Algorithms for Garside calculus

Patrick Dehornoy[a,1], Volker Gebhardt[b,1,2]

[a] *Laboratoire de Mathématiques Nicolas Oresme, CNRS UMR 6139, Université de Caen, 14032 Caen, France*
[b] *University of Western Sydney, Centre for Research in Mathematics, Locked bag 1797, Penrith, NSW 2751, Australia*

Abstract

Garside calculus is the common mechanism that underlies a certain type of normal form for the elements of a monoid, a group, or a category. Originating from Garside's approach to Artin's braid groups, it has been extended to more and more general contexts, the latest one being that of categories and what are called Garside families. One of the benefits of this theory is to lead to algorithms solving effectively the naturally occurring problems, typically the Word Problem. The aim of this paper is to present and solve these algorithmic questions in the new extended framework.

Keywords: normal form, word problem, category presentation, groupoid of fractions, Garside calculus, Garside family, germ
2000 MSC: 20F10, 18B40, 20F36, 68Q17

F.A. Garside (1969) solved the Word and Conjugacy Problems in Artin's braid group B_n (Artin, 1947) by describing the latter as a group of fractions and analyzing the involved monoid in terms of its divisibility relation. This approach was continued and extended in several steps, first to Artin-Tits groups of spherical type (Brieskorn and Saito, 1972; Deligne, 1972; Adyan, 1984; Thurston, 1988; Charney, 1992; Epstein et al., 1992; El-Rifai and Morton, 1994), then to a larger family of groups now known as Garside groups (Dehornoy and Paris, 1999; Dehornoy, 2002). More recently, it was realized that going to a categorical context allows for capturing further examples (Krammer, 2008; Bessis, 2006b,a; Digne and Michel, 2006), and a coherent theory has now emerged with a central unifying notion called Garside families (Dehornoy et al., 2013a,b): the central notion is a certain way of decomposing the elements of the reference category or its groupoid of fractions and a Garside family is what makes the construction possible.

What we do in this paper is to present and analyze the main algorithms arising in this new, extended context of Garside families, with two main directions, namely recognizing that a candidate family is a Garside family and using a Garside family to compute in the category, typically finding distinguished decompositions and solving the Word Problem along the lines of Dehornoy (2008). This results in a corpus of about twenty algorithms that are proved to be correct, analyzed, and given examples. We do not address the Conjugacy Problem here, as extending the

Email addresses: `patrick.dehornoy@unicaen.fr` (Patrick Dehornoy), `v.gebhardt@uws.edu.au` (Volker Gebhardt)

URL: `http://www.math.unicaen.fr/~dehornoy` (Patrick Dehornoy), `http://www.uws.edu.au/crm` (Volker Gebhardt)

[1] Both authors acknowledge support under Australian Research Council's Discovery Projects funding scheme (project number DP1094072).

[2] Volker Gebhardt acknowledges support under the Spanish Project MTM2010-19355.

methods of Gebhardt (2005) and Gebhardt and González-Meneses (2010) will require further developments that we keep for a subsequent work.

The paper consists of six sections. The preliminary Section 0 explains the intuitive objective of Garside theory, namely to obtain distinguished expressions for elements of an algebraic structure, and it presents some examples that motivate (and justify) the extension of its techniques to a category theoretic setting. Section 1 is a review of Garside families and the derived notions involved in the approach, together with some basic results that appear in other sources. Next, we address the question of effectively recognizing Garside families and we describe and analyze algorithms doing it: in Section 2, we consider the case when the ambient category is specified using a presentation (of a certain type), whereas, in Section 3, we consider the alternative approach when the category is specified using what is called a germ. Finally, the last two sections are devoted to those computations that can be developed once a Garside family is given. In Section 4, we consider computations taking place in the reference category or monoid ("positive case"), whereas, in Section 5, we address similar questions in the groupoid or group of fractions of the reference category ("signed case").

