Degröbnerization and Its Applications: Reverse Engineering of Gene Regulatory Networks

Michela Ceria[1,*,†], Samuel Lundqvist[2,†] and Ferdinando Mora[3,†]

[1]*Dipartimento di Meccanica, Matematica e Management, Politecnico di Bari, Via Orabona 4, I-70125 Bari, Italy*
[2]*Department of Mathematics, Stockholm University, SE-106 91 Stockholm, Sweden*
[3]*Dipartimento di Matematica, Università di Genova, Via Dodecaneso 35, 16146, Genova, Italy*

Abstract

In this short note, we consider an application of ideals of points in biology and algebraic statistics proposed by Laubenbacher and Stigler. We do this from the *Degröbnerization* perspective, avoiding unnecessary computations of Gröbner bases, and instead, when possible, leaning on linear algebra and combinatorial methods.

Keywords

Ideals of points, normal forms, vector space bases, Degröbnerization

1. Introduction

Following the convention of [13], let $\mathcal{P} := \mathbf{k}[x_1, ..., x_n]$ be the polynomial ring over a field $\mathbf{k}$, and let I be a zero-dimensional ideal in $\mathcal{P}$. Then $\mathcal{P}/I$ is finite as a vector space over $\mathbf{k}$. Let A denote a set of polynomials such that its image under the map $\mathcal{P} \to \mathcal{P}/I$ is a vector space basis for $\mathcal{P}/I$.

Having a basis for $\mathcal{P}/I$ means that we can define a normal form for elements in $\mathcal{P}$ as $\mathrm{Nf}(p, A) = \sum_{t\in A} c_t t$, where the coefficients $\{c_t\}$ are uniquely determined by requiring that $p - \sum_{t\in A} c_t t$ is zero under the map $\mathcal{P} \to \mathcal{P}/I$.

The most common way of choosing a vector space basis for $\mathcal{P}/I$ is as the image of the *Gröbner escalier* with respect to a fixed term order under the map $\mathcal{P} \to \mathcal{P}/I$. Recall that the Gröbner escalier, sometimes referred to as the Gröbner *staircase*, or the *standard monomials*, is the set of monomials outside the leading monomial ideal with respect to the term order.

We denote the Gröbner escalier of I by $N(I)$. The *border* of $N(I)$ is the set of terms $\{x_i t : t \in N(I)\} \setminus N(I)$. With this notation, the set $\{t - \mathrm{Nf}(t, N(I)) : t$ on the border of $N(I)\}$ becomes a Gröbner basis for I.

Given a finite set of points $\mathbf{X} = \{P_1, ..., P_N\} \subset \mathbf{k}^n$, let us denote by $P_i = (a_{1i}, ..., a_{ni})$, $1 \leq i \leq N$, the coordinates of the points and by $I(\mathbf{X})$ the zero dimensional ideal of all polynomials vanishing on $\mathbf{X}$; *the vanishing ideal* with respect to $\mathbf{X}$.

SC^2 *2021: 6th International Workshop on Satisfiability Checking and Symbolic Computation, August 19–20, 2021*
*Corresponding author.
†These authors contributed equally.
michela.ceria@gmail.com (M. Ceria); samuel@math.su.se (S. Lundqvist); 5919@unige.it (F. Mora)

One important feature of a Gröbner basis is that it provides a way to compute normal forms via Buchberger's reduction. We stress, however, that for vanishing ideals of points, the tools provided by this reduction process should be considered non-optimal, and we will in Proposition 1 recall an alternative way of efficiently computing the coefficients of the normal form $\sum_{t\in A} c_t t$, which goes in line with the idea of Degröbnerization.

There are several algorithms available for computing a vector space basis for $\mathcal{P}/I$. For our purposes the most important ones are the Buchberger-Möller algorithm and the Cerlienco-Mureddu Correspondence. We point the reader to the paper [1] for a discussion of other methods, in particular the Möller algorithm. We remark also that the treatment of the Buchberger-Möller algorithm and the Cerlienco-Mureddu Correspondence is done in a more careful way in [1]. sation

The Buchberger-Möller algorithm [12] is a method based on linear algebra that determines $N(I(\mathbf{X}))$, the border, and the Gröbner basis (by computing the normal form of the border elements).

The Cerlienco and Mureddu algorithm [3, 4, 5] computes $N(\mathbf{X})$ combinatorially by defining a bijection (the Cerlienco-Mureddu Correspondence) $\Phi : \mathbf{X} \to N(I(\mathbf{X}))$ between $\mathbf{X}$ and $N(I(\mathbf{X}))$, with respect to the *lexicographical term ordering*.

The main ways of implementing the Cerlienco-Mureddu Correspondence is by means of the *Lex Game* [6], and the *Iterative Lex Game* [1, 2]. The number of comparisons needed in the Lex Game, which iterates over the coordinates, is bounded by $nN + N^2$, see [10]. However, for many purposes it is important to be able to iterate over the points, and the Iterative Lex Game does this with complexity $O(N^2 n \log(N))$.

Le us finally recall the following proposition.

Proposition 1 ([10]). *Let $\mathbf{X} = \{P_1, ..., P_N\}$ be a finite set of points, let $I = I(\mathbf{X}) \subseteq \mathbf{k}[x_1, ..., x_n]$ be the vanishing ideal with respect to $\mathbf{X}$, and let $A = \{t_1, ..., t_N\} \subset \mathbf{k}[x_1, ..., x_n]$ be such that its image under the map $\mathcal{P} \to \mathcal{P}/I$ is a vector space basis for $\mathcal{P}/I$.*

Then, for each $f \in \mathbf{k}[x_1, ..., x_n]$, we have

$$Nf(f, A) = (t_1, ..., t_N)(A[\mathbf{X}]^{-1})^t (f(P_1), ..., f(P_N))^t, \tag{1}$$

where $A[\mathbf{X}]$ is the matrix whose rows are the evaluations of A at the elements of $\mathbf{X}$, and where $Nf(f, A)$ is the normal form of f with respect to $I(\mathbf{X})$ and the basis.

The above Proposition 1 shows that, as soon as we have $\mathbf{X}$ and a basis of the quotient algebra $\mathcal{P}/I(\mathbf{X})$, computing the normal form of any polynomial modulo $I(\mathbf{X})$ is only a matter of *evaluations* and *linear algebra*.

2. Gröbner technologies for gene regulatory networks

We now briefly describe the model for gene regulatory networks introduced by Laubenbacher and Stigler [8]. For a more detailed treatment, we refer to [1].

Let p be a prime and let $\mathbf{X} = \{P_1, ..., P_N\} \subseteq (\mathbb{Z}_p)^n$ be a set of points, and consider the map $F : (\mathbb{Z}_p)^n, \to (\mathbb{Z}_p)^n, F(P_i) = (f_1(P_i), \ldots, f_n(P_i))$, where

$$f_j : (\mathbb{Z}_p)^n \to \mathbb{Z}_p, 1 \leq j \leq n : (f_1(P_i), \ldots, f_n(P_i)) = P_{i+1}, i = 1, \ldots, N-1. \tag{2}$$

These functions represent the dynamical system in question, where the variables represent the genes, and the points represent the states the genes can be in.

What is done in [8] is to separately compute the functions f_j, $1 \leq j \leq n$, as *polynomials in normal form*, namely in reduced form modulo the ideal $I(\mathbf{X})$.

In a further paper [9] Lauenbacher and Stigler apply their tool to the *design of experiments* [14] remarking that their algorithm can be applied simply adapting Eq.(2) after redefining F as

$$F(P_i) = (f_1(P_i), \ldots, f_n(P_i)) = Q_i, i = 1, \ldots, N$$

for specific given points $Q_i \in \mathbf{k}^n, i = 1, \ldots, N$.

3. The Gröbnerian approach versus the Degröbnerization approach

To solve both problems, Laubenbacher and Stigler propose to first determine a particular solution for each f_j, then compute a Gröbner basis $\mathcal{G}$ of $I(\mathbf{X})$, via the Buchberger-Möller algorithm, and finally determine the normal form of each particular solution using Buchberger's reduction.

The complexity of this procedure is reported, see [8], to be quadratic in the number of variables and exponential in the number N of time points, but in fact, using a trick in the reduction process, see [1], the algorithm of [8] can be improved to work within the complexity $O(n^2N^3((\log(p))^2))$.

In the Degröbnerization approach, the evaluation $F(P_i)$ of a particular solution is directly read from $\mathbf{X}$, a *linear basis* for $\mathbf{k}[x_1, ..., x_n]/I$ is computed by the Lex Game, or the iterative Lex Game algorithm, and Proposition 1 is used for the normal form computation.

Even if we use the slower iterative Lex Game algorithm for computing the linear basis, we obtain an improvement by a factor n; the complexity using the iterative Lex Game algorithm is $O(nN^3(\log(p))^2)$ [1]. As we usually have $n > N$ in these biological applications [7], this improvement is substantial.

Let us remark that both the algorithm by Laubenbacher and Stigler [8] and our Degröbnerization approach share the same input and output, but uses alternative and in some sense dual approaches to describe and process the data. The major practical advantage of the latter approach is the shortcut achieved by skipping the computation of a Gröbner basis. But the two approaches represent a real change of perspective. In the Gröbner framework, data are represented via the *polynomial ideal*. On the other hand, Degröbnerization represents data via the *quotient algebra*, with its multiplicative structure.

We have discussed the problem in the original setting where the coefficients are considered in the finite field $\mathbb{Z}_p$; however both algoritms hold (and in principle could have important applications [14]) in the general case of an arbitrary field. We refer the reader to [1] for a careful analysis of this approach.

References

[1] Michela Ceria, Samuel Lundqvist, Teo Mora, *Degröbnerization: A political manifesto*. To appear in Applicable Algebra in Engineering, Communication and Computing (2022).

[2] Michela Ceria, Teo Mora, *Combinatorics of ideals of points: a Cerlienco-Mureddu-like approach for an iterative lex game*, available in arxiv as arXiv:1805.09165 [math.AC].

[3] Luigi Cerlienco , Marina Mureddu, *Algoritmi combinatori per l'interpolazione polinomiale in dimensione* ≥ 2, Publ. I.R.M.A. Strasbourg, 461/S-24 Actes 24^e Séminaire Lotharingien (1993), 39–76.

[4] Luigi Cerlienco, Marina Mureddu, *From algebraic sets to monomial linear bases by means of combinatorial algorithms*, Discrete Math. 139 (1995), 73–87.

[5] Luigi Cerlienco, Marina Mureddu, *Multivariate Interpolation and Standard Bases for Macaulay Modules*, J. Algebra 251 (2002), 686–726.

[6] Bálint Felszeghy, Balázs Ráth, Lajos Rónyai, *The lex game and some applications*, Journal of Symbolic Computation 41, no 6 (2006), 663–681.

[7] Winfried Just, Brandilyn Stigler, *Computing Gröbner bases of ideals of few points in high dimensions*, Communications in Computer Algebra 40, no. 3 (2006), 65–96.

[8] Reinhard Laubenbacher, Brandilyn Stigler, *A computational algebra approach to the reverse engineering of gene regulatory networks*, Journal of Theoretical Biology, 229, no. 4 (2004), 523–537, Academic Press.

[9] Reinhard Laubenbacher, Brandilyn Stigler, *Design of experiments and biochemical network inference* in Algebraic and Geometric Methods in Statistics. Eds: Gibilisco, Riccomagno, Rogantin, Wynn, Cambridge University Press (2008).

[10] Samuel Lundqvist, *Vector space bases associated to vanishing ideals of points*, Journal of Pure and Applied Algebra 214 no. 4 (2010), 309–321.

[11] Samuel Lundqvist, *Complexity of comparing monomials and two improvements of the Buchberger-Möller algorithm*, MMISC 2008, Lecture Notes in Comput. Sci. 5393 (2008), 105–125.

[12] Hans Michael Möller, Bruno Buchberger, *The construction of multivariate polynomials with preassigned zeros*, in European Computer Algebra Conference (1982), 24–31, Springer.

[13] Teo Mora, *Solving Polynomial Equation Systems* 4 Vols., Cambridge University Press,
I DOI: $10.1017/CBO9780511542831$ (2003), II DOI: $10.1017/CBO9781107340954$ (2005),
III DOI: $10.1017/CBO9781139015998$ (2015), IV DOI:$10.1017/CBO9781316271902$ (2016).

[14] Giovanni Pistone, Eva Riccomagno, Maria Piera Rogantin, *Methods in algebraic statistics for the design of experiments*, in Optimal Design and Related Areas in Optimsation and Statistics (2009), 97–132, Springer.