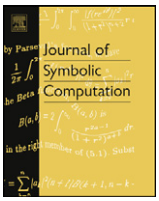
Journal of Symbolic Computation 48 (2013) 101–109

Contents lists available at SciVerse ScienceDirect

Journal of Symbolic Computation

www.elsevier.com/locate/jsc



Minimal generating sets of non-modular invariant rings of finite groups

Simon A. King

Fakultät für Mathematik und Informatik, Mathematisches Institut, Friedrich-Schiller-Universität Jena, D-07737 Jena, Germany

ARTICLE INFO

Article history:
Received 8 March 2012
Accepted 8 May 2012
Available online 15 May 2012

Keywords:
Invariant ring
Fundamental invariants
Irreducible secondary invariants
Truncated Gröbner basis

ABSTRACT

It is a classical problem to compute a minimal set of invariant polynomials generating the invariant ring of a finite group as a sub-algebra. We present here a new algorithmic solution in the non-modular case.

Our algorithm only involves very basic operations and is based on well-known ideas. In contrast to the algorithm of Kemper and Steel, it does not rely on the computation of primary and (irreducible) secondary invariants. In contrast to the algorithm of Thiéry, it is not restricted to permutation representations.

With the first implementation of our algorithm in SINGULAR, we obtained minimal generating sets for the natural permutation action of the cyclic groups of order up to 12 in characteristic 0 and of order up to 15 for finite fields. This was far out of reach for implementations of previously described algorithms. By now our algorithm has also been implemented in MAGMA.

As a by-product, we obtain a new algorithm for the computation of irreducible secondary invariants that, in contrast to previously studied algorithms, does not involve a computation of all reducible secondary invariants.

1. Introduction

Let G be a finite group linearly acting on a polynomial ring R over a field, such that the characteristic of R does not divide the order of G ("non-modular case"). It is well known that the invariant ring $R^G = \{r \in R:\ g.r = r,\ \forall g \in G\}$ is a finitely generated sub-algebra of R. A minimal (with respect to inclusion) set of homogeneous generators for R^G is called a set of *fundamental invariants*. Let $\beta(R^G)$ be the maximal degree occurring in fundamental invariants for R^G. By Noether's bound, $\beta(R^G) \leqslant |G|$.

E-mail address: simon.king@uni-jena.de.

0747-7171/$ – see front matter
http://dx.doi.org/10.1016/j.jsc.2012.05.002

Kemper and Steel (1999) (see also Derksen and Kemper, 2002) proposed an algorithm for the computation of fundamental invariants that works in three steps. First, one computes primary invariants of R^G. Secondly, one computes irreducible secondary invariants with respect to the primary invariants. Primary and irreducible secondary invariants together generate R^G. In the third step, one removes some primary invariants so that one eventually obtains a minimal generating set. Each of the three steps may be difficult, depending on the example. The Kemper–Steel algorithm has been implemented in various computer algebra systems, e.g., in Magma (Bosma et al., 1997) or Singular (Greuel et al., 2005).

Thiéry (2001c) suggested a direct algorithm for the computation of a minimal generating set. Thiéry's algorithm is not based on the computation of primary invariants, but uses the incremental construction of *SAGBI bases*. The disadvantage is that it is restricted to the special case of permutation groups, i.e., groups acting as subgroup of the symmetric group of the set of variables of R. Moreover, Thiéry's algorithm crucially depends on good a priori estimates for $\beta(R^G)$. Unfortunately, well-known upper bounds for $\beta(R^G)$ are, in general, far from being optimal. Thiéry's algorithm is implemented in the library PerMuVAR of MuPAD (Thiéry, 2001a). The extensive benchmark at Thiéry (2001b) compares the implementation of the Kemper–Steel algorithm in Magma with Thiéry's algorithm in MuPAD.

By Noether's degree bound, $\beta(R^G) \leqslant |G|$. However, even if $|G|$ is small, the resulting invariant rings can be surprisingly complex. For example, consider the natural permutation action of the finite cyclic group C_n of order n on $R = K[x_1, \ldots, x_n]$, where K is either $\mathbb{Q}$ or a finite prime field of characteristic coprime to n. For $n = 9$, there are 119 fundamental invariants for R^{C_n}, which was first found by Thiéry (2001c); at that time the fundamental invariants for $n \geqslant 10$ were unknown.

The aim of this paper is to describe another algorithm to compute fundamental invariants in the non-modular case. Our algorithm is direct and is thus fundamentally different from the Kemper–Steel algorithm. But in contrast to Thiéry's algorithm, it is not restricted to the case of permutation groups. Moreover, it does not rely on an a priori upper bound for $\beta(R^G)$. Instead, while incrementally constructing the set of generators, we obtain information allowing us to estimate $\beta(R^G)$ a posteriori.