0. Motivation

Consider the free Abelian monoid $\mathbb{N}^n$. We can define a partial order on $\mathbb{N}^n$ by saying that $(f_1, \dots, f_n) \preccurlyeq (g_1, \dots, g_n)$ holds if we have $f_i \leqslant g_i$ for $i = 1, \dots, n$. It is obvious that $\preccurlyeq$ is a lattice order, that is, that any two elements $(f_1, \dots, f_n)$ and $(g_1, \dots, g_n)$ of $\mathbb{N}^n$ have a unique least common upper bound and a unique greatest common lower bound with respect to $\preccurlyeq$: the former is given by $(\max\{f_1, g_1\}, \dots, \max\{f_n, g_n\})$, while the latter is given by $(\min\{f_1, g_1\}, \dots, \min\{f_n, g_n\})$.

Define $\Delta = (1, \dots, 1) \in \mathbb{N}^n$. Clearly, the set $\mathcal{S} = \{f \in \mathbb{N}^n \mid f \preccurlyeq \Delta\}$ generates the monoid $\mathbb{N}^n$. In fact, every element of $\mathbb{N}^n$ has a *unique* representation as a product of the form $g_1 g_2 \cdots g_k$ such that $g_1, \dots, g_k$ are nontrivial elements of $\mathcal{S}$ and $1 \neq f \preccurlyeq g_i$ implies $g_{i-1} f \notin \mathcal{S}$ for $i = 2, \dots, k$ and arbitrary f. The latter condition is a maximality statement: it says that, in the sequence $g_1, \dots, g_k$, it is not possible to extract a nontrivial fragment f from the entry g_i and incorporate it in the previous entry g_{i-1} without leaving the generating set $\mathcal{S}$.

While the above construction may seem artificial for the (fairly trivial) situation of the free Abelian monoid $\mathbb{N}^n$, a similar construction has proved to be an extremely powerful tool in a more complicated setting, namely that of the braid group B_n, respectively the braid monoid B_n^+.

The braid monoid admits a natural partial order, called the *prefix order*, defined as follows: For f, g in B_n^+ we say that $f \preccurlyeq g$ holds if g can be written as a product fh with h in B_n^+. (The prefix order can be defined for any left-cancellative monoid. Analogously, one can define a *suffix order* for any right-cancellative monoid.) At the center of the solutions to the Word and Conjugacy Problems in B_n found by Garside and others lies the *half twist* Δ, defined as the braid in B_n^+ in which any two strands cross exactly once. The set $\mathcal{S} = \{f \in B_n^+ \mid f \preccurlyeq \Delta\}$ generates the monoid B_n^+ and, moreover, every element of B_n^+ has a *unique* representation as a product of the form $g_1 g_2 \cdots g_k$ such that $g_1, \dots, g_k$ are nontrivial elements of $\mathcal{S}$ and $1 \neq f \preccurlyeq g_i$ implies $g_{i-1} f \notin \mathcal{S}$ for $i = 2, \dots, k$ and arbitrary f.

The discovery of various other situations that allow for an analogous construction led to the introduction of the notion of *Garside monoids.* A monoid M is called a Garside monoid if it satisfies the following conditions: (i) it is cancellative; (ii) it is a lattice with respect to the prefix order $\preccurlyeq$ and with respect to the suffix order $\succcurlyeq$; (iii) it is strongly Noetherian, meaning that there exists a map $\lambda : M \to \mathbb{N}$ such that, for f, g in M, $g \neq 1$ implies $\lambda(g) \geqslant 1$ and

$\lambda(fg) \geqslant \lambda(f) + \lambda(g)$, and (iv) there exists an element Δ of M, called a *Garside element* of M, such that the set $\mathcal{S} = \{f \in M \mid f \preccurlyeq \Delta\}$ equals the set $\{f \in M \mid \Delta \succcurlyeq f\}$, and $\mathcal{S}$ is a finite generating set of M. In this context, every element of M has a *normal decomposition* as a product of the form $g_1 g_2 \cdots g_k$ such that $g_1, \ldots, g_k$ are nontrivial elements of $\mathcal{S}$ and $1 \neq f \preccurlyeq g_i$ implies $g_{i-1} f \notin \mathcal{S}$ for $i = 2, \ldots, k$ and arbitrary f.

