Ideal Membership Problem for Boolean Minority and Dual Discriminator

Arpitha P. Bharathi
IDSIA-USI, Lugano, Switzerland

Monaldo Mastrolilli
IDSIA-SUPSI, Lugano, Switzerland

Abstract

The polynomial Ideal Membership Problem (IMP) tests if an input polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ with coefficients from a field $\mathbb{F}$ belongs to a given ideal $I \subseteq \mathbb{F}[x_1, \ldots, x_n]$. It is a well-known fundamental problem with many important applications, though notoriously intractable in the general case. In this paper we consider the IMP for polynomial ideals encoding combinatorial problems and where the input polynomial f has degree at most $d = O(1)$ (we call this problem IMP_d).

A dichotomy result between "hard" (NP-hard) and "easy" (polynomial time) IMPs was achieved for Constraint Satisfaction Problems over finite domains [6, 34] (this is equivalent to IMP_0) and IMP_d for the Boolean domain [23], both based on the classification of the IMP through functions called polymorphisms. For the latter result, there are only six polymorphisms to be studied in order to achieve a full dichotomy result for the IMP_d. The complexity of the IMP_d for five of these polymorphisms has been solved in [23] whereas for the ternary minority polymorphism it was incorrectly declared in [23] to have been resolved by a previous result. In this paper we provide the missing link by proving that the IMP_d for Boolean combinatorial ideals whose constraints are closed under the minority polymorphism can be solved in polynomial time. This completes the identification of the precise borderline of tractability for the IMP_d for constrained problems over the Boolean domain. We also prove that the proof of membership for the IMP_d for problems constrained by the dual discriminator polymorphism over any finite domain can also be found in polynomial time. Bulatov and Rafiey [8] recently proved that the IMP_d for this polymorphism is decidable in polynomial time, without needing a proof of membership. Our result gives a proof of membership and can be used in applications such as Nullstellensatz and Sum-of-Squares proofs.

2012 ACM Subject Classification Mathematics of computing → Gröbner bases and other special bases; Mathematics of computing → Combinatoric problems

Keywords and phrases Polynomial ideal membership, Polymorphisms, Gröbner basis theory, Constraint satisfaction problems

Digital Object Identifier 10.4230/LIPIcs.MFCS.2021.16

Related Version *Full Version*: https://arxiv.org/abs/2006.16422 [4]

Funding This research was supported by the Swiss National Science Foundation project 200020-169022 "Lift and Project Methods for Machine Scheduling Through Theory and Experiments".

1 Introduction

The study of polynomial ideals and related algorithmic problems goes back to David Hilbert [17]. The methods developed in this area to date find a wide range of applications in mathematics and computer science. In this paper we consider the polynomial Ideal Membership Problem, where we want to decide if a given polynomial belongs to a given ideal. This problem is a fundamental algorithmic problem with important applications in solving polynomial systems (see e.g. [12]), polynomial identity testing [12, 30] and underlies proof systems such as Nullstellensatz and Polynomial Calculus (see e.g. [2, 9, 15]).

46th International Symposium on Mathematical Foundations of Computer Science (MFCS 2021).
Editors: Filippo Bonchi and Simon J. Puglisi; Article No. 16; pp. 16:1–16:20
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

To introduce the problem formally, let $\mathbb{F}[x_1, \dots, x_n]$ be the ring of polynomials over a field $\mathbb{F}$ with indeterminates $x_1, \dots, x_n$. A polynomial *ideal* I is a subset of the polynomial ring $\mathbb{F}[x_1, \dots, x_n]$ with two properties: for any two polynomials f, g in I, $f + g$ also belongs to I and so does hf for any polynomial h. By the Hilbert Basis Theorem [16] every ideal I has a finite generating set $F = \{f_1, \dots, f_r\} \subset I$ such that for every $f \in \mathbb{F}[x_1, \dots, x_n]$, we have $f \in I$ if and only if there is an "ideal membership proof", namely a set of polynomials $\{h_1, \dots, h_r\} \subset \mathbb{F}[x_1, \dots, x_n]$ such that $f = h_1 f_1 + \ldots + h_r f_r$. The polynomial Ideal Membership Problem (IMP) is to find out if a polynomial f belongs to an ideal I or not, given a set F of generators of the ideal (we use IMP_d to denote IMP when the input polynomial f has degree at most $d = O(1)$). The IMP is, in general, notoriously intractable. The results of Mayr and Meyer show that it is EXPSPACE-complete [24, 25].

Semidefinite programming (SDP) relaxations have been a powerful technique for approximation algorithm design ever since Goemans and Williamson celebrated result of Max-Cut [14]. With the aim to construct stronger and stronger SDP relaxations, the Sum-of-Squares (SoS) hierarchy has emerged as the most promising set of relaxations (see e.g. [20]). However, we still do not know the answer to even very basic questions about its power. For example, we do not even know when SoS is guaranteed to run in polynomial time! As recently observed by O'Donnell [26], bounded degree SoS proof does not necessarily imply its low bit complexity, showing that the often repeated claim, that for any fixed degree SoS runs in polynomial time, is far from true. O'Donnell raised the open problem to establish useful conditions under which "small" SoS proof can be guaranteed. With this aim, a first elegant sufficient condition is due to Raghavendra and Weitz [27, 33]. For each instance $\mathcal{C}$ of a combinatorial problem, the set of polynomials that vanish at every point of the set of solutions of $\mathcal{C}$ is called the *combinatorial ideal* of $\mathcal{C}$ and denoted by $I_{\mathcal{C}}$. To satisfy Raghavendra and Weitz's criterion, it is necessary (but also sufficient) that the ideal membership problem IMP_d for each $I_{\mathcal{C}}$ is polynomial time solvable and that ideal membership proofs can be efficiently found too.[1] So the tractability of the ideal membership proof ensures that SoS runs in polynomial time for combinatorial problems. This is currently the only known general result that addresses the SoS bit complexity issue. However, the IMP_d tractability criterion of Raghavendra and Weitz suffers from a severe limitation, namely it is not clear which restrictions on combinatorial problems can guarantee an efficient computation of the IMP_d proofs for combinatorial ideals.

The Constraint Satisfaction Problem (CSP) provides a general framework for a wide range of combinatorial problems, where we are given a set of variables and a set of constraints, and we have to decide whether the variables can be assigned values that satisfy the constraints. There are useful connections between IMP_d and the CSP: for example a CSP instance $\mathcal{C}$ is unsatisfiable if and only if $1 \in I_{\mathcal{C}}$. It follows that CSP is just the special case of IMP_d with $d = 0$ (see Appendix A.1 for more details on Ideal-CSP correspondence). Restrictions on CSPs, called $\text{CSP}(\Gamma)$, in which the type of constraints is limited to relations from a set Γ, have been successfully applied to study the computational complexity classification (and other algorithmic properties) of CSPs (see [7] for an excellent survey).

Motivated by the aforementioned issue of Raghavendra and Weitz criterion, Mastrolilli [23] initiated a systematic study of the IMP_d tractability for combinatorial ideals of the form $\text{IMP}_d(\Gamma)$ arising from combinatorial problems from $\text{CSP}(\Gamma)$ for a set of relations Γ over the Boolean domain. The classic dichotomy result of Schaefer [29] gives the complexity of $\text{CSP}(\Gamma)$ (and therefore of $\text{IMP}_0(\Gamma)$) for the Boolean domain: $\text{CSP}(\Gamma)$ is solvable in polynomial time

[1] Note that answering whether a polynomial belongs to a certain ideal does not necessarily mean finding an ideal membership proof of that.

if all constraints are closed under one of six polymorphisms (majority, minority, MIN, MAX, constant 0 and constant 1), else it is NP-complete. Mastrolilli [23] claimed a dichotomy result for the $\mathrm{IMP}_d(\Gamma)$ for the Boolean domain: for any constant $d \geq 1$, the $\mathrm{IMP}_d(\Gamma)$ of Boolean combinatorial ideals is solvable in polynomial time if all constraints are closed under one of four polymorphisms (majority, minority, MIN, MAX), else it is coNP-complete. In [23], for three polymorphisms (majority, MIN, MAX), it is shown that $\mathrm{IMP}_d(\Gamma)$ is polynomial time solvable, and moreover ideal membership proofs can be efficiently found, too. Whereas for the ternary minority polymorphism it was incorrectly declared to have been resolved by a previous result[2]. As a matter of fact the complexity of the $\mathrm{IMP}_d(\Gamma)$ for the ternary minority polymorphism is open. It was mistakenly assumed in [23] that computing the (mod 2) Gröbner basis in lexicographic order was sufficient to solve the IMP_d problem in polynomial time, but the issue is that we require polynomials to be over $\mathbb{R}$ and not GF(2).

We address these issues in this paper and therefore establish the full dichotomy result claimed in [23] (see [22] for an updated version of the paper). To ensure efficiency of the $\mathrm{IMP}_d(\Gamma)$ for the ternary minority polymorphism, it is sufficient to compute a d-truncated Gröbner basis in the graded lexicographic order (see Definition 8 and Appendix A for more details). This is achieved by first showing that we can easily find a Gröbner basis in the lexicographic order for the combinatorial ideal. Since polynomials in this Gröbner basis can have degrees up to n and coefficients of exponential size, we show how this basis can be converted to a d-truncated Gröbner basis in the graded lexicographic order in polynomial time, whose polynomials have degrees up to d and coefficients of constant size. This efficiently solves the $\mathrm{IMP}_d(\Gamma)$ for combinatorial ideals whose constraints are over a language Γ closed under the minority polymorphism. Together with the results in [23, 22], our result allows to complete the answer of the aforementioned question by allowing to identify the precise borderline of tractability of the Boolean $\mathrm{IMP}_d(\Gamma)$. Thus the following summarizes our first result of this paper:

▶ **Theorem 1.** *Let Γ be a constraint language over the Boolean domain that is closed under the minority polymorphism. For each instance $\mathcal{C}$ of $CSP(\Gamma)$, the d-truncated reduced Gröbner basis in the graded lexicographic monomial ordering of the combinatorial ideal $I_{\mathcal{C}}$ can be computed in $n^{O(d)}$ time.*

▶ **Corollary 2.** *If Γ is closed under the minority polymorphism, then the ideal membership proofs of $IMP_d(\Gamma)$ over the Boolean domain can be computed in polynomial time for $d = O(1)$.*

After the appearance of a preliminary version of this paper [4], Bulatov and Rafiey [8] have recently obtained exciting new results. For a finite domain $D = \{0, 1, \ldots, p-1\}$ with prime p elements, they consider the affine polymorphism $\otimes : D^3 \to D$ defined as $\otimes(a, b, c) = a - b + c \pmod p$ (it is easy to see that this is the minority polymorphism for the Boolean domain). By building on our approach, they prove that a d-truncated Gröbner basis can be computed in time $n^{O(d)}$ for any fixed prime p.

In [3], we began the generalization of $\mathrm{CSP}(\Gamma)$ (viz. $\mathrm{IMP}_0(\Gamma)$) by working on the corresponding $\mathrm{IMP}_d(\Gamma)$ for any $d = O(1)$ in the ternary domain, which expands the known set of tractable IMP_d cases by providing a suitable class of combinatorial problems. We considered problems constrained under the dual discriminator polymorphism and prove that we can find the reduced Gröbner basis of the corresponding combinatorial ideal in polynomial time. This ensures that we can check if any degree d polynomial belongs to the combinatorial ideal or not in polynomial time, and provide proof of membership if it does. Among the very interesting results obtained in [8], the authors show that the IMP_d is solvable in polynomial

[2] This was pointed out by Andrei Bulatov, Akbar Rafiey and Stanislav Živný.

time for *any* finite domain for problems constrained under the dual discriminator. This was done by eliminating permutation constraints in some sense through a pre-processing step and converting an instance $\mathcal{C} = (X, D, C)$ to an instance $\mathcal{C}' = (X', D, C')$ where $X' \subseteq X$ and $C' \subseteq C$. Moreover, a polynomial $f(X)$ was converted to a polynomial $f'(X')$ such that $f \in I_{\mathcal{C}}$ if and only if $f' \in I_{\mathcal{C}}$. They calculated a Gröbner basis of $I_{\mathcal{C}'}$, in polynomial time, which reflected the remaining constraints. This gives a proof of membership of f' in $I_{\mathcal{C}'}$ if it does belong to the ideal, but it is not yet known as to how to recover the proof of membership for f in $I_{\mathcal{C}}$. Meanwhile our results in [3] gives proof of membership, but is only constrained to a 3-element domain.

In this paper, we compute a Gröbner basis for the entire combinatorial ideal over a finite domain by showing that a Gröbner basis of the ideal associated with permutation constraints can also be calculated in polynomial time. We forego the pre-processing step of [8], include the permutation constraints and directly calculate a Gröbner basis of $I_{\mathcal{C}}$. The set of polynomials that the elements of the Gröbner basis can come from is polynomial in size and hence we show that a proof of membership can also be calculated in polynomial time as required in [28]. The following summarizes the second result of this paper:

▶ **Theorem 3.** *Let Γ be a constraint language over a finite domain D that is closed under the dual discriminator polymorphism. For each instance $\mathcal{C}$ of CSP(Γ), a Gröbner basis in the graded lexicographic monomial ordering of the combinatorial ideal $I_{\mathcal{C}}$ can be computed in time polynomial in the number of variables. The polynomials in this basis have degree at most $|D|$.*

▶ **Corollary 4.** *If Γ is closed under the dual discriminator polymorphism, then membership proofs for IMP(Γ), over a finite domain, can be computed in polynomial time.*

The study of CSP-related IMPs is in its early stages. The results obtained in this paper are steps towards the long term and challenging goal of extending the celebrated dichotomy results of CSP(Γ) for finite domain [6, 34] to IMP(Γ). This would provide a complete CSP-related characterization of when the IMP tractability criterion is applicable.

Due to space limitations, we provide a sketch for some of the proofs, and the complete proofs will be updated in [4]. To make the paper more self-contained, some essential background and standard (according to the book [12]) Gröbner basis notations can be found in Appendix A.

2 Preliminaries

Let D denote a finite set (*domain*). By a k-ary ***relation*** R on a domain D we mean a subset of the k-th cartesian power D^k; k is said to be the *arity* of the relation. We often use relations and (affine) varieties interchangeably since both essentially represent a set of solutions. A ***constraint language*** Γ over D is a set of relations over D. A constraint language is ***finite*** if it contains finitely many relations, and is *Boolean* if it is over the 2-element domain $\{0, 1\}$. A ***constraint*** over a constraint language Γ is an expression of the form $R(x_1, \ldots, x_k)$ where R is a relation of arity k contained in Γ, and the x_i are variables. A constraint is satisfied by a mapping ϕ defined on the x_i if $(\phi(x_1), \ldots, \phi(x_k)) \in R$.

▶ **Definition 5.** *The* (nonuniform) CONSTRAINT SATISFACTION PROBLEM (CSP) *associated with language Γ over D is the problem CSP(Γ) in which: an instance is a triple $\mathcal{C} = (X, D, C)$ where $X = \{x_1, \ldots, x_n\}$ is a set of n variables and C is a set of constraints over Γ with variables from X. The goal is to decide whether or not there exists a solution, i.e. a mapping $\phi : X \to D$ satisfying all of the constraints. We will use Sol($\mathcal{C}$) to denote the set of solutions of $\mathcal{C}$.*

Moreover, we follow the algebraic approach to Schaefer's dichotomy result [29] formulated by Jeavons [18] where each class of CSPs that are polynomial time solvable is associated with a polymorphism.

▶ **Definition 6.** *An operation $f : D^m \to D$ is a **polymorphism** of a relation $R \subseteq D^k$ if for any choice of m tuples from R (allowing repetitions), it holds that the tuple obtained from these m tuples by applying f coordinate-wise is in R. If this is the case we also say that f* preserves *R, or that R is* invariant *or* closed *with respect to f. A polymorphism of a constraint language Γ is an operation that is a polymorphism of every $R \in \Gamma$.*

For a given instance $\mathcal{C}$ of CSP(Γ), the vanishing ideal of its solution set, $\mathbf{I}(Sol(\mathcal{C}))$, is called its ***combinatorial ideal*** and is denoted by $I_{\mathcal{C}}$ (see Definition 20 in Appendix A). We call polynomials of the form $\Pi_{a\in D}(x_i - a)$ ***domain polynomials***, denoted by $dom(x_i)$. They describe the fact that $Sol(\mathcal{C}) \subseteq D^n$. For a more detailed Ideal-CSP correspondence we refer to Appendix A.1.

▶ **Definition 7.** *The* Ideal Membership Problem *associated with language Γ is the problem IMP(Γ) in which the input consists of a polynomial $f \in \mathbb{F}[X]$ and a CSP(Γ) instance $\mathcal{C} = (X, D, C)$. The goal is to decide whether f lies in the combinatorial ideal $I_{\mathcal{C}}$. We use IMP$_d$(Γ) to denote IMP(Γ) when the input polynomial f has degree at most d.*

The Gröbner basis G of an ideal is a set of generators such that $f \in \langle G\rangle \iff f|_G = 0$, where $f|_G$ denotes the remainder of f divided by G (see [12] or Appendix A.2 for more details and notations).

▶ **Definition 8.** *If G is a Gröbner basis of an ideal in $\mathbb{F}[x_1, \ldots, x_n]$, the **d-truncated Gröbner basis** G' of G is defined as*

$$G' = G \cap \mathbb{F}[x_1, \ldots, x_n]_d,$$

where $\mathbb{F}[x_1, \ldots, x_n]_d$ is the set of polynomials of degree less than or equal to d.

It is not necessary to compute a Gröbner basis of $I_{\mathcal{C}}$ in its entirety to solve the IMP$_d$. Since the input polynomial f has degree $d = O(1)$, the only polynomials from G that can possibly divide f in the graded lexicographic order (see Definition 26 in Appendix A.2), are those that are in G'. The remainders of such divisions are also in $\mathbb{F}[x_1, \ldots, x_n]_d$. Therefore, by Proposition 32 and Corollary 33, the membership test can be computed by using only polynomials from G' and therefore we have

$$f \in I_{\mathcal{C}} \cap \mathbb{F}[x_1, \ldots, x_n]_d \iff f|_{G'} = 0.$$

From the previous observations it follows that if we can compute G' in $n^{O(d)}$ time then this yields an algorithm that runs in $n^{O(d)}$ time for the IMP$_d$ (note that the size of the input polynomial f is bounded by $n^{O(d)}$).

3 Boolean Minority

The Boolean Minority polymorphism is an affine polymorphism defined as follows. Note that there is only one such polymorphism for the Boolean domain.

▶ **Definition 9.** *For a finite domain D, a ternary operation $\otimes$ is called a minority polymorphism if $\otimes(a, a, b) = \otimes(a, b, a) = \otimes(b, a, a) = b$ for all $a, b \in D$.*

3.1 Gröbner bases in lex order

Consider an instance $\mathcal{C} = (X = \{x_1, \dots, x_n\}, D = \{0, 1\}, C)$ of CSP(Γ) where Γ is $\otimes$-closed. Any constraint of $\mathcal{C}$ can be written as a system of linear equations over GF(2) (see e.g. [10]). These linear systems with variables $x_1, \dots, x_n$ can be solved by Gaussian elimination. If there is no solution, then we have from Hilbert's Weak Nullstellensatz (Theorem 25) that $1 \in I_{\mathcal{C}} \iff \mathit{Sol}(\mathcal{C}) = \emptyset \iff I_{\mathcal{C}} = \mathbb{R}[\mathbf{x}]$. If $1 \in I_{\mathcal{C}}$ the reduced Gröbner basis is $\{1\}$. We proceed only if $\mathit{Sol}(\mathcal{C}) \neq \emptyset$. In this section, we assume the lex order $>_{\mathsf{lex}}$ with $x_1 >_{\mathsf{lex}} x_2 >_{\mathsf{lex}} \cdots >_{\mathsf{lex}} x_n$. We also assume that the linear system has $r \leq n$ equations and is already in its reduced row echelon form with x_i as the leading monomial of the i-th equation. Let $\mathit{Supp}_i \subset [n]$ such that $\{x_j : j \in \mathit{Supp}_i\}$ is the set of variables appearing in the i-th equation of the linear system except for x_i. Let the i-th equation be $R_i = 0 \pmod 2$ where

$$R_i := x_i \oplus f_i, \tag{1}$$

with $i \in [r]$ and f_i is the Boolean function $(\bigoplus_{j \in \mathit{Supp}_i} x_j) \oplus \alpha_i$ and $\alpha_i = 0/1$.

