Resolutions and Betti Numbers of Polynomial Modules via Involutive Bases

Mario Albert[1, *], **Matthias Fetzer**[1,**], and **Werner M. Seiler**[1,***]

[1] Institut für Mathematik, Universität Kassel, 34132 Kassel, Germany

We describe a novel approach to the computation of free resolutions and of Betti numbers of polynomial modules based on a combination of the theory of involutive bases with algebraic discrete Morse theory. This approach allows for the first time to compute Betti numbers (even single ones) without determining a whole resolution which in many cases drastically reduces the computation time.

1 Introduction

Computing (minimal) free resolutions of polynomial modules is a fundamental task in commutative algebra and algebraic geometry. Resolutions are e.g. used to determine derived functors like Ext or Tor. The Betti numbers which encode the size of the minimal resolution contain important geometric and topological information. Furthermore, many homological invariants like the Castelnuovo-Mumford regularity or the projective dimension are defined via the minimal resolution. They represent important complexity measures for modules.

Unfortunately, computing free resolutions is a rather expensive task. As a rough rule of thumb one may say that each module in the resolution requires one Gröbner basis computation (the usually applied algorithms do not really work this way, but the total computation time is similar). In principle, determining only the Betti numbers should be much cheaper, as one is interested only in the ranks of the modules appearing in the resolution and not in the explicit form of the differentials. But all existing computer algebra systems need about the same time for the two tasks. The reason is simply that internally a resolution is determined from which the Betti numbers are derived. In fact, even for computing the regularity or the projective dimension often a whole resolution is used.

In this short note, we briefly survey some recent results of the authors on a novel approach to compute resolutions and Betti numbers; for more details and the proofs we refer to [1, 2]. It is based on a combination of the theory of involutive bases and algebraic discrete Morse theory. This approach provides an explicit formula for the differential of a (generally non-minimal, but highly structured) resolution and thus also allows for computing only arbitrary parts of it. Furthermore, it leads to the first algorithm to determine only the Betti numbers without computing the whole resolution. It is even possible to obtain individual Betti numbers. We briefly comment on a first implementation of this approach in the CoCoALib and on some benchmarks comparing its performance with the corresponding functions in Macaulay2 and Singular.

2 Involutive Bases and Free Resolutions

Involutive bases represent a special kind of (typically non-reduced) Gröbner bases with additional combinatorial properties. They were introduced by Gerdt and Blinkov [3] who combined ideas from the Janet-Riquier theory of differential equations with the classical Gröbner bases theory; for an in depth discussion of their basic properties see [4] or [5, Chapts. 3&4]. In such a basis one associates with each generator a subset of the variables as multiplicative variables (determined by the chosen involutive division) and it is then only allowed to multiply this generator by polynomials in these variables. Every involutive basis defines a Stanley decomposition of the ideal (or submodule) generated by it and thus yields immediately the Hilbert function. For computing involutive bases one uses algorithms different from the Buchberger algorithm and in benchmarks it was shown that, even if one is only interested in the reduced Gröbner basis, these are highly competitive.

In [6], it was shown that in particular Pommaret bases lead to a much closer intertwining of theoretical and computational commutative algebra than standard Gröbner bases, as many important invariants like the regularity or the depth are immediately visible from a Pommaret basis. The main reason is their syzygy theory. By the involutive Schreyer theorem one obtains a Pommaret basis of the first syzygy module by considering the involutive standard representation of the products of the generators with their respective *non*-multiplicative variables. As with the classical Schreyer theorem, iteration leads then to a free resolution. The crucial difference to the classical situation is that here one can make precise statements about the *shape* of the arising resolution without any further computations. In particular, one immediately obtains upper bounds for the

* e-mail mario.albert@mathematik.uni-kassel.de
** e-mail fetzer@mathematik.uni-kassel.de
*** Corresponding author: e-mail seiler@mathematik.uni-kassel.de

Betti numbers which in the special case of a componentwise linear ideal are even the exact values. This leads to the following result which trivially entails Hilbert's syzygy theorem.