While the framework of Garside monoids has proved efficient, it has emerged that the hypotheses are too restrictive: There are many natural examples in which normal decompositions of elements completely analogous to the ones described above exist, although the structure in question is not a Garside monoid; we mention some of them below, referring to Dehornoy et al. (2013a) for details. By contrast, the framework of Garside families in a category, which we shall define below, will cover all these examples.

Infinite braids

We can define the *infinite braid group* B_∞ as the direct limit of the system obtained from $B_1, B_2, B_3, \ldots$ with the natural embedding of B_i into B_{i+1} for each i. The monoid B_∞^+ is defined in the same way from the monoids $B_1^+, B_2^+, \ldots$; it admits no finite generating family: any finite subfamily can only involve finitely many strands, but B_∞^+ contains braids involving crossings of strands with arbitrarily large index. Nevertheless, the family $\mathcal{S}_\infty = \bigcup_{i \geqslant 1} \{f \in B_\infty^+ \mid f \preccurlyeq \Delta_i\}$, where Δ_i denotes the Garside element of the Garside monoid B_i^+, allows normal decompositions of elements of B_∞^+ analogous to the ones described above.

Klein bottle group

Consider $K = \mathbb{Z} \rtimes_\varphi \mathbb{Z}$, where φ is the automorphism of $\mathbb{Z}$ defined by $\varphi(1) = -1$, and let K^+ be the submonoid of K generated by the elements $(1, 0)$ and $(0, 1)$ of K. It can be shown that K^+ admits the presentation $\langle \mathtt{ab} \mid \mathtt{a} = \mathtt{bab} \rangle$. As $\mathtt{ba} \preccurlyeq \mathtt{a}$ holds, the monoid K^+ is not Noetherian. Nevertheless, the set $\mathcal{S} = \{f \in K^+ \mid f \preccurlyeq \mathtt{a}^2\}$ allows normal decompositions of elements of K^+ analogous to the ones described above.

Wreathed free Abelian group

Consider the wreath product $\mathcal{T}_n = \mathbb{Z} \wr \mathfrak{S}_n$, that is, the semidirect product of the free Abelian group $\mathbb{Z}^n$ and the symmetric group $\mathfrak{S}_n$ with $\mathfrak{S}_n$ acting on $\mathbb{Z}^n$ by permuting the components. We define $\mathcal{T}_n^+$ to be the monoid $\mathbb{N} \wr \mathfrak{S}_n$. Write $\mathbf{0}$ for the element $(0, \ldots, 0)$ of $\mathbb{N}^n$ and $\mathbf{1}$ for the element $(1, \ldots, 1)$ of $\mathbb{N}^n$. Clearly, the elements of $\mathcal{T}_n$ of the form $(\mathbf{0}, \pi)$ with $1 \neq \pi \in \mathfrak{S}_n$ are nontrivial invertible elements of finite order. In particular, $\mathcal{T}_n^+$ cannot be a Garside monoid, as the quotient group of a Garside monoid is torsion-free. Nevertheless, the set $\mathcal{S} = \{(\boldsymbol{v}, 1) \mid \boldsymbol{v} \preccurlyeq \mathbf{1}\}$, where $\preccurlyeq$ is the partial order of the free Abelian monoid $\mathbb{N}^n$ we met earlier, allows normal decompositions of elements of $\mathcal{T}_n$ analogous to the ones described above.