With the implementation of our algorithm in Singular, we found fundamental invariants for R^{C_n} for small finite fields K in coprime characteristic for all $n \leqslant 15$. For $n = 15$ and $K = GF(2)$, a set of fundamental invariants is formed by 1494 polynomials. To the best of the author's knowledge, this example is unfeasible for the current implementations of both Kemper–Steel's and Thiéry's algorithm.

In early 2007, we compared the implementation of our algorithm in Singular 3-0-3 with the implementation of the Kemper–Steel algorithm in Magma V2.13-8, based on all permutation groups on 7 and 8 variables and some further actions on 9 and 10 variables. Our algorithm proved very efficient and was often faster by factors between 50 and 1000; details are available in King (2007b).

Since version V2.15, Magma uses our algorithm as well (Bosma et al., 2010), so that with the current versions of the computer algebra systems we can only compare different implementations of the same algorithm. Therefore and for the sake of brevity, we restrict our benchmarks essentially to the afore-mentioned natural permutation actions of cyclic groups. The implementation of our algorithm in Magma often performs better than the one in Singular, which is probably due to a faster computation of Gröbner bases in Magma.

1.1. Outline of the algorithm

Let $S \subset R^G$ be formed by the elements of degree $\leqslant d-1$ of a set of fundamental invariants, and let $I \subset R$ be the ideal generated by S. Fundamental invariants of degree d are found among the images of the degree d standard monomials of I under the Reynolds operator. Whether or not a monomial image is a fundamental invariant, can be tested using a Gröbner basis up to degree d of I. If sufficiently many generators are found, I will be 0-dimensional. Thus, I only has finitely many standard monomials, and we denote their maximal degree by $\beta(S)$. There can be no fundamental invariants beyond degree $\beta(S)$, so that the algorithm will eventually stop.

Our algorithm uses Gröbner bases in two ways. Firstly, Gröbner bases with a degree bound are used to detect fundamental invariants. Kemper and Steel (1999) achieve the same by solving linear algebra problems that may become rather huge. Of course, mathematically, there is not much of

a difference between these two approaches. Secondly, Gröbner bases help to determine whether an incrementally constructed set of elements of R^G is a generating set.

A modification of our algorithm can be used to compute irreducible secondary invariants for a given set of primary invariants. So, in a way, our algorithm is opposite to the Kemper–Steel algorithm: They use irreducible secondary invariants for computing fundamental invariants, whereas we use fundamental invariants to compute irreducible secondary invariants.

2. Ingredients of the algorithm

This section recalls some definitions and provides various easy lemmas that we use to prove the correctness of the algorithm that we describe in detail in the final subsection of this section.

We fix a monomial order on a polynomial ring R with n variables over some field K. Let G be a finite group, linearly acting on R. We denote the action of $g \in G$ on $r \in R$ by $g.r \in R$. Let $R^G = \{r \in R:\ g.r = r,\ \forall g \in G\}$ be the invariant ring. Obviously, it is a sub-algebra of R, and we aim at computing a minimal set of homogeneous generators for R^G. We study here the *non-modular* case, i.e., the characteristic of K does not divide the order of G. Note that according to Kemper (1998), algorithms for the non-modular case are useful also in the modular case.

2.1. The Reynolds operator

Considering the non-modular case, we can use the Reynolds operator $\mathrm{Rey}: R \to R^G$ defined by

$$\mathrm{Rey}(r) = \frac{1}{|G|} \sum_{g \in G} g.r$$

for $r \in R$. Note that, if G happens to act by permuting variables, expressing the Reynolds operator in terms of orbit sums can provide a more efficient computation. The following is well known and easy to prove:

Lemma 1. *The Reynolds operator* $\mathrm{Rey}: R \to R^G$ *is a surjective morphism of* R^G*-modules that restricts to the identity on* R^G.

2.2. Truncated Gröbner bases

The completeness criterion of our algorithm relies on the computation of a homogeneous Gröbner basis of the ideal generated by the fundamental invariants. For efficiency, we compute this Gröbner basis degree by degree, as we find the generators of the invariant ring in increasing degree, and we also use these partially computed Gröbner bases for the detection of new generators of the invariant ring. Let us define the notions involved here. For the theoretical background, we refer to Greuel and Pfister (2008).

If an ideal $I \subset R$ is homogeneous (i.e., it can be generated by homogeneous polynomials) then it has a Gröbner basis $\mathcal{G}$ formed by homogeneous polynomials. The S-polynomial of two homogeneous polynomials of degree d is homogeneous of degree at least d, and if $p \in R$ is homogeneous of degree d, then the normal form of p with respect to a set of homogeneous polynomials is either zero or homogeneous of degree d. It follows that Buchberger's algorithm can be used to compute the elements of $\mathcal{G}$ incrementally, in increasing degrees.

Definition 2.

(1) For any subset $S \subset R$, we denote by $\langle\langle S \rangle\rangle \subset R$ the sub-algebra generated by S, and by $\langle S \rangle \subset R$ the ideal generated by S. For $d > 0$, let R_d^G be the set of homogeneous invariant polynomials of degree d. For an ideal $I \subset R$, let $lm(I)$ be the set of leading monomials of elements of I.

(2) A finite set $\{g_1, \ldots, g_k\} \subset I$ of homogeneous polynomials is a *homogeneous Gröbner basis up to degree* d of the ideal $\langle g_1, \ldots, g_k \rangle$, iff $\mathrm{rem}(S(g_i, g_j); g_1, \ldots, g_k) = 0$ or $\deg(S(g_i, g_j)) > d$, for all $i, j = 1, \ldots, k$.

(3) If $\mathcal{G}_d$ is a homogeneous Gröbner basis up to degree d of $\langle S\rangle$ and $p \in R$ is a homogeneous polynomial of degree $\leqslant d$, then let $\mathrm{rem}(p;\mathcal{G}_d)$ be the *normal form* of p obtained by iterated polynomial division by the elements of $\mathcal{G}_d$.

It is an easy consequence of Buchberger's criterion, that the subset $\mathcal{G}_d \subset \mathcal{G}$ formed by the elements of degree at most d of a homogeneous Gröbner basis $\mathcal{G}$ is a homogeneous Gröbner basis up to degree d of the ideal $\langle \mathcal{G}_d\rangle \subset \langle \mathcal{G}\rangle$, in the sense of the previous definition.

If $\mathcal{G}$ is a homogeneous Gröbner basis of an ideal I and $p \in R$ is homogeneous of degree d, then the classical solution of the ideal membership problem asserts that $p \in I$ if and only if $\mathrm{rem}(p;\mathcal{G}) = 0$. Since p can only be reduced by polynomials of degree $\leqslant d$, we obtain $p \in I$ if and only if $\mathrm{rem}(p;\mathcal{G}_d) = 0$.

In particular, it follows that $\langle \mathcal{G}_d\rangle$ coincides with the ideal generated by the elements of $\langle \mathcal{G}\rangle$ of degree at most d.

2.3. Finding generators of the invariant ring

In this subsection, we discuss where we should search for further generators if we want to extend an incomplete generating set $S \subset R^G$.

Definition 3. For $S \subset R$, let $mon(S)$ be the set of all standard monomials of $\langle S\rangle$, i.e., those monomials of R that are not contained in $lm(\langle S\rangle)$. Let $mon_d(S)$ be the standard monomials of $\langle S\rangle$ of degree d. Let $B_d(S) = \mathrm{Rey}(mon_d(S))$.

Note that $mon_d(S)$ and thus $B_d(S)$ are easy to compute if a homogeneous Gröbner basis at least up to degree d of $\langle S\rangle$ is known. Moreover, if $\mathcal{G}$ is a homogeneous Gröbner basis of some homogeneous ideal, then $mon_d(\mathcal{G}) = mon_d(\mathcal{G}_d)$, since $\langle \mathcal{G}_d\rangle$ contains all elements of $\langle \mathcal{G}\rangle$ of degree at most d.

Lemma 4. *Let $S \subset R^G$. Then, $S \cup \mathrm{Rey}(mon(S))$ is a generating set for R^G. In particular, if $R^G_d \not\subset \langle\langle S\rangle\rangle$ then $B_d(S) \not\subset \langle\langle S\rangle\rangle$.*

Proof. By the graded Nakayama lemma (Lemma 3.5.1 in Derksen and Kemper, 2002), $mon(S)$ generates R as $\langle\langle S\rangle\rangle$-module. Therefore and since $\mathrm{Rey}: R \to R^G$ is a surjective homomorphism of $\langle\langle S\rangle\rangle$-modules, $S \cup \mathrm{Rey}(mon(S))$ is a generating set for R^G. The second part of the lemma is a direct consequence. □

So, in increasing degree d starting with $d = 1$ and $S = \emptyset$, we may loop through all b in the finite set $B_d(S)$, and add b to the set S of previously found generators if $b \notin \langle\langle S\rangle\rangle$. In that way, one incrementally constructs a generating set of R^G, consisting of homogeneous invariant polynomials. In fact, it is a *minimal* generating set (Thiéry, 2001c). We can test whether $b \in \langle\langle S\rangle\rangle$ according to the following lemma. Again, the lemma is well known, but we include a proof for completeness.

Lemma 5. *Let $S \subset R^G$ be a set of homogeneous invariant non-constant polynomials. Assume that $R^G_{d'} \subset \langle\langle S\rangle\rangle$ for all $d' < d$, and assume that we are in the non-modular case. Let $b \in R^G_d$. We have $b \in \langle\langle S\rangle\rangle$ if and only if $b \in \langle S\rangle$.*

Proof. If $b \in \langle\langle S\rangle\rangle$ then $b \in \langle S\rangle$. Any $b \in \langle S\rangle$ can be written as a finite sum $b = \sum_i p_i q_i$ with homogeneous polynomials $p_i \in R$ and $q_i \in S$. We have $b = \mathrm{Rey}(b) = \sum_i \mathrm{Rey}(p_i) q_i$ by Lemma 1. Since the elements of S are non-constant, the p_i are of degree at most $d - 1$. Hence, $\mathrm{Rey}(p_i) \in R^G_{d'}$ for some $d' < d$. Thus $\mathrm{Rey}(p_i) \in \langle\langle S\rangle\rangle$ by hypothesis. Therefore, $b \in \langle\langle S\rangle\rangle$. □

We verify $b \in \langle S\rangle$ by reduction of b with respect to a homogeneous Gröbner basis $\mathcal{G}$ of $\langle S\rangle$ up to degree d. After adding b to the set of generators, we easily obtain a homogeneous Gröbner basis up to degree d of $\langle S \cup \{b\}\rangle$, by the following lemma.

Lemma 6. *Let $\mathcal{G} \subset R$ be a homogeneous Gröbner basis up to degree d of $\langle \mathcal{G} \rangle$. Let $b \in R$ be a homogeneous polynomial of degree d, and $b \notin \langle \mathcal{G} \rangle$. Then $\mathcal{G} \cup \{\mathrm{rem}(b; \mathcal{G})\}$ is a homogeneous Gröbner basis up to degree d of $\langle \mathcal{G} \cup \{b\} \rangle$.*

Proof. Let $r = \mathrm{rem}(b; \mathcal{G})$. Since $b \notin \langle \mathcal{G} \rangle$ and both b and the elements of $\mathcal{G}$ are homogeneous, we have $r \neq 0$, $\deg(r) = d$, and $\langle \mathcal{G} \cup \{b\} \rangle = \langle \mathcal{G} \cup \{r\} \rangle$.

By hypothesis, the S-polynomials of pairs of elements of $\mathcal{G}$ are of degree $> d$ or reduce to 0 modulo $\mathcal{G}$. We now consider the S-polynomials of r and elements of $\mathcal{G}$. Let $g \in \mathcal{G}$. By definition of the normal form, we have $lm(g) \nmid lm(r)$. Therefore the S-polynomial of r and g is of degree $> d = \deg(r)$. This implies that $\mathcal{G} \cup \{r\}$ is a homogeneous Gröbner basis up to degree d of $\langle \mathcal{G} \cup \{b\} \rangle$. □

2.4. The completeness criterion

There remains one problem: The preceding lemmas allow for an incremental construction of a minimal generating set of R^G, in increasing degrees—but in what degree shall we stop the construction? By definition, we can stop after having found the generators in degree $\beta(R^G)$. So, we could adopt a general estimate for $\beta(R^G)$ like Noether's bound $\beta(R^G) \leqslant |G|$. However, such general a priori estimates are very often far from being optimal. Therefore, we prefer to derive an estimate for $\beta(R^G)$ from the previously constructed generators, due to the following proposition.

Definition 7. For $S \subset R$, define $\beta(S) = \sup\{\deg(p) \colon\ p \in mon(S) \cup S\}$.

Proposition 8. *Let $S \subset R^G$ be a finite set. We have $\beta(R^G) \leqslant \beta(S)$. If S is a generating set of R^G, then $\langle S \rangle$ is zero-dimensional, and thus $\beta(S)$ is finite and can be computed from a Gröbner basis of $\langle S \rangle$.*

Proof. It is a direct consequence of Lemma 4 that $\beta(R^G) \leqslant \beta(S)$.

Recall that R is a polynomial ring with n variables, say, $x_1, \ldots, x_n$. Since G is finite, we obtain an integral equation for x_i over R^G by $\prod_{g \in G}(t - g.x_i) \in R^G[t]$. Hence, R is integral over R^G and is thus a finite dimensional R^G-module.