Theorem 2.1 ([6, Theorem 6.1]) *Let $\mathcal{H}$ be the Pommaret basis of the submodule $\mathcal{U} \subseteq \mathcal{P}^m$ over the polynomial ring $\mathcal{P} = \Bbbk[x_1, \dots, x_n]$ and denote by $\beta_0^{(k)}$ the number of generators in $\mathcal{H}$ with k multiplicative variables. If $d = \min\{k \mid \beta_0^{(k)} > 0\}$, then there exists a free resolution*

$$0 \longrightarrow \mathcal{P}^{r_{n-d}} \longrightarrow \cdots \longrightarrow \mathcal{P}^{r_0} \longrightarrow \mathcal{U} \longrightarrow 0 \tag{1}$$

where the ranks of the free modules are given by

$$r_i = \sum_{k=1}^{n-i} \binom{n-k}{i} \beta_0^{(k)} . \tag{2}$$

For notational simplicity, we omitted here the gradings of the free modules and thus gave only a bound for the total Betti numbers. It is, however, trivial to provide a bigraded version of this theorem. The formula for the ranks is a consequence of the fact that, although we do not explicitly know the higher syzygies without further computations, we can predict the leading terms of the Pommaret bases of each syzygy module. The resolution (1) is in general non minimal (the minimal resolution is obtained only for componentwise linear modules). However, its "bounding box" is optimal: it has always the same length and the same maximal breadth as the minimal resolution.

Theorem 2.2 ([6, Theorems 8.11 & 9.2]) *In the situation of Theorem 2.1, the projective dimension of the submodule $\mathcal{U}$ is $n - d$ and its Castelnuovo-Mumford regularity equals the maximal degree of a generator in the Pommaret basis $\mathcal{H}$.*

Theorem 2.1 also holds for some other involutive bases, e. g. for Janet bases which possess from a computational point some advantages over Pommaret bases. In the recent paper [7], we developed a term order free version of this result using so-called marked bases which are of considerable interest for the construction of Hilbert schemes. An axiomatic framework unifying and generalising these different variants based on the novel concept of a resolving decomposition is contained in [8]. However, in all these generalisations one looses Theorem 2.2, as one now only obtains simple bounds for the projective dimension and the regularity from the given basis. In the case of Janet bases, we showed in [2] with an explicit example that the difference in the breadth of the resolution can even become arbitrarily large. However, in benchmarks with ideals more typical for applications, it turned out that very often one still obtains an optimal "bounding box" and in the few cases where this was not the case, the difference was very small.

3 Algebraic Discrete Morse Theory

Algebraic discrete Morse theory was independently developed by Sköldberg [9] and by Jöllenbeck and Welker [10]. It provides techniques for the explicit reduction of non-minimal resolutions based on graph theoretical considerations. With every resolution a graph is associated which encodes the positions of the non-vanishing entries of the differentials. Any Morse matching in this graph allows then the construction of a smaller resolution. In particular, this approach even allows for the reduction of infinite resolutions to finite ones in one step.

In a later work [11], Sköldberg showed how a finite resolution can be obtained for any polynomial module admitting a presentation of a special form. The starting point is a two-sided Koszul complex which defines a resolution of finite length by with modules of infinite rank. Then it is assumed that the given polynomial module $\mathcal{M}$ has *initially linear syzygies* (for a given term order), i. e. that it possesses a presentation

$$0 \longrightarrow \ker \eta \longrightarrow \bigoplus_{\alpha=1}^{m} \mathcal{P}\mathbf{w}_\alpha \xrightarrow{\eta} \mathcal{M} \longrightarrow 0 \tag{3}$$

where the leading module $\operatorname{lt} \ker \eta$ is generated by elements of the form $x_i \mathbf{w}_\alpha$. If $x_i \mathbf{w}_\alpha$ is a generator of $\operatorname{lt} \ker \eta$, then the variable x_i is called *critical* for $\mathbf{w}_\alpha$. Any such presentation induces a Morse matching on the two-sided Koszul complex that leads to finite resolution of the submodule $\mathcal{U}$. One can derive an explicit closed form expression for the differential in this resolution which we only sketch here. We denote the free generators of the ℓth module in the resolution by $\mathbf{v}_I \mathbf{w}_\alpha$ where $I = (i_1, \dots, i_\ell)$ is an ascending integer sequence $1 \leq i_1 < \cdots < i_\ell \leq n$ such that for each entry i_k the variable x_{i_k} is critical for the generator $\mathbf{w}_\alpha$. Then the differential can be expressed in the form