Ribbon categories

For $n \geqslant 2$ and $1 \leqslant i, j < n$, define $\mathcal{BR}_n^+(i, j) = \{\beta \in B_n^+ \mid \sigma_i \beta = \beta \sigma_j\}$ and let $\mathcal{BR}_n^+$ be the category whose object set is $\{1, \ldots, n-1\}$ and whose family of morphisms with source i and target j is $\mathcal{BR}_n^+(i, j)$. Considering the Garside element Δ_n and the prefix order $\preccurlyeq_n$ of the braid monoid B_n^+, we define $\mathcal{S} = \bigcup_{1 \leqslant i, j < n} \{\beta \in \mathcal{BR}_n^+(i, j) \mid \beta \preccurlyeq_n \Delta_n\}$. It can be shown that, with respect to the partial order $\preccurlyeq$ induced by left-divisibility in the category $\mathcal{BR}_n^+$, the elements of $\mathcal{BR}_n^+$ admit normal decompositions as products of elements of $\mathcal{S}$, analogous to the ones described above. The

notion of the ribbon category, and the existence of normal decompositions, prove to be crucial for the analysis of normalizers and centralizers in braid groups.

The existence of the above examples (and of many other similar ones), which do not enter the framework of Garside groups but nevertheless seem to share their main properties, clearly suggests that a more general unifying framework should exist, and our claim is that the Garside families investigated in the rest of this paper provide such a framework. The motivation for the extension is twofold: the first goal is to capture more cases, the second one is to isolate the really essential assumptions, thus deepening our understanding of the structures involved.

1. The general context

In this introductory section, we present the background of categories and Garside families, together with some general existence and uniqueness results that will be used and, often, refined in the sequel of the paper. Proofs appear in other sources and they will be omitted in general.

1.1. Categories

The general context is that of categories, which should be seen here just as monoids with a partial product, that is, one that is not necessarily defined everywhere. A *precategory* (or multigraph) is a family $\mathcal{A}$ plus two maps, "source" and "target", of $\mathcal{A}$ to another family $\mathcal{O}bj(\mathcal{A})$ (the objects of $\mathcal{A}$), and a *category* is a precategory equipped with a partial multiplication such that fg exists if and only if the target of f, denoted $\mathrm{trg}(f)$, coincides with the source of g, denoted $\mathrm{src}(g)$. The multiplication is associative whenever defined and, in addition, has a neutral element 1_x for each object x, that is, $1_x g = g = g 1_y$ holds for every g with source x and target y. If C is a category, the family of all neutral elements is denoted by $\mathbf{1}_C$ and, for $\mathcal{A}$ included in C and x, y in $\mathcal{O}bj(C)$, the family of elements of $\mathcal{A}$ with source x and target y is denoted by $\mathcal{A}(x, y)$. A monoid is the special case of a category when there is only one object, so that the product is always defined. It is convenient to represent the elements of a category using arrows, so that an element g with source x and target y is represented by $x \xrightarrow{g} y$.

Free categories and paths

If $\mathcal{A}$ is a precategory, the free category generated by $\mathcal{A}$ is the family $\mathcal{A}^*$ of all $\mathcal{A}$-paths, that is, all finite sequences $(s_1, \dots, s_p)$ of elements of $\mathcal{A}$ such that the target of s_{i-1} is the source of s_i for every i, together with, for each object x, an empty path ε_x, and equipped with concatenation of paths. When w is an $\mathcal{A}$-path, we denote by $\|w\|$ the length of w, and, for $1 \leqslant i \leqslant \|w\|$, we denote by $w[i]$ the i-th entry in w. For $\mathcal{B}$ included in $\mathcal{A}$, we denote the family of all $\mathcal{B}$-paths of length p by $\mathcal{B}^{[p]}$. If w_1, w_2 are two paths, we denote by $w_1|w_2$ the concatenation of w_1 and w_2 when it exists, that is, when the target of w_1 (defined to be the target of the last entry in w_1) coincides with the source of w_2 (defined to be the source of the first entry in w_2). We identify a length one path with its unique entry. Then a length p path $(s_1, \dots, s_p)$ is the concatenation of the length one paths made of its successive entries, so that it can be denoted by $s_1|\cdots|s_p$. When $\mathcal{A}$ is a set, that is, a precategory with one object only, the condition about source and targets vanishes, and it is usual to say $\mathcal{A}$-*word* for $\mathcal{A}$-path.

