



HAL
open science

Completion in Operads via Essential Syzygies

Philippe Malbos, Isaac Ren

► **To cite this version:**

Philippe Malbos, Isaac Ren. Completion in Operads via Essential Syzygies. 46th International Symposium on Symbolic and Algebraic Computation, Jul 2021, Saint Petersburg, Russia. 10.1145/3452143.3465552 . hal-03258883

HAL Id: hal-03258883

<https://hal.science/hal-03258883>

Submitted on 12 Jun 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Completion in Operads via Essential Syzygies

Philippe Malbos

Univ Lyon, Université Claude Bernard Lyon 1
CNRS UMR 5208, Institut Camille Jordan
France

Isaac Ren

École Normale Supérieure de Lyon
France

Abstract

We introduce an improved Gröbner basis completion algorithm for operads. To this end, we define operadic rewriting systems as a machinery to rewrite in operads, whose rewriting rules do not necessarily depend on an ambient monomial order. A Gröbner basis of an operadic ideal can be seen as a confluent and terminating operadic rewriting system; thus, the completion of a Gröbner basis is equivalent to the completion of a rewriting system. We improve the completion algorithm by filtering out redundant S-polynomials and testing only essential ones. Finally, we show how the notion of essential S-polynomials can be used to compute Gröbner bases for syzygy bimodules. This work is motivated by the computation of minimal models of associative algebras and symmetric operads. In this direction, we show how our completion algorithm extends to the case of shuffle operads.

CCS Concepts

• **Computing methodologies** → **Combinatorial algorithms;**
Algebraic algorithms.

Keywords

Non commutative Gröbner bases; computing with syzygies; completion algorithms; rewriting in operads

1 Introduction

The fundamental theorem of any algebraic rewriting machinery is the critical branching theorem, or Buchberger criterion, which characterizes confluence by algebraic obstructions described as *critical branchings* or *critical pairs* in rewriting theory [14, 20], and as *S-polynomials* in Gröbner basis theory [4, 19]. This theorem is the core of *completion algorithms* of a rewriting system, which resolve the obstructions in order to reach confluence. The efficiency of completion algorithms depends on several aspects: 1/ the algebraic structure on which we rewrite (strings, commutative or non-commutative polynomials, planar trees...), 2/ the strategy used to orient the rules added during completion, 3/ the choice of which critical branchings to resolve and the prediction non-essential ones, 4/ the order in which we resolve these critical branchings. For instance, Buchberger’s algorithm consists in exploring all branchings, but several methods have been introduced to improve completion by reducing the set to be explored via *syzygies* of generating polynomials, which describe relations between obstructions [9, 15].

The objective of this work is twofold. Firstly, we introduce a rewriting machinery for non-symmetric operads that generalizes the Gröbner basis approach of [3, 8] by removing the constraint of a monomial order for the orientation of the rewriting rules. Secondly, we define a completion algorithm for rewriting in an operadic context by restricting the examination of critical branchings to *essential*

ones. The essential branchings are defined by considering relations between critical branchings with respect to a monomial order on contexts. We show that the relations between these branchings correspond to syzygies of the defining rewriting rules.

This work is aimed at developing rewriting methods for calculating with operads. Operads provide an algebraic language to describe types of algebras in topology and algebra [16, 18]. They are usually defined with symmetric actions, which are incompatible with rewriting, but *shuffle operads*, introduced by Dotsenko and Khoroshkin in [7], allow for rewriting methods all while preserving the information of symmetries. In particular, these rewriting methods apply to the computation of explicit free resolutions of operads [7, 17]. The problem of finding a procedure that computes a minimal model for a given operad is still open. In the rewriting approach, this calls for a notion of minimal branchings in all dimensions of the resolution. In this article, we present our algorithm in the case of non-symmetric operads, but the shuffle case can be treated in the same way. Moreover, our notion of essential branchings can also be used to ‘go up in dimension’, and we expect to deduce an algorithm calculating minimal resolutions of operads.

The paper is structured as follows. In Section 2, we define the notions of rewriting systems and Gröbner bases for non-symmetric operads. We introduce the notion of essential branchings, and we give an algorithm to compute essential branchings with respect to a monomial order on contexts. In Section 3, we show the main result of this paper, Theorem 3.3, which characterizes the confluence of a terminating operadic rewriting system in terms of essential branchings. From this result, we deduce a completion algorithm, which transforms terminating rewriting systems into confluent ones by resolving essential branchings. In Section 4, we show that essential syzygies generate the bimodule of all syzygies of a convergent operadic rewriting system. As a conclusion, in Section 5, we give an alternative definition of essential branchings, allowing us to define essential branchings involving $n > 2$ rewriting rules. We also explain how to adapt our constructions to the cases of algebras over operads and shuffle operads.

2 Operadic rewriting systems

In this section, we introduce rewriting systems for non-symmetric operads, which we relate to the notion of Gröbner bases for operads. We define the notion of essential branchings as ‘minimal’ obstructions to local confluence, and an algorithm to compute the essential branchings of an operadic rewriting system.

Operads. A *collection* is a sequence $V = (V(n))_{n \in \mathbb{N}}$ of spaces (over a field \mathbf{k}) indexed by *arities* $n \geq 0$. We denote by $\text{AR}(x)$ the arity of an element of V . A (*non-symmetric*) *operad* is a collection P with an *identity* element ε in $P(1)$, equipped with composition maps

$$\circ : P(k) \otimes P(n_1) \otimes \dots \otimes P(n_k) \rightarrow P(n_1 + \dots + n_k)$$

restricts to a monomial order $<_{\text{cont-path-lex}}$ on monomial contexts. Fixing a total order on R , this gives a rewriting monomial order $<_{\text{rm-path-lex}}$.

Gröbner bases. Fix a monomial order $<$ on $\mathcal{T}(\Sigma)$. The *leading monomial* of a polynomial f of $\mathcal{F}(\Sigma)$ is the greatest monomial in the support of f wrt the order $<$, denoted by $\text{lm}_<(f)$. We define the associated rule $\vec{\text{lm}}_<(f) : \text{lm}_<(f) \rightarrow 1/\text{lc}_<(f)(f - \text{lm}_<(f))$, where $\text{lc}_<(f)$ denotes the coefficient of $\text{lm}_<(f)$ in the polynomial f . For a set of polynomials G of $\mathcal{F}(X)$, we denote by $\vec{\text{lm}}_<(G) := \{\vec{\text{lm}}_<(g) \mid g \in G\}$ the corresponding set of rules. We say that G is a *Gröbner basis* of an ideal I of $\mathcal{F}(\Sigma)$ if G generates I and the ORS $(X, \vec{\text{lm}}_<(G))$ is confluent, or equivalently, if every polynomial f in I reduces to 0 wrt $\vec{\text{lm}}_<(G)$.