Now assume $\langle\langle S \rangle\rangle = R^G$. By the graded Nakayama lemma (Lemma 3.5.1 in Derksen and Kemper, 2002), the standard monomials of $\langle S \rangle$ provide $\langle\langle S \rangle\rangle$-module generators for R. Since R is a finite dimensional R^G-module, $\langle\langle S \rangle\rangle$ only has a finite number of standard monomials, and is thus zero-dimensional. □

2.5. The algorithm

Starting with $S = \emptyset$, we successively add fundamental invariants to S. Our strategy is to work with a truncated rather than with a complete homogeneous Gröbner basis of $\langle S \rangle$, whenever possible. Sometimes, a truncated Gröbner basis also suffices for testing whether $\langle S \rangle$ is of dimension 0. However, in general one needs a Gröbner basis of $\langle S \rangle$ *without* degree restriction for that purpose. To avoid needless computations, we use the following trick.

By definition, in degree $\beta(R^G)$ we will find a homogeneous generator of R^G, but in degree $\beta(R^G) + 1$ we don't. Hence, only if our incremental construction of S arrives at some degree d, such that there is an element of S in degree $d-2$ but none in degree $d-1$, does it make sense to compute a Gröbner basis of $\langle S \rangle$ without degree restriction. If $\dim(\langle S \rangle) = 0$, which is tested using the Gröbner basis, then we obtain an estimate $\beta(R^G) \leqslant \beta(S)$ by Proposition 8 telling us in what degree we can stop the incremental search. We thus obtain the following algorithm for the computation of a minimal generating set of R^G, where G is any finite matrix group.

Algorithm Invariant Algebra

(1) Construct the Reynolds operator $\mathrm{Rey} \colon R \to R^G$.
Let $S = \mathcal{G} = \emptyset$ and define $\beta = \beta(S) = \infty$.

(2) For increasing degree d, starting with $d = 1$:
  (a) If S contains elements of degree $d-2$ but no elements of degree $d-1$:
    (i) Replace $\mathcal{G}$ by a *complete* Gröbner basis of $\langle S\rangle$.
    (ii) If $\dim(\langle S\rangle) = 0$ (which is tested using $\mathcal{G}$), then replace β by $\beta(S)$, which is easily computed using $\mathcal{G}$. If $d > \beta$ then break and return S.
    If S contains elements of degree $d-1$, then replace $\mathcal{G}$ by a homogeneous Gröbner basis $\mathcal{G}$ of $\langle S\rangle$ up to degree d.
  (b) Compute $B_d(S)$ using $\mathcal{G}$ and Rey.
  (c) For all $b \in B_d(S)$:
    If $\mathrm{rem}(b;\mathcal{G}) \neq 0$ then replace S by $S \cup \{b\}$ and $\mathcal{G}$ by $\mathcal{G} \cup \{\mathrm{rem}(b;\mathcal{G})\}$.
  (d) If $d = \beta$ then break and return S.

Theorem 9. *Algorithm* Invariant Algebra *returns a set of fundamental invariants of* R^G.

Proof. By Lemma 6, $\mathcal{G}$ is a homogeneous Gröbner basis of $\langle S\rangle$ at least up to degree d in all steps of the algorithm. By Lemmas 4 and 5, the algorithm successively extends S to a set of fundamental invariants. By Proposition 8, the algorithm terminates in finite time, but not before S generates R^G. □

Our algorithm has a very simple structure based on elementary methods, and probably this is why it works so well. It differs widely from both the algorithms described in Kemper and Steel (1999) (using a Hironaka decomposition and linear algebra) and in Thiéry (2001c) (using SAGBI bases).

One group action is particularly difficult to study with our algorithm: The natural action of the symmetric group $\mathfrak{S}_n$ on n variables. Of course, the set S of elementary symmetric polynomials is a minimal generating set, and $\beta(R^{\mathfrak{S}_n}) = n$. However, for $n = 8$, one has $\beta(S) = 28$.

Nevertheless, in most of the examples that we computed, our completeness criterion $\beta(R^G) \leqslant \beta(S)$ works almost perfectly, i.e., $\beta(S) - \beta(R^G)$ is usually equal to zero and only in few cases bigger than one. And of course, the result for the natural action of the symmetric group is well known from theory, so we can certainly live with that exception.

3. Computational results

3.1. Transitive permutation groups

A classical test bed for the computation of minimal generating sets of invariant rings of finite groups is provided by transitive permutation groups (Thiéry, 2001c, 2001b). These are groups acting on a polynomial ring R over a field K by permuting variables, such that any two variables are related by the group action. An extensive benchmark comparing Thiéry's algorithm implemented in MuPAD with Kemper and Steel's algorithm implemented in Magma is provided by Thiéry (2001b). Benchmarks involving an implementation of our algorithm in Singular 3-0-3 are provided in King (2007b), based on all permutation actions on 7 or 8 variables and some examples on 9 and 10 variables. In the majority of examples, the implementation of our algorithm worked much faster than the implementation of the Kemper–Steel algorithm in Magma V2.13-8, typically by factors between 50 and 1000.

3.2. Cyclic groups

Here, we consider the natural permutation action of the cyclic group $\mathcal{C}_n$ of order n on $R = K[x_1, \ldots, x_n]$, where K is either $\mathbb{Q}$ or a finite prime field of characteristic coprime to n, and the generator of C_n maps x_i to x_{i+1} for $i = 1, \ldots, n-1$ and x_n to x_1.

The maximal degree occurring in a minimal generating set is, by Noether's bound, of course at most $|\mathcal{C}_n| = n$, hence, quite small. However, the minimal number of generators of $R^{\mathcal{C}_n}$ is surprisingly large. According to Thiéry (2001b), the invariant ring of $\mathcal{C}_{10}$ in characteristic 0 was out of reach at that time. With our algorithm, we obtain fundamental invariants for $\mathcal{C}_{12}$ in characteristic zero and even for $\mathcal{C}_{15}$ in characteristic 2.

Table 1
Natural action of C_n on n variables (characteristic 0). SINGULAR version 3-0-3 on a Linux x86_64 platform with AMD Opteron 248 processors (2.2 GHz) and a memory limit of 16 GB.

n	time [s]	mem. [Mb]	# generators (sorted by degree)
6	0.05	0.746	1, 3, 6, 6, 2, 2
7	0.17	1.25	1, 3, 8, 12, 12, 6, 6
8	1.54	2.25	1, 4, 10, 18, 16, 8, 4, 4
9	35.6	11.92	1, 4, 14, 26, 32, 18, 12, 6, 6
10	298.3	54.16	1, 5, 16, 36, 48, 32, 12, 8, 4, 4
11	1187	116	1, 5, 20, 50, 82, 70, 50, 30, 20, 10, 10
12	2010 min	2160	1, 6, 24, 64, 104, 84, 36, 20, 12, 8, 4, 4

Table 2
Natural action of C_n on n variables (characteristic $p > 0$). SINGULAR version 3-0-3 on a Linux x86_64 platform with AMD Opteron 248 processors (2.2 GHz) and a memory limit of 16 GB.

n	p	time [s]	mem. [Mb]	# generators (sorted by degree)
6	5	0.03	0.746	1, 3, 6, 6, 2, 2
7	2	0.09	0.746	1, 3, 8, 12, 12, 6, 6
8	3	0.34	1.25	1, 4, 10, 18, 16, 8, 4, 4
9	2	1.65	1.86	1, 4, 14, 26, 32, 18, 12, 6, 6
10	3	12.7	4.48	1, 5, 16, 36, 48, 32, 12, 8, 4, 4
11	2	73.5	9.33	1, 5, 20, 50, 82, 70, 50, 30, 20, 10, 10
12	5	693	33.2	1, 6, 24, 64, 104, 84, 36, 20, 12, 8, 4, 4
13	2	4079	81.1	1, 6, 28, 84, 168, 180, 132, 84, 60, 36, 24, 12, 12
14	3	25 280	304.3	1, 7, 32, 104, 216, 242, 162, 96, 42, 30, 18, 12, 6, 6
15	2	99 873	780.4	1, 7, 38, 130, 306, 388, 264, 120, 88, 56, 40, 24, 16, 8, 8

Table 3
Natural action of C_n on n variables (characteristic 0), on a Compute-Server i7 (2.8 GHz, 16 GB). Computation time in [s].

n	6	7	8	9	10	11
SINGULAR 3-1-4	0.03	0.08	0.53	11.1	105.3	1427.9
MAGMA V2.17-13	0.01	0.02	0.08	0.53	3.07	42.73

Tables 1 and 2 provide the timings with an old version of SINGULAR, which we used when we first computed these invariant rings. For $n \leqslant 5$ the computations are finished in almost no time, so we omit them in our tables. Table 1 provides the result for $n = 6, \ldots, 12$ in characteristic 0. Table 2 provides the results for $n = 6, \ldots, 15$ in small prime characteristic $p > 0$, of course such that p does not divide n (non-modular case). Apparently this is much easier than characteristic 0. The reason is that in characteristic 0 the coefficients occurring in the Gröbner bases become very huge. By consequence, it takes too long to compute normal forms.