Presentations

Every category that is generated by a family $\mathcal{A}$ is a quotient of the free category $\mathcal{A}^*$, and, for $\mathcal{R}$ a family of pairs of $\mathcal{A}$-paths, it is said to admit the presentation $(\mathcal{A};\mathcal{R})$ if it is isomorphic to $\mathcal{A}^*/{\equiv^+_{\mathcal{R}}}$ where ${\equiv^+_{\mathcal{R}}}$ is the congruence on $\mathcal{A}^*$ generated by $\mathcal{R}$. The elements of $\mathcal{R}$ are called *relations*, and, in this context, it is customary to write $u = v$ instead of (u, v) for a relation and, in concrete examples, to omit the concatenation sign, thus writing $s_1 \cdots s_p = t_1 \cdots t_q$ rather than $s_1|\cdots|s_p = t_1|\cdots|t_q$. If $(\mathcal{A};\mathcal{R})$ is a presentation, we write $\langle \mathcal{A} \,|\, \mathcal{R}\rangle^+$ for the category presented by $(\mathcal{A};\mathcal{R})$—which is determined only up to isomorphism—and, for w an $\mathcal{A}$-path, we write $[w]$ for the ${\equiv^+_{\mathcal{R}}}$-class of w, that is, for the element of $\langle \mathcal{A} \,|\, \mathcal{R}\rangle^+$ represented by w—which is also the evaluation of the path w in the category $\langle \mathcal{A} \,|\, \mathcal{R}\rangle^+$.

Cancellativity

All categories we shall consider here will have to satisfy some cancellativity condition, at least on one side.

Definition 1.1. A category C is called *left-cancellative* (*resp. right-cancellative*) if $fg = fg'$ (*resp.* $gf = g'f$) implies $g = g'$ for all f, g, g' in C.

A category is called *cancellative* if it is both left- and right-cancellative. In a left- (or right-) cancellative category, an element has a left-inverse if and only if it has a right-inverse, and so there is a unique, non-ambiguous notion of *invertible* element. For C a left-cancellative category, we denote by $C^\times$ the subgroupoid of C consisting of all invertible elements. An invertible element will be called *nontrivial* if it is not an identity-element 1_x.

Divisibility, lcm, lcm-selector

Associated with every (left-cancellative) category—hence, in particular, every monoid—comes a natural left-divisibility relation.

Definition 1.2. Assume that C is a left-cancellative category. For f, g in C, we say that f is a *left-divisor* of g, or, equivalently, that g is a *right-multiple* of f, denoted $f \preccurlyeq g$, if there exists g' in C so that $fg' = g$ holds.

The hypothesis that the ambient category is left-cancellative is needed to guarantee that left-divisibility is a partial preordering; the associated equivalence relation is right-multiplication by an invertible element: the conjunction of $f \preccurlyeq g$ and $g \preccurlyeq f$ is equivalent to the existence of an invertible element e satisfying $fe = g$, which will be denoted by $f =^\times g$ hereafter.

Simple derived notions stem from the left-divisibility relation, corresponding to greatest common lower bound and least common upper bound. We say that h is a *greatest common left-divisor*, or *left-gcd* of f and g if h left-divides f and g and every left-divisor of f and g left-divides h. Symmetrically, we say h is a *least common right-multiple*, or *right-lcm* of f and g if h is a right-multiple of f and g and every right-multiple of f and g is a right-multiple of h. Right-lcms and left-gcds, if they exist, are unique up to right-multiplication by an invertible element, hence unique if the ambient category has no nontrivial invertible element. In the latter case, we write $f \vee g$ and $f \wedge g$ for the right-lcm and the left-gcd of f and g (if they exist), and $f \setminus g$ for the unique element that satisfies $f \vee g = f(f \setminus g)$.