Now, let E be a set of rewriting polynomials. Every element e of E can be written $e = \sum_i \lambda_i C_i[\alpha_i]$, where $\lambda_i \in \mathbf{k} \setminus \{0\}$, C_i is a monomial context, and $\alpha_i \in R$. We define the rule $\vec{\text{lm}}_{<_{\text{rm}}}(e) : \text{lm}_{<_{\text{rm}}}(e) \rightarrow 1/\text{lc}_{<_{\text{rm}}}(e)(e - \text{lm}_{<_{\text{rm}}}(e))$, where $\text{lc}_{<_{\text{rm}}}(e)$ denotes the coefficient of $\text{lm}_{<_{\text{rm}}}(e)$ in e . Denote by $\vec{\text{lm}}_{<_{\text{rm}}}(E)$ the set of these rules for $e \in E$. We say that E is a *Gröbner basis* of a submodule M of $\mathcal{F}(\Sigma)\langle R \rangle$ wrt $<_{\text{rm}}$ if E generates M as a $\mathcal{F}(\Sigma)$ -bimodule and every rewriting polynomial s in M reduces to 0 wrt $\vec{\text{lm}}_{<_{\text{rm}}}(E)$.

Branchings and confluence. A *branching* (resp. *local branching*) is a pair (a, b) of rewriting paths (resp. rewriting steps) such that $a \neq b$ and $s(a) = s(b)$. We classify local branchings into three types:

- i) *additive branchings*: $(\lambda a + \mu 1_v + 1_h, \lambda 1_u + \mu b + 1_h)$, where $a : u \rightarrow f, b : v \rightarrow g \in \mathcal{R}_X^m, \lambda, \mu \in \mathbf{k} \setminus \{0\}, h$ is a polynomial, $u \neq v$, and $u, v \notin \text{Supp}(h)$.
- ii) *multiplicative branchings*: $(\lambda C[a, 1_v] + 1_h, \lambda C[1_u, b] + 1_h)$, where C is a monomial two-hole context, $a : u \rightarrow f, b : v \rightarrow c \in \mathcal{R}_X^m, \lambda \in \mathbf{k} \setminus \{0\}, h$ is a polynomial, and $C[u, v] \notin \text{Supp}(h)$.
- iii) *intersecting branchings*: the rest of the local branchings.

We define a well-founded partial order \sqsubseteq on branchings by setting, for every monomial context C , polynomial h , and $\lambda \in \mathbf{k} \setminus \{0\}$,

$$(a, b) \sqsubseteq (\lambda C[a] + 1_h, \lambda C[b] + 1_h).$$

The *critical branchings* are the minimal intersecting branchings for this order.

Now, fix a rewriting monomial order $<_{\text{rm}}$ on \mathcal{R}_X^m . An *essential branching* of X is a critical branching $(C[\alpha], D[\beta])$, where $C[\alpha] <_{\text{rm}} D[\beta]$ and they are consecutive for this order, that is there does not exist a rewriting monomial $E[\gamma]$ such that $C[\alpha] <_{\text{rm}} E[\gamma] <_{\text{rm}} D[\beta]$. We denote by $\mathcal{E}(X)$ the set of essential branchings of X . Algorithm 1 computes the essential branchings of an ORS X with respect to a monomial order.

The *S-polynomial* of a branching (a, b) of X is the polynomial $S(a, b) := t(a) - t(b)$. It is called *essential* when the branching is so. A branching (a, b) is *confluent* if there exist two rewriting paths $t(a) \rightarrow h$ and $t(b) \rightarrow h$ with the same target, or equivalently, if its *S-polynomial* reduces to 0. The ORS X is *confluent* (resp. *locally confluent*) at a polynomial f if all its branchings (resp. local branchings) with source f are confluent. It is *confluent* (resp. *locally confluent*) if it is so at every polynomial, and *convergent* if it is confluent and terminating.

rec BRANCHINGSWITHROOT($\Sigma, R, <_{\text{rm}}, \alpha, \vec{v}, i_0$)

Input: An ORS $X = (\Sigma, R)$,
an order $<_{\text{rm}}$ on \mathcal{R}_X^m ,
a rewriting rule α of R of arity n ,
a list of monomials v_1, \dots, v_n in $\mathcal{T}(\Sigma)$,
 $i_0 \leq \text{AR}(v_1) + \dots + \text{AR}(v_n)$.

Output: A set of essential branchings whose first rewriting step contains $(\alpha \mid \vec{v})$, and where the first $i_0 - 1$ inputs are untouched.

```

if  $\text{SIZE}(\vec{v}) < \max_{\beta \in R} |s(\beta)|$  then
   $P \leftarrow \{D[\beta] \in \mathcal{R}_X^m \mid D[s(\beta)] = (s(\alpha) \mid \vec{v}), \text{ and } (\alpha \mid \vec{v}) <_{\text{rm}} D[\beta]\}$ ;
  if  $P \neq \emptyset$  then
    return  $\{((\alpha \mid \vec{v}), \min(P, <_{\text{rm}}))\}$ ;
  else
    return
       $\bigcup_{\substack{i_0 \leq i \\ x \in \Sigma}} \text{BRANCHINGSWITHROOT}(X, <_{\text{rm}}, \alpha, \vec{v} \circ_i x, i)$ ;
  else
    return  $\emptyset$ ;

```

ESSENTIALBRANCHINGS($\Sigma, R, <_{\text{rm}}$)

Input: An ORS $X = (\Sigma, R)$,
A rewriting monomial order $<_{\text{rm}}$ on \mathcal{R}_X^m .
Output: The set of essential branchings of (Σ, R) .
return $\bigcup_{\alpha \in R} \text{BRANCHINGSWITHROOT}(\Sigma, R, <_{\text{rm}}, \alpha, \vec{e}, 1)$;

Algorithm 1: ESSENTIALBRANCHINGS and its auxiliary recursive function, BRANCHINGSWITHROOT

Let X be a convergent ORS, $<$ a monomial order on $\mathcal{T}(\Sigma)$, and $<_{\text{rm}}$ a total order on R , and consider the associated monomial order $<_{\text{rm}}$ on rewriting monomials. A *normalization strategy* ρ associates to each polynomial f of $\mathcal{F}(X)$ a rewriting path from f to its normal form as follows. If f is a normal form, then $\rho_f := 1_f$. Otherwise, write $f = \lambda u + g$, where u is the greatest reducible monomial of f wrt $<$, $\lambda \in \mathbf{k} \setminus \{0\}$, and $u \notin \text{Supp}(g)$, and let $C[\alpha]$ be the greatest rewriting monomial wrt $<_{\text{rm}}$ that reduces u . We then set

$$\rho_f := (\lambda C[\alpha] + g) \cdot \rho_{\lambda C[t(\alpha)] + g}.$$

Since X is terminating, ρ is well-defined.