In Table 3, we provide more recent timings: We compare the implementations of our algorithm in SINGULAR 3-1-4 and MAGMA V2.17-13, on a Compute-Server i7 (2.8 GHz, 16 GB).

Note that in all examples, the number of generators in each degree is the same in characteristic 0 and in non-modular prime characteristic. It is in fact conjectured that this is always the case (Thiéry, 2007).

3.3. Permutation action on pairs

To work in prime characteristic is not the only way to simplify the computations. Following a suggestion of Kemper (2006), we study here the action of $\mathfrak{S}_5$ on non-ordered pairs, which yields a 10-dimensional representation of $\mathfrak{S}_5$. We could describe that representation of $\mathfrak{S}_5$ by a transitive permutation group on 10 variables. However, in that formulation of the problem, our algorithm would take a very long time to find a minimal generating set. One can decompose the representation into a 1-, a 4- and a 5-dimensional irreducible representation, and in this form, the representation is given by the matrices

$$M_1 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & -\frac{2}{3} & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{2}{3} & \frac{1}{3} & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{2}{3} & -\frac{2}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix},$$

$$M_2 = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & -\frac{2}{3} & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{2}{3} & \frac{1}{3} & -\frac{2}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{2}{3} & -\frac{2}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & -1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 \end{pmatrix}.$$

Originally, we used it as an example for the computation of irreducible secondary invariants, but of course it is also a nice example for the computation of a minimal generating set. Our algorithm Invariant Algebra executed in Singular version 3-0-2 finds the fundamental invariants after 47.8 minutes using 4.4 GB in characteristic 0 respectively after 84.2 seconds using 81.7 MB in characteristic 7. In both cases, there is a minimal number of $1, 2, 4, 7, 10, 13, 13, 4, 2$ generators sorted by degree.

However, that example illustrates a potential disadvantage of using Gröbner bases. Namely, by changes in the computation of Gröbner bases in Singular, the computation of the same example takes considerably longer with version 3-1-3 than with 3-0-2. In contrast, Magma V2.17-13 seems to compute the occurring Gröbner bases more easily, and only needs 37.5 s on a Compute-Server i7 (2.8 GHz, 16 GB) to compute the fundamental invariants, even in characteristic 0.

3.4. Application to irreducible secondary invariants

An algorithm for the computation of secondary invariants is described in Kemper (1998), Kemper and Steel (1999) and Derksen and Kemper (2002). Sometimes one is only interested in *irreducible* secondary invariants. By a slight modification, our algorithm provides a very efficient way to compute irreducible secondary invariants, without the need to compute the reducible secondary invariants as well.

For that aim, let P be a set of primary invariants. In Step (1) of algorithm Invariant Algebra, let $S = P$ and let $\mathcal{G}$ be a Gröbner basis of P. The rest of the algorithm remains unchanged. In the end, it returns the union of P with a set of irreducible secondary invariants. Note that this algorithm does not involve an application of Molien's Theorem. So, it even applies when the Molien series is difficult to compute.

Our algorithm works particularly well if there are many secondary invariants but only few *irreducible* secondary invariants. This is the case in the following example, that was originally motivated by our work in low-dimensional topology (King, 2006, 2007a). We work in $R = \mathbb{Q}[x_1, \ldots, x_{20}]$. To simplify notation, let e_i be the column vector with 1 in position i and 0 otherwise. Then, we obtain a 20-dimensional representation of the symmetric group $\mathfrak{S}_3$ by the following two matrices:

$$M_1 = (e_2e_1e_3e_{19}e_9e_{13}e_{17}e_{11}e_5e_{15}e_8e_{16}e_6e_{14}e_{10}e_{12}e_7e_{20}e_4e_{18}),$$

$$M_2 = (e_1e_3e_2e_4e_6e_5e_{10}e_9e_8e_7e_{13}e_{16}e_{11}e_{19}e_{20}e_{12}e_{18}e_{17}e_{14}e_{15}).$$