Definition 1.3. We say that a left-cancellative category C *admits right-lcms* (*resp. admits local right-lcms*) if any two elements of C with the same source (*resp.* if any two elements of C that admit a common right-multiple) admit a right-lcm.

Finally, we shall sometimes need to choose right-lcms explicitly. The following terminology is then natural.

Definition 1.4. Assume that $\mathcal{C}$ is a left-cancellative category, and $\mathcal{A}$ is a generating subfamily of $\mathcal{C}$. A *right-lcm selector* on $\mathcal{A}$ is a partial map $\theta : \mathcal{A} \times \mathcal{A} \to \mathcal{A}^*$ such that, for all s, t in $\mathcal{A}$, the elements $\theta(s, t)$ and $\theta(t, s)$ are defined if and only if s and t admit a right-lcm and, in this case, there exists a right-lcm h of s and t such that both $s|\theta(s, t)$ and $t|\theta(t, s)$ represent h.

Note that, if $\mathcal{C}$ is a left-cancellative category that admits no nontrivial invertible element, then the map $(f, g) \mapsto f \setminus g$ is a right-lcm selector on $\mathcal{C}$.

The notions of a right-divisor and left-multiple are defined symmetrically, and so are the derived notions of a right-gcd, and a left-lcm selector.

Noetherianity conditions

Definition 1.5. A left-cancellative category $\mathcal{C}$ is called *right-Noetherian* if every bounded $\prec$-increasing sequence in $\mathcal{C}$ is finite.

By extension, we say that a (positive) presentation is right-Noetherian if the associated category is right-Noetherian. Standard results (see for instance Lévy, 1979) give the following criterion for establishing Noetherianity conditions.

Lemma 1.6. *A presentation $(\mathcal{A}; \mathcal{R})$ is right-Noetherian if and only if there exists a map λ of $\mathcal{A}^*$ to the ordinals that is $\equiv^+_{\mathcal{R}}$-invariant and satisfies $\lambda(s) > 0$ for every s in $\mathcal{A}$ and $\lambda(s|w) > \lambda(w)$ for all w in $\mathcal{A}^*$ and s in $\mathcal{A}$.*

Note that every presentation $(\mathcal{A}; \mathcal{R})$ such that $\mathcal{R}$ consists of relations of the form $u = v$ with u, v of the same length is right-Noetherian, as one can then define $\lambda(w)$ to be the length of w.

Ore category, groupoid of fractions

Finally, we recall that a groupoid is a category in which every element is invertible, a group corresponding to the special case when there is only one object, that is, the product is always defined.

We say that a groupoid $\mathcal{G}$ is a *groupoid of left-fractions* for a subcategory $\mathcal{C}$ if every element of $\mathcal{G}$ admits an expression of the form $f^{-1}g$ with f, g in $\mathcal{C}$. The following result of Ore is classical:

Proposition 1.7. *(Clifford and Preston, 1961) Say that a category is* left-Ore *(resp.* right-Ore*) if it is cancellative and any two elements that have the same target (resp. source) admit a common left-multiple (resp. right-multiple). Then a category embeds in a groupoid of left-fractions (resp. of right fractions) if and only if it is left-Ore (resp. right-Ore).*

A category that is both left- and right-Ore will be called an *Ore category*.

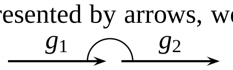
1.2. Normal decompositions and Garside families

The central idea in our approach consists in introducing distinguished decompositions of a certain type for the elements of the considered category. These decompositions involve a reference subfamily of the ambient category and correspond to the principle of recursively selecting maximal left-divisors lying in the reference family.

Definition 1.8. Assume that $\mathcal{C}$ is a left-cancellative category. For $\mathcal{S}$ included in $\mathcal{C}$, a $\mathcal{C}$-path $g_1|\cdots|g_p$ is called *$\mathcal{S}$-greedