Gröbner Bases and Their Complexity

Master's thesis (online version)

in the field of Computer Science

submitted by

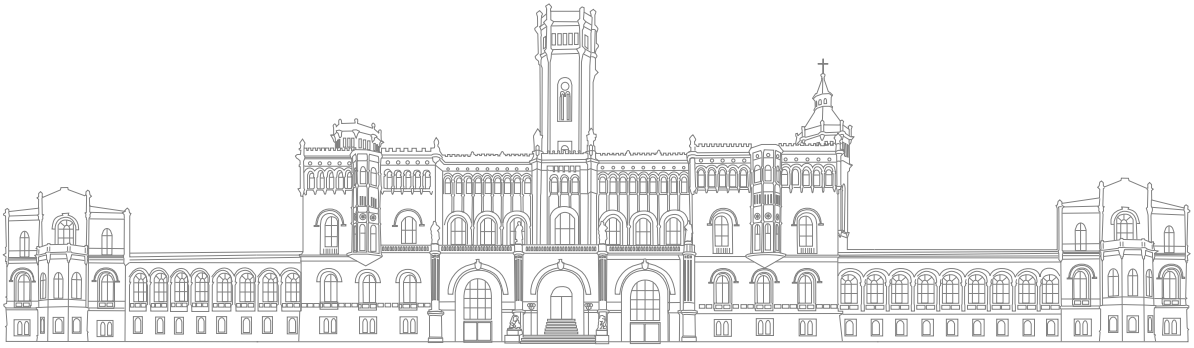
Leo Kayser

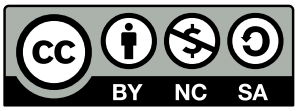
October 19, 2022

First examiner:	Prof. Dr. Heribert Vollmer
Second examiner:	PD Dr. Arne Meier
Advisor:	Sabrina Gaube
Matriculation no.:	10006908

Institute of Theoretical Computer Science

Gottfried Wilhelm Leibniz Universität Hannover





Abstract

Polynomial equations and their solutions play a ubiquitous role both for theoretical and practical applications. A closely related problem is that of ideal membership, i. e. deciding whether a polynomial is a linear combination of other polynomials. Gröbner bases provide a well-developed machinery to deal with these problems symbolically as a part of computer algebra systems. In this thesis we give an introduction to the theory of Gröbner bases with a view towards computational complexity. In particular, we discuss algorithmic aspects and upper bounds depending on the classes of polynomials considered. Furthermore, lower bounds on the size and computational complexity of Gröbner bases and related problems are presented. While often useful in practice, the worst-case complexity of Gröbner bases is located in EXPSPACE.

Contents

Introduction

Whether a particular Gröbner basis computation is feasible is usually hard to predict in advance, so Gröbner basis computations are still an adventurous business.

Gregor Kemper [24, p. 127]

Many real-world problems can be modeled using systems of polynomial equations. While numerical algorithms can approximate real solutions with floating point arithmetic, some applications such as cryptography require exact manipulation or benefit from closed-form solutions. This motivates the study of symbolic algorithms to examine solvability and other properties of systems of polynomial equations. An important question which arises in this context is that of *ideal membership*: Given polynomials $f_1, \ldots, f_s \in \mathbb{C}[X_1, \ldots, X_n]$, decide whether a given polynomial f is a linear combination of the f_i:

$$f = h_1 f_1 + \cdots + h_s f_s, \qquad h_i \in \mathbb{C}[X_1, \ldots, X_n].$$

The set of such f is called the ideal I generated by the f_i. Hilbert's Nullstellensatz asserts that the system of equations $f_1(x) = \cdots = f(x) = 0$ has a solution $x \in \mathbb{C}^n$ if and only if $1 \notin I$, thus the ideal membership problem "contains" consistency questions about polynomial equations.

In the case of polynomials in one variable $f, g \in \mathbb{C}[X_1]$, the familiar division algorithm can be used to determine a quotient q and remainder r such that

$$f = q \cdot g + r, \qquad \deg r < \deg g.$$

In particular $r = 0$ if and only if f is a multiple of g, so this solves the ideal membership problem. When generalizing this algorithm to several multivariate polynomials, problems of non-uniqueness of the remainder arise. This leads to the notion of a Gröbner basis, a special set of ideal generators which can be characterized as "behaving well" with respect to the division algorithm. A lot of problems about polynomials and ideals can be solved by first calculating a Gröbner basis, for example the aforementioned ideal membership problem.

In this thesis we will explore the computational complexity of ideal membership, Gröbner bases and related problems. The three chapters each deal with mathematical foundations, algorithmic upper bounds and lower bounds respectively. The first chapter introduces various notions from commutative algebra such as various classes of polynomials, monomial orders

and the division algorithm in its general form. Gröbner bases and normal forms are defined, various characterizations are presented and questions about uniqueness will be answered.

Chapter two is divided into two parts: The first part deals with Buchberger's algorithm, the first and most influential algorithm for Gröbner basis computation. The rest of the chapter is dedicated to upper bounds on the complexity of the ideal membership problem and Gröbner bases, culminating in a rather involved EXPSPACE-algorithm for both problems.

The final chapter provides a chain of complexity theoretic reductions from a generic EXPSPACE-complete problem to both ideal membership and Gröbner bases. The main protagonist here are (commutative) Thue systems, close relatives of formal grammars, whose structure can be cleverly embedded into binomial ideals. We also show that for some polynomials $f_1, \ldots, f_s$, a Gröbner basis of the corresponding ideal consists of double-exponentially many elements, which shows that in the worst case Gröbner bases are too large to be useful.

The aim of this thesis is to introduce the reader to the rich theory of Gröbner bases with its connections to symbolic algorithms, combinatorial problems and interesting complexity-theoretic results. Along the way we will mention many related results and give pointers to the literature.

Acknowledgment

I would like to thank my supervisor Sabrina for her encouragement and for providing useful ideas and resources. Also thanks to my fellow students Jan and Tobias for helpful discussions along the way. Last but not the least I would also like to thank my family and friends for encouraging and supporting me whenever I needed them.

1

Gröbner bases

In this first chapter we introduce the main protagonists of this thesis: Polynomial ideals and their Gröbner bases. We start by describing several interesting problems from both theoretical and applied sciences which can be reduced to solving a system of polynomial equations or more generally to the problem of ideal membership. Motivated by these problems we then generalize the familiar polynomial division algorithm to several polynomials in several variables, which naturally leads to the definition of a Gröbner basis. We discuss the topic of uniqueness of normal forms and Gröbner bases, and apply these techniques to the class of binomial ideals.

Most of the material presented here is standard and contained in any textbook on computational commutative algebra; we loosely follow von zu Gathen & Gerhard [23, Chapter 21] and occasionally other texts [19, Chapter 1], [24, Chapter 9], [28, Chapter 1&2].

1.1 Motivating examples

Many interesting problems from the sciences and engineering can be modeled by polynomial equations. We give an example from robotics [23, Example 21.1].

Example 1.1 *(The reachability problem in robotics).* We model a simple robotic arm operating in the plane $\mathbb{R}^2$. It consists of two straight line seqments $\overline{OP}$ and $\overline{PQ}$ of length 3 and 1 respectively, with the origin $O = (0,0)$ and two variable points $P = (x,y)$, $Q = (z,w)$. The length of the parts constrain the possible configurations of the joins P and Q and enforce the following two equations

$$x^2 + y^2 = 9, \qquad (x-z)^2 + (y-w)^2 = 1, \qquad x,y,z,w \in \mathbb{R}. \tag{1.1}$$

Consider the line parameterized by $L = \left\{ (t, 5-2t) \in \mathbb{R}^2 \;\middle|\; s \in \mathbb{R} \right\}$. Can the robot reach L, i. e. is there a configuration with $Q \in L$? In this case the answer is yes, for example with $P = (0,3)$, $Q = (1,3)$. On the other hand, the line $L' = \left\{ (t, 10-2t) \in \mathbb{R}^2 \;\middle|\; s \in \mathbb{R} \right\}$ seems out of reach, but it is not immediately obvious how prove the *non-solvability* of such a problem (in this case, L' has distance > 4 from the origin).

More generally, one can parameterize $L_{\lambda,\mu} = \left\{ (t, \lambda t + \mu) \in \mathbb{R}^2 \;\middle|\; s \in \mathbb{R} \right\}$ and ask: for which values of λ, μ is there a solution? Of course, more complicated robots can be modeled with multiple joints and other constraints, then the number of equations which must be satisfied

simultaneously increases. ⌟

Nonlinear algebra is the broad field of mathematics studying polynomial equations, their structure and how to solve them, using techniques from various disciplines such as algebraic geometry, computer algebra, optimization and representation theory. An overview of several interesting applications of different flavors such as biochemical reactions, computer visions or statistics is presented by Breiding et al. [7]. The solutions to the preceding problems are mostly defined over the real or complex numbers, on the other hand applications in cryptography are often interested in solutions defined over $\mathbb{Q}$ or finite fields.

Another interesting application is that of "automatic" theorem proving, adapted from [23, Example 21.2].

Example 1.2 *(Proving theorems in euclidean geometry).* Consider a triangle with vertices A, B, C in the plane, an elementary theorem states that the perpendicular bisectors of the three sides all meet in a common point, the *circumcenter*. After scaling and rotating the triangle we may assume $A = (0,0)$, $B = (1,0)$ and $C = (x,y)$ for $x, y \in \mathbb{R}$ $(y \neq 0)$. The three bisectors can be

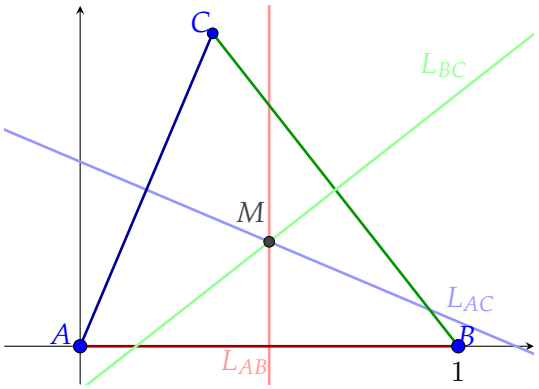


Figure 1.1: The perpendicular bisectors of a triangle intersecting in the circumcenter M.

parametrized as

$$L_{AB} = \left\{ \left(\tfrac{1}{2}, r\right) \,\middle|\, r \in \mathbb{R} \right\}, \quad L_{AC} = \left\{ \left(\tfrac{x}{2} + sy, \tfrac{y}{2} - sx\right) \,\middle|\, s \in \mathbb{R} \right\},$$
$$L_{BC} = \left\{ \left(\tfrac{x+1}{2} + ty, \tfrac{y}{2} - t(x-1)\right) \,\middle|\, t \in \mathbb{R} \right\}$$

Let $M = (u, v)$ be a point, the condition that M lies on each of these lines can be described as an equation

$$f_{AB}(u,v) = u - \frac{1}{2} = 0 \tag{1.2}$$
$$f_{AC}(u,v) = x\left(u - \frac{x}{2}\right) - y\left(\frac{y}{2} - v\right) = 0 \tag{1.3}$$
$$f_{BC}(u,v) = (x-1)\left(u - \frac{x+1}{2}\right) - y\left(\frac{y}{2} - v\right) = 0 \tag{1.4}$$

Clearly two of the three equations can be satisfied (for fixed x, y) at the same time, as two non-parallel lines intersect. The question is whether there is a solution to $f_{AB} = f_{AC} = f_{BC} = 0$ at the same time. The answer is yes, since we have

$$f_{BC} = f_{AC} - f_{AB}$$

and hence if two of the polynomials vanish, then so does the third. Other geometric properties of the triangle can also be described using polynomial equations, and one may hope to similarly prove more involved statements. ⌟

The previous problem was solved by showing that a given polynomial is a linear combination of other polynomials. The coefficients (here $1, -1$) may even be polynomials themselves, the conclusion is still valid. This is an instance of the polynomial ideal membership, one of the central problems of this thesis.

1.2 The ideal membership problem

We now formally introduce the notion of polynomial ideals and the membership problem. Let $\mathbb{N} = \{0, 1, 2, \dots\}$ denote the natural numbers and let $\mathbb{K}$ be a field such as $\mathbb{Q}, \mathbb{C}, \mathbb{F}_p$ and $\underline{X} = \{X_1, \dots, X_n\}$.

Definition 1.3 (Polynomial). A *polynomial* over $\mathbb{K}$ is an expression

$$f = \sum_{\alpha \in \mathbb{N}^n} f_\alpha \cdot X_1^{\alpha_1} \cdots X_n^{\alpha_n}, \qquad f_\alpha \in \mathbb{K},\ f_\alpha \neq 0 \text{ only for finitely many } \alpha \in \mathbb{N}^n,$$

the ring of polynomials is $\mathbb{K}[\underline{X}] = \mathbb{K}[X_1, \dots, X_n]$. The expression X^α is a *monomial*, the set of monomials will be denoted as $\mathrm{Mon}(\underline{X})$ or Mon_n. The f_α are the coefficients of f. If $f_\alpha \neq 0$, then $f_\alpha X^\alpha$ is a *term* of f, the set $\mathrm{supp}(f) \coloneqq \left\{ X^\alpha \in \mathrm{Mon}_n \;\middle|\; f_\alpha \neq 0 \right\}$ is the *support* of f. The *(total) degree* of $f \neq 0$ is the number

$$\deg f \coloneqq \max \left\{ |\alpha| = \alpha_1 + \cdots + \alpha_n \;\middle|\; f_\alpha \neq 0 \right\} \in \mathbb{N}^{[1]}.$$

The zero polynomial has degree $-\infty$ by convention. ⌟

The previous definition makes sense not only over a field $\mathbb{K}$ but rather over any commutative ring, for example $\mathbb{Z}[\underline{X}]$ is the ring of polynomials with integer coefficients. In concrete examples we will use variables such as $\{X, Y, Z\}$ instead, for example

$$f = 4XY^3 - \frac{5}{42}Y^2 - 12XY + 14 \in \mathbb{Q}[X, Y], \qquad \mathrm{supp}(f) = \{XY^3, XY, Y^2, 1\}, \quad \deg f = 4.$$

[1] $0 \in$

If $x_1, \dots, x_n \in \mathbb{K}$ are elements of the base field, then we can evaluate a polynomial f as

$$f(x_1, \dots, x_n) = \sum_{\alpha} f_\alpha x_1^{\alpha_1} \dots x_n^{\alpha_n} \in \mathbb{K}.$$

Remark. In this way f defines a function $f : \mathbb{K}^n \to \mathbb{K}$ and this function uniquely determines f as long as $\deg f > |\mathbb{K}|$ (for example if $\mathbb{K}$ is infinite). Therefore, it is generally harmless to identify polynomials with polynomial functions, but we will treat polynomials as "abstract sums" most of the time.

Appendix A.1 defines some basic notions of ring theory; we will only need the polynomial ring. Let $F \subseteq \mathbb{K}[\underline{X}]$ be a set of polynomials; the ideal generated by F is the set of (polynomial) linear combinations of elements of F:

$$\langle F \rangle_{\mathbb{K}[\underline{X}]} := \left\{ h_1 f_1 + \dots + h_s f_s \;\middle|\; s \in \mathbb{N},\ f_i \in F,\ h_i \in \mathbb{K}[\underline{X}] \right\}.$$

If the ring $\mathbb{K}[\underline{X}]$ is understood in the context, then the subscript will be omitted. Assuming for a second that we fix a suitable representation of multivariate polynomials over $\mathbb{K}$ (we will do this in section 1.7). The ideal membership problem is the following decision problem.

Definition 1.4 (Ideal membership problem, $\mathsf{IM}_{\mathbb{K}}$).

- *Input:* $(f, f_1, \dots, f_s)$ multivariate polynomials from $\mathbb{K}[X_1, \dots, X_n]$
- *Output:* Decide whether $f \in \langle f_1, \dots, f_n \rangle$ ⌟

Example 1.5. A *monomial ideal* is an ideal $I \subseteq \mathbb{K}[\underline{X}]$ generated by a set of monomials $A \subseteq \mathrm{Mon}_n$. In this case, ideal membership is an easy task: ⌟

Lemma 1.6. *If I is an ideal generated by monomials A, then $f \in I$ if and only each monomial $X^\beta \in \mathrm{supp}(f)$ is divided by some $X^\alpha \in A$.*

Proof. Indeed, this is clearly sufficient, conversely write $f \in I$ as $f = \sum_{j=1}^{N} c_j X^{\gamma_j} X^{\alpha_j}$, $c_j \in \mathbb{K}$, $X^{\alpha_j} \in A$. If $X^{\gamma_j + \alpha_j}$ doesn't occur in f, then all such terms can be omitted from the sum, leaving only terms $X^{\gamma_j+\alpha_j}$ occurring in f. If furthermore $X^{\gamma_j+\alpha_j} = X^{\gamma_k+\alpha_k}$, then we can combine the two terms to $(c_j + c_k)X^{\gamma_j+\alpha_j}$ so that we have exactly one summand for each term in f. This presentation proves the claim. □

This yields a trivial polynomial time algorithm for deciding monomial ideal membership; just compare the exponents of each term $X^\beta \in \mathrm{supp}(f)$ to each α and check if $\alpha_i \leq \beta_i$ for $i = 1, \dots, n$. We will see that ideal membership is, in general, *much* harder, but it is interesting to restrict the polynomials to various classes of polynomials and explore their complexity. Interesting classes include

- All polynomials ($\mathsf{IM}_{\mathbb{K}}$)

- Monomials X^α (the previous example)
- Binomials $aX^\alpha + bX^\beta$ or even *pure binomials* $X^\alpha - X^\beta$
- Homogeneous polynomials $\sum_{|\alpha|=d} f_\alpha X^\alpha$, this case will be denoted by $\mathsf{IM}_{\mathrm{h},\mathbb{K}}$.

In the homogeneous case ideal membership can also be simplified: For a polynomial f denote its homogeneous components by $f^{(d)} := \sum_{|\alpha|=k} f_\alpha X^\alpha$, then we have

Lemma 1.7. *If F consists of homogeneous polynomials, then $f \in \langle F \rangle$ if and only if $f^{(k)} \in \langle F \rangle$ for $k = 0, \ldots, \deg(f)$. Furthermore, if f is homogeneous and*

$$f = h_1 f_1 + \cdots + h_s f_s, \qquad f_i \in F,\ h_i \in \mathbb{K}[\underline{X}],$$

then the h_i may be chosen to be homogeneous of degree $\deg h_i = \deg f - \deg f_i$.

The proof is similar to Lemma 1.6, see for example [19, Lemma 2.2.7]. Such degree bounds can be used to find the h_i, this will be exploited in chapter 2.

The ideal membership problem can be used to decide whether a set of polynomial equations $f_1 = \cdots = f_s = 0$ has a solution. The following theorem holds true for any algebraically closed field, for simplicity we specialize to the complex field.

Theorem 1.8 (Hilbert's Nullstellensatz). *A set of polynomials $f_1, \ldots, f_s \in \mathbb{C}[\underline{X}]$ has a simultaneous zero in $\mathbb{C}^n$ if and only if $1 \notin \langle f_1, \ldots, f_s \rangle$.*

If $1 = \sum_{i=1}^s h_i f_i$ and $x \in \mathbb{C}^n$, then $1 = \sum_{i=1}^s h_i(x) f_i(x)$, hence at least one of the f_i does not vanish on x. The proof of the other direction is much more involved, see for example [19, Theorem 3.5.2] or [24, Corollary 1.8]. This motivates the following variant of the ideal membership problem:

Definition 1.9 (Nullstellensatz, $\mathsf{HNST}_\mathbb{K}$).

- *Input:* $(f_1, \ldots, f_s)$ multivariate polynomials from $\mathbb{K}[X_1, \ldots, X_n]$
- *Output:* Decide whether $1 \notin \langle f_1, \ldots, f_n \rangle$ ⌟

Remark. One might ask why we don't define $\mathsf{HNST}_\mathbb{Q}$ to ask about solutions in $\mathbb{Q}$ but rather in $\mathbb{C}$. The reason is that this problem is computationally tractable (in the Turing model), while it is *not known* if the existence of rational solutions is even decidable! The (negative) answer to Hilbert's tenth problem shows that the set

$$\left\{ f \in \mathbb{Z}[X_1, \ldots, X_n] \;\middle|\; n \geq 1 \text{ and } f(x) = 0 \text{ for some } x \in \mathbb{Z}^n \right\}$$

is undecidable. On the other hand, solvability in the ground field is $\mathsf{NP}_\mathbb{K}$ complete in the Blum–Shub–Smale model of computation, in analogy to how SAT is NP-complete in the bit model [5, Chapter 5].

1.3 Polynomial division

We now review the familiar notion of polynomial division with remainder. Let $f, g \in \mathbb{K}[X]$ be univariate polynomials, $g \neq 0$, then there exist unique $q, r \in \mathbb{K}[X]$ satisfying

$$f = gq + r, \qquad \deg r < \deg g.$$

The polynomials g, r can be obtained algorithmically: For a polynomial of the form

$$f = a_n X^n + \cdots + a_1 X + a_0, \quad a_n \neq 0$$

let $\operatorname{LT}(f) \coloneqq a_n X^n$ be its *leading term* (this notion will be formally introduced for multivariate polynomials in a moment). The procedure is described in Algorithm 1.

Algorithm 1 Univariate polynomial division

Require: $f, g \in \mathbb{K}[X]$, $g \neq 0$
Ensure: $f = gq + r$, $\deg r < \deg g$
1: $q \leftarrow 0, r \leftarrow f$
2: **while** $\deg r \geq \deg g$ **do** ▷ **The equation $f = gq + r$ is a loop invariant.**
3: $q_0 \leftarrow \operatorname{LT}(g)/\operatorname{LT}(r)$
4: $q \leftarrow q + q_0$
5: $r \leftarrow r - g \cdot q_0$ ▷ **The degree of r strictly decreases.**
6: **end while**

We wish to generalize this procedure to multivariate polynomial division. At first glance there are two obvious obstacles:

- In order to even specify "quotient and remainder" of a division, we need to be able to compare the degree of multivariate polynomials. The total degree is not an useful measure of size here: Consider $f = XY$, $g = X + Y$, then there do not exists polynomials $q, r \in \mathbb{K}[X, Y]$ with $XY = (X + Y)q + r$ such that r has total degree < 2.
- In order to extend the univariate division algorithm we would like to talk about the *leading terms* of a polynomial f. Again, there might be several valid choices, even of the same total degree as f.

We can solve both problems at once by introducing *monomial orderings*. Recall that monomials are in natural bijection with tuples of natural numbers:

$$\mathbb{N}^n \ni \alpha = (\alpha_1, \ldots, \alpha_n) \mapsto X^\alpha = X_1^{\alpha_1} \cdots X_n^{\alpha_n} \in \operatorname{Mon}_n.$$

If we choose a "suitable" ordering on the set Mon_n, then we can talk about the largest monomial occurring in a polynomial f, and use this to define the leading term and degree $\deg f \in \mathbb{N}^n$.

Clearly such an order should be a total order (also called a linear order), and it should be compatible with the monoid structure of Mon_n. This leads to the following definition

Definition 1.10 (Monomial ordering). A *monomial ordering* $\prec$ is an order on the set $\mathbb{N}^n$ (equivalently, on the set of monomials Mon_n) with the following properties:

(i) $\prec$ is a *total order*: For all $\alpha, \beta \in \mathbb{N}^n$ we have $\alpha \preceq \beta$ or $\beta \preceq \alpha$.

(ii) For $\alpha, \beta, \gamma \in \mathbb{N}^n$ with $\alpha \prec \beta$ we also have $\alpha + \gamma \prec \beta + \gamma$.

(iii) $(0, \dots, 0) \prec \alpha$ for all $\alpha \in \mathbb{N}^n \setminus \{(0, \dots, 0)\}$. ⌟

It turns out that under assumtion (i)+(ii), condition (iii) is equivalent to $\prec$ being a well-ordering [23, Corollary 21.20]:

(iii′) $\prec$ is a *well-order*: Every non-empty set $M \subseteq \mathbb{N}^n$ has a minimal element with respect to $\prec$.

We can (and will) assume that a monomial ordering sorts the variables themselves in descending order $X_1 \succ X_2 \succ \cdots \succ X_n$.

Example 1.11. The following are examples of monomial orderings:

(i) The *lexicographic ordering* $\prec_{\text{lex}}$ is defined as

$$\alpha \prec_{\text{lex}} \beta \quad \text{if and only if} \quad \text{the first nonzero entry in } \alpha - \beta \text{ is negative.}$$

In other words, $X^\alpha \prec_{\text{lex}} X^\beta$ if the number of X_1's in X^α is less than in X^β, or they are equal and the number of X_2's in X^α is less than in X^β, or they are equal too and so on.

(ii) The *graded lexicographic ordering* $\prec_{\text{grlex}}$ is defined as

$$\alpha \prec_{\text{grlex}} \beta \quad \text{if and only if} \quad \sum_i \alpha_i < \sum_i \beta_i \text{ or } \left(\sum_i \alpha_i = \sum_i \beta_i \text{ and } \alpha \prec_{\text{lex}} \beta \right).$$

In other words, the graded lexicographic ordering first sorts by total degree, and then uses lexicographic ordering as a tie-breaker.

(iii) The *graded reverse lexicographic order* $\prec_{\text{grevlex}}$ is defined as

$$\alpha \prec_{\text{grlex}} \beta \quad \text{if and only if} \quad \sum_i \alpha_i < \sum_i \beta_i \text{ or } \left(\sum_i \alpha_i = \sum_i \beta_i \text{ and } -\beta \prec_{\text{lex}} -\alpha \right).$$

For example with $X \succ Y \succ Z$ we have

$$Y^3 \prec_{\text{lex}} XYZ \prec_{\text{lex}} X^2, \qquad X^2 \prec_{\text{grlex}} XYZ \prec_{\text{grlex}} Y^3, \qquad X^2 \prec_{\text{grevlex}} Y^3 \prec_{\text{grevlex}} XYZ. \quad ⌟$$

From now on we will consider polynomial rings equipped with a monomial ordering, denoted as $\mathbb{K}[\underline{X}]_\prec$ (omitting the subscript if no confusion may occur). We now generalize the notions of degree and leading term to such polynomial rings. For this we add the formal symbol $-\infty$ to our set of multiindices with the properties

$$-\infty \prec \alpha, \quad -\infty + \alpha = -\infty \qquad \forall \alpha \in \mathbb{N}^n.$$

Definition 1.12. Let $f = \sum_\alpha c_\alpha X^\alpha$ be a polynomial. If $f = 0$, then we define $\operatorname{mdeg}(0) := -\infty$, otherwise $f \neq 0$ and we define:

(i) The *multidegree* $\operatorname{mdeg}(f) \in \mathbb{N}^n$ of f is the largest exponent α among the terms with respect to $\prec$.

(ii) If $\alpha := \operatorname{mdeg}(f)$, then we define the *leading term* of f as $\operatorname{LT}(f) = c_\alpha X^\alpha$ with *leading coefficient* $\operatorname{LC}(f) := c_\alpha \in \mathbb{K}^\times$ and *leading monomial* $\operatorname{LM}(f) = X^\alpha \in \operatorname{Mon}_n$. ⌟

The multidegree enjoys properties similar to the usual degree: Let f, g be polynomials, then

- $\operatorname{mdeg}(f \cdot g) = \operatorname{mdeg}(f) + \operatorname{mdeg}(g)$,
- $\operatorname{mdeg}(f + g) \leq \max\{\operatorname{mdeg}(f), \operatorname{mdeg}(g)\}$, with equality if $\operatorname{mdeg}(f) \neq \operatorname{mdeg}(g)$,
- if $f, g \neq 0$, then $\operatorname{LT}(fg) = \operatorname{LT}(f) \cdot \operatorname{LT}(g)$ and similarly for LM, LC.

With this notion we can describe a division algorithm for multivariate polynomials which is very similar to algorithm 1. Our goal here is slightly different from before: In the univariate case we searched for q, r with $f = qg + r$ with "small" remainder, measured by the degree. Here we ask for the remainder to contain *no* terms divisible by the leading term of g, which is a weaker condition than having a smaller multidegree.

Algorithm 2 Multivariate polynomial division (single divisor)

Require: $f, g \in \mathbb{K}[X_1, \ldots, X_n]_\prec$, $g \neq 0$
Ensure: $f = qg + r$, $\operatorname{LM}(g) \nmid t$ for all monomials $t \in \operatorname{supp}(r)$.

```
q ← 0, r ← 0, p ← f
while p ≠ 0 do
    if LM(g) | LM(p) then
        q_0 ← LT(p)/LT(g)
        q ← q + q_0
        p ← p − g · q_0
    else
        r ← r + LT(p)
        p ← p − LT(p)
    end if
end while
```

Lemma 1.13. *Algorithm 2 terminates, more specifically* $\operatorname{mdeg}(p)$ *strictly decreases each iteration if* $p \neq 0$*. It produces* $q, r \in \mathbb{K}[\underline{X}]$ *with* $f = qg + r$ *such that the leading term of* g *divides no term in* r *and* $\operatorname{mdeg}(qg) \preceq \operatorname{mdeg}(f)$*.*

Proof. We first show that $\operatorname{mdeg}(p_{\text{new}}) \prec \operatorname{mdeg}(p)$; this is clear in the else-branch, as the leading term is removed. In the other case

$$\begin{aligned}\operatorname{mdeg}(p_{\text{new}}) &= \operatorname{mdeg}(p - g \operatorname{LT}(p)/\operatorname{LT}(g)) \\ &\preceq \max\{\operatorname{mdeg} p, \operatorname{mdeg}(g) + \operatorname{mdeg}(p) - \operatorname{mdeg}(g)\} = \operatorname{mdeg}(p).\end{aligned}$$

But $\operatorname{LT}(g \cdot q_0) = \operatorname{LT}(p)$, so we must have a strict inequality.

