

A Saturation Algorithm for Homogeneous Binomial Ideals

Deepanjan Kesh and Shashank K Mehta

Indian Institute of Technology Kanpur

Kanpur, India, Pin Code - 208016

deepkesh@cse.iitk.ac.in, skmehta@cse.iitk.ac.in

Let I be an ideal in the polynomial ring $k[\mathbf{x}]$ over a field k . Saturation of I by the product $x_1 \cdots x_n$, denoted by $I : (x_1 \cdots x_n)^\infty$ is the ideal $\{f : x_1^{a_1} \cdots x_n^{a_n} f \in I, a_i \geq 0, 1 \leq i \leq n\}$. Binomials in the ring are defined as polynomials with at most two terms [1]. Ideals with a binomial basis are called binomial ideals. Toric ideals are examples of homogeneous binomial ideals.

We describe a fast algorithm to compute the saturation, $I : (x_1 \cdots x_n)^\infty$, of a homogeneous binomial ideal I . Here we would like to note that there are several algorithms to saturate pure difference binomial ideals [2], which are a special case of homogeneous binomial ideals.

Before proceeding, we will need some notations. U_i will denote the multiplicatively closed set $\{x_1^{a_1} \cdots x_{i-1}^{a_{i-1}} : a_j \geq 0, 1 \leq j < i\}$. \prec_i will denote a graded reverse lexicographic term order with x_i being the smallest. $\varphi_i : k[\mathbf{x}] \rightarrow k[\mathbf{x}][U_i^{-1}]$ is the natural localization map $r \mapsto r/1$.

Algorithm 1 describes the saturation algorithm due to Sturmfels [3] in the context of binomial ideals. Algorithm 2 describes the proposed algorithm. The primary motivation for the new approach is that the time complexity of Gröbner basis is a strong function of the number of variables. In the proposed algorithm, a Gröbner basis is computed in the i -th iteration in i variables. This requires the computation of a Gröbner basis over the ring $k[\mathbf{x}][U_i^{-1}]$. The Gröbner basis over such a ring is not known in the literature. Thus, we propose a variant of Gröbner bases, called pseudo Gröbner bases, and appropriately modify the Buchberger's algorithm to compute it.

Definition 1 A basis G of an homogeneous binomial ideal $I \subset k[\mathbf{x}][U_i^{-1}]$ is called a pseudo-Gröbner basis of I , if G can be partitioned into two sets G_1, G_2 , such that every binomial of I reduces to 0 (mod G_2) by G_1 , with respect to a given term-order.

Theorem 1 Let (G_1, G_2) be a pseudo Gröbner basis of a homogeneous binomial ideal I in $k[\mathbf{x}][U_i^{-1}]$ with respect to \prec_i . Then $(G_1 : x_i^\infty, G_2 : x_i^\infty)$ is a pseudo Gröbner basis of $I : x_i^\infty$.

Further details of the algorithm can be found in [4]. One thing to note is that the algorithm works only for binomial ideals, and it crucially uses the fact that the ring is localized with monomials. We have not been able to generalize the notions to general polynomial ideals or to polynomial ideals over function fields.

In the table given below, we present some preliminary experimental results of the application of the proposed algorithm in computing toric ideals. We compare our algorithm with the Sturmfels' algorithm [3] and Project and Lift [2], the best algorithm known to date to compute toric ideals. As expected, the table shows that our algorithm performs much better than the Sturmfels' algorithm, as our algorithm is specifically designed for binomial ideals.

Data: A homogeneous binomial ideal,
 $I \subset k[\mathbf{x}]$.
Result: $I : (x_1, \dots, x_n)^\infty$
1 for $i \leftarrow n$ **to** 1 **do**
2 $G \leftarrow$ Gröbner basis of I w.r.t. \prec_i ;
3 $I \leftarrow \langle \{f \div (x_1, \dots, x_n)^\infty \mid f \in G\} \rangle$;
4 end
5 return I ;
Algorithm 1: Sturmfels' Algorithm

Data: A homogeneous binomial ideal,
 $I \subset k[\mathbf{x}]$.
Result: $I : (x_1, \dots, x_n)^\infty$
1 for $i \leftarrow n$ **to** 1 **do**
2 $G \leftarrow$ Pseudo Gröbner basis of $\varphi_i(I)$
 w.r.t. \prec_i ;
3 $I \leftarrow \langle \{\varphi_i^{-1}(f \div (x_1, \dots, x_n)^\infty) \mid f \in G\} \rangle$
 ;
4 end
5 return I ;
Algorithm 2: Proposed Algorithm

To compare with the Project and Lift, we implemented it as reported on page 19 of [2], without optimizations reported in the subsequent pages. Similar optimizations are applicable in our algorithm and it too is implemented without the same in these experiments. The typical results are presented in the table given below. For a definitive comparison we intend to implement our algorithm with all possible optimizations and compare with 4ti2 [2], which is the optimal implementation of their algorithm.

Our intuition as to why our algorithm is doing better, in these experiments, compared to Project and Lift is that their algorithm uses Sturmfels' saturation algorithm as a subroutine, though the extent to which it uses the algorithm depends on the input ideal. On the other hand, our algorithm computes all saturations by the same approach.

Number of variables	Size of basis		Time taken (in sec.)		
	Initial	Final	Sturmfels'	Project and Lift	Proposed
8	4	186	0.30	0.12	0.10
	6	597	2.61	0.60	0.64
10	6	729	3.20	1.10	0.50
	8	357	2.40	0.40	0.29
12	6	423	1.70	0.90	0.27
	8	2695	305.00	60.00	27.20
14	10	1035	10.50	4.20	2.50

Table 1: Preliminary experimental results comparing Sturmfels', Project-and-Lift and our proposed algorithms

References

- [1] David Eisenbud and Bernd Sturmfels. Binomial ideals. *Duke Mathematical Journal*, 84:1–45, 1996.
- [2] Raymond Hemmecke and Peter N. Malkin. Computing generating sets of lattice ideals and Markov bases of lattices. *Journal of Symbolic Computation*, 44:1463–1476, 2009.
- [3] Bernd Sturmfels. Gröbner Bases and Convex Polytopes. University Lecture Series, *American Mathematical Society*, volume 8, 1995.
- [4] Deepanjan Kesh and Shashank K Mehta. A Saturation Algorithm for Homogeneous Binomial Ideals, *Manuscript*. <http://www.cse.iitk.ac.in/users/deepkesh/downloads/manuscript.pdf>, 2011.