Examples.

- i) Consider the *associative operad* presented by one generator x and one rule:

$$\alpha : \begin{array}{c} 1 \quad 2 \\ \diagdown \quad \diagup \\ x \\ \diagup \quad \diagdown \\ \quad 3 \end{array} \rightarrow \begin{array}{c} \quad 2 \quad 3 \\ \diagdown \quad \diagup \\ x \\ \diagup \quad \diagdown \\ 1 \end{array}$$

also written $x \circ_1 x \rightarrow x \circ_2 x$. There is one critical branching, and so one essential branching, $(\alpha \circ_1 x, x \circ_1 \alpha)$.

- ii) Consider the associative algebra presented by $\langle x \mid \alpha : x^3 \rightarrow 0 \rangle$, seen as an operad concentrated in arity 1. The critical branchings are $(\alpha x, x \alpha)$ and $(x x \alpha, \alpha x x)$, and only the first one is essential.

iii) Consider the ORS with three generators x, y, z and the following rewriting rule:

$$(x \mid yz) \rightarrow (x \mid xx) + (y \mid yy) + (z \mid zz)$$

There are no essential branchings. However, this rule cannot be the result of an orientation by a monomial order, since there is always a monomial on the righthand side greater than $(x \mid yz)$. Note that if we orient this rule by a monomial order, we would get two essential branchings.

3 Completion using essential branchings

In this section, we show that the obstructions to the confluence of a terminating ORS can be reduced to the confluence of the essential branchings, which, as we will explain in the next section, corresponds to existence of essential syzygies. We deduce an improved completion algorithm for operadic rewriting systems.

LEMMA 3.1 ([11]). *Let X be an ORS.*

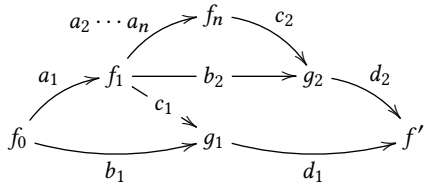
- i) *For every path a in \mathcal{R}_X , there exists a zig-zag sequence of rewriting steps from $s(a)$ to $t(a)$.*
- ii) *For every path $f_0 \xrightarrow{a_1} f_1 \xrightarrow{a_2} \dots \xrightarrow{a_n} f_n$ of length n such that X is confluent at f_i for all $i \geq 1$, there exist rewriting paths b and c such that $b \cdot c^-$ has source f_0 and target f_n .*

PROOF. Consider an edge $a = \lambda C[\alpha] + 1_g$ of \mathcal{R}_X . Write $g = \mu C[s(\alpha)] + h$, where $C[s(\alpha)] \notin \text{Supp}(h)$ and $\mu \in \mathbf{k}$. Then the zig-zag of rewriting paths

$$((\lambda + \mu)C[\alpha] + 1_h) \cdot (\mu C[\alpha] + 1_{\lambda C[t(\alpha)] + h})^-$$

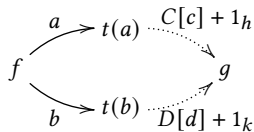
has the same source and target as a . We deduce the first point immediately.

For the second point, we proceed by induction on the length n of the path. We have proven the point for $n = 1$. For $n \geq 1$, consider the diagram

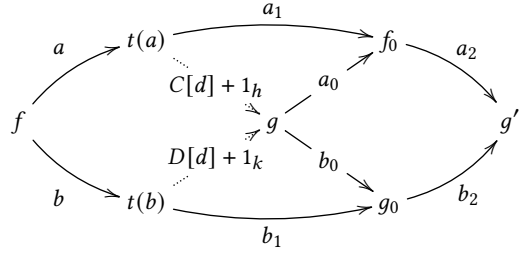


The rewriting paths b_1, c_1 are given by the first point, and the rewriting paths b_2, c_2 are given by induction hypothesis. Since we suppose X confluent at f_1 , we get the rewriting paths d_1, d_2 . Thus $(b_1 \cdot d_1) \cdot (c_2 \cdot d_2)^-$ has source f_0 and target f_n . \square

LEMMA 3.2. *Let X be an ORS, f a polynomial such that X is confluent at every polynomial $g <_R f$, and (a, b) a local branching of source f . Then (a, b) is confluent if there exist rewriting paths c, d , one-hole contexts C, D of $\mathcal{F}(\Sigma)$, and polynomials h, k in $\mathcal{F}(\Sigma)$ as in the following diagram*



PROOF. We shall construct the following confluence diagram:



Let us consider the top half of the diagram. Writing the rewriting path c as $c_1 \cdot \dots \cdot c_p$ and the context C as $\sum_{j=1}^q \lambda_j C_j$, the idea is to apply each c_i to each C_j , leaving the other monomials unchanged: for $1 \leq i \leq p$ and $1 \leq j \leq q$, we choose the edge in \mathcal{R}_X

$$c_{i,j} := \sum_{\ell=1}^{j-1} \lambda_\ell C_\ell[t(c_i)] + \lambda_j C_j[c_i] + \sum_{\ell=j+1}^q \lambda_\ell C_\ell[s(c_j)] + 1_h,$$

and write $g_{i,j} := t(c_{i,j})$ and $g_{i,0} := s(c_{i,1}) = t(c_{i-1,q})$. Then we have the path in \mathcal{R}_X

$$\begin{aligned} t(a) &= g_{1,0} \xrightarrow{c_{1,2}} \dots \xrightarrow{c_{1,q}} g_{1,q} = g_{2,0} \xrightarrow{c_{2,1}} \dots \\ &\dots \xrightarrow{c_{p-1,q}} g_{p-1,q} = g_{p,0} \xrightarrow{c_{p,1}} \dots \xrightarrow{c_{p,q}} g_{p,q} = g. \end{aligned}$$

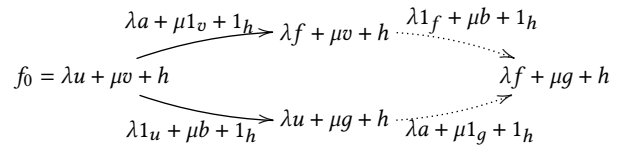
Moreover, for every i, j , $g_{i,j} <_R f$, so X is confluent at $g_{i,j}$. Following the second point of Lemma 3.1, there exist rewriting paths a_0, a_1 as in the diagram. Similarly, we show the existence of the rewriting paths b_0, b_1 . Since $g <_R f$, we can apply the confluence hypothesis to the branching (a_0, b_0) , so we get the rewriting paths a_2, b_2 . \square

THEOREM 3.3. *Let X be a terminating ORS. If every essential S -polynomial reduces to 0, then every S -polynomial reduces to 0.*

PROOF. It suffices to prove that every branching of X is confluent. Since every essential S -polynomial reduces to 0, every essential branching is confluent.

We proceed by well-founded induction on the sources of the branchings of X , with the order $<_R$, to prove that X is confluent at every polynomial of $\mathcal{F}(\Sigma)$. A reduced polynomial cannot be the source of a local branching, so X is confluent at every $f \in \text{NF}(X)$. Now, fix a nonreduced polynomial f_0 of $\mathcal{F}(\Sigma)$, and assume that X is confluent at every $g <_R f_0$. Then we proceed by case analysis on the types of local branchings.

Additive branchings. Let $(\lambda a + \mu 1_v + 1_h, \lambda 1_u + \mu b + 1_h)$ be an additive branching of source f_0 , where $a : u \rightarrow f, b : v \rightarrow g \in \mathcal{R}_X^m$, $\lambda, \mu \in \mathbf{k} \setminus \{0\}$, and $h \in \mathcal{F}(\Sigma)$, with $u \neq v$ and $u, v \notin \text{Supp}(h)$. We have the following diagram:



The dotted arrows are rewriting paths in context, so by Lemma 3.2, this local branching is confluent.

Multiplicative branchings. Let $(\lambda C[a, 1_v] + 1_h, \lambda C[1_u, b] + 1_h)$ be a multiplicative branching of source f_0 , where C is a monomial two-hole context, $a : u \rightarrow f, b : v \rightarrow g \in \mathcal{R}_X^m$, $\lambda \in \mathbf{k} \setminus \{0\}$, $h \in \mathcal{F}(\Sigma)$, and $C[u, v] \notin \text{Supp}(h)$. We have the following diagram:

$$\begin{array}{ccc}
 & \lambda C[a, 1_v] + 1_h & \lambda C[1_f, b] + 1_h \\
 & \nearrow & \dashrightarrow \\
 f_0 = \lambda C[u, v] + h & \lambda C[f, v] + h & \lambda C[f, g] + h \\
 & \searrow & \dashrightarrow \\
 & \lambda C[1_u, b] + 1_h & \lambda C[a, 1_g] + 1_h
 \end{array}$$

The dotted arrows are rewriting paths in context, so by Lemma 3.2, this local branching is confluent.

Critical branchings. Let $(C[\alpha], D[\beta])$ be a critical branching of source f_0 . If the branching is essential, then it is confluent by hypothesis. Otherwise, there exists a rewriting monomial $E[\gamma]$ such that $C[\alpha] <_{\text{rm}} E[\gamma] <_{\text{rm}} D[\beta]$, and we get two branchings $(C[\alpha], E[\gamma])$ and $(E[\gamma], D[\beta])$. The branching $(C[\alpha], E[\gamma])$ is either multiplicative or intersecting. If it is multiplicative, then it is confluent by the multiplicative case. Otherwise, it is either non-minimal for \sqsubseteq or a critical branching. In the non-minimal case, there exists a factorisation

$$\begin{array}{ccccc}
 & C_1 C_0[\alpha] & \longrightarrow & C_1 C_0[t(\alpha)] & \longrightarrow & C_1[a_0] \\
 C_1 C_0[s(\alpha)] = C_1 E_0[s(\gamma)] & \searrow & & \searrow & & \searrow \\
 & C_1 E_0[\gamma] & \longrightarrow & C_1 E_0[t(\gamma)] & \longrightarrow & C_1[b_0] \\
 & & & & & \searrow \\
 & & & & & C_1[h_0]
 \end{array}$$

where $(C_0[\alpha], E_0[\gamma])$ is a critical branching and C_1 is a monomial context. In the critical case, the branching is either essential, or not. If it is essential, then it is confluent by hypothesis. Otherwise, we proceed by induction on the number of rewriting monomials $F[\delta]$ such that $C[\alpha] <_{\text{rm}} F[\delta] <_{\text{rm}} E[\gamma]$.

We proceed similarly for the branching $(E[\gamma], D[\beta])$: in every case, we can write

- the branching $(C[\alpha], E[\gamma])$ as $(C_1 C_0[\alpha], C_1 E_0[\gamma])$, with C_1 a monomial context and $(C_0[\alpha], E_0[\gamma])$ a confluent branching,
- the branching $(E[\gamma], D[\beta])$ as $(D_1 E'_0[\gamma], D_1 D_0[\beta])$, with D_1 a monomial context and $(E'_0[\gamma], D_0[\beta])$ a confluent branching.

We can then construct the confluent diagram

$$\begin{array}{ccccccc}
 & & & C_1[a_0] & & & \\
 & & & \nearrow & & & \\
 C[\alpha] & \longrightarrow & C_1 C_0[t(\alpha)] & & C_1[b_0] & \longrightarrow & C_1[h_0] \\
 & \searrow & & & & & \searrow \\
 C[s(\alpha)] & \longrightarrow & E[\gamma] & \longrightarrow & E[t(\gamma)] & & h \\
 & \searrow & & & & & \searrow \\
 D[\beta] & \longrightarrow & D_1 D_0[t(\beta)] & & D_1[a'_0] & \longrightarrow & D_1[h'_0] \\
 & & & & D_1[b'_0] & \longrightarrow & c'_1
 \end{array}$$

where the left squares are the aforementioned confluent branchings in context and the right square is given by induction hypothesis.

Non-critical intersecting branchings. Finally, let $(\lambda C[\alpha] + 1_h, \lambda D[\beta] + 1_h)$ be an intersecting branching of source f_0 that is not critical, where $\alpha : u \rightarrow f, \beta : v \rightarrow g \in R$, C, D are monomial contexts, $\lambda \in \mathbf{k} \setminus \{0\}$, and $h \in \mathcal{F}(\Sigma)$, such that $C[u] \notin \text{Supp}(h)$. Let C_0, C_1, D_0 be the monomial contexts such that $C_1 C_0 = C$, $C_1 D_0 = D$, and $(C_0[\alpha], D_0[\beta])$ is a critical branching. By the previous case, we have the confluence diagram

$$\begin{array}{ccc}
 & C_0[\alpha] & \longrightarrow & C_0[f] & \xrightarrow{a'_0} & h_0 \\
 C_0[u] = D_0[v] & \searrow & & \searrow & & \searrow \\
 & D_0[\beta] & \longrightarrow & D_0[g] & \xrightarrow{b'_0} & h_0
 \end{array}$$

Applying the context λC_1 and adding h , we get the following diagram

$$\begin{array}{ccc}
 & \lambda C[\alpha] + 1_h & \longrightarrow & \lambda C[f] + h & \dashrightarrow & \lambda C_1[a'_0] + 1_h \\
 \lambda C[u] + h = \lambda D[v] + h & \searrow & & \searrow & & \searrow \\
 & \lambda D[\beta] + 1_h & \longrightarrow & \lambda D[g] + h & \dashrightarrow & \lambda C_1[b'_0] + 1_h
 \end{array}$$

Applying Lemma 3.2, we conclude that the local branching is confluent.

Thus we have shown that the ORS X is locally confluent at f_0 . To conclude the induction step, we show that X is confluent at f_0 by the diamond lemma. Let (a, b) be a non-local branching of source f_0 , and write $a = a_0 \cdot a_1, b = b_0 \cdot b_1$, where a_0, b_0 are rewriting steps and a_1, b_1 are rewriting paths. We construct the confluence diagram

$$\begin{array}{ccccccc}
 & & & a_1 & & & \\
 & & & \nearrow & & & \\
 a_0 & \longrightarrow & g_0 & \xrightarrow{a'_0} & f'_0 & \xrightarrow{a_2} & g_1 \\
 & \searrow & & & & & \searrow \\
 f_0 & \longrightarrow & h_0 & \xrightarrow{b'_0} & f'_0 & \xrightarrow{c} & g_2 \\
 & \searrow & & & & & \searrow \\
 & & & & & & k \\
 & & & & & & \nearrow \\
 & & & & & & b_2
 \end{array}$$

as follows. By the previous arguments, the local branching (a_0, b_0) is confluent, yielding the rewriting paths a'_0, b'_0 . Since both $f_0 > g_0$ and $f_0 > h_0$, the induction hypothesis applies to the branching (a_1, a'_0) to give the rewriting paths a_2, c , and then to the branching $(b'_0 \cdot c, b_1)$ to give the rewriting paths d, b_2 . \square

As a consequence of Theorem 3.3 and [6, Thm. 1], we have the following result.

COROLLARY 3.4. *Let I be an ideal of $\mathcal{F}(\Sigma)$ and (Σ, R) a convergent ORS such that R is compatible with a monomial order and $I(R) = I$. If the set of essential S -polynomials reduces to 0, then ∂R is a Gröbner basis of I .*

Completion algorithm. From Theorem 3.3, we deduce a completion algorithm COMPLETE, Algorithm 2, which transforms a terminating ORS into a convergent one presenting the same operad. The algorithm only tests essential S -polynomials, using the algorithm ESSENTIALBRANCHINGS defined above. The essential branchings are computed with respect to a monomial order on rewriting monomials $<_{\text{rm}}$ given by the function REWRITINGMONOMIALORDER($<^2, \dashv$),

which takes as arguments a monomial order $<^2$ on $\mathcal{T}(\Sigma)$ and a total order \dashv on R , and returns the monomial order on rewriting monomials induced by $<^2$, \dashv , and the path-lexicographic order on contexts.

COMPLETE($\Sigma, R, <^1, <^2, \dashv$)

Input: An ORS (Σ, R) compatible with $<^1$, monomial orders $<^1, <^2$ on $\mathcal{T}(\Sigma)$, and a total order \dashv on R .

Output: A convergent ORS (Σ, T) such that $I(\Sigma, R) \simeq I(\Sigma, T)$.

$T \leftarrow R$;

$<_{\text{rm}} \leftarrow \text{REWRITINGMONOMIALORDER}(<^2, \dashv)$;

$\mathcal{B} \leftarrow \text{ESSENTIALBRANCHINGS}(\Sigma, T, <_{\text{rm}})$;

while $\mathcal{B} \neq \emptyset$ **do**

 Choose (a, b) in \mathcal{B} ;

$\mathcal{B} \leftarrow \mathcal{B} \setminus \{(a, b)\}$;

$u \leftarrow \text{NF}_{\rho, <^1}(t(f), T)$;

$v \leftarrow \text{NF}_{\rho, <^1}(t(g), T)$;

$\gamma \leftarrow \overline{\text{lm}}_{<^1}(u - v)$;

if $\gamma \neq 0$ **then**

$\mathcal{B} \leftarrow$

 UPDATEESSENTIALBRANCHINGS($\Sigma, T, \mathcal{B}, <_{\text{rm}}, \gamma$);

$T \leftarrow T \cup \{\gamma\}$;

return (Σ, T) ;

Algorithm 2: The completion algorithm COMPLETE

Note that we extend and update during the total order on R to a total order on the rules of T . Indeed, the sources of the new rules γ never coincide the other rules in T . The algorithm COMPLETE requires Algorithm 3, which updates the essential branchings during completion.

Algorithm 2 adds to the set of rules R the rules needed to resolve the essential critical branchings. By construction, the added rules do not change the presented operad. At the output of the algorithm, we get a convergent set of rules T . Termination follows from the fact that all added rules are oriented with respect the monomial order $<^1$. Note that T may be infinite if the completion algorithm loops indefinitely.

This algorithm improves Buchberger's completion algorithm [4] when there are non-essential branchings. However, when the algorithm starts with an ORS whose rules' sources are all of size two and are pairwise distinct, then every branching is essential.

Example. Consider the ORS

$$\left\langle x_2, y_1, z_1 \left| \begin{array}{l} \alpha : (x \mid y y 1) \rightarrow (x \mid z z 1), \\ \beta : (x \mid y 1 y) \rightarrow (x \mid z 1 z), \\ \gamma : (x \mid 1 y y) \rightarrow (x \mid 1 z z) \end{array} \right. \right\rangle$$

There are three critical branchings: $(\alpha \circ_3 y, \beta \circ_2 y)$, $(\alpha \circ_3 y, \gamma \circ_1 y)$, $(\beta \circ_2 y, \gamma \circ_1 y)$. The contexts involved, in path-lexicographic order, are $\square_3 \circ_3 y < \square_3 \circ_2 y < \square_3 \circ_1 y$. Thus COMPLETE will execute as follows:

- The initial set of essential branchings is $\mathcal{B} = \{(\alpha \circ_3 y, \beta \circ_2 y), (\alpha \circ_3 y, \gamma \circ_1 y)\}$.

rec UPDATEBRANCHINGSWITHROOT($\Sigma, T, \mathcal{B}, <_{\text{rm}}, \alpha, \gamma, \vec{v}, i_0$)

Input: An ORS $X = (\Sigma, T)$,

 an order $<_{\text{rm}}$ on \mathcal{R}_X^m ,

 the set \mathcal{B} of essential branchings of (Σ, T) ,

 a rewriting rule α of R of arity n ,

 a rewriting rule γ ,

 a list of monomials v_1, \dots, v_n ,

$i_0 \leq \text{AR}(v_1) + \dots + \text{AR}(v_n)$.

Output: An updated set of essential branchings whose first rewriting step contains $(\alpha \mid \vec{v})$, and where the first $i_0 - 1$ inputs are untouched.

if $\text{SIZE}(\vec{v}) < \max_{\beta \in R} |s(\beta)|$ **then**

if there exists $((\alpha \mid \vec{v}), D[\beta]) \in \mathcal{B}$ **then**

$Q \leftarrow \{E[\gamma] \in \mathcal{R}_X^m$

$\mid (\alpha \mid \vec{v}) <_{\text{rm}} E[\gamma] <_{\text{rm}} D[\beta]\} \cup \{D[\beta]\}$;

return $\{((\alpha \mid \vec{v}), \min(Q, <_{\text{rm}}))\}$;

else

$Q \leftarrow \{E[\gamma] \in \mathcal{R}_X^m$

$\mid E[s(\alpha)] = (s(\alpha) \mid \vec{v}) \text{ and } (\alpha \mid \vec{v}) <_{\text{rm}} E[\gamma]\}$;

if $Q \neq \emptyset$ **then**

return $\{((\alpha \mid \vec{v}), \min(Q, <_{\text{rm}}))\}$;

else

return

$\bigcup_{\substack{i_0 \leq i \\ x \in \Sigma}} \text{UPDATEBRANCHINGSWITHROOT}(\Sigma, T, \mathcal{B},$

$<_{\text{rm}}, \alpha, \gamma, \vec{v} \circ_i x, i)$;

else

return \emptyset ;

UPDATEESSENTIALBRANCHINGS($\Sigma, T, \mathcal{B}, <_{\text{rm}}, \gamma$)

Input: An ORS $X = (\Sigma, R)$,

 A rewriting monomial order $<_{\text{rm}}$ on \mathcal{R}_X^m .

Output: The set of essential branchings of X .

return

$\bigcup_{\alpha \in R} \text{UPDATEBRANCHINGSWITHROOT}(\Sigma, T, \mathcal{B}, <_{\text{rm}}, \alpha, \gamma, \vec{e}, 1)$;

Algorithm 3: UPDATEESSENTIALBRANCHINGS and its auxiliary recursive function, UPDATEBRANCHINGSWITHROOT

- Consider the first essential branching $(\alpha \circ_3 y, \beta \circ_2 y)$: the targets of these rewriting monomials are already in normal form, so we add $\delta : (x \mid z y z) \rightarrow (x \mid z z y)$.
- No new branchings are created, so \mathcal{B} remains unchanged.
- There remains one essential branching $(\alpha \circ_3 y, \gamma \circ_1 y)$. Similarly, the targets of the two rewriting monomials are already in normal form, so we add $\zeta : (x \mid y z z) \rightarrow (x \mid z z y)$.
- There are no new essential branchings. \mathcal{B} is empty, so we return the convergent ORS $(\{x, y, z\}, \{\alpha, \beta, \gamma, \delta, \zeta\})$.

4 Gröbner bases for syzygies

In this section, we show that essential and multiplicative syzygies generate the bimodule of syzygies of a convergent ORS X . In particular, when the rules of X are compatible with a monomial order, we show how to construct a Gröbner basis of this bimodule with respect a fixed normalization strategy.

Syzygies as loops. Let X be an ORS. A *loop* of X is a zig-zag $a = a_1^{\varepsilon_1} \cdot \dots \cdot a_n^{\varepsilon_n}$ of rewriting steps, where $a_i = \lambda_i C_i[\alpha_i] + 1_{g_i}$ and $\varepsilon_i = \pm 1$, such that $s(a_1^{\varepsilon_1}) = t(a_n^{\varepsilon_n})$. To such a loop, we associate the syzygy $\sum_{i=1}^n \varepsilon_i \lambda_i C_i[\alpha_i]$. Conversely, for a syzygy $\sum_{i=1}^n \lambda_i C_i[\alpha_i]$, consider the following path in the graph \mathcal{R}_X :

$$\begin{aligned} a &= (\lambda_1 \alpha_1 + \lambda_2 C[s(\alpha_2)] + \dots + \lambda_n C[s(\alpha_n)]) \\ &\quad \cdot (\lambda_1 C[t(\alpha_1)] + \lambda_2 \alpha_2 + \dots + \lambda_n C[s(\alpha_n)]) \\ &\quad \dots \quad \cdot (\lambda_1 C[t(\alpha_1)] + \lambda_2 C[t(\alpha_2)] + \dots + \lambda_n \alpha_n) \end{aligned}$$

We have $s(a) = t(a) = C[s(\alpha_1)] + \dots + C[s(\alpha_n)]$. By the first point of Lemma 3.1, there exists a zig-zag sequence of rewriting steps that loops on $s(a) = t(a)$. In particular, a confluence diagram can be seen as a loop, and so it is associated to a syzygy.

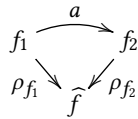
Given a normalization strategy ρ , for each essential branching (f, g) , let $(f \cdot \rho_f, g \cdot \rho_g)$ be the confluence given by ρ , and take the corresponding *essential syzygy*. For each multiplicative branching $(C[\alpha, s(\beta)], C[s(\alpha), \beta])$, take the *multiplicative syzygy* $C[\alpha, \partial(\beta)] - C[\partial(\alpha), \beta]$. Denote by $\text{Syz}(X, \rho)$ the set of essential and multiplicative syzygies. This notion of syzygy allows us to reformulate Theorem 3.3 as follows:

PROPOSITION 4.1. *Let X be a terminating ORS and let ρ be a normalisation strategy. For every branching of X , there is a confluence diagram such that the associated syzygy is generated by $\text{Syz}(X, \rho)$.*

The proof of this result is obtained following the arguments of the proof of Theorem 3.3 and keeping track of the syzygies corresponding to each confluent diagram.

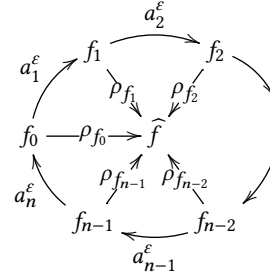
PROPOSITION 4.2. *Let X be a convergent ORS and ρ a normalization strategy. The set $\text{Syz}(X, \rho)$ generates the $\mathcal{F}(X)$ -bimodule of syzygies $\mathcal{S}(X)$.*

PROOF. Let a be a rewriting step in \mathcal{R}_X^+ . By Proposition 4.1, the associated syzygy of the confluence diagram of the branching $(\rho_{f_1}, a \cdot \rho_{f_2})$:



is generated by $\text{Syz}(X, \rho)$. Now consider a syzygy \mathfrak{s} of X . As shown above, it corresponds to $\sum_{i=1}^n \varepsilon_i \partial(a_i)$, where $a_i \in \mathcal{R}_X^+$ and $\varepsilon_i = \pm 1$ such that $a_1^{\varepsilon_1} \cdot a_2^{\varepsilon_2} \cdot \dots \cdot a_n^{\varepsilon_n}$ is a loop in \mathcal{G}_X . This loop can be drawn

as a circular pie



where \widehat{f} is the unique normal form of the polynomials f_i . Each 'slice' of the pie defines a syzygy generated by $\text{Syz}(X, \rho)$. Combining these syzygies, we conclude that the syzygy \mathfrak{s} is generated by $\text{Syz}(X, \rho)$. \square

Considering the operad presented by X , we compute syzygies modulo the multiplicative syzygies, and we get the following result.

COROLLARY 4.3. *Let X be a convergent ORS, ρ a normalisation strategy, and \overline{X} the operad presented by X . The set of essential syzygies generates the \overline{X} -bimodule of syzygies $\overline{\mathcal{S}}(X)$.*

Let us prove the main result of this section, which corresponds to a non-commutative version of Schreyer's syzygy theorem [21], already apparent in Janet's approach [13].

THEOREM 4.4. *Let X be a convergent ORS compatible with a monomial order $<$ and ρ a normalisation strategy. The set $\text{Syz}(X, \rho)$ forms a Gröbner basis of the $\mathcal{F}(X)$ -bimodule $\mathcal{S}(X)$ with respect to $<_{rm}$.*

PROOF. Let $\mathfrak{s} = \sum_{i=1}^n \lambda_i C_i[\alpha_i]$ be a syzygy in $\mathcal{S}(X)$, where $\lambda_i \in \mathbf{k} \setminus \{0\}$, $C_i \in \mathcal{R}_X^m$ and $\alpha_i \in R$. Reduce \mathfrak{s} to a normal form wrt $\text{Syz}(X, \rho)$ and the order $<_{rm}$, which we write $\mathfrak{t} = \sum_{j=1}^p \mu_j D_j[\beta_j]$, with $\mu_j \in \mathbf{k} \setminus \{0\}$, $D_j \in \mathcal{R}_X^m$ and $\beta_j \in R$. To show that $\text{Syz}(X, \rho)$ forms a Gröbner basis of $\mathcal{S}(X)$, it suffices to show $\mathfrak{t} = 0$.

For all $1 \leq j < j' \leq p$, $D_j[s(\beta_{j'})] \neq D_{j'}[s(\beta_j)]$. Otherwise, this would give a local branching of the form $(D_j[\beta_j], D_{j'}[\beta_{j'}])$. By convergence of X , this would give a syzygy whose leading term is either $D_j[\beta_j]$ or $D_{j'}[\beta_{j'}]$. By Proposition 4.2, the set $\text{Syz}(X, \rho)$ generates all syzygies, so in particular, the previous syzygy could be reduced by an element of $\text{Syz}(X, \rho)$. This would contradict the fact that \mathfrak{t} is a normal form.

As a consequence, the monomials $D_j[s(\beta_j)]$ in \mathfrak{t} cannot cancel each other out. Moreover, by definition of syzygies,

$$\sum_{j=1}^p \mu_j D_j[\partial(\beta_j)] = \sum_{i=1}^n \lambda_i C_i[\partial(\alpha_i)] = 0.$$

The only possibility is $p = 0$, so $\mathfrak{t} = 0$. \square

5 Further developments

In this article, we have introduced a notion of rewriting systems for non-symmetric operads whose rules do not depend on a monomial order. We have shown how to optimize a completion procedure that given a terminating ORS produces a convergent one by resolving only essential S-polynomials. We also have proved how to compute a Gröbner basis of the bimodule of syzygies using essential syzygies. Several improvements of these constructions and results are

possible. In particular, some variations are possible on the notion of essential S -polynomials. Moreover, we can extend our completion algorithm to algebras over operads and shuffle operads.

Essential branchings: continuation. One of the objectives of the rewriting approach developed in this work is to compute explicit free resolutions, and more generally cofibrant replacements for associative algebras [1, 2, 11] and operads [7, 17]. In this direction, it is necessary to extend the definition of essential branchings to higher dimensions. To do so, we consider an alternate definition of essential branchings based on the notion of *crowns*.

Let $X = (\Sigma, R)$ be an ORS. A *reducible divisor* of $u \in \mathcal{T}(\Sigma)$ is a minimal non-reduced monomial v such that $v \subseteq u$. Denote by $D(u)$ the set of reducible divisors of u . When the composition $(u \mid \vec{v})$ is defined, we say that \vec{v}

- i) *creates reducible divisors* if $D(u \mid \vec{v}) \setminus D(u)$ is nonempty,
- ii) *creates reducible divisors context-minimally* if, in addition, for all submonomials w_i of v_i and monomials $\vec{w}' \neq \vec{v}$ such that $(u \mid \vec{v}) = ((u \mid \vec{w}') \mid \vec{w}')$, $D(u \mid \vec{w}') = D(u)$,
- iii) *is a crown on u* if the monomials in \vec{v} are reduced and create reducible divisors context-minimally.

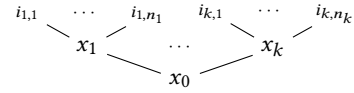
By definition, essential branchings correspond to critical branchings of the form $((\alpha \mid \vec{v}), D[\beta])$ where \vec{v} is a crown on $s(\alpha)$. In this way, essential branchings can be generalized to a notion of *essential overlappings* involving n rewriting rules as done in [17], where the authors show that higher dimensional essential overlappings generate higher dimensional syzygies on the presented operad \bar{X} , providing a resolution of the operad presented by the ORS X , and thus extending Corollary 4.3 to higher dimensions.

Algebras over operads. Our approach can be adapted to rewriting systems for algebras over non-symmetric operads. Indeed, an algebra A over a non-symmetric operad P can be represented by the polynomials of arity 0 of an operad whose generators are those of P and A , the latter being of arity 0, satisfying the relations of the operad and those of the algebra [8]. Gröbner bases have been developed for algebras over certain operads, including the commutative operad and associative operad for usual algebras, as well as the Lie operad [5] and the free operad on a binary generator, which defines non-associative non-commutative algebras [10]. We expect that our approach is a first step towards a unified definition of critical branchings for rewriting systems in algebraic structures.

The case of shuffle operads. In this article, for simplicity's sake, our algorithms are written for non-symmetric operads. However, it would work for other types of operads, such as shuffle operads and colored versions of non-symmetric and shuffle operads. *Shuffle operads*, which encode symmetric actions explicitly, were introduced by Dotsenko and Khoroshkin in [6] to study homological properties of linear symmetric operads by a rewriting approach. They introduced a notion of Gröbner bases for shuffle operads with respect to a total order on tree monomials and a completion algorithm that extends Buchberger's completion algorithm.

A *free shuffle operad* consists of the same monomials as a non-symmetric operad, except that we allow certain permutations of

inputs: a shuffle monomial is of the form



where the inputs are permuted such that $i_{j,1} < \dots < i_{j,n_j}$ for all $1 \leq j \leq k$ and $i_{1,1} < \dots < i_{k,1}$. Our constructions and algorithms on non-symmetric operads can be extended to shuffle operads: in particular, we have an analogous notion of essential branchings. The key modification consists in the definition of the set of monomials $\mathcal{T}(\Sigma)$ by taking into account the shuffle composition: $(x_0 \mid f \ x_1 \dots x_k)$, where f is the surjection associated to the permutation on the inputs. The shuffle composition extends to contexts and the confluence properties work as well. Theorems 3.3 and 4.4 still apply, and the algorithms ESSENTIALBRANCHINGS and COMPLETE can be adapted to the shuffle case.

References

- [1] David J. Anick. On the homology of associative algebras. *Trans. Amer. Math. Soc.*, 296(2):641–659, 1986.
- [2] David J. Anick and Edward L. Green. On the homology of quotients of path algebras. *Comm. Algebra*, 15(1-2):309–341, 1987.
- [3] Murray R. Bremner and Vladimir Dotsenko. *Algebraic operads*. CRC Press, Boca Raton, FL, 2016. An algorithmic companion.
- [4] Bruno Buchberger. *An Algorithm for Finding the Basis Elements in the Residue Class Ring Modulo a Zero Dimensional Polynomial Ideal*. PhD thesis, University of Innsbruck, 1965. *J. Symbolic Comput.* Vol. 41 (3-4): 475–511, 2006.
- [5] W. A. de Graaf and J. Wisliceny. Constructing bases of finitely presented lie algebras using gröbner bases in free algebras. In *Proceedings of the 1999 International Symposium on Symbolic and Algebraic Computation, ISSAC '99*, page 37–43, New York, NY, USA, 1999. Association for Computing Machinery.
- [6] Vladimir Dotsenko and Anton Khoroshkin. Gröbner bases for operads. *Duke Math. J.*, 153(2):363–396, 2010.
- [7] Vladimir Dotsenko and Anton Khoroshkin. Quillen homology for operads via Gröbner bases. *Doc. Math.*, 18:707–747, 2013.
- [8] Vladimir Dotsenko and Bruno Vallette. Higher Koszul duality for associative algebras. *Glasg. Math. J.*, 55(A):55–74, 2013.
- [9] Rüdiger Gebauer and H. Michael Möller. On an installation of Buchberger's algorithm. *J. Symbolic Comput.*, 6(2-3):275–286, 1988. Computational aspects of commutative algebra.
- [10] Lothar Gerritzen. Tree polynomials and non-associative Gröbner bases. *J. Symbolic Comput.*, 41(3-4):297–316, 2006.
- [11] Yves Guiraud, Eric Hoffbeck, and Philippe Malbos. Convergent presentations and polygraphic resolutions of associative algebras. *Math. Z.*, 293(1-2):113–179, 2019.
- [12] Eric Hoffbeck. A Poincaré-Birkhoff-Witt criterion for Koszul operads. *Manuscripta Math.*, 131(1-2):87–110, 2010.
- [13] Maurice Janet. Sur les systèmes d'équations aux dérivées partielles. *Journal de mathématiques pures et appliquées*, 8(3):65–151, 1920.
- [14] Donald Knuth and Peter Bendix. Simple word problems in universal algebras. In *Computational Problems in Abstract Algebra (Proc. Conf., Oxford, 1967)*, pages 263–297. Pergamon, Oxford, 1970.
- [15] Viktor Levandovskyy, Tobias Metzloff, and Karim Abou Zeid. Computation of free non-commutative Gröbner Bases over \mathbb{Z} with Singular:Letterplace. In *ISSAC 2020 - International Symposium on Symbolic and Algebraic Computation*, Kalamata, Greece, July 2020.
- [16] Jean-Louis Loday. La renaissance des opérades. *Astérisque*, 237:Exp. No. 792, 3, 47–74, 1996. Séminaire Bourbaki, Vol. 1994/95.
- [17] Philippe Malbos and Isaac Ren. Shuffle polygraphic resolutions for operads. preprint, arXiv:2012.15718, December 2020.
- [18] J. P. May. *The geometry of iterated loop spaces*. Springer-Verlag, Berlin-New York, 1972. Lectures Notes in Mathematics, Vol. 271.
- [19] Teo Mora. An introduction to commutative and noncommutative Gröbner bases. *Theoret. Comput. Sci.*, 134(1):131–173, 1994. Second International Colloquium on Words, Languages and Combinatorics (Kyoto, 1992).
- [20] Maurice Nivat. Congruences parfaites et quasi-parfaites. In *Séminaire P. Dubreil, 25e année (1971/72), Algèbre, Fasc. 1, Exp. No. 7*, page 9. Secrétariat Mathématique, Paris, 1973.
- [21] Frank-Olaf Schreyer. Die berechnung von syzygien mit dem verallgemeinerten weierstrass'schen divisionssatz. Diplom Thesis, University of Hamburg, Germany, 1980, 1980.