We use the following sub-optimal primary invariants:

$$x_1 + x_2 + x_3, \quad x_1x_2 + x_1x_3 + x_2x_3, \quad x_1x_2x_3, \quad x_4 + x_{14} + x_{19},$$
$$x_4x_{14} + x_4x_{19} + x_{14}x_{19}, \quad x_4x_{14}x_{19}, \quad x_5 + x_6 + x_8 + x_9 + x_{11} + x_{13},$$
$$x_8x_9 + x_5x_{11} + x_6x_{13}, \quad x_6x_8 + x_5x_9 + x_{11}x_{13},$$
$$x_5x_8 + x_6x_9 + x_6x_{11} + x_9x_{11} + x_5x_{13} + x_8x_{13},$$
$$x_5x_6x_{11} + x_5x_8x_{11} + x_8x_9x_{11} + x_5x_6x_{13} + x_6x_9x_{13} + x_8x_9x_{13},$$
$$x_5^6 + x_6^6 + x_8^6 + x_9^6 + x_{11}^6 + x_{13}^6, \quad x_{12} + x_{16}, \quad x_{12}x_{16},$$
$$x_7 + x_{10} + x_{15} + x_{17} + x_{18} + x_{20}, \quad x_7x_{17} + x_{10}x_{18} + x_{15}x_{20},$$
$$x_{10}x_{15} + x_{17}x_{18} + x_7x_{20}, \quad x_7x_{15} + x_{10}x_{17} + x_7x_{18} + x_{15}x_{18} + x_{10}x_{20} + x_{17}x_{20},$$
$$x_7x_{10}x_{17} + x_7x_{15}x_{17} + x_7x_{10}x_{18} + x_{15}x_{17}x_{20} + x_{10}x_{18}x_{20} + x_{15}x_{18}x_{20},$$
$$x_7^6 + x_{10}^6 + x_{15}^6 + x_{17}^6 + x_{18}^6 + x_{20}^6.$$

In this example, there are 248 832 secondary invariants of maximal degree 26, among which are 283 irreducible secondary invariants of maximal degree 4. The sheer number of secondary invariants (which can be computed by Molien's Theorem) makes the computations unfeasible for any algorithm that is based on the generation of power products, as the one described in Kemper (1998), Kemper and Steel (1999) and Derksen and Kemper (2002). However, our algorithm executed in SINGULAR 3-0-3 just needs few seconds to compute all irreducible secondary invariants.

References

Bosma, W., Cannon, J., Playoust, C., 1997. The Magma algebra system I: The user language. J. Symbolic Comput. 24 (3/4), 235–265.

Bosma, W., Cannon, J.J., Fieker, C., Steel, A. (Eds.), 2010. Handbook of Magma Functions. Edition 2.16. 5017 pp.

Derksen, H., Kemper, G., 2002. Computational invariant theory. In: Invariant Theory and Algebraic Transformation Groups, I. In: Encyclopaedia Math. Sci., vol. 130. Springer-Verlag, Berlin.

Greuel, G.-M., Pfister, G., Schönemann, H., 2005. SINGULAR 3-0-2. A Computer Algebra System for Polynomial Computations. Centre for Computer Algebra, University of Kaiserslautern. http://www.singular.uni-kl.de.

Greuel, G.M., Pfister, G., 2008. A Singular Introduction to Commutative Algebra, second ed. Springer, Berlin.

Kemper, G., 1998. Computational invariant theory. In: The Curves Seminar at Queen's, vol. XII. Kingston, ON, 1998. In: Queen's Papers in Pure and Appl. Math., vol. 114. Queen's Univ., Kingston, ON, pp. 5–26.

Kemper, G., Steel, A., 1999. Some algorithms in invariant theory of finite groups. In: Dräxler, P., Michler, G.O., Ringel, C.M. (Eds.), Proceedings of the Euroconference on Computational Methods for Representations of Groups and Algebras. Essen, 1997. In: Progr. Math., vol. 173. Birkhäuser, Basel, pp. 267–285.

Kemper, G., 2006. Personal communication.

King, S., 2006. Ideal Turaev–Viro invariants. Sib. Èlektron. Mat. Izv. 3, 62–66.

King, S., 2007a. Ideal Turaev–Viro invariants. Topology Appl. 154, 1141–1156.

King, S., 2007b. Minimal generating sets of non-modular invariant rings of finite groups. Preprint, arXiv:math/0703035v3.

Thiéry, N.M., 2001a. Library PerMuVAR of MuPAD. http://PerMuVAR.sourceforge.net/.

Thiéry, N.M., 2001b. Summary of computations of minimal generating sets of invariant rings of transitive permutation groups. Comparative benchmark. See http://permuvar.sourceforge.net/Groups/.

Thiéry, N.M., 2001c. Computing minimal generating sets of invariant rings of permutation groups with SAGBI-Gröbner basis. In: Discrete Models: Combinatorics, Computation, and Geometry. Paris, 2001. In: Discrete Math. Theor. Comput. Sci. Proc. AA. Maison Inform. Math. Discrèt. (MIMD), Paris, pp. 315–328.

Thiéry, N.M., 2007. Personal communication.