Enhanced Computations of Gröbner Bases in Free Algebras as a New Application of the Letterplace Paradigm

Viktor Levandovskyy
RWTH Aachen University
Templergraben 64
52062 Aachen, Germany
viktor.levandovskyy@math.rwth-aachen.de

Grischa Studzinski*
RWTH Aachen University
Templergraben 64
52062 Aachen, Germany
grischa.studzinski@rwth-aachen.de

Benjamin Schnitzler†
RWTH Aachen University
Templergraben 64
52062 Aachen, Germany
benjamin.schnitzler@rwth-aachen.de

ABSTRACT

Recently, the notion of "letterplace correspondence" between ideals in the free associative algebra $K\langle\mathbf{X}\rangle$ and certain ideals in the so-called letterplace ring $K[\mathbf{X} \mid \mathbb{N}]$ has evolved. We continue this research direction, started by La Scala and Levandovskyy, and present novel ideas, supported by the implementation, for effective computations with general non-graded ideals in the free algebra by utilizing the generalized letterplace correspondance. In particular, we provide a direct algorithm to compute Gröbner bases of non-graded ideals. Surprizingly we realize its behavior as "homogenizing without a homogenization variable". Moreover, we develop new shift-invariant data structures for this family of algorithms and discuss about them.

Furthermore we generalize the famous criteria of Gebauer-Möller to the non-commutative setting and show the benefits for the computation by allowing to skip unnecessary critical pairs. The methods are implemented in the computer algebra system SINGULAR. We present a comparison of performance of our implementation with the corresponding implementations in the systems MAGMA [BCP97] and GAP [GAP13] on the representative set of nontrivial examples.

Categories and Subject Discriptors:
G.4 [**Mathematical Software**]: Algorithm design and analysis; Certification and testing;
I.1.2 [**Algorithms**]

General Terms:
Algorithms, Theory

Keywords:
non-commutative Groebner Bases, Free Algebra, Gebauer-Moeller, non-graded Ideals, Letterplace

*Supported by DFG priority project of SPP 1489, Project LE 2697/2-1

†Partially supported by DFG priority project SPP 1489, Project LE 2697/2-1

1. INTRODUCTION

Let K be an arbitrary field and $F = K\langle\mathbf{X}\rangle$ be the free associative algebra. Consider $P = K[\mathbf{X} \mid \mathbb{N}]$ the corresponding letterplace ring as introduced in [LL09]. Hereby the set of variables or *letters* $\mathbf{X}$ is blended with another structure, so-called *places* from $\mathbb{N}$. We denote the generators by $x_i(k)$ with x_i resp. k being the letter resp. the place. The *letterplace monoid* $[\mathbf{X} \mid \mathbb{N}]$ contains the neutral element 1 and all finite products $x_{i_1}(k_1)\ldots x_{i_d}(k_d)$ for $x_{i_j} \in X$ and $k_j \in \mathbb{N}$. The corresponding monoid ring is called the *letterplace ring* $K[\mathbf{X} \mid \mathbb{N}]$. It is an infinitely-generated commutative K-algebra, generated by the set $\{x_i(j) : x_i \in X, j \in \mathbb{N}\}$. Note, that $K[\mathbf{X} \mid \mathbb{N}]$ is not Noetherian and hence its ideals have usually infinite generating sets.

The monoid $\mathbb{N}$ acts on P by shifting the places, thus providing an important additional structure. We call the number $s(m) = \min\{k_i \mid m = x_{i_1}(k_1)\ldots x_{i_d}(k_d)\} - 1$ the *shift* of the monomial m. For each $n \in \mathbb{N}$ and $m = x_{i_1}(k_1)\ldots x_{i_d}(k_d) \in [\mathbf{X} \mid \mathbb{N}]$ n acts on m as $n \cdot m =: x_{i_1}(k_1+n)\ldots x_{i_d}(k_d+n)$. From now on $\cdot$ will be used to denote this action. We call the *place support* of $m \in [\mathbf{X} \mid \mathbb{N}]$ the set of places, occuring in m. A monomial m is called *place-multilinear*, if each number from the place support of m appears at most once.

Indeed, there is an embedding of K-vector spaces $\iota : F \to P$, $\iota(x_{i_1} \cdot \ldots \cdot x_{i_d}) = x_{i_1}(1)\ldots x_{i_d}(d)$ for all monomials $m = x_{i_1}\ldots x_{i_d} \in \langle\mathbf{X}\rangle$. However, ι is not a ring homomorphism. Denote by $V := im(\iota)$, then V is spanned by all monomials of $K[\mathbf{X} \mid \mathbb{N}]$, whose place support is of the form $1,\ldots,d$ for some $d \in \mathbb{N}$. In particular, such elements have shift 0 and are place-multilinear.

We aim at studying infinite structures, which are *invariant under the action of the shift* or shortly *shift-invariant.* Let $J \subset K[\mathbf{X} \mid \mathbb{N}]$ be an ideal, then it is *shift-invariant* if $\forall s \in \mathbb{N}$ holds $s \cdot J \subseteq J$. Observe, that for any ideal $L \subset K[\mathbf{X} \mid \mathbb{N}]$ the ideal $\cup_{s\in\mathbb{N}} s \cdot L$ is shift-invariant by construction. Let us define a shift-invariant vector space $V' = \cup_{s\in\mathbb{N}} s \cdot V$.

DEFINITION 1.1. *For a graded ideal $J \subset K[\boldsymbol{X} \mid \mathbb{N}]$, where the ring is graded with respect to the total degree function of letters* $\mathbf{deg}()$ *we denote by J_d the graded component of total degree d. We call J a letterplace ideal if J is generated by $\bigcup\limits_{s,d\in\mathbb{N}} s \cdot (J_d \cap V)$. In this case, J is shift-invariant and place-multigraded.*

The following theorem is the key result proven by Levandovskyy and La Scala in [LL09].

THEOREM 1.2. *Let $I \trianglelefteq K\langle \boldsymbol{X}\rangle$ be a graded ideal and set $J = \langle \iota(I)\rangle \subseteq K[\boldsymbol{X} \mid \mathbb{N}]$. Then J is a letterplace ideal of $K[\boldsymbol{X} \mid \mathbb{N}]$. Conversely, let $J \trianglelefteq K[\boldsymbol{X} \mid \mathbb{N}]$ be a letterplace ideal and set $I = \iota^{-1}(J \cap V)$. Then I is a graded ideal of $K\langle \boldsymbol{X}\rangle$. The mappings $I \to J$ and $J \to I$ define a bijective correspondence between graded ideals of $K\langle \boldsymbol{X}\rangle$ and letterplace ideals of $K[\boldsymbol{X} \mid \mathbb{N}]$.*

With this correspondence in mind we would like to introduce the main idea behind the computation of Gröbner bases via the letterplace approach: one adds all shifts of the elements of a generating set to the set and computes a commutative Gröbner basis. After removing superfluous elements one is then left with a generating system corresponding to a Gröbner basis.