3.2 From (mod 2) to regular arithmetic Gröbner basis

In this section, we show how to transform R_i's into polynomials in regular arithmetic. The idea is to map R_i to a polynomial R'_i over $\mathbb{R}[x_1, \dots, x_n]$ such that $a \in \{0, 1\}^n$ satisfies $R_i = 0$ if and only if a satisfies $R'_i = 0$. Moreover, R_i is such that it has the same leading term as R'_i. We produce a set of polynomials G_1 and prove that G_1 is the reduced Gröbner basis of $I_{\mathcal{C}}$ over $\mathbb{R}[x_1, \dots, x_n]$ in the lex ordering. We define R'_i as

$$R'_i := x_i - M(f_i) \tag{2}$$

where

$$M(f_i) = \begin{cases} \sum_{k=1}^{|\mathit{Supp}_i|} \left((-1)^{k-1} \cdot 2^{k-1} \sum_{\{x_{j_1}, \dots, x_{j_k}\} \subseteq \mathit{Supp}_i} x_{j_1} x_{j_2} \cdots x_{j_k} \right) & \text{when } \alpha_i = 0 \\ 1 + \sum_{k=1}^{|\mathit{Supp}_i|} \left((-1)^{k} \cdot 2^{k-1} \sum_{\{x_{j_1}, \dots, x_{j_k}\} \subseteq \mathit{Supp}_i} x_{j_1} x_{j_2} \cdots x_{j_k} \right) & \text{when } \alpha_i = 1 \end{cases} \tag{3}$$

▶ **Lemma 10.** *Consider the following set of polynomials:*

$$G_1 = \{R'_1, \dots, R'_r, x_{r+1}^2 - x_{r+1}, \dots, x_n^2 - x_n\}, \tag{4}$$

where R'_i is from Equation (2). G_1 is the reduced Gröbner basis of $I_{\mathcal{C}}$ in the lexicographic order $x_1 >_{\mathsf{lex}} x_2 >_{\mathsf{lex}} \dots, >_{\mathsf{lex}} x_n$.

Proof. For any two Boolean variables x and y,

$$x \oplus y = x + y - 2xy. \tag{5}$$

By repeatedly using Equation (5) to obtain the equivalent expression for f_i, we see that $R_i = 0 \pmod 2$ and $R'_i = 0$ have the same set of 0/1 solutions. Therefore $\mathbf{V}(\langle G_1 \rangle)$ is equal to $\mathit{Sol}(\mathcal{C})$. This implies that $\langle G_1 \rangle \subseteq I_{\mathcal{C}}$. Moreover, $\mathrm{LM}(R_i) = \mathrm{LM}(R'_i) = x_i$, by construction. For every pair of polynomials in G_1 the reduced S-polynomial is zero as the leading monomials of any two polynomials in G_1 are relatively prime. By Buchberger's Criterion (see Theorem 36) it follows that G_1 is a Gröbner basis of $\langle G_1 \rangle$ over $\mathbb{R}[x_1, \dots, x_n]$ (according to the lex order).

In fact, it can be seen by inspection that G_1 is the *reduced* Gröbner basis of $\langle G_1 \rangle$. To prove that $I_{\mathcal{C}} = \langle G_1 \rangle$, we need to prove that any $p \in I_{\mathcal{C}} \implies p \in \langle G_1 \rangle$. It is enough to prove that $p|_{G_1} = 0$ as this implies $p \in \langle G_1 \rangle$. We have that $p|_{G_1}$ cannot contain variable x_i for all $1 \leq i \leq r$. Hence $p|_{G_1}$ is multilinear in $x_{r+1}, x_{r+2}, \ldots, x_n$. Each tuple of D^{n-r} extends to exactly that $n-$tuple in $\mathit{Sol}(\mathcal{C})$ whose coordinate associated with x_i $(1 \leq i \leq r)$ is the unique value x_i takes to satisfy $x_i \oplus f_i = 0$ (see Equation (1) and Equation (2)). As $p|_{G_1}$ is multilinear in $x_{r+1}, x_{r+2}, \ldots, x_n$, there are at most 2^{n-r} coefficients. Since every point of D^{n-r} is a solution of $p|_{G_1}$, we see that every coefficeint of $p|_{G_1}$ is zero and hence $p|_{G_1}$ is the zero polynomial. Hence G_1 is the reduced Gröbner basis of $I_{\mathcal{C}}$. ◀

Note that the reduced Gröbner basis in Equation (4) can be "efficiently" computed by exploiting the high degree of symmetry in each $M(f_i)$ and using elementary symmetric polynomials with variables from Supp_i.

3.3 Computing a truncated Gröbner basis

Now that we have the reduced Gröbner basis in lex order, we show how to obtain the d-truncated reduced Gröbner basis in grlex order in polynomial time for any fixed $d = O(1)$. Before we describe our conversion algorithm, we show how to expand a product of Boolean functions. This expansion will play a crucial step in our algorithm.

3.3.1 Expansion of a product of Boolean functions

In this section, we show a relation between a product of Boolean functions and (mod 2) sums of the Boolean functions, which is heavily used in our conversion algorithm in Section 3.3.2. We have already seen from Equation (5) that if f, g are two Boolean functions,[3] then

$$2 \cdot f \cdot g = f + g - (f \oplus g).$$

Hence it can be proved by repeated use of the above equation that the following holds for Boolean functions f_1, $f_2, \ldots, f_m$:

$$f_1 \cdot f_2 \cdots f_m = \frac{1}{2^{m-1}} \Bigg[\sum_{i \in [m]} f_i - \sum_{\{i,j\} \subset [m]} (f_i \oplus f_j) + \sum_{\{i,j,k\} \subset [m]} (f_i \oplus f_j \oplus f_k) + \cdots + (-1)^{m-1} (f_1 \oplus f_2 \oplus \cdots \oplus f_m) \Bigg]. \tag{6}$$

We call each Boolean function of the form $(f_{i_1} \oplus \cdots \oplus f_{i_k})$ in Equation (6) as a **Boolean term**. We call the Boolean term $(f_1 \oplus f_2 \oplus \cdots \oplus f_m)$ as the **longest Boolean term** of the expansion. Thus, a product of Boolean functions can be expressed as a linear combination of Boolean terms. Note that Equation (6) is *symmetric* with respect to $f_1, f_2, \ldots, f_m$ as any f_i interchanged with f_j produces the same expression. It is no coincidence that we chose the letter f in the above equation: we later apply this identity using f_j from $R_j := x_j \oplus f_j$ (see Section 3.1). When we use Equation (6) in the conversion algorithm, we will have to evaluate a product of at most d functions, i.e. $m \leq d = O(1)$. We now see in the right hand side of Equation (6) that the coefficient $1/2^{m-1}$ is of constant size and there are $O(1)$ many Boolean terms.

[3] We earlier considered Boolean variables, but the same holds for Boolean functions.

3.3.2 Our conversion algorithm

The FGLM [13] conversion algorithm is well known in computer algebra for converting a given reduced Gröbner basis of a zero dimensional ideal in some ordering to the reduced Gröbner basis in any other ordering. However, it does so with $O(nD(\langle G_1\rangle)^3)$ many arithmetic operations, where $D(\langle G_1\rangle)$ is the dimension of the $\mathbb{R}$-vector space $\mathbb{R}[x_1,\ldots,x_n]/\langle G_1\rangle$ (see Proposition 4.1 in [13]). $D(\langle G_1\rangle)$ is also equal to the number of common zeros (with multiplicity) of the polynomials from $\langle G_1\rangle$, which would imply that for the combinatorial ideals considered in this paper, $D(\langle G_1\rangle) = O(2^{n-r})$. This exponential running time is avoided in our conversion algorithm, which is a variant the FGLM algorithm, by exploiting the symmetries in Equation (3) and by truncating the computation up to degree d.

Some notations necessary for the algorithm are as follows: G_1 and G_2 are the reduced Gröbner basis of $\langle G_1\rangle$ in lex and grlex ordering respectively. $\mathrm{LM}(G_i)$ is the set of leading monomials of polynomials in G_i for $i \in \{1,2\}$. Since we know G_1, we know $\mathrm{LM}(G_1)$, whereas G_2 and $\mathrm{LM}(G_2)$ are constructed by the algorithm. $B(G_1)$ is the set of monomials that cannot be divided (considering the lex order) by any monomial of $\mathrm{LM}(G_1)$. Therefore, $B(G_1)$ is the set of all multilinear monomials in variables $x_{r+1},\ldots,x_n$. Similarly, $B(G_2)$ is the set of monomials that cannot by divided (considering the grlex order) by any monomial of $\mathrm{LM}(G_2)$. Recall the definition of f_i for $i \leq r$ from Section 3.1. For $i > r$, for notational purposes, we define the Boolean function $f_i := x_i$.

▶ **Lemma 11.** *Consider a monomial q such that $deg(q) \leq d$. Then $q|_{G_1}$ can be expressed as a linear combination of Boolean terms.*

Proof. Consider $q = x_{i_1}x_{i_2}\cdots x_{i_k}$ where $k \leq d$. Then from Equations (1) and (2), $q|_{G_1} = f_{i_1}f_{i_2}\cdots f_{i_k}$ and the lemma holds using Equation (6). ◀

Let elements b_i of $B(G_2)$ be arranged in increasing grlex order. We construct a set A in our algorithm such that its elements a_i are defined as $a_i = b_i|_{G_1}$ written as linear combinations of Boolean terms using Lemma 11. We say that a Boolean term f of a_i "appears in a_j" for some $j < i$ if the longest Boolean term of a_j is $f \oplus \alpha$ where $\alpha = 0/1$.

Let Q be the set of all monomials m such that $1 <_{\mathsf{grlex}} deg(m) \leq_{\mathsf{grlex}} d$. We recommend the reader to refer to the example in [4] for an intuitive working of the algorithm. The conversion is described in full in Algorithm 1 (we assume $1 \notin I_{\mathcal{C}}$, else $G_1 = \{1\} = G_2$ and we are done).

▶ **Lemma 12.** *The set A is such that every a_i is a linear combination of existing $b_j|_{G_1}$'s ($j < i$) and the longest Boolean term of $b_i|_{G_1}$.*

Proof. By definition, element a_i is added to A when a monomial q is added to $B(G_2)$ where $b_i = q$ and $a_i = b_i|_{G_1}$ expressed in Boolean terms (see Algorithm 1). This means that q is not divisible by any monomial in $\mathrm{LM}(G_2)$. We prove the lemma by induction on the degree of q. Note that $b_1 = 1$ and hence $a_1 = b_1|_{G_1} = 1$.

If $deg(q) = 1$, then q is some x_i and $x_i|_{G_1}$ is one of $0, 1$ or f_i. If $x_i|_{G_1}$ is either 0 or 1, then it appears in a_1. We are now in the "else" condition of Algorithm 1, so q should be added to $\mathrm{LM}(G_2)$ and not $B(G_2)$. Hence $x_i|_{G_1}$ can be neither 0 nor 1 and the lemma holds for $deg(q) = 1$ as f_i is the longest Boolean term.

Let us assume the statement holds true for all monomials with degree less than m. Consider q such that $deg(q) = m$ and $q = x_{i_1}x_{i_2}\ldots x_{i_m}$ where i_j's need not be distinct, and the lemma holds for every monomial $<_{\mathsf{grlex}}$ q. Then $q|_{G_1} = f_{i_1}\cdot f_{i_2}\cdots f_{i_m}$. Let $(f_{j_1}\oplus\cdots\oplus f_{j_k})$ be a Boolean term in the expansion of $q|_{G_1}$ (by using Equation (6)), that

Algorithm 1 Computing the d-truncated reduced Gröbner basis.

```
Input: Degree $d$, $G_1$, $Q$.
Output: $d$-Truncated versions of $G_2$, $B(G_2)$.
Initialization: $G_2 = \emptyset$, $B(G_2) = \{b_1 = 1\}$, $A = \{a_1 = 1\}$.
while $Q \neq \emptyset$ do
    Let $q$ be the smallest (according to grlex order) monomial in $Q$.
    Find $q|_{G_1}$, by which we simply replace any occurrence of $x_i$ by the Boolean
      functions $f_i$.
    Expand $q|_{G_1}$ by using Equation (6).
    if the longest Boolean term of $q|_{G_1}$ does not appear in any $a \in A$ then
        Write $q|_{G_1}$ as a linear combination of $b_i|_{G_1}$ and its longest Boolean term (see
          Lemma 12).
        Add this polynomial to $A$ and add $q$ to $B(G_2)$.
    else
        Every Boolean term of $q|_{G_1}$ can be written as linear combinations of $b_j|_{G_1}$'s.
          Note that if the longest Boolean term $f$ appears in $a$ as $f \oplus 1$, then we use
          $f \oplus 1 = 1 - (f)$ (see Equation (5)). Thus we have
          $q|_{G_1} = \sum_j k_j b_j|_{G_1} \implies q - \sum_j k_j b_j \in \langle G_1 \rangle$.
        Add the polynomial $q - \sum_j k_j b_j$ to $G_2$ and $q$ to $\mathrm{LM}(G_2)$.
        Delete any monomial in $Q$ that $q$ can divide.
    Delete $q$ from $Q$.
$G_2$ is the $d$-truncated reduced Gröbner basis.
```

is not the longest Boolean term, so $\{j_1, \dots, j_k\} \subset \{i_1, \dots, i_m\}$ and $k < m$. Consider the monomial $x_{j_1} x_{j_2} \dots x_{j_k}$. We will now prove that $x_{j_1} x_{j_2} \dots x_{j_k}$ is in fact some $b_l \in B(G_2)$ and there exists $a_l \in A$ which is a linear combination of $b_i|_{G_1}$'s and $(f_{j_1} \oplus \dots \oplus f_{j_k})$. The monomial $x_{j_1} x_{j_2} \dots x_{j_k}$ either belongs to $LM(G_2)$ or $B(G_2)$. If $x_{j_1} x_{j_2} \dots x_{j_k} \in \mathrm{LM}(G_2)$ then it divides q, a contradiction to our choice of q. Therefore, $x_{j_1} x_{j_2} \dots x_{j_k} = b_l \in B(G_2)$. Clearly $b_l <_{\mathsf{grlex}} q$ and the induction hypothesis applies, so there exists $a_l \in A$ such that

$$b_l|_{G_1} = a_l = \sum_{i<l} c_i b_i|_{G_1} + c_0(f_{j_1} \oplus \dots \oplus f_{j_k})$$

where c_i's are constants. Then we simply use the above equation to substitute for the Boolean term $f_{j_1} \oplus \dots \oplus f_{j_k}$ in $q|_{G_1}$ as a linear combination of $b_i|_{G_1}$ where $i \leq l$. We can do this for every Boolean term of $q|_{G_1}$ except the longest one. Hence the lemma holds. ◀

▶ **Theorem 13.** *The conversion algorithm terminates for every input G_1 and correctly computes a d-truncated reduced Gröbner basis, with the grlex ordering, of the ideal $\langle G_1 \rangle$ in polynomial time.*

Proof. Algorithm 1 runs at most $|Q| = O(n^d)$ times. Evaluation of any $q|_{G_1}$ can be done in $O(n)$ steps (see Equation (6)), checking if previous a_i's appear (and replacing every Boolean term appropriately if it does) takes at most $O(n^d)$ steps since there are at most $|Q|$ many elements in A. Hence the running time of the algorithm is $O(n^{2d})$.

Suppose the set of polynomials $\{g_1, g_2, \dots, g_k\}$ is the output of the algorithm for some input G_1. Clearly, $deg(g_i) \leq d$ for all $i \in [k]$. We now prove by contradiction that the output is the d-truncated Gröbner basis of the ideal $\langle G_1 \rangle$ with the grlex ordering. Suppose g is a polynomial of the ideal with $deg(g) \leq d$, but no $\mathrm{LM}(g_i)$ can divide $\mathrm{LM}(g)$. In fact, since every $g_i \in \langle G_1 \rangle$ we can replace g by $g|_{\{g_1,g_2,\dots,g_k\}}$ (g generalises the reduced S-polynomial). The fact that $g \in \langle G_1 \rangle$ and $g|_{G_1} = 0$ implies that $\mathrm{LM}(g)$ is a linear combination of monomials that are less than $\mathrm{LM}(g)$ (in the grlex order) and hence must be in $B(G_2)$, i.e

$$g|_{G_1} = 0 \implies \mathrm{LM}(g)|_{G_1} = \sum_i k_i b_i|_{G_1}$$

where every $b_i \in B(G_2)$ and $b_i <_{\mathsf{grlex}} \mathrm{LM}(g)$. When the algorithm runs for $q = \mathrm{LM}(g)$, since q was not added to $\mathrm{LM}(G_2)$,

$$\mathrm{LM}(g)|_{G_1} = \sum_j k_j b_j|_{G_1} + f$$

where f is the longest Boolean term of $\mathrm{LM}(g)|_{G_1}$ which does not appear in any previous element of A. But the two equations above imply that $\sum_i k_i b_i|_{G_1} = \sum_j k_j b_j|_{G_1} + f$, which proves that there exists some $b_l \in B(G_2)$ such that a_l has f as its longest Boolean term, so f should have appeared in a_l, a contradiction. Therefore the output is a d-truncated Gröbner basis. Although unnecessary for the IMP_d, we also prove that the output is reduced: every non leading monomial of every polynomial in the output comes from $B(G_2)$ and no leading monomial is a multiple of another by construction. ◀

Thus we have proof of Theorem 1 and Corollary 2.

4 Dual discriminator

We assume in this section that the solution set is non-empty, $D \subset \mathbb{F}$ is any finite domain and the polymorphism in question is the dual discriminator ∇. The dual discriminator is a majority polymorphism [19, 1] and is often used as a starting point in many CSP-related classifications [1]. For a finite domain D, a ternary operation f is called a majority polymorphism if $f(a, a, b) = f(a, b, a) = f(b, a, a) = a$ for all $a, b \in D$.

▶ **Definition 14.** *The dual discriminator, denoted by ∇, is a majority polymorphism such that $\nabla(a, b, c) = a$ for pairwise distinct $a, b, c \in D$.*

The constraints for ∇-closed problems can be assumed to be binary [19] and are of three types: permutation constraints, complete constraints and two-fan constraints [31, 11]. Bulatov and Rafiey [8] recently proved that the $\mathrm{IMP}(\Gamma)$ over a finite domain is decidable in polynomial time, without showing a proof of membership. They did so by cleverly eliminating the permutation constraints, but were unable to recover a proof for the original problem. We show that a Gröbner basis of the ideal restricted to the permutation constraints can be computed in polynomial time in Section 4.1. We then show in Section 4.2 that the Gröbner basis of constraints that are complete and two-fan constraints can come from a fixed set (see Definition 17). We prove in Section 4.3 that the Gröbner basis of the entire ideal can be found in polynomial time. This Gröbner basis is independent of degree d of the input polynomial: it only contains polynomials with degree less than or equal to $|D|$. Due to space constraints, we give a gist of the proofs as the full proofs will be updated in [4].

4.1 Permutation constraints

A permutation constraint is of the form $R(x_i, x_j)$ where $R = \{(a, \pi_{ij}(a)) \mid a \in D_{ij}\}$ for some $D_{ij} \subseteq D$ and some bijection $\pi_{ij} : D_{ij} \to D'_{ij}$, where $D'_{ij} \subseteq D$. Let $\mathcal{P}$ be the set of input permutation constraints. We can assume that there exists at most one permutation constraint over every pair of variables: if there are two on the same set of variables, then their intersection is a permutation constraint. Let $R_{ij}(x_i, x_j)$ represent the unique permutation constraint on variables x_i, x_j, if one exists.

Informally, the goal is to make larger constraints called *chained permutation constraints* (CPC's). Permutation constraints on overlapping variables can be linked to form a larger constraint by using bijections. For example, if there exists $R_{ij}(x_i, x_j), R_{jk}(x_j, x_k) \in \mathcal{P}$, we form a new constraint on x_i, x_j, x_k by using π_{ij} and π_{jk}: the chained permutation constraint is $R(x_i, x_j, x_k)$ where

$$R = \{(\pi_{ij}^{-1}(a), a, \pi_{jk}(a)) \mid a \in D'_{ij} \cap D_{jk}\}.$$

The number of tuples in any CPC is always less than or equal to the domain size, since there is always a bijection between any two variables of a CPC. The constructing of CPC's can be carried out by the arc consistency algorithm described in [21]. A brief working of the algorithm tailored to our application is as follows: let $J \subset [n]$ be an index set for the CPC's (it becomes clear later why there can be at most $\lfloor n/2 \rfloor$ of them but we use n for convenience). We initialise $J = \emptyset$. A general chained permutation constraint CPC_i is defined as $R_i(X_i)$ where R_i is a relation and $X_i \subseteq X$ is a variable set. We keep track of the values that each variable is allowed to take, i.e., S_a is the set of solutions of x_a that satisfies CPC_i for all $x_a \in X_i$. The sets S_a and X_i are updated as CPC_i grows. We define $\sigma_{ab} : S_a \to S_b$ to be the bijection between any two pairs of variables $x_a, x_b \in X_i$. Hence $\sigma_{ba} = \sigma_{ab}^{-1}$. Let σ_{aa} denote the identity function for all $a \in [n]$. For any permutation constraint R_{pq} in $\mathcal{P}$, one of the four is true:

- neither x_p nor x_q belong to $\cup_{j \in J} X_j$: in which case we create a CPC. We define $\text{CPC}_i = R_{pq}(x_p, x_q)$ and $X_i = \{x_p, x_q\}$ where $i \in [n] \setminus J$.
- $x_p \in X_i$ and $x_q \notin \cup_{j \in J} X_j$, in which case we expand CPC_i to include R_{pq} and x_q is included in X_i.
- $x_p \in X_i$, $x_q \in X_j$ and $i \neq j$, in which case CPC_i and CPC_j have a permutation constraint linking two of their variables, so we combine the two CPC's into one. The set $X_i \cup X_j$ is the new X_i and j is deleted from J.
- both $x_p, x_q \in X_i$, in which case we update CPC_i to retain only the common solutions between CPC_i and R_{pq}.

As $R_{pq}(x_p, x_q)$ is now accounted for in some CPC, it is deleted from $\mathcal{P}$. The algorithm runs until $\mathcal{P}$ is empty. Once $\mathcal{P}$ is empty, CPC_i is defined as

$$\text{CPC}_i := R_i(X_i = \{x_{i_1}, x_{i_2}, \ldots, x_{i_r}\}) \text{ where } R_i = \{(a, \sigma_{i_1 i_2}(a), \ldots, \sigma_{i_1 i_r}(a)) \mid a \in S_{i_1}\}$$

for each $i \in J$.

▶ Remark 15. For $i \neq j$, $X_i \cap X_j = \emptyset$.

▶ **Lemma 16.** *Let I_{CPC_i} be the combinatorial ideal associated with CPC_i. A Gröbner basis of $\sum_i I_{\text{CPC}_i}$ can be calculated in polynomial time.*

The main idea behind the proof is as follows: suppose we see a relation as a matrix where each tuple is a row. Then the arity is equal to the number of columns. The relation R_i in $\text{CPC}_i = R_i(X_i)$ is such that it has at most $|D|!$ pairwise distinct columns, as there exists a bijection between every pair of variables in X_i. If the columns corresponding to x_j and x_k are the same, then the polynomial $x_j - x_k \in I_{\text{CPC}_i}$. We can separate these linear polynomials, and the problem reduces to finding a Gröbner basis of the ideal associated with a constraint that has at most $|D|!$ variables and $|D|$ tuples. This implies that a Gröbner basis can be computed where the polynomials have degree at most $|D|$. We in fact do not need to find the Gröbner basis of I_{CPC_i} yet. We show in the Section 4.3 as to how we can compute the rest of the polynomials in the Gröbner basis of $I_{\mathcal{C}}$ by using the relations R_i, the sets S_j, the bijections σ_{kl} and polynomials that define complete and two-fan constraints.

4.2 Complete and two-fan constraints

A complete constraint is of the form $R(x_i, x_j)$ where $R = D_i \times D_j$ for some $D_i, D_j \subseteq D$. The polynomials that can represent these constraints are $\prod_{a \in D_i}(x_i - a)$ and $\prod_{b \in D_j}(x_j - b)$. We call these polynomials *partial domain polynomials*. If no such input explicitly exists for a variable, the domain polynomial in that variable itself is the partial domain polynomial. A two-fan constraint is of the form $R(x_i, x_j)$ where $R = \{(\{a\} \times D_j) \cup (D_i \times \{b\})\}$ for some $D_i, D_j \subseteq D$ with $a \in D_i, b \in D_j$. This constraint can be represented by the set of polynomials $\{(x_i - a)(x_j - b), \prod_{c \in D_i}(x_i - c), \prod_{d \in D_j}(x_j - d)\}$.

▶ **Definition 17.** *The set of polynomials $\mathcal{D}$, $\mathcal{F}$ and $\mathcal{L}$ is defined as follows:*

$$\begin{aligned}
&\mathcal{D} = \{\Pi_{a \in A}(x_i - a) \mid i \in [n], A \subseteq D\},\\
&\mathcal{F} = \{(x_i - a)(x_j - b) \mid i, j \in [n], i \neq j\},\\
&\mathcal{L} = \{x_i - \alpha_2 - (x_j - \beta_2)(\alpha_1 - \alpha_2)/(\beta_1 - \beta_2) \mid i, j \in [n], i \neq j\},
\end{aligned}$$

for all $a, b, \alpha_1, \alpha_2, \beta_1, \beta_2 \in D$ where $\alpha_1 \neq \alpha_2$ and $\beta_1 \neq \beta_2$.

In other words $\mathcal{D} \cup \mathcal{F}$ is the set of all complete constraints and two-fan constraints and $\mathcal{L}$ is the set of polynomials in two variables that pass through two points $(\alpha_1, \beta_1), (\alpha_2, \beta_2) \in D^2$ where $\alpha_1 \neq \alpha_2$ and $\beta_1 \neq \beta_2$. Let $G \subset \mathcal{D} \cup \mathcal{F}$ be the set of polynomials that describes the input complete constraints and two-fan constraints of an instance of $\mathrm{IMP}_d(\Gamma)$. Let $I_{\mathrm{CF}} = \langle G \rangle$ (combinatorial ideal for the Complete and two-Fan constraints). Then

$$I_{\mathcal{C}} = \sum_{i \in J} I_{\mathrm{CPC}_i} + I_{\mathrm{CF}} = \sum_{i \in J} I_{\mathrm{CPC}_i} + \langle G \rangle .$$

▶ **Lemma 18.** *The reduced Gröbner basis of I_{CF} can be calculated in polynomial time and is a subset of $\mathcal{D} \cup \mathcal{F} \cup \mathcal{L}$.*

Proof sketch. For any pair $f, g \in G$, we show that there are polynomials $H \subset \langle G \rangle$ such that $H \subset \mathcal{D} \cup \mathcal{F} \cup \mathcal{L}$ and $S(f, g)|_H = 0$. We then include these polynomials in G, i.e. $G := G \cup H$. The cases already considered in Lemma 5.16 of [8] are when:

- $f, g \in \mathcal{F}$ where $f = (x_i - a)(x_j - b)$, $g = (x_i - c)(x_k - d)$ for $a = c$ and $a \neq c$,
- $f \in \mathcal{D}, g \in \mathcal{F}$ where $f = \Pi_{a \in D_i}(x_i - a)$, $g = (x_i - c)(x_j - b)$ and $c \in D_i$.

Of the remaining cases, the case that deserves most attention is when $f, g \in \mathcal{F}$ produces a permutation constraint (i.e., when $f = (x_i - a)(x_j - b)$ and $g = (x_i - c)(x_j - d)$ where $a \neq c$ and $b \neq d$).

Hence, the S-polynomial for every two polynomials in G is such that there are polynomials in I_{CF} that belong in $\mathcal{D} \cup \mathcal{F} \cup \mathcal{L}$ which reduce the S-polynomial to zero. In fact, it is not difficult to see that the reduced Gröbner basis is also a subset of $\mathcal{D} \cup \mathcal{F} \cup \mathcal{L}$. Since $|\mathcal{D} \cup \mathcal{F} \cup \mathcal{L}| = O(n^2)$, the reduced Gröbner basis of I_{CF} can be calculated in polynomial time. ◀

Algorithm 2 Calculating Gröbner basis.

```
   Input: $G, \mathrm{CPC}_i$.
   Output: Gröbner basis of $I_{\mathcal{C}}$.
 1 Compute and replace $G$ by the reduced Gröbner basis of $I_{\mathrm{CF}}$.
 2 for every $g = \Pi_{a \in D_p}(x_p - a) \in G \cap \mathcal{D}$ do
 3     if $D_p \neq S_p$ then
 4         $S_p := S_p \cap D_p$. Suppose $x_p \in X_i$.
 5         $S_k := \{\sigma_{pk}(a) \mid a \in S_p\}$ for every $x_k \in X_i \setminus \{x_p\}$.
 6         Replace $g$ by $\Pi_{a \in S_p}(x_p - a)$ in $G$. Go to Line 1.
 7 Let $C = G \cap \mathcal{F}$.
 8 while $C \neq \emptyset$ do
 9     Choose $g = (x_p - a)(x_q - b) \in C$. Suppose $x_p \in X_i$.
10     if $a \notin S_p$ then
11         Add $x_q - b$ to $G$ if $a \notin S_q$ else add $x_p - a$ to $G$. Go to Line 1.
       /* At this point $a \in S_p$ and $b \in S_q$.                              */
12     if $x_q \in X_j$ for some $i \neq j$ then
13         if $b \notin S_q$ then
14             Add $x_p - a$ to $G$. Go to Line 1.
           /* At this point $a \in S_p$ and $b \in S_q$.                          */
15         Let $B := \{(x_k - \sigma_{pk}(a))(x_l - \sigma_{ql}(b)) \mid x_k \in X_i, x_l \in X_j\} \setminus \{g\}$.
16         if $\exists h \in B$ such that $h|_G \neq 0$ then
17             $G := G \cup B$. Go to Line 1.
18     if $x_q \notin \cup_{j \in J} X_j$ then
19         Let $B := \{(x_k - \sigma_{pk}(a))(x_q - b) \mid x_k \in X_i\}$.
20         if $\exists h \in B$ such that $h|_G \neq 0$ then
21             $G := G \cup B$. Go to Line 1.
22     Delete $g$ from $C$.
23 Calculate $\mathcal{G}_i$ for every $i$.
24 A Gröbner basis of $I_{\mathcal{C}}$ is $\cup_i \mathcal{G}_i \cup G$.
```

4.3 Computing a Gröbner basis

▶ **Theorem 19.** *A Gröbner basis of the combinatorial ideal $I_{\mathcal{C}}$ can be calculated in polynomial time.*

Proof sketch. Let G be the reduced Gröbner basis of I_{CF}. Then,

$$I_{\mathcal{C}} = \sum_{i \in J} I_{\mathrm{CPC}_i} + I_{\mathrm{CF}} = \sum_{i \in J} \langle \mathcal{G}_i \rangle + \langle G \rangle .$$

For two polynomials $f, g \in \cup_i \mathcal{G}_i \cup G$, we see what the reduced S-polynomial can imply. The straightforward cases are when

- $f, g \in \mathcal{G}_i$: here $S(f,g)|_{\mathcal{G}_i} = 0$ since $\mathcal{G}_i$ is the reduced Gröbner basis of I_{CPC_i},
- $f \in \mathcal{G}_i, g \in \mathcal{G}_j$ where $i \neq j$: as f and g don't share any variable in common (see Remark 15), the leading monomials are relatively prime, hence $S(f,g)|_{\{f,g\}} = 0$,
- $f, g \in \mathcal{D} \cup \mathcal{F} \cup \mathcal{L}$: here $S(f,g)|_G = 0$ because of Lemma 18.

The only cases to examine is when $f \in \mathcal{G}_i$ and $g \in G \subset \mathcal{D} \cup \mathcal{F} \cap \mathcal{L}$. In each case, polynomials from $\cup_i \mathcal{G}_i$ and $\mathcal{D} \cup \mathcal{F} \cup \mathcal{L}$ reduce $S(f, g)$ to zero.

Clearly, this Gröbner basis is independent of degree d of the input polynomial. Hence, we have proof of Theorem 3 and Corollary 4. ◀

5 Conclusion

The IMP_d tractability for combinatorial ideals has useful practical applications as it implies bounded coefficients in Sum-of-Squares proofs. A dichotomy result between "hard" (NP-hard) and "easy" (polynomial time) IMPs was achieved for the IMP_0 [6, 34] over the finite domain nearly thirty years after that over the Boolean domain [29]. The IMP_d for $d = O(1)$ over the Boolean domain was tackled by Mastrolilli [23] based on the classification of the IMP through polymorphisms, where the complexity of the IMP_d for five of six polymorphisms was solved. We solve the remaining problem, i.e. the complexity of the $\text{IMP}_d(\Gamma)$ when Γ is closed under the ternary minority polymorphism. This is achieved by showing that the d-truncated reduced Gröbner basis can be computed in polynomial time, thus completing the missing link in the dichotomy result of [23]. We also show that a proof of membership can be found in polynomial time regarding the $\text{IMP}(\Gamma)$ for which constraints are closed under the dual discriminator polymorphism. We believe that generalizing the dichotomy results of solvability of the IMP_d for a finite domain is an interesting and challenging goal that we leave as an open problem.

References

1 Libor Barto, Andrei Krokhin, and Ross Willard. Polymorphisms, and How to Use Them. In Andrei Krokhin and Stanislav Zivny, editors, *The Constraint Satisfaction Problem: Complexity and Approximability*, volume 7 of *Dagstuhl Follow-Ups*, pages 1–44. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, Dagstuhl, Germany, 2017. doi:10.4230/DFU.Vol7.15301.1.

2 Paul Beame, Russell Impagliazzo, Jan Krajícek, Toniann Pitassi, and Pavel Pudlák. Lower bound on Hilbert's Nullstellensatz and propositional proofs. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, pages 794–806, 1994.

3 Arpitha P. Bharathi and Monaldo Mastrolilli. Ideal Membership Problem and a Majority Polymorphism over the Ternary Domain. In Javier Esparza and Daniel Kráľ, editors, *45th International Symposium on Mathematical Foundations of Computer Science (MFCS 2020)*, volume 170 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 13:1–13:13, Dagstuhl, Germany, 2020. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPIcs.MFCS.2020.13.

4 Arpitha P. Bharathi and Monaldo Mastrolilli. Ideal membership problem for boolean minority, 2020. arXiv:2006.16422.

5 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. *Journal of Symbolic Computation*, 41(3):475–511, 2006. Logic, Mathematics and Computer Science: Interactions in honor of Bruno Buchberger (60th birthday). doi:10.1016/j.jsc.2005.09.007.

6 Andrei A. Bulatov. A dichotomy theorem for nonuniform CSPs (best paper award). In *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 319–330, 2017.

7 Andrei A. Bulatov. Constraint satisfaction problems: Complexity and algorithms. *ACM SIGLOG News*, 5(4):4–24, November 2018. doi:10.1145/3292048.3292050.

8 Andrei A. Bulatov and Akbar Rafiey. On the complexity of csp-based ideal membership problems, 2020. arXiv:2011.03700.

9 Samuel R. Buss and Toniann Pitassi. Good degree bounds on Nullstellensatz refutations of the induction principle. *J. Comput. Syst. Sci.*, 57(2):162–171, 1998.

10 Hubie Chen. A rendezvous of logic, complexity, and algebra. *ACM Comput. Surv.*, 42(1):2:1–2:32, December 2009. `doi:10.1145/1592451.1592453`.

11 Martin C. Cooper, David A. Cohen, and Peter G. Jeavons. Characterising tractable constraints. *Artificial Intelligence*, 65(2):347–361, 1994. `doi:10.1016/0004-3702(94)90021-3`.

12 David A. Cox, John Little, and Donal O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra*. Springer Publishing Company, Incorporated, 4th edition, 2015.

13 Jean-Charles Faugère, Patrizia M. Gianni, Daniel Lazard, and Teo Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *Journal of Symbolic Computation*, 16(4):329–344, 1993. `doi:10.1006/jsco.1993.1051`.

14 Michel X Goemans and David P Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM (JACM)*, 42(6):1115–1145, 1995.

15 Dima Grigoriev. Tseitin's tautologies and lower bounds for Nullstellensatz proofs. In *39th Annual Symposium on Foundations of Computer Science, FOCS '98, November 8-11, 1998, Palo Alto, California, USA*, pages 648–652, 1998.

16 David Hilbert. Ueber die theorie der algebraischen formen. *Mathematische Annalen*, 36:473–534, 1890. `doi:10.1007/BF01208503`.

17 David Hilbert. Ueber die vollen invariantensysteme. *Mathematische Annalen*, 42:313–373, 1893. URL: `http://eudml.org/doc/157652`.

18 Peter Jeavons. On the algebraic structure of combinatorial problems. *Theoretical Computer Science*, 200(1):185–204, 1998. `doi:10.1016/S0304-3975(97)00230-2`.

19 Peter Jeavons, David Cohen, and Marc Gyssens. Closure properties of constraints. *J. ACM*, 44(4):527–548, 1997. `doi:10.1145/263867.263489`.

20 Monique Laurent. *Sums of Squares, Moment Matrices and Optimization Over Polynomials*, pages 157–270. Springer, New York, 2009. `doi:10.1007/978-0-387-09686-5_7`.

21 Alan K. Mackworth. Consistency in networks of relations. *Artificial Intelligence*, 8(1):99–118, 1977. `doi:10.1016/0004-3702(77)90007-8`.

22 Monaldo Mastrolilli. The complexity of the ideal membership problem and theta bodies for constrained problems over the boolean domain. *CoRR, to appear in ACM Transactions on Algorithms*, abs/1904.04072, 2019. `arXiv:1904.04072`.

23 Monaldo Mastrolilli. The complexity of the ideal membership problem for constrained problems over the boolean domain. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '19, pages 456–475, Philadelphia, PA, USA, 2019. Society for Industrial and Applied Mathematics. URL: `http://dl.acm.org/citation.cfm?id=3310435.3310464`.

24 Ernst W. Mayr. Membership in polynomial ideals over q is exponential space complete. In B. Monien and R. Cori, editors, *STACS 89*, pages 400–406, Berlin, Heidelberg, 1989. Springer Berlin Heidelberg.

25 Ernst W. Mayr and Albert R. Meyer. The complexity of the word problems for commutative semigroups and polynomial ideals. *Advances in Mathematics*, 46(3):305–329, 1982. `doi:10.1016/0001-8708(82)90048-2`.

26 Ryan O'Donnell. SOS Is Not Obviously Automatizable, Even Approximately. In Christos H. Papadimitriou, editor, *8th Innovations in Theoretical Computer Science Conference (ITCS 2017)*, volume 67 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 59:1–59:10, Dagstuhl, Germany, 2017. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik. `doi:10.4230/LIPIcs.ITCS.2017.59`.

27 Prasad Raghavendra and Benjamin Weitz. On the Bit Complexity of Sum-of-Squares Proofs. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *44th International Colloquium on Automata, Languages, and Programming (ICALP 2017)*, volume 80 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 80:1–80:13, Dagstuhl, Germany, 2017. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik. `doi:10.4230/LIPIcs.ICALP.2017.80`.

28 Prasad Raghavendra and Benjamin Weitz. On the bit complexity of sum-of-squares proofs. In *44th International Colloquium on Automata, Languages, and Programming, ICALP, Poland*, pages 80:1–80:13, 2017.

29 Thomas J. Schaefer. The complexity of satisfiability problems. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC '78, pages 216–226, New York, NY, USA, 1978. ACM. doi:10.1145/800133.804350.

30 Amir Shpilka. Recent results on polynomial identity testing. In Alexander Kulikov and Nikolay Vereshchagin, editors, *Computer Science – Theory and Applications*, pages 397–400, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

31 Ágnes Szendrei. *Clones in universal algebra.* Les Presses de l'Université de Montréal, 1986.

32 Marc R.C. van Dongen. *Constraints, Varieties, and Algorithms.* PhD thesis, Department of Computer Science, University College, Cork, Ireland, 2002. URL: http://csweb.ucc.ie/~dongen/papers/UCC/02/thesis.pdf.

33 Benjamin Weitz. *Polynomial Proof Systems, Effective Derivations, and their Applications in the Sum-of-Squares Hierarchy.* PhD thesis, EECS Department, University of California, Berkeley, May 2017. URL: http://www2.eecs.berkeley.edu/Pubs/TechRpts/2017/EECS-2017-38.html.

34 Dmitriy Zhuk. A proof of the CSP dichotomy conjecture. *J. ACM*, 67(5):30:1–30:78, 2020. doi:10.1145/3402029.

A Ideals, Varieties and Constraints

Let $\mathbb{F}$ denote an arbitrary field (for the applications of this paper $\mathbb{F} = \mathbb{R}$). Let $\mathbb{F}[x_1, \dots, x_n]$ be the ring of polynomials over a field $\mathbb{F}$ and indeterminates $x_1, \dots, x_n$. Let $\mathbb{F}[x_1, \dots, x_n]_d$ denote the subspace of polynomials of degree at most d.

▶ **Definition 20.** *The ideal (of $\mathbb{F}[x_1, \dots, x_n]$) generated by a finite set of polynomials $\{f_1, \dots, f_m\}$ in $\mathbb{F}[x_1, \dots, x_n]$ is defined as*

$$\boldsymbol{I}(f_1, \dots, f_m) \overset{\text{def}}{=} \left\{ \sum_{i=1}^{m} t_i f_i \mid t_1, \dots, t_m \in \mathbb{F}[x_1, \dots, x_n] \right\}.$$

The set of polynomials that vanish in a given set $S \subset \mathbb{F}^n$ is called the **vanishing ideal** *of S and denoted: $\boldsymbol{I}(S) \overset{\text{def}}{=} \{f \in \mathbb{F}[x_1, \dots, x_n] : f(a_1, \dots, a_n) = 0 \ \forall (a_1, \dots, a_n) \in S\}$.*

▶ **Definition 21.** *An ideal $\mathtt{I}$ is* **radical** *if $f^m \in \mathtt{I}$ for some integer $m \geq 1$ implies that $f \in \mathtt{I}$.*

Another common way to denote $\mathbf{I}(f_1, \dots, f_m)$ is by $\langle f_1, \dots, f_m \rangle$ and we will use both notations interchangeably.

▶ **Definition 22.** *Let $\{f_1, \dots, f_m\}$ be a finite set of polynomials in $\mathbb{F}[x_1, \dots, x_n]$. We call $\boldsymbol{V}(f_1, \dots, f_m) \overset{\text{def}}{=} \{(a_1, \dots, a_n) \in \mathbb{F}^n | \ f_i(a_1, \ \dots, a_n) = 0 \quad 1 \leq i \leq m\}$ the* **affine variety** *defined by $f_1, \dots, f_m$.*

▶ **Definition 23.** *Let $\mathtt{I} \subseteq \mathbb{F}[x_1, \dots, x_n]$ be an ideal. We will denote by $\boldsymbol{V}(\mathtt{I})$ the set $\boldsymbol{V}(\mathtt{I}) = \{(a_1, \dots, a_n) \in \mathbb{F}^n | f(a_1, \dots, a_n) = 0 \quad \forall f \in \mathtt{I}\}$.*

▶ **Theorem 24** ([12], Th.15, p.196)**.** *If I and J are ideals in $\mathbb{F}[x_1, \dots, x_n]$, then $\boldsymbol{V}(I \cap J) = \boldsymbol{V}(I) \cup \boldsymbol{V}(J)$.*

A.1 The Ideal-CSP Correspondence

Indeed, let $\mathcal{C} = (X, D, C)$ be an instance of the CSP(Γ) (see Definition 5). Without loss of generality, we shall assume that $D \subset \mathbb{N}$ and $D \subseteq \mathbb{F}$.

Let $Sol(\mathcal{C})$ be the (possibly empty) set of all feasible solutions of $\mathcal{C}$. In the following, we map $Sol(\mathcal{C})$ to an ideal $I_{\mathcal{C}} \subseteq \mathbb{F}[X]$ such that $Sol(\mathcal{C}) = \mathbf{V}(I_{\mathcal{C}})$.

Let $Y = (x_{i_1}, \ldots, x_{i_k})$ be a k-tuple of variables from X and let $R(Y)$ be a non empty constraint from C. In the following, we map $R(Y)$ to a generating system of an ideal such that the projection of the variety of this ideal onto Y is equal to $R(Y)$ (see [32] for more details).

Every $v = (v_1, \ldots, v_k) \in R(Y)$ corresponds to some point $v \in \mathbb{F}^k$. It is easy to check [12] that $\mathbf{I}(\{v\}) = \langle x_{i_1} - v_1, \ldots, x_{i_k} - v_k\rangle$, where $\langle x_{i_1} - v_1, \ldots, x_{i_k} - v_k\rangle \subseteq \mathbb{F}[Y]$ is radical. By Theorem 24, we have

$$R(Y) = \bigcup_{v \in R(Y)} \mathbf{V}(\mathbf{I}(\{v\})) = \mathbf{V}\left(I_{R(Y)}\right), \text{ where } I_{R(Y)} = \bigcap_{v \in R(Y)} \mathbf{I}(\{v\}), \tag{7}$$

where $I_{R(Y)} \subseteq \mathbb{F}[Y]$ is zero-dimensional and radical ideal since it is the intersection of radical ideals (see [12], Proposition 16, p.197). Equation (7) states that constraint $R(Y)$ is a variety of $\mathbb{F}^k$. It is easy to find a generating system for $I_{R(Y)}$:

$$I_{R(Y)} = \langle \prod_{v \in R} (1 - \prod_{j=1}^{k} \delta_{v_j}(x_{i_j})), \prod_{j \in D}(x_{i_1} - j), \ldots, \prod_{j \in D}(x_{i_k} - j)\rangle, \tag{8}$$

where $\delta_{v_j}(x_{i_j})$ are indicator polynomials, i.e. equal to one when $x_{i_j} = v_j$ and zero when $x_{i_j} \in D \setminus \{v_j\}$; polynomials $\prod_{j \in D}(x_{i_k} - j)$ force variables to take values in D and will be denoted as *domain polynomials*.

The smallest ideal (with respect to inclusion) of $\mathbb{F}[X]$ containing $I_{R(Y)} \subseteq \mathbb{F}[\mathbf{x}]$ will be denoted $I_{R(Y)}^{\mathbb{F}[X]}$ and it is called the $\mathbb{F}[X]$-module of I. The set $Sol(\mathcal{C}) \subset \mathbb{F}^n$ of solutions of $\mathcal{C} = (X, D, C)$ is the intersection of the varieties of the constraints:

$$Sol(\mathcal{C}) = \bigcap_{R(Y) \in C} \mathbf{V}\left(I_{R(Y)}^{\mathbb{F}[X]}\right) = \mathbf{V}(I_{\mathcal{C}}), \text{ where } I_{\mathcal{C}} = \sum_{R(Y) \in C} I_{R(Y)}^{\mathbb{F}[X]}. \tag{9}$$

The following properties follow from Hilbert's Nullstellensatz.

▶ **Theorem 25.** *Let $\mathcal{C}$ be an instance of the CSP(Γ) and $\mathtt{I}_{\mathcal{C}}$ defined as in (9). Then*

$$\text{(Weak Nullstellensatz)} \quad \boldsymbol{V}(\mathtt{I}_{\mathcal{C}}) = \emptyset \Leftrightarrow 1 \in \boldsymbol{I}(\mathtt{I}_{\mathcal{C}}) \Leftrightarrow \mathtt{I}_{\mathcal{C}} = \mathbb{F}[X], \tag{10}$$

$$\text{(Strong Nullstellensatz)} \quad \boldsymbol{I}(\boldsymbol{V}(\mathtt{I}_{\mathcal{C}})) = \sqrt{\mathtt{I}_{\mathcal{C}}}, \tag{11}$$

$$\text{(Radical Ideal)} \quad \sqrt{\mathtt{I}_{\mathcal{C}}} = \mathtt{I}_{\mathcal{C}}. \tag{12}$$

Theorem 25 follows from a simple application of the celebrated and basic result in algebraic geometry known as Hilbert's Nullstellensatz. In the general version of Nullstellensatz it is necessary to work in an algebraically closed field and take a radical of the ideal of polynomials. In our special case it is not needed due to the presence of domain polynomials. Indeed, the latter implies that we know a priori that the solutions must be in $\mathbb{F}$ (note that we are assuming $D \subseteq \mathbb{F}$).

A.2 Gröbner bases

In this section we suppose a fixed monomial ordering $>$ on $\mathbb{F}[x_1, \dots, x_n]$ (see [12], Definition 1, p.55), which will not be defined explicitly. We can reconstruct the monomial $x^\alpha = x_1^{\alpha_1} \cdots x_n^{\alpha_n}$ from the n-tuple of exponents $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$. This establishes a one-to-one correspondence between the monomials in $\mathbb{F}[x_1, \dots, x_n]$ and $\mathbb{Z}^n_{\geq 0}$. Any ordering $>$ we establish on the space $\mathbb{Z}^n_{\geq 0}$ will give us an ordering on monomials: if $\alpha > \beta$ according to this ordering, we will also say that $x^\alpha > x^\beta$. The two monomial orderings that we use in this paper are the lexicographic order $>_{\mathsf{lex}}$ and the graded lexicographic ordering $>_{\mathsf{grlex}}$.

▶ **Definition 26.** *Let $\alpha = (\alpha_1, \dots, \alpha_n), \beta = (\beta_1, \dots, \beta_n) \in \mathbb{Z}^n_{\geq 0}$ and $|\alpha| = \sum_{i=1}^n \alpha_i$, $|\beta| = \sum_{i=1}^n \beta_i$.*

(i) *We say $\alpha >_{lex} \beta$ if, in the vector difference $\alpha - \beta \in \mathbb{Z}^n$, the left most nonzero entry is positive. We will write $x^\alpha >_{lex} x^\beta$ if $\alpha >_{lex} \beta$.*

(ii) *We say $\alpha >_{grlex} \beta$ if $|\alpha| > |\beta|$, or $|\alpha| = |\beta|$ and $\alpha >_{lex} \beta$.*

▶ **Definition 27.** *For any $\alpha = (\alpha_1, \cdots, \alpha_n) \in \mathbb{Z}^n_{\geq 0}$ let $x^\alpha \stackrel{\text{def}}{=} \prod_{i=1}^n x_i^{\alpha_i}$. Let $f = \sum_\alpha a_\alpha x^\alpha$ be a nonzero polynomial in $\mathbb{F}[x_1, \dots, x_n]$ and let $>$ be a monomial order.*

(i) *The* **multidegree** *of f is $\operatorname{multideg}(f) \stackrel{\text{def}}{=} \max(\alpha \in \mathbb{Z}^n_{\geq 0} : a_\alpha \neq 0)$.*

(ii) *The* **degree** *of f is $deg(f) = |\operatorname{multideg}(f)|$. In this paper, this is always according to grlex order.*

(iii) *The* **leading coefficient** *of f is $\mathrm{LC}(f) \stackrel{\text{def}}{=} a_{\operatorname{multideg}(f)} \in \mathbb{F}$.*

(iv) *The* **leading monomial** *of f is $\mathrm{LM}(f) \stackrel{\text{def}}{=} x^{\operatorname{multideg}(f)}$ (with coefficient 1).*

(v) *The* **leading term** *of f is $\mathrm{LT}(f) \stackrel{\text{def}}{=} \mathrm{LC}(f) \cdot \mathrm{LM}(f)$.*

The concept of *reduction*, also called *multivariate division* or *normal form computation*, is central to Gröbner basis theory. It is a multivariate generalization of the Euclidean division of univariate polynomials.

▶ **Definition 28.** *Fix a monomial order and let $G = \{g_1, \dots, g_t\} \subset \mathbb{F}[x_1, \dots, x_n]$. Given $f \in \mathbb{F}[x_1, \dots, x_n]$, we say that* ***$f$* reduces to *$r$* modulo *$G$***, *written $f \to_G r$, if f can be written in the form $f = A_1 g_1 + \cdots + A_t g_t + r$ for some $A_1, \dots, A_t, r \in \mathbb{F}[x_1, \dots, x_n]$, such that:*

(i) *No term of r is divisible by any of $\mathrm{LT}(g_1), \dots, \mathrm{LT}(g_t)$.*

(ii) *Whenever $A_i g_i \neq 0$, we have $\operatorname{multideg}(f) \geq \operatorname{multideg}(A_i g_i)$.*

The polynomial remainder r is called a **normal form of *f* by *G*** *and will be denoted by $f|_G$.*

A normal form of f by G, i.e. $f|_G$, can be obtained by repeatedly performing the following until it cannot be further applied: choose any $g \in G$ such that $\mathrm{LT}(g)$ divides some term t of f and replace f with $f - \frac{t}{\mathrm{LT}(g)} g$. Note that the order we choose the polynomials g in the division process is not specified.

In general a normal form $f|_G$ is not uniquely defined. Even when f belongs to the ideal generated by G, i.e. $f \in \mathbf{I}(G)$, it is not always true that $f|_G = 0$.

▶ **Example 29.** Let $f = xy^2 - y^3$ and $G = \{g_1, g_2\}$, where $g_1 = xy - 1$ and $g_2 = y^2 - 1$. Consider the graded lexicographic order (with $x > y$) and note that $f = y \cdot g_1 - y \cdot g_2 + 0$ and $f = 0 \cdot g_1 + (x - y) \cdot g_2 + x - y$.

This non-uniqueness is the starting point of Gröbner basis theory.

▶ **Definition 30.** *Fix a monomial order on the polynomial ring $\mathbb{F}[x_1, \ldots, x_n]$. A finite subset $G = \{g_1, \ldots, g_t\}$ of an ideal $\mathtt{I} \subseteq \mathbb{F}[x_1, \ldots, x_n]$ different from $\{0\}$ is said to be a* **Gröbner basis** *(or* **standard basis***) if $\langle \mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_t)\rangle = \langle \mathrm{LT}(\mathtt{I})\rangle$, where we denote by $\langle \mathrm{LT}(\mathtt{I})\rangle$ the ideal generated by the elements of the set $\mathrm{LT}(\mathtt{I})$ of leading terms of nonzero elements of $\mathtt{I}$.*

▶ **Definition 31.** *A* **reduced Gröbner basis** *for a polynomial ideal $\mathtt{I}$ is a Gröbner basis G for $\mathtt{I}$ such that:*

(i) $\mathrm{LC}(g) = 1$ *for all $g \in G$.*

(ii) *For all $g \in G$, g cannot reduce any other polynomial from G, i.e $f|_g = f$ for every $f \in G \setminus \{g\}$.*

It is known (see [12], Theorem 5, p.93) that for a given monomial ordering, a polynomial ideal $I \neq \{0\}$ has a reduced Gröbner basis (see Definition 31), and the reduced Gröbner basis is unique.

▶ **Proposition 32** ([12], Proposition 1, p.83)**.** *Let $\mathtt{I} \subset \mathbb{F}[x_1, \ldots, x_n]$ be an ideal and let $G = \{g_1, \ldots, g_t\}$ be a Gröbner basis for $\mathtt{I}$. Then given $f \in \mathbb{F}[x_1, \ldots, x_n]$, f can be written in the form $f = A_1 g_1 + \cdots + A_t g_t + r$ for some $A_1, \ldots, A_t, r \in \mathbb{F}[x_1, \ldots, x_n]$, such that:*

(i) *No term of r is divisible by any of $\mathrm{LT}(g_1), \ldots, \mathrm{LT}(g_t)$.*

(ii) *Whenever $A_i g_i \neq 0$, we have $\mathrm{multideg}(f) \geq \mathrm{multideg}(A_i g_i)$.*

(iii) *There is a unique $r \in \mathbb{F}[x_1, \ldots, x_n]$.*

In particular, r is the remainder on division of f by G no matter how the elements of G are listed when using the division algorithm.

▶ **Corollary 33** ([12], Corollary 2, p.84)**.** *Let $G = \{g_1, \ldots, g_t\}$ be a Gröbner basis for $\mathtt{I} \subseteq \mathbb{F}[x_1, \ldots, x_n]$ and let $f \in \mathbb{F}[x_1, \ldots, x_n]$. Then $f \in \mathtt{I}$ if and only if the remainder on division of f by G is zero.*

▶ **Definition 34.** *We will write $f|_F$ for the remainder of f by the ordered s-tuple $F = (f_1, \ldots, f_s)$. If F is a Gröbner basis for $\langle f_1, \ldots, f_s\rangle$, then we can regard F as a set (without any particular order) by Proposition 32.*

The "obstruction" to $\{g_1, \ldots, g_t\}$ being a Gröbner basis is the possible occurrence of polynomial combinations of the g_i whose leading terms are not in the ideal generated by the $\mathrm{LT}(g_i)$. One way (actually the only way) this can occur is if the leading terms in a suitable combination cancel, leaving only smaller terms. The latter is fully captured by the so called S-polynomials that play a fundamental role in Gröbner basis theory.

▶ **Definition 35.** *Let $f, g \in \mathbb{F}[x_1, \ldots, x_n]$ be nonzero polynomials. If $\mathrm{multideg}(f) = \alpha$ and $\mathrm{multideg}(g) = \beta$, then let $\gamma = (\gamma_1, \ldots, \gamma_n)$, where $\gamma_i = \max(\alpha_i, \beta_i)$ for each i. We call x^γ the* **least common multiple** *of $\mathrm{LM}(f)$ and $\mathrm{LM}(g)$, written $x^\gamma = \mathrm{lcm}(\mathrm{LM}(f), \mathrm{LM}(g))$. The* ***S*-polynomial** *of f and g is the combination $S(f,g) = \frac{x^\gamma}{\mathrm{LT}(f)} \cdot f - \frac{x^\gamma}{\mathrm{LT}(g)} \cdot g$.*

The use of S-polynomials to eliminate leading terms of multivariate polynomials generalizes the row reduction algorithm for systems of linear equations. If we take a system of homogeneous linear equations (i.e.: the constant coefficient equals zero), then it is not hard to see that bringing the system in triangular form yields a Gröbner basis for the system.

▶ **Theorem 36** (**Buchberger's Criterion**)**.** *(See e.g. [12], Theorem 3, p.105) A basis $G = \{g_1, \ldots, g_t\}$ for an ideal $\mathtt{I}$ is a Gröbner basis if and only if $S(g_i, g_j) \to_G 0$ for all $i \neq j$.*

By Theorem 36 it is easy to show whether a given basis is a Gröbner basis. Indeed, if G is a Gröbner basis then given $f \in \mathbb{F}[x_1, \ldots, x_n]$, $f|_G$ is unique and it is the remainder on division of f by G, no matter how the elements of G are listed when using the division algorithm.

Algorithm 3 Buchberger's Algorithm.

```
 1: Input: A finite set $F = \{f_1, \ldots, f_s\}$ of polynomials
 2: Output: A finite Gröbner basis $G$ for $\langle f_1, \ldots, f_s \rangle$
 3: $G := F$
 4: $C := G \times G$
 5: while $C \neq \emptyset$ do
 6:     Choose a pair $(f, g) \in C$
 7:     $C := C \setminus \{(f, g)\}$
 8:     $h := S(f, g)|_G$
 9:     if $h \neq 0$ then
10:         $C := C \cup (G \times \{h\})$
11:         $G := G \cup \{h\}$
12:     end if
13: end while
14: Return G
```

Furthermore, Theorem 36 leads naturally to an algorithm for computing Gröbner bases for a given ideal $I = \langle f_1, \ldots, f_s \rangle$: start with a basis $G = \{f_1, \ldots, f_s\}$ and for any pair $f, g \in G$ with $S(f, g)|_G \neq 0$ add $S(f, g)|_G$ to G. This is known as Buchberger's algorithm [5] (for more details see Algorithm 3 in Section A.2.1).

Note that Algorithm 3 is non-deterministic and the resulting Gröbner basis in not uniquely determined by the input. This is because the normal form $S(f, g)|_G$ (see Algorithm 3, line 8) is not unique as already remarked. We observe that one simple way to obtain a deterministic algorithm (see [12], Theorem 2, p. 91) is to replace $h := S(f, g)|_G$ in line 8 with $h := S(f, g)|_G$ (see Definition 34), where in the latter G is an ordered tuple. However, this is potentially dangerous and inefficient. Indeed, there are simple cases where the combinatorial growth of set G in Algorithm 3 is out of control very soon.

A.2.1 Construction of Gröbner Bases

Buchberger's algorithm [5] can be formulated as in Algorithm 3. The pairs that get placed in the set C are often referred to as *critical pairs*. Every newly added reduced S-polynomial enlarges the set C. If we use $h := S(f, g)|_G$ in line 8 then there are simple cases where the situation is out of control. This combinatorial growth can be controlled to some extent be eliminating unnecessary critical pairs.