1)^n$ termos, e um polinômio com $\Omega((M+1)^n)$ termos não nulos será dito denso. Polinômios esparsos são aqueles que possuem poucos termos não nulos quando comparados com este máximo. Vamos definir uma noção mais precisa de esparsidade motivada em parte pela região de aplicabilidade dos algoritmos desenvolvidos neste capítulo. Diremos que um polinômio com grau limitado por M em cada variável é esparso se o seu número de termos não nulos for menor que $(n \cdot M)$.

Dado $F = \sum F_\alpha x_1^{\alpha_1} \cdots x_n^{\alpha_n} \in \mathbb{F}[x_1, \ldots, x_n]$ definimos o conjunto de suporte de F como

$$\mathrm{Supp}(F) := \{(\alpha_1, \ldots, \alpha_n) \in \mathbb{N}^n : F_\alpha \neq 0\}.$$

Definimos também a seguinte operação: dado um polinômio $F \in \mathbb{F}[x_1, \ldots, x_n]$. Então dividimos F por todo x_i possível e chamamos o polinômio resultante de $\mathrm{clean}(F)$.

3.3 Reduções

As reduções que apresentaremos nesta seção transformam um polinômio multivariado $F(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$, com $n \geq 3$, em um polinômio bivariado.

Uma importante característica de nosso método é que nossas reduções nunca aumentam o número de termos.

Considere $F = \sum_{i=0}^{t} F_i x^{\alpha_i} \in \mathbb{Z}[x_1, \ldots, x_n]$ e p um número primo. Seja $S = \{0, 1, 2, \ldots, c\}$ um subconjunto finito em $\mathbb{N}$, s um gerador para o grupo multiplicativo $\mathbb{F}_p^*$ e $d = 2(\max_{1 \leq i \leq n}(\text{grau}_{x_i}(F))) + 1$. Então as reduções são definidas da seguinte maneira: escolha a_i e $b_i \in S$ aleatoriamente para $1 \leq i \leq n$ e defina

$$R_1^{a,b,p} : x_i \rightarrow s^{d^{i-1} \bmod (p-1)} \bmod(p) X^{a_i} Y^{b_i} \text{ para } 1 \leq i \leq n. \tag{3.7}$$

$$R_2^{a,b,p} : x_i \rightarrow s^{2d^{i-1} \bmod (p-1)} \bmod(p) X^{a_i} Y^{b_i} \text{ para } 1 \leq i \leq n. \tag{3.8}$$

Aplicando a Redução 3.7 em um termo de F obtemos:

$$R_1^{a,b,p}(F_i x^{\alpha_i}) = R_1^{a,b,p}(F_i x_1^{\alpha_{i,1}} \cdots x_n^{\alpha_{i,n}}) = F_i (s X^{a_1} Y^{b_1})^{\alpha_{i,1}} \cdots (s^{d^{n-1}} X^{a_n} Y^{b_n})^{\alpha_{i,n}} =$$

$$F_i s^{\alpha_{i,1} + \cdots + \alpha_{i,n} d^{n-1}} X^{a_1 \alpha_{i,1} + \cdots + a_n \alpha_{i,n}} Y^{b_1 \alpha_{i,1} + \cdots + b_n \alpha_{i,n}} = F_i s^{\ell(\alpha_i)} X^{a \cdot \alpha_i} Y^{b \cdot \alpha_i}$$

com $\ell(\alpha_i) = \alpha_{i,1} + \alpha_{i,2} d + \cdots + \alpha_{i,n} d^{n-1}$, $a \cdot \alpha_i = a_1 \alpha_{i,1} + \cdots + a_n \alpha_{i,n}$ e $b \cdot \alpha_i = b_1 \alpha_{i,1} + \cdots + b_n \alpha_{i,n}$. Do mesmo modo, quando aplicamos a Redução 3.8 no mesmo termo obtemos:

$$R_2^{a,b,p}(F_i x^{\alpha_i}) = F_i s^{2\ell(\alpha_i)} X^{a \cdot \alpha_i} Y^{b \cdot \alpha_i}.$$

Agora, vamos citar os principais passos em nossos algoritmos. Dado um polinômio multivariado $F(x_1, \ldots, x_n) \in \mathbb{Z}[x_1, \ldots, x_n]$, aplicamos as reduções 3.7 e 3.8 em F. Então fatoramos os polinômios bivariados $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ em $\mathbb{F}_p[X, Y]$ e usando seus fatores construímos os fatores multivariados de F. Como veremos na seção 3.4, se G é um fator irredutível de F, então podemos recuperar G a partir dos fatores de $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ se o número de termos de G permanece o mesmo após as reduções, ou seja, se a seguinte condição for satisfeita

G e $R_1^{a,b,p}(G)$ têm o mesmo número de termos.

Note que, se a condição acima é satisfeita, então $R_1^{a,b,p}(G)$ e $R_2^{a,b,p}(G)$ terão os mesmos monômios. A propriedade seguinte mostra que G e $R_1^{a,b,p}(G)$ terão o mesmo número de termos com boa probabilidade quando G for um polinômio esparso.

Propriedade 3.3.1. *Seja $S = \{0, 1, 2, \ldots, c\}$ um subconjunto finito em $\mathbb{Z}$. Considere o polinômio multivariado $G = \sum_{i=0}^{m} G_i x^{\beta_i} \in \mathbb{Z}[x_1, \ldots, x_n]$ e um número primo p tal que $p \nmid G_i$ para $1 \leq i \leq m$. Então, para escolhas aleatórias de a_i's e b_i's de S temos que*

$$P(nops(G) = nops(R_1^{a,b,p}(G))) > 1 - \frac{m(m+1)}{2|S|^2}. \tag{3.9}$$

Dem.: Note que

$$P(\text{nops}(G) \neq \text{nops}(R_1^{a,b,p}(G)))$$

$$= P(X^{a\cdot\beta_i} Y^{b\cdot\beta_i} = X^{a\cdot\beta_j} Y^{b\cdot\beta_j} \text{ para certos } i, j \in \{0, \ldots, m\})$$

$$= P(a \cdot \beta_i = a \cdot \beta_j \text{ e } b \cdot \beta_i = b \cdot \beta_j) = P(b \cdot \beta_i = b \cdot \beta_j | a \cdot \beta_i = a \cdot \beta_j) P(a \cdot \beta_i = a \cdot \beta_j).$$

Usando o lema probabilístico de separação, veja [8], encontramos que

$$P(a \cdot \beta_i = a \cdot \beta_j) < \frac{(m+1)m}{2|S|}$$

e

$$P(b \cdot \beta_i = b \cdot \beta_j | a \cdot \beta_i = a \cdot \beta_j) < \frac{1}{|S|}.$$

Então

$$P(\text{nops}(G) \neq \text{nops}(R_1^{a,b,p}(G))) < \left(\frac{(m+1)m}{2|S|}\right)\left(\frac{1}{|S|}\right) = \frac{(m+1)m}{2|S|^2}.$$

□

Note que, de acordo com a desigualdade (3.9), quando G for um polinômio multivariado esparso, então a probabilidade de G e $R_1^{a,b,p}(G)$ terem o mesmo número de termos é alta.

Ordem Monomial Utilizada

Agora, iremos definir as ordens monomiais que serão usadas ao longo deste capítulo. Considere $a = (a_1, \ldots, a_n)$ e $b = (b_1, \ldots, b_n) \in \mathbb{N}^n$. Definiremos as ordens monomiais sobre $\mathbb{Z}[x_1, \ldots, x_n]$ e $\mathbb{F}_p[X, Y]$ de tal maneira que, dado $G \in \mathbb{Z}[x_1, \ldots, x_n]$, então

$$\mathrm{lt}(R_1^{a,b,p}(G)) = R_1^{a,b,p}(\mathrm{lt}(G)) \text{ e } \mathrm{lt}(R_2^{a,b,p}(G)) = R_2^{a,b,p}(\mathrm{lt}(G)),$$

isto é, os termos líderes de $R_1^{a,b,p}(G)$ e $R_2^{a,b,p}(G)$ venham do termo líder de G, se G e $R_1^{a,b,p}(G)$ tiverem o mesmo nops.

Sejam $\alpha = (\alpha_1, \ldots, \alpha_n)$ e $\beta = (\beta_1, \ldots, \beta_n) \in \mathbb{N}^n$. Então vamos definir a seguinte ordem monomial sobre $\mathbb{Z}[x_1, \ldots, x_n]$:

1. $x^\beta > x^\alpha$ se $a \cdot \beta > a \cdot \alpha$.
2. se $a \cdot \beta = a \cdot \alpha$ então $x^\beta > x^\alpha$ se $b \cdot \beta > b \cdot \alpha$.
3. se $a \cdot \beta = a \cdot \alpha$ e $b \cdot \beta = b \cdot \alpha$ então $x^\beta > x^\alpha \Leftrightarrow x^\beta >_{lex} x^\alpha$ onde as variáveis estão ordenadas com $x_1 > x_2 > \cdots > x_n$.

De agora até o final deste capítulo, sempre que estivermos trabalhando em $\mathbb{Z}[x_1, \ldots, x_n]$ iremos usar a ordem monomial que acabamos de definir. E quando trabalharmos em $\mathbb{F}_p[X, Y]$ iremos usar a ordem lexicográfica com $X > Y$. Usando estas ordens monomiais, automaticamente provamos o lema a seguir.

Lema 3.3.1. *Dados $a = (a_1, \ldots, a_n)$ e $b = (b_1, \ldots, b_n) \in \mathbb{N}^n$. Seja $G = \sum_{i=0}^{m} G_i x^{\beta_i} \in \mathbb{Z}[x_1, \ldots, x_n]$ com $lt(G) = G_m x^{\beta_m}$. Se*

$$nops(G) = nops(R_1^{a,b,p}(G)). \tag{3.10}$$

então

$$lt(R_1^{a,b,p}(G)) = R_1^{a,b,p}(lt(G)) \text{ e } lt(R_2^{a,b,p}(G)) = R_2^{a,b,p}(lt(G)).$$

3.4 Algoritmo Básico

No intuito de fatorar um polinômio multivariado $F \in \mathbb{Z}[x_1, \ldots, x_n]$, começamos escolhendo inteiros não negativos $a_1, \ldots, a_n$ e $b_1, \ldots, b_n$, e um número primo p (veja Seção 3.7). Então, aplicamos as reduções 3.7 e 3.8 em F. Usando apenas os fatores esparsos de $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ que possuam o mesmo conjunto de suporte, construímos os fatores multivariados de F usando logaritmos discretos.

Suponha que $[g_1, \ldots, g_r]$ sejam os fatores esparsos mônicos irredutíveis de clean$(R_1^{a,b,p}(F))$ sobre $\mathbb{F}_p$. Agora, se Supp(g_i) = Supp(g_j) para algum $i \neq j$ então retire g_i e g_j do conjunto e coloque $(g_i \cdot g_j)$ mod p. Após esta operação, nomeie o conjunto de Esparsos(clean$(R_1^{a,b,p}(F))$). Suponha que tenhamos calculado Esparsos(clean$(R_1^{a,b,p}(F))$) e Esparsos(clean$(R_2^{a,b,p}(F))$), então defina o seguinte conjunto

$$\text{Padrão}(p) := [(g_1, g_2) : g_1 \in \text{Esparsos}(\text{clean}(R_1^{a,b,p}(F))) \text{ e}$$
$$g_2 \in \text{Esparsos}(\text{clean}(R_2^{a,b,p}(F))) \text{ com Supp}(g_1) = \text{Supp}(g_2)].$$

Algoritmo 3.4.1. (*Fatoração em* $\mathbb{Z}[x_1, \ldots, x_n]$)
Entrada: Um polinômio multivariado $F \in \mathbb{Z}[x_1, \ldots, x_n]$.
Saída: Os fatores esparsos de F *ou Falhou.*

passo 1: {*Cota*} *Escolha um número primo p satisfazendo as condições da Seção 3.7 e calcule um gerador s de* $\mathbb{F}_p^*$.

passo 2: {*Reduções*} *Escolha* a_i *e* b_i *aleatoriamente de um conjunto* $\in S = \{0, 1, 2, \cdots, c\}$ *dos* $\mathbb{N}$ *para* $1 \leq i \leq n$. *E então calcule* $R_1^{a,b,p}(F)$ *e* $R_2^{a,b,p}(F)$.

passo 3: {*Fatoração bivariada*} *Fatore* $R_1^{a,b,p}(F)$ *e* $R_2^{a,b,p}(F)$ *em* $\mathbb{F}_p[X,Y]$. *Calcule* $Esparsos(clean(R_1^{a,b,p}(F)))$ *e* $Esparsos(clean(R_2^{a,b,p}(F)))$.
Se $(Esparsos(clean(R_1^{a,b,p}(F))) = [\,])$ *ou* $(Esparsos(clean(R_2^{a,b,p}(F))) = [\,])$ ***então retorne***(*F provavelmente não tem fatores esparsos*) *e pare o algoritmo.*

passo 4: ***Se*** $(Padrão(p) = [\,])$ ***então*** *vá para o passo 2*
senão chame$(Algoritmo\ 3.4.2(F, Padrão(p)))$.

Gostaríamos de observar que nosso método terá um problema combinatorial se F possuir dois fatores ou mais que tenham o mesmo conjunto de suporte. Devido a isso calculamos, anteriormente, os conjuntos Esparsos de tal modo que possuam apenas elementos com conjunto de suporte distinto.

No próximo algoritmo contruímos os fatores multivariados de F usando os fatores bivariados de $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ que tenham o mesmo conjunto de suporte. Usaremos a notação $\text{coeff}(G, X^{e_x}Y^{e_y})$ para denotar o coeficiente de $X^{e_x}Y^{e_y}$ em $G \in \mathbb{F}_p[X,Y]$ e a operação de colocar um elemento a em um conjunto A será denotada por $[\text{op}(A), a]$, e, dado $F = \sum_{i=1}^{t} F_0 x^{\alpha_i} \in \mathbb{Z}[x_1, \ldots, x_n]$, definimos $\text{iconteúdo}(F) = \text{mdc}(F_0, \ldots, F_t) \in \mathbb{Z}$ como sendo o conteúdo inteiro de F.