DEFINITION 1.3.

- *Let J be a letterplace ideal of $K[X|P]$ and $H \subset K[X|P]$. We say that H is a letterplace basis of J if $H \subset \bigcup_{d\in\mathbb{N}} J_d \cap V$ and $\bigcup_{s\in\mathbb{N}} s \cdot H$ is a generating set of the ideal J.*
- *Let J be an ideal of $K[X|P]$ and $H \subset J$. Then H is called a (Gröbner) shift-basis of J if $\bigcup_{s\in\mathbb{N}} s \cdot H$ is a (Gröbner) basis of J.*

The following theorem from [LL09] derscribes the connection between Gröbner shift-basis and the original Gröbner basis for the ideal of the free algebra.

THEOREM 1.4.

- *Let I be a graded two-sided ideal of $K\langle \boldsymbol{X}\rangle$ and put $J = \langle \bigcup_{s\in\mathbb{N}} s \cdot \iota(I)\rangle$. Moreover, let $G \subset \bigcup_{d\in\mathbb{N}} I_d$ and define $H = \iota(G) \subset \bigcup_{d\in\mathbb{N}} J_d \cap V$. Then G is a generating set of I as a two-sided ideal if and only if H is a letterplace basis of J.*
- *Let $I \trianglelefteq K\langle \boldsymbol{X}\rangle$ be a graded two-sided ideal and put $J = \langle \bigcup_{s\in\mathbb{N}} s \cdot \iota(I)\rangle$. Moreover, let H be a Gröbner letterplace basis of J and put $G = \iota^{-1}(H \cap V) \subset \bigcup_{d\in\mathbb{N}} I_d$. Then G is a two-sided Gröbner basis of I.*

This allows one to use commutative methods in infinite dimension "up to shifting". Since $K\langle \mathbf{X}\rangle$ is not Noetherian as well, one expects a finite output rather rarely. Instead, one passes to the semi-algorithms, that is the computations are performed up to given degree bound.

ALGORITHM 1.5.

Input: *G_0, a generating set for an graded ideal $I \trianglelefteq K\langle \boldsymbol{X}\rangle$*
Output: *G, a Gröbner basis for I*
  $H := \iota(G_0 \setminus \{0\})$;
  $P = \{(f, s\cdot g) \mid f, g \in H, s \in \mathbb{N}, f \neq s\cdot g, \mathbf{gcd}(\mathbf{lm}(f), \mathbf{lm}(s\cdot g)) \neq 1, \mathbf{lcm}(\mathbf{lm}(f), \mathbf{lm}(s\cdot g)) \in V\}$;
  while $P \neq \emptyset$ ***do***
    Choose $(f, s\cdot g) \in P$;
    $P = P \setminus (f, s\cdot g)$;
    $h := \text{REDUCE}(S(f, s\cdot g), \bigcup_{t\in\mathbb{N}} t \cdot H)$;
    if $h \neq 0$ ***then***
      $P := P \cup \{(h, s\cdot g) \mid g \in H, s \in \mathbb{N}, \mathbf{gcd}(\mathbf{lm}(h), \mathbf{lm}(s\cdot g)) \neq 1, \mathbf{lcm}(\mathbf{lm}(h), \mathbf{lm}(s\cdot g)) \in V\}$;
      $P := P \cup \{(g, s\cdot h) \mid g \in H, s \in \mathbb{N}, \mathbf{gcd}(\mathbf{lm}(g), \mathbf{lm}(s\cdot h)) \neq 1, \mathbf{lcm}(\mathbf{lm}(g), \mathbf{lm}(s\cdot h)) \in V\}$;
      $H := H \cup \{h\}$;
    end if
  end while;
  $G := \iota^{-1}(H)$;
  return G;

As we can see, the algorithm resembles Buchberger's algorithm and in this formulation can be seen as a member of "critical pair and completion" family of algorithms. The crucial novelty is the presence of an additional structure via shift.

Analysis shows, that indeed the set P in the algorithm can be shortened to contain exactly the pairs corresponding to overlaps of leading monomials of the participating elements. This is due to the fact that the condition $\mathbf{lcm}(\mathbf{lm}(f), \mathbf{lm}(s\cdot g)) \in V$ gets rid of superfluous elements produced by senseless shifting. Nevertheless, in order to establish the set P as well as for the reduction step, a large amount of shifted polynomials will be generated.

The algorithm above has been implemented in SINGULAR in an almost verbatim way in order to check the new idea with letterplace and shifting. Nevertheless the naive implementation performed nicely [LL09, SL13]. In this article we report on the work done for optimizing and generalizing this algorithm.

For simplicity we will only consider graded monomial orderings, that is $m > n$ if $deg(m) > deg(n) \quad \forall m, n \in K\langle \mathbf{X}\rangle$.

2. A SEPARATING INVARIANT

It is easy to see that the letterplace Gröbner basis algorithm depends heavily on the computation of shifts and a large set of shifted polynomials is generated in each step. However, only a few of them are needed. Luckily there is better way to search for critical pairs. Therefore we start with a closer investigation of the shift action.

In commutative computer algebra one often uses *exponent vectors* in order to determine if a monomial divides another.

EXAMPLE 2.1. *Consider $m_1 = x_1^{a_1} \cdots x_n^{a_n}$, $m_2 = x_1^{b_1} \cdots x_n^{b_n} \in K[x_1, \ldots, x_n]$. Then $m_1 | m_2 \Leftrightarrow \forall i \; a_i \leq b_i$.*

For the free algebra there has been no direct way to attach exponent vectors from $\mathbb{N}^k$ to monomials of $\langle \mathbf{X}\rangle$. However, the letterplace ring is commutative, so there is a way to use the exponent vectors, since the support of a monomial is always finite.

EXAMPLE 2.2. *Consider $K[x_1, x_2, x_3 \mid \mathbb{N}]$ and take $p = x_1(1)x_3(2)x_2(3) + x_2(1)x_3(2)$. We order the variables by the lowest place first, that is $x_i(k) < x_i(l)$ if $k > l$. So for $x_1(1)x_3(2)x_2(3)$ we have the exponent vector $(1, 0, 0,\ 0, 0, 1,\ 0, 1, 0)$ and for $x_2(1)x_3(2)$ we have $(0, 1, 0,\ 0, 0, 1)$. In other words, one of the natural ways to order the variables of $K[\boldsymbol{X} \mid \mathbb{N}]$ is to use blocks of original variables; in the example it is*

$$x_1(1), x_2(1), x_3(1), x_1(2), x_2(2), x_3(2), x_1(3), x_2(3), x_3(3), \ldots$$

REMARK 2.3. *Take $m \in V' \subset K[\boldsymbol{X} \mid \mathbb{N}]$ of total degree d and shift s with exponent vector $e \in \mathbb{N}^k$. Then $k = d + s$ and $e = (e_1, \ldots, e_{d+s})$, where e_i is a block of length n for all $1 \leq i < d$. Then:*

- *For $1 \leq i < s$:* $\quad e_i$ *contains only zeros.*
- *For $s \geq i$:* $\quad e_i$ *contains exactly one* 1 *and* $n-1$ *zeros. Note that the one is on position j, if and only if $(x_j \mid i)|m$.*

Lemma 2.4. *Take two monomials $m, m' \in V'$ such that $m' = s \cdot m$ for some $s \in \mathbb{N}$ and construct their exponent vectors e and e', respectively, as explained above. Suppose that $\mathbf{deg}(m) = d$. By setting $\tilde{e} := (e'_s, \dots, e'_{s+d})$ we obtain $\tilde{e} = e$.*

Proof. Denote by $\tilde{m}$ the monomial corresponding to $\tilde{e}$. Since $shift(m') \geq s$ we have $e'_i = (0, \dots, 0)\ \forall i < s$. We conclude that $shift(\tilde{m}) = shift(m') - s$ and $s \cdot \tilde{m} = m'$, which already implies $\tilde{m} = m$ and thus $\tilde{e} = e$. $\square$

Definition 2.5. *Let $m \in V'$ with shift s and $\mathbf{deg}(m) = d > 0$ and assume the exponent vector e has been constructed as before. Set $\tilde{e} = (e_s, \dots, e_{d+s})$. Construct an integer vector D as follows: The first entry of D is the unique position of* 1 *in e_s. For $1 < i \leq d$ denote by z the number of zeros between the two occurring* 1 *in the vector (e_{s+i}, e_{s+i-1}). Then we set the $i-th$ entry of D to $z+1$. We call D a distance vector and denote by dv the map that assigns to each monomial $m \in V'$ its distance vector. Moreover, we postulate $dv(1) := 0 \in \mathbb{N}^1$.*

Example 2.6. *As above consider $p = x_1(1)x_3(2)x_2(3) + x_2(1)x_3(2)$. The distance vector for $x_1(1)x_3(2)x_2(3)$ is given by $(1,5,2)$ and for $x_2(1)x_3(2)$ we have $(2,4)$.*

Proposition 2.7. *The map dv is invariant with respect to the shift action, that is it separates the orbits. Moreover, for all $m, m' \in V'$ we have $dv(m) = dv(m') \Leftrightarrow m' = s \cdot m$ or $m = s \cdot m'$ for some $s \in \mathbb{N}$.*

In particular, this leads to the fact recognition of shifted monomials in $K[\mathbf{X} \mid \mathbb{N}]$ and at the same time – via letterplace encoding – to the divisibility check $m|m'$ for monomials $m, m' \in K\langle \mathbf{X} \rangle$.

Definition 2.8. *For two distance vectors d and d' we say that d is contained in d', if $d = 0$ or $size(d') \geq size(d)$ and there exists i such that $d[1] = d'[i] + in - \sum_{1}^{i-1} d'[i]$ and $d[j] = d'[i+j-1]$ for $1 < j \leq size(d)$.*

Lemma 2.9. *Let $m, m' \in V'$ be two monomials. Then $m|m'$ if and only if $dv(m)$ is contained in $dv(m')$.*

Corollary 2.10.
Take two monomials $w, w' \in \langle X \rangle$. Then $w|w' \Leftrightarrow dv(\iota(w))$ is contained in $dv(\iota(w'))$.

Remark 2.11.
The closer analysis of Algorithm 1.5 shows, that one creates new critical pairs in the pairset P with shifts of elements from the would-be-Gröbner-basis H. In addition, the reduction of a polynomial takes place with respect to all shifts of H. In the practical but still naive version of the algorithm, running with a given degree bound, one possibility is to store all shifts of elements of H and add to the source of the pairset all shifts of new polynomials as well. By this approach one can use the usual commutative divisibility on monomials and also use classical monomial orderings for comparisons.

Remark 2.12. *Now we follow a different way: by using distance vectors we can make a fast division test for monomials in V' respectively V. In particular, the shift of a monomial can be read off and stored during the computation of its distance vector.*

Thus keeping the distance vector of a leading monomial of a polynomial adjoint to the polynomial data in the Algorithm 1.5 directly improves the algorithm. Namely, the overlaps-based computation of critical pairs is more effective and one can directly use special optimized algorithms for the shift-divisibility and shift-reductions, vital for the performance of the Algorithm.

3. NON-GRADED IDEALS

Although the theoretical aspect of the correspondence between non-graded ideals and their homogenized counterparts is technically involved, the basic idea is similar to the classical homogenization in the commutative case. While being algorithmically feasible in the Noetherian case, the computation of a Gröbner basis of a non-graded ideal in the non-Noetherian case has the following problem: A non-graded ideal $I \in K\langle \mathbf{X} \rangle$ has a finite Gröbner basis, while the homogenized set of generators leads to an infinite Gröbner basis.

As mentioned before in [Sca12] La Scala proposed a generalization of the letterplace approach to the non graded case by introducing another variable to homogenize the generators of an ideal and translating the process of non commutative homogenization to the letterplace ring. While showing good results this approach does not solve the general problem.

Since there are concrete examples of this behavior, communicated to us by Victor Ufnarovskij, we are looking for direct Gröbner basis theory for non-graded ideals of $K\langle \mathbf{X} \rangle$.

3.1 Place grading, or homogenization without a homogenization variable

To improve the method of La Scala our first step is to show that introducing a new variable is superfluous. In fact one can use the structure given by the letterplace ring quite successfully.

Definition 3.1.

- *Denote by $W' \subset K[\boldsymbol{X} \mid \mathbb{N}]$ the vector space, spanned by all place-multilinear monomials, that is monomials of $[X \mid \mathbb{N}]$, whose place support is irredundant as a set. Let $W \subset W'$ be spanned by all place-multilinear monomials of shift zero.*
- *For a monomial $m \in W$, define the* ***place-degree*** *$\mathbf{pdeg}(m)$ to be the highest place occurring in the place-support of m and we set $\mathbf{pdeg}(m) = 0$ for $m \in K$ by convention. For a polynomial $p \in W \setminus \{0\}$ we set $\mathbf{pdeg}(p) = \max_i \{\mathbf{pdeg}(m_i) | p = \sum a_i m_i, a_i \in \mathbb{K} \setminus \{0\}\}$.*
- *If there is $1 \leq k \leq \mathbf{pdeg}(m)$, such that k is not in the place-support, we say m has a hole at place k. The number of holes between the first occurring variable and the last one is called the* ***place defect*** *of m.*
- *Let $\cdot_{lp}$ be the* ***letterplace multiplication*** *on $K[\boldsymbol{X} \mid \mathbb{N}]$ [LL09], that is $m_1 \cdot_{lp} m_2 = m_1(\mathbf{pdeg}(m_1) \cdot m_2)$ for monomials $m_1, m_2 \in [X \mid \mathbb{N}]$.*
- *Define $W_k = \{w \in W \mid \mathbf{pdeg}(w) = k\} \subseteq W$.*

Proposition 3.2. *The following holds:*

1. $W' = \bigcup_{s\in\mathbb{N}_0} s\cdot W$.
2. $\mathbf{pdeg}(m_1 \cdot_{lp} m_2) = \mathbf{pdeg}(m_1) + \mathbf{pdeg}(m_2) = \mathbf{pdeg}(m_2 \cdot_{lp} m_1)$ *and thus* $W_l \cdot_{lp} W_k \subseteq W_{l+k}$ $\forall l, k \in \mathbb{N}_0$.
3. $\forall m \in W$: $\mathbf{pdeg}(m) = \mathbf{deg}(m) +$ *shift*$(m)+$ *place-defect*(m).
4. $W = \bigoplus_{k\in\mathbb{N}_0} W_k$ *is graded with respect to place degree.*
5. $V_0 = W_0 = K$, $V_1 = W_1 = \oplus_{i=1}^n Kx_i(1)$ *and* $V_k \subsetneq W_k$ $\forall k \geq 2$. *Thus* $V \subsetneq W$ *and* $V' \subsetneq W'$.
6. *Place grading respects shifts, that is* $s \cdot W_k \subset W_{k+s}$ $\forall k, s \in \mathbb{N}$ *holds.*

Example 3.3. *To get a deeper insight into the structure of* W *a look into the graded parts is useful. It is easy to see that* $W_0 = K\cdot\{1\} = V_0$ *and* $W_1 = K\cdot\{x_i \mid 1 \leq i \leq n\} = V_1$. *According to the definition,* $W_2 = V_2 \oplus (1\cdot V_1)$. *Further on, we find* $W_3 = V_3 \oplus (1\cdot W_2) \oplus (W_1 \times 2\cdot W_1)$, *where* $W_1 \times 2\cdot W_1 = \{w(2\cdot\tilde{w}) \mid w, \tilde{w} \in W_1\}$. *Substituting previous expressions we obtain* $W_3 = V_3 \oplus (1\cdot V_2) \oplus (2\cdot V_1) \oplus (V_1 \times 2\cdot V_1)$.

Recall that the letterplace ring is equipped with the shift action, which also defines an equivalence relation: $m_1 \simeq m_2$ if either $m_1 = s\cdot m_2$ or $m_2 = s\cdot m_1$ for some $s \in \mathbb{N}$. Using this relation we can identify a monomial of the free algebra with the orbit of monomials from $K[\mathbf{X} \mid \mathbb{N}]$ under the shift action. A natural choice for the representative of an orbit is $\iota(m) \in V$.

Definition 3.4. *Let* $G \subset K\langle \boldsymbol{X}\rangle$ *be a set of polynomials and put* $\tilde{G} = \iota(G)$. *Then for each* $\tilde{p} = \sum_i a_i\tilde{m}_i \in \tilde{G}$ *with* $a_i \in K, \tilde{m}_i \in [X \mid \mathbb{N}]$ *we set*

$$p_h = \sum_i a_i(\mathbf{pdeg}(\tilde{p}) - \mathbf{pdeg}(m_i)) \cdot m_i \in K[\boldsymbol{X} \mid \mathbb{N}].$$

Then p_h *is graded with respect to* $\mathbf{pdeg}$ *(or place-homogeneous). We call* p_h *the place-homogenization of* $\tilde{p}$.

So instead of adding a new variable to the free algebra one is able to use places to homogenize polynomials. Note that normally one adds commutators to the ideal in order to allow the new variable to be commuted. This is also possible with the use of holes in view of the fact, that any representative of an orbit can be chosen, which implies that any polynomial in $K[\mathbf{X} \mid \mathbb{N}]$ can be viewed as place-homogenized.

Using this new place-homogenization the algorithm presented in [LL09] can be applied, since the homogenization is shift-invariant. However, dehomogenization is not as simple, since monomials with positive place-defect are not in an orbit with an element of V under the shift-action. However, if a s-polynomial is computed or one does reduction it is easy to see that polynomials with positive place-defect will occur.

Definition 3.5. *Define a* K*-linear map* $\mathbf{shrink} : W' \to V$ *as follows:* $\mathbf{shrink}(m) = m$ *for* $m \in V$. *Now suppose that* $m \in W' \setminus V$ *with* $1 \leq \mathbf{deg}(m) = d$ *and* $\mathbf{pdeg}(m) = d' > d$, *then there exist* $s_1, \ldots, s_d \in \mathbb{N}_0$ *such that* $m = s_1 \cdot x_{i_1}(1) \ldots s_d \cdot x_{i_d}(d)$, *where* $s_d = d' - d \geq 1$. *We put* $\mathbf{shrink}(m) := x_{i_1}(1)\cdots x_{i_d}(d) \in V$.

Definition 3.6.
- *Define an equivalence relation on* W' *respectively on* W *by* $m_1 \sim m_2 \Leftrightarrow \mathbf{shrink}(m_1) = \mathbf{shrink}(m_2)$. *We set* $\sim: W' \to W'/\sim$ *to be the natural surjection.*
- *Define a map* $\star : V \times V \to W'/\sim$ *as follows:* $v_1 \star v_2 := [v_1(\mathbf{pdeg}(v_1)\cdot v_2)]$.

Lemma 3.7. *Define a* K*-linear map*

$$\eta : V \to W'/\sim, \quad f = \sum_i a_i m_i \mapsto [f].$$

Then η *is an isomorphism of vector spaces.*

Proof. Since V is stable under shrinking, η is injective. Let $w \in W'$ be place-multilinear, then $\mathbf{shrink}(w) \in V$ belongs to $[w]$ and thus η is surjective. □

Note that the inverse map for η is given by $\mathbf{shrink}$. If we identify a residue class $[w] \in W'/\sim$ with the unique element $v \in V$ contained in this class we can think of $\star$ as a multiplication on V, which respects the total degree of polynomials, thus giving V an K-algebra structure.

Lemma 3.8. *Define the map* $\star : V \times V \to V$, $(v_1, v_2) \mapsto \mathbf{shrink}(v_1(\mathbf{pdeg}(v_1)\cdot v_2))$.