$$d(\mathbf{v}_I \mathbf{w}_\alpha) = \sum_{K,\mu,\gamma} \sum_{J,\beta} \sum_{\pi} \rho_\pi \big(Q^{I,\alpha}_{K,\mu,\gamma} \mathbf{v}_K \eta(x^\mu \mathbf{w}_\gamma) \big) \tag{4}$$

where the ranges of the various summations follow from the Morse graph associated to the resolution and in particular π ranges over certain reduction pathes in this graph (depending on J and β). The determination of the maps ρ_π and of the constants $Q^{I,\alpha}_{K,\mu,\gamma} \in \Bbbk$ requires some normal form computations.

Sköldberg did not consider the question when a presentation of the special form (3) actually exists and how one could find it. Given an arbitrary presentation $\mathcal{M} \cong \mathcal{P}^m/\mathcal{U}$ with a polynomial submodule $\mathcal{U} \subseteq \mathcal{P}^m$, any resolution of $\mathcal{U}$ immediately yields one of $\mathcal{M}$. We showed in [1] that any Pommaret basis of $\mathcal{U}$ automatically induces a presentation of $\mathcal{U}$ of the required form where the critical variables are just the non-multiplicative variables of the generators. Thus given a Pommaret basis of a submodule $\mathcal{U}$, it have now two resolutions available: the one given by Theorem 2.1 and the one given by Sköldberg's approach with differential (4). Although the underlying construction principles appear very different, we have the following result.

Theorem 3.1 ([1]) *The two resolutions are isomorphic. More precisely, Sköldberg's construction yields for each syzygy module a Pommaret basis with the same leading terms as in the resolution* (1)*.*

The first statement is simple to prove, as the relation between critical and multiplicative variables, respectively, immediately implies that the two resolutions have the same shape and then the claim follows from standard arguments (see e. g. [12, Theorem 1.16]). The second statement is considerably harder and it implies that the two resolutions are essentially the same. There can only be some irrelevant differences in the tail terms of the various Pommaret bases. By combining Sköldberg's construction with Pommaret bases, it is thus transformed from a theoretical tool to a fully effective algorithm for the explicit determination of free resolutions of polynomial modules.

4 Betti Numbers

"Morally", it should be much cheaper to determine only the Betti numbers instead of a whole resolution, as the Betti numbers encode only the shape of the minimal resolution without any information about the differential. However, in practise one observes that in every computer algebra system the time to compute a Betti diagram is essentially the same as the time for a whole resolution. The reason is simple. The standard approach to determine Betti numbers is as follows. A not necessarily minimal resolution is computed and then tensored with the ground field $\Bbbk$ considered as $\mathcal{P}$-module. Then elementary linear algebra over $\Bbbk$ yields the Betti numbers. The effect of the tensoring is that all non-constant terms in the differential and thus most of the data just computed are immediately thrown away.

In our approach, we have the explicit formula (4) for the differential of our resolution. A closer analysis of it (detailed in [1]) reveals that one can derive very simple criteria which summands will lead to constant entries in the differential. This observation makes it feasable to compute directly only the constant part of the differential without any other terms in the differential. This leads to a drastic reduction in the computation time. In fact, it is even easily possible to compute individual Betti numbers, as one can simply determine only the relevant portion of the constant part of the differential.

A more theoretical and technically rather involved analysis of the differential (4) shows that in principle it even suffices to determine a very small number of constant entries (one per generator in the Pommaret basis). All other constant entries are identical with these up to at most a sign. Currently, it is not yet clear to what extent this result can be used for a further acceleration of the computations.

5 Implementation and Benchmarks

We implemented the above described approach to computing resolutions and Betti numbers in the C++ library CoCoALib [13] underlying the computer algebra system CoCoA specialised in commutative algebra. This library also contains an implementation of Janet bases by the authors. More detailed information about the implementation can be found in [2]. We performed a large number of benchmark computations using standard test examples in Gröbner bases theory. We compared the performance of our implementation for both computing minimal resolutions and computing only Betti numbers with Macaulay2 [14] and Singular [15]. Detailed tables with timing and a lot of other data can be found in [1, 2]. Here we only comment on some basic observations.

For the medium sized examples used as benchmarks, our implementation was often slower for determining minimal resolutions. The main reason is that the resolution induced by a Pommaret basis is sometimes much larger than the minimal one and then considerable time has to be spent for minimising it (a process for which we currently use a very simple method which probably can still be optimised). It turned out that the crucial parameter is the size of the Pommaret basis compared to the size of the minimal Gröbner basis. In particular for modules with a structure similar to toric ideals, the Pommaret basis can be much larger. However, it could also be observed that our approach scales much better than the traditional ones and hence the larger the examples are the more competitive our implementation becomes. One should also remark that our approach can be trivially parallelised in a massive way. In principle, one could use for the determination of each entry of the differential a different core, as the required computations are independent of each other.

If one is only interested in the Betti numbers, the outcome was completely different. For most examples, our implementation was not just a bit faster but by orders of magnitude. Thus it was possible to consider much larger examples and the already above mentioned scaling effect became even more pronounced. Of course, these results are not really surprising, as the constant part represents usually only a very small portion of the whole resolution and thus the total size of the resolution becomes much less relevant. We may conclude that currently our approach represents for most examples the by far fastest method for determining Betti numbers.

Based on the above mentioned observations, one approach for further improvements consists of using alternative involutive bases which lead to smaller bases in typical examples. Here some progress has been made in recent years by Gerdt and Blinkov like the introduction of the alex division [16] and some generalisations of involutive bases [private communication]. In all cases it follows from the results in [8] that our approach remains valid, but we have not yet a working implementation.

References

[1] M. Albert, M. Fetzer, E. Sáenz de Cabezón, and W. Seiler, J. Symb. Comp. **68/2**, 4–26 (2015).

[2] M. Albert, M. Fetzer, and W. Seiler, Janet bases and resolutions in CoCoALib, in: Computer Algebra in Scientific Computing — CASC 2015, edited by V. Gerdt, W. Koepf, W. Seiler, and E. VorozhtsovLecture Notes in Computer Science 9301 (Springer-Verlag, Berlin, 2015), pp. 15–29.

[3] V. Gerdt and Y. Blinkov, Math. Comp. Simul. **45**, 519–542 (1998).

[4] W. Seiler, Appl. Alg. Eng. Comm. Comp. **20**, 207–259 (2009).

[5] W. Seiler, Involution — The Formal Theory of Differential Equations and its Applications in Computer Algebra, Algorithms and Computation in Mathematics 24 (Springer-Verlag, Berlin, 2010).

[6] W. Seiler, Appl. Alg. Eng. Comm. Comp. **20**, 261–338 (2009).

[7] M. Albert, C. Bertone, M. Roggero, and W. Seiler, Marked bases over quasi-stable modules, Preprint arXiv:1511.03547, 2015.

[8] M. Albert and W. Seiler, Resolving decompositions for polynomial modules, Preprint Uni Kassel, 2016.

[9] E. Sköldberg, Trans. Amer. Math. Soc. **358**, 115–129 (2006).

[10] M. Jöllenbeck and V. Welker, Minimal Resolutions via Algebraic Discrete Morse Theory, Memoirs Amer. Math. Soc. 197 (AMS, 2009).

[11] E. Sköldberg, Resolutions of modules with initially linear syzygies, Preprint arXiv:1106.1913, 2011.

[12] D. Eisenbud, The Geometry of Syzygies, Graduate Texts in Mathematics 229 (Springer-Verlag, New York, 2005).

[13] J. Abbott and A. Bigatti, CoCoALib: a C++ library for doing computations in commutative algebra, Available at http://cocoa.dima.unige.it/cocoalib.

[14] D. Grayson and M. Stillman, Macaulay2 1.6, a software system for research in algebraic geometry, 2013, http://www.math.uiuc.edu/Macaulay2/.

[15] W. Decker, G. Greuel, G. Pfister, and H. Schönemann, Singular 4-0-2 — a computer algebra system for polynomial computations, 2012, http://www.singular.uni-kl.de.

[16] V. Gerdt and Y. Blinkov, Involutive division generated by an antigraded monomial ordering, in: Computer Algebra in Scientific Computing — CASC 2011, edited by V. Gerdt, W. Koepf, E. Mayr, and E. VorozhtsovLecture Notes in Computer Science 6885 (Springer-Verlag, Berlin, 2011), pp. 158–174.