Algoritmo 3.4.2. (*Construção de polinômios multivariados.*)
Entrada: $(F, Padrão(p))$.
Saída: Os fatores esparsos de F *ou Falhou.*

passo 0: Inicialize $FS := [\,]$.

passo 1: {*Mesmo suporte*} ***Para*** *cada par* $(g_1, g_2) \in Padrão(p)$ ***faça***

$$\tilde{g_1} := lc(R_1^{a,b,p}(lt(F)))g_1 \in \mathbb{F}_p[X,Y]$$

$$\tilde{g_2} := lc(R_2^{a,b,p}(lt(F)))g_2 \in \mathbb{F}_p[X,Y]$$

e defina $Supp$=$Supp(g_1)$.

passo 1.1: {*Logaritmo discreto*} ***Para*** *cada* $e = (e_x, e_y) \in Supp$ ***faça***
calcule ℓ_e *mod* $(p-1)$ *usando logaritmo discreto na equação*

$$(coeff(\tilde{g}_1, X^{e_x}Y^{e_y}))^{-1} \cdot coeff(\tilde{g}_2, X^{e_x}Y^{e_y}) = s^{\ell_e}$$

e converta ℓ_e *em base d:*

$$\ell_e = e_{i,1} + e_{i,2}d + \cdots + e_{i,n}d^{n-1}.$$

passo 1.2: Defina

$$\tilde{G} := \textstyle\sum_{e \in Supp} coeff(\tilde{g}_1, X^{e_x}Y^{e_y}) s^{-\ell_e} x_1^{e_{i,1}} x_2^{e_{i,2}} \cdots x_n^{e_{i,n}} \in \mathbb{F}_p[x_1, \ldots, x_n]$$

passo 1.3: Converta $\tilde{G}$ *para* $\mathbb{Z}[x_1, \ldots, x_n]$ *e defina*

$$G = \frac{clean(\tilde{G})}{iconteúdo(\tilde{G})} \in \mathbb{Z}[x_1, \ldots, x_n] \ .$$

passo 1.4: ***Se*** $(G|F)$ ***então*** $FS := [op(FS), G]$.

passo 2: ***Se*** $(FS = [\ ])$ ***então*** *retorne*(*Falhou*) ***senão*** *retorne*(FS).

No próximo teorema mostraremos que, se G for um fator esparso de F em $\mathbb{Z}[x_1, \ldots, x_n]$, então podemos calcular G a partir dos fatores bivariados de $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ em $\mathbb{F}_p[X, Y]$ usando logaritmo discreto sempre que G e $R_1^{a,b,p}(G)$ tiverem o mesmo número de termos. Gostaríamos de observar que consideramos clean$(R_1^{a,b,p}(G))$, clean$(R_2^{a,b,p}(G))$ irredutíveis em $\mathbb{F}_p[X, Y]$ nas hipóteses do teorema a seguir para evitarmos argumentos combinatórios na demonstração.

Teorema 3.4.1. *Seja* $F = \sum_{i=0}^{t} F_i x^{\alpha_i} \in \mathbb{Z}[x_1, \ldots, x_n]$ *com* $d = 2(max_{1 \le i \le n}(grau_{x_i}(F)) + 1$ *e sejam* $a_1, \ldots, a_n$ *e* $b_1, \ldots, b_n$ *inteiros não-negativos. Suponha que* $G = \sum_{i=0}^{m} G_i x^{\beta_i}$ *é um fator irredutível de* F *em* $\mathbb{Z}[x_1, \ldots, x_n]$ *e* $H = \sum_{i=0}^{L} H_i x^{\gamma_i}$ *seu cofator e tenhamos um número primo* $p > max_{1 \le i \le m}\{|H_L \dot{G}_i|, \ell(\beta_i) + \ell(\gamma_L)\}$ *tal que* $nops(G) = nops(R_1^{a,b,p}(G))$ *e* $clean(R_1^{a,b,p}(G))$, $clean(R_2^{a,b,p}(G))$ *são irredutíveis em* $\mathbb{F}_p[X, Y]$, *então podemos recuperar* G *a partir dos fatores de* $R_1^{a,b,p}(F)$ *e* $R_2^{a,b,p}(F)$ *em* $\mathbb{F}_p[X, Y]$ *usando logaritmos discretos.*

Dem.: Vamos considerar $G = \sum_{i=0}^{m} G_i x^{\beta_i}$ um fator irredutível de $F = \sum_{i=0}^{t} F_i x^{\alpha_i}$ e $H = \sum_{i=0}^{L} H_i x^{\gamma_i}$ seu cofator em $\mathbb{Z}[x_1, \ldots, x_n]$. De acordo com a nossa notação temos

$$(\sum_{i=0}^{t} F_i x^{\alpha_i}) = (\sum_{i=0}^{m} G_i x^{\beta_i}) \cdot (\sum_{i=0}^{L} H_i x^{\gamma_i}) \tag{3.11}$$

com

$$\mathrm{lt}(F) = \mathrm{lt}(G) \cdot \mathrm{lt}(H) \text{ , ou seja, } F_t x^{\alpha_t} = (G_m x^{\beta_m}) \cdot (H_L x^{\gamma_L}).$$

Aplicando a redução 3.7 em (3.11) obtemos

$$R_1^{a,b,p}(F) = R_1^{a,b,p}(G) \cdot R_1^{a,b,p}(H) \in \mathbb{F}_p[X, Y]. \tag{3.12}$$

E fatorando $R_1^{a,b,p}(F)$ em $\mathbb{F}_p[X, Y]$ temos que

$$R_1^{a,b,p}(F) = A_1 X^{u_x} Y^{u_y} g_1 \cdots g_r \in \mathbb{F}_p[X, Y] \tag{3.13}$$

onde $A_1 \in \mathbb{F}_p$, $u_x, u_y \geq 0$ e $g_1, \ldots, g_r$ são os fatores mônicos irredutíveis de $\mathrm{clean}(R_1^{a,b,p}(F))$ em $\mathbb{F}_p[X, Y]$. Usando as equações (3.12) e (3.13) observamos que

$$\mathrm{clean}(R_1^{a,b,p}(G)) \cdot \mathrm{clean}(R_1^{a,b,p}(H)) = A_1 g_1 \cdots g_r \in \mathbb{F}_p[X, Y].$$

Então

$$\mathrm{clean}(R_1^{a,b,p}(G)) = G_m s^{\ell(\beta_m)} \cdot \bar{g_1}$$

onde $\bar{g_1} = g_i$ para algum $i \in \{1, \ldots, r\}$. Realizando operações similares usando a redução 3.8 obtemos

$$\mathrm{clean}(R_2^{a,b,p}(G)) = G_m s^{2\ell(\beta_m)} \cdot \bar{g_2}$$

onde $\bar{g_2}$ é um fator irredutível mônico de $\mathrm{clean}(R_2^{a,b,p}(G))$ em $\mathbb{F}_p[X, Y]$. Observe que quando a redução 3.7 é aplicada sobre G temos

$$R_1^{a,b,p}(G) = \sum_{i=1}^{m} G_i s^{\ell(\beta_i)} X^{a \cdot \beta_i} Y^{b \cdot \beta_i}$$

com $\mathrm{lt}(R_1^{a,b,p}(G)) = R_1^{a,b,p}(\mathrm{lt}(G)) = G_m s^{\ell(\beta_m)} X^{a\cdot\beta_m} Y^{b\cdot\beta_m}$ e

$$R_2^{a,b,p}(G) = \sum_{i=1}^{m} G_i s^{2\ell(\beta_i)} X^{a\cdot\beta_i} Y^{b\cdot\beta_i}$$

com $\mathrm{lt}(R_2^{a,b,p}(G)) = R_2^{a,b,p}(\mathrm{lt}(G)) = G_m s^{2\ell(\beta_m)} X^{a\cdot\beta_m} Y^{b\cdot\beta_m}$, pelo Lema 3.3.1. Note que $\mathrm{clean}(R_1^{a,b,p}(G))$, $\mathrm{clean}(R_2^{a,b,p}(G))$, $\bar{g}_1$, $\bar{g}_2$ têm o mesmo conjunto de suporte. Se considerarmos que $\bar{g}_1 = \sum_{i=1}^{m} C_{1,i} X^{u_i} Y^{v_i}$ com $C_{1,m} = 1$ e $\bar{g}_2 = \sum_{i=1}^{m} C_{2,i} X^{u_i} Y^{v_i}$ com $C_{2,m} = 1$, então temos as seguintes igualdades

$$\bar{g}_1 = G_m^{-1} s^{-\ell(\beta_m)} \mathrm{clean}(R_1^{a,b,p}(G)) = \sum_{i=1}^{m} G_m^{-1} G_i s^{(\ell(\beta_i)-\ell(\beta_m))} X^{u_i} Y^{v_i}$$

e

$$\bar{g}_2 = G_m^{-1} s^{-2\ell(\beta_m)} \mathrm{clean}(R_2^{a,b,p}(G)) = \sum_{i=1}^{m} G_m^{-1} G_i s^{2(\ell(\beta_i)-\ell(\beta_m))} X^{u_i} Y^{v_i}$$

Para fixar $\ell(\beta_i) - \ell(\beta_m)$ multiplicamos $\bar{g}_1$ por $F_t s^{\ell(\alpha_t)} = \mathrm{lc}(R_1^{a,b,p}(\mathrm{lt}(F)))$ e $\bar{g}_2$ por $F_t s^{2\ell(\alpha_t)} = \mathrm{lc}(R_2^{a,b,p}(\mathrm{lt}(F)))$. Sabendo que

$$\mathrm{lc}(R_1^{a,b,p}(\mathrm{lt}(F))) = \mathrm{lc}(R_1^{a,b,p}(\mathrm{lt}(G))) \cdot \mathrm{lc}(R_1^{a,b,p}(\mathrm{lt}(H)))$$

$$F_t s^{\ell(\alpha_t)} = G_m s^{\ell(\beta_m)} \cdot H_L s^{\ell(\gamma_L)}.$$

definimos os polinômios $\tilde{g}_1$ e $\tilde{g}_2$ como segue

$$\tilde{g}_1 = F_t s^{\ell(\alpha_t)} \bar{g}_1 = \sum_{i=1}^{m} H_L G_i s^{(\ell(\beta_i)+\ell(\gamma_L))} X^{u_i} Y^{v_i} \tag{3.14}$$

e

$$\tilde{g}_2 = F_t s^{2\ell(\alpha_t)} \bar{g}_2 = \sum_{i=1}^{m} H_L G_i s^{2(\ell(\beta_i)+\ell(\gamma_L))} X^{u_i} Y^{v_i}. \tag{3.15}$$

Usando as definições de $\bar{g}_1$ e $\bar{g}_2$ obtemos os seguintes polinômios

$$\tilde{g}_1 = F_t s^{\ell(\alpha_t)} \bar{g}_1 = \sum_{i=1}^{m} F_t s^{\ell(\alpha_t)} C_{1,i} X^{u_i} Y^{v_i} = \sum_{i=1}^{m} \tilde{C_{1,i}} X^{u_i} Y^{v_i} \tag{3.16}$$

com $\tilde{C_{1,i}} = F_t s^{\ell(\alpha_t)} C_{1,i}$, e

$$\tilde{g}_2 = F_t s^{2\ell(\alpha_t)} \bar{g}_2 = \sum_{i=1}^{m} F_t s^{2\ell(\alpha_t)} C_{2,i} X^{u_i} Y^{v_i} = \sum_{i=1}^{m} \tilde{C_{2,i}} X^{u_i} Y^{v_i} \tag{3.17}$$

com $\tilde{C_{2,i}} = F_t s^{2\ell(\alpha_t)} C_{2,i}$. Igualando (3.14) com (3.16) e (3.15) com (3.17) obtemos o seguinte sistema:

$$\tilde{C_{1,i}} = H_L G_i s^{(\ell(\beta_i)+\ell(\gamma_L))}$$

e

$$\tilde{C_{2,i}} = H_L G_i s^{2(\ell(\beta_i)+\ell(\gamma_L))}$$

para $0 \leq i \leq m$. Usando logaritmo discreto na próxima equação

$$(\tilde{C_{1,i}})^{-1}(\tilde{C_{2,i}}) = s^{(\ell(\beta_i)+\ell(\gamma_L))}. \tag{3.18}$$

encontramos $\ell(\beta_i) + \ell(\gamma_L) \bmod (p-1)$ e o convertemos em base d

$$\ell(\beta_i) + \ell(\gamma_L) = \underbrace{(\beta_{i,1} + \gamma_{L,1})}_{x_1^{\beta_{i,1}+\gamma_{L,1}}} + \cdots + \underbrace{(\beta_{i,n} + \gamma_{L,n})d^{n-1}}_{x_n^{\beta_{i,n}+\gamma_{L,n}}}$$

para construirmos o seguinte monômio multivariado

$$x_1^{\beta_{i,1}+\gamma_{L,1}} \cdots x_n^{\beta_{i,n}+\gamma_{L,n}} = x^{\beta_i+\gamma_L}.$$

Também calculamos

$$H_L G_i = \tilde{C_{1,i}} s^{-(\ell(\beta_i)+\ell(\gamma_L))} \bmod p.$$

Realizando os cálculos acima para os coeficientes $(\tilde{C_{1,i}}, \tilde{C_{2,i}})$ para $0 \leq i \leq m$ acabamos construindo o seguinte polinômio multivariado.

$$\tilde{G} = \sum_{i=0}^{m} H_L G_i x^{\beta_i+\gamma_L} \in \mathbb{F}_p[x_1, \ldots, x_n]. \tag{3.19}$$

Convertendo o polinômio (3.19) para $\mathbb{Z}$ obtemos

$$\tilde{G} = H_L x^{\gamma_L} G \in \mathbb{Z}[x_1, \ldots, x_n]. \tag{3.20}$$

Tirando o conteúdo inteiro do polinômio (3.20) e dividindo por x^{γ_L} recuperamos G. Acabamos de provar que usando os fatores de $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ podemos recuperar o fator G de F usando logaritmos discretos desde que $\text{nops}(G) = \text{nops}(R_1^{a,b,p}(G))$ seja satisfeita. □

Gostaríamos de observar que no Teorema 3.4.1 poderíamos retirar a hipótese $\text{clean}(R_1^{a,b,p}(F))$ ser irredutível sobre $\mathbb{F}_p$ se na equação (3.13) considerarmos todas as combinações possíveis entre os fatores $g_1, \ldots, g_r$ sobre $\mathbb{F}_p$. Similarmente poderíamos retirar a hipótese $\text{clean}(R_2^{a,b,p}(F))$ irredutível sobre $\mathbb{F}_p$.

3.5 Algoritmo Prático

No Algoritmo 3.4.1 escolhemos um primo satisfazendo as cotas da Seção 3.7 e isto leva a primos grandes, o que reduz a eficiência prática do algoritmo. Para resolver este problema apresentaremos nesta seção um algoritmo que trabalha com alguns números primos relativamente pequenos e então usa uma generalização do algoritmo de Garner para recuperar os fatores esparsos do polinômio multivariado.

Dado $F = \sum_{i=0}^{t} F_i x^{\alpha_i} \in \mathbb{Z}[x_1, \ldots, x_n]$, nossa meta é calcular os fatores esparsos irredutíveis de F. Suponha que G é um fator esparso irredutível de F . Então diremos que um número primo p é bom se

$$\text{nops}(G) = \text{nops}(G \bmod p).$$

Em outras palavras, p é um número primo bom se p não divide qualquer coeficiente de G.

Propriedade 3.5.1. *Dado um polinômio multivariado* $G = \sum_{i=0}^{m} G_i x^{\beta_i} \in \mathbb{Z}[x_1, \ldots, x_n]$ *e seja* $P \subseteq \mathbb{N}$ *um subconjunto finito de números primos com* $B_0 = min(P)$*. Se* p *é escolhido aleatoriamente de* P*, então*

$$P(nops(G) = \ nops(G \ mod \ p)) \geq \left(1 - \frac{log_{B_0}|G|_\infty}{\#P}\right)^{m+1}. \tag{3.21}$$

Dem.: Note que

$$P(\text{nops}(G) = \text{nops}(G \bmod p)) = P(p \nmid G_i \text{ para } 0 \leq i \leq m) = \prod_{i=0}^{m} P(p \nmid G_i).$$

Mas $P(p \nmid G_i) = 1 - P(p \mid G_i)$ e $P(p \mid G_i) \leq \frac{\log_{B_0}(G_i)}{\#P}$, o que implica que

$$P(\text{nops}(G) = \text{nops}(G \bmod p)) \geq \prod_{i=0}^{m} \left(1 - \frac{\log_{B_0}(G_i)}{\#P}\right) \geq \left(1 - \frac{\log_{B_0}(|G|_\infty)}{\#P}\right)^{m+1}.$$

□

Observando as Propriedades 3.3.1 e 3.5.1 podemos concluir que se G é um fator esparso de F então

$$\text{nops}(G) = \text{nops}(R_1^{a,b,p}(G))$$

acontece com boa probabilidade.

Suponha que $G = \sum_{i=0}^{m} G_i x^{\beta_i}$ seja um fator esparso irredutível de F e $H = \sum_{i=0}^{L} H_i x^{\gamma_i}$ seu cofator em $\mathbb{Z}[x_1, \ldots, x_n]$. Dados primos bons $p_1, \ldots, p_r$, ou seja, p_j não divide qualquer coeficiente de G para $1 \leq j \leq r$, e inteiros não negativos a_i, b_i para $1 \leq i \leq n$ tais que $\text{nops}(G) = \text{nops}(R_1^{a,b,p_j}(G))$ para $1 \leq j \leq r$. Se considerarmos que $\text{clean}(R_1^{a,b,p_j}(G))$ e $\text{clean}(R_2^{a,b,p_j}(G))$ são irredutíveis sobre $\mathbb{F}_p$ para $1 \leq j \leq r$. Então, usando cálculos similares aos apresentados na demonstração do Teorema 3.4.1, podemos calcular $H_L \cdot G_i \bmod p_j$ e $\ell(\beta_i) + \ell(\gamma_L) \bmod p_j - 1$ para $1 \leq i \leq m$. E usando cada primo p_j com $1 \leq j \leq r$ obtemos

$$(\ell(\beta_i) + \ell(\gamma_L)) \bmod p_1 - 1, \ldots, (\ell(\beta_i) + \ell(\gamma_L)) \bmod p_r - 1 \tag{3.22}$$

e

$$H_L \cdot G_i \bmod p_1, \ldots, H_L \cdot G_i \bmod p_r \tag{3.23}$$

para $1 \leq i \leq m$. E usando o algoritmo de Garner generalizado (veja Seção 3.6) com as congruências (3.22) e (3.23) obtemos

$$(\ell(\beta_i) + \ell(\gamma_L)) \bmod (\text{mmc}(p_1 - 1, \ldots, p_r - 1)) \tag{3.24}$$

e

$$H_L \cdot G_i \bmod (p_1 \cdots p_r) \tag{3.25}$$

para $1 \leq i \leq m$. Ainda seguindo os passos da demonstração do Teorema 3.4.1 construímos o seguinte polinômio multivariado:

$$\tilde{G} := \sum_{i=1}^{m} H_L G_i x^{\beta_i + \gamma_L} \in \mathbb{F}_{p_1 \cdots p_r}[x_1, \ldots, x_n]. \tag{3.26}$$

E convertendo o polinômio (3.26) para $\mathbb{Z}$, obtemos

$$\tilde{G} = H_L \cdot x^{\gamma_L} \cdot G \in \mathbb{Z}[x_1, \ldots, x_n]. \tag{3.27}$$

Tirando o conteúdo inteiro do polinômio (3.27) e dividindo por x^{γ_L} recuperamos o fator esparso irredutível G de F.

Considere $F \in \mathbb{Z}[x_1, \ldots, x_n]$, $a_1, \ldots, a_n$ e $b_1, \ldots, b_n$ inteiros não negativos. Então diremos que Padrão(p) = Padrão(q) se

$$\text{nops(Padrão}(p)) = \text{nops(Padrão}(q))$$

e se, para cada par $(g_1, g_2) \in$ Padrão(p), existe um par $(h_1, h_2) \in$ Padrão(q) tal que Supp(g_1) = Supp(h_1).

Suponha que tenhamos $p_1, \ldots, p_r$ números primos tais que

$$\text{Padrão}(p_1) = \cdots = \text{Padrão}(p_r).$$

Então defina Supp = [Padrão$(p_1), \ldots,$ Padrão(p_r)].

Algoritmo 3.5.1. (*Construção de polinômios multivariados*)
Entrada: $(F, Supp)$.
Saída: Fatores esparsos de F *ou Falhou.*

passo 0: Inicialize $FS := [\ ]$;

passo 1: ***Para*** *cada par* $(g_1, g_2) \in Padrão(p_1)$ ***faça***

passo 1.1: Defina

$$C_{g_1,g_2} = [(g_1^1, g_2^1), \ldots, (g_1^r, g_2^r)]$$

onde $(g_1^i, g_2^i) \in$ *Padrão*(p_i) *e* $Supp(g_1^i) = Supp(g_1)$*.(Note que* $(g_1, g_2) = (g_1^1, g_2^2)$*).*

passo 1.2: ***Para*** $j = 1$ ***até*** r ***faça***

$$\tilde{g_1^j} := lc(R_1^{a,b,p_j}(lt(F)))g_1^j \in \mathbb{F}_{p_j}[X, Y]$$

$$\tilde{g_2^j} := lc(R_2^{a,b,p_j}(lt(F)))g_2^j \in \mathbb{F}_{p_j}[X, Y]$$

passo 1.3: ***Para*** *cada* $e = (e_x, e_y) \in Supp(g_1)$ ***faça***

passo 1.3.1: $\{$*Logaritmo discreto*$\}$

Para $j = 1$ ***até*** r ***calcule*** $\ell_{e,j}$ *mod* $(p_j - 1)$ *usando logaritmo discreto na seguinte equação*

$$(coeff(\tilde{g_1^j}, X^{e_x}Y^{e_y}))^{-1} \cdot coeff(\tilde{g_2^j}, X^{e_x}Y^{e_y}) = s_j^{\ell_{e,j}}$$

e

$$c_{e,j} = s_j^{-\ell_{e,j}} \cdot coeff(\tilde{g_1^j}, X^{e_x}Y^{e_y}) \ mod \ p_j.$$

passo 1.3.2: $\{$*Algoritmo de Garner generalizado*$\}$

$$\ell_e = Garner([\ell_{e,1}, \ldots, \ell_{e,r}], [p_1 - 1, \ldots, p_r - 1])$$

$$c_e = Garner([c_{e,1}, \ldots, c_{e,r}], [p_1, \ldots, p_r])$$

passo 1.4: Defina

$$\tilde{G} := \textstyle\sum_{e \in Supp(g_1)} c_e x_1^{\ell_{e_1}} x_2^{\ell_{e_2}} \cdots x_n^{\ell_{e_n}} \in \mathbb{F}_{p_1 \cdots p_r}[x_1, \ldots, x_n]$$

onde $\ell_e = \ell_{e_1} + \ell_{e_2} d + \cdots + \ell_{e_n} d^{n-1}$.

passo 1.5: Converta $\tilde{G}$ *para* $\mathbb{Z}[x_1, \ldots, x_n]$ *e defina*

$$G = \frac{clean(\tilde{G})}{iconteúdo(\tilde{G})} \in \mathbb{Z}[x_1, \ldots, x_n] \ .$$

passo 1.6: ***Se*** $(G|F)$ ***então*** $FS := [op(FS), G]$.

passo 2: ***Se*** $(FS = [\ ])$ ***então*** *retorne*$(Falhou)$ ***senão*** *retorne*(FS).

Note que, no passo 1.3.2 do algoritmo 3.5.1 usamos o algoritmo de Garner generalizado (ver seção 3.6) pois os números $p_1 - 1, \ldots, p_r - 1$ não são coprimos.

Agora apresentaremos um algoritmo iterativo. A cada passo escolhemos um número primo diferente e então aplicamos o algoritmo 3.4.2 para checar se podemos obter fatores esparsos usando apenas este primo. Se isto não funcionar, então mantemos os fatores bivariados esparsos para construirmos o conjunto Supp e deste modo chamarmos o Algoritmo 3.5.1.

Algoritmo 3.5.2. (*Algoritmo prático*)
Entrada: $F \in \mathbb{Z}[x_1, \ldots, x_n]$.
Saída: Fatores esparsos de F *ou Falhou.*

passo 1: Inicialize Padrões$:= [\ ]$ *e escolha um conjunto de números primos* $P = \{p_1, \ldots, p_r\}$ *satisfazendo as cotas da Seção 3.7. Escolha* a_i *e* b_i *aleatoriamente de um subconjunto* $S = \{0, 1, 2, \ldots, c\}$ *de* $\mathbb{N}$ *para* $1 \leq i \leq n$.

passo 2: Retire $p \in P$.

Fatore $R_1^{a,b,p}(F)$ *e* $R_2^{a,b,p}(F)$ *em* $\mathbb{F}_p[X, Y]$.

Calcule Esparsos$(clean(R_1^{a,b,p}(F)))$ *e Esparsos*$(clean(R_2^{a,b,p}(F)))$.

Se $(Esparsos(clean(R_1^{a,b,p}(F))) = [\ ])$ *ou* $(Esparsos(clean(R_2^{a,b,p}(F))) = [\ ])$ ***então***

- ***retorne****(*F *provavelmente não tem fatores esparsos) e* ***pare*** *o algoritmo.*

senão *Calcule Padrão*(p).
Se $(Padrão(p) = [\,])$ ***então***

- *vá para o passo 2.*

senão$(Padrão(p) \neq [\,])$ ***então***

- ***chame***$(Algoritmo\ 3.4.2(F, Padrão(p)))$ *para tentar construir fatores esparsos de* F. *Se encontrar fatores esparsos* ***então*** *retorne-os e* ***pare*** *o algoritmo.* ***Se*** *Falhou acontecer, continue o procedimento.*

[Vamos verificar se $Padrão(p)$ *está em um padrão existente ou irá gerar um novo padrão]*
Se $(Padrão(p) \in Supp$, *para algum* $Supp \in Padrões)$ ***então***

- $Supp := [op(Supp), Padrão(p)]$.
- ***chame***$(Algoritmo\ 3.5.1(F, Supp))$ *para tentar construir fatores esparsos de* F. ***Se*** *encontrar fatores esparsos então retorne-os e* ***pare*** *o algoritmo.* ***Se*** *Falhou acontecer, vá para o passo 2.*

Senão$(Padrão(p)$ *gera um novo padrão)* ***então***

- $Supp(nops(Padrões) + 1) := [Padrão(p)]$.
- $Padrões := [op(Padrões), Supp(nops(Padrões) + 1)]$.
- *vá para o passo 2.*

Gostaríamos de salientar que quando calculamos o conjunto P no algoritmo acima sempre escolhemos números primos que não dividam o coeficiente líder de F.

3.6 Algoritmo de Garner generalizado

Nesta seção iremos estudar o algoritmo de Garner generalizado que é usado no algoritmo 3.5.1, pois neste método podemos trabalhar com números $m_1, \ldots, m_t$ (veja equação (3.28)) que não são coprimos entre si.

Suponha que R é um domínio Euclidiano. Dados $m_1, \ldots, m_t \in R$ e $r_1, r_2, \ldots, r_t \in R$. Queremos encontrar um elemento $a \in R$ tal que

$$a \equiv r_i (\bmod m_i) \text{ para } 1 \leq i \leq t. \tag{3.28}$$

Representamos um elemento de R em base mista:

$$a = a_0 + a_1 m_1 + a_2 m_1 m_2 + \cdots + a_{t-1} m_1 \cdots m_{t-1}$$

onde $a_i \in R$, é um resíduo módulo m_{i+1}, $0 \leq i \leq t-1$, dito representante da base mista. Calcularemos a_i, iterativamente começando em a_0. Primeiro $a \equiv r_1 (\bmod m_1)$ e $a \equiv a_0 (\bmod m_1)$. Então obtemos

$$a_0 = r_1 \bmod m_1.$$

Para encontrar a_1 note que $a \equiv r_2 (\bmod m_2)$ e $a \equiv a_0 + a_1 m_1 (\bmod m_2)$. Logo $r_2 - a_0 \equiv a_1 m_1 (\bmod m_2)$. Se $d_1 = \text{mdc}(m_1, m_2)$ divide $r_2 - a_0$ então

$$\tfrac{r_2 - a_0}{d_1} \equiv \tfrac{a_1 m_1}{d_1} (\bmod \tfrac{m_2}{d_1})$$

e obtemos

$$a_1 = (\tfrac{m_1}{d_1})^{-1} \tfrac{r_2 - a_0}{d_1} (\bmod \tfrac{m_2}{d_1}).$$

Se d_1 não divide $r_2 - a_0$, então (3.28) não tem solução e paramos o procedimento. Até agora temos que $a = a_0 + a_1 m_1$. Em geral, suponha que tenhamos encontrado $a_0, a_1, \ldots, a_{i-1}$ tais que $b = a_0 + a_1 m_1 + \cdots + a_{i-1} m_1 \cdots m_{i-1}$ satisfaz $b \equiv r_{j+1} (\bmod m_{j+1})$ com $a_j = (\frac{m_1 \cdots m_j}{d_j})^{-1} (\frac{r_{j+1} - b}{d_j}) (\bmod \frac{m_{j+1}}{d_j})$ onde $d_j = \text{mdc}(m_1 \cdots m_j, m_{j+1})$ divide $r_{j+1} - b$ para $0 \leq j \leq i-1$. Queremos encontrar a_i tal que

$$b + a_i m_1 \cdots m_i \equiv r_{i+1}(\bmod m_{i+1}).$$

Então $r_{i+1} - b \equiv a_i m_1 \cdots m_i(\bmod m_{i+1})$ e se $d_i = \text{mdc}(m_1 \cdots m_i, m_{i+1})$ divide $r_{i+1} - b$ obtemos que

$$\tfrac{r_{i+1}-b}{d_i} \equiv a_i(\tfrac{m_1 \cdots m_i}{d_i})(\bmod \tfrac{m_{i+1}}{d_i})$$

e assim

$$a_i = (\tfrac{m_1 \cdots m_i}{d_i})^{-1}(\tfrac{r_{i+1}-b}{d_i})(\bmod \tfrac{m_{i+1}}{d_i}).$$

Então $a = b + a_i m_1 \cdots m_i$ satisfaz $a \equiv r_{i+1}(\bmod m_{i+1})$. Se d_i não divide $r_{i+1} - b$, então (3.28) não tem solução e paramos o procedimento.

Algoritmo 3.6.1. *(Algoritmo de Garner generalizado)*
Entrada: $m_1, \ldots, m_r \in R$ *e* $r_1, \ldots, r_t \in R$
Saída: $a \in R$ *satisfazendo (3.28) ou Falhou.*

Passo 1: $a := r_1 \ mod \ m_1$, $m := m_1$.

Passo 2: ***Para*** $i := 1$ ***até*** $t - 1$ ***faça***
$d := mdc(m, r_{i+1})$,
se $(d \ divide \ r_{i+1} - a)$ ***então***
$u := (\frac{m}{d})^{-1} \ mod \ \frac{m_{i+1}}{d}$,
$v := u(\frac{r_{i+1}-a}{d}) \ mod \ \frac{m_{i+1}}{d}$,
$a := a + vm$,
$m := m \cdot m_{i+1}$
senão
retorne$(Falhou)$ *e pare o algoritmo.*

Passo 3: ***retorne***(a).

3.7 Cotas

Nesta seção calculamos as cotas teóricas necessárias em cada algoritmo.

Começamos calculando as cotas teóricas para o algoritmo 3.4.1.

Para calcular $\ell(\beta_i)+\ell(\gamma_L)$ na equação (3.18) usando logaritmo discreto precisamos

$$\ell(\beta_i)+\ell(\gamma_L)=(\beta_{i,1}+\gamma_{L,1})+(\beta_{i,2}+\gamma_{L,2})d+\cdots+(\beta_{i,n}+\gamma_{L,n})d^{n-1}<p-1.$$

Mas

$$\beta_{i,1}+\gamma_{L,1},\beta_{i,2}+\gamma_{L,2},\ldots,\beta_{i,n}+\gamma_{L,n}<d=2(\max(\text{grau}_{x_i}(F)))+1$$

o quê implica

$$(\beta_{i,1}+\gamma_{L,1})+(\beta_{i,2}+\gamma_{L,2})d+\cdots+(\beta_{i,n}+\gamma_{L,n})d^{n-1}<d+d^2+\cdots+d^n=\tfrac{d(d^n-1)}{d-1}<2d^n.$$

Então precisamos de um número primo p satisfazendo $p\geq 2d^n$. Na equação (3.19) precisamos que

$$p>|H_L\cdot G_i|.$$

Mas $|H_L\cdot G_i|=|H_L|\cdot|G_i|\leq|F_t|\cdot|G|_\infty$. Usando [6] temos que $|G|_\infty<c^{n\cdot d}|F|_\infty$ onde $c<\sqrt{6}$. Isto implica que

$$p>|F_t|c^{n\cdot d}|F|_\infty.$$

Concluímos que devemos trabalhar com um primo p satisfazendo $p>\max\{2d^n,|F_t|c^{n\cdot d}|F|_\infty\}$.

Observação: Queremos resolver a equação (3.18) eficientemente. Para este propósito usaremos o método de Pohlig-Hellman [23] o qual tira vantagem da

fatoração da ordem do grupo. Defina $B = \max\{2d^n, |F_t|c^{n\cdot d}|F|_\infty\}$, então queremos calcular o menor inteiro positivo i tal que p é primo e $|\mathbb{F}_p^*| = p - 1 = 2^m i$ onde $m = \lceil \log_2 B \rceil$.

Agora, iremos calcular as cotas para os algoritmos 3.5.1 e 3.5.2.

Para calcular $\ell(\beta_i) + \ell(\gamma_L)$ na equação (3.24) usando o algoritmo de Garner generalizado precisamos

$$\ell(\beta_i) + \ell(\gamma_L) = (\beta_{i,1} + \gamma_{L,1}) + (\beta_{i,2} + \gamma_{L,2})d + \cdots + (\beta_{i,n} + \gamma_{L,n})d^{n-1} < \operatorname{mmc}(p_1 - 1, \ldots, p_r - 1).$$

Mas

$$\beta_{i,1} + \gamma_{L,1}, \beta_{i,2} + \gamma_{L,2}, \ldots, \beta_{i,n} + \gamma_{L,n} < d = 2(\max(\operatorname{grau}_{x_i}(F))) + 1$$

o que implica

$$(\beta_{i,1}+\gamma_{L,1})+(\beta_{i,2}+\gamma_{L,2})d+\cdots+(\beta_{i,n}+\gamma_{L,n})d^{n-1} < d+d^2+\cdots+d^n = \tfrac{d(d^n-1)}{d-1} < 2d^n.$$

Então precisamos de um conjunto de números primos $p_1, \ldots, p_r$ tais que

$$\operatorname{mmc}(p_1 - 1, \ldots, p_r - 1) \geq 2d^n. \tag{3.29}$$

Na equação (3.25) precisamos que

$$p_1 \cdots p_r > |H_L \cdot G_i|.$$

Mas $|H_L \cdot G_i| = |H_L| \cdot |G_i| \leq |F_t| \cdot |G|_\infty$. Usando [6] sabemos que $|G|_\infty < c^{n\cdot d}|F|_\infty$ onde $c < \sqrt{6}$. Isto implica que precisamos

$$p_1 \cdots p_r > |F_t|c^{n\cdot d}|F|_\infty \tag{3.30}$$

Logo, devemos trabalhar com um conjunto de números primos $p_1, \ldots, p_r$ satisfazendo as condições (3.29) e (3.30).

3.8 Análise

Nesta seção contaremos a complexidade dos algoritmos pelo número de operações usadas em $\mathbb{F}_p$. Dados a, $n \in \mathbb{Z}$ podemos calcular $a^n \bmod p$ via quadrados repetidos, veja [33] pag. 73, usando $O(\log(n))$ operações em $\mathbb{F}_p$. Em [36] um polinômio $f(x, y)$ sobre $\mathbb{F}_p$ de grau total n pode ser completamente fatorado usando $O(\mathrm{n}^{4.89}\log^2(\mathrm{n})\log(\mathrm{p}))$ operações em $\mathbb{F}_p$. Se $(p-1)$ tem somente fatores primos pequenos então [23] necessita de $O(\log^2 p)$ operações para calcular logaritmos discreto sobre $\mathbb{F}_p^*$.

No próximo teorema contaremos apenas as operações principais executadas nos algoritmos 3.4.1 and 3.5.2.

Teorema 3.8.1. *Para n fixo, espera-se que o Algoritmo 3.4.1, quando aplicado a um polinômio $F \in \mathbb{Z}[x_1, \ldots, x_n]$ com $M = max_{1 \le i \le n}(grau_{x_i}(F))$, termine usando*

$$O(nops(F)log(M) + M^{5.89}log^2(M)) \tag{3.31}$$

operações em $\mathbb{F}_p$, onde p é um primo satisfazendo $p = O(\gamma^M)$ para certa constante $\gamma < \sqrt{6}$, conforme Seção 3.7, e $nops(F) \le (M+1)^n$.

Dem.: Vamos considerar que os inteiros não negativos a_i, b_i sejam limitados por $c \in \mathbb{N}$ para $1 \le i \le n$ e $d = 2M + 1$. No passo 2, calculamos primeiro $(d^{i-1} \bmod p-1)$ e então $(s^{d^{i-1}} \bmod p)$ para $1 \le i \le n$. Note que $d^{i-1} \le d^{n-1}$ e calculamos $(d^{n-1} \bmod p-1)$ usando $\lceil \log(n-1) \rceil$ operações em $\mathbb{F}_p$. Em seguida calculamos $(s^{d^{i-1}} \bmod p)$ onde $(d^{i-1} \le p-1)$. Então calculamos $(s^{p-1} \bmod p)$ usando $O(\log(p-1))$ operações em $\mathbb{F}_p$. Logo, via quadrados repetidos $s^{d^{i-1}} \bmod p$ para $1 \le i \le n$ pode ser feito em $O(n\log(n-1)+n\log(p-1))$ operações. Depois disso, considerando a redução 3.7, substituímos cada x_i em F por $s^{d^{i-1}} X^{a_i} Y^{b_i} \bmod p$. No pior caso F tem um monômio $x_1^M \cdot x_2^M \cdots x_n^M$. Note que $s^{d^{i-1}} \bmod p < p$, então usamos, para calcular n vezes $(p^M \bmod p)$, $O(n \log(M))$ operações em $\mathbb{F}_p$ para cada monômio de F. Logo o passo 2 usa

$$O(n\log(n-1) + n\log(p-1) + \mathrm{nops}(F)n\log(M)) = O(M + \mathrm{nops}(F)\log(M)). \tag{3.32}$$

operações em $\mathbb{F}_p$ considerando que $\text{nops}(F) = O(M+1)^n$. No passo 3, a operação principal é fatorar $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ sobre $\mathbb{F}_p$. Note que $\text{grau}(R_1^{a,b,p}(F))$, $\text{grau}(R_2^{a,b,p}(F)) \leq 2nMc$. Então a fatoração bivariada pode ser feita usando [36] em

$$O((2nMc)^{4.89}\log^2(2nMc)\log(p)) = O(M^{5.89}\log^2(M)) \tag{3.33}$$

operações em $\mathbb{F}_p$. No passo 4, quando chamamos o algoritmo 3.4.2, nossa principal operação será o cálculo dos logaritmos discreto. Suponha que G é um fator esparso de F. Então, pela nossa definição de esparsidade, $\text{nops}(G) \leq n \cdot (\max_{1\leq i\leq n}(\text{grau}_{x_i}(G))) \leq n \cdot M$. Agora, se $(g_1, g_2) \in \text{Padrão}(p)$, então $\text{nops}(g_1) \leq \text{nops}(G) \leq n \cdot M$. Logo, se considerarmos que F tem r fatores esparsos então no passo 4 calculamos $r(nM)$ logaritmos discretos e usamos no total

$$O(rnM\log^2(p)) = O(M^4) \tag{3.34}$$

operações em $\mathbb{F}_p$ com $r \leq nM$. Considerando as operações realizadas em (3.32), (3.33) e (3.34). Espera-se que o algoritmo 3.4.1 termine usando $O(\text{nops}(F)\log(M) + M^{5.89}\log^2(M))$ operações em $\mathbb{F}_p$. □

Gostaríamos de observar que para representar um número em $\mathbb{F}_p$ são necessários $O(\log(p))$ bits. A operação de multiplicação de elementos representados com até N bits tem custo de $O(N^2)$ operações binárias. Inversos em $\mathbb{F}_p$ podem ser calculados usando o algoritmo de Euclides Estendido, que, para entradas com até N bits usa $O(N^2)$ operações binárias. Então espera-se que o Algoritmo 3.4.1 termine usando

$$O(M^2\text{nops}(F)\log(M) + M^{7.89}\log^2(M)) \tag{3.35}$$

operações binárias.

No Algoritmo 3.5.2 usamos o algoritmo de Garner generalizado que pode ser calculado em $O(n^2)$ operações em $\mathbb{F}_p$ conforme [11]. Onde, n é o número de congruências r_i mod m_i, veja seção 3.6, e cada $m_i < p$.

Teorema 3.8.2. *Para n fixo, espera-se que o Algoritmo 3.5.2, quando aplicado a um polinômio* $F \in \mathbb{F}[x_1, \ldots, x_n]$ *com* $M = max_{1\leq i\leq n}(grau_{x_i}(F))$*, termine usando*

$$O(\sigma M^{4.89} log^2(M) + \sigma nops(F) log(M)) \tag{3.36}$$

operações sobre corpos finitos de precisão simples. Na equação (3.36), σ representa o número de primos usados pelo Algoritmo 3.5.2 e $p = max_{1\leq i\leq\sigma}\{p_i\}$. Conforme a Seção 3.7 temos que $\sigma = O(M)$, e $nops(F) \leq (M+1)^n$.

Dem.: Vamos considerar que os inteiros não negativos a_i, b_i sejam limitados por $c \in \mathbb{N}$ para $1 \leq i \leq n$ e $d = 2M + 1$. Suponha que tenhamos primos bons $p_1 < p_2 < \ldots < p_\sigma = p$ e inteiros não negativos $a_1, \ldots, a_n$ e $b_1, \ldots, b_n$ tal que $\text{nops}(G) = \text{nops}(R_1^{a,b,p_i}(G))$ para cada p_i com $1 \leq i \leq \sigma$. Consideramos que se G é um fator irredutível de F então $\text{clean}(R_1^{a,b,p_i}(G))$ e $\text{clean}(R_2^{a,b,p_i}(G))$ são irredutíveis sobre $\mathbb{F}_p$ para $1 \leq i \leq \sigma$. No algoritmo 3.5.2 podemos calcular $R_1^{a,b,p_i}(F)$ e $R_2^{a,b,p_i}(F)$ usando $O(n\log(n-1) + n\log(p_i - 1) + \text{nops}(F)n\log(M))$ operações em $\mathbb{F}_p$. Logo, para $1 \leq i \leq \sigma$ usamos um total de

$$O(\sigma n\log(n-1) + n\sum_{i=1}^{\sigma}\log(p_i - 1) + \sigma\text{nops}(F)n\log(M)) = O(\sigma\text{nops}(F)\log(M)) \tag{3.37}$$

operações em $\mathbb{F}_p$. As (2σ) fatorações bivariadas podem ser feitas em

$$O((2Mcn)^{4.89}\log^2(2Mcn)\sum_{i=1}^{\sigma}\log(p_i)) = O(\sigma M^{4.89}\log^2(M)) \tag{3.38}$$

operações em $\mathbb{F}_p$. Quando chamamos o algoritmo 3.5.1, as operações principais serão os logaritmos discreto e o algoritmo de Garner generalizado. Consideremos que G seja um fator esparso de F. Então, pela nossa definição de esparsidade, $\text{nops}(G) \leq n \cdot \max_{1\leq i\leq n}(\text{grau}_{x_i}(G)) \leq nM$. Temos, então, que $\text{nops}(\text{Padrão}(p_i)) \leq nM$. Para cada par $(g_1, g_2) \in \text{Padrão}(p_1)$ fazemos nM logaritmos discreto para cada $1 \leq i \leq \sigma$. Em um total de $O(nM\sum_{i=1}^{\sigma}\log^2(p_i))$ operações em $\mathbb{F}_p$. Considerando que F tem r fatores esparsos realizamos

$$O(rnM\sum_{i=1}^{\sigma}\log^2(p_i)) = O(\sigma M^2) = O(M^3) \tag{3.39}$$

operações em $\mathbb{F}_p$. Quando chamamos o algoritmo de Garner generalizado fazemos

$$O(\sum_{i=1}^{\sigma} i^2) = O(\sigma^3) = O(M^3) \tag{3.40}$$

operações em $\mathbb{F}_p$. Observando (3.37), (3.38), (3.39) e (3.40), esperamos que o Algoritmo 3.5.2 termine usando $O(\sigma M^{4.89}\log^2(M) + \sigma \text{nops}(F)\log(M))$ operações sobre corpos finitos de precisão simples.

□

Nos Algoritmos 3.4.1 e 3.5.2, cada vez que construímos um candidato a fator multivariado $G \in \mathbb{Z}[x_1, \ldots, x_n]$ de F testamos se G divide F em $\mathbb{Z}[x_1, \ldots, x_n]$. Gostaríamos de observar que esta operação não foi adicionada nas análises acima pois estamos contando apenas as operações realizadas em $\mathbb{F}_p$.

3.9 Experimentos Computacionais

Nesta seção, apresentaremos nossos experimentos computacionais. Nas tabelas a seguir iremos comparar nosso algoritmo de fatoração 3.5.2 com o algoritmo de fatoração usado no programa de computador matemático Maple. Iremos citar os passos principais usados no processo de fatoração multivariada do programa Maple, para maiores detalhes veja o artigo [4]. Dado $F(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$, então eles reduzem F para o polinômio bivariado $F(x_1, x_2, a_3, \ldots, a_n)$ com $a_3, \ldots, a_n \in \mathbb{F}$ pois esta substituição produzirá a fatoração padrão correta na maioria das vezes. Os fatores bivariados de $F(x_1, x_2, a_3, \ldots, a_n)$ são então levantados usando levantamento de Hensel para a contrução dos verdadeiros fatores multivariados.

Para cada linha das próximas tabelas fizemos 10 iterações e então calculamos a média. A cada iteração escolhemos os fatores esparsos e densos aleatoriamente e então multiplicamos estes fatores. Depois fatoramos este produto usando nosso algoritmo e usando o Maple separadamente.

Esparsos

Nas próximas tabelas fatoramos polinômios multivariados que possuem apenas fatores esparsos. Na coluna Esparsos colocamos o número de fatores esparsos

dos polinômios usados. Na coluna tempo(Maple) colocamos o tempo gasto pelo programa matemático Maple para fatorar os polinômios e na coluna tempo(Algoritmo 3.5.2) colocamos o tempo gasto pelo Algoritmo 3.5.2 para fatorar os mesmos polinômios.

Na tabela a seguir fatoramos polinômios em $\mathbb{Z}[x_1, \ldots, x_5]$. Note que a célula localizada na terceira coluna e quarta linha está vazia. Isto significa que durante nossos experimentos o programa Maple não conseguiu fatorar tais polinômios.

Grau total de F	Esparsos	tempo(Maple)	tempo(Algoritmo 3.5.2)
72	6	5.87 min	2.93 min
83	7	25.91 min	5.09 min
95	8	23.31 min	12,79 min
106	9	-	31.87 min

Na tabela a seguir fatoramos polinômios em $\mathbb{Z}[x_1, \ldots, x_6]$.

Grau total de F	Esparsos	tempo(Maple)	tempo(Algoritmo 3.5.2)
60	5	3.02 min	1.68 min
72	6	15.28 min	3.70 min
84	7	47.65 min	8.52 min
96	8	77.73 min	26.39 min

Na tabela a seguir fatoramos polinômios em $\mathbb{Z}[x_1, \ldots, x_7]$.

Grau total de F	Esparsos	tempo(Maple)	tempo(Algoritmo 3.5.2)
48	4	0.30 min	0.50 min
60	5	6.13 min	1.35 min
72	6	34.80 min	2.24 min

Esparsos e densos

Nas próximas tabelas fatoramos polinômios multivariados que possuem fatores esparsos e densos. Nos experimentos que apresentaremos abaixo procedemos da seguinte maneira: dado um polinômio multivariado, primeiro usamos o Algoritmo 3.5.2 para extrairmos os fatores esparsos. Depois usamos o programa Maple para fatorar os fatores densos. O tempo gasto por este método será denotado por tempo(híbrido) que significa tempo(Algoritmo 3.5.2)+tempo(Maple). Na coluna Densos colocamos o número de fatores densos dos polinômios usados

Na tabela a seguir fatoramos polinômios que possuem fatores esparsos e densos em $\mathbb{Z}[x_1, \ldots, x_4]$.

Grau total de F	Esparsos	Densos	tempo(Maple)	tempo(híbrido)
34	2	2	0.64 min	0.40 min
51	3	3	9.76 min	3.42 min
68	4	4	57.29 min	14.19 min

Na tabela a seguir fatoramos polinômios que possuem fatores esparsos e densos em $\mathbb{Z}[x_1, \ldots, x_5]$.

Grau total de F	Esparsos	Densos	tempo(Maple)	tempo(híbrido)
34	2	2	5.24 min	1.03 min
51	3	3	147.11 min	13.93 min

Observações adicionais: Durante nossos experimentos computacionais, quando utilizamos o algoritmo 3.5.2, sempre trabalhamos com números primos da ordem de 7 dígitos pois deste modo sempre usamos números de precisão simples.

3.10 Conclusão

Neste capítulo desenvolvemos um novo algoritmo de fatoração polinomial multivariada. Gostaríamos de enfatizar que desde os anos 80 não haviam sido criados novos tipos de reduções polinomiais. Nosso método mostrou-se eficiente quando usado para fatorar polinômios multivariados que possuem apenas fatores esparsos, bem como para extrair fatores esparsos de polinômios multivariados que têm fatores densos e esparsos.

Trabalhos futuros

- Durante nossos experimentos computacionais constatamos que as reduções 3.7 e 3.8 mantiveram a irredutibilidade dos polinômios. Portanto, pretendemos obter argumentos probabilísticos, para as reduções 3.7 e 3.8, garantindo que se $F \in \mathbb{Z}[x_1, \ldots, x_n]$ é irredutível então $R_1^{a,b,p}(F)$ será irredutível sobre $\mathbb{F}_p$.

- Quando fatoramos $R_1^{a,b,p}(F)$ e $R_2^{a,b,p}(F)$ no algoritmo 3.5.2 usamos apenas seus fatores esparsos para construirmos os candidatos a fatores multivariados esparsos. Durante nossos experimentos computacionais observamos que estas fatorações bivariadas consomem grande parte do tempo gasto em nosso procedimento. Então, pretendemos desenvolver métodos rápidos para fatoração bivariada que extraiam somente os fatores esparsos dos polinômios bivariados.

4 MDC DE POLINÔMIOS MULTIVARIADOS VIA POLITOPOS DE NEWTON

Neste capítulo estudamos critérios geométricos de politopos para determinar coprimalidade entre polinômios multivariados. Nossa principal contribuição é o desenvolvimento de um algoritmo que trabalha em tempo polinomial (sobre o número de monômios) para detectar coprimalidade entre polinômios multivariados usando politopos de Newton. Também mostramos como construir o máximo divisor comum (mdc) entre dois polinômios bivariados usando seus polígonos de Newton associados. Os resultados obtidos neste capítulo foram submetidos à publicação, conforme [2].

4.1 Introdução

Um resultado probabilístico bem conhecido garante que dados dois polinômios multivariados sobre um domínio integral, eles são quase sempre primos entre si (veja na literatura, por exemplo [17, 33] ou neste capítulo na seção 4.5). Então, quando precisamos calcular o máximo divisor comum entre dois polinômios multivariados, é de grande utilidade aplicarmos um teste preliminar de coprimalidade antes de efetivamente procurarmos o mdc. Essa é nossa principal contribuição neste capítulo. Apresentamos um algoritmo que testa coprimalidade entre polinômios multivariados usando propriedades de seus politopos de Newton associados. Nosso teste preliminar mostrou-se realmente eficiente quando usado em polinômios multivariados esparsos com grau grande e coeficientes grandes. Para polinômios bivariados acabamos apresentando um algoritmo que usa seus polígonos de Newton associados para efetivamente calcular o mdc.

Um dos primeiros resultados conhecidos ligando geometria e álgebra de polinômios foi feito por Ostrowski [22] em 1921. Ele mostrou que se um polinômio fosse redutível então seu politopo de Newton associado seria decomponível em

relação à soma de Minkowski. Mais recentemente, Shuhong Gao [7, 9] usou propriedades geométricas de politopos para construir famílias de polinômios irredutíveis absolutamente, e, em [25, 26], Shuhong Gao e seus colaboradores acabaram efetivamente fatorando polinômios bivariados usando seus polígonos de Newton associados.

Neste capítulo usamos propriedades geométricas de politopos de Newton para obtermos critérios de coprimalidade entre polinômios multivariados. Observamos que, se politopos não possuem fatores em comum, então os polinômios representados por estes politopos serão coprimos. Mais precisamente, mostramos que se politopos de Newton não possuem arestas paralelas então não terão fatores em comum. Este fato é a chave de nosso teste preliminar, ou seja, dados dois polinômios multivariados, nosso algoritmo verifica se seus politopos de Newton associados possuem arestas paralelas.

Para transformarmos nosso critério em um algoritmo, precisamos determinar as arestas de um politopo. Faremos isto calculando seu grafo facial [29]. Em nossa implementação, utilizamos o programa matemático livre polymake [12] para calcularmos o grafo facial. Gostaríamos de observar que determinar o grafo facial de um politopo depende basicamente do seu número de vértices. Então, podemos observar que nosso algoritmo funcionará bem com polinômios multivariados esparsos de grau grande, pois esta classe possui politopos de Newton com poucos vértices.

Outra vantagem de nosso teste preliminar é a de que, se forem dados dois polinômios cujos politopos de Newton não possuem arestas paralelas em comum, então estes polinômios serão coprimos e permanecerão coprimos mesmo que seus coeficientes sejam modificados arbitrariamente e sobre qualquer corpo, desde que os monômios que correspondem aos vértices dos politopos permaneçam não nulos, ou seja, que os politopos de Newton permaneçam invariantes.

Para polinômios bivariados que não são relativamente primos apresentamos um algoritmo que calcula o mdc usando polígonos de Newton.

Nosso capítulo é organizado como segue. Na seção 4.2 apresentamos a terminologia e teoria necessária para o capítulo e nosso critério de coprimalidade. Na seção 4.3 apresentamos o teorema que é a chave de nosso teste preliminar. Na seção 4.4 apresentamos nosso teste preliminar e descrevemos como é feita a implementação do algoritmo. Na seção 4.5 estão os experimentos computacionais realizados, os quais indicam que nosso teste preliminar é realmente eficiente quando usado em polinômios multivariados esparsos. E terminamos o capítulo, descrevendo como construir o mdc entre dois polinômios bivariados usando seus polígonos de Newtow associados, na seção 4.6.

4.2 Politopo de Newton

Nesta seção, apresentamos o material necessário sobre a teoria de conjuntos convexos e a terminologia usada no capítulo. Fazemos a conexão entre uma propriedade geométrica de politopos e sua relação com polinômios. Para mais propriedades de politopos, veja [5, 27].

Um conjunto convexo em $\mathbb{R}^n$ é um conjunto tal que os pontos sobre o segmento de linha unindo quaisquer dois pontos do conjunto pertencem ao conjunto; a envoltória convexa de um conjunto de pontos é o menor conjunto convexo que os contenha; e a envoltória convexa de um conjunto finito de pontos é dita um politopo convexo. Um ponto de um politopo é dito um vértice se este não pertence ao interior de qualquer segmento de linha contido no politopo. Um politopo é sempre a envoltória convexa de seus vértices.

Dados dois politopos A e B definimos sua soma de Minkowski por

$$A + B = \{a + b : a \in A, b \in B\}.$$

E diremos que A e B são os *fatores* de $A + B$.

Seja $f \in \mathbb{F}[x_1, \ldots, x_n]$, um polinômio não constante sobre $\mathbb{F}$, onde $\mathbb{F}$ é um corpo arbitrário. O politopo de Newton, denotado por P_f, associado ao

polinômio

$$f = \sum f_{i_1,\ldots,i_n} x_1^{i_1} \cdots x_n^{i_n}$$

é a envoltória convexa do conjunto de suporte de f,

$$Supp(f) = \{(i_1, \ldots, i_n) \in \mathbb{N}^n : f_{i_1,\ldots,i_n} \neq 0\}.$$

Um politopo integral é um politopo cujos vértices tem coordenadas inteiras, e diremos que um politopo integral é decomponível integralmente, ou simplesmente decomponível, se este pode ser escrito como a soma de Minkowski de dois politopos integrais, cada um destes com no mínimo dois pontos integrais. Um fator em uma decomposição integral é dito um fator integral. Diremos que um politopo integral é integralmente indecomponível, ou simplesmente indecomponível, se este não é decomponível. Neste capítulo trabalharemos somente com politopos integrais pois eles sempre representarão politopos de Newton associados a polinômios.

O teorema a seguir, enunciado por Ostrowski em [22], relaciona a fatoração algébrica de um polinômio f com a geometria do seu politopo de Newton associado.

Teorema 4.2.1 (Ostrowski)**.** *Sejam f, g e $h \in \mathbb{F}[x_1, \ldots, x_n]$. Se $f = g \cdot h$ então $P_f = P_g + P_h$.*

Sejam f e $g \in \mathbb{F}[x_1, \ldots, x_n]$ dois polinômios não constantes sobre um corpo arbitrário $\mathbb{F}$. O único polinômio mônico h satisfazendo (i) $h|f$ e $h|g$, e (ii) se $s|f$ e $s|g$ então $s|h$, para todo $s \in \mathbb{F}[x_1, \ldots, x_n]$ é o *máximo divisor comum*, denotado por $h = \text{mdc}(f, g)$. Dois polinômios f e $g \in \mathbb{F}$ são ditos *relativamente primos* ou *coprimos* se $\text{mdc}(f, g) = 1$

O próximo corolário estabelece uma condição geométrica suficiente para decidir quando dois polinômios são coprimos.

Corolário 4.2.1 (Critério de Coprimalidade)**.** *Sejam f e $g \in \mathbb{F}[x_1, \ldots, x_n]$, ambos não divisíveis por x_i para todo $1 \leq i \leq n$. Se P_f e P_g não possuem fatores integrais*

em comum então

$$mdc(f, g) = 1.$$

Dem.: Suponha por contradição que $\text{mdc}(f, g) = h$, onde $\text{grau}(h) \geq 1$ e h não possui somente um termo pois f e g não são divisíveis por qualquer x_i. Então P_h tem no mínimo dois pontos integrais. Sejam $f = h \cdot s$ e $g = h \cdot t$ para certos s, $t \in \mathbb{F}[x_1, \ldots, x_n]$. Note que P_s e P_t tem no mínimo dois pontos integrais cada, porque f e g são ambos não divisíveis por qualquer x_i. Pelo teorema 4.2.1, $P_f = P_h + P_s$ e $P_g = P_h + P_t$, então P_h é um fator integral comum de P_f e P_g, contradizendo nossa afirmação de que P_f e P_g não possuem fatores integrais em comum. □

A meta principal deste capítulo é mostrar como decidir de maneira eficiente se politopos de Newton possuem fatores em comum. Gostaríamos de observar que o problema de decidir se um politopo é decomponível é *NP-completo*, veja [9]. Ou seja, encontrar fatores de politopos é um problema díficil e cada vez pior a medida que a dimensão do politopo aumenta. Deste modo, podemos dizer que não é claro que o critério dado no Corolário 4.2.1 possa ser transformado em um algoritmo eficiente.

Agora, iremos fazer algumas observações sobre as vantagens de trabalharmos com politopos de Newton.

Suponha que tenhamos dois polinômios multivariados f e g cujos politopos de Newton associados P_f e P_g, não tenham fatores integrais em comum. Então pelo nosso critério de coprimalidade temos que $\text{mdc}(f, g) = 1$. As próximas observações seguem do fato de que um politopo é sempre a envoltória convexa de seus vértices.

- Os polinômios permanecerão coprimos sobre qualquer corpo maior desde que os termos cujos expoentes correspondem aos vértices do politopo de Newton não se anulem, isto é, desde que os politopos permaneçam os mesmos.

- Podemos mudar arbitrariamente os coeficientes dos polinômios desde que os coeficientes dos monômios que correspondem aos vértices mantenham-se não nulos, e ainda assim os polinômios permanecerão coprimos.

- Podemos acrescentar monômios a f e g e eles permanecerão coprimos desde que seus politopos de Newton permaneçam os mesmos. Isto significa que podemos adicionar monômios que correspondam a pontos interiores dos seus politopos de Newton.

4.3 Critério de Coprimalidade

Seja P um politopo em $\mathbb{R}^n$. Uma face de P é a intersecção de P com um hiperplano de suporte de P. Um vértice é uma face de dimensão 0. Uma face de dimensão 1 é um segmento de linha, dito uma aresta de P. Uma face de dimensão uma a menos que a de P é dita uma faceta de P. O próximo lema, que pode ser encontrado em [5, 7, 9, 27], descreve como decompor faces de politopos em relação à soma de Minkowski.

Lema 4.3.1. *Seja $P = Q + R$ onde Q e R são politopos em $\mathbb{R}^n$. Então cada face de P é a soma de Minkowski de únicas faces de Q and R.*

A partir deste fato podemos observar que a soma de Minkowski $u + v$ de duas faces u e v, é uma aresta se u e v são arestas paralelas ou se u ou v é uma aresta e a outra é apenas um ponto. Note que se u e v não são paralelas ou se u ou v é uma face de dimensão maior que 1, então $u + v$ tem dimensão maior que 1. Logo, acabamos de provar o seguinte:

Lema 4.3.2. *Seja $P = Q + R$ onde Q e R são politopos em $\mathbb{R}^n$. Então cada aresta de P é a soma de Minkowski de únicas arestas paralelas de Q e R ou a soma de Minkowski de uma aresta e um ponto.*

Considere $conv(v_0, v_1)$ uma aresta de um politopo P unindo os vértices v_0 e v_1. Vamos definir

$$\text{vet}(v_0, v_1) = v_1 - v_0,$$

o vetor derivado da aresta $conv(v_0, v_1)$. Definimos $\text{vet}(P)$ o conjunto composto pelos vetores derivados de todas as arestas de P. Dizemos que um vetor v em $\text{vet}(P)$ é múltiplo do vetor u se existe um número real λ tal que $v = \lambda u$. Dados quaisquer dois politopos P e Q escreveremos

$$\text{vet}(Q) \hookrightarrow \text{vet}(P)$$

se cada vetor de $\text{vet}(Q)$ é múltiplo de um vetor de $\text{vet}(P)$. E dizemos que

$$\text{vet}(Q) \not\hookrightarrow \text{vet}(P)$$

se $\text{vet}(Q)$ e $\text{vet}(P)$ não possuem vetores múltiplos.

Proposição 4.3.1. *Sejam P e Q politopos em $\mathbb{R}^n$. Se Q é um fator de P então*

$$vet(Q) \hookrightarrow vet(P).$$

Dem.: Já observamos que se $P = Q + R$, então cada aresta de P é a soma de Minkowski de únicas arestas de Q e R ou a soma de Minkowski de uma aresta e um ponto, com todas as arestas paralelas. Então, cada aresta $conv(q_1, q_2)$ de Q unindo os vértices q_1 e q_2 de Q é paralela a alguma aresta $conv(p_1, p_2)$ de P. Logo, $\text{vet}(q_1, q_2) = \lambda \cdot \text{vet}(p_1, p_2)$ para algum $\lambda \in \mathbb{R}$. □

Esta proposição leva-nos ao seguinte critério para decidir quando dois politopos não possuem fatores integrais em comum.

Corolário 4.3.1. *Sejam P e Q politopos em $\mathbb{R}^n$. Se*

$$vet(Q) \not\hookrightarrow vet(P),$$

então P e Q não possuem fatores integrais em comum.

Dem.: Suponha, por contradição, que R é um fator comum de P e Q. Seja $r \in \text{vet}(R)$. Pela proposição 4.3.1, exite $u \in \text{vet}(P)$, $v \in \text{vet}(Q)$, $\lambda_1, \lambda_2 \in \mathbb{R}$ tal que $r = \lambda_1 u = \lambda_2 v$, mostrando que u e v são múltiplos, contradizendo a hipótese. □

Aplicando estes resultados temos o seguinte teorema.

Teorema 4.3.1. *Considere* $f, g \in \mathbb{F}[x_1, \ldots, x_n]$. *Se*

$$vet(P_f) \not\leftrightarrow vet(P_g)$$

então

$$mdc(f, g) = 1.$$

Dem.: Pelo Corolário 4.3.1, P_f e P_g não têm fatores integrais em comum. Então, pelo Critério de Coprimalidade, temos que $\text{mdc}(f, g) = 1$. □

4.4 Partes da Implementação

Se usarmos o Teorema 4.3.1 para propósitos práticos, ainda precisaríamos encontrar um procedimento eficiente para calcular as arestas de um politopo.

4.4.1 Grafo Facial

O grafo facial de um politopo é um grafo direcionado acíclico onde os nodos são as faces de P e um arco conecta faces F e G se e somente se G é uma faceta de F. Em cada nodo do grafo facial temos os vértices de P que pertencem a face representada pelo nodo. Para maiores detalhes sobre o grafo facial de um politopo sugerimos [29]. Também podemos usar o programa livre polymake, disponível na internet no endereço

http://www.math.tu-berlin.de/diskregeom/polymake/,

que ajuda na pesquisa em teoria de politopos. Particularmente, polymake calcula o grafo facial de um politopo.

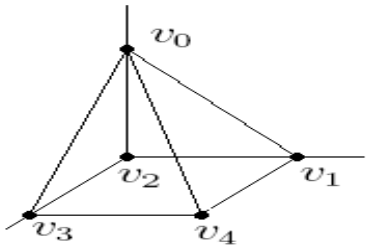

Figura 4.1: Um politopo com 5 vértices

Por exemplo, considere a pirâmide dada na figura 4.1 que pode ser representada como o politopo

$$P = conv((0,0,1),(1,0,0),(0,0,0),(0,1,0)(1,1,0)),$$

onde $v_0 = (0,0,1)$, $v_1 = (1,0,0)$, $v_2 = (0,0,0)$, $v_3 = (0,1,0)$ e $v_4 = (1,1,0)$ são os vértices de P.

Usando o programa polymake calculamos o grafo facial do politopo P, apresentado na figura 4.2.

No grafo da figura 4.2, o nodo $(0,1,4)$ representa uma faceta de P de dimensão 3. Esta faceta é a envoltória convexa dos vértices v_0, v_1 e v_4. O nodo $(1,4)$ representa uma aresta P unindo os vértices v_1 e v_4. Note que, os nodos $(0,1,4)$ e $(1,4)$ estão conectados por um arco pois a aresta $conv(v_1, v_4)$ é uma face da faceta $conv(v_0, v_1, v_4)$.

4.4.2 Teste preliminar para o mdc de Polinômios Multivariados

Neste momento vamos transformar nosso critério em um algoritmo que funcionará como um teste preliminar de mdc. Nosso teste preliminar funciona da seguinte maneira. Dados dois polinômios multivariados, calculamos os grafos faciais dos politopos de Newton associados a f e g. Então, verificamos se os politopos possuem arestas paralelas em comum. Se eles não têm arestas paralelas em comum, então $\text{mdc}(f,g) = 1$ pelo Teorema 4.3.1.

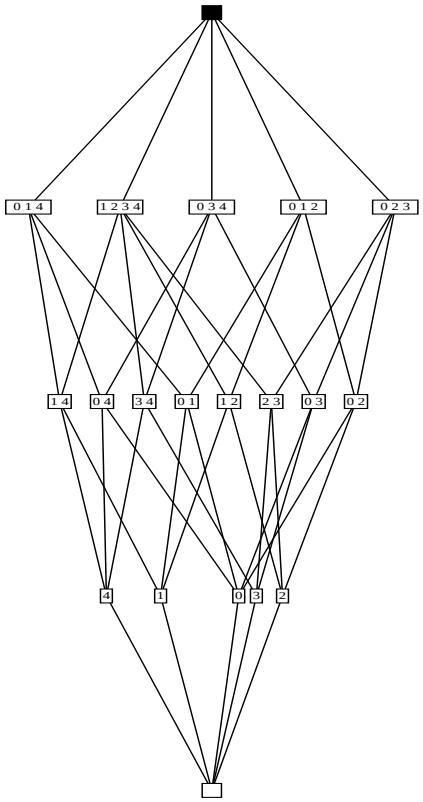

Figura 4.2: Grafo Facial

Algoritmo 4.4.1 (Teste preliminar para o mdc)**.** .
Entrada: polinômios não nulos f *e* $g \in \mathbb{F}[x_1, \ldots, x_n]$ *não divisíveis por qualquer* x_i.
Saída: $mdc(f, g) = 1$ *ou Inconclusivo.*

passo 1 *: Obter* $Supp(f)$ *e* $Supp(g)$.

passo 2 *: Calcular o grafo facial de* $P_f = conv(Supp(f))$ *e* $P_g = conv(Supp(g))$ *usando o programa polymake.*

passo 3 *:* ***Se*** $vet(P_f) \not\hookrightarrow vet(P_g)$ ***então*** $retorne(mdc(f, g) = 1)$ *senão* $retorne(Inconclusivo)$.

Tempo computacional: Sejam m e l os números de monômios dos polinômios f and g, respectivamente. O passo 1 do algoritmo pode ser executado em tempo linear dependendo de m e l.

No passo 2, Seidel [29] calcula, para um conjunto S de s pontos em $\mathbb{R}^n$, o grafo facial da envoltória convexa de S em $O(s^2 + L \log s)$, onde L é o número de faces. Se considerarmos m e l como o número de monômios de f e g como acima e, sem perda de generalidade $m \geq l$, então, no pior caso, o passo 2 pode ser executado em

$$O(m^2 + L_f \log(m)),$$

onde o número de faces de P_f, L_f, pode variar entre $\Omega(1)$ e $O(m^{\lfloor \frac{n}{2} \rfloor})$.

O tempo para verificar o paralelismo das arestas no passo 3 é dominado pelo produto do número de arestas de P_f pelo número de arestas de P_g. E, o número de arestas de cada politopo é dominado pelo número de vértices que satisfaz:

$$\sharp(\text{vet}(P_f)) \leq \binom{m}{2} = \frac{m^2 - m}{2}.$$

Então, se obtivermos sucesso em nosso teste preliminar, acabamos verificando se $\text{vet}(Q) \not\hookrightarrow \text{vet}(P)$ em

$$O((\frac{m^2 - m}{2})^2),$$

e concluímos que nosso teste preliminar trabalha em tempo polinomial sobre os números de monômios.

Observações:

1 Gostaríamos de observar que os passos 2 e 3 de nosso algoritmo dependem do número de vértices dos politopos P_f e P_g. Então, concluímos que nosso algoritmo funciona melhor com polinômios esparsos de grau grande, pois esta classe de polinômios tem politopos de Newton associados com poucos vértices e conseqüentemente um número pequeno de faces.

2 Nosso teste preliminar não depende do tamanho dos coeficientes dos polinômios envolvidos.

3 Se nosso algoritmo obtém sucesso, isto é, se o teste preliminar retornar $\text{mdc}(f, g) = 1$, então f e g permanecerão coprimos sob qualquer corpo,

desde que os coeficientes dos monômios que correspondem aos vértices de P_f e P_g não se anulem.

A partir destas observações, podemos concluir que este algoritmo é um teste preliminar útil para polinômios multivariados esparsos com grau grande e grandes coeficientes.

4.5 Experimentos Computacionais

Nesta seção apresentamos experimentos computacionais que indicam que nosso teste preliminar é realmente eficiente quando usado em polinômios esparsos com grau grande e coeficientes grandes.

A utilidade do nosso algoritmo vem do fato que o mdc de dois polinômios é quase sempre 1, ou seja, antes de aplicarmos um método para calcular efetivamente o mdc, cabe usarmos nosso teste preliminar. Por completude, iremos apresentar um argumento probabilístico bem conhecido mostrando que dados dois polinômios aleatórios, a probabilidade de eles serem coprimos é quase 1.

Vamos considerar $f,\ g \in \mathbb{F}[x_1, \ldots, x_n]$ com $\text{grau}_{x_i}(f),\ \text{grau}_{x_i}(g) \leq d$ para $1 \leq i \leq n$, então um resultado bem conhecido diz que o $\text{mdc}(f, g)$ tem grau positivo se e somente se $res(f, g) = 0$, onde $res(f, g)$ é o resultante de f e g. Temos que

$$\text{grau}(res(f, g)) \leq 2(n-1)d^2.$$

Seja $S \subset \mathbb{F}$ um conjunto finito com $s = \sharp(S)$, então

$$prob(res(f, g)(a) = 0 : a \in S^n) \leq \frac{2(n-1)d^2}{s}.$$

E, já que $\text{mdc}(f, g) \neq 1$ se e somente se $res(f, g) = 0$, então

$$prob(\text{mdc}(f, g) \neq 1) \leq \frac{2(n-1)d^2}{s}.$$

Considerando s arbitrariamente grande, a conclusão segue.

Nas próximas tabelas usamos polinômios multivariados com coeficientes em $\mathbb{Z}$, e elas indicam que nosso teste preliminar é realmente eficiente quando usado em polinômios multivariados esparsos com grau grande. Nas tabelas a seguir trabalhamos apenas com polinômios esparsos. Denotamos Maple para o tempo que o programa matemático Maple gasta para determinar o mdc e teste preliminar para o tempo gasto pelo nosso teste preliminar.

Para cada linha das próximas tabelas fizemos 10 iterações e então calculamos a média. Nas tabelas o tempo é medido em segundos. Em cada iteração os polinômios f e g foram escolhidos arbitrariamente com apenas o grau fixado. Nas tabelas a seguir os polinômios inteiros usados têm coeficientes cujo módulo é menor que 100.

Polinômios esparsos com 2 variáveis.

$(\text{grau}(f), \text{grau}(g))$	Maple	Teste preliminar	$\text{mdc}(f, g)$
(1000,1000)	0.0104	0.0075	1
(10000,10000)	0.66	0.0082	1
(100000,100000)	62.91	0.0113	1

Polinômios esparsos com 3 variáveis.

$(\text{grau}(f), \text{grau}(g))$	Maple	Teste preliminar	$\text{mdc}(f, g)$
(1000,1000)	0.0083	0.0122	1
(10000,10000)	0.5406	0.011	1
(100000,100000)	48.0185	0.0084	1

Polinômios esparsos com 6 variáveis.

$(\text{grau}(f), \text{grau}(g))$	Maple	Teste preliminar	$\text{mdc}(f, g)$
(1000,1000)	0.0057	0.0188	1
(10000,10000)	0.211	0.0204	1
(100000,100000)	23.9003	0.0192	1

Polinômios esparsos com 10 variáveis.

$(\text{grau}(f), \text{grau}(g))$	Maple	Teste preliminar	$\text{mdc}(f, g)$
(1000,1000)	0.0039	0.0251	1
(10000,10000)	0.1464	0.025	1
(100000,100000)	12.5432	0.0249	1

Note que os polinômios detectados coprimos através do nosso método permanecerão coprimos mesmo que mudemos arbitrariamente seus coeficientes, desde que os coeficientes que correspondem aos vértices dos politopos permaneçam não nulos.

Nas próximas tabelas usamos nosso teste preliminar para detectar coprimalidade entre polinômios multivariados sobre $\mathbb{F}_p$ com $p = 100000007$.

Polinômios esparsos com 8 variáveis.

$(\text{grau}(f), \text{grau}(g))$	Maple	Teste preliminar	$\text{mdc}(f, g)$
(10000,10000)	3.889	0.016	1
(20000,20000)	5.117	0.020	1
(30000,30000)	7.437	0.036	1

Polinômios esparsos com 12 variáveis.

$(\text{grau}(f), \text{grau}(g))$	Maple	Teste preliminar	$\text{mdc}(f, g)$
(10000,10000)	2.460	0.023	1
(20000,20000)	4.886	0.024	1
(30000,30000)	11.946	0.020	1

Observação 1: quando detectamos coprimalidade entre polinômios usando nosso método, então os polinômios permanecerão coprimos sobre qualquer corpo maior.

Observação 2: para esses tipos de polinômios nosso teste preliminar usa em torno de 5MB de memória enquanto o Maple usa em torno de 2000MB de memória.

Nossos experimentos foram feitos dentro do Maple. Dados f e g, primeiro calculamos o suporte (coletamos os expoentes vetoriais), então chamamos polymake (fora do Maple) para calcular o grafo facial, coletamos as arestas e verificamos se existem múltiplas entre elas. Então comparamos o tempo usado pelo Maple para este procedimento todo e o tempo usado pelo Maple para calcular o mdc de f e g.

Gostaríamos de observar que durante nossas extensivas simulações nunca encontramos dois polinômios aleatórios com $\text{mdc} \neq 1$.

4.6 Calculando o mdc de Polinômios Bivariados via Polígonos de Newton

Nesta seção descrevemos como encontrar todos fatores integrais em comum entre dois polígonos em $\mathbb{R}^2$ e então usamos estes fatores para calcular o mdc entre dois polinômios bivariados.

Dado um polígono P em $\mathbb{R}^2$, consideramos $v_0, v_1, \ldots, v_{m-1}$ os vértices de P ordenados ciclicamente na direção horária. Então, definimos o conjunto de arestas de P como $\text{vet}(P) = \{E_1, \ldots, E_n\}$, com $E_i = v_i - v_{i-1}{=}(a_i, b_i)$ para $1 \leq i \leq m$, onde a_i, $b_i \in \mathbb{Z}$ e $v_m = v_0$. Um vetor $v = (a, b) \in \mathbb{Z}^2$ é dito um vetor primitivo se $\text{mdc}(a, b) = 1$. Seja $n_i = \text{mdc}(a_i, b_i)$, então defina $e_i = (\frac{a_i}{n_i}, \frac{b_i}{n_i})$. Então $E_i = n_i e_i$ onde e_i é um vetor primitivo para $1 \leq i \leq m$.

A seqüência de vetores $\{n_i e_i\}_{1 \leq i \leq m}$, a qual chamaremos de *seqüência de arestas*, identifica unicamente o polígono P sob translação determinada por v_0. Defina $\text{vetprim}(P) = \{e_1, \ldots, e_m\}$. Como a fronteira do polígono é um caminho fechado, temos que $\sum_{i=1}^{m} n_i e_i = (0, 0)$.

O próximo lema mostra como encontrar polígonos integrais em comum entre dois polígonos quaisquer.

Lema 4.6.1. *Sejam P e Q dois polígonos integrais com seqüências de arestas $\{n_i e_i\}_{1 \leq i \leq p}$ e $\{n_i' e_i'\}_{1 \leq i \leq q}$, respectivamente. Então, P e Q tem um fator integral em comum R se e somente se a seqüência de arestas de R é da forma $\{n_k'' e_k''\}_{1 \leq k \leq r}$ com $e_k'' \in vetprim(P) \cap vetprim(Q)$, onde $e_k'' = e_i = e_j'$ com $e_i \in vetprim(P)$ e $e_j' \in vetprim(Q)$, e $0 \leq n_k'' \leq min\{n_i, n_j'\}$ para $1 \leq k \leq r$.*

Dem.: Seja R um fator integral em comum entre P e Q. Consideramos que a seqüência de arestas de R é $\{E_k''\}_{1 \leq k \leq r}$. Sabemos que cada E_k'' é fator de alguma aresta de P e Q. Então E_k'' é fator de $n_i e_i$ para algum $1 \leq i \leq p$ e $n_j' e_j'$ para algum $1 \leq j \leq q$, com $e_k'' = e_i = e_j'$. Então $E_k'' = n_k'' e_k''$ com $0 \leq n_k'' \leq min\{n_i, n_j'\}$ para $1 \leq k \leq r$. Agora, se a seqüência de arestas de R é da forma $\{n_k'' e_k''\}_{1 \leq k \leq r}$ então

cada aresta $n_k^{''}e_k^{''}$ é fator de alguma aresta de P e Q. Então, R é um fator comum de P e Q. □

Note que, de acordo com o Lema 4.3.1, se dois polígonos P e Q possuem arestas tais que $\text{vetprim}(P) \cap \text{vetprim}(Q) = \emptyset$ então estes polígonos não possuem fatores integrais em comum.

Sejam f e $g \in \mathbb{F}[x, y]$ e sejam $\{n_i e_i\}_{1 \leq i \leq p}$ e $\{n_i^{'} e_i^{'}\}_{1 \leq i \leq q}$ as seqüências de arestas dos polígonos P_f e P_g, respectivamente. Observe que $vetprim(P_f) = \{e_1, \ldots, e_p\}$ e $vetprim(P_g) = \{e_1^{'}, \ldots, e_q^{'}\}$. Pelo Teorema 4.3.1, se $vetprim(P_f) \cap vetprim(P_g) = \emptyset$ então $\text{mdc}(f, g) = 1$.

Portanto, neste momento estamos interessados no caso em que

$$vetprim(P_f) \cap vetprim(P_g) \neq \emptyset,$$

e possivelmente $\text{mdc}(f, g) \neq 1$. Particularmente, queremos calcular o $\text{mdc}(f, g)$ a partir dos fatores em comum dos polígonos P_f e P_g.

Defina o conjunto $vetprim(P_h) = vetprim(P_f) \cap vetprim(P_g) = \{e_1^{''}, \ldots, e_r^{''}\}$.

Agora, defina o conjunto P_h com seqüência de arestas $\{n_1^{''}e_1^{''}, \ldots, n_r^{''}e_r^{''}\}$ com $n_i^{''} = min\{n_i, n_i^{'}\}$. Note que cada $n_i^{''}e_i^{''}$ para $1 \leq i \leq r$ é fator de alguma aresta de P_f e P_g.

Primeiro suponha que a seqüência de arestas $\{n_i^{''}e_i^{''}\}_{1 \leq i \leq r}$ tem a seguinte propriedade

$$\sum_{i=1}^{r} n_i^{''}e_i^{''} = (0, 0).$$

Então P_h define um polígono que é fator de P_f e P_g. Vamos associar ao polígono P_h o polinômio genérico $h = \sum h_{ij}x^i y^j$ onde cada (i, j) corresponde a um ponto integral em P_h. Um modo eficiente de encontrar o polinômio h é apresentado por Gao e seus colaboradores em [25] e pode ser visto como um método derivado do levantamento de Hensel padrão. Ao final, testamos se h divide ou não f e g.

Proposição 4.6.1. *Seja $h \in \mathbb{F}[x,y]$ o polinômio genérico associado ao polígono P_h e suponha que h divide f e g. Então*

$$mdc(f,g) = h.$$

Dem.: Suponha que $\text{mdc}(f,g) = h^{'}$ com $h^{'} \neq h$. Então, pelo teorema 4.2.1, $P_{h^{'}}$ é um fator integral de P_f e P_g. Assim $P_{h^{'}} = \{k_i e_i, \ldots, k_r e_r\}$ com $\sum_{i=1}^{r} k_i e_i = (0,0)$ e $0 \leq k_i \leq n_i^{''}$. Note que o lema 4.6.1 implica que $P_{h^{'}}$ é fator de P_h. Mas, h divide f e g. Então h divide $h^{'} = \text{mdc}(f,g)$. E, além disso, P_h é um fator de $P_{h^{'}}$, contradição. Deste modo, $h^{'} = h$ e $\text{mdc}(f,g) = h$. □

Agora, vamos estudar o seguinte caso $\sum_{i=1}^{r} n_i^{''} e_i^{''} \neq (0,0)$. A idéia é procurar uma seqüência de arestas $\{k_{ji} e_i^{''}\}_{1 \leq i \leq r}$ tal que $0 \leq k_{ji} \leq n_i^{''}$ e $\sum_{i=1}^{r} k_{ji} e_i^{''} = (0,0)$. Gostaríamos de observar que este é um problema *NP-completo*, e para detalhes sobre um algoritmo para resolvê-lo, citamos [9].

Note que, se não existe uma seqüência de arestas $\{k_{ji} e_i\}_{1 \leq i \leq r}$ com $0 \leq k_{ji} \leq n_i$ e $\sum_{i=1}^{r} k_{ji} e_i = (0,0)$, então $\text{mdc}(f,g) = 1$.

Considere $P_1, \ldots, P_l$ os polígonos cujas seqüências de arestas têm a propriedade $\{k_{ji} e_i\}_{1 \leq i \leq r}$ com $0 \leq k_{ji} \leq n_i$ e $\sum_{i=1}^{r} k_{ji} e_i = (0,0)$ para $1 \leq j \leq l$.

Então, cada P_j leva-nos a um polinômio genérico h_j, o qual é um candidato a fator de f e g.

Proposição 4.6.2. *Seja $M = mdc(f,g)$. Então existe $i \in \{1, \ldots, l\}$ tal que*

$$h_i = M.$$

Dem.: Seja $M = \text{mdc}(f,g)$. Então P_M é fator de P_f e P_g, e a seqüência de arestas que representa P_M tem a forma $\{k_i e_i\}_{1 \leq i \leq r}$ com $0 \leq k_i \leq n_i$ e $\sum_{i=1}^{r} k_i e_i = (0,0)$. Então $P_M = P_j$ para algum $j \in \{1, \ldots, l\}$. Assim $M = h_j$. □

Terminamos esta seção apresentando o algoritmo para calcular o mdc entre dois polinômios bivariados.

Algoritmo 4.6.1 (Construção do mdc entre polinômios bivariados)**.** .
Entrada: f *e* $g \in \mathbb{F}[x, y]$.
Saída: $mdc(f, g)$.

passo 1 : Calcular os fatores integrais em comum P_h *entre* P_f *e* P_g *usando [9].*

passo 2 : ***Se*** P_f *e* P_g *não tem fatores integrais em comum,* ***então*** *retorne*$(mdc(f, g) = 1)$.

passo 3 : ***Para*** *cada fator integral em comum* P_h ***faça****:*

passo 3.1 : Usando [25], verifique se P_h *leva-nos a um divisor* h *de* f *e* g.

passo 4 : ***Se*** *nenhum fator integral leva-nos a um divisor de* f *e* g, ***então*** *retorne*$(mdc(f, g) = 1)$.

passo 5 : Entre todos divisores de f *e* g*: retorne*$(mdc(f, g) =$*divisor com maior grau em* $x)$.

4.6.1 Um Exemplo

Agora, iremos apresentar um exemplo explicando como encontrar o mdc de dois polinômios bivariados usando polígonos de Newton.

Sejam $f = 1 + 2x^2 + 2x^2y^2 + y^2 + x^4 + x^4y^2 + xy^2 + x^3y^2 + x^3y^4 + xy^4$ e $g = x + 2x^2y + xy^2 + y + x^3 + x^4y + x^3y^2 + x^2y^2 + x^3y^3 + x^2y^4 + xy^3 \in \mathbb{R}$, cujos polígonos de Newton estão representados na figura 4.3.

A seqüência de arestas de P_f é dado por $\{n_ie_i\}_{1\leq i\leq 6}$ onde $n_1 = 4$, $n_2 = 2$, $n_3 = 1$, $n_4 = 2$, $n_5 = 1$, $n_6 = 2$ e $e_1 = (1, 0)$, $e_2 = (0, 1)$, $e_3 = (-1, 2)$, $e_4 = (-1, 0)$, $e_5 = (-1, -2)$, $e_6 = (0, -1)$. E a seqüência de arestas de P_g é dado por $\{n_i^{'}e_i^{'}\}_{1\leq i\leq 7}$ onde $n_1^{'} = 2$, $n_2^{'} = 1$, $n_3^{'} = 1$, $n_4^{'} = 1$, $n_5^{'} = 1$, $n_6^{'} = 1$, $n_7^{'} = 1$ e $e_1^{'} = (1, 0)$, $e_2^{'} = (1, 1)$, $e_3^{'} = (-1, 2)$, $e_4^{'} = (-1, 1)$, $e_5^{'} = (-1, -1)$, $e_6^{'} = (-1, -2)$, $e_7^{'} = (1, -1)$.

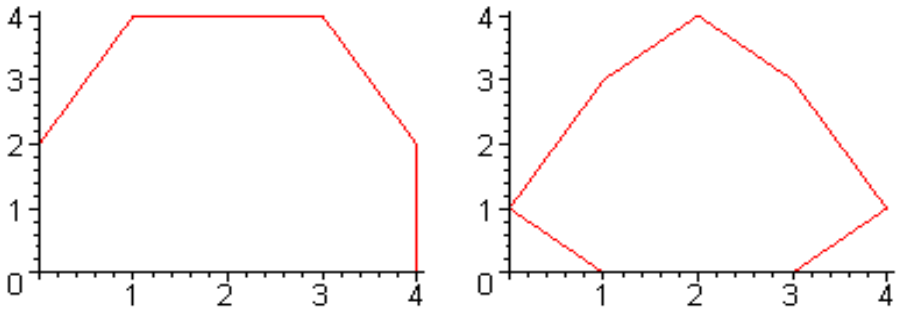

Figura 4.3: Polígonos de Newton associados aos polinômios bivariados f e g

Temos que

$$vetprim(P_f) = \{(1,0),(0,1),(-1,2),(-1,0),(-1,-2),(0,-1)\}$$

e

$$vetprim(P_g) = \{(1,0),(1,1),(-1,2),(-1,1),(-1,-1),(-1,-2),(1,-1)\}.$$

Então

$$vetprim(P_h) = vetprim(P_f) \cap vetprim(P_g) = \{(1,0),(-1,2),(-1,-2)\}$$

com $n_1'' = \text{mínimo}\{4,2\} = 2$, $n_2'' = \text{mínimo}\{1,1\} = 1$ e $n_3'' = \text{mínimo}\{1,1\} = 1$.

Note que

$$\sum_{i=1}^{3} n_i'' e_i'' = 2(1,0) + 1(-1,2) + 1(-1,-2) = (0,0).$$

Então $h = h_{00} + h_{10}x + h_{20}x^2 + h_{11}xy + h_{12}xy^2$ é o polinômio genérico associado ao polígono P_h como ilustrado na figura 4.4.

Após aplicar o levantamento de Hensel modificado apresentado em [25], encontramos $h = 1 + x^2 + xy^2$. Que divide f e g. Assim, pela proposição 4.6.1 $h = \text{mdc}(f,g)$.

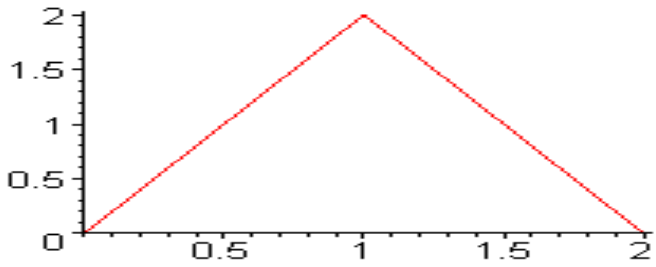

Figura 4.4: Commun summand of P_f and P_g.

4.7 Conclusão

Neste capítulo apresentamos um algoritmo eficiente, baseado em politopos de Newton, que pode determinar coprimalidade entre dois polinômios multivariados. Sua utilidade é funcionar como um teste preliminar antes de aplicarmos um método para calcular o mdc efetivamente, já que este mostrou-se eficaz quando usado com polinômios multivariados esparsos com grau grande. Também mostramos como calcular o mdc de polinômios bivariados usando polígonos de Newton.

5 CONCLUSÃO

Nesta tese de doutorado trabalhamos com temas relacionados a polinômios multivariados. Terminamos escrevendo um survey sobre o Teorema da irredutibilidade de Hilbert, desenvolvendo um novo algoritmo para fatoração polinomial multivariada e um novo método para determinar coprimalidade entre polinômios multivariados.

No Capítulo 2 fizemos uma revisão bibliográfica sobre o Teorema da irredutibilidade de Hilbert. Abordamos em detalhes as demonstrações da versão clássica deste teorema feita pelo próprio Hilbert [13] em 1892 e das versões efetivas feitas por Erich Kaltofen [14] e Shuhong Gao [8].

No Capítulo 3 desenvolvemos um novo algoritmo para fatoração polinomial multivariada baseado em novas reduções, as quais têm como principal característica manter a esparsidade do polinômio. Nosso método mostrou-se eficiente tanto para fatorar polinômios multivariados que têm apenas fatores esparsos, quanto para extrair fatores esparsos de polinômios que possuem fatores esparsos e densos. Os resultados obtidos neste capítulo foram submetidos para publicação, conforme [1].

No capítulo 4 estudamos critérios geométricos de politopos para determinar coprimalidade entre polinômios multivariados. Nossa principal contribuição foi o desenvolvimento de um algoritmo que trabalha em tempo polinomial (sobre o número de monômios) para detectar coprimalidade entre polinômios multivariados usando politopos de Newton. Também mostramos como construir o máximo divisor comum (mdc) entre dois polinômios bivariados usando seus polígonos de Newton associados. Os resultados obtidos neste capítulo também foram submetidos à publicação, conforme [2].

Como trabalho a ser feito a partir desta tese de doutorado, acreditamos que possamos desenvolver argumentos probabilísticos garantindo que as nossas reduções apresentadas no capítulo 3 manterão a irredutibilidade dos polinômios

com boa probabilidade. Também iremos desenvolver um algoritmo para calcular o mdc entre polinômios multivariados usando um método similar ao desenvolvido no capítulo 3.

BIBLIOGRAFIA

[1] Allem, L. E., Gao, S., and Trevisan, V. Factoring multivariate integral polynomials using discrete logarithm. *preprint*.

[2] Allem, L. E., and Trevisan, V. Gcd of multivariate polynomials via newton polytopes. *Submetido*, 1–16.

[3] Bajaj, C., Canny, J., Garrity, T., and Warren, J. Factoring rational polynomials over complex numbers. *SIAM J. Comput. 22* (1993), 318–331.

[4] Bernardin, L., and Monagan, M. B. Efficient multivariate factorization over finite fields. *In Proceedings of AAECC '97, Lecture Notes in Computer Science, Springer-Verlag, 1255* (1997), 15–28.

[5] Ewald, G. *Combinatorial Convexity and Algebraic Geometry, GTM 168.* Springer, 1996.

[6] Fond, A. O. G. *Transcedental and Algebraic Numbers*, vol. 2, Seminumerical Algorithms. Dover, New York, 1960.

[7] Gao, S. Absolute irreducibility of polynomials via newton polytopes. *J. of Algebra 237* (2001), 501–520.

[8] Gao, S. Factoring mutivariate polynomials via partial differential equations. *Mathematics of Computation 72* (2003), 801–822.

[9] Gao, S., and Lauder, A. Decomposition of polytopes and polynomials. *Discrete and Computational Geometry 26* (2001), 89–104.

[10] Gao, S., and Lauder, A. G. Hensel lifting and bivariate polynomial factorisation over finite fields. *Mathematics of Computation 71* (2002), 1663–1676.

[11] Garner, H. L. The residue number system. *IRE Trans. Eletronic Computers 8* (1959), 140–147.

[12] Gawrilow, E., and Joswig, M. Geometric reasoning with polymake. *url: http://arxiv.org/math.CO/0507273* (2005).

[13] Hilbert, D. Über die irreduzibilität ganzer rationaler funktionen mit ganzzahligen koeffizienten. *Journal reine angew Mathematik 110* (1892), 104–129.

[14] Kaltofen, E. Effective hilbert irreducibility. *Information and Control 66* (1985), 123–137.

[15] Kaltofen, E. Polynomial-time reductions from multivariate to bi- and univariate integral polynomial factorization. *SIAM Journal on Computing 14* (1985), 469–489.

[16] Kaltofen, E. Effective noether irreducibility forms and aplications. *Journal of Computer and System Sciences 50* (1995), 274–295.

[17] Knuth, D. E. *The art of computer programing*, vol. 2, Seminumerical Algorithms. Addison-Wesley, 1973.

[18] Lang, S. *Diophantine Equations*. Interscience Tracts in Pure and Applied Mathematics, 1962.

[19] Lecerf, G. Sharp precision in hensel lifting for bivariate polynomial factorization. *Math. Comp. 75* (2006), 921–933.

[20] Lecerf, G. Improved dense multivariate polynomial factorization algorithms. *Journal of Symbolic Computation 42* (2007), 477–494.

[21] Lecerf, G. New recombination algorithm for bivariate polynomial factorization based on hensel lifting. *Appl. Algebra Eng., Commun. Comput. 21* (2010), 151–176.

[22] Ostrowski, A. M. Über die bedeutung der theorie der konvexen polyeder für die formale algebra. *Jahresberichte Deutsche Math. Verein 30* (1921), 98–99.

[23] Pohlig, S., and Hellman, M. An improved algorithm for computing logarithms over gf(p) and its cryptographic significance. *IEEE Trans. Info. Th., 24* (1978), 106–110.

[24] Prasolov, V. V. *Polynomials*. Springer, 2001.

[25] Salem, F. A., Gao, S., and Lauder, A. G. Factoring polynomials via polytopes. *Proceeding of ISSAC 2004* (2004), 4–11.

[26] Salem, F. A., Gao, S., and Lauder, A. G. Factoring polynomials via polytopes: Extended version. *Report PRD-RR-04-07 Oxford University Computing Laboratory* (2004).

[27] Schneider, R. *Convex bodies: the Brunn-Minkowski theory - Encycopledia of Mathematics and its Applications*. Cambridge, 1993.

[28] Schwartz, J. T. Fast probabilistic algorithms for verification of polynomial identities. *J. ACM 27* (1980), 710–717.

[29] Seidel, R. Constructing higher-dimensional convex hulls at logarithmic cost per face. *Proceedings of the eighteenth annual ACM symposium on Theory of computing* (1986), 404–413.

[30] Siegel, C. L. *Topics in complex funtion theory*, vol. 1, Elliptic Functions and Uniformization Theory. John Wiley and Sons, 1969.

[31] van Hoeij, M. Factoring polynomials and the knapsack problem. *Journal of Number Theory 95* (2002), 167–189.

[32] von zur Gathen, J. Irreducibility of multivariate polynomials. *J. of Comput. System Sci 31* (1985), 225–264.

[33] VON ZUR GATHEN, J., AND GERHARD, J. *Modern Computer Algebra*. Cambridge University Press, 1999.

[34] VON ZUR GATHEN, J., AND KALTOFEN, E. Factoring sparse multivariate polynomials. *J. of Comput. System Sci 31* (1985), 265–287.

[35] WALKER, R. J. *Algebraic Curves*. Princeton University Press, 1950.

[36] WAN, D. Factoring polynomials over large finite fields. *Math. Comp., 54* (1990), 755–770.

[37] WANG, P. S. An improved multivariate polynomial factoring algorithm. *Math. Comp. 32(144)* (1978), 1215–1231.

[38] WANG, P. S., AND ROTHSCHILD, L. P. Factoring multivariate polynomials over the integers. *Mathematics of Computation 29* (1975), 935–950.

[39] ZIPPEL, R. Probabilistic algorithms for sparse polynomials. *Symbolic and algebraic computation(EUROSAM '79, Internat. Sympos., Marseille, 1979), 72* (1979), 216–226.

ÍNDICE