1. $\star$ *is bilinear.*
2. *We have* $p \star q = 0 \Leftrightarrow p = 0 \vee q = 0$.
3. $\star$ *is associative.*

Proof.
1. Recall that $\mathbf{pdeg}$ of a polynomial is the highest occurring place in any monomial with non-zero coefficient. Moreover, we have $\mathbf{shrink}(v_1(s\cdot v_2)) = \mathbf{shrink}(v_1(s'\cdot v_2))$ $\forall v_1, v_2 \in V$ if $s, s' \geq \mathbf{pdeg}(v_1)$ holds. The claim follows by the linearity of $\mathbf{shrink}$, shift and the multiplication on $K[\mathbf{X} \mid \mathbb{N}]$ as a ring.
2. Because we have $p \star q = (\sum_d^D p_d) \star (\sum_e^E q_e) = \sum_{k=0}^{D+E}(\sum_{d=0}^k p_d q_{k-d})$, where p_d, q_e are $\mathbf{pdeg}-$graded components of p and q respectively we need to proof the claim for $\mathbf{deg}$-graded components $\sum_d p_d q_{k-d}$ only. Because $p, q \in V$ we have that $\sum_d p_d q_{k-d}$ is homogeneous and the claim follows by the fact, that we can use the letterplace multiplication for the graded case.
3. Routine computation. □

Indeed the map $\star$ can be extended to a map $K[\mathbf{X} \mid \mathbb{N}] \times K[\mathbf{X} \mid \mathbb{N}] \to V$, which enjoys similar properties.

THEOREM 3.9. *Define a K-linear map*

$$\vartheta : (K\langle \boldsymbol{X}\rangle, \cdot) \to (V, \star),\ p = \sum_{c_j \in K\setminus\{0\}} c_j m_j \mapsto \sum c_j \iota(m_j).$$

Then ϑ is a K-algebra isomorphism.

PROOF. By definition ϑ is K-linear and we have $\vartheta(p + q) = \vartheta(p) + \vartheta(q)\ \forall p, q \in K\langle \mathbf{X}\rangle$. We need to show that $\vartheta(p \cdot q) = \vartheta(p) \star \vartheta(q)\ \forall p, q \in K\langle \mathbf{X}\rangle$. We have $\vartheta(p) \star \vartheta(q) = \vartheta(\sum_i a_i p_i) \star \vartheta(\sum_j b_j q_j) = \sum_{i,j} a_i b_i \vartheta(p_i) \star \vartheta(q_j)$. Because $\vartheta(p_i) \star \vartheta(q_j) = \iota(p_i) \star \iota(q_j) = \mathbf{shrink}(\iota(p_i)(\mathbf{deg}(p) \cdot \iota(q_j))) = \iota(p_i)(\mathbf{deg}(p_i) \cdot \iota(q_j)) = \iota(p_i q_j)$ where we used the remark given in the proof of 3.8. So we have $\vartheta(p) \star \vartheta(q) = \sum_{i,j} a_i b_i \iota(p_i q_j) = \sum_{i,j} a_i b_i \vartheta(p_i q_j) = \vartheta(p \cdot q)$ using linearity of ϑ. Because of 3.8 (2) ϑ is injective. Since V is defined as the image of ι we have that ϑ is also surjective, which completes the proof. □

We have revealed that one can interpret holes in letterplace monomials as traces of the appearance of homogenization variable. The results of La Scala for correspondence of ideals and generating systems, especially Gröbner bases can be used for the correspondence here, thereby proving the correctness of our method.

3.2 Saturation on the fly

Knowing about the problem behind homogenization mentioned earlier there are two steps one can take in order to avoid it. The first one is to apply an ordering which allows one to simplify the homogenization in each step and the second one is an alternative for homogenization which natural occurs on the letterplace ring, namely the use of distance vectors.

In our opinion, by using graded techniques on the homogenized ideal, our aim is not to compute the trusted homogenized ideal, but to come as directly as possible to the non-graded Gröbner basis, which is usually obtained via the post-computation of the saturation.

As the first step let us recall the homogenization.

DEFINITION 3.10. *Consider the free algebra $K\langle \boldsymbol{X}\rangle$ and let h be a new variable commuting with all $x_i \in \boldsymbol{X}$. Define $\overline{\boldsymbol{X}} = \boldsymbol{X} \cup \{h\}$ and $F = K\langle \overline{\boldsymbol{X}}\rangle$. Then each $p \in K\langle \boldsymbol{X}\rangle$ is the image of some homogeneous element $\overline{p} \in K\langle \overline{\boldsymbol{X}}\rangle$ under the algebra homomorphism Φ defined via $\Phi(x_i) = x_i$, $\Phi(h) = 1$. More precisely, if we have $f = \sum_{k=0}^{d} p_k$ with $p_k \in K\langle \boldsymbol{X}\rangle_k$, $p_d \neq 0$, then $\tilde{p} = \sum_{k=0}^{d} p_k h^{d-k}$ is a homogeneous element, satisfying $\Phi(\tilde{p}) = p$.*

REMARK 3.11. *In order to compute a Gröbner basis via classical homogenization one has to employ an ordering that has the following property: $h^k \mid \mathbf{lm}(\tilde{p})$ then h^k divides each term occurring in $\tilde{p}$ with non-zero coefficient. An example for such orderings can be found in [Li12] and in [BB98] as well as in [Mor88] there is a full introduction to this topic. In particular, note that for a homogenized polynomial $\tilde{p}$ we always have $h \nmid \mathbf{lm}(p)$ with respect to such an ordering.*

After one has computed the Gröbner basis of a homogenized ideal, a saturation of the result with respect to h must be computed. If we introduce the commutators to the homogenized ideal one is always able to move the homogenization variable to the end of each monomial using reduction if needed. Indeed, for each computed s-polynomial p, such that $h^k|\mathbf{lm}(p)$, one can replace p with the polynomial p/h^k. This procedure is called saturation on the fly, because p/h^k belongs to the saturated homogenized ideal. This allows one to reduce significantly the total degree of considered polynomials during the computation. Note, that the somewhat analogous effect in the commutative case can be achieved by using the notion of ecart.

Recognizing holes as traces of the homogenization, one can apply the method presented by La Scala rather effectively. The big advantage hereby is that one does not need to introduce an extra variable and in each step of the algorithm a sort of saturation-on-the-fly is applied. Also, it is not necessary to choose a special ordering for the homogenization variable. In the following we present the full algorithm.

Note that the classical operations with polynomials (creation of s-polynomials, reductions etc.) usually produces holes in the monomials of inhomogeneous input. Hence the new reduction routine SHRINK-REDUCE is introduced, which applies shrinking after each elementary reduction step $f = f - c_i m_i h$, where h is an appropriately shifted reductor, $c_i \in K \setminus \{0\}$ and $m_i \in V$.

ALGORITHM 3.12.
Input: *G_0, a generating set for an ideal $I \trianglelefteq K\langle \boldsymbol{X}\rangle$*
Output: *G, a Gröbner basis for I*
  $H := \iota(G_0 \setminus \{0\})$;
  $P = \{(f, s\cdot g) \mid f, g \in H, s \in \mathbb{N}, f \neq s\cdot g, \mathbf{gcd}(\mathbf{lm}(f), \mathbf{lm}(s\cdot g)) \neq 1, \mathbf{lcm}(\mathbf{lm}(f), \mathbf{lm}(s\cdot g)) \in V\}$;
  while *$P \neq \emptyset$* ***do***
    Choose $(f, s\cdot g) \in P$;
    $P = P \setminus (f, s\cdot g)$;
    $h :=$ SHRINK-REDUCE*$(\mathbf{shrink}(S(f, s\cdot g)), \bigcup_{t\in\mathbb{N}} t\cdot H)$;*
    if *$h \neq 0$* ***then***
      $P := P \cup \{(h, s\cdot g) \mid g \in H, s \in \mathbb{N}, \mathbf{gcd}(\mathbf{lm}(h), \mathbf{lm}(s\cdot g)) \neq 1, \mathbf{lcm}(\mathbf{lm}(h), \mathbf{lm}(s\cdot g)) \in V\}$;
      $P := P \cup \{(g, s\cdot h) \mid g \in H, s \in \mathbb{N}, \mathbf{gcd}(\mathbf{lm}(g), \mathbf{lm}(s\cdot h)) \neq 1, \mathbf{lcm}(\mathbf{lm}(g), \mathbf{lm}(s\cdot h)) \in V\}$;
      $H := H \cup \{h\}$;
    end if
  end while;
  $G := \iota^{-1}(H)$;
  return *G;*

THEOREM 3.13. *If the algorithm above terminates it returns a reduced Gröbner basis for the ideal I.*

PROOF. As explained before H can be viewed as a set of homogenized generators, where holes were added at the end of each monomial. Since leading monomials are not affected by the homogenization P clearly contains all critical pairs, as shown in the proof for graded ideals.
So the only thing to prove is the correctness of the computation of H, which is clear by the correspondence given in the previous section. □

3.3 Applying the new data structure and the new homogenization

Equipped with the knowledge that we can compute Gröbner bases of non-graded ideals using the letterplace approach without introducing direct homogenization one can ask if there is a better way, because applying shrinking to each new s-polynomial can be very inefficient. Luckily there is better way.

As in the section before the methods of distance vectors can be applied to reconstruct Buchberger's original algorithm. It is important to note that the ideal and generating set correspondence still holds even without explicit homogenization. In addition to that, distance vectors can be used to represent monomials in the shift invariant way. By switching to this new representation one can multiply monomials more effectively.

PROPOSITION 3.14. *Denote by* $\mathbf{lg}$ *the size of a distance vector. For two monomials* $m_1, m_2 \in \langle X \rangle$ *set* $\tilde{m}_1 := \iota(m_1)$, $\tilde{m}_2 := \iota(m_2)$, $dm_1 := dv(\tilde{m})$, $dm_2 := dv(\tilde{m}_2)$.
Define a new vector d *by setting* $d[1 \dots \mathbf{lg}(dm_1)] = dm_1$,
$d[\mathbf{lg}(dm_1)+1] = \mathbf{lg}(dm_1)n - (\sum_{k=1}^{\mathbf{lg}(dm_1)} dm_1[k]) + dm_2[1]$,
$d[(\mathbf{lg}(dm_1)+2)\dots(\mathbf{lg}(dm_1)+\mathbf{lg}(dm_2))] = dm_2[2\dots\mathbf{lg}(dm_2)]$. *Then* $dv(\iota(m_1 m_2)) = d$.

PROOF. To see that the claim is correct one only needs to notice that the entry $d[\mathbf{lg}(dm_1)+1]$ is exactly the gap in the exponent vector of $\tilde{m}_1 \mathbf{lg}(dm_1) \cdot \tilde{m}_2$ between the last variable of $\tilde{m}_1$ and the first of $\mathbf{lg}(dm_1) \cdot \tilde{m}_2$. $\square$

REMARK 3.15. *Using this multiplication allows one to completely eliminate the need for shrinking. Since the shift is not needed either, one has a sparse representation of the orbit under the shift action on a monomial. Since the procedure is directly inherited from the methods of homogenization, its correctness is granted.*

4. GEBAUER-MÖLLER'S CRITERION

In commutative as well as in non-commutative Gröbner basis theory it is well-known that the practical use of criteria to reduce the set of critical pairs has very effective impact on the performance. Out of several criteria, first formulated by Buchberger, the product criterion in the case of free algebras is naturally applied during the consideration of overlaps of polynomials. The chain criterion applies as well, but it can be refined further, following the work of Gebauer and Möller [GM88] in the commutative case. Gebauer-Möller's criterion has been generalized to the setup of modules in [KR00] and [KR05], while in the non-commutative case Mora gave a detailed presentation of superfluos pairs in [Mor94], which was adapted to fit practical computations, as for example in [Xiu12].
Here we will present the theoretical layout as well as the practical use of the criterion in the letterplace framework.
We want to point out that our research was done simultaneously and independently, comparing to the recent work [KX13].

For this section we will assume that each set $P \subset K\langle \mathbf{X} \rangle$ is *interreduced*, meaning $\forall p, q \in P, p \neq q : \mathbf{lm}(p) \nmid \mathbf{lm}(q)$ and that each $p \in P$ is monic.

4.1 The non-commutative theory

In the non-commutative version of Buchberger's algorithm one constructs s-polynomials from so-called *obstructions*, that is a six-tuple $(l, p, r; \lambda, q, \rho)$ with $l, r, \lambda, \rho \in K\langle \mathbf{X} \rangle$, $p, q \in P$ and $\mathbf{lm}(lpr) = \mathbf{lm}(\lambda q \rho)$.
The classical "product criterion theorem" states that only those pairs need to be considered, leading monomials of which involve an *overlap*, that is $\mathbf{lm}(p) = ab$ and $\mathbf{lm}(q) = bc$ for some monomials a, b, c. Therefore one only has to consider pairs $\pi = (1, p_i, r; \lambda, p_j, 1)$, such that $\mathbf{lm}(p_i r) = \mathbf{lm}(\lambda p_j)$.

DEFINITION 4.1. *For an obstruction* $\pi = (1, p_i, r; \lambda, p_j, 1)$ *we denote by* $\mathbf{cm}(\pi) := \mathbf{lm}(p_i r) = \mathbf{lm}(p_i) r = \lambda \mathbf{lm}(p_j)$ *the common multiple of* p_i *and* p_j *with respect to the overlap considered in* π.

Let us consider a set of polynomials P and construct the set of all critical pairs $\pi(P)$ by searching for overlaps in the leading monomials. We want to apply the criteria to $\pi(P)$ to reduce its size.

THEOREM 4.2.
Suppose that we are given a set of polynomials P*, its set of critical pairs* $\pi(P)$ *and a pair* $\pi = (1, p_i, r_i; \lambda_k, p_k, 1) \in \pi(P)$.