The assignments in line 5/6 and in line 8/9 preserve the loop invariant

$$f = p + qg + r.$$

Also, q is only incremented by monomials q_0 such that $\operatorname{mdeg}(q_0 \cdot g) = \operatorname{mdeg}(p) \preceq \operatorname{mdeg}(f)$, so $\operatorname{mdeg}(qg) \preceq \operatorname{mdeg}(f)$. Furthermore, the if-condition in line 3 ensures that only terms indivisible by $\operatorname{LT}(g)$ are added to r in line 8. When the algorithm terminates we have $p = 0$ and hence the claimed identities are ensured. □

The remainder is denoted as $r = \operatorname{rem}(f; g)$, it assigns any such f a (unique) representative of f modulo $\langle g \rangle$. This solves the divisibility problem for multivariate polynomials, as

$$g \mid f \quad \text{if and only if} \quad \operatorname{rem}(f; g) = 0.$$

Proof. If $\operatorname{rem}(f; g) = 0$, then $f = qg$. Conversely if $g \mid f$, then $g \mid f - qg = r$, i.e. $r = gh$. Assume $r \neq 0$, then this means $\operatorname{LT}(r) = \operatorname{LT}(g) \operatorname{LT}(h)$, contradicting $\operatorname{LM}(g) \nmid \operatorname{LM}(r)$. □

1.4 The normal form algorithm and Gröbner bases

The previous observation may be rephrased as a simple case of the ideal membership problem: $f \in \langle g \rangle$ if and only if $\operatorname{rem}(f; g) = 0$. In general, ideals in polynomials rings need not be principal, but require several generators. For example $I = \langle X_1, \ldots, X_n \rangle$ cannot be generated by less than n elements. In order to solve the ideal membership problem we would like to extend the division algorithm to take as input both f and the generators $g_1, \ldots, g_s$ such that

$$f \in \langle g_1, \ldots, g_s \rangle \quad \text{if and only if} \quad \operatorname{rem}(f; g_1, \ldots, g_n) = 0.$$

This will turn out to be a more involved task and leads to the notion of Gröbner bases. We first formalize the notion of a normal form.

Definition 1.14 (Normal form, NF_G). Let $S \subseteq \mathbb{K}[\underline{X}]$ and $f \in \mathbb{K}[\underline{X}]$. f is in *normal form* with respect to S if $\mathrm{LM}(g) \nmid t$ for any $t \in \mathrm{supp}(f)$ and $g \in S \setminus \{0\}$.

The set of *normal forms* of f with respect to a *finite* set $G = \{g_1, \dots, g_s\}$ is the set $\mathrm{NF}_G(f)$ of $r \in \mathbb{K}[\underline{X}]$ such that

(i) r is in normal form with respect to G;

(ii) $f = r + \sum_{i=1}^{s} q_i g_i$ for suitable $q_i \in \mathbb{K}[\underline{X}]$ with $\mathrm{mdeg}(q_i g_i) \preceq \mathrm{mdeg}(f)$. ⌟

Notice that $\mathrm{mdeg}(r) = \mathrm{mdeg}(f - \sum_{i=1}^{s} q_i g_i) \preceq \max_{\prec}\{\mathrm{mdeg}(f), \mathrm{mdeg}(q_i g_i)\} = \mathrm{mdeg}(f)$.

In analogy to multivariate polynomial division with a single divisor we want to find a normal form r together with the $q_1, \dots, q_s$ from the previous definition. This is actually not much harder to achieve than in case $s = 1$, a slight modification of the previous algorithm yields the normal form algorithm 3. The key idea is to check the condition $\mathrm{LM}(g_i) \mid \mathrm{LM}(p)$ against *all* $g_1, \dots, g_s$ and then proceed in the same way.

Algorithm 3 The normal form algorithm

Require: $f, g_1, \dots, g_s \in \mathbb{K}[X_1, \dots, X_n]_{\prec}$, $g_1, \dots, g_s \neq 0$
Ensure: $f = q_1 g_1 + \cdots + q_s g_s + r$, $\mathrm{LM}(g_i) \nmid t$ for all terms $t \in \mathrm{supp}(r)$ and all i.

```
1:  (q_1, ..., q_s) ← (0, ..., 0), r ← 0, p ← f
2:  while p ≠ 0 do
3:      if LM(g_i) | LM(p) for some i then
4:          Choose such an i ∈ {1, ..., s}          ▷ e. g. the smallest such i
5:          q_0 ← LT(p)/LT(g_i)
6:          q_i ← q_i + q_0
7:          p ← p − q_0 · g_i
8:      else
9:          r ← r + LT(p)
10:         p ← p − LT(p)
11:     end if
12: end while
```

With this in mind the following lemma is proven exactly the same way as before.

Lemma 1.15. *Algorithm 3 terminates and* $\mathrm{mdeg}(p)$ *strictly decreases each iteration if* $p \neq 0$. *It produces* $q_1, \dots, q_s, r$ *as in Definition 1.14.*

Algorithm 3 allows us to define $\mathrm{rem}(f; g_1, \dots, g_s)$ as before (if we always choose the least possible i in line 4), but unfortunately this *fails* to solve the ideal membership problem in general!

Example 1.16. Consider $f = XY^2 - X$, $g_1 = XY + 1$, $g_2 = Y^2 - 1$ and $\prec_{\mathrm{lex}}$, then $\mathrm{rem}(f; g_1, g_2) = -X - Y$, but $f = X \cdot g_2 \in \langle g_1, g_2 \rangle$. Of course, this does not happen if we chose $i = 2$ in line 4 of the algorithm, indeed $\mathrm{rem}(f; g_2, g_1) = 0$. ⌟

We would like to consider generating sets of I behaving well with this ambiguity in the division algorithm, it turns out that Gröbner bases have this property.

Definition 1.17. The *initial ideal* of $I \subseteq \mathbb{K}[\underline{X}]_\prec$ is the monomial ideal

$$\operatorname{IN}(I) := \left\langle \left\{ \operatorname{LM}(g) \,\middle|\, 0 \neq g \in I \right\} \right\rangle . \qquad \lrcorner$$

We have the following characterizations:

Theorem 1.18 (Characterizations of Gröbner bases). *Let I be an ideal and $G = \{g_1, \ldots, g_s\} \subseteq I$. The following are equivalent:*

(a) $\operatorname{IN}(I) = \langle \operatorname{LT}(G) \rangle$

(b) *For all $f \in \mathbb{K}[\underline{X}]$ there is a* unique *r with $f - r \in I$ such that no $\operatorname{LT}(g_i)$ divides any $m \in \operatorname{supp}(r)$.*

(c) *For all $f \in \mathbb{K}[\underline{X}]$ we have $f \in I$ if and only if $\operatorname{rem}(f; g_1, \ldots, g_s) = 0$.*

Moreover, if any of these properties is satisfied, then for any input f the result r of algorithm 3 is independent of the choices of i in line 5.

Proof. (a)⇒(b): Algorithm 3 guarantees the existence of a normal form $r \in \operatorname{NF}_G(f)$ which satisfies the condition by definition. Now consider two decompositions $f = h_1 + r_1 = h_2 + r_2$, $h_1, h_2 \in I$. Then $r_1 - r_2 = h_2 - h_1 \in I$, hence $\operatorname{LM}(r_1 - r_2) \in \operatorname{IN}(I) = \left\langle \operatorname{LM}(g_1), \ldots, \operatorname{LM}(g_s) \right\rangle$. Assume $r_1 \neq r_2$, then by Lemma 1.6 some monomial in $r_1 - r_2$ is divisible by some $\operatorname{LM}(g_i)$, hence such a monomial must occur in either r_1 or r_2, a contradiction.

(b)⇒(c): Let $f \in \mathbb{K}[\underline{X}]$. If $\operatorname{rem}(f; g_1, \ldots, g_s) = 0$, then $f \in \left\langle g_1, \ldots, g_s \right\rangle \subseteq I$. Conversely let $f \in I$, then the Algorithm yields a decomposition $f = q_1 g_1 + \cdots + q_s g_s + r, r = \operatorname{rem}(f; g_1, \ldots, g_s)$, such that r the desired properties from (ii). But $r' = 0$ also has these properties (recall $f - 0 \in I$), so by uniqueness we must have $r = 0$. This also shows the independence of choices in Algorithm 3 in line 5.

(c)⇒(a): In order to show that $\operatorname{IN}(I) = \left\langle \operatorname{LM}(g_1), \ldots, \operatorname{LM}(g_s) \right\rangle$ it suffices to check that $\operatorname{LM}(f) \in \left\langle \operatorname{LM}(g_1), \ldots, \operatorname{LM}(g_s) \right\rangle$ for any $f \in I$. Write such a f as $f = q_1 g_1 + \cdots + q_s g_s$ (by assumption the remainder is zero). Lemma 1.15 tells us that $\operatorname{mdeg}(q_i g_i) \preceq \operatorname{mdeg}(f)$ for each i, and we must have equality for some i, since otherwise $\operatorname{mdeg}(q_1 g_1 + \cdots + q_s g_s) \prec \operatorname{mdeg}(f)$. Pick such an i, then $\operatorname{LM}(f) = \operatorname{LM}(q_i g_i) = \operatorname{LM}(q_i) \operatorname{LM}(g_i) \in \left\langle \operatorname{LM}(g_1), \ldots, \operatorname{LM}(g_s) \right\rangle$. □

Definition 1.19 (Gröbner basis). A *Gröbner basis* of an ideal $I \subseteq \mathbb{K}[\underline{X}]$ is a finite set $G = \{g_1, \ldots, g_s\} \subseteq I$ satisfying any of the equivalent conditions in Theorem 1.18, in particular, $\langle G \rangle = I$ by (c). $\lrcorner$

Remark. Characterization (a) is often used in the literature as the definition of a Gröbner basis, as it applies in the more general setting where $\mathbb{K}$ is replaced with an arbitrary commutative ring [24, Remark 9.11] or a "non-global" monomial ordering is used [19, Section 1.2, 1.6]. Another reason is that we can prove existence of Gröbner bases easily, see the following corollary.

If we slightly modify the normal form algorithm 3, then independence of choice in said algorithm is actually also equivalent to being a Gröbner basis. This leads to the notion of *confluence of rewrite relations* and is, for example, explored in the book by Kreuzer & Robbiano, where even more characterizations of Gröbner bases are presented [28, Theorem 2.4.1].

Corollary 1.20. *Any ideal $I \subseteq \mathbb{K}[\underline{X}]$ admits a Gröbner basis.*

Proof. The set $\operatorname{LM}(I)$ generates the ideal $\operatorname{IN}(I)$ (by definition). As the ring $\mathbb{K}[\underline{X}]$ is Noetherian (Hilbert's basis theorem A.1), any generating set contains a finite generating subset $\{\operatorname{LM}(g_1), \dots, \operatorname{LM}(g_m)\}$. Then $G = \{g_1, \dots, g_m\}$ is a Gröbner basis of I by characterization (a). □

Example 1.21. Any finite set of monomials G is a Gröbner basis for the ideal $\langle G \rangle$. The polynomials $g_1 = XY + 1$, $g_2 = Y^2 - 1$ from example 1.16 are not a Gröbner basis (they violate (c)). ⌟

Theorem 1.18 and its proof have revealed that normal forms with respect to Gröbner bases are unique, and suggests that there is a normal form with respect to the whole ideal I.

Theorem 1.22 (The normal form map NF_I). *Let $I \subseteq \mathbb{K}[\underline{X}]$ be an ideal.*

(i) *For each $f \in \mathbb{K}[\underline{X}]$ there exists a unique f^* such that $f - f^* \in I$ and f^* is in normal form with respect to I.*

We denote this element as $\mathrm{NF}_I(f) = f^*$.

(ii) *If G is a Gröbner basis of I, then $\mathrm{NF}_G(f) = \{\mathrm{NF}_I(f)\}$, in particular, any two Gröbner bases of I define the same (unique) normal form.*

(iii) *The map $\mathrm{NF}_I \colon \mathbb{K}[\underline{X}] \to \mathbb{K}[\underline{X}]$ is $\mathbb{K}$-linear with kernel $\ker \mathrm{NF}_I = I$.*

We note that the map NF_I is *not* multiplicative, for example $\mathrm{NF}_{\langle x^2 - x \rangle}(x^2) = x = \mathrm{NF}_{\langle x^2 - x \rangle}(x)$.

Proof. (i) Let G be a Gröbner basis of I, then by 1.18(a) being in normal form with respect to G is equivalent to being in normal form with respect to I. Hence by 1.18(b) there is a *unique* $r \in \mathbb{K}[\underline{X}]$ in normal form with respect to I and with $f - r \in I$.

(ii) We just showed that the polynomial(s) in NF_G satisfy the condition in (i).

(iii) For linearity consider $f, g \in \mathbb{K}[\underline{X}]$, $\lambda \in \mathbb{K}$, $f^* = \mathrm{NF}_I(f)$, $g^* = \mathrm{NF}_I(g)$. Then λf^* and $f^* + g^*$ are also in normal form with respect to I (the monomials which occur must also occur in f^* or g^*). We also have

$$\lambda f - \lambda f^* = \lambda(f - f^*) \in I \quad \text{and} \quad (f + g) - (f^* + g^*) = (f - f^*) + (g - g^*) \in I,$$

so $\mathrm{NF}_I(\lambda f) = \lambda f^*$ and $\mathrm{NF}_I(f + g) = f^* + g^*$. The fact that $\ker \mathrm{NF}_I = \ker \mathrm{NF}_G = I$ is a reformulation of Theorem 1.18(c). □

We can characterize the normal form by a minimality condition: We can extend $\prec$ to finite sets of monomials $M, N \subseteq \mathrm{Mon}_n$ by looking for the largest element not in the other set

$$M \prec_{\mathfrak{P}} N \quad \text{if and only if} \quad \max_{\prec}(M \setminus N) \prec \max_{\prec}(N \setminus M).$$

The order $\prec_{\mathfrak{P}}$ is still a well-order [43, Lemma 2.4].

Lemma 1.23. *Let $I \subseteq \mathbb{K}[\underline{X}]$ be an ideal, $f \in \mathbb{K}[\underline{X}]$ and set $[f]_{\equiv_I} := \{ f + h \mid h \in I \}$. Then $\mathrm{NF}_I(f)$ is the unique element of $[f]_{\equiv_I}$ with minimal support (with respect to $\prec_{\mathfrak{P}}$).*

Proof. Consider a $f' \in [f]_{\equiv_I}$ with minimal support (f' exists since $\prec_{\mathfrak{P}}$ is a well-order). Any reduction step in the normal form algorithm strictly decreases the support with respect to $\prec_{\mathfrak{P}}$, so by assumption $f' = \mathrm{NF}_I(f') \overset{f'-f \in I}{=} \mathrm{NF}_I(f)$. □

1.5 Reduced Gröbner bases and uniqueness

If G is a Gröbner basis of some ideal I, then $G \cup \{f\}$ is also a Gröbner basis for any polynomial f. Hence there are many Gröbner bases for the same ideal. A first step towards making Gröbner bases unique is the observation that if $f, g \in G$ with $\mathrm{LM}(f) \mid \mathrm{LM}(g)$, then $G \setminus \{g\}$ is still a Gröbner basis (see below). This leads to the following definition:

Definition 1.24 (Interreduced Gröbner basis). A set of polynomials S is *normalized* if $0 \notin S$ and $\mathrm{LC}(f) = 1$ for all $f \in S$.

A Gröbner basis G is *interreduced* if it is normalized and for each $g \in G$ we have $g \notin \langle \mathrm{LM}(G \setminus \{g\}) \rangle$. ⌟

Lemma 1.25. *Let G be a Gröbner basis of I.*

(i) *If there is a $g \in G$ such that $\mathrm{LM}(g) \in \langle \mathrm{LM}(G \setminus \{g\}) \rangle$, then $G \setminus \{g\}$ is also a Gröbner basis of I.*

(ii) *Assume that G is normalized. Then G is interreduced if and only if it is minimal, i. e. it does not properly contain a Gröbner basis of I. In particular, G contains an interreduced Gröbner basis.*

(iii) *Any two interreduced Gröbner bases G, G' of I have the same length and $\mathrm{LT}(G) = \mathrm{LT}(G')$*

Proof. (i) Using the definition of a Gröbner basis, we have

$$\mathrm{IN}(I) = \langle G \rangle = \langle \{\mathrm{LT}(g)\} \cup \mathrm{LT}(G \setminus \{g\}) \rangle = \langle \mathrm{LT}(G \setminus \{g\}) \rangle .$$

By Theorem 1.18(a), this is again a Gröbner basis of I.

(ii) If G is minimal, then by (i) it must be interreduced. Conversely, if $G' \subsetneq G$ is a Gröbner basis for I and $g \in G \setminus G'$, then $\mathrm{LM}(g) \in \mathrm{IN}(I) = \langle \mathrm{LM}(G') \rangle$.

(iii) Let $g \in G$, then $\operatorname{LM}(g) \in \operatorname{IN}(I) = \langle \operatorname{LM}(G') \rangle$. Since this is a monomial ideal, we have $\operatorname{LM}(g') \mid \operatorname{LM}(g)$ for some $g' \in G'$ (Lemma 1.6). Repeating this argument with g' yields a $g'' \in G$ with $\operatorname{LM}(g'') \mid \operatorname{LM}(g')$, so $\operatorname{LM}(g'') \mid \operatorname{LM}(g)$. As G is interreduced, we get that $g = g''$ and hence $\operatorname{LT}(g) = \operatorname{LT}(g') = \operatorname{LT}(g'')$ (here we use that G and G' are normalized). Thus $g \in \operatorname{LT}(G')$, as g was arbitrary we have $\operatorname{LT}(G) \subseteq \operatorname{LT}(G')$ and by symmetry $\operatorname{LT}(G') \subseteq \operatorname{LT}(G)$. Since all leading terms are distinct (again by interreducedness) we have $|G| = |\operatorname{LT}(G)| = |G'|$. □

Example 1.26. If I is a monomial ideal generated by $G = \{X^{\alpha_1}, \dots, X^{\alpha_s}\}$, then G is interreduced if and only if none of the monomials divide each other. By Lemma 1.6 these monomials are precisely the *minimal* monomials in $\operatorname{Mon}_n \cap I$ with respect to divisibility. ⌟

Unfortunately, interreduced Gröbner bases are not unique either, for example $\{X, Y\}$ and $\{X + Y, Y\}$ are both interreduced Gröbner bases for $I = \langle X, Y \rangle$ (since they have the same leading monomial). This indicates that the lower terms must also be considered, which leads to the "correct" definition.

Definition 1.27 (Reduced Gröbner basis). A Gröbner basis G is *reduced* if it is normalized and for all $g \in G$ and *any* $m \in \operatorname{supp}(g)$ we have $m \notin \big\langle \operatorname{LM}(G \setminus \{g\}) \big\rangle$. ⌟

Theorem 1.28. *Any ideal $I \subseteq \mathbb{K}[\underline{X}]$ admits a unique reduced Gröbner basis.*

Proof. **Existence:** By the previous lemma there exists an interreduced Gröbner basis $\{g_1, \dots, g_s\}$ for I. For $i = 1, \dots, s$ define inductively

$$h_i \coloneqq \operatorname{rem}(g_i; h_1, \dots, h_{i-1}, g_{i+1}, \dots, g_s).$$

We claim that $\operatorname{LT}(h_i) = \operatorname{LT}(g_i)$ by induction on i. By construction we can write

$$g_i = \sum_{j=1}^{i-1} q_j h_j + \sum_{j=i+1}^{s} q_j g_j + h_i, \qquad \operatorname{mdeg}(q_j h_j), \operatorname{mdeg}(q_j g_j) \preceq \operatorname{mdeg}(g_i). \tag{1.5}$$

By induction $\operatorname{LT}(q_j h_j) = \operatorname{LT}(q_j g_j)$ for $j = 1, \dots, i-1$, but by interreducedness none of the terms $\operatorname{LT}(q_j g_j)$ can equal $\operatorname{LT}(g_i)$ $j = 1, \dots, i-1, i+1, \dots, s$. In order for (1.5) to hold, we must therefore have $\operatorname{LT}(g_i) = \operatorname{LT}(h_i)$.

Now $G' \coloneqq \{h_1, \dots, h_s\}$ is a reduced Gröbner basis for I: By construction $G' \subseteq I$ and by the claim $\langle \operatorname{LT}(G') \rangle = \langle \operatorname{LT}(G) \rangle = \operatorname{IN}(I)$, so G' is a Gröbner basis for I. Clearly it is normalized, and by construction no term in h_i is divisible by any leading monomial of any other h_i (coinciding with the leading monomials of the respective g_i).

Uniqueness: Let G, G' be two reduced Gröbner bases for I. By the previous lemma we get $G = \{g_1, \dots, g_s\}$, $G' = \{g'_1, \dots, g'_s\}$ with $\operatorname{LT}(g_i) = \operatorname{LT}(g'_i)$. No monomial in $g_i - g'_i$ is divisible by any $\operatorname{LM}(G)$: This is clear for $\operatorname{LM}(G \setminus \{g_i\})$ by reducedness of G and G', and $\operatorname{LM}(g_i) \nmid \operatorname{LT}(g_i - g'_i)$, as $\operatorname{LM}(g_i - g'_o) \prec \operatorname{LT}(g_i)$. This shows that $g_i - g'_i$ coincides with its remainder $\operatorname{rem}(g_i - g'_i; G)$, which vanishes since $g_i - g'_i \in I$. So $g_i = g'_i$, as i was arbitrary we get $G = G'$. □

Thus ideals of polynomials are uniquely characterized by their reduced Gröbner basis. They can also be used to find the minimal size of a Gröbner basis for I:

Corollary 1.29. *The reduced Gröbner basis of an ideal has minimal length (number of elements) and minimal largest multidegree among all Gröbner bases of I.*

Proof. Any Gröbner basis G of I can be transformed into the reduced Gröbner basis G_0 by following the steps in Theorem 1.28. These steps never increase the number or multidegree of the elements, hence the claim follows. □

We use this uniqueness property to define a decision problem associated to Gröbner bases:

Definition 1.30 (Reduced Gröbner basis membership problem, $\mathsf{GROEBM}_{\mathbb{K}}$).

- *Input:* $(g, f_1, \ldots, f_s)$ multivariate polynomials from $\mathbb{K}[X_1, \ldots, X_n]$
- *Output:* Decide whether or not g is contained in the unique reduced Gröbner basis of $I = \langle f_1, \ldots, f_n \rangle$ ⌟

The normal form map NF_I can also be used to characterize the unique reduced Gröbner basis directly. We follow the notation of [30, Section 5].

Definition 1.31 ((Ir)reducible, minimally reducible). Let $I \subseteq \mathbb{K}[\underline{X}]$ be an ideal.

(i) A polynomial $f \in \mathbb{K}[\underline{X}]$ is *reducible with respect to I* if $\mathrm{NF}_I(f) \neq f$, and otherwise *irreducible* with respect to I.

(ii) A monomial m is called *minimally reducible with respect to I* if it is reducible, but all proper divisors $m' \mid m$ are irreducible with respect to I. ⌟

A polynomial f is reducible with respect to I if and only if for any Gröbner basis G there is a $g \in G$ with $\mathrm{LM}(g) \mid \mathrm{LM}(f)$.

Theorem 1.32. *The unique reduced Gröbner basis of I is given by the set*

$$\left\{ m - \mathrm{NF}_I(m) \;\middle|\; m \in \mathrm{Mon}_n \text{ is minimally reducible} \right\} \subseteq I.$$

Proof. Let G be the reduced Gröbner basis of I and G' the set defined in the theorem.

If $g \in G$, $m = \mathrm{LT}(g)$, then $m - g$ is in normal form with respect to I (as G is reduced), so $\mathrm{NF}_I(m) = m - g$. In particular m is reducible and $g = m - \mathrm{NF}_I(m)$. Assume that some proper divisor $m' \mid m$ were reducible, i. e. $\mathrm{LM}(g') \mid m'$ for some $g' \in G$, then $\mathrm{LM}(g') \mid m = \mathrm{LM}(g)$, which is impossible for G interreduced. We conclude $g \in G'$.

On the other hand, let $m - \mathrm{NF}_I(m) \in G'$, then $\mathrm{LM}(g) \mid m$ for some $g \in G$. Since m is *minimally* reducible, we must have $\mathrm{LM}(g) = m$. Applying the normal form algorithm 3 to m yields the remainder $m - g$, since no term of $\mathrm{LT}(g) - g$ is in $\langle \mathrm{LT}(G) \rangle$ by reducedness of G. So $m - \mathrm{NF}_I(m) = g \in G$ as desired. □

Thus, it is (in theory) possible to enumerate the reduced Gröbner basis by enumerating all monomials m and calculating normal forms of (divisors of) m. This will be crucial in proving an exponential space lower bound on the task of calculating the reduced Gröbner basis.

1.6 The case of binomial ideals

To illustrate the theory we have introduced so far, we apply it to binomial ideals. This serves both as an interesting example as well as a fundamental ingredient in proving lower bound on the length of Gröbner bases in chapter 3. We partially follow Koppenhagen & Mayr [27], for a detailed exposition of binomial ideals and their properties consider the article by Eisenbud & Sturmfels [15].

Let $G = (a_1 X^{\alpha_1} + b_1 X^{\beta_1}, \dots, a_s X^{\alpha_s} + b_s X^{\beta_s})$ and $I = \langle G \rangle$ be a binomial ideal; if all $a_i = 1$, $b_i = -1$, then we have a pure difference ideal. We start by showing that both the normal form algorithm and the (reduced) Gröbner basis associated to G resp. I again produce binomials and pure differences.

Lemma 1.33. *Let m be a monomial and f a binomial.*

(i) $\operatorname{rem}(m; G)$ *is a term and* $\operatorname{rem}(f; G)$ *is a binomial.*

(ii) *The reduced Gröbner basis of I consists of binomials.*

(iii) $\operatorname{NF}_I(m)$ *is a term and* $\operatorname{NF}_I(f)$ *is a binomial.*

If f and G consist of pure binomials, then the preceding statements hold true if "binomial" is replaced with "pure binomial" and "term" by "monomial".

Proof. (i) If p is a monomial, then each time a reduction in line 4–7 is applied, the resulting p is a term again, and even a monomial if G consists of pure binomials. If no $\operatorname{lt}(g_i) \mid p$, then $r = p$ is the remainder.

Similarly, if p is a binomial and $\operatorname{lm}(g_i) \mid \operatorname{lm}(p)$, then $p - \frac{\operatorname{lt}(p)}{\operatorname{lt}(g_i)} \cdot g_i$ is again a binomial, as the leading terms in this subtraction cancel. On the other hand, if no $\operatorname{lm}(g_i) \mid \operatorname{lm}(p)$, then in line 9–10 we have $r \leftarrow \operatorname{lt}(p) = a \pm X^{\alpha}$ and $p \leftarrow p - \operatorname{lt}(p) = bX^{\beta}$. From that point on p is a monomial and the previous discussion applies (with the additional constant factor b) and the result is a binomial.

Furthermore, if all binomials involved are actually differences of monomials, then the result will again be a pure binomial (note that $b = \pm 1$ in this case).

(ii) We postpone the proof for a few pages until the introduction of Buchberger's algorithm, see example 2.7.

(iii) This follows from the previous two statements, as $\operatorname{NF}_I(f) = \operatorname{rem}(f; G')$, where G' is the reduced Gröbner basis of I. □

We now turn to the case of pure binomial ideals and draw a connection to congruence relations on monomials, that is, equivalence relations compatible with multiplication (Definition A.4). We define $X^\alpha \equiv_I X^\beta$ if and only if $X^\alpha - X^\beta \in I$. This makes sense for all ideals I, but for pure a pure difference ideals we obtain a description of the normal form map, Gröbner basis and initial ideal in terms of equivalence classes.

Theorem 1.34 (Koppenhagen & Mayr 1999 [27]). *Let I be a pure difference ideal with reduced Gröbner basis G.*

(i) *For $m \in \mathrm{Mon}_n \mathrm{NF}_I(m)$ is the minimal monomial in the equivalence class $[m]_{\equiv_I}$ (w.r.t. $\prec$).*

(ii) *If $X^\alpha - X^\beta \in G$, $\alpha \succ \beta$, then X^β is the minimal monomial in $[X^\alpha]_{\equiv_I}$.*

(iii) *For any $X^\alpha \in \mathrm{IN}(I)$ there is a $X^\beta \prec X^\alpha$ such that $X^\alpha - X^\beta \in I$.*

(iv) $\mathrm{Mon}_n \cap \mathrm{IN}(I)$ *consists of the monomials m which are* not *the minimal element in $[m]_{\equiv_I}$.*

Proof. (i) By the previous Lemma, $\mathrm{NF}_I(m)$ is a monomial which is by definition in $[m]_{\equiv_I}$. We also know that the normal form is a polynomial of minimal multidegree with $m - \mathrm{NF}_I(m) \in I$ (Lemma 1.23), so $\mathrm{NF}_I(m)$ is the unique minimal monomial in the class of m.

(ii) This is a direct consequence of Theorem 1.32 and statement (i).

(iii) Let $G = \{X^{\alpha_1} - X^{\beta_1}, \ldots, X^{\alpha_s} - X^{\beta_s}\}$ be the reduced Gröbner basis of I, consisting of differences of monomials by the previous lemma. As $X^\alpha \in \mathrm{IN}(I) = \langle \mathrm{LT}(G) \rangle$, there is a $g = X^{\alpha_i} - X^{\beta_i} \in G$ with $\mathrm{LT}(g) = X^{\alpha_i} \mid X^\alpha$. Let $\gamma := \alpha - \alpha_i$, $\beta := \beta_i + \gamma$; then $X^\gamma g = X^\alpha - X^\beta \in I$ has leading term X^α.

(iv) Let $X^\alpha \in \mathrm{IN}(I)$, then (iii) yields a X^β with $X^\beta \equiv_i X^\alpha$ and $X^\beta \prec X^\alpha$. Conversely, if m is not the minimal element of $[m]_{\equiv_I}$, then by (i) $\mathrm{NF}_I(m) \prec m$, so $m - \mathrm{NF}_I(m) \in I$ has leading monomial m. □

1.7 Representing polynomials

In order to perform algorithmic manipulation on polynomials we need to fix a way to represent polynomials and their coefficients over a (finite) fixed alphabet. Let $\mathrm{enc}\colon K \to \mathbb{K}$ be an encoding, $K \subseteq \Sigma^*$. In order to do arithmetic calculations in the field of coefficients, we would like to be able to perform the usual field operations $+, -, \cdot, /$ at low computational cost. More precisely, for $a, b \in K$, one can calculate $c, d, e, f \in K$ in polynomial time with

$$\mathrm{enc}(c) = \mathrm{enc}(a) + \mathrm{enc}(b), \quad \mathrm{enc}(d) = -\mathrm{enc}(a), \quad \mathrm{enc}(e) = \mathrm{enc}(a) \cdot \mathrm{enc}(b),$$
$$\mathrm{enc}(f) = \mathrm{enc}(a)^{-1} \text{ if } \mathrm{enc}(a) \neq 0.$$

Furthermore, equality of elements $\mathrm{enc}(a) = \mathrm{enc}(b)$ must also be decidable in polynomial time.

This will be our standing assumption for the rest of the thesis. For example, if $\mathbb{K}$ is a finite field, then we can calculate any of the arithmetic operations in constant time with a (large but constant size) lookup table. Infinite fields pose a more interesting challenge.

Example 1.35. If $\mathbb{K} = \mathbb{Q}$ then we can use the following encoding scheme: We encode $q = a/b \in \mathbb{Q}$ with $b > 0$, as $\mathrm{bin}(a)/\mathrm{bin}(b)$. We can then use integer arithmetic to calculate

$$\frac{a}{b} + \frac{c}{d} = \frac{ad + bc}{bd}, \quad -\frac{a}{b} = \frac{-a}{b}, \quad \dots$$

and can test for equality using $\frac{a}{b} = \frac{c}{d}$ if and only if $ad = cb$. Notice that the denominators will swell pretty quickly, so it might be well-advised to simplify the fraction if desired. ⌟

Remark. Strictly speaking, this is *not* efficient enough for good upper bounds. Instead one should require addition and multiplication to be in AC^0 and NC^1 respectively, in order to obtain a *well-endowed* ring [43, Definition 3.14]. This is necessary in order to enable fast matrix operations as in section 2.5. But we will only consider the integers and rational numbers, which are in fact well-endowed, so we do not elaborate this further.

Suppose now that we have fixed a suitable encoding scheme for $\mathbb{K}$. Then there are two important choices to be made in order to encode polynomials from $\mathbb{K}[X_1, \dots, X_n]$:

(i) Are the exponents of the monomials encoded in binary or is a monomial represented by a sequence of variables? The former is called binary exponent representation, the latter unary representation.

(ii) Are all coefficients written down (up to the leading term), or only those in the support of f? The first case is a dense encoding, the second case is sparse encoding.

We will use spare polynomial encoding with binary exponent representation, although it is remarkable that the lower bounds also hold true with unary exponent notation.

We now explain how to represent and calculate a monomial order. Robbiano showed that any monomial order $\prec$ on Mon_n can be represented by a finite set of weight vectors $W = (W_1, \dots, W_n)$, in the following sense: Let $W_k = (w_{k,i})_i \in \mathbb{R}^n$, the weight vectors can be interpreted as linear functions

$$W_k \colon \mathbb{N}^n \to \mathbb{R}, \qquad W_k(\alpha) \coloneqq \sum_{i=1}^{n} w_{k,i}\alpha_i.$$

Then W induces a monomial order with $\alpha \succ_W \beta$ if and only if there is a $k \in \{1, \dots, n\}$ with

$$W_k(\alpha) > W_k(\beta) \text{ and for all } j < k \; W_j(\alpha) = W_j(\beta).$$

This is very similar to the definition of the lexicographic order $\prec_{\text{lex}}$, and in a sense *all* monomial orders can be understood as lexicographic orders with weights:

Theorem 1.36 (Robbiano 1985 [44]). *For any monomial order $\prec$ there is a weight matrix $W \in \mathrm{Mat}(n\times n, \mathbb{R})$ with $\prec_W = \prec$.*

Example 1.37. In the case of $\prec_{\text{lex}}$ the weights have the simple form $W_k(\alpha) = \alpha_k$, so W is the identity matrix. The graded lexicographic order can be represented by

$$\begin{aligned} W_1(\alpha) &= \alpha_1 + \cdots + \alpha_n = |\alpha| \\ W_k(\alpha) &= \alpha_{k-1}, \qquad k = 2, \ldots, n. \end{aligned}$$

Similarly, for $\prec_{\text{grevlex}}$ the following weight functions may be used

$$\begin{aligned} W_1(\alpha) &= \alpha_1 + \cdots + \alpha_n = |\alpha| \\ W_k(\alpha) &= -\alpha_{n-k+2}, \qquad k = 2, \ldots, n. \end{aligned}$$

⌟

Dubé, Mishra & Yap [13] gave a constructive proof of this theorem and also showed that the entries of W may taken to be non-negative. In the previous example $\prec_{\text{grevlex}}$ can be represented by $W_k(\alpha) = \alpha_k + \cdots + \alpha_n$, $k = 1, \ldots, n$.

In order to represent an ordering in a finite number of bits, we restrict ourselves to the case where the weights are non-negative rational numbers $W \in \mathrm{Mat}(n\times n, \mathbb{Q}_{\geq 0})$. This is not a harsh restriction, as for any $\prec$ and $D \in \mathbb{N}$ there is a rational matrix $\tilde{W}$ such that $\prec_{\tilde{W}}$ coincides with $\prec$ on monomials of degree $\leq D$.

Remark. The representation of monomial orders as weight matrices is mainly useful for theoretical considerations (for example in Chapter 2). In actual computer algebra systems it is usually much faster to directly implement the monomial orders of interest.

2

Algorithms and upper bounds

In this chapter we describe two fundamentally different approaches to the task of computing Gröbner bases and solving the ideal membership problem. The first is Buchberger's Algorithm, which uses so-called S-polynomials to enlarge a given set of generators to a Gröbner basis. The second approach is based on degree bounds for the polynomials in an ideal membership certificate or a reduced Gröbner basis, and uses linear algebra techniques to obtain exponential space algorithms. We also compare complexity-theoretic upper bounds results for different classes of polynomials.

The material on Buchberger's algorithm is standard to computational commutative algebra texts, we follow [24, Chapter 9]. The content of the later sections is inspired by survey articles of Mayr [32] and Mayr & Toman [35] together with the original literature.

2.1 S-Polynomials

We start with proving a criterion for a set $G = \{g_1, \ldots, g_s\}$ to constitute a Gröbner basis of $\langle G \rangle$. The greatest common divisor and least common multiple for monomials X^α, X^β is defined similarly to the integers as

$$\gcd(X^\alpha, X^\beta) := X_1^{\max\{\alpha_1,\beta_1\}} \cdots X_n^{\max\{\alpha_n,\beta_n\}}, \quad \operatorname{lcm}(X^\alpha, X^\beta) := X_1^{\min\{\alpha_1,\beta_1\}} \cdots X_n^{\min\{\alpha_n,\beta_n\}}.$$

Definition 2.1 (S-Polynomial). Let $0 \neq f, g \in \mathbb{K}[\underline{X}]$ be two polynomials. The *S-polynomial* is defined as

$$\operatorname{Spoly}(f,g) := \frac{\operatorname{LT}(g)}{t} \cdot f - \frac{\operatorname{LT}(f)}{t} \cdot g, \qquad t := \gcd(\operatorname{LM}(f), \operatorname{LM}(g)). \qquad \lrcorner$$

Equivalently, using $\gcd(X^\alpha, X^\beta) \cdot \operatorname{lcm}(X^\alpha, X^\beta) = X^\alpha \cdot X^\beta$, we can write

$$\operatorname{Spoly}(f,g) := \frac{\operatorname{LC}(g)t'}{\operatorname{LM}(f)} \cdot f - \frac{\operatorname{LC}(f)t'}{\operatorname{LM}(g)} \cdot g, \qquad t' := \operatorname{lcm}(\operatorname{LM}(f), \operatorname{LM}(g)).$$

Example 2.2. The S-polynomial of two monomials is always zero. If f, g are differences of monomials or more generally binomials, then their S-polynomial is also a difference of monomials

resp. a binomial. For example, if $f = X^{\alpha_1} - X^{\beta_1}$, $g = X^{\alpha_2} - X^{\beta_2}$ with $\alpha_i > \beta_i$, then

$$\operatorname{Spoly}(f,g) = \frac{X^{\alpha_2}}{t}\cdot(X^{\alpha_1}-X^{\beta_1}) - \frac{X^{\alpha_1}}{t}\cdot(X^{\alpha_2}-X^{\beta_2}) = \frac{X^{\alpha_1+\beta_2}}{t} - \frac{X^{\alpha_2+\beta_1}}{t}, \quad t \coloneqq \gcd(X^{\alpha_1}, X^{\alpha_2}). \qquad \lrcorner$$

The key property of the S-polynomial is that the leading terms of $\frac{\operatorname{LT}(g)}{t}\cdot f$ and $\frac{\operatorname{LT}(f)}{t}\cdot g$ cancel, as both equal $\operatorname{LC}(f)\operatorname{LC}(g)t'$. With this notion we can formulate and prove Buchberger's criterion.

Theorem 2.3 (Buchberger's criterion). *Let $G = (g_1, \dots, g_s)$ be a finite list of polynomials. The following are equivalent:*

(a) *G is a Gröbner basis of $I = \langle G\rangle$;*

(b) *For all $f, g \in G$ we have $0 \in \operatorname{NF}_G(\operatorname{Spoly}(f,g))$;*

(b′) *For all $f, g \in G$ we have $\operatorname{rem}(\operatorname{Spoly}(f,g); G) = 0$.*

Proof. We follow the proof given by Kemper [24, Theorem 9.12].

(a)⇒(b′)⇒(b): All S-polynomials of elements $f, g \in G \subseteq I$ are linear combinations of elements of G and hence lie in $I = \langle G\rangle$. By characterization 1.18(c) we have $\operatorname{Spoly}(f,g) \operatorname{rem} G = 0$, and thus clearly $0 \in \operatorname{NF}_G(\operatorname{Spoly}(f,g))$.

(b)⇒(a): Assume (b) but also that G is *not* a Gröbner basis. We will arrive at a contradiction picking a minimal counterexample (with respect to the well-order $\prec$) and then construct an even smaller counterexample; thus showing that no such counterexample can exist.

Step 1: Construct a minimal expression of a counterexample.

By 1.18(a) there is a $f \in I$ with $\operatorname{LM}(f) \notin \langle \operatorname{LT}(G)\rangle$, since $f \in I$ the set

$$H = \left\{ (h_1, \dots, h_s) \in \mathbb{K}[\underline{X}]^s \;\middle|\; f = h_1 g_1 + \cdots + h_s g_s \right\}$$

is not empty. Consider the map $H \to \mathbb{N}^n$ which assigns $(h_1, \dots, h_s)$ the maximum multidegree $\max_i \operatorname{mdeg}(h_i g_i)$ with respect to $\prec$. As $\prec$ is a well-order, this map takes its minimal value α for some $(h_1, \dots, h_s) \in H$; let $t \coloneqq X^\alpha$. Since

$$f = \sum_i h_i g_i, \tag{2.1}$$

$\operatorname{LM}(f)$ occurs in some $\operatorname{supp}(h_i g_i)$, but not as its leading monomial (as $\operatorname{LM}(f) \notin \langle \operatorname{LM}(G)\rangle$), so $\alpha \succeq \operatorname{mdeg}(h_i g_i) \succ \operatorname{mdeg}(f)$. In particular, t does not occur in $\operatorname{supp}(f)$, and so comparing coefficients of t in (2.1) and using $\alpha \succeq \operatorname{mdeg}(h_i g_i)$ yields

$$0 = \sum_{i=1}^{s} c_i \cdot \operatorname{LC}(g_i), \qquad c_i \coloneqq \begin{cases} \operatorname{LC}(h_i) & \text{if } \operatorname{LM}(h_i g_i) = t \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

By construction of t we must have $c_i \neq 0$ for some i, after reordering the g_i we may assume $c_1, \neq 0$.

Step 2: Apply (b) to the S-polynomials $\operatorname{Spoly}(g_i, g_1)$.

Let $i \geq 2$ with $c_i \neq 0$, then $\operatorname{LM}(g_1), \operatorname{LM}(g_i) \mid t$ by (2.2) and hence $t_i := \operatorname{lcm}(\operatorname{LM}(g_1), \operatorname{LM}(g_i)) \mid t$. The S-polynomial is then

$$\operatorname{Spoly}(g_i, g_1) = \frac{\operatorname{LC}(g_1)t_i}{\operatorname{LM}(g_i)} \cdot g_i - \frac{\operatorname{LC}(g_i)t_i}{\operatorname{LM}(g_1)} \cdot g_1$$

and by the observation after the definition of S-polynomials we see $\operatorname{LM}(\operatorname{Spoly}(g_i, g_1)) \prec t_i$. By (b) 0 is a normal form of $\operatorname{Spoly}(g_i, g_1)$, hence by definition we have representations

$$\operatorname{Spoly}(g_i, g_1) = \sum_{j=1}^{s} q_{i,j} g_j, \qquad \operatorname{LM}(q_{i,j} g_j) \preceq \operatorname{LM}(\operatorname{Spoly}(g_i, g_1)) \prec t_i, \quad j = 1, \ldots, s.$$

Let $s_i := t/t_i \cdot \operatorname{Spoly}(g_i, g_1)$, we derive two expressions for s_i:

(i) Using $t = \operatorname{LM}(h_i)\operatorname{LM}(g_i) = \operatorname{LM}(h_1)\operatorname{LM}(g_1)$ we see

$$s_i = \frac{\operatorname{LC}(g_1)t}{\operatorname{LM}(g_i)} \cdot g_i - \frac{\operatorname{LC}(g_i)t}{\operatorname{LM}(g_1)} \cdot g_1 = \operatorname{LC}(g_1)\operatorname{LM}(h_i) \cdot g_i - \operatorname{LC}(g_i)\operatorname{LM}(h_1) \cdot g_1.$$

(ii) The normal form representation of $\operatorname{Spoly}(g_i, g_1)$ from above together with the multidegree inequalities yield

$$s_i = \sum_{j=1}^{s} \frac{t}{t_i} q_{i,j} g_j, \qquad \operatorname{LM}\left(\frac{t}{t_i} q_{i,j} g_j\right) \prec t, \quad j = 1, \ldots, s.$$

Step 3: Construct a smaller representation $(h'_1, \ldots, h'_s) \in H$, leading to a contradiction.

Let $g := \sum_{i=1}^{r} c_i \operatorname{LM}(h_i) g_i$, we can rewrite this sum in order to apply the previously obtained identities:

$$g = \frac{1}{\operatorname{LC}(g_1)} \left(\sum_{i=2}^{s} c_i \Big(\underbrace{\operatorname{LC}(g_1)\operatorname{LM}(h_i)g_i - \operatorname{LC}(g_i)\operatorname{LM}(h_1)g_1}_{= s_i \text{ by (i)}} \Big) + \underbrace{\sum_{i=1}^{s} c_i \operatorname{LC}(g_i)\operatorname{LM}(h_1)g_1}_{= 0 \text{ by (2.2)}} \right)$$

$$= \sum_{i=2}^{r} \frac{c_i}{\operatorname{LC}(g_1)} \cdot s_i \overset{\text{(ii)}}{=} \sum_{j=1}^{s} \Big(\underbrace{\sum_{i=2}^{s} \frac{c_i}{\operatorname{LC}(g_1)} \frac{t}{t_i} q_{i,j}}_{=: \tilde{h}_j} \Big) g_j, \qquad \operatorname{LM}(\tilde{h}_j g_j) \prec t, \quad j = 1, \ldots, s.$$

We thus have two combinations of g as a linear combination of the g_i, which can be combined

to obtain a new representation $(h'_1, \dots, h'_s) \in H$:

$$f = f - g + g = \sum_{i=1}^{r} \Big(\underbrace{h_i - c_i \operatorname{LM}(h_i) + \tilde{h}_i}_{=: h'_i} \Big) g_i.$$

It remains to how that $\max_i \operatorname{mdeg}(h'_i g_i) \prec \alpha$. If $c_i = 0$, then $\operatorname{mdeg}(h_i g_i) \prec \alpha$ and hence

$$\operatorname{mdeg}(h'_i g_i) = \operatorname{mdeg}(h_i g_i + \tilde{h}_i g_i) \preceq \max\{h_i g_i, \tilde{h}_i g_i\} \prec \alpha.$$

On the other hand if $c_i \neq 0$, then $\operatorname{mdeg}(h_i - c_i \operatorname{LM}(h_i)) = \operatorname{mdeg}(h_i - \operatorname{LT}(h_i)) \succ \operatorname{mdeg}(h_i)$ and

$$\operatorname{mdeg}(h'_i g_i) \preceq \max\{\operatorname{mdeg}(h_i - c_i \operatorname{LM}(h_i)), \tilde{h}_i\} + \operatorname{mdeg}(g_i) \prec \alpha. \qquad \square.$$

Corollary 2.4. *Given a set of polynomials $G = \{g_1, \dots, g_s\} \subseteq \mathbb{K}[\underline{X}]_\prec$, the question whether G is a Gröbner basis of $\langle G \rangle$ is decidable.*

Proof. By Buchberger's criterion it suffices to verify $\operatorname{rem}(\operatorname{Spoly}(g_i, g_j), G) = 0$ for all i, j. $\square$

2.2 Buchberger's algorithm

In this section we present Buchberger's algorithm, historically the first algorithm calculating the Gröbner basis of an ideal, introduced by Buchberger in his PhD thesis [8].

Algorithm 4 Buchberger's algorithm

Require: $f_1, \dots, f_s \in \mathbb{K}[X_1, \dots, X_n]_\prec$
Ensure: G is a Gröbner basis of $I = \langle f_1, \dots, f_s \rangle$

```
G ← {f_1, ..., f_s} \ {0}
P := { (f_i, f_j) | 1 ≤ i < j ≤ s }
while P ≠ ∅ do
    Pick (f, g) ∈ P
    P ← P \ {(f, g)}
    s ← Spoly(f, g)
    s* ← rem(s; G)
    if s* ≠ 0 then
        P ← P ∪ { (s*, g) | g ∈ G }
        G ← G ∪ {s*}
    end if
end while
```

Theorem 2.5 (Correctness of Buchberger's algorithm). *Algorithm 4 terminates after a finite number of steps and calculates a Gröbner basis G of the ideal I generated by the input $f_1, \dots, f_s$.*

Proof. **Termination:** We first show that the branch in line 8–10 for $s^* \neq 0$ is only taken a finite number of times. Indeed, if $s^* \neq 0$, then in particular $\operatorname{LM}(s^*) \notin \langle \operatorname{LM}(G) \rangle$ and $\langle \operatorname{LM}(G) \rangle \subsetneq \langle \operatorname{LM}(G \cup \{s^*\}) \rangle$. So each pass of line 10 strictly increases the ideal $\langle \operatorname{LM}(G) \rangle$ and by the Noether property of $\mathbb{K}[\underline{X}]$ any ascending chain of ideals must eventually become stationary (see A.1). From this point on each pass through the loop decreases the size of P in line 5, so the algorithm terminates with $P = \emptyset$.

Correctness: Let G be the final result of the algorithm and G_0 the set at any point in the algorithm. We have the following invariants:

- $\langle G_0 \rangle = I$ (as $f_1, \ldots, f_s \in G_0$, line 1).
- For $f, g \in G_0$, $s = \operatorname{Spoly}(f, g)$, either $0 \in \operatorname{NF}_{G_0}(s)$ or G contains a normal form $\operatorname{NF}_{G_0}(s)$ (line 10).

In particular this applies to $G_0 = G$. If G contains a normal form r of $s = \operatorname{Spoly}(f, g)$, then $0 = r - r$ is also a normal form of s with respect to G. Hence by Buchberger's criterion 2.3 the set G is a Gröbner basis of I. □

Example 2.6. We take the polynomials $g_1 = XY + 1$, $g_2 = Y^2 - 1$ from example 1.16 with $\prec_{\text{lex}}$. We have $s_1 = \operatorname{Spoly}(g_1, g_2) = -X - Y$ and $s_1^* = \operatorname{rem}(s; g_1, g_2) = s_1$. Hence we add s_1^* to G and consider the new pairs (s_1^*, g_1) and (s_1^*, g_2). We have

$$s_2 = \operatorname{Spoly}(s_1^*, g_1) = -Y^2 + 1, \qquad s_3 = \operatorname{Spoly}(s_1^*, g_2) = -Y^3 - X$$

Now $s_2^* = \operatorname{rem}(s_2; g_1, g_2, s_1^*) = 0$ and $s_3^* = \operatorname{rem}(s_3; g_1, g_2, s_1^*) = 0$, hence the algorithm terminates and we obtain the Gröbner basis $\{XY + 1, Y^2 - 1, -X - Y\}$. ⌟

Remark. Consider a set of polynomials $\{f_1, \ldots, f_s\} \subseteq \mathbb{K}[\underline{X}]$ containing $X_i^2 - X_i$ for $X_i \in \underline{X}$. These polynomials enforce that any solution to this system takes values in $\{0, 1\}$. In this way Boolean formulae and equations can be expressed using polynomial systems, and one can techniques from commutative algebra to reason about such statements. For an interesting example see the "Gröbner proof system" by Clegg, Edmonds & Impagliazzo [10], who use Gröbner bases and Buchberger's algorithm to produce proof certificates for tautologies.

If one is interested in a reduced Gröbner basis, then algorithm 5 can be applied after producing an arbitrary Gröbner basis G:

The correctness of this algorithm follows from the results and proofs of Lemma 1.25 and Theorem 1.28. For example, the Gröbner basis from the previous example can be reduced to $\{X + Y, Y^2 - 1\}$. As a first application, we fill the gap in the proof of lemma 1.33 as promised.

Example 2.7 *(The reduced Gröbner basis of a binomial ideal).* If $G = \langle f_1, \ldots, f_s \rangle$ is a set of binomials, then any S-polynomial s formed from this set is also a binomial, and so is $s^* \operatorname{rem} G$ by Lemma 1.33(i). Hence Buchberger's algorithm adds only binomials to G and thus produces some Gröbner basis consisting of differences of monomials. Similarly, the algorithm reducing

Algorithm 5 Reduction of a Gröbner basis

Require: G a Gröbner basis of I
Ensure: G'' is the unique reduced Gröbner basis of I

```
G' ← ∅
for all g ∈ G do
    if g ≠ 0 and LM(g') ∤ LM(g) for all g' ∈ G' then        ▷ Interreduction
        G' ← G' ∪ {g/LC(g)}                                 ▷ Normalization
    end if
end for
G'' ← ∅
for all i = 1, ..., |G'| do                                 ▷ G' = {g_1, ..., g_r}
    h ← rem(g_i; G'' ∪ {g_{i+1}, ..., g_r})                 ▷ Reduction
    G'' ← G'' ∪ {h}
end for
```

this Gröbner basis will only delete or normalize elements from G and then introduce some remainders which are again binomials. Hence the reduced Gröbner basis consists of binomials. The same argument works for pure binomials and for homogeneous polynomials. ⌟

We also note that the combination of Buchberger's algorithm and the normal form algorithm yield:

Theorem 2.8. *The ideal membership problem* $\mathsf{IM}_{\mathbb{K}}$ *and (reduced) Gröbner bases* $\mathsf{GROEBM}_{\mathbb{K}}$ *are decidable (as long as the field operations are computable).*

The run-time of Buchberger's algorithm depends heavily on the number of Polynomials s which have to be added to G. Also, the majority of the computation is spent calculating the remainder of the s with respect to increasingly complicated sets G. A good implementation uses various techniques to reduce the number of unnecessary normal form calculations, for example [19, Exercise 1.7.2]:

Lemma 2.9 (Product criterion). *If* $f, g \in \mathbb{K}[\underline{X}]$ *with* $\gcd(\mathrm{LM}(f), \mathrm{LM}(g)) = 1$, *then*

$$0 \in \mathrm{NF}_{\{f,g\}}(\mathrm{Spoly}(f, g))$$

and hence $\mathrm{Spoly}(f, g)$ *may be skipped in Buchberger's algorithm.*

Remark. One of the most widely used improvements of Buchberger's algorithm are the algorithms F_4 and F_5 by Faugère [16]. For some bounds on the complexity of F_5 see for example the work of Bardet, Faugère & Salvy [2].

2.3 Degree bounds

In this section we gather some upper bounds on the degrees of polynomials appearing in the problems of our interest, following Mayr [32] and Mayr & Toman [35]. We start with the

ideal membership problem $\mathsf{IM}_{\mathbb{K}}$, let $f, f_1, \ldots, f_s \in \mathbb{K}[\underline{X}]$, then we are interested in bounds depending on the following factors:

- s, the number of generators $f_1, \ldots, f_s$.
- n, the number of variables in $\underline{X} = \{X_1, \ldots, X_n\}$.
- $d' := \deg f$, $d_i := \deg f_i$, the degrees. Without loss of generality we may assume $d_1 \geq \cdots \geq d_s$ and $d := \max_i d_i$.

In order to decide whether $f \in \langle f_1, \ldots, f_s \rangle$, one could try to systematically search the space of polynomials $(h_1, \ldots, h_s) \in \mathbb{K}[\underline{X}]^s$ in order to find a solution to

$$f = h_1 f_1 + \cdots + h_s f_s, \qquad h_1, \ldots, h_s \in \mathbb{K}[\underline{X}]. \tag{2.3}$$

A first step in order to make this feasible is to bound the degree of the h_i, historically the first result is due to Hermann, see Mayr & Meyer for a short proof [33, Appendix].

Theorem 2.10 (Hermann 1926 [20]). *If the equation (2.3) admits a solution, then there is a solution with*

$$\deg h_i \leq d' + (sd)^{2^n}, \qquad i = 1, \ldots, s.$$

In the next section we will see how this and the bounds below may be used to turn ideal membership, normal form and Gröbner basis calculation into problems from linear algebra.

A natural question is to ask whether this degree can be improved, since a better bound yields a system of linear equations with fewer variables and equations. In this generality, the double-exponential nature of the Hermann bound is unavoidable in the following sense: For any $k, d \in \mathbb{N}$ there exist polynomials $(f, f_1, \ldots, f_s)$, where $s = 10k + \mathcal{O}(1)$, the number of variables is $n = 10k + \mathcal{O}(1)$ and the degree is $\mathcal{O}(d)$, such that any solution to (2.3) has at least one h_i with

$$\deg(h_i) > n + d^{2^{k-1}}.$$

This result is due to Mayr & Meyer [33], we will prove a variant of this in chapter 3. On the other hand, if the f_i satisfy additional properties, then better bounds can be proved. The case of homogeneous polynomials has already been mentioned in Lemma 1.7, there the degrees of the h_i are simply bounded by d'.

Another important special case is the case $f = 1$, i. e. the Nullstellensatz case for $\mathsf{HNST}_{\mathbb{K}}$. Here a single exponential bound was proven by Brownawell for $\mathbb{K} = \mathbb{C}$, which was improved and extended to arbitrary fields by Kollár.

Theorem 2.11 (Kóllar 1988 [26]). *If $1 \in \langle f_1, \ldots, f_s \rangle$, then there exist $h_1, \ldots, h_s$ with $1 = \sum_{i=1}^s h_i f_i$ such that*

$$\deg h_i \leq \deg(h_i f_i) \leq N := \max\{3, d\} \prod_{i=1}^{\max\{s,n\}-1} \max\{3, \deg f_i\} \leq \max\{3, d\}^{\max\{s,n\}}.$$

This result is sometimes called the *effective Nullstellensatz*, as it yields explicit bounds on the degrees occurring in Hilbert's Nullstellensatz.

Finally, we give an example on how the dimension of the ideal influences the degree bound, for more on dimension see for example [19, Chapter 5].

Definition 2.12 (Dimension of an ideal). The *dimension* $\dim(I)$ of an ideal $I \subseteq \mathbb{K}[\underline{X}]$ is the largest integer $m \in \mathbb{N}$ such that there exist variables $X_{i_1}, \dots, X_{i_m} \in \underline{X}$ with

$$I \cap \mathbb{K}[X_{i_1}, \dots, X_{i_m}] = (0).$$ ⌟

Theorem 2.13 (Dickenstein et al. 1991 [12]). *Assume* (2.3) *admits a solution and the ideal I is of dimension zero. Then there is a solution with*

$$\deg(h_i f_i) \leq n d^{2n} + d^n + d + \deg(f).$$

We now turn to degree bounds of Gröbner bases, let $\prec$ be a monomial order on Mon_n. We can reduce to the homogeneous case as follows: Let $\underline{X} = \{X_1, \dots, X_n\}$ and add a new variable X_0.

Definition 2.14 ((De)homogenization). Let $f \in \mathbb{K}[X_1, \dots, X_n]$ of degree d, then its *homogenization* is

$${}^{\mathrm{h}}f = \sum_{\alpha \in \mathbb{N}^n,\, |\alpha| \leq d} f_\alpha X_0^{d-|\alpha|} X_1^{\alpha_1} \cdots X_n^{\alpha_n} = X_0^d f(\tfrac{X_1}{X_0}, \dots, \tfrac{X_n}{X_0}) \in \mathbb{K}[X_0, \dots, X_n].$$

If $g \in \mathbb{K}[X_0, \dots, X_n]$ is homogeneous, then its *dehomogenization* is

$${}^{\mathrm{a}}g = g(1, X_1, \dots, X_n) \in \mathbb{K}[X_1, \dots, X_n].$$ ⌟

We define an ordering on $\mathrm{Mon}(\{X_0, \dots, X_n\})$ via $X^\alpha \prec_{\mathrm{h}} X^\beta$ if and only if

$$\deg X^\alpha < \deg X^\beta \text{ or } \left(\deg X^\alpha = \deg X^\beta \text{ and } X_1^{\alpha_1} \cdots X_n^{\alpha_n} \prec X_1^{\beta_1} \cdots X_n^{\beta_n}\right).$$

Lemma 2.15. *Let $I = \langle f_1, \dots, f_r \rangle \subseteq \mathbb{K}[\underline{X}]$ be an ideal and $J = \langle {}^{\mathrm{h}}f_1, \dots, {}^{\mathrm{h}}f_r \rangle \subseteq \mathbb{K}[X_0, \dots, X_n]$. If $g_1, \dots, g_s$ is a homogeneous Gröbner basis of J with respect to $\prec_{\mathrm{h}}$, then ${}^{\mathrm{a}}g_1, \dots, {}^{\mathrm{a}}g_s$ is a Gröbner basis of I.*

This is not a difficult result, but it requires additional concepts from commutative algebra, see for example [37, Lemma 2.5] or [34, Lemma 7]. This shows that upper bound on the degree of Gröbner bases elements of homogeneous ideals (in $n+1$ variables) automatically yields bounds on the degree of Gröbner bases of arbitrary ideals (in n variables). Some conditional results in this direction were obtained by Möller & Mora [37], but the most influential result

is the unconditional degree bound by Dubé:

Theorem 2.16 (Dubé 1990 [14]). *Let $I = \langle f_1, \dots, f_s \rangle \subseteq \mathbb{K}[X_1, \dots, X_n]$ be an ideal and $d = \max_i \deg f_i$. Then the reduced Gröbner basis of I consists of polynomials g_i of degree*

$$\deg g_i \leq 2\left(\frac{d^2}{2} + d\right)^{2^{n-1}}.$$

If the f_i are homogeneous, then the exponent can be improved to 2^{n-2}.

The proof uses the Hilbert function of a homogeneous ideal and cone decompositions, for an overview see the paper by Mayr & Ritscher [34], who also prove dimension-dependent upper bounds.

Theorem 2.17 (Mayr & Ritscher 2013 [34]). *Let $\mathbb{K}$ be an infinite field. Let $I = \langle f_1, \dots, f_s \rangle \subsetneq \mathbb{K}[\underline{X}]$ be an ideal of dimension r generated by homogeneous f_i of degrees $d_1 \geq \cdots \geq d_s$. Then the reduced Gröbner basis of I consists of polynomials g_i of degree*

$$\deg g_i \leq 2\left(\frac{1}{2} d_1 \cdots d_{n-r} + d_1\right)^{2^{r-1}}$$

If the polynomials are not necessarily homogeneous, then a similar bound with exponent 2^r can be achieved. The degree bound by Dubé will enable us to prove an exponential space upper bound on the complexity of computing Gröbner bases.

2.4 From polynomials to linear algebra

We continue to use notation from the previous section with $f, f_1, \dots, f_s$ of degrees $d', d_1, \dots, d_s$, $d := \max_i d_i$ in n variables. Furthermore, let $\prec$ be a monomial order, given by a weight matrix $W \in \mathrm{Mat}(n \times n, \mathbb{Q}_{\geq 0})$ as in section 1.7. We will now describe how to translate the following problems into problems of linear algebra:

(i) Decide whether $f = h_1 f_1 + \cdots + h_s f_s$ for suitable $h_i \in \mathbb{K}[\underline{X}]$.

(ii) Calculate $\mathrm{NF}_I(f)$, where $I = \langle f_1, \dots, f_s \rangle$ with respect to $\prec$.

In order to obtain concrete upper bounds on the space required by a Turing machine to solve these problems, we work over the field of rational numbers $\mathbb{K} = \mathbb{Q}$ from here on until the end of this chapter. The input to these algorithms consists of the sequence of polynomials $f, f_1, \dots, f_s$ and in second case the weight matrix W. The rational numbers and the exponents of the monomials are encoded in binary (or fractions of binary numbers). Let ℓ be the length of the input, then we have the following estimates

$$d', d \leq 2^\ell, \qquad n, s \leq \ell. \tag{2.4}$$

We first consider the case of ideal membership. Let D be a estimate of the degree of a set of solution $\deg h_i \leq D$, for example the Hermann bound $D = d' + (sd)^{2^n}$. We use the notation f_α to indicate the coefficient of X^α in f. Then the solution of (i) depends on the finitely many coefficients in

$$h_i = \sum_{\alpha \in \mathbb{N}^n,\, |\alpha| \leq D} h_{i,\alpha} X^\alpha, \qquad h_{i,\alpha} \in \mathbb{K}.$$

A standard counting argument shows that the number of monomials of degree $\leq D$ in n variables is $\binom{n+D}{n} \leq (D+1)^n$. Finding a solution to (i) is then equivalent to solving the following enormous system of *linear* equations:

- **Variables:** $q \coloneqq s \cdot \binom{n+D}{n}$ unknows $h_{i,\alpha}$, indexed by $\{1, \dots, s\} \times \{\, \alpha \in \mathbb{N}^n \mid |\alpha| \leq D \,\}$.
- **Equations:** $r \coloneqq \binom{n+D'}{n}$ equations where $D' = d + D$, corresponding to the coefficient of X^α in (i)

$$f_\alpha = \sum_{\beta \in \mathbb{N}^n, \beta \leq \alpha} \sum_{i=1}^{s} f_{i,\alpha-\beta} h_{i,\beta}, \qquad \alpha \in \mathbb{N}^n,\ |\alpha| \leq D'.$$

By construction the ideal membership problem is equivalent to solvability of this system of linear equations. In order to represent this as a matrix, we choose an ordering of the sets M, M' of monomials of degree $\leq D$ and $\leq D'$ respectively, for example $\prec_{\text{lex}}$. Then it makes sense to talk about the i-th monomial in each set, and the system of equations can be written as

$$\mathcal{A}H = \mathcal{B}, \qquad \mathcal{A} \in \mathrm{Mat}(r \times q, \mathbb{Q}),\ H \in \mathbb{Q}^q,\ \mathcal{B} \in \mathbb{Q}^r.$$

In this presentation we have:

- $\mathcal{B}_i$ is the coefficient of the i-th monomial in f (which is zero if $i > d'$).
- Write $j = k|M| + k' - 1$, $1 \leq k' \leq |M|$, then H_j is the coefficient of the k'-th monomial in h_k.
- $\mathcal{A}_{i,j}$ consists of the coefficients of the f_k as follows: Let X^α be the i-th monomial X^β be the l-th monomial in f_k (as in the previous case), then $\mathcal{A}_{i,j} = f_{\alpha-\beta}$.

In the language of linear algebra, the equation $\mathcal{A}H = \mathcal{B}$ has a solution H if and only if $\mathcal{B}$ is in the linear span of the columns of $\mathcal{A}$, or equivalently if the matrix $[\mathcal{A}|\mathcal{B}]$ has the same rank as $\mathcal{A}$. This reduces the ideal membership to the task of computing a rank of a matrix. The size of this matrix can be estimated in the input length as follows:

- The entries of $\mathcal{A}$ and $\mathcal{A}'$ are coefficients of $f, f_1, \dots, f_s$, and hence of size $\leq \ell$.
- The degree D is bounded by

$$D = d' + (sd)^{2^n} \overset{(2.4)}{\leq} \ell + (\ell 2^\ell)^{2^\ell} \leq 2^{2^{3\ell}}$$

using the generous estimate $\ell \le 2^\ell$.

- The number of equations r is bounded by

$$r = \binom{D'+n}{n} \le (D+d+1)^n \overset{(2.4)}{\le} (2^{2^{3\ell}} + 2^\ell + 1)^\ell \le 2^{2^{5\ell}}.$$

- Similarly, the number of variables q is bounded by

$$q = s\binom{D+n}{n} \le s(D+1)^n \overset{(2.4)}{\le} \ell(2^{2^{3\ell}} + 1)^\ell \le 2^{2^{5\ell}}.$$

Thus we have proven

Theorem 2.18. *The ideal membership problem* $\mathsf{IM}_{\mathbb{Q}}$ *can be reduced to the rank computation of two matrices of size* $2^{2^{5\ell}} \times 2^{2^{5\ell}}$ *whose entries are coefficients of the input. Furthermore, the polynomials* h_i *can be obtained as solutions to a system of linear equations of the same characteristics.*

We now turn to the calculation of the normal form. Here a similar approach may be used, namely we have the polynomial equation

$$h_0 = \mathrm{NF}_I(f) = f + \sum_{i=1}^{s} f_i h_i \tag{2.5}$$

and we are looking for a solution with $\mathrm{supp}(h_0)$ *minimal* (Lemma 1.23). In order to do this we need an estimate on $\deg \mathrm{NF}_I(f)$. Let

$$A = \max\{\, a_{k,i}, b_{k,i} \mid \tfrac{a_{k,i}}{b_{k,i}} \text{ in } W \,\}$$

be the largest natural number occurring in the weight matrix W and let G be a bound on the degree of the reduced Gröbner basis of I. Kühnle & Mayr [30, Section 2], expanding on [13], proved the following upper bound:

Theorem 2.19. *For any* $f \in \mathbb{K}[\underline{X}]$ *we have*

$$\deg \mathrm{NF}_I(f) \le \big((nG)^n A^{2n} \deg(f)\big)^{n+1} =: N.$$

Remark. If a specific monomial order $\prec$ is considered, then better bounds are available. For example, if $\prec$ is degree-dominated, i. e. $\deg X^\alpha > \deg X^\beta$ implies $X^\alpha \succ X^\beta$, then of course $\mathrm{mdeg}\,\mathrm{NF}_I(f) \le \mathrm{mdeg}\, f$ implies $\deg \mathrm{NF}_I(f) \le \deg f$.

We applying the Hermann bound to $f - \mathrm{NF}_I(f) = \sum_{i=1}^s h_i f_i$ to obtain

$$\deg h_i \le \deg(f - \deg h_0) + (sd)^{2^n} \le N + (sd)^{2^n} =: \tilde{D}.$$

We use a strategy similar to the ideal membership case to turn (2.5) into a system of linear

equations, for additional details see [43, Chapter 6]. We introduce an extra factor $f_0 = 1$ to recreate the shape of the ideal membership equation

$$f_0 \cdot h_0 + f_1 h_1 + \cdots + f_s h_s = f \qquad \rightsquigarrow \qquad \left[\mathcal{A} \middle| E\right] \cdot \begin{bmatrix} H \\ H_0 \end{bmatrix} = \mathcal{B}.$$

This time H contains the coefficients of $h_1, \ldots, h_s$, H_0 contains the coefficients of h_0, $\mathcal{A}$ contains the coefficients of $f_1, \ldots, f_s$, E contains 0's and 1's for the coefficients of f_0 and $\mathcal{B}$ contains the coefficients of f.

We now sketch how this system can be used to find the normal form $f^* = \mathrm{NF}_I(f)$. We know that there is *some* solution (for example $h_0 = f$) when all monomials in h_0 are allowed. The normal form is characterized as the unique polynomial $f^* \in f + I$ with minimal support $\mathrm{supp}(f^*)$ with respect to $\prec_{\mathfrak{P}}$ (see Lemma 1.23). The key idea is to systematically remove monomials from h_0 together with the corresponding row from $\mathcal{A}$ and see whether the system still has a solution. We order the columns of $[A|E]$ so that the columns in E are in ascending order with respect to the monomials in h_0, i. e. the lowest monomials $1, X_n, \ldots$ come first. Then we have a maximal regular minor (an invertible square sub-matrix of maximal size) corresponding to the minimal solution h_0. One can decide whether a given column indexed by k is contained in this minor (i. e. if the corresponding monomial X^α is in $\mathrm{supp}(f^*)$) by comparing the rank of the minor of the first k rows/columns to the rank of the first $k-1$ rows/columns: If the rank is different, then X^α is indispensable, otherwise it is not in the support.

The maximal degree of a monomial in (2.5) is bounded by $\max\{N, \tilde{D} + d\} = \tilde{D} + d$, so the number of equations is bounded by $(\tilde{D} + d + 1)^n$. Using similar estimates as before, we obtain the following result.

Theorem 2.20. *The computation of (elements of) the support* $\mathrm{supp}\,\mathrm{NF}_{\langle f_1,\ldots,f_s\rangle}(f)$ *can be reduced to rank computations of matrices of size* $2^{2^{\mathcal{O}(\ell)}} \times 2^{2^{\mathcal{O}(\ell)}}$ *whose entries are coefficients of the input (or* $0, 1$*). Furthermore, the normal form itself can be obtained as solutions to a system of linear equations of the same characteristics.*

We note here that this is true even if the degree estimate for $d' = \deg f$ is not 2^ℓ but rather $2^{2^{\mathcal{O}(\ell)}}$, this will become important when enumerating a Gröbner basis in algorithm 6.

2.5 Fast linear algebra on PRAMs

In this section we briefly introduce a the PRAM model for parallel computation, for details and a modern treatment see for example the book by Parhami [40]. Since we will only use abstract results about the complexity on PRAMs, only an overview of the model is given.

A PRAM (parallel random access machine) consists of

- a set of global registers, containing arbitrary integers,
- a number of processors $P_0, \dots, P_{p(n)-1}$ (depending on the input size), each with their own local registers,
- a finite program consisting of arithmetic operations, branching and read/write access to local or global memory.

Each processor runs on the same program with the knowledge of its id $0 \leq i < p(n)$ and the number $p(n)$. The input is contained in the global register, for example if the input is a matrix $A \in \mathrm{Mat}(n \times n, \mathbb{Z})$, then the global registers contain the n^2 coefficients. The execution is performed synchronously, and read/write conflicts are handled "properly".

The efficiency of an algorithm is expressed both in the number $p(n)$ of processors used, as well as the number of steps $t(n)$ taken until each processor halts. A central result relating the PRAM model to (deterministic) Turing machines is the *Parallel Computation Thesis*.

Theorem 2.21 (Fortune & Wyllie 1978 [17]). *Let $t(n) \geq \log(n)$.*

(i) *If $L \in \mathsf{SPACE}(t(n))$, then L is accepted by a PRAM in parallel time $\mathcal{O}(t(n))$.*

(ii) *If L is accepted by a PRAM in parallel time $t(n)$, then $L \in \mathsf{SPACE}(t(n)^2)$.*

In particular, parallel time complexity is polynomially related to sequential space complexity.

Thus, in order to obtain space-efficient algorithms for problems such as rank computation, we can employ efficient parallel algorithms. Of particular relevance are the following results due to Csanky.

Theorem 2.22 (Csanky 1976 [11]). *Given an integer $n \times n$ matrix, the tasks of*

(i) *matrix inversion*

(ii) *solving a system of linear equations*

(iii) *computing the determinant*

(iv) *computing the coefficients of the characteristic polynomial*

can be solved in parallel time $\mathcal{O}(\log^2(n))$ using $n^{\mathcal{O}(1)}$ processors.

Ibarra, Moran & Rosier extended this list to include matrix rank calculation.

Theorem 2.23 (Ibarra, Moran & Rosier 1980 [22]). *The problem of calculating the rank of a $n \times n$ matrix can be solved in parallel time $\mathcal{O}(\log^2 n)$ using $\mathcal{O}(n^4)$ processors.*

Remark. The papers cited here use the arithmetic machine model, i. e. directly manipulating integers (or rational numbers). Some care has to be taken in order to translate these results to the bit model, Pan describes some appropriate methods [39].

2.6 Upper bounds on ideal membership

We finally apply the space-efficient linear algorithms from the previous section to the ideal membership and normal form problems.

Theorem 2.24 (Mayr 1989 [31], $\mathsf{IM}_{\mathbb{Q}} \in \mathsf{EXPSPACE}$). *There exists a deterministic algorithm which on input $f, f_1, \ldots, f_s$ decides whether there exist $h_1, \ldots, h_s$ with $f = \sum_{i=1}^{s} h_i f_i$, using no more than exponential working space.*

Proof sketch. Using polynomial space we can clear denominators of the input polynomials $f, f_1, \ldots, f_s$ and hence may assume that all polynomials have integer coefficients. We first describe a *parallel* algorithm for this task.

Theorem 2.23 describes an algorithm for matrix rank calculation in parallel time $\mathcal{O}(\log n)$, where n is the size of the matrix. The matrices $\mathcal{A}$, $\mathcal{A}'$ from Theorem 2.18 have size $2^{2^{\mathcal{O}(\ell)}}$ and hence there is a parallel algorithm calculating the rank of $\mathcal{A}$ in time $2^{\mathcal{O}(\ell)}$ using $2^{2^{\mathcal{O}(\ell)}}$ processors. Notice that we cannot write down these matrices explicitly (in exponential space), so instead each time this parallel algorithm requests an entry of $\mathcal{A}$ (by indexing it), we calculate it "on the fly" from the input polynomials. The time required to do this is negligible, since the entries are coefficients of the f_i which can be indexed efficiently.

Now apply the parallel computation hypothesis (Theorem 2.21) to turn this exponential parallel time algorithm into an exponential space algorithm. □

With some additional care one can extend this algorithm to output the solution in the same space bounds, for details see Theorem 4 in the work of Mayr [31].

Theorem 2.25. *With notation as in the previous theorem, if a solution exists, then the algorithm can write it to an output tape in exponential work space.*

Notice however that the size of the h_i themselves may be double-exponential by the Hermann bound and the results of chapter 3, hence it is important to distinguish the work tape from the output tape. A similar argument proves the analogous statement for the normal form, see also [43, Theorem 6.2].

Theorem 2.26. *There is an exponential space algorithm which on input $f, f_1, \ldots, f_s$ computes the normal form $\mathrm{NF}_{\langle f_1,\ldots,f_s\rangle}(f)$.*

Again, the size of the normal form may exceed exponential space, so care has to be taken if this algorithm is used as a subroutine.

If the polynomials are of a specific type, then we can use the other bounds from section 2.3 to give better estimates on the degrees of the h_i. This leads to matrices of single exponential space, and we conclude:

Theorem 2.27 ($\mathsf{IM}_{\mathbf{h},\mathbb{Q}}, \mathsf{HNST}_{\mathbb{Q}} \in \mathsf{PSPACE}$). *In the situation of Theorem 2.24, if*

(i) *the polynomials $f_1, \ldots, f_s$ are homogeneous or*

(ii) $f = 1$ *(the "Nullstellensatz" case),*

then the corresponding problem can be solved in polynomial space.

While the homogeneous ideal membership problem $\mathsf{IM}_{h,\mathbb{Q}}$ is indeed hard for PSPACE (and hence complete), for the problem $\mathsf{HNST}_{\mathbb{Q}}$ much better results are available:

Theorem 2.28 (Koiran 1996 [25]). *Under the assumption of the Generalized Riemann Hypothesis (GRH) we have* $\mathsf{HNST}_{\mathbb{Q}} \in \mathsf{RP}^{\mathsf{NP}} \subseteq \mathsf{PH}$*, in fact,* $\mathsf{HNST}_{\mathbb{Q}} \in \mathsf{AM} = \mathsf{BP} \cdot \mathsf{NP}$.

For the definition the Arthur-Merlin class see for example [1, Section 8.2]. Since $\mathsf{HNST}_{\mathbb{Q}}$ is NP-hard, this implies the surprising result that (under the GRH assumption) $\mathsf{P} \neq \mathsf{NP}$ if and only if $\mathsf{HNST}_{\mathbb{Q}} \notin \mathsf{P}$. For details and improvements (weakening the GRH assumption) see the work of Rojas [45]. An overview over known complexity bounds for several variants of ideal membership is presented by Mayr & Toman [35, Table 1].

2.7 Computing a Gröbner basis in EXPSPACE

We finally present an algorithm computing the reduced Gröbner basis of an ideal $I = \langle f_1, \ldots, f_n \rangle$ in exponential space. It is important to distinguish the work tape from the output tape, since the output may consist of double-exponentially many polynomials, each of which might contain double-exponentially many terms. The key to algorithm 6 is to enumerate the possible leading monomials one by one, and then comparing them to their normal forms in a sufficiently space-efficient way.

Algorithm 6 Enumerating the reduced Gröbner basis

Require: $f_1, \ldots, f_s \in \mathbb{K}[X_1, \ldots, X_n]_{\prec}$, W a weight matrix for $\prec$
Ensure: Yields all elements of the reduced Gröbner basis G of $\langle f_1, \ldots, f_s \rangle$

1: $d \leftarrow \max\{\deg f_1, \ldots, \deg f_s\}$
2: $D \leftarrow 2\left(\frac{d^2}{2} + d\right)^{2^{n-1}}$ ▷ **The Dubé bound 2.16**
3: **for all** $\alpha \in \mathbb{N}^n, |\alpha| \leq D$ **do**
4: $\quad m \leftarrow X^\alpha$
5: $\quad$ **if** $m = \mathrm{NF}_{\langle f_1,\ldots,f_s \rangle}(m)$ **then**
6: $\quad\quad$ **continue** in line 3 ▷ **m is not reducible**
7: $\quad$ **end if**
8: $\quad$ **for** $i \in \{ j \mid \alpha_j > 0 \}$ **do**
9: $\quad\quad m' \leftarrow X_1^{\alpha_1} \cdots X_i^{\alpha_i - 1} \cdots X_n^{\alpha_n}$
10: $\quad\quad$ **if** $m' \neq \mathrm{NF}_{\langle f_1,\ldots,f_s \rangle}(m')$ **then**
11: $\quad\quad\quad$ **continue** in line 3 ▷ **m is not minimally reducible**
12: $\quad\quad$ **end if**
13: $\quad$ **end for**
14: $\quad$ **yield** $m - \mathrm{NF}_{\langle f_1,\ldots,f_s \rangle}(m)$ ▷ **m is minimally reducible**
15: **end for**

Theorem 2.29. *Algorithm 6 enumerates the reduced Gröbner basis of* $I = \langle f_1, \dots, f_s \rangle$ *in exponential work space and double-exponential time.*

Proof. **Correctness:** By Theorem 1.32, the reduced Gröbner basis G consists of polynomials of the form $m - \mathrm{NF}_I(m)$ where m is a monomial which is minimally reducible. The Dubé bound gives a upper bound on the degree of the elements of G (Theorem 2.16), i. e. of m. The algorithm iterates through all possible leading terms m (line 3) and checks if m is reducible (line 5-8) and if any divisor $m' \mid m$ is irreducible (line 11-14). It suffices to check this for m' with $\deg m' = \deg m - 1$, so in line 16 the algorithm has verified that m is minimally reducible and it yields the corresponding element.

Space requirement: Let ℓ be the input size, then $n, s \leq \ell$ and $d \leq 2^\ell$. With this we have $D \leq 2^{2^{\mathcal{O}(\ell)}}$, which fits into the working memory of exponential size.

We can enumerate the monomials m of degree $\leq D$ in any order we like. In order to do this in exponential space, use n binary counters $\alpha_1, \dots, \alpha_n$ from 0 to D and then take $m = X_1^{\alpha_1} \cdots X^{\alpha_n}$. This requires $\mathcal{O}(n \log D)$ space.

We use the normal form algorithm from the previous section on input $(m, f_1, \dots, f_s, W)$; this requires exponential space in $(f_1, \dots, f_s, W)$ (!) by the remark after Theorem 2.20. We can not afford to write down the resulting polynomial in its entirety, but we can check (term by term) whether it coincides with m. This technique is applied in line 5 and 10, while in line 14 we can write the result (with an additional minus sign) to the output tape.

Time requirement: The number of configurations of an algorithm working in space $2^{\mathcal{O}(n)}$ is bounded by $2^{2^{\mathcal{O}(n)}}$, and hence its time requirement is at most double-exponential. □

Corollary 2.30 ($\mathsf{GROEBM}_{\mathbb{Q}} \in \mathsf{EXPSPACE}$**).** *There is an exponential space algorithm deciding whether* g *is contained in the reduced Gröbner basis of* $\langle f_1, \dots, f_s \rangle$.

Proof. Indeed, modify algorithm 6 as follows:

- In line 15, instead of writing $g' := m - \mathrm{NF}_{\langle f_1, \dots, f_s \rangle}(m)$ to the output, compare it to g, if they coincide then accept.
- After line 17, reject.

This algorithm has essentially the same space requirements and decides $\mathsf{GROEBM}_{\mathbb{Q}}$. □

Remark. This algorithm is worst-case optimal in the sense that $\mathsf{GROEBM}_{\mathbb{Q}}$ is EXPSPACE-hard and there exist inputs $f_1, \dots, f_s$ such that the corresponding reduced Gröbner basis G indeed has double-exponentially many elements. On the other hand, this algorithm is "uniformly impractical" in the sense that even if G consists of few elements, then algorithm 6 *still* requires exponential space (as D is extremely large, and the normal form algorithm also uses exponential space on all inputs).

Ritscher improved this approach using S-polynomials and by increasing D only as far as necessary to obtain a more space efficient algorithm [43]. For the class of pure binomial

ideals Koppenhagen & Mayr [27] devised an algorithm using combinatorial tools. While this algorithm requires exponential space (this is necessary, see chapter 3), it does not make use of the parallel computation hypothesis.

For low-dimensional ideals better bounds are known.

Theorem 2.31 (Krick & Logar 1991 [29]). *If $I = \left\langle f_1, \ldots, f_s \right\rangle \subseteq \mathbb{K}[\underline{X}]$ is an ideal of dimension ≤ 1, then a Gröbner basis can be calculated in time $2^{\mathcal{O}(n)}$ assuming unit cost arithmetic of $\mathbb{K}$.*

3

Lower bounds

In this chapter we prove an exponential space lower bound for the ideal membership problem and the reduced Gröbner basis membership problem. For this we introduce the notion of Thue systems, which are essentially presentations of semigroups. The key result is that there is a family of commutative Thue systems which can produce words of double exponential length in the size of the presentation. The chain of reductions from a generic EXPSPACE-problem to Gröbner bases is displayed in Figure 3.1. We also introduce the concept of Church-Rosser systems, which are a string-replacement analogue of Gröbner bases. These notions will lead to examples of ideals with Gröbner bases of double-exponential length and degree.

The primary source for most of the chapter is the original work by Mayr & Meyer [33] and Huynh [21].

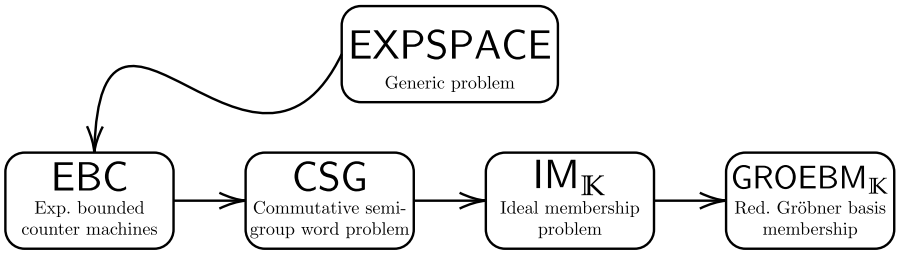


Figure 3.1: The chain of complexity-theoretic reductions.

Remark. In the literature, for example [33] and [32], the type of reduction is not polynomial time many-one reductions. Instead, more restrictive log-lin reductions are used: The reduction function f may use logarithmic space and $|f(x)| \in \mathcal{O}(|x|)$. With this notion, the problems considered here are not complete for $\mathsf{EXPSPACE} = \bigcup_{k\geq 1} \mathsf{SPACE}(2^{\mathcal{O}(n^k)})$, but rather $\mathsf{ESPACE} := \mathsf{SPACE}(2^{\mathcal{O}(n)})$. This has the advantage that if A is a ESPACE-hard problem with respect to log-lin reductions, then there is a $\varepsilon > 0$ such that $A \notin \mathsf{SPACE}(2^{\varepsilon n})$, proving exponential ressource lower bounds [33, Section 2] (this is essentially an application of the space hierarchy theorem [1, Theorem 4.8]).

In this thesis we only consider polynomial time reductions for simplicity, although the interested reader will have no difficulty in verifying that the reductions $\mathsf{EBC} \leq \mathsf{CSG} \leq \mathsf{IM}_{\mathbb{K}} \leq \mathsf{GROEBM}_{\mathbb{K}}$ are actually log-lin reductions.

3.1 Thue systems

We start by introducing a decision problem which is closely related to ideal membership: The word problem for finitely presented commutative semigroups CSG. We present it in the general context of string rewriting systems (SRS), a concept closely related to formal grammars.

Definition 3.1 (String rewriting system, Thue system). A *string rewriting system* or *semi-Thue system* consists of an alphabet Σ together with a finite set of production rules $\mathcal{P} \subseteq \Sigma^* \times \Sigma^*$, written as

$$\mathcal{P} = \{\alpha_1 \to \beta_1, \dots, \alpha_s \to \beta_s\}.$$

A *Thue system* is semi-Thue system such that if $\mathcal{P}$ includes the rule $\alpha \to \beta$, then $\mathcal{P}$ also includes the reverse rule $\beta \to \alpha$. A production $(\alpha \to \beta) \in \mathcal{P}$ can be applied to two strings $x, y \in \Sigma^*$ if there is a decomposition

$$x = \gamma\alpha\delta, \quad y = \gamma\beta\delta, \qquad \gamma, \delta \in \Sigma^*,$$

in symbols $x \Rightarrow_{\mathcal{P}} y$. Let $\Rightarrow^*_{\mathcal{P}}$ be the reflexive and transitive hull of $\Rightarrow_{\mathcal{P}}$, i. e. $x \Rightarrow^*_{\mathcal{P}} y$ if and only if x can be rewritten into y using a finite sequence of rules from $\mathcal{P}$. Two Thue systems $(\Sigma, \mathcal{P})$, $(\Sigma, \mathcal{P}')$ are *equivalent* if $\Rightarrow^*_{\mathcal{P}}$ and $\Rightarrow^*_{\mathcal{P}'}$ coincide. ⌟

If $\mathcal{P}$ is a Thue system, then this is an equivalence relation, and even a congruence relation (see Definition A.4, in essence this means that the set of equivalence classes again form a monoid). Hence we get a monoid $\Sigma^*/(\Rightarrow^*_{\mathcal{P}})$ generated by Σ where two strings coincide if and only if they can be derived from each other. The equivalence class of x is denoted as $[x]_{\mathcal{P}}$

Example 3.2. In order to indicate how expressive SRS are, we sketch how to simulate any Turing machine using string rewriting rules, for details see the original paper by Post [42]. Consider a 1-tape Turing machine M with states $Z = \{z_1, \dots, z_n\}$ over $\{0, 1, \square\}$. We can encode a configuration of the machine as a string over $Z \cup \{0, 1, \square, \triangle\}$ as

$$\triangle a_{-l} a_{-l+1} \dots a_{-1} z a_0 a_1 \dots a_r \triangle, \qquad a_{-l}, \dots, a_r \in \{0, 1, \square\},\ z \in Z,$$

where z is the current state, the head is on tape symbol t_0 and $\triangle$ is the first tape cell which has not been visited yet in each direction. If $\delta_M(z, a) = (z', a', \mathrm{L})$, then this transition can be described by the four production rules

$$bza \to z'ba' \ (b \in \{0, 1, \square\}), \quad \triangle za \to \triangle z'\square a',$$

and similarly for a move to the right. In this way one can construct a string rewriting system which mimics the behavior of M in a precise way when starting with the string $\triangle z_0 x \triangle$. This easily implies that the statement $x \Rightarrow^*_{\mathcal{P}} y$ for given $(\mathcal{P}, x, y)$ is undecidable; even for fixed $\mathcal{P}$ if a universal Turing machine M is used. It turns out that if M is deterministic, then this

translation still works if the reverse rules are included, so string equivalence with respect to a Thue system is also undecidable (in fact RE-complete). ⌟

In this sense Thue systems are too powerful to be useful as a complexity-theoretic lower bound, since Gröbner bases and ideal membership are computable (even in EXPSPACE). But if we move from strings (with non-commuting letters) to *commutative* strings, then the corresponding Thue systems do become decidable.

Definition 3.3 ($\Sigma^\oplus$, Commutative Thue system). Let Σ be a finite alphabet, then $\Sigma^\oplus$ is the *free commutative monoid* over Σ

$$\Sigma^\oplus := \mathrm{Mon}(\Sigma) \cong \Sigma^*/\!\sim, \quad u \sim v \text{ if and only if } |u|_x = |v|_x \ \forall x \in \Sigma.$$

A *commutative Thue system* is a Thue system $(\Sigma, \mathcal{P})$ defined over commutative strings $\Sigma^\oplus$.
Convention: From now on only commutative Thue systems will be considered. When specifying $\mathcal{P}$ as a list of rules of the form

$$\alpha \to \beta \tag{$\heartsuit$}$$

we implicitly always add the converse rules, too. We write $\gamma \overset{(\heartsuit)}{\Rightarrow} \gamma'$ for a forward application of the rule and $\gamma \overset{(\heartsuit)}{\Leftarrow} \gamma'$ for an application of the reverse rule, but we stress that $\Rightarrow_\mathcal{P}$ is a *symmetric* relation. If the concrete derivation is not important, then we simply write $\alpha \equiv_\mathcal{P} \beta$. If the Thue system is understood from the context, then we omit the subscript. ⌟

Remark. In the literature commutative Thue systems are defined as "ordinary" Thue systems containing the rules $xy \to yx$ for all $x, y \in \Sigma$, which is of course equivalent to our definition.

Commutative Thue systems can be understood as "monomial replacement systems"; this insight is a key ingredient in the reduction to the ideal membership problem. They also coincide with finite presentations of commutative semigroups in the following sense:

Example 3.4 *(Commutative Thue systems "are" finite commutative monoid presentations).* If $\mathcal{P}$ is a commutative Thue system over Σ, then $M := \Sigma^\oplus/(\Rightarrow_\mathcal{P}^*)$ is a monoid generated by the classes of elements of Σ, and commutativity of the strings ensures that M is commutative. This commutative monoid coincides with the finitely presented monoid $\langle \Sigma \mid \mathcal{P} \rangle$, since the relation $\Rightarrow_\mathcal{P}^*$ is the congruence relation generated by $\mathcal{P}$ (Lemma A.6).

Conversely, if $\langle \Sigma \mid \mathcal{R} \rangle$ is a finitely presented monoid, then by adding the reverse rules we get a commutative Thue system $\mathcal{P}$ such that $\Sigma^\oplus/(\Rightarrow_\mathcal{P}^*) \cong \langle \Sigma \mid \mathcal{R} \rangle$. ⌟

We finally define the word problem for finitely presented commutative semigroups.

Definition 3.5 (Word problem for commutative semigroups, CSG).

- *Input:* $(\Sigma, \mathcal{R}, \alpha, \beta)$, where Σ is a finite set of symbols, $\mathcal{R}$ a list of generating congruences $\{(\alpha_i, \beta_i)\}_{i=1}^s \subseteq \Sigma^\oplus \times \Sigma^\oplus$ and $\alpha, \beta \in \Sigma^\oplus$.
- *Output:* Decide whether $\alpha \equiv_\mathcal{R} \beta$ in the commutative semigroup $\langle \Sigma \mid \mathcal{R} \rangle$.

If the input is restricted to congruences where both sides have the same length, then the corresponding problem is denoted by CSG_h ⌟

In other words, we want to decide whether $\alpha \Rightarrow^*_{\mathcal{P}} \beta$, where $\mathcal{P}$ is the commutative Thue system obtained from $\mathcal{R}$.

A first hardness result is the following construction similar to example 3.2, appearing in the survey of Mayr [32, Theorem 17]. Let M be a deterministic single-tape Turing machine working in space $f(n)$, so the following portion of the initial tape content

$$\underbrace{\square \dots \square}_{f(n)} x_1 x_2 \dots x_n \underbrace{\square \dots \square}_{f(n)}$$

of size $N \coloneqq |x| + 2f(|x|)$ contains all cells ever visited in the course of computation of M on x. We can index the cells by $1, \dots, N$, let

$$\Sigma_{M,n} \coloneqq Q \cup \left\{ s_{b,i} \;\middle|\; b \in \{\square, 0, 1\},\ i = 1, \dots, N \right\} \cup \left\{ p_i \;\middle|\; i = 1, \dots, N \right\}$$

then a configuration of M in state z on position j with symbol b_i in tape cell i can be represented as a string of length $N + 2$ over $\Sigma_{M,n}$ as

$$z p s_1 \dots s_N \in \Sigma^{\oplus}_{M,n}, \qquad z \in Z,\ p \in \{p_1, \dots, p_N\}\ s_i = s_{b_i,i},\ i = 1, \dots, N. \tag{3.1}$$

Intuitively, we create N copies for each symbol in the working alphabet of M so that the presence of the i-th copy indicates that the corresponding symbol is currently in cell i, and p_j indicates that the head is on cell j. Let α_x be the encoding of the initial configuration of M on input x and $\beta_x = z_\mathrm{a} p_1 s_{\square,1} \dots s_{\square,N}$, where z_a is the unique accepting state of M.

Lemma 3.6. *With the preceding notation there is a commutative Thue system $\mathcal{P}_{M,n}$ over $\Sigma_{M,n}$ such that $\alpha_x \Rightarrow^*_{\mathcal{P}_{M,n}} \beta_x$ if and only if $x \in L(M)$.*

Proof. We define $\mathcal{P}_{M,n}$ to contain the following rules:

(i) If $\delta_M(z, a) = (z', a', \mathrm{L})$, then add $z p_i s_{a,i} \to z' p_{i-1} s_{a',i}$ for $i = 2, \dots, N$.

(ii) If $\delta_M(z, a) = (z', a', \mathrm{R})$, then add $z p_i s_{a,i} \to z' p_{i+1} s_{a',i}$ for $i = 1, \dots, N-1$.

(iii) Add $z_\mathrm{a} s_{a,i} \to z_\mathrm{a} s_{\square,i}$ and $z_\mathrm{a} p_i \to z_\mathrm{a} p_1$ for all $i = 1, \dots, N$ and $a \in \{0, 1\}$.

It is clear that any string $\alpha_x \Rightarrow^*_{\mathcal{P}_{M,n}} w$ is of the form (3.1) since all rules preserve the structure. Furthermore, an application of the rules $\mathcal{M} \coloneqq$ (i) $\cup$ (ii) corresponds to a (valid) transition of a configuration of M into the next one.

Assume $x \in L(M)$, then after a finite number of steps the machine M reaches a configuration in state z_a, so $\alpha_x \Rightarrow^*_{\mathcal{M}} z_\mathrm{a} p_j s_1 \dots s_N$. Using the rules from (iii) we can "clean up" the string to

derive β_x as desired:

$$z_a p_j s_1 \dots s_N \Rightarrow z_a p_1 s_1 \dots s_N \Rightarrow z_a p_1 s_{\square,1} \dots s_N \Rightarrow \dots \Rightarrow z_a p_1 s_{\square,1} \dots s_{\square,N} = \beta_x.$$

Conversely, assume $\alpha_x \Rightarrow^*_{\mathcal{P}_{M,n}} \beta_x$ and consider a repetition-free derivation

$$\alpha_x = \gamma_0 \Rightarrow \gamma_1 \Rightarrow \dots \Rightarrow \gamma_m = \beta_x.$$

As β_x contains z_a, there is a minimal $m_0 \geq 0$ such that γ_{m_0} contains z_a.
Claim: In this derivation $\gamma_0 \Rightarrow^*_{\mathcal{P}_n} \gamma_{m_0}$ only (forward) rules from $\mathcal{M}$ are applied.

Indeed, rules from (iii) and their converse are not applicable by minimality of m_0. Assume that a converse rule from $\mathcal{M}$ is used in some $\gamma_{k-1} \Rightarrow_{\mathcal{P}} \gamma_k$ and choose k maximal with this property. $\gamma_k \neq \gamma_{m_0}$, as z_a does not occur on any left hand side in $\mathcal{M}$ (z_a is a final state), so $\gamma_k \Rightarrow_{\mathcal{P}} \gamma_{k+1}$ applies a forward rule from $\mathcal{M}$ by maximality of k. We are in the following situation:

$$\gamma_{k-1} \Leftarrow_{\mathcal{M}} \gamma_k \text{ using } z'' p_h s_{a'',i} \leftarrow z p_i s_{a,i}, \qquad \gamma_k \Rightarrow_{\mathcal{M}} \gamma_{k+1} \text{ using } z p_i s_{a,i} \rightarrow z' p_j s_{a',i}.$$

For any configuration there is (at most) a single applicable rule from $\mathcal{M}$ corresponding to the deterministic transition $\delta_M(z, a)$. In this situation the rule is determied by γ_k as both share z, p_i and therefore $s_{a,i}$, so *the same production* is applied and the reverse and forward application cancel each other. Thus $\gamma_{k-1} = \gamma_{k+1}$, but the derivation was assumed to be repetition-free, so no reverse rule could have been applied in the first place, proving the claim.

We have established that $\alpha_x = \gamma_0 \Rightarrow^*_{\mathcal{P}_{M,n}} \gamma_{m_0}$ consists of forward rules from $\mathcal{M}$ corresponding to the behavior of M on x. Since γ_{m_0} contains the accepting state, we see that $x \in L(M)$ and the proof is complete. □

If the function f is bounded by a polynomial, then this yields a polynomial-time reduction from $L(M)$ to the word problem for commutative semigroups.

Theorem 3.7 ($\mathsf{CSG_h}$ is PSPACE-hard). *If $A \in$ PSPACE, then $A \leq^{\mathrm{P}}_{\mathrm{m}} \mathsf{CSG_h}$, in particular* CSG *and* $\mathsf{CSG_h}$ *are* PSPACE*-hard.*

Proof. Let $A \in$ PSPACE be a language over $\{0, 1\}$ and let M be a deterministic 1-tape Turing machine deciding A in space bounded by $f(n) = c \cdot n^k$. Let $(\Sigma_{M,n}, \mathcal{P}_{M,n})$ be the commutative presentation from the previous lemma (i. e. containing the rules (i)–(iii)) for $N = f(n) = n + 2cn^k$. Notice that by construction, all rule are homogeneous. The map

$$f(x) \coloneqq (\Sigma_{M,|x|}, \mathcal{P}_{M,|x|}, \alpha_x, \beta_x)$$

can be computed in deterministic polynomial time and reduces A to $\mathsf{CSG_h}$ by Lemma 3.6. □

It turns out that CSG is *not* in PSPACE, in contrast to CSG_h, but rather EXPSPACE-hard. Our approach here does not generalize to this case, as we need a variable for each tape cell and such an alphabet cannot be written down by a polynomial-time algorithm. On the other hand, non-homogeneous rules allow the derivation strings of superpolynomial length (compared to the input string and the set of rules), see section 3.4. Thus our next step is to find a model of computation whose configurations can be encoded with long strings from $\Sigma^\oplus$ for *small* Σ.

3.2 Counter machines

It turns out that a convenient model of computation to be simulated by commutative semigroup presentations are *Counter machines*. A classical reference is Minsky's book [36, Chapter 11 and 14], where these machines are called *program machines* (and use a slightly different instruction set). Informally, these are finite automata together with a fixed number of counters. The counters can be INCreased, DECreased or used to Branch the program flow on a Zero. More formally:

Definition 3.8 (Counter machine). A *counter machine* with $k \geq 1$ counters is a tuple $C = (Q, \delta, q_0, q_\mathrm{a})$ where

- Q is a finite set of states
- δ is a transition function

$$\delta \colon Q \setminus \{q_\mathrm{a}\} \to (\{\mathtt{INC}_1, \dots, \mathtt{INC}_k, \mathtt{DEC}_1, \dots, \mathtt{DEC}_k\} \times Q) \cup (\{\mathtt{BZ}_1, \dots, \mathtt{BZ}_k\} \times Q \times Q)$$

- $q_0 \in Q$ is the initial state and $q_\mathrm{a} \in Q$ is the final/accepting state. ⌟

A configuration of C is described by the current state and the content of the k counters, i. e. a tuple in $Q \times \mathbb{Z}^k$. The transition function is used to define the (unique) successive configuration as follows: Let $(q, c_1, \dots, c_k)$ be the current configuration, $q \neq q_\mathrm{a}$.

(i) If $\delta(q) = (\mathtt{op}_j, q')$, $\mathtt{op} \in \{\mathtt{INC}, \mathtt{DEC}\}$ then the next configuration is

$$(q, c_1, \dots, c_k) \vdash_C (q', c'_1, \dots, c'_k), \qquad c'_i = \begin{cases} c_j + 1 & \text{if } i = j \text{ and } \mathtt{op} = \mathtt{INC} \\ c_j - 1 & \text{if } i = j \text{ and } \mathtt{op} = \mathtt{DEC} \\ c_i & \text{if } i \neq j \end{cases}$$

(ii) If $\delta(q) = (\mathtt{BZ}_j, q', q'')$, the next configuration is

$$(q, c_1, \dots, c_k) \vdash_C \begin{cases} (q', c_1, \dots, c_k) & \text{if } c_j = 0 \\ (q'', c_1, \dots, c_k) & \text{if } c_j \neq 0 \end{cases}$$

As usual, we denote the transitive and reflexive closure of the transition relation as $\vdash_C^*$.

Example 3.9 *(Multiplication and division by 2).* In this example we assume that counter 1 holds a non-negative integer and counter 2 is empty.

(i) We can move the content of counter 1 to counter 2 (emptying the former) and transition from state q to q' with the following instructions:

$$q \mapsto (\mathtt{BZ}_1, q', a), \quad a \mapsto (\mathtt{INC}_2, b), \quad b \mapsto (\mathtt{DEC}_1, q).$$

This has the effect of $(q, N, 0) \vdash^* (q', 0, N)$.

(ii) We can similarly double the content of counter 1 if we increase twice instead of once:

$$q \mapsto (\mathtt{BZ}_1, q', c), \quad c \mapsto (\mathtt{INC}_2, d), \quad d \mapsto (\mathtt{INC}_2, e), \quad e \mapsto (\mathtt{DEC}_1, q).$$

This has the effect of $(q, N, 0) \vdash^* (q', 0, 2N)$. Using the copy instructions from (i) we can then move the result back to counter 1.

(iii) Using the same trick we can also halve the first counter and branch into q_{even} or q_{odd} depending on the remainder:

$$q \mapsto (\mathtt{BZ}_1, q_{\text{even}}, f), \quad f \mapsto (\mathtt{DEC}_1, g), \quad g \mapsto (\mathtt{BZ}_1, q_{\text{odd}}, h), \quad h \mapsto (\mathtt{DEC}_1, i), \quad i \mapsto (\mathtt{INC}_2, q).$$

This has the effect of $(q, 2N, 0) \vdash^* (q_{\text{even}}, 0, N)$ and $(q, 2N + 1, 0) \vdash^* (q_{\text{odd}}, 0, N)$ and we can again move the result back if desired. ⌟

We say that the counter machine C accepts an input $N \in \mathbb{Z}$ if and only if

$$(q_0, N, 0, \ldots, 0) \vdash_C^* (q_a, 0, \ldots, 0).$$

Lemma 3.10 (Counter machines are Turing complete). *For any Turing machine M on the input alphabet $\Sigma = \{0, 1\}$ there exists a 3-counter machine C such that M halts on $x \in \{0, 1\}^*$ if and only if C accepts the binary number $N = 1x$.*

A leading 1 is necessary to discriminate different strings that would otherwise describe the same binary number, for example 0 vs 00. The proof uses *stack machine* as an intermediary machine model, for our purposes, this is a finite state machine together with (in our case) two stacks S_1, S_2. Depending on the current state and the top symbols of the stacks (including observing an empty stack #), the machine transitions into a unique successor state and can *push* and *pop*[1] single elements from both stacks:

$$\delta_S \colon Q \times \{0, 1, \#\}^2 \to Q \times (\{S_1, S_2\} \times \{.\text{pop}, .\text{push}(0), .\text{push}(1)\})^*$$

[1] A pop operation on an empty stack has no effect.

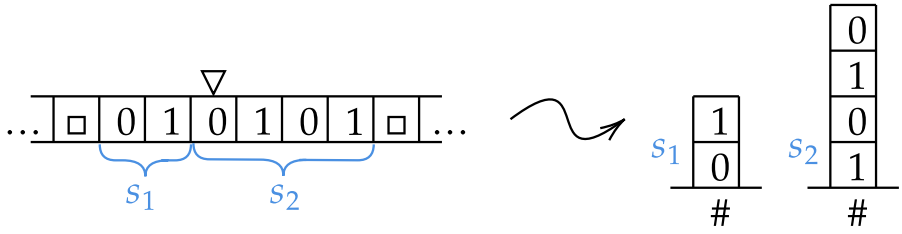


Figure 3.2: How to represent the tape of a Turing machine with two stacks.

Proof of Lemma 3.10. Let $M = (Z, \{0,1\}, \square, \delta_M, z_0, z_a)$ be the Turing machine in question, so $\delta_M : Z \times \{0,1,\square\} \to Z \times \{0,1\} \times \{L,R\}$.

Step 1: The behavior of M can be simulated by two stacks.
We translate M and x into a stack machine as follows: The set of states is $Z \times \{0,1\}$ If the tape content is $\ldots \square a_0 a_1 \ldots a_k \square \ldots$ with head at position i, then S_1 contains the left portion $(a_{i-1}, a_{i-2}, \ldots, a_0, \#)$, S_2 the right portion $(a_i, \ldots, a_k, \#)$, see Figure 3.2. We translate δ_M into δ_S as follows:

$$\begin{aligned} \delta_M(z,i) = (z', i', R) \quad &\rightsquigarrow \quad \delta_S(z,c,i) = (z', S_1.\text{push}(i'); S_2.\text{pop}) \\ \delta_M(z,i) = (z', i', L) \quad &\rightsquigarrow \quad \delta_S(z,c,i) = (z', S_1.\text{pop}; S_2.\text{pop}; S_2.\text{push}(i'); S_2.\text{push}(c)) \end{aligned}$$

(if $c = \#$ then omit the last push). The starting configuration is $S_1 = (\#)$ and $S_2 = (x_1, x_2, \ldots, x_n, \#)$.

Step 2: The behavior of a single stack can be simulated by two counters.
A stack with content $(a_1, \ldots, a_s, \#)$ is represented by a counter c with value $1 a_s a_{s-1} \ldots a_1$ in binary. Pushing a $0/1$ is performed by doubling using a second empty counter c' as in Example 3.9 and adding 1 if desired. Peeking at the stack corresponds to extracting the lowest bit, we implement this as in Example 3.9 together with a check if the counter has value 1 (i. e. represents the empty stack). A pop is performed by checking if the stack is nonempty ($c \neq 1$) and then dividing by 2.

Step 3: Translating M into a counter machine C.
Two stacks can be simulated using three counters if c_1, c_2 hold the actual data and c_3 is used as storage for the stack operations as described in step 2. Thus it is straightforward (although a bit tedious) to translate the stack machine behavior δ_S from step 1 into a transition function for a counter machine. At the start of computation we initialize the second counter (with a single increment) and then "flip" the content of counter c_1 to c_2 in order to have the leftmost bit of the input as the least significant bit of the value of c_2 (compare step 2), then counter c_1 and c_2 precisely represent the initial configuration of the stack machine. Finally we add instructions to empty all counters if the state z_a is reached, and then transition to q_a. □

Remark. In the sequel we only need this result for k-counter machines for *some* (fixed) $k > 0$ (although the construction of $\mathcal{P}_C$ is somewhat simpler if $k \leq 4$, see below), the lemma shows that 3 counters suffice. It turns out that 2-counter machines are already universal, but only if

the input is encoded in a *very* specific way [36, Theorem 14.1-1].

The computation of C on input $N \in \mathbb{N}$ is said to be *bounded by* $n \in \mathbb{N}$ if the counters of *all* intermediate configurations $(z_0, N, 0, \ldots, 0) \vdash_C^* (z, c_1, \ldots, c_k)$ are in the range $0 \leq c_j \leq n$. Notice that the construction in the previous lemma has the following property: If M is operating in space $f(n)$, then the constructed counter machine will have its counters bounded by $2^{f(n)+1}$, as this is a bound on the number $1T$, where T is the (binary) number spelled out on the tape.

Theorem 3.11 (EBC **is** EXPSPACE**-complete).** *The language of* exponentially bounded counter machines EBC *is* EXPSPACE*-complete:*

- *Input:* $C = (Q, \delta_C, q_0, q_a)$*, a 3-counter-machine*
- *Output: Decide whether* C *accepts* 0 *and has computation bounded by* $2^{2^{|Q|}}$

Proof. **Step 1:** EBC is in EXPSPACE.

A Turing machine can easily simulate the configuration transitions. Each counter and the current state is stored as a binary number on a separate tape and the instructions `INC`, `DEC`, `BE` can be executed using only the cells belonging to the counters. If the computation of C is bounded by $2^{2^{|Q|}}$, then the space resources of this Turing machine are bounded by $3 \cdot 2^{|Q|} + |Q|$, which is exponential in the input length of C.

Step 2: $A \leq_{\mathrm{m}}^{\mathrm{P}}$ EBC hard for all $A \in$ EXPSPACE.

Let M be a 1-tape machine deciding A which visiting at most $2^{c|x|^k}$ cells on the working tape on input x for some fixed $c, k > 0$. In order to show $A \leq_{\mathrm{m}}^{\mathrm{P}}$ EBC we use the 3-counter machine constructed in the previous lemma. For $x \in \{0,1\}$ let C_x be the following 3-counter machine:

(i) Starting from q_0 use $O(|x|)$ instructions to count c_1 to the value $1x$ using the binary representation and the doubling algorithm from example 3.9.

(ii) Then add the instructions from Lemma 3.10 to simulate the behavior of M.

(iii) Artificially enlarge the set of states Q (if necessary) in order to ensure $2^{|Q|} \geq 2^{c|x|^k} + 1$.

Then the computation of C_x is bounded by $2^{2^{c|x|^k}+1} \leq 2^{2^{|Q|}}$ and C_x accepts 0 if and only if $x \in L(M)$. Step (ii) only depends on M and not on x, so the map $x \mapsto \langle C_x \rangle$ can be computed in polynomial time. □

Remark. It may seem kind of arbitrary to consider the computation bounded by $2^{2^{|Q|}}$. Indeed, if we require C to have its computation bounded by $2^{|Q|}$ instead, then this problem becomes PSPACE-complete.

3.3 Simulating counter machines with CSG

We now describe how to encode the behavior of a counter machine using commutative Thue systems. Let C be a counter machine whose computation is bounded by the number e, then a configuration of C is encoded over $\Sigma'_C = Q \cup \{A_1, B_1, A_2, B_2, A_3, B_3\}$ as

$$\mathrm{rep}(q, c_1, c_2, c_3) \coloneqq q A_1^{c_1} B_1^{\mathrm{e}-c_1} A_2^{c_2} B_2^{\mathrm{e}-c_2} A_3^{c_3} B_3^{\mathrm{e}-c_3} \in {\Sigma'_C}^{\oplus}. \tag{3.2}$$

This is essentially a unary representation of the counter values. The initial configuration is $\alpha \coloneqq q_0 B_1^{\mathrm{e}} B_2^{\mathrm{e}} B_3^{\mathrm{e}}$ and the terminal configuration is $\beta \coloneqq q_{\mathrm{a}} B_1^{\mathrm{e}} B_2^{\mathrm{e}} B_3^{\mathrm{e}}$. We translate the function δ_C into congruence rules $\mathcal{P}'_C$ as follows:

(INC) If $\delta_C(q) = (\mathtt{INC}_i, q')$, then add the rule $qB_i \to q'A_i$.

(DEC) If $\delta_C(q) = (\mathtt{INC}_i, q')$, then add the rule $qA_i \to q'B_i$.

The i-th counter is zero if and only if the configuration string contains B_i^e and nonzero if and only if it contains A_i. Thus we can define

(BZ) If $\delta_C(q) = (\mathtt{BZ}_i, q', q'')$, then add the rules $qB_i^{\mathrm{e}} \to q'B_i^{\mathrm{e}}$ and $qA_i \to q''A_i$.

Notice that the application of any rule to strings of the form (3.2) yields again a string of this form.

Lemma 3.12. *Let $\Sigma'_C, \mathcal{P}'_C, \alpha, \beta$ be as above, then C accepts 0 and has computation bounded by e if and only if $\alpha \equiv_{\mathcal{P}'_C} \beta$ in the commutative semigroup $\langle \Sigma'_C \mid \mathcal{P}'_C \rangle$.*

Proof. Assume first that C has its computation bounded by e. If $(q, c_1, c_2, c_3) \vdash_C (q, c'_1, c'_2, c'_3)$, then $\mathrm{rep}(q, c_1, c_2, c_3) \Rightarrow_{\mathcal{P}'_C} \mathrm{rep}(q', c'_1, c'_2, c'_3)$, hence by induction we obtain that $(q_0, 0, 0, 0) \vdash_C^* (q_{\mathrm{a}}, 0, 0, 0)$ implies $\alpha \Rightarrow_{\mathcal{P}'_C}^* \beta$.

Conversely assume $\alpha \Rightarrow_{\mathcal{P}'_C}^* \beta$ and consider a repetition-free derivation

$$\alpha = \gamma_0 \Rightarrow_{\mathcal{P}'_C} \gamma_1 \Rightarrow_{\mathcal{P}'_C} \cdots \Rightarrow_{\mathcal{P}'_C} \gamma_m = \beta.$$

The same argument as in the proof of Lemma 3.6 shows that this derivation only uses forward rules for (INC), (DEC), (BZ). This sequence of derivations hence describes the configuration transitions of C and witnesses that C accepts 0 with computation bounded by e. □

Unfortunately, this does not immediately allow us to obtain a polynomial-time reduction EBC $\leq_{\mathrm{m}}^{\mathrm{P}}$ CSG, since α, β and $\mathcal{P}'_C$ would have to contain the expression $B_i^{2^{2^n}}$ where $n \coloneqq |Q|$. We use the following result which will be proved in the next section:

Theorem 3.13 (Mayr & Meyer, 1982 [33]). *For given n there exists a commutative semigroup presentation $\langle \Sigma_n \mid \mathcal{P}_n \rangle$ of length $\mathcal{O}(n)$ with the following properties:*

(i) Σ_n *contains* S, F, C_i, B_i, $i = 1, \dots, 4$ *among other symbols.*

(ii) *The only words containing* S *or* F *and derivable from* $SC_i w$, $w \in \{B_1, \dots, B_4\}^{\oplus}$ *are* $SC_i w$ *and* $FC_i B_i^{2^{2^n}} w$.

(iii) *Similarly, the only words containing* S *or* F *and derivable from* $FC_i w$ *are* $FC_i w$ *and* $SC_i w'$ *with* $w = w' B_i^{2^{2^n}}$.

With this we can "compress" the presentation $\mathcal{P}'_C$ to $(\Sigma_C, \mathcal{P}_C)$ where Let

$$\Sigma_C := \Sigma_n \cup \Sigma'_C \dot\cup \{ q_\uparrow, q_\downarrow \mid q \in Q \text{ with } \delta_C(q) = (\mathtt{BZ}_j, \dots) \} \dot\cup \{q_{0,0}, \dots, q_{0,3}, q_{\mathrm{a},0}, \dots, q_{\mathrm{a},3}\}.$$

Let $\mathrm{e}_n := 2^{2^n}$, then Σ_C contains the following symbols with specific purpose:

- Σ_C contains Σ_n to create the strings of double-exponential length.
  $\rightsquigarrow$ $\mathcal{P}_C$ contains $\mathcal{P}_n$.
- Σ_C contains the symbols from Σ'_C encoding the configurations of C.
  $\rightsquigarrow$ $\mathcal{P}_C$ contains $\mathcal{P}'_C$ *except* for the (BE) rules $qB_i^{\mathrm{e}_n} \to q'B_i^{\mathrm{e}_n}$.
- For each branch-on-zero state $q \in Q$ Σ_C contains two auxiliary "states" whose purpose is to verify that the configuration contains $B_i^{2^{2^n}}$ by breaking down this string ($q_\downarrow$) and then building it back up and transitioning to the successor state ($q_\uparrow$)
  $\rightsquigarrow$ $\mathcal{P}_C$ contains for each $q \in Q$ with $\delta_C(q) = (\mathtt{BE}_i, q', q'')$ the rules

$$q \to q_\downarrow FC_i \qquad (\searrow)$$
$$q_\downarrow SC_i \to q_\uparrow SC_i \qquad (\curvearrowright)$$
$$q_\uparrow FC_i \to q'. \qquad (\nearrow)$$

- Finally, Σ_C contains $q_{0,0}, \dots, q_{\mathrm{a},3}$ to expand the initial and collaps the final configuration.
  $\rightsquigarrow$ $\mathcal{P}_C$ contains the rules

$$q_{0,0} \to q_{0,1} SC_1 \qquad (0_0)$$
$$q_{0,1} FC_1 \to q_{0,2} SC_2 \qquad (0_1)$$
$$q_{0,2} FC_2 \to q_{0,3} SC_3 \qquad (0_2)$$
$$q_{0,3} FC_3 \to q_0 \qquad (0_3)$$
$$q_{\mathrm{a}} \to q_{\mathrm{a},3} FC_3 \qquad (\mathrm{a}_3)$$
$$q_{\mathrm{a},3} SC_3 \to q_{\mathrm{a},2} FC_2 \qquad (\mathrm{a}_2)$$
$$q_{\mathrm{a},2} SC_2 \to q_{\mathrm{a},1} FC_1 \qquad (\mathrm{a}_1)$$
$$q_{\mathrm{a},1} SC_1 \to q_{\mathrm{a},0}. \qquad (\mathrm{a}_0)$$

We first note that this new presentation $\mathcal{P}_C$ still behaves as $\mathcal{P}'_C$.

Lemma 3.14. *We have the following equivalences:*

(i) $q_{0,0} \equiv_{\mathcal{P}_C} q_0 B_1^{\mathrm{e}_n} B_2^{\mathrm{e}_n} B_3^{\mathrm{e}_n} = \mathrm{rep}(q_0, 0, 0, 0)$ *and* $q_{\mathrm{a},0} \equiv_{\mathcal{P}_C} q_{\mathrm{a}} B_1^{\mathrm{e}_n} B_2^{\mathrm{e}_n} B_3^{\mathrm{e}_n} = \mathrm{rep}(q_{\mathrm{a}}, 0, 0, 0)$.

(ii) *If* $\mathrm{rep}(q, c_1, c_2, c_3) \equiv_{\mathcal{P}'_C} \mathrm{rep}(q', c'_1, c'_2, c'_3)$, *then* $\mathrm{rep}(q, c_1, c_2, c_3) \equiv_{\mathcal{P}_C} \mathrm{rep}(q', c'_1, c'_2, c'_3)$.

Proof. (i) This is a consequence of (0_0)–(0_3) and Theorem 3.13:

$$q_{0,0} \overset{(0_0)}{\Rightarrow} q_{0,1} S C_1 \overset{(3.13)}{\Rightarrow^*} q_{0,1} F C_1 B_1^{\mathrm{e}_n} \overset{(0_1)}{\Rightarrow} q_{0,2} S C_2 B_1^{\mathrm{e}_n} \overset{(3.13)}{\Rightarrow^*} q_{0,2} F C_2 B_1^{\mathrm{e}_n} B_2^{\mathrm{e}_n}$$
$$\overset{(0_2)}{\Rightarrow} q_{0,3} S C_2 B_1^{\mathrm{e}_n} B_2^{\mathrm{e}_n} \overset{(3.13)}{\Rightarrow^*} q_{0,3} F C_3 B_1^{\mathrm{e}_n} B_2^{\mathrm{e}_n} B_3^{\mathrm{e}_n} \overset{(0_3)}{\Rightarrow} q_0 B_1^{\mathrm{e}_n} B_2^{\mathrm{e}_n} B_3^{\mathrm{e}_n}$$

The second derivation is completely analogous with (a_0)–(a_3) instead.

(ii) $\mathcal{P}_C$ inherits all rules from $\mathcal{P}_C$ except for $q B_i^{\mathrm{e}_n} \to q' B_i^{\mathrm{e}_n}$. If this rules is applied in a derivation (say with $i = 1$) to $\mathrm{rep}(q, \mathrm{e}_n, c_2, c_3) \Rightarrow_{\mathcal{P}'_C} \mathrm{rep}(q', \mathrm{e}_n, c_2, c_3)$, then we have

$$\mathrm{rep}(q, \mathrm{e}_n, c_2, c_3) = q B_1^{\mathrm{e}_n} \gamma \overset{(\searrow)}{\Rightarrow} q_\downarrow F C_1 B_1^{\mathrm{e}_n} \gamma \overset{(3.13)}{\Rightarrow^*} q_\downarrow S C_1 \gamma \overset{(\smile)}{\Rightarrow} q_\uparrow S C_1 \gamma$$
$$\overset{(3.13)}{\Rightarrow^*} q_\uparrow F C_1 B_1^{\mathrm{e}_n} \gamma \overset{(\nearrow)}{\Rightarrow} q' B_1^{\mathrm{e}_n} \gamma = \mathrm{rep}(q', \mathrm{e}_n, c_2, c_3). \qquad \square$$

The difficult part is to prove that the inverse is also true: An equivalence of configurations by $\mathcal{P}_C$ implies the equivalence by $\mathcal{P}'_C$. Consider the set of all possible configurations

$$W = \left\{ \mathrm{rep}(q, c_1, c_2, c_3) \;\middle|\; q \in Q,\ 0 \le c_1, c_2, c_3 \le \mathrm{e}_n \right\}.$$

Lemma 3.15. (i) *Any derivation* $q_{0,0} \Rightarrow^*_{\mathcal{P}_C} q_{\mathrm{a},0}$ *contains* $\mathrm{rep}(q_0, 0, 0, 0)$ *and* $\mathrm{rep}(q_{\mathrm{a}}, 0, 0, 0)$.

(ii) *For all* $w, w' \in W$ *we have* $w \equiv_{\mathcal{P}_C} w'$ *if and only if* $w \equiv_{\mathcal{P}'_C} w$.

With this result we can finally prove the promised hardness result for CSG.

Theorem 3.16 (CSG is EXPSPACE-hard). *The map* $C \mapsto (\Sigma_C, \mathcal{P}_C, q_{0,0}, q_{\mathrm{a},0})$ *defines a polynomial-time many-one reduction* $\mathsf{EBC} \le^{\mathrm{P}}_{\mathrm{m}} \mathsf{CSG}$. *In particular,* CSG *is* EXPSPACE*-hard.*

Proof. The map can clearly be computed in polynomial time (provided this is true for $\mathcal{P}_n$, which we will see in the next section). If $C \in \mathsf{EBC}$, then $\mathrm{rep}(q_0, 0, 0, 0) \equiv_{\mathcal{P}'_C} \mathrm{rep}(q_{\mathrm{a}}, 0, 0, 0)$ by Lemma 3.12 and consequently $q_{0,0} \equiv_{\mathcal{P}_C} q_{\mathrm{a},0}$ by Lemma 3.14. Conversely, if $q_{0,0} \equiv_{\mathcal{P}_C} q_{\mathrm{a},0}$, then by Lemma 3.15 and again Lemma 3.12 $C \in \mathsf{EBC}$. $\square$

Proof of Lemma 3.15. (i) Any derivation $q_{0,0} \Rightarrow^*_{\mathcal{P}_C} q_{\mathrm{a},0}$ must replace $q_{0,0}$, the only rule allowing for this is (0_0), introducing $q_{0,1} S C_1$. Only the rules (0_0)–(0_3) and $\mathcal{P}_n$ are applicable until the first word containing some $q \in Q$ is produced (which will be q_0 by (0_3)).

In order to apply (0_1) the previous rules from $\mathcal{P}_n$ must create a string containing F, by 3.13 the only possibility being $FC_1B_1^{e_n}$. If the derivation is repetition-free, then the only applicable rule now is (0_1), so the derivation has the form

$$q_{0,0} \overset{(0_0)}{\Rightarrow} q_{0,1}SC_1 \Rightarrow^*_{\mathcal{P}_n} q_{0,1}FC_1B_1^{e_n} \overset{(0_1)}{\Rightarrow} q_{0,2}SC_2B_1^{e_n}.$$

Repeating this argument two more times shows that the word $q_0B_1^{e_n}B_2^{e_n}B_3^{e_n}$ must occur in the derivation $q_{0,0} \Rightarrow^*_{\mathcal{P}_C} q_{a,0}$. Essentially the same reasoning starting from $q_{a,0}$ and using (a_0)–(a_3) shows that $q_aB_1^{e_n}B_2^{e_n}B_3^{e_n}$ is contained in the derivation, too.

(ii) The backward implication was proven in Lemma 3.15, so assume $\alpha = \mathrm{rep}(q, c_1, c_2, c_3) \equiv_{\mathcal{P}_C} \mathrm{rep}(q', c_1', c_2', c_3') = \beta$. We proceed by induction on the length of a derivation, the case of length 0 being trivial. Consider a shortest (necessarily repetition-free) derivation

$$\alpha = \gamma_0 \Rightarrow_{\mathcal{P}_C} \gamma_1 \Rightarrow_{\mathcal{P}_C} \cdots \Rightarrow_{\mathcal{P}_C} \gamma_r = \beta$$

and let $\gamma_m \Rightarrow_{\mathcal{P}_r} \gamma_{m+1}$ be the first step using one of the "new" rules $(\searrow)$,...,(a_0). Then all previous steps in the derivation $\gamma_0, \ldots, \gamma_k$ consists of rules from $\mathcal{P}'_C$ (the rules from $\mathcal{P}_n$ cannot be applied in the absence of C_i), so $\gamma_k \in W$ and $\mathrm{rep}(q, c_1, c_2, c_3) \equiv_{\mathcal{P}'_C} \gamma_m$.

The only rules possibly applicable to γ_m are $(\searrow)$,$(\nearrow)$,(0_3),(a_3), as all other rules require the presence of symbols not in $\gamma_m \in W$.

(0_3) Then γ_m contains state q_0 which is replaced by $q_{0,3}$, and hence the only rules applicable to γ_{m+1} containing $q_{0,3}FC_3$ are rules from $\mathcal{P}_n$. Theorem 3.13 tells us that $\mathcal{P}_n$ can only be used to produce $q_{0,3}SC_3$, furthermore, the maximum number e_n of B_3's are removed in this process (and so the string contains neither B_3's nor A_3's). Then the only applicable rule is (0_2) and we can repeat this argument twice to arrive at $q_{0,1}SC_1$. Now the only applicable rule replaces this with $q_{0,0}$ at which point no rule can be applied.

(a_3) The same line of reasoning prohibits the occurrence of this rule in a repetition-free derivation.

$(\searrow)$ Again only rules from $\mathcal{P}_n$ can be applied to γ_{m+1} and by applying Theorem 3.13 twice we must have

$$\begin{aligned}\gamma_m = qB_i^{e_n}w &\overset{(\searrow)}{\Rightarrow} \gamma_{m+1} = q_\downarrow FC_iB_i^{e_n}w \Rightarrow^*_{\mathcal{P}_n} q_\downarrow SC_iw \\ &\overset{(\smile)}{\Rightarrow} q_\uparrow SC_iw \Rightarrow^*_{\mathcal{P}_n} q_\uparrow FC_iB_i^{e_n}w \overset{(\nearrow)}{\Rightarrow} q'B_i^{e_n}w = \gamma_{m+l}\end{aligned}$$

where q' is determined by $\delta_C(q)$ and w determined by γ_m. Hence $\gamma_m \equiv_{\mathcal{P}'_C} \gamma_{m+l} \in W$.

$(\nearrow)$ The same argument yields $\gamma_m \equiv_{\mathcal{P}'_C} \gamma_{m+l} \in W$ for some $l \geq 1$.

By induction $\gamma_{k+l} \equiv_{\mathcal{P}'_C} \beta$ and thus $\alpha \equiv_{\mathcal{P}'_C} \beta$. □

3.4 Producing words of double-exponential length

In this section we prove Theorem 3.13, we follow the exposition by Bayer & Stillman [4]. Let $e_n := 2^{2^n}$. The alphabet Σ_n consists of $10(n+1)$ characters

$$\Sigma_n := \bigcup_{r=0}^{n} G_r, \quad G_r := \{s_r, f_r, b_{r,1}, b_{r,2}, b_{r,3}, b_{r,4}, c_{r,1}, c_{r,2}, c_{r,3}, c_{r,4}\}.$$

Definition 3.17 (Level of a word, box). The symbols in G_r are defined to be of *level* r. A word $w \in \Sigma_n^{\oplus}$ is said to be of *level* r if all symbols in w are of level $\geq r$ and

$$|w|_{s_r} + |w|_{f_r} = 1, \qquad \sum_{i=1}^{4} |w|_{c_{r,i}} = 1, \qquad |w|_{s_j} = |w|_{f_j} = 0 \quad \text{for } j > r.$$

The set of level r / $\leq r$ words is $\mathrm{LVL}(r)/\mathrm{LVL}(\leq r)$, if $w \in \mathrm{LVL}(r)$, then denote by $\mathrm{box}(x)$ the (unique) combination of s/f and c_i in x. ⌟

Notice that not all words are assigned a level. In order to ease notation, when fixing r we use S, F, B_i, C_i to denote the symbols of level r and s, f, b_i, c_i for level $r-1$ respectively. The defining relations in $\mathcal{P}_n$ are defined inductively as follows: $\mathcal{P}_0$ contains the four rules

$$s_0 c_{0,i} \to f_0 c_{0,i} b_{0,i}^2, \qquad i = 1, \ldots, 4. \tag{O$_i$}$$

For $r \geq 1$ define $\mathcal{P}_r$ to contain $\mathcal{P}_{r-1}$ and the ten rules

$$S \to s c_1 \tag{A}$$
$$f c_1 \to s c_2 \tag{B}$$
$$f c_2 C_i b_2 \to f c_2 C_i B_i b_3 \qquad i = 1, \ldots, 4 \tag{C$_i$}$$
$$f c_2 b_1 \to f c_3 b_4 \tag{D}$$
$$s c_3 \to s c_2 \tag{E}$$
$$s c_3 \to f c_4 \tag{F}$$
$$s c_4 \to F \tag{G}$$

The rules move words (of a given level) around in the boxes, this can be visualized as in figure 3.3. We first verify that the string $F C_i B_i^{e_n}$ (from level $r = n$) can actually be derived from $S C_i$ in the commutative Thue system defined by $\mathcal{P}_n$ over Σ_n.

Lemma 3.18. *For $r = 0, \ldots, n$ and $i = 1, \ldots, 4$ we have*

$$s_r c_{i,r} \Rightarrow_{\mathcal{P}_n}^{*} f_r c_{i,r} b_{i,r}^{e_r}. \tag{V}$$

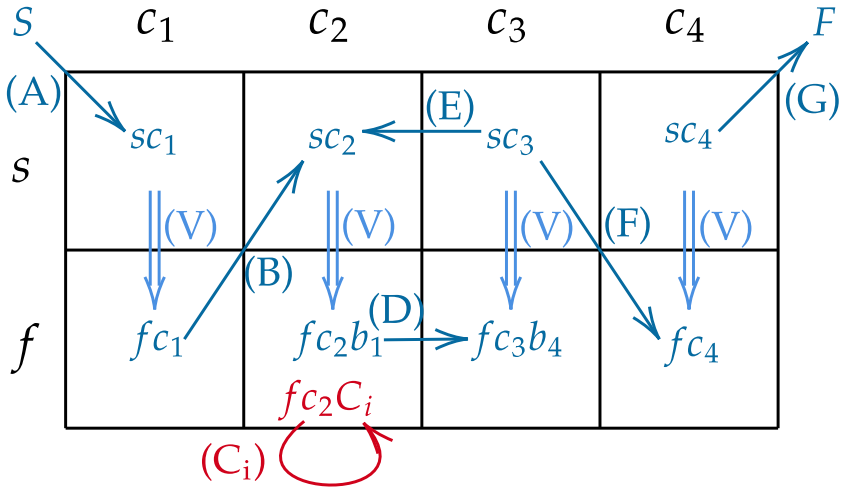


Figure 3.3: Rules (A)–(G),(V) in the context of boxes.

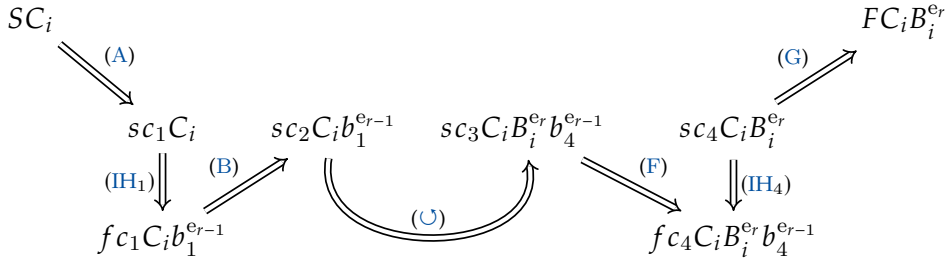
Proof. We proceed by induction on r, the case $r = 0$ (and $e_0 = 2$) follows by applying (O_i). Now considering level $r \geq 1$, we assume the induction hypothesis (IH_i) $sc_i \Rightarrow^*_{\mathcal{P}_n} fc_i b_i^{e_{r-1}}$ for $i = 1, \ldots, 4$. We can "exchange" b_1 for $B_i^{e_{r-1}} b_4$ in the presence of sc_2C_i as follows:

$$sc_2C_ib_1 \overset{(IH_2)}{\Rightarrow^*} fc_2C_ib_1b_2^{e_{r-1}} \overset{(C_i)}{\Rightarrow^*} fc_2C_iB_i^{e_{r-1}}b_1b_3^{e_{r-1}}$$
$$\overset{(D)}{\Rightarrow} fc_3C_iB_i^{e_{r-1}}b_3^{e_{r-1}}b_4 \overset{(IH_3)}{\Leftarrow^*} sc_3C_iB_i^{e_{r-1}}b_4 \overset{(E)}{\Rightarrow} sc_2C_iB_i^{e_{r-1}}b_4$$

If we repeat this sequence of derivations e_{r-1}-many times except for the final application of (E), then we obtain

$$sc_2C_ib_1^{e_{r-1}} \Rightarrow^*_{\mathcal{P}_n} sc_3C_i(B_i^{e_{r-1}}b_4)^{e_{r-1}}. \tag{↻}$$

With this in mind we can finally prove the assertion, following figure 3.3



□

Remark. The derivation $SC_i \Rightarrow^*_{\mathcal{P}} FC_iB_i^{e_r}$ in the proof is extremely long, this is not a surprise, as we need to derive a string of length 2^{2^n} and each individual rule only increases the length of the string by at most 2 symbols. To be precise, let ℓ_n denote the length of above derivation,

then $\ell_0 = 1$ and

$$\ell_n = 3 + 2\ell_{n-1} + e_{n-1} \cdot (2 + e_{n-1} + 2\ell_{n-1}) > 2^{2^n} + 2^{2^{n-1}} \cdot 2\ell_{n-1}$$

For example, the derivation $SC_i \Rightarrow^*_{\mathcal{P}_4} FC_iB_i^{65536}$ requires 3 658 397 steps.

We now prove that this is the *only* (repetition-free) derivation from SC_i. We use the following notation: If $(\Sigma, \mathcal{P})$ is a commutative Thue system, then let $\mathcal{G}(\mathcal{P})$ be the infinite undirected graph with vertices $\Sigma^\oplus$ and edges (α, β) for each derivation $\alpha \Rightarrow_{\mathcal{P}} \beta$. With this notation we want to prove that the component of SC_i in $\mathcal{G}(\mathcal{P}_n)$ consists of a simple path from SC_i to $FC_iB_i^{e_n}$.

For this, consider a simplified version of $\mathcal{G}(\mathcal{P}_r)$: The homomorphism $p_r : \Sigma_n^\oplus \to \Sigma_n^\oplus$ is defined on the generators $v \in \Sigma_n$ as

$$p_r(v) = \begin{cases} v & v \in \Sigma_{r-1} \cup \{s_r, f_r\} \\ \varepsilon & \text{otherwise.} \end{cases}$$

Then $\mathcal{Q}_r = p_r(\mathcal{P}_r)$ differs from $\mathcal{P}_r$ simply in the level r-rule (C_i) which is replaced by the single rule

$$fc_2b_2 \to fc_2b_3. \tag{$\tilde{C}$}$$

Applying the homomorphism p_r to the derivation from Lemma 3.18, we immediately obtain

Lemma 3.19. *We have $s_r \Rightarrow^*_{\mathcal{Q}_r} f_r$.*

Theorem 3.20 (Bayer & Stillman [4]). *Let $w \in \textsc{Lvl}(r)$.*

(i) *If $w \Rightarrow^*_{\mathcal{P}_r} x$, then $x \in \textsc{Lvl}(\leq r)$. If $S \Rightarrow^*_{\mathcal{Q}_r} x$, then $x \in \{S, F\}$ or $x \in \textsc{Lvl}(\leq r-1)$*

(ii) *The component of w in $\mathcal{G}(\mathcal{P}_r)$ and the component of S in $\mathcal{G}(\mathcal{Q}_r)$ contains no cycles.*

(iii) *If $x, y \in \textsc{Lvl}(\geq r)$ are distinct and $x \Rightarrow^*_{\mathcal{P}_r} y$, then there is a unique simple path in $\mathcal{G}(\mathcal{P}_r)$ connecting x and y and there is a $\gamma \in \Sigma_n^\oplus$ such that $\{x, y\} = \{\gamma SC_i, \gamma FC_iB_i^{e_r}\}$. There is a unique simple path in $\mathcal{G}(\mathcal{Q}_r)$ connecting S and F.*

We first do some preliminary work. Fix once again a level r and consider the map

$$\beta : \textsc{Lvl}(r-1) \to \mathbb{N} \times \mathbb{N}, \qquad x \mapsto (|x|_{b_1} + |x|_{b_4}, |x|_{b_2} + |x|_{b_3}).$$

In the derivation of Lemma 3.18 the values of $\beta(x)$ were determined by $\mathrm{box}(x)$ as in table 3.1, we now prove that this is true for any derivation.

	c_1	c_2	c_3	c_4
s	$(0,0)$	$(e,0)$	$(e,0)$	$(0,0)$
f	$(e,0)$	(e,e)	(e,e)	$(e,0)$

Table 3.1: Values of β on the boxes, $e := \mathrm{e}_{r-1}$.

Lemma 3.21. *Let $w \in \textsc{Lvl}(r)$ and assume Theorem 3.20(i)+(iii) is true in the case $r-1$. If there is a single $x_0 \in \textsc{Lvl}(r-1) \cap [w]_{\mathcal{P}_r}$ such that the value $\beta(x_0)$ is according to $\mathrm{box}(x_0)$ in table 3.1, then this is true for* all *$x \in \textsc{Lvl}(r-1) \cap [w]_{\mathcal{P}_r}$. The same is true for $\mathcal{G}(\mathcal{Q}_r)$.*

Proof. Consider a simple path in $\mathcal{G}(\mathcal{P}_r)$ from x_0 to $x \in \textsc{Lvl}(r-1)$. This path is an alternating sequence of segments of rules $\mathcal{P}_r \setminus \mathcal{P}_{r-1}$ and of segments contained in $\mathcal{G}(\mathcal{P}_{r-1})$.

- Rules (A) or (G) do not appear on the path, as this would result in a word without $c_{j,i}$ for $j \leq r-1$, and no rule of $\mathcal{P}_r$ can be applied (except for (A) or (G) again but the path is simple).
- Rules (B)–(F) preserve being of level $r-1$ and also preserve correctness of table 3.1.
- Segments in $\mathcal{G}(\mathcal{P}_{r-1})$ preserve being of level $\leq r-1$ by Theorem 3.20(i) and must end in a word of level $r-1$. Theorem 3.20(iii) then shows that the start and end of this segment also respect the values in Table 3.1.

Thus, since the statement is true for x_0 and is preserved along the path, it holds true for x. The proof for $\mathcal{G}(\mathcal{Q}_r)$ is essentially the same, as rule ($\tilde{\text{C}}$) also respects the table. □

Proof of Theorem 3.20. We proceed by induction on r, starting with $r = 0$. The arguments for $\mathcal{G}(\mathcal{Q}_r)$ only depend on the induction hypothesis for $\mathcal{G}(\mathcal{P}_r)$, hence the start only needs to consider $\mathcal{P}_0$.

If w contains $c_{0,i}$ and s_0 or f_0, then there is at most one applicable rule, namely (O_i) in *one* direction. This rule clearly preserves the property of being of level 0. Since there is at most one applicable rule, we immediately get that the component of w is either a single word or a pair of words separated by one application of (O_i), this proves both (ii) and (iii).

Now let $r \geq 1$ and assume that the Theorem is proven for the case $r-1$.

(i) Depending on whether w contains S or F, the rule (A) or (G) is the only applicable rule and yields a neighbor x_0 of level $r-1$. This word has $\mathrm{box}(x_0) \in \{sc_1, fc_4\}$ and no b_i's, hence Lemma 3.21 holds true.

Let x be any word of level $\leq r-1$ in the component of w. Any rule from $\mathcal{P}_{r-1}$ applied to x leads to words of level $\leq r-1$ by induction hypothesis (i). Any rule from $\mathcal{P}_n \setminus \mathcal{P}_{r-1}$ or ($\tilde{\text{C}}$) either leads to

- a word w' of level $r-1$ (in the case of (B)–(F),($\tilde{\text{C}}$)) or

- a word w' containing S or F and no symbols of level $\leq r-1$ except possibly $b_1, \dots, b_4$ (in the case of (A) or (G)). In this case we must have $\mathrm{box}(x) \in \{sc_1, fc_4\}$ and Lemma 3.21 tells us that x contains no b_i, so w' is of level r. Furthermore, in this case x is the only neighbor of w'.

Thus the words on any simple path in $\mathcal{G}(\mathcal{P}_r)$ or $\mathcal{G}(\mathcal{Q}_r)$ from w

$$w \Rightarrow x_0 \Rightarrow^* x \Rightarrow w'$$

are all of level $\leq r-1$ except possibly the last one being of level r (x is the only neighbor of w', hence such a word cannot appear inside simple path). This proves (i) for $\mathcal{G}(\mathcal{P}_r)$, in $\mathcal{G}(\mathcal{Q}_r)$ we start at $w = S$, then w' must be a single S or F as no rule of $\mathcal{Q}_r$ introduces other symbols of level $\geq r$.

(ii) Consider a simple cycle $\mathcal{C}$ in a component of S in $\mathcal{G}(\mathcal{Q}_r)$, i. e.

$$x_0 \Rightarrow_{\mathcal{P}_r} x_1 \Rightarrow_{\mathcal{P}_r} \cdots \Rightarrow_{\mathcal{P}_r} x_m = x_0, \qquad m \geq 2,\ x_i \neq x_j \text{ for } 0 \leq i < j \leq m-1.$$

Step 1: Translate into a statement purely about words of level $r-1$.

The cycle $\mathcal{C}$ does not contain S or F (having only a single neighbor), hence by (i) all x_i are in $\mathrm{Lvl}(\leq r-1)$. At least one of the x_i must be of level $r-1$, since otherwise no edges stemming from $\mathcal{Q}_r \setminus \mathcal{P}_{r-1}$ are used and $\mathcal{C}$ is a cycle on a component of a level $r-1$ word inside $\mathcal{G}(\mathcal{P}_{r-1})$, contradicting the induction hypothesis (ii) for $\mathcal{G}(\mathcal{P}_{r-1})$.

Let $y_0, y_1, \dots, y_k = y_0 \in \mathrm{Lvl}(r-1)$ be all words in $\mathcal{C}$ of level $r-1$ (in order of appearance). As in the proof of Lemma 3.21 we see that y_j is joined to $y_{j\pm1}$ by either a rule (B),($\tilde{\text{C}}$),(D)–(F) from $\mathcal{Q}_r \setminus \mathcal{P}_{r-1}$ or $sc_i \Rightarrow_{\mathcal{P}_{r-1}} fc_i b_i^{e_{r-1}}$ (V) by hypothesis (iii).
Step 2: Put the y_i into boxes and observe the transitions.

Hence no y_j is in box sc_1 or sc_4, as words in these boxes have only one outgoing edge (case (V)). By extension, no y_j is in box fc_1 or fc_4, since otherwise one of $y_{j\pm1}$ would necessarily be in box sc_1 or sc_4. Hence all y_j lie in boxes $\{sc_2, fc_2, fc_3, sc_3\}$. We must have a sort of circular traversal of the boxes for the following reason:

- If $\mathrm{box}(y_j) \in \{sc_2, sc_3, fc_3\}$, then by inspection of (B)–(F),(V)

$$\{\mathrm{box}(y_{j-1}), \mathrm{box}(y_{j+1})\} = \begin{cases} \{sc_3, fc_2\} & \text{if } \mathrm{box}(y_j) \in \{sc_2, fc_3\} \\ \{sc_2, fc_3\} & \text{if } \mathrm{box}(y_j) = sc_3. \end{cases}$$

- If $\mathrm{box}(y_j) = fc_2$, then by Lemma 3.21 $|y_j|_{b_2} + |y_j|_{b_3} = e_{r-1}$. The only applicable rule within the box is ($\tilde{\text{C}}$), trading b_2 for b_3. The other rules are (D) (to box fc_3) and case (V) (to box sc_3, only applicable if $|y_j|_{b_2} = e_{r-1}$). If (D) leads to, say, y_{j+1}, then the vertical rule (V) is

only applicable if $|y_{j+1}|_{b_3} = e_{r-1}$, hence y_j must be part of a sequence

$$y_l \overset{(V)}{\Rightarrow} y_{l+1} \underbrace{\overset{(C_i)}{\Rightarrow} \dots \overset{(C_i)}{\Rightarrow} y_j \overset{(C_i)}{\Rightarrow} \dots \overset{(C_i)}{\Rightarrow}}_{e_{r-1} \text{ applications of } (C_i)} y_{r-2} \overset{(D)}{\Rightarrow} y_{r-1} \overset{(V)}{\Rightarrow} y_r.$$

Step 3: Derive a contradiction from these observations.

These considerations show that the sequence of y_i's must loop around the four boxes in a fixed direction, say $sc_2 \rightsquigarrow fc_2 \rightsquigarrow fc_3 \rightsquigarrow sc_3 \rightsquigarrow sc_2$. Along such a loop, rule (D) is the only rule affecting b_4 and it strictly increases the number $|y_j|_{b_4}$. But this is clearly impossible in a closed cycle, hence the component of S in $\mathcal{G}(\mathcal{Q}_r)$ contains no cycle.

Assume there is a simple cycle $\mathcal{C}$ in $\mathcal{G}(\mathcal{P}_r)$, then the homomorphism p_r maps this to a cycle $p_r(\mathcal{C})$ in $\mathcal{G}(\mathcal{Q}_r)$. $p_r(\mathcal{C})$ is in the component of $p_r(w) \in \{S, F\}$, but this is the same component (Lemma 3.19). As any rule in $\mathcal{P}_r$ involves symbols not deleted by p_r, $p_r(\mathcal{C})$ is a non-trivial cycle in $\mathcal{G}(\mathcal{Q}_r)$, a contradiction.

(iii) The existence of a path $S \Rightarrow^*_{\mathcal{Q}_r} F$ is Lemma 3.19, uniqueness is immediate from (ii), as distinct simple paths can be patched together to form am non-trivial cycle.

Let $x, y \in \text{LVL}(\geq r)$ be distinct with $x \Rightarrow^*_{\mathcal{P}_r} y$, the uniqueness of such a path is again a consequence of (ii). Since there is *some* rule from $\mathcal{P}_r$ applicable to x, we must have $x \in \text{LVL}(r)$, the same is true for y. We must have $p_r(x) \neq p_r(y)$, since otherwise the path from x to y is mapped to a nontrivial cycle in $\mathcal{G}(\mathcal{Q}_r)$, contradicting (ii). Without loss of generality, let $p_r(x) = S$, $p_r(y) = F$.

Let $\text{box}(x) = SC_i$, then $\text{box}(y) = FC_i$, as all rules of $\mathcal{P}_r$ preserve the level r symbol C_i. The image of the path in $\mathcal{G}(\mathcal{Q}_r)$ is a path from S to F, hence the unique sequence of derivations from Lemma 3.19. This information together with the presence of C_i (and no other C_j) allows us to conclude that the path from x to y uses the sequence of derivations from Lemma 3.18. In particular, if $x = \gamma SC_i$, then $y = \gamma FC_iB_i^{e_n}$. □

Theorem 3.13 is now an easy consequence:

Proof of theorem 3.13. The presentation $\langle \Sigma_n \mid \mathcal{P}_n \rangle$ is of length $O(n)$ with $10(n+1)$ variables and $10n + 4$ relations of length ≤ 9 (the longest being (C_i)).

If $SC_i \Rightarrow^*_{\mathcal{P}_n} w$ such that w contains F, then w is of level n and Theorem 3.20(iii) says that

$$SC_i = \gamma SC_i, \qquad w = \gamma FC_iB_i^{e_n} = FC_iB_i^{e_n}.$$

The proof of the second claim in Theorem 3.13 is analogous. □

Remark. While we mostly care about $\mathcal{P}_n$ and the derivation $SC_i \Rightarrow^*_{\mathcal{P}_r} FC_iB_i^{e_n}$, the structure of $\mathcal{G}(\mathcal{Q}_r)$ is used crucially in the final step of the argument. This is one explanation to why we

took the extra mile to introduce $\mathcal{Q}_r$ and prove statements for *both* graphs. Another reason is that $\mathcal{Q}_r$ is useful for proving degree lower bounds on the ideal membership, see below.

We also remark that this construction has been dramtically improved by Yap from $10n+\mathcal{O}(1)$ variables down to only $2n+\mathcal{O}(1)$ [47]. Furthermore, if rule (O_i) contains $b_{0,i}^d$ instead of $b_{0,i}^2$, then the same construction yields a commutative Thue system which can count to d^{2^n} [4].

3.5 Hardness of the ideal membership problem

We now reduce CSG to the ideal membership problem (over an arbitrary field $\mathbb{K}$). Let $(\Sigma, \mathcal{P}, \alpha, \beta)$ be a tuple as in Definition 3.5, $\Sigma = \{x_1, \ldots, x_n\}$, $\mathcal{P} = \{\alpha_i \equiv \beta_i\}_{i=1,\ldots,s}$. To a word $\alpha \in \Sigma^*$ we associate the monomial $X^\alpha := X_1^{|\alpha|_{x_1}} \cdots X_n^{|\alpha|_{x_n}}$. The corresponding input to $\mathsf{IM}_\mathbb{K}$ is the set of polynomials $(f, f_1, \ldots, f_s)$ with

$$f := X^\beta - X^\alpha, \quad f_i := X^{\beta_i} - X^{\alpha_i}, \quad i = 1, \ldots, s.$$

As these polynomials have coefficients in $\{-1, 0, 1\}$, these polynomials are defined over any field (or ring).

Theorem 3.22. *With the preceding notation, the following statements are equivalent:*

(a) $\alpha \equiv_\mathcal{P} \beta$;

(b) $f \in I_\mathbb{Z} := \langle f_1, \ldots, f_s \rangle_{\mathbb{Z}[\underline{X}]}$;

(c) $f \in I_\mathbb{K} := \langle f_1, \ldots, f_s \rangle_{\mathbb{K}[\underline{X}]}$ *for* any *field* $\mathbb{K}$.

Proof. (a)⇒(b): Consider a derivation of length N

$$\alpha = \gamma_0 \Rightarrow_\mathcal{P} \gamma_1 \Rightarrow_\mathcal{P} \cdots \Rightarrow_\mathcal{P} \gamma_N = \beta.$$

Each step corresponds to an application of a rule $\alpha_{i_k} \equiv \beta_{i_k}$ from $\mathcal{P}$ such that

$$\gamma_{k-1} = \omega_k \alpha_{i_k}, \quad \gamma_k = \omega_k \beta_{i_k}, \qquad \omega_k \in \Sigma^\oplus,\ k = 1, \ldots, N.$$

Thus we get a telescoping sum

$$X^\beta - X^\alpha = \sum_{k=1}^N X^{\omega_k} \big(\underbrace{X^{\beta_{i_k}} - X^{\alpha_{i_k}}}_{= f_{i_k}} \big) \in I_\mathbb{Z}.$$

(b)⇒(c): Any $\mathbb{Z}[\underline{X}]$-linear combination is valid over an arbitrary field $\mathbb{K}$.

(c)⇒(a): This is the most difficult implication. Our strategy is to consider the fields $\mathbb{Q}$ and $\mathbb{F}_p = \mathbb{Z}/p\mathbb{Z}$ (p prime) first, and then generalize to arbitrary fields.

Step 1. $\mathbb{K} = \mathbb{Q}$. Consider a linear combination

$$f = \sum_k g_k f_{i_k}, \qquad g_k \in \mathbb{Q}[\underline{X}]. \tag{3.3}$$

Let $d \in \mathbb{N}_{>0}$ be a common denominator of all coefficients of all g_k, then $dg_k \in \mathbb{Z}[\underline{X}]$. By artificially increasing the number of summands in (3.3), we may assume that all dg_k are monomials with coefficient ± 1, for example

$$dg_k = 3XY - 2Y^2 + 2 \quad \rightsquigarrow \quad XY + XY + XY - Y^2 - Y^2 + 1 + 1.$$

Thus we obtain a (potentially much larger) sum

$$dX^\beta - dX^\alpha = \sum_{k=1}^{N} (-1)^{\varepsilon_k} X^{\omega_k} \cdot (X^{\beta_{j_k}} - X^{\alpha_{j_k}}). \tag{3.4}$$

We claim that there is a derivation $\alpha \Rightarrow_{\mathcal{P}}^* \beta$ of length at most N. Indeed, since the term X^α occurs on the left side of (3.4), this monomial must occur in one of the summands, i. e. $\pm X^{\omega_m}(X^{\beta_{j_m}} - X^{\alpha_{j_m}}) = X^{\gamma_1} - X^\alpha$. Removing this summand yields

$$dX^\beta - (d-1)X^\alpha - X^{\gamma_1} = \sum_{k=1,\, k\neq m}^{N} (-1)^{\varepsilon_k} X^{\omega_k}(X^{\beta_{j_k}} - X^{\alpha_{j_k}}). \tag{3.5}$$

such that $\alpha \Rightarrow_{\mathcal{P}} \gamma_1$ by application of the rule $\alpha_{j_m} \equiv \beta_{j_m}$. If $\gamma_1 = \beta$ then we are done, otherwise apply the same reasoning to the monomial X^{γ_1} to inductively obtain $\gamma_2, \gamma_3, \ldots$. This process ends after at most $N_0 \leq N$ steps with $\gamma_{N_0} = \beta$, since each step removes a summand from (3.4).

Step 2. $\mathbb{K} = \mathbb{F}_p$. Here nearly the same proof applies: Take a linear combination as in (3.3), this time over $\mathbb{F}_p[\underline{X}]$, then we can interpret the g_k as some polynomials with integer coefficients (being unique modulo p). We can again rewrite this sum as in (3.4), this time with $d = 1$ (no need to kill denominators). Then the same replacement scheme yields a sequence $\alpha \Rightarrow_{\mathcal{P}} \gamma_1 \Rightarrow_{\mathcal{P}} \gamma_2 \ldots$. The left hand side always reads $X^\beta - X^{\gamma_i}$ at each step, so no cancellation can happen (even mod p) unless $\gamma_j = \beta$ after at most N steps and we obtain a derivation $\alpha \Rightarrow_{\mathcal{P}}^* \beta$.

Step 3. $\mathbb{K}$ arbitrary. Any field contains either $\mathbb{Q}$ or some $\mathbb{F}_p$, its so-called *prime field*. All polynomials considered are defined over the prime field, so Lemma A.2 shows that $f \in I_{\mathbb{K}}$ already implies $I_{\mathbb{Q}}$ or $I_{\mathbb{F}_p}$ respectively, so by the previous two cases $\alpha \equiv_{\mathcal{P}} \beta$. □

This shows that the equivalence relation $\equiv_{\mathcal{P}}$ is basically the relation $\equiv_I$ from Lemma 1.34 on monomials. In this way we get a reduction from commutative semigroups to polynomial ideals.

Theorem 3.23 (CSG reduces to $\mathsf{IM}_{\mathbb{K}}$). *We have* $\mathsf{CSG} \leq_{\mathrm{m}}^{\mathrm{P}} \mathsf{IM}_{\mathbb{K}}$ *for any field* $\mathbb{K}$ *(and even for* $\mathbb{K} = \mathbb{Z}$*). This is also true for the homogeneous variants, i. e.* $\mathsf{CSG}_{\mathrm{h}} \leq_{\mathrm{m}}^{\mathrm{P}} \mathsf{IM}_{\mathrm{h},\mathbb{K}}$.

Proof. Indeed, the previous theorem shows

$$(\Sigma, \mathcal{P}, \alpha, \beta) \in \mathsf{CSG} \quad \text{if and only if} \quad (f, f_1, \ldots, f_s) \in \mathsf{IM}_{\mathbb{K}}.$$

Moreover the reduction mapping is computationally trivial, regardless of the encoding of coefficients (the only coefficients required are two constants ± 1) or monomials (the total degree is bounded).

For the second statement it suffices to notice that the $f_1, \ldots, f_s$ are homogeneous if $\mathcal{P}$ consists of homogeneous congruence rules. □

Together with the previous hardness results and the upper bounds from section 2.6 we see:

Corollary 3.24. *The problem* $\mathsf{IM}_{\mathbb{K}}$ *is* EXPSPACE*-hard and* CSG*,* $\mathsf{IM}_{\mathbb{Q}}$ *are* EXPSPACE*-complete. The homogeneous variant* $\mathsf{IM}_{\mathrm{h},\mathbb{K}}$ *is* PSPACE*-hard and* $\mathsf{CSG}_{\mathrm{h}}$*,* $\mathsf{IM}_{\mathrm{h},\mathbb{Q}}$ *are* PSPACE*-complete.*

It is interesting to consider the analogous problem in the context of algebraic complexity theory. Bürgisser has adapted these techniques to prove an analogous results for algebraic circuits.

Theorem 3.25 (Bürgisser 1998 [9]). *For any infinite field* $\mathbb{K}$ *there is a constant* c *such that any sequence of algebraic circuits* $(\mathcal{C}_n)_{n\in\mathbb{N}}$ *deciding* $\mathsf{IM}_{\mathbb{K}}$ *has* $\operatorname{depth}(\mathcal{C}_n) \geq 2^{cn}$ *for* $n \gg 0$.

Remark. Recall the Hermann bound which states that if $f = \sum_{i=1}^{s} h_i f_i$, then the h_i may be chosen of degree $\leq \deg(f) + (s \cdot \max_i \deg f_i)^{2^n}$. The results from the previous section show that this bound is asymptotically tight as follows: We have seen that there is a unique derivation $S \equiv_{\mathcal{Q}_n} F$ which contains words of length $> \mathrm{e}_{n-1}$. This corresponds to a polynomial equation $S - X^F = \sum_j h_i f_i$ and the proof of 3.22 shows that the degree of any term $h_i f_i$ gives a upper bound on the length of words in the derivation. Thus, any such h_i must have degree double-exponential in n, for details see the paper by Bayer & Stillman [4, Theorem 2.4].

3.6 Church-Rosser systems

We return to the topic of Gröbner bases. The previous section gave a connection between commutative semigroup presentations and binomial ideals. We now extend this connection and define a notion analogous to Gröbner bases, following the paper by Huynh [21]. Fix a monomial order $\prec$ on $\Sigma^{\oplus} \cong \mathrm{Mon}_n$ and consider a commutative Thue system $\mathcal{P}$.

Definition 3.26 ((Ir)reducible words). Let $\mathcal{P}_{\succ} := \{ (l, r) \in \mathcal{P} \mid l \succ r \}$. A word $x \in \Sigma^{\oplus}$ is *reducible* with respect to $\mathcal{P}$ if $x \Rightarrow_{\mathcal{P}_{\succ}} y$ for some $y \in \Sigma^{\oplus}$, and irreducible otherwise. The transformation $x \Rightarrow_{\mathcal{P}_{\succ}} y$ is called a reduction step. ⌟

Notice that if $x \Rightarrow_{\mathcal{P}_\succ} y$, then by the defining property of a monomial order, $x \succ y$. As $\prec$ is a well-order, any sequence of reduction steps must be of finite length and hence every word is equivalent to an irreducible word with respect to $\mathcal{P}$. Of course, such an irreducible word is generally not unique, similarly to how normal forms of polynomials are not unique. But we have seen that this does not happen for Gröbner bases (Theorem 1.18(b)), so we use this to define an analogous notion for commutative Thue systems.

Definition 3.27 (Church-Rosser system, reduced). A commutative Thue-system $(\Sigma, \mathcal{P})$ is a *Church-Rosser system* if for any two words $u \equiv_{\mathcal{P}} v$, both irreducible with respect to $\mathcal{P}$, we have $u = v$.

A Church-Rosser system is *reduced* if for all $(l, r) \in \mathcal{P}$ both l and r are irreducible with respect to $\mathcal{P} \setminus \{(l, r), (r, l)\}$. ⌟

We use the notation from the previous section to translate commutative Thue systems into ideal generators.

Theorem 3.28. *Let $\mathcal{P}$ be a commutative Thue system and*

$$G = \left\{ X^\alpha - X^\beta \;\middle|\; (\alpha, \beta) \in \mathcal{P}, \alpha \succ \beta \right\} \subseteq \mathbb{K}[\underline{X}].$$

Then $\mathcal{P}$ is a Church-Rosser system if and only if G is a Gröbner basis of $I := \langle G \rangle$. Furthermore, $\mathcal{P}$ is reduced if and only if G is reduced.

Proof. Assume first that G is a Gröbner basis. Let $u, v \in \Sigma^\oplus$, $u \equiv_{\mathcal{P}} v$, both u, v irreducible with respect to $\mathcal{P}$. Then X^u and X^v are in normal form with respect to G (compare Definition 1.14), and hence $\mathrm{NF}_I(X^u) = X^u$, $\mathrm{NF}_I(X^v) = X^v$, as G is a Gröbner basis. Theorem 1.34 tells us that both X^u and X^v are the unique minimal element of their equivalence classes with respect to $\equiv_I$. But by assumption $u \equiv_{\mathcal{P}} v$, by Theorem 3.22 we have $[X^u]_{\equiv_I} = [X^v]_{\equiv_I}$, and so they must coincide.

Now assume that $\mathcal{P}$ is a Church-Rosser system. Let $X^u \in \mathrm{IN}(I)$ and assume u were irreducible with respect to $\mathcal{P}$. Theorem 1.34 yields a $\beta \prec u$ such that $X^u - X^\beta \in I$. Theorem 3.22 implies that $\beta \equiv_{\mathcal{P}} u$, and after applying a finite number of reduction steps we obtain $\beta \Rightarrow^*_{\mathcal{P}_\prec} v$ with v irreducible, too. So $u \equiv_{\mathcal{P}} v$ and both are irreducible, by the Church-Rosser property we have $u = v$, which contradicts $u \succ \beta \succeq v$. Hence there is a $(l, r) \in \mathcal{P}_\prec$ with $\mathrm{LM}(X^l - X^r) = X^l \mid X^u$, and G is a Gröbner basis by characterization 1.18(a).

It is immediate from the definitions that G is reduced if and only if $\mathcal{P}$ is reduced. □

Corollary 3.29. *For every congruence relation $\sim$ on $\Sigma^\oplus$ there exists a* unique *Church-Rosser system $\mathcal{P}$ with $\equiv_{\mathcal{P}} = \sim$.*

Proof. There exists *some* commutative semigroup presentation $\mathcal{P}_0$ with $\sim = \equiv_{\mathcal{P}_0}$ by Rédei's theorem A.7. Let $I = \left\langle \{ X^\alpha - X^\beta \mid (\alpha, \beta) \in \mathcal{P}_0 \} \right\rangle$ and let G be the reduced Gröbner basis of

I. Then $\mathcal{P} = \{ (l, r) \mid X^l - X^r \in G \}$ is a Church-Rosser system equivalent to $\mathcal{P}_0$ Uniqueness follows immediately from the uniqueness of reduced Gröbner bases. □

We now translate some notation and results form the polynomial world to the land of Thue systems. Fix a commutative Thue system $\mathcal{P}$

Definition 3.30. For a word $u \in \Sigma^\oplus$ let $\mathrm{NF}_\mathcal{P}(u)$ be the minimal word in $[u]_{\equiv_\mathcal{P}}$ with respect to $\prec$. We define the sets

$$U_\mathcal{P} = \{ u \in \Sigma^\oplus \mid \mathrm{NF}_\mathcal{P}(u) = u \}, \qquad A_\mathcal{P} \coloneqq \Sigma^\oplus \setminus U_\mathcal{P}. \qquad ⌟$$

We denote the minimal elements of a set $A \subseteq \Sigma^\oplus$ with respect to string containment by $\min A$.

Lemma 3.31. *Let $\mathcal{P}$ be a reduced Church-Rosser system.*

(i) *$A_\mathcal{P}$ is the set of elements reducible with respect to $\mathcal{P}$.*

(ii) $\mathrm{Min}\, A_\mathcal{P} = \{ l \mid (l, r) \in \mathcal{P} \}$

Proof. As $\mathcal{P}$ is a Church-Rosser system, a string $u \in \Sigma^\oplus$ is irreducible with respect to $\mathcal{P}$ if and only if it is minimal in its congruence class $[u]_\mathcal{P}$, i. e. in $U_\mathcal{P}$. This proves the first statement, the second statement is a direct translation of Theorem 1.32. □

3.7 The size of a reduced Gröbner basis

In this section we restrict ourselves to a *degree-dominating* monomial order $\prec$ with the property

$$|u| < |v| \quad \Longrightarrow \quad u \prec v.$$

For example $\prec_{\mathrm{grlex}}$ and $\prec_{\mathrm{grevlex}}$ have this property, but $\prec_{\mathrm{lex}}$ does not.

Theorem 3.32 (Huynh 1986 [21]). *For each n there exists an ideal I_n generated by $\mathcal{O}(n)$ differences of monomials of total degree $\mathcal{O}(1)$ with the following properties:*

Any Gröbner basis of I_n has at least 2^{2^n} elements and the maximal total degree of its elements is at least 2^{2^n}.

By Corollary 1.29 the equivalence of reduced Church-Rosser systems and reduced Gröbner bases of pure difference ideals, it suffices to prove the analogous statement for Church-Rosser systems:

Theorem 3.33. *For each n there exists a commutative Thue system $\mathcal{R}_n$ generated by $\mathcal{O}(n)$ rules of length $\mathcal{O}(1)$ with the following properties:*

The reduced Church-Rosser system generating the same congruence as $\mathcal{R}_n$ has at least 2^{2^n} elements and the maximal length of one side of the rules is 2^{2^n}.

To prove this, we follow the strategy of Huynh [21], but we simplify the construction of a system $\mathcal{R}_n$ with the following property:

Lemma 3.34. *Let $u = rSC_3B_1^eB_2^f \in \Sigma^\oplus$ with $e + f = \mathrm{e}_n$. Then u is* not *minimal in $[u]_{\mathcal{R}_n}$ with respect to $\prec$, but any proper substring $u' \mid u$ is the minimal element of $[u']_{\mathcal{R}_n}$.*

We first show that this property implies the theorem.

Proof of Theorem 3.33. Consider the e_n+1 distinct words $u_k := rSC_3B_1^kB_2^{\mathrm{e}_n-k}$, $k = 0, \dots, \mathrm{e}_n$, each of length $\mathrm{e}_n + 2$. By Lemma 3.34 each $u_k \in A_{\mathcal{R}_n}$ is a *minimal* element of this set, and hence by Lemma 3.31 we have

$$\{u_0, \dots, u_{\mathrm{e}_n}\} \subseteq \{\, l \mid (l, r) \in \mathcal{P} \,\}$$

where $\mathcal{P}$ is the reduced Church-Rosser system equivalent to $\mathcal{R}_n$. Hence $|\mathcal{P}| > \mathrm{e}_n$ and $\mathcal{P}$ contains rules of length $> \mathrm{e}_n$. □

We now construct $\mathcal{R}_n$: Recall the construction of $(\Sigma_n, \mathcal{P}_n)$ from section 3.4, then the alphabet is $\Sigma_n \dot\cup \{r, r', r'', \bar B_1, \bar B_2\}$ and $\mathcal{R}_n$ consists of $\mathcal{P}_n$ and the additional rules

$$rFC_3B_1 \to rFC_3\bar B_1B_4 \qquad rFC_3B_2 \to rFC_3\bar B_2B_4 \tag{R}$$
$$rFC_3 \to r'FC_1 \tag{R$'_1$}$$
$$r'SC_1 \to r'FC_3 \tag{R$'_2$}$$
$$rFC_3 \to r''FC_4 \tag{R$''_1$}$$
$$r''SC_4\bar B_2 \to r''SC_4\bar B_1. \tag{R$''_2$}$$

The rules and their intended behavior is visualized in figure 3.4. The proof of Lemma 3.34 makes use of the properties of $\mathcal{P}_n$ from Theorem 3.13, and the additional fact that in the derivation (V_i) $SC_i \Rightarrow^* FC_iB_i^{\mathrm{e}_n}$ any intermediate expression has length ≥ 3.

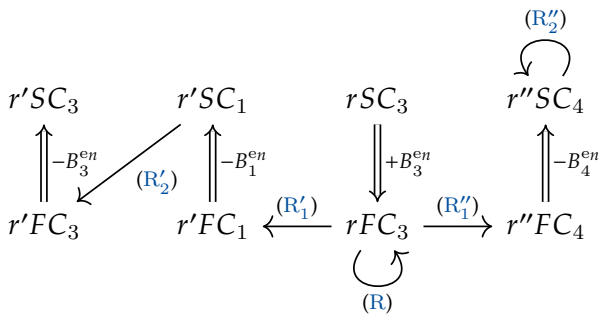


Figure 3.4: A visualization of the rules in $\mathcal{R}_n$.

Proof of Lemma 3.34. Let $u = rSC_3B_1^eB_2^f$ and assume first that $e + f = \mathrm{e}_n$, then

$$u \overset{(V_3)}{\Rightarrow} rFC_3B_1^eB_2^fB_3^{\mathrm{e}_n} \overset{(R)}{\Rightarrow}{}^* rFC_3\bar{B}_1^e\bar{B}_2^fB_3^{\mathrm{e}_n}B_4^{\mathrm{e}_n} \overset{(R_1'')}{\Rightarrow} r''FC_4\bar{B}_1^e\bar{B}_2^fB_3^{\mathrm{e}_n}B_4^{\mathrm{e}_n} \overset{(V_4)}{\Leftarrow} r''SC_4\bar{B}_1^e\bar{B}_2^fB_3^{\mathrm{e}_n}$$
$$\overset{(R_2'')}{\Rightarrow}{}^* r''SC_4\bar{B}_1^{\mathrm{e}_n}B_3^{\mathrm{e}_n} \overset{(V_4)}{\Rightarrow} r''FC_4\bar{B}_1^{\mathrm{e}_n}B_3^{\mathrm{e}_n}B_4^{\mathrm{e}_n} \overset{(R_1'')}{\Leftarrow} rFC_3\bar{B}_1^{\mathrm{e}_n}B_3^{\mathrm{e}_n}B_4^{\mathrm{e}_n} \overset{(R)}{\Leftarrow}{}^* rFC_3B_1^{\mathrm{e}_n}B_3^{\mathrm{e}_n}.$$

This shows that all such u are equivalent to $rFC_3B_1^{\mathrm{e}_n}B_3^{\mathrm{e}_n}$. We then have

$$rFC_3B_1^{\mathrm{e}_n}B_3^{\mathrm{e}_n} \overset{(R_1')}{\Rightarrow} r'FC_1B_1^{\mathrm{e}_n}B_3^{\mathrm{e}_n} \overset{(V_1)}{\Leftarrow} r'SC_1B_3^{\mathrm{e}_n} \overset{(R_2')}{\Rightarrow} r'FC_3B_3^{\mathrm{e}_n} \overset{(V_3)}{\Leftarrow} r'SC_3$$

and since $|u| = \mathrm{e}_n + 3 > 3$, we have $u \succ r'SC_3$, so u is not minimal in its equivalence class.

Now consider any proper substring u' of u. If u' does not contain r, then only rules from $\mathcal{P}_n$, leading to strictly longer words. If u' does not contain S or C_3, then no rule is applicable at all and $[u']_{\mathcal{R}_n} = \{u'\}$, so u' is trivially minimal. Hence we may assume $u' = rSC_3B_1^eB_2^f$, $e + f < \mathrm{e}_n$. In order to prove minimality it suffices to show that for *any* $v \neq u'$ with $v \equiv_{\mathcal{R}_n} u'$ we have $|v| > |u'|$, as this implies $v \succ u'$.

Consider a repetition-free derivation from u' to v, the prefix of v is the substring of v of the form $(r|r'|r'')(S|F)(C_1|\dots|C_4)$ (the structure of $\mathcal{R}_n$ easily implies that V contains at most one such word). Then $\mathcal{P}_n$ (in the form of Theorem 3.13) and (R)–(R_2'') imply that the only possible transitions between words of different prefixes behave as displayed in Figure 3.4.

First assume that the derivation uses (parts of) (V_1) or (V_4), and let v' be word in the derivation prior to the first such step. The rules (R),(R_1'),(R_1'') and the transformation (V_3) leave the numbers

$$n_v = |v|_{B_1} + |v|_{B_2} + |v|_{\bar{B}_1} + |v|_{\bar{B}_2}, \qquad \bar{n}_v = |v|_{\bar{B}_1} + |v|_{\bar{B}_2} - |v|_{B_4}$$

invariant, so $n_{v'} = n_u = e + f < \mathrm{e}_n$, $\bar{n}_{v'} = 0$. In particular $|v'|_{B_1}, |v'|_{B_4} < \mathrm{e}_n$, so neither ($V_1$) nor ($V_4$) can ever be completely applied.

The derivation $u \Rightarrow^* v$ necessarily starts by following the unique derivation in $\mathcal{P}_n$ from $rSC_3B_1^eB_2^f$ to $u'' := rFC_3B_1^eB_2^fB_3^{\mathrm{e}_n}$. Any intermediate word is strictly longer than u'. We have $|u''| = |u| + \mathrm{e}_n$, and it is not hard to see that (R),(R_1'),(R_1'') and parts of the derivations (V_1), (V_4) only lead to words of length $> |u''| - \mathrm{e}_n \geq |u|$. We conclude that $v = u'$ or $|v| > |u'|$ and hence u' is minimal in its equivalence class. □

In chapter 2 we have seen that upper bounds on the degree of Gröbner bases of *homogeneous* ideals lead to upper bounds on the degree of arbitrary ideals (Lemma 2.15). Conversely, lower bounds for arbitrary ideals also apply to homogeneous ideals, in particular Theorem 3.32 also holds for homogeneous ideals. Mayr & Ritscher also gave some lower bounds depending on the dimension of the ideal.

Theorem 3.35. *There are a monomial ordering and a family of ideals $I_{r,n} \subseteq \mathbb{K}[X_1, \dots, X_n$ of dimension*

$\dim(I_{r,n}) \leq r$, *for all* $r, n \in \mathbb{N}$, $r \leq n$, *which are generated by generated by* $\mathcal{O}(n)$ *polynomials of degrees bounded by* d *such that each Gröbner basis has a maximal degree of at least* $d^{(n-r)2^{(1/2-\varepsilon)r}}$ *for any* $\varepsilon > 0$ *and sufficiently large* $d, r \in \mathbb{N}$.

This shows that the upper bound from Theorem 2.17 is somewhat tight, in particular the degree of Gröbner bases may be loosely described as $2^{n^{\Theta(1)}2^{\Theta(r)}}$.

3.8 Hardness results of Gröbner bases

We finally prove some results on the complexity of Gröbner bases. Fix a monomial order $\prec$ which is degree-dominating.

Theorem 3.36. *Any algorithm which on input* $F = (f_1, \ldots, f_s)$ *computes the reduced Gröbner basis* G *of* $I = \langle F \rangle$ *with respect to* $\prec_{grlex}$ *uses in the worst case at least space* $2^{\Omega(n)}$ *and time* $2^{2^{\Omega(n)}}$ *in the length of* F. *This is independent of the base field* $\mathbb{K}$, *the encoding of field elements or whether binary or unary exponent representation of the monomials is used. This also holds true when restricting homogeneous ideals.*

Proof. Indeed, the polynomials from Theorem 3.32 are differences of monomials and hence use coefficients $\{0, 1, -1\}$. Furthermore, the terms have bounded degree d_0, so dense or sparse encoding of the monomials does not change the input length up to a constant factor.

The output consist of $\geq 2^{2^n} = \mathrm{e}_n$ different polynomials, so the algorithm requires at least e_n steps and passes through at least e_n different configurations of the work tape. This implies that at least 2^n work tape cells have to be visited in order to represent e_n different internal configurations. □

This lower bound matches the space and time requirements of Theorem 2.29 (for $\mathbb{K} = \mathbb{Q}$).

Remark. We have seen that the size of the output of the Turing machine dictates a lower bound on the resources used by said machine. This is however not enough to prove that the *individual elements* of the output are hard to describe. For example, consider the function

$$F(n) = (0, 1, 2, \ldots, 2^n - 1),$$

where numbers are encoded in binary. Then the length $|F(n)|$ is exponential in n and hence double-exponential in $|\mathrm{bin}(n)|$, but on the other hand it is computationally trivial to decide whether $m \in \mathbb{N}$ is part of the list $F(n)$. This is *not* true in the case of the reduced Gröbner basis, as shown next.

Theorem 3.37 ($\mathsf{GROEBM}_{\mathbb{K}}$ **is** EXPSPACE**-hard).** *The problem to decide whether a given polynomial* g *is a member of the reduced Gröbner basis of* $\langle f_1, \ldots, f_s \rangle$ *is hard for* EXPSPACE, *regardless of the specified monomial order* $\prec$.

Proof. The reduction EBC $\leq_{\mathrm{m}}^{\mathrm{P}}$ CSG $\leq_{\mathrm{m}}^{\mathrm{P}}$ $\mathsf{IM}_{\mathbb{K}}$ shows that $\mathsf{IM}_{\mathbb{K}}$ is EXPSPACE-hard even when restricting to the following subset A:

- The polynomials $f, f_1, \ldots, f_s$ are differences of monomials of positive degree.
- The polynomial f whose membership has to be decided is of the form $f = x - y, x, y \in \underline{X}$.

After renumbering the variables we may assume that $x' > y$ for all $x' \in \underline{X} \setminus \{y\}$, i. e. y is the smallest variable with respect to $<$. This implies that y is also the smallest monomial different from 1: Any monomial m contains some variable x', and hence $m \geq x' \geq y$. Moreover $1 \notin [m]_{\equiv_I}$ for any monomial $m \neq 1$, since the f_i have positive degree (in Thue system language: No rule is applicable to the empty string).

Let $I = \langle f_1, \ldots, f_s \rangle$ be an ideal generated by pure differences and G the reduced Gröbner basis of I. With this notation we claim

$$x - y \in \langle f_1, \ldots, f_s \rangle \quad \text{if and only if} \quad x - y \in G.$$

If $x - y \in G$, then surely $x - y \in I$, as $G \subseteq I$. Conversely, assume $x - y \in I$, then Theorem 1.34 shows that $\mathrm{NF}_I(x) = y$ as $y \equiv_I x$ and y is *the* smallest monomial in $\mathrm{Mon}_n \setminus \{1\}$. The monomial x is minimally reducible with respect to I (Definition 1.31), as it is reducible ($\mathrm{NF}_I(x) = y \neq x$), but its only proper divisor 1 is irreducible. Then Theorem 1.32 shows that $x - y \in G$.

This proves $A \leq_{\mathrm{m}}^{\mathrm{P}} \mathsf{GROEBM}_{\mathbb{K}}$, where the reduction function is the identity map, so $\mathsf{GROEBM}_{\mathbb{K}}$ is EXPSPACE-hard, too. □

Combining this with the exponential space algorithm for reduced Gröbner basis membership from chapter 2 yields the following completeness result:

Corollary 3.38. *The problem* $\mathsf{GROEBM}_{\mathbb{Q}}$ *is* EXPSPACE*-complete.*

Remark. The hardness result of $\mathsf{GROEBM}_{\mathbb{K}}$ is implicitly contained in the literature, although not mentioned explicitly (to the knowledge of the author). The phrase "Gröbner bases are EXPSPACE-complete" mostly refers to the results of Theorem 2.29 and 3.36.

We close our discussion on the complexity of Gröbner bases with a recent result due to Rolnick & Spencer which shows that Gröbner bases are hard to approximate. Let $1 \geq \varepsilon > 0$ be a constant and define the *ε-Fractional Gröbner problem* as follows:

- Input: $F = \{f_1, \ldots, f_s\} \subseteq \mathbb{K}[\underline{X}]$ polynomials
- Output: (F', G), where $F' \subseteq F$ with $|F'| \geq \varepsilon \cdot |F|$ and G is a Gröbner basis of $\langle F' \rangle$

Theorem 3.39 (Rolnick & Spencer 2018 [46]). *For infinite fields* $\mathbb{K}$*, the ε-Fractional Gröbner problem is* NP*-hard for every fixed $\varepsilon > 0$. This is true for any monomial order and even when the polynomials are of degree ≤ 3.*

The authors remark that this holds true even if polynomial time is measured not only in the size of the input, but also on the output [46, Theorem 3]. This is remarkable, since we know that Gröbner bases may become exceedingly large even for small inputs.

Conclusion

In this thesis we discussed the polynomial ideal membership and its connection to Gröbner bases. We have seen how Gröbner bases can be constructed using Buchberger's algorithm and how the multivariate division algorithm calculates normal forms which can be used to ideal membership. Although this approach is successful in many cases, especially with improved algorithms such as Faugère's F_5 algorithm, the worst-case complexity points to the limitations of Gröbner bases.

We have seen that both the ideal membership problem $\mathsf{IM}_{\mathbb{Q}}$ as well as the reduced Gröbner basis membership problem $\mathsf{GROEBM}_{\mathbb{Q}}$ are EXPSPACE-complete, in particular both problems are computationally very involved in their full generality. While it is intuitively clear that Gröbner bases must be at least as complex as the problem they are trying to solve (in an appropriate sense), the other results surveyed here suggest that they are sub-optimal in many important cases. Namely, the ideal membership problem for homogeneous polynomials $\mathsf{IM}_{\mathrm{h},\mathbb{Q}}$ is PSPACE-complete, but Gröbner basis computation for homogeneous ideals still requires exponential space in the worst case. Even worse, the number and degree of elements in a Gröbner basis of a homogeneous ideal may be double-exponential in the number of variables and generators of the ideal. Still, in many practical cases Gröbner bases are a good first choice for problems in computer algebra.

There are many other interesting topics related to the ideas presented here, we would like to point to three particular ones.

- When presented with a problem involving Gröbner basis computation, it is often difficult to estimate whether the computation will be a matter of minutes or months. Hence an important question is: What properties of an ideal make their Gröbner basis complicated? This is particularly important for applications in computational algebraic geometry, where many problems can be reduced to a Gröbner basis computation of some sort. We have seen that the number of variables and the dimension play a role both for the complexity of the ideal membership as well as the size of the reduced Gröbner basis. An important piece of the puzzle is the *regularity* of the ideal, a notion which is explored in the paper by Bayer & Mumford [3].
- Gröbner bases provide an approach to calculate normal forms of polynomials with respect to an ideal. If the system of polynomials has a finite number of solutions and is sufficiently general, then the set of solutions is closely connected to the quotient ring

$\mathbb{C}[\underline{X}]/\langle f_1, \ldots, f_s \rangle$, for example its dimension over $\mathbb{C}$ is the number of solutions. Normal forms provide a way to compute effectively in this ring, see for example the *truncated normal form* by Telen [38]. For an analysis of the performance of several normal form algorithms see the work of Parkinson et. al [41].

- While symbolic computations on systems of polynomial equations are important, for many applications approximations to solutions are sufficient, especially for large systems. One of the most successful techniques is *homotopy continuation*, where a system of equations with known solutions is continuously transformed into the target system, and the solution paths are tracked. A particular implementation, which allows for certifying zeros (i. e. providing a certificate that a given approximate zero corresponds to an *actual* solution in $\mathbb{C}^n$) is HomotopyContinuation.jl [6].

A

Appendix

A.1 Commutative Algebra

All rings in this thesis are assumed to be commutative and with unity. An ideal of a ring is an additive subgroup $I \subseteq A$ closed under multiplication with elements from A, i. e. $a \cdot f \in I$ for $a \in A$, $f \in I$. If $F \subseteq A$ is a subset, then the ideal generated by F is

$$\langle F \rangle_A = \left\{ a_1 f_1 + \cdots + a_m f_m \;\middle|\; a_i \in A,\ f_i \in F \right\},$$

this is the smallest ideal I containing F and F is a *generating set* of I. If $F = \{f_1, \ldots, f_m\}$ then we write $\langle f_1, \ldots, f_m \rangle := \langle F \rangle$. An ideal of the form $\langle f \rangle$ is a *principal ideal*.

A ring A is *Noetherian* if one of the following equivalent conditions is satisfied [24, Thm. 2.9]

(i) Any generating set of an ideal contains a finite generating set.

(i′) Every ideal of A admits a finite generating set.

(ii) A satisfies the *ascending chain condition* on ideals: Any chain of ideals $I_0 \subseteq I_1 \subseteq I_2 \subseteq \ldots$ eventually becomes stationary: $I_{n_0} = I_n$ for all $n \geq n_0$.

(iii) Any nonempty set of ideals in A has a maximal element with respect to inclusion.

The most important ring for our purposes is the ring of polynomials in a set of indeterminates over a field $\mathbb{K}[X_1, \ldots, X_n]$. A classical result asserts that this is a Noetherian ring.

Theorem A.1 (Hilbert basis theorem). *If A is a Noetherian ring, then so is $A[X]$. In particular $\mathbb{K}[X_1, \ldots, X_n]$ is a Noetherian ring.*

Proof. See for example [24, Thm. 2.11] or [19, Thm. 1.3.5] □

If $A \subseteq B$ are rings and $F \subseteq A$, then in general $\langle F \rangle_B \cap A \supseteq \langle A \rangle$, but we have equality in the following special case:

Lemma A.2. *Let $\mathbb{L} \supseteq \mathbb{K}$ be two fields, $f, f_1, \ldots, f_s \in \mathbb{K}[\underline{X}]$, then we have*

$$f \in \langle f_1, \ldots, f_k \rangle_{\mathbb{L}[\underline{X}]} \quad \Longrightarrow \quad f \in \langle f_1, \ldots, f_k \rangle_{\mathbb{K}[\underline{X}]}$$

Proof. $\mathbb{L}$ is a vector space over $\mathbb{K}$, we can choose[1] a basis $\{b_i\}_{i\in I}$ containing $b_{i_0} = 1 \in \mathbb{L}$. This then also constitutes a basis of the free module $\mathbb{L}[\underline{X}]$ over $\mathbb{K}[\underline{X}]$. Now consider an expression

$$f = \sum_{k=1}^{s} g_k f_k, \qquad g_k \in \mathbb{L}[\underline{X}].$$

We can write $g_k = \sum_{i\in I} g_{ik} b_i$ with finitely many nonzero $g_{ik} \in \mathbb{K}[\underline{X}]$, then the former equation becomes

$$f \cdot b_{i_0} = f = \sum_{k=1}^{s} \left(\sum_{i\in I} g_{ik} b_i \right) f_j = \sum_{i\in I} \left(\sum_{k=1}^{s} g_{ik} f_k \right) b_i.$$

Comparing the coefficient of the basis element $b_{i_0} = 1$ in this equation yields

$$f = \sum_{k=1}^{s} g_{i_0 k} f_k \in \langle f_1, \ldots, f_k \rangle_{\mathbb{K}[\underline{X}]}. \qquad \square$$

A.2 Commutative semigroups

A *semigroup* is a set together with an associative binary operation $(G, *)$, as usual we will omit the operation, i. e. write ab instead of $a * b$. A monoid is a semigroup with a neutral element ε; one can always adjoin a neutral element to a semigroup if desired and since we will only consider monoids, we use these notions exchangeably. Our binary operation will be commutative in most interesting cases, hence we are really studying "commutative monoids".

Example A.3 *(Free (commutative) monoids).* The most well-known semigroup in computer science is the *free monoid* Σ^* over a finite alphabet Σ, consisting of strings of letters from Σ with string concatenation as the composition.

Similarly, the free commutative monoid over the alphabet $\Sigma = \{x_1, \ldots, x_n\}$ is the set of (commuting) monomials

$$\mathrm{Mon}(\Sigma) = \left\{ x_1^{d_1} \cdots x_n^{d_n} \;\middle|\; d_1, \ldots, d_n \geq 0 \right\}.$$

This semigroup is isomorphic to the commutative monoid of integer vectors $(\mathbb{N}^n, +)$. ⌟

A homomorphism of semigroups is a map of sets $f : G \to H$ with $f(ab) = f(a)f(b)$ for $a, b \in G$, in the case of monoids we also require that $f(\varepsilon_G) = \varepsilon_H$.

Remark. Notice that the concept of "kernel of a homomorphism" from group theory is not well-behaved in this general situation. For once, the kernel of $f : G \to H$ is usually defined as $f^{-1}(\varepsilon_H)$, and H needn't have a neutral element. And even if both G and H have neutral elements, then we may have $f^{-1}(\varepsilon_H) = \{\varepsilon_G\}$ even though f is *not* injective! Consider for

[1] This can be easily done for finite extension of $\mathbb{K}$, and is *always* possible assuming the axiom of choice.

example

$$f\colon \mathbb{N}^2 \to \mathbb{N}, \qquad f(a_1, a_2) \coloneqq a_1 + a_2,$$

then $f^{-1}(0) = \{(0,0)\}$, but in general $n \in \mathbb{N}$ has precisely $n+1$ preimages

$$f^{-1}(n) = \{(0,n), (1,n-1), \ldots, (n,0)\} \subseteq \mathbb{N}^2.$$

The previous example show that the naive definition of a kernel fails to measure the number of preimages in any meaningful way. But we can still define the following:

Definition A.4 (Congruence relation). A congruence relation on a semigroup G is an equivalence relation $\equiv \subseteq G \times G$ (i. e. reflexive, symmetric and transitive) such that for $a, a', b, b' \in G$

$$a \equiv a' \text{ and } b \equiv b' \quad \Longrightarrow \quad ab \equiv ab'. \qquad \lrcorner$$

If $f\colon G \to H$ is a homomorphism of semigroups, then

$$a \equiv_f b \quad :\Leftrightarrow \quad f(a) = f(b)$$

defines a congruence relation on G. Conversely, if $\equiv$ is a congruence relation on G, then the set of equivalence classes $G/\equiv$ forms a semigroup (with $[a]_\equiv [b]_\equiv \coloneqq [ab]_\equiv$) and the surjective homomorphism

$$\pi\colon G \to G/\equiv, \quad a \mapsto [a]_\equiv$$

defines the same equivalence relation $\equiv_\pi \,= \equiv$.

Definition A.5 (Commutative semigroup presentation). A semigroup presentation is a pair $\Sigma, \mathcal{R}$, where X is a set of generators and $\mathcal{R} \subseteq \Sigma^\oplus \times \Sigma^\oplus$ is a set of relations. The congruence relation $\equiv_\mathcal{R}$ generated by $\mathcal{R}$ is the smallest congruence relation on $\Sigma^\oplus$ containing $\mathcal{R}$. The commutative semigroup $\langle \Sigma \mid \mathcal{R} \rangle$ is the quotient of $\Sigma^\oplus / \equiv_\mathcal{R}$.

The presentation is finite if both X and $\mathcal{R}$ are finite. A commutative semigroup is finitely presented if it is isomorphic to $\langle \Sigma \mid \mathcal{R} \rangle$ for some finite presentation. $\lrcorner$

Lemma A.6. *If $\mathcal{R}$ is a set of relations, then $x \equiv_\mathcal{R} y$ if and only if there is a (possibly empty) sequence $x = x_0, x_1, \ldots, x_n = y$ such that*

$$x_{i-1} = \gamma_i \alpha_i, \quad x_i = \gamma_i \beta_i$$

for suitable $\gamma_i \in \Sigma^\oplus$ and (α_i, β_i) or (β_i, α_i) in $\mathcal{R}$.

Proof. $\equiv_\mathcal{R}$ contains $\mathcal{R}$, by symmetry all (β, α) for $(\alpha, \beta) \in \mathcal{R}$ and by reflexivity all (γ, γ), $\gamma \in \Sigma^\oplus$. Since it is a congruence relation, it contains all pairs $(x_{i-1}, x_i) = (\gamma_i \alpha_i, \gamma_i \beta_i)$ and hence by transitivity also (x, y) as in the lemma.

Conversely, it is easy to see that the relation

$$x \equiv' y \text{ if there exists a sequence as above}$$

is already a congruence relation, so the two relations coincide. □

It turns out that every finitely generated commutative monoid is in fact finitely presented!

Theorem A.7 (Rédei). *If G is finitely generated, i. e. we have a surjective homomorphism $f : \Sigma^{\oplus} \twoheadrightarrow G$ for a finite set Σ, then $\equiv_f$ is generated by finitely many relations.*

We can sketch a elegant and short proof due to Freyd [18]. If G is a monoid, then its monoid ring $\mathbb{Q}[G]$ is the $\mathbb{Q}$-algebra whose underlying vector space has G as a basis and the multiplication is defined by G on this basis and extended by bilinerarity. For example, $\mathbb{Q}[\{x_1, \ldots, x_n\}^{\oplus}]$ is just the polynomial ring in n variables.

Proof (sketch). Let $N := \Sigma^{\oplus} \times \Sigma^{\oplus}$. Assume that $\equiv_f$ is not finitely generated, then there exists an infinitely ascending sequence $\mathcal{R}_1 \subsetneq \mathcal{R}_2 \subsetneq \ldots$ contained in N such that $\equiv_{\mathcal{R}_i} \subsetneq \equiv_{\mathcal{R}_{i+1}}$ for all i. Let $G_i := \Sigma^{\oplus}/\equiv_{\mathcal{R}_i}$, then this gives rise to a chain of surjective but non-bijective homomorphisms

$$N \twoheadrightarrow G_1 \twoheadrightarrow G_2 \twoheadrightarrow \ldots.$$

These maps yield surjective ring homomorphisms $\mathbb{Q}[\Sigma^{\oplus}] \twoheadrightarrow \mathbb{Q}[G_i]$, let I_i be the kernel. By construction $I_1 \subsetneq I_2 \subsetneq \ldots$ is a strictly increasing chain of ideals in $\mathbb{Q}[\Sigma^{\oplus}] \cong \mathbb{Q}[X_1, \ldots, X_{|\Sigma|}]$, but this contradicts the Noetherian property of polynomial rings (Hilbert's basis theorem A.1)! □

Index

List of Symbols

$\mathbb{F}_q$	the finite field of order q (unique up to isomorphism)
$\mathbb{N}, \mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$	the natural numbers (with 0), the integers, the rational numbers, the real numbers and the complex numbers
$\underline{X}$	a set of variables $\{X_1, \ldots, X_n\}$
$\mathrm{Mon}_n, \mathrm{Mon}(S)$	the set of monomials in n variables and in S
$\mathrm{supp}(f)$	the monomials occurring in a polynomial f
$\mathsf{IM}_{\mathbb{K}}$	the ideal membership problem over the field $\mathbb{K}$
$\mathsf{HNST}_{\mathbb{K}}$	the decision problem to Hilbert's Nullstellensatz
$\prec, \prec_{\mathfrak{P}}$	a monomial order and the induced order on finite sets
$\prec_{\mathrm{lex}}, \prec_{\mathrm{grlex}}, \prec_{\mathrm{grevlex}}$	the lexicographic, graded lexicographic and graded reverse lexicographic order
$\mathrm{LT}(f), \mathrm{LM}(f), \mathrm{LC}(f)$	the leading term, leading monomial and leading coefficient of a polynomial f
$\mathrm{mdeg}(f)$	the multidegree of a polynomial f
$\mathrm{rem}(f; g_1, \ldots, g_s)$	the remainder of f with respect to $g_1, \ldots, g_s$
$\mathrm{IN}(I)$	the initial ideal generated by leading terms of I
$\mathrm{NF}_G(f), \mathrm{NF}_I(f)$	the set of normal forms of f with respect to a set G or an ideal I
$\mathsf{GROEBM}_{\mathbb{K}}$	the membership problem for reduced Gröbner bases of polynomials over $\mathbb{K}$
$\equiv_I, [f]_{\equiv_I}$	the congruence relation defined by the ideal I and the equivalence class of f
$W, \prec_W$	a weight matrix and the associated monomial order
$\gcd(X^\alpha, X^\beta), \mathrm{lcm}(X^\alpha, X^\beta)$	the greatest common divisor and least common multiple of two monomials
$\mathrm{Spoly}(f, g)$	the S-polynomial of f, g
${}^{\mathrm{h}}f, {}^{\mathrm{a}}g$	the homogenization of f and dehomogenization of g
$\mathcal{P}, \Rightarrow_{\mathcal{P}}, \equiv_{\mathcal{P}}$	a Thue system and the derivation relations
$\Sigma^{\oplus}$	the set of commutative words over Σ
$\vdash_C, \vdash_C^*$	the configuration transition relation for a (counter) machine C and its transitive reflexive closure

$\mathrm{rep}(q, c_1, c_2, c_3)$	the string representation of a configuration of a counter machine
$(\Sigma_n, \mathcal{P}_n)$	the commutative Thue system by Mayr & Meyer counting to 2^{2^n}
$\textsc{Lvl}(r)$	the set of words in $\Sigma_n^{\oplus}$ of level r
$\mathrm{box}(x)$	the box of x of the form $(S\|F)(C_1, \ldots, C_4)$
$\mathcal{G}(\mathcal{P})$	the graph associated to a commutative Thue system $\mathcal{P}$
$U_{\mathcal{P}}, A_{\mathcal{P}}$	the set of irreducible and reducible words with respect to a commutative Thue system $\mathcal{P}$

List of Definitions and Theorems

List of Figures

Bibliography

[1] Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge: Cambridge University Press, 2009. ISBN: 9780521424264. DOI: 10.1017/CBO9780511804090 (cit. on pp. 37, 41).

[2] Magali Bardet, Jean-Charles Faugère, and Bruno Salvy. "On the complexity of the F5 Gröbner basis algorithm". In: *Journal of Symbolic Computation* 70 (2015), pp. 49–70. ISSN: 0747-7171. DOI: https://doi.org/10.1016/j.jsc.2014.09.025 (cit. on p. 28).

[3] Dave Bayer and David Mumford. *What can be computed in algebraic geometry?* 1993. DOI: 10.48550/ARXIV.ALG-GEOM/9304003 (cit. on p. 71).

[4] David Bayer and Michael Stillman. "On the complexity of computing syzygies". In: *Journal of Symbolic Computation* 6.2 (1988), pp. 135–147. ISSN: 0747-7171. DOI: https://doi.org/10.1016/S0747-7171(88)80039-7 (cit. on pp. 54, 56, 60, 62).

[5] Lenore Blum et al. *Complexity and Real Computation*. Springer New York, Oct. 2012. 472 pp. ISBN: 1461268737. DOI: 10.1007/978-1-4612-0701-6 (cit. on p. 7).

[6] Paul Breiding and Sascha Timme. "HomotopyContinuation.jl: A Package for Homotopy Continuation in Julia". In: *Mathematical Software – ICMS 2018*. Ed. by James H. Davenport et al. Cham: Springer International Publishing, 2018, pp. 458–465. ISBN: 978-3-319-96418-8 (cit. on p. 72).

[7] Paul Breiding et al. *Nonlinear Algebra and Applications*. 2021. DOI: 10.48550/ARXIV.2103.16300 (cit. on p. 4).

[8] Bruno Buchberger. "Bruno Buchberger's PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal". In: *Journal of Symbolic Computation* 41.3 (2006). Logic, Mathematics and Computer Science: Interactions in honor of Bruno Buchberger (60th birthday), pp. 475–511. ISSN: 0747-7171. DOI: https://doi.org/10.1016/j.jsc.2005.09.007 (cit. on p. 26).

[9] Peter Bürgisser. "On the Parallel Complexity of the Polynomial Ideal Membership Problem". In: *Journal of Complexity* 14.2 (1998), pp. 176–189. ISSN: 0885-064X. DOI: https://doi.org/10.1006/jcom.1998.0472 (cit. on p. 62).

[10] Matthew Clegg, Jeff Edmonds, and Russell Impagliazzo. "Using the Groebner basis algorithm to find proofs of unsatisfiability". In: *Proceedings of STOC'96* (Mar. 2000). DOI: 10.1145/237814.237860 (cit. on p. 27).

[11] Laszlo Csanky. "Fast Parallel Matrix Inversion Algorithms". In: *SIAM Journal on Computing* 5.4 (1976), pp. 618–623. DOI: 10.1137/0205040 (cit. on p. 35).

[12] Alicia Dickenstein et al. "The membership problem for unmixed polynomial ideals is solvable in single exponential time". In: *Discrete Applied Mathematics* 33.1 (1991), pp. 73–94. ISSN: 0166-218X. DOI: https://doi.org/10.1016/0166-218X(91)90109-A (cit. on p. 30).

[13] Thomas Dubé, B. Mishra, and Chee-Keng Yap. "Admissible Orderings and Bounds for Gröbner Basis Normal Form Algorithms". In: (1986) (cit. on pp. 21, 33).

[14] Thomas W. Dubé. "The Structure of Polynomial Ideals and Gröbner Bases". In: *SIAM Journal on Computing* 19.4 (Aug. 1990), pp. 750–773. DOI: 10.1137/0219053 (cit. on p. 31).

[15] David Eisenbud and Bernd Sturmfels. "Binomial Ideals". In: *Duke Math. J.* 84 (Feb. 1994). DOI: 10.1215/S0012-7094-96-08401-X (cit. on p. 18).

[16] Jean Charles Faugère. "A New Efficient Algorithm for Computing Gröbner Bases without Reduction to Zero (F5)". In: *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*. ISSAC '02. Lille, France: Association for Computing Machinery, 2002, pp. 75–83. ISBN: 1581134843. DOI: 10.1145/780506.780516 (cit. on p. 28).

[17] Steven Fortune and James Wyllie. "Parallelism in Random Access Machines". In: *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*. STOC '78. San Diego, California, USA: Association for Computing Machinery, 1978, pp. 114–118. ISBN: 9781450374378. DOI: 10.1145/800133.804339 (cit. on p. 35).

[18] Peter J. Freyd. "Rédei's finiteness theorem for commutative semigroups". In: 1968. DOI: 10.1090/S0002-9939-1968-0227290-4 (cit. on p. 76).

[19] Gert-Martin Greuel and Gerhard Pfister. *A Singular Introduction to Commutative Algebra*. Springer Berlin Heidelberg, Nov. 2007. 712 pp. ISBN: 3540735410. DOI: 10.1007/978-3-540-73542-7 (cit. on pp. 3, 7, 13, 28, 30, 73).

[20] Grete Hermann. "Die Frage der endlich vielen Schritte in der Theorie der Polynomideale". In: *Mathematische Annalen* 95 (1926), pp. 736–788 (cit. on p. 29).

[21] Dung T. Huynh. "A Superexponential Lower Bound for Gröbner Bases and Church-Rosser Commutative Thue Systems". In: *Inf. Control.* 68 (1986), pp. 196–206 (cit. on pp. 41, 62, 64, 65).

[22] Oscar H. Ibarra, Shlomo Moran, and Louis E. Rosier. "A note on the parallel complexity of computing the rank of order n matrices". In: *Information Processing Letters* 11.4,5 (1980) (cit. on p. 35).

[23] Jürgen Gerhard Joachim von zur Gathen. *Modern Computer Algebra*. Cambridge University Press, Mar. 2017. 812 pp. ISBN: 1107039037. DOI: 10.1017/CBO9781139856065 (cit. on pp. 3, 4, 9).

[24] Gregor Kemper. *A course in commutative algebra*. Springer, 2011, p. 246. ISBN: 9783642035456 (cit. on pp. 1, 3, 7, 13, 23, 24, 73).

[25] Pascal Koiran. "Hilbert's Nullstellensatz Is in the Polynomial Hierarchy". In: *Journal of Complexity* 12.4 (1996), pp. 273–286. ISSN: 0885-064X. DOI: https://doi.org/10.1006/jcom.1996.0019 (cit. on p. 37).

[26] Janos Kollar. "Sharp Effective Nullstellensatz". In: *Journal of the American Mathematical Society* 1.4 (1988), pp. 963–975. ISSN: 08940347, 10886834. URL: http://www.jstor.org/stable/1990996 (cit. on p. 29).

[27] Ulla Koppenhagen and Ernst W. Mayr. "An Optimal Algorithm for Constructing the Reduced Gröbner Basis of Binomial Ideals". In: *Journal of Symbolic Computation* 28.3 (1999), pp. 317–338. ISSN: 0747-7171. DOI: https://doi.org/10.1006/jsco.1999.0285 (cit. on pp. 18, 19, 39).

[28] Martin Kreuzer and Lorenzo Robbiano. *Computational Commutative Algebra 1*. Springer Berlin Heidelberg, 2000. DOI: 10.1007/978-3-540-70628-1 (cit. on pp. 3, 14).

[29] Teresa Krick and Alessandro Logar. "Membership problem, Representation problem and the Computation of the Radical for one-dimensional Ideals". In: *Effective Methods in Algebraic Geometry*. Ed. by Teo Mora and Carlo Traverso. Boston, MA: Birkhäuser Boston, 1991, pp. 203–216. ISBN: 978-1-4612-0441-1. DOI: 10.1007/978-1-4612-0441-1_13 (cit. on p. 39).

[30] Klaus Kühnle and Ernst W. Mayr. "Exponential Space Computation of Gröbner Bases". In: *Proceedings of the 1996 International Symposium on Symbolic and Algebraic Computation*. ISSAC '96. Zurich, Switzerland: Association for Computing Machinery, 1996, pp. 63–71. ISBN: 0897917960. DOI: 10.1145/236869.236900 (cit. on pp. 17, 33).

[31] Ernst W. Mayr. "Membership in polynomial ideals over Q is exponential space complete". In: *STACS 89*. Ed. by B. Monien and R. Cori. Berlin, Heidelberg: Springer Berlin Heidelberg, 1989, pp. 400–406. ISBN: 978-3-540-46098-5 (cit. on p. 36).

[32] Ernst W. Mayr. "Some Complexity Results for Polynomial Ideals". In: *Journal of Complexity* 13.3 (1997), pp. 303–325. ISSN: 0885-064X. DOI: 10.1006/jcom.1997.0447 (cit. on pp. 23, 28, 41, 44).

[33] Ernst W. Mayr and Albert R. Meyer. "The complexity of the word problems for commutative semigroups and polynomial ideals". In: *Advances in Mathematics* 46.3 (Dec. 1982), pp. 305–329. DOI: 10.1016/0001-8708(82)90048-2 (cit. on pp. 29, 41, 50).

[34] Ernst W. Mayr and Stephan Ritscher. "Dimension-dependent bounds for Gröbner bases of polynomial ideals". In: *Journal of Symbolic Computation* 49 (2013). The International Symposium on Symbolic and Algebraic Computation, pp. 78–94. ISSN: 0747-7171. DOI: doi.org/10.1016/j.jsc.2011.12.018 (cit. on pp. 30, 31).

[35] Ernst W. Mayr and Stefan Toman. "Complexity of Membership Problems of Different Types of Polynomial Ideals". In: *Algorithmic and Experimental Methods in Algebra, Geometry, and Number Theory*. Ed. by Gebhard Böckle, Wolfram Decker, and Gunter Malle. Cham: Springer International Publishing, 2017, pp. 481–493. ISBN: 978-3-319-70566-8. DOI: 10.1007/978-3-319-70566-8_20 (cit. on pp. 23, 28, 37).

[36] Marvin L. Minsky. *Computation: Finite and Infinite Machines*. USA: Prentice-Hall, Inc., 1967. ISBN: 0131655639 (cit. on pp. 46, 49).

[37] H. Michael Möller and Ferdinando Mora. "Upper and lower bounds for the degree of Groebner bases". In: *EUROSAM 84*. Ed. by John Fitch. Berlin, Heidelberg: Springer Berlin Heidelberg, 1984, pp. 172–183. ISBN: 978-3-540-38893-7 (cit. on p. 30).

[38] Bernard Mourrain, Simon Telen, and Marc Van Barel. *Truncated Normal Forms for Solving Polynomial Systems: Generalized and Efficient Algorithms*. 2018. DOI: 10.48550/ARXIV.1803.07974 (cit. on p. 72).

[39] Victor Pan. "Complexity of parallel matrix computations". In: *Theoretical Computer Science* 54.1 (1987), pp. 65–85. ISSN: 0304-3975. DOI: https://doi.org/10.1016/0304-3975(87)90019-3 (cit. on p. 35).

[40] Behrooz Parhami. *Introduction to Parallel Processing: Algorithms and Architectures*. Boston, MA: Springer US, 2002. ISBN: 978-0-306-46964-0. DOI: 10.1007/0-306-46964-2_5 (cit. on p. 34).

[41] Suzanna Parkinson et al. *Analysis of Normal-Form Algorithms for Solving Systems of Polynomial Equations*. 2021. DOI: 10.48550/ARXIV.2104.03526 (cit. on p. 72).

[42] Emil L. Post. "Recursive Unsolvability of a problem of Thue". In: *Journal of Symbolic Logic* 12.1 (1947), pp. 1–11. DOI: 10.2307/2267170 (cit. on p. 42).

[43] Stephan Ritscher. "Degree Bounds and Complexity of Gröbner Bases of Important Classes of Polynomial Ideals". PhD thesis. Technische Universität München, 2012 (cit. on pp. 15, 20, 34, 36, 38).

[44] Lorenzo Robbiano. "Term orderings on the polynomial ring". In: *EUROCAL '85*. Ed. by Bob F. Caviness. Berlin, Heidelberg: Springer Berlin Heidelberg, 1985, pp. 513–517. ISBN: 978-3-540-39685-7 (cit. on p. 21).

[45] J. Maurice Rojas. "Dedekind Zeta Functions and the Complexity of Hilbert's Nullstellensatz". In: *arXiv: Number Theory* (2003) (cit. on p. 37).

[46] David Rolnick and Gwen Spencer. "On the robust hardness of Gröbner basis computation". In: *Journal of Pure and Applied Algebra* 223.5 (2019), pp. 2080–2100. ISSN: 0022-4049. DOI: https://doi.org/10.1016/j.jpaa.2018.08.016 (cit. on pp. 68, 69).

[47] Chee K. Yap. "A new lower bound construction for commutative thue systems with applications". In: *Journal of Symbolic Computation* 12.1 (1991), pp. 1–27. ISSN: 0747-7171. DOI: https://doi.org/10.1016/S0747-7171(08)80138-1 (cit. on p. 60).