1. *If there exist two pairs* $\pi_1 = (1, p_i, r_i'; \lambda_j, p_j, 1), \pi_2 = (1, p_j, r_j; \lambda_k', p_k, 1) \in \pi(P) \setminus \{\pi\}$*, such that* $\mathbf{lm}(p_j) | \mathbf{cm}(\pi)$*, then the s-polynomial* $s(\pi)$ *of* π *will reduce to zero.*

2. *If there exists a pair* $\pi_1 = (1, p_j, r_j; \lambda_k', p_k, 1) \in \pi(P) \setminus \{\pi\}$*, such that* $\mathbf{cm}(\pi_1)$ *divides* $\mathbf{cm}(\pi)$ *from the right, then the s-polynomial* $s(\pi)$ *of* π *will reduce to zero.*

PROOF. 1. Because of the assumptions we have $\mathbf{lm}(p_j) = abc$, $\mathbf{lm}(p_k) = bct_k$ and $\mathbf{lm}(p_i) = t_i ab$ for some monomials a, b, c, t_i, t_k. Since P is interreduced, none of the leading monomials can divide the overlap cofactors. This implies $\lambda_k = t_i a$ and $r_i = ct_k$. Moreover, the existence of π_1 and π_2 and the form of the leading monomials imply that there exist pairs $\pi_1' = (1, p_i, c; t_i, p_j, 1)$ and $\pi_2' = (1, p_j, t_k; a, p_k, 1)$. Then $s(\pi) = p_i ct_k - t_i a p_k = t_i abct_k + \mathbf{tail}(p_i)ct_k - t_i abct_k - t_i a \mathbf{tail}(p_k) \to -t_i \mathbf{tail}(p_j)t_k + \mathbf{tail}(p_i)ct_k + t_i \mathbf{tail}(p_j)t_k - t_i a \mathbf{tail}(p_k) = -s(\pi_1')t_k - t_i s(\pi_s') \to 0$.
Note that the reductions used are performed according to the fixed monomial ordering.

2. We first note that $\mathbf{lm}(p_j)r_j = \mathbf{lm}(p_j r_j) = \mathbf{lm}(\lambda_k' p_k) = \lambda_k' \mathbf{lm}(p_k)$ and $\tilde{l}\mathbf{lm}(p_j)r_j = \tilde{\lambda}\lambda_k' \mathbf{lm}(p_k) = \lambda_k \mathbf{lm}(p_k) = \mathbf{lm}(p_i)r_i$ for some monomials $\tilde{l}, \tilde{\lambda}$. This already implies $\tilde{l} = \tilde{\lambda}$ and $\tilde{\lambda}\lambda' = \lambda_k$. Moreover, $\tilde{l}\mathbf{lm}(p_j)r_j = \mathbf{lm}(p_i)r_i$ implies that one of the following holds:

   - $\tilde{l}\mathbf{lm}(p_j) | \mathbf{lm}(p_i)$. Then the set of polynomials is not interreduced, which leads to a contradiction.
   - There exists $\hat{r}_i$ such that $r_j = \hat{r}_i r_i$. This implies the existence of a pair $(1, p_i, \hat{r}_i; \tilde{l}, p_j, 1)$ and the claim follows from the first case. $\square$

REMARK 4.3. *One can apply these criteria in a straight forward way: If the set of critical pairs during some step of Buchberger's algorithm has been constructed, then one can* *just check the pairs and search for redundant ones. However, to decide if a monomial divides another is not as cheap and easy as in the commutative case. So, in this situation the usage of distance vectors leads to much more effective computations.*

4.2 Translation to letterplace

Our final goal now is to translate the criteria into the letterplace realm. Notably, in this criterion there is no distinction between graded and non-graded cases.

THEOREM 4.4. *Let P be the set of critical pairs. Suppose it contains a pair $\pi = (p_i, s \cdot p_k)$ for $p_i, p_k \in W \subset K[\boldsymbol{X} \mid \mathbb{N}]$ and $s \in \mathbb{N}$.*

1. *If there exist two pairs $\pi_1 = (p_i, s' \cdot p_j)$ and $\pi_2 = (p_j, s'' \cdot p_k)$, such that $\mathbf{lm}(s' \cdot p_j) | \mathbf{lcm}(p_i, s \cdot p_k)$, then the s-polynomial $s(\pi)$ of π will reduce to zero.*
2. *If there exists a pair $\pi_1 = (p_j, s \cdot p_k) \neq \pi$, such that $\mathbf{lcm}(p_j, s \cdot p_k)$ divides $\mathbf{lcm}(p_i, s \cdot p_k)$, then the s-polynomial $s(\pi)$ of π will reduce to zero.*

REMARK 4.5. *Ad* 1.*: We have $s'' = s - s'$. This follows immediately from the non-commutative proof and the form of the overlap. In the case, when the shifts are already known, commutative methods can be used to check the divisibility. For condition* 1. *this is especially easy, since from the concrete pair we check its shift is known.*
Ad 2.*: Since we assume that the shift of p_k is the same for π and π_1, the condition, that $\mathbf{cm}(\pi_1)$ divides $\mathbf{cm}(\pi)$ from the right, is always satisfied.*

5. IMPLEMENTATION AND TIMINGS

The new methods have been implemented into the kernel part of the computer algebra system SINGULAR. As mentioned before, the implementation of the letterplace structure is discussed in [LL09], so we will not discuss this any further. For an introduction to SINGULAR we refer to the online-manual [DGPS12].

We will now present some important examples and compare our timings with those given by the implementation of letterplace Gröbner bases by Viktor Levandovskyy in the current distribution of SINGULAR, as well as with the implementations in GAP and MAGMA. We must mention that the older implementation in SINGULAR [LL09] has been released for graded ideals; its functionality with non-graded ideals is experimental.

Note that the implementation of the LETTERPLACE:DVEC algorithm is not yet distributed with SINGULAR. The merge of our development branch with the main branch of SINGULAR will be done soon.

All tests were performed on a PC equipped with two Intel Core i7 Quadcore Processor (8×2933 MHz) with 16GB RAM running Linux.

We used MAGMA V2.18-12 [BCP97], GAP Version 4.5.6 [GAP13] with the package GBNP, version 1.0.1 and SINGULAR version 3-1-6.

Testing methodology. In order to make the tests reproducible, we used the new SDEVALV2 framework, created by Albert Heinle of the SYMBOLICDATA project ([BG00]) for our benchmarking. It means that the input polynomials have been out into the system SYMBOLICDATA. Then, for each computer algebra system the files to be executed were generated by the SYMBOLICDATA using scripts, written by ourselves for this purpose. With the help of SDEVALV2 the *computing task* was formed, put to the compute server, executed and evaluated. The functions of SYMBOLICDATA as well as the data are free to use. In such a way our comparison is easily and trustfully reproducible by any other person. Note, that among other the function, which is used to measure the time, can be customized within this approach.

5.1 Examples

Many of the examples are explained in detail in [LL09] or [Stu10] and we use the same notation. In the following we explain only the new ones.

One-relator quotients

In [CHN] the authors present a list of 48 examples of one relator quotients of the modular group. All these examples were considered with a degree bound by the total degree of the maximal generator. The enumeration is chosen according to the paper and the examples are denoted by H.

LS

The examples LS_5d9 and LS_6d10 were presented to us during discussions with Roberto La Scala and are connected to Clifford algebras. Infinite Gröbner bases are expected from this generating sets, therefore degree bounds are employed. The first number denotes the number of generators, while the number following the d denotes the degree bound.

Example	Sing 1	Sing 2	Magma	GAP
$2tri\_4v7d$	4.10	1.75	1.40	31.67
$3nilp\_d6$	0.41	0.29	0.96	4.76
$3nilp\_d10$	2410.15†	36.65	2.89	31.08
$4nilp\_d8$	380.23†	747.95	10.25	1133.82
$Braid3\_11$	273.40†	15.73	1.52	185.39
$Braid4\_11$	51.82	3.10	1.14	31.97
$plBraid3d\_6$	0.18	0.08	0.91	926.80
$lp1\_10$	31.31	2.33	1.00	11.10
$lv2d10$	0.23	0.15	0.78	3.29
s_e6d10	10.56	1.84	1.12	12.45
s_e6d13	976.32	44.74	7.81	274.63
$s\_eha112d10$	1.12	0.26	0.96	6.20
$s\_eha112d12$	462.36	4.19	1.40	62.40
s_f4_d10	4.35	0.58	0.97	5.35
s_f4_d15	1103.33 †	147.31	13.54	2241.62
s_ha11_d10	2.18	0.32	0.81	3.51
LS_5d9	23.46	2.49	0.79	2.90
LS_6d10	411.33 †	704.97	16.86	372.06
$C\_4\_1\_7W$	3.23	1.19	0.91	5.76
$C\_4\_1\_7Y$	0.09	0.09	0.91	2.91
H_5	0.62	0.24	0.62	2.90
H_19	0.88	0.32	0.62	2.99
H_37	0.86	0.32	0.68	2.89
H_40	0.98	0.29	0.62	2.89
H_48	0.88	0.31	0.62	2.91

5.2 Timings

In [LL09] we used external time measuring for the whole computation via `/usr/bin/time` command. This included the initializing of a computer algebra system as well as the loading of standard libraries. In this paper we use IEEE standard for measuring (POSIX.2) and present the timings for the *system* record of the time output.

In the following tables the resulting timings are presented. Sing 1 refers to the implementation by Viktor Levandovskyy, currently distributed with SINGULAR, while Sing 2 is the new implementation of the authors using distance vectors. Results are presented in seconds. By † we denote the situation when the computation run out of memory after the indicated time.
While MAGMA is still faster in some cases the timings show that the new letterplace implementation can compete with other systems and since the test were done with a first implementation there is still space for improvements.

Acknowledgements

We would like to thank Roberto La Scala for numerous discussions on the topics, connected to this article and for examples he suggested to us. The expertise and friendliness of Martin Kreuzer and Victor Ufnarovskij was a big help in our research. Special thanks go to Hans Schönemann and Olexander Motsak from the SINGULAR team, whose advices and help in technical questions greatly contributed to the software development. We are grateful to Simon King and to David Green for the motivation and inspiration via concrete problems. With the help of Albert Heinle and the package SDEVALV2 [HLN13] for SYMBOLICDATA [BG00] the testing became comfortable, reproduceable and transparent.

6. REFERENCES

[BB98] Miguel Angel Borges and Mijail Borges. Gröbner bases property on elimination ideal in the noncommutative case. In B. Buchberger and F. Winkler, editors, *Gröbner bases and applications*, pages 323–337. Cambridge University Press, 1998.

[BCP97] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. `http://dx.doi.org/10.1006/jsco.1996.0125`.

[BG00] Olaf Bachmann and Hans-Gert Gräbe. The SYMBOLICDATA project. In *Reports on Computer Algebra*, volume 27. 2000. Available from: `http://www.mathematik.uni-kl.de/~zca`.

[CHN] Marston Conder, George Havas, and M.F. Newman. On one-relator quotients of the modular group. 2011. Cambridge University Press. London Mathematical Society Lecture Note Series 387, 183-197.

[DGPS12] W. Decker, G.-M. Greuel, G. Pfister, and H. Schönemann. SINGULAR 3-1-6 — A computer algebra system for polynomial computations. 2012. Available from: `http://www.singular.uni-kl.de`.

[GAP13] The GAP Group. *GAP – Groups, Algorithms, and Programming, Version 4.6.3*, 2013. Available from: `http://www.gap-system.org`.

[GM88] Rüdiger Gebauer and H. Michael Möller. On an installation of Buchberger's algorithm. *J. Symb. Comput.*, 6(2-3):275–286, 1988. `doi:10.1016/S0747-7171(88)80048-8`.

[HLN13] Albert Heinle, Viktor Levandovskyy and Andreas Nareike. Symbolicdata:sdeval - benchmarking for everyone. 2013. submitted.

[KR00] Martin Kreuzer and Lorenzo Robbiano. *Computational commutative algebra. 1.* Springer-Verlag, Berlin, 2000.

[KR05] Martin Kreuzer and Lorenzo Robbiano. *Computational commutative algebra. 2.* Springer-Verlag, Berlin, 2005.

[KX13] Martin Kreuzer and Xingqiang Xiu. Non-commutative Gebauer-Moeller criteria. 2013. `arXiv:arXiv/1302.3805`.

[Li12] Huishi Li. *Gröbner Bases in Ring Theory.* World Scientific Publishing, 2012.

[LL09] R. La Scala and V. Levandovskyy. Letterplace ideals and non-commutative Gröbner bases. *J. Symbolic Computation*, 44(10):1374–1393, 2009. `doi:doi:10.1016/j.jsc.2009.03.002`.

[Mor88] Teo Mora. Gröbner bases in non-commutative algebras. *Proceedings of ISSAC conference, 358:150–161, 1988.* Lecture Notes in Computer Science. Springer-Verlag.

[Mor94] Teo Mora. An introduction to commutative and non-commutative Gröbner bases. *Theoretical Computer Science*, 134:131–173, 1994.

[Sca12] Roberto La Scala. Extended letterplace correspondence for nongraded noncommutative ideals and related algorithms. 2012. `arxiv.org/abs/1206.6027`.

[SL13] Roberto La Scala and Viktor Levandovskyy. Skew polynomial rings, Gröbner bases and the letterplace embedding of the free associative algebra. *Journal of Symbolic Computation*, 48:110 – 131, 2013. `doi:10.1016/j.jsc.2012.05.003`.

[Stu10] Grischa Studzinski. Algorithmic computations for factor algebras. Diploma thesis, RWTH Aachen, 2010. Available from: `http://www.math.rwth-aachen.de/~Grischa.Studzinski/DA.pdf`.

[Xiu12] Xingqiang Xiu. *Non-Commutative Gröbner Bases and Applications.* PhD thesis, University of Passau, 2012. Available from: `http://www.opus-bayern.de/uni-passau/volltexte/2012/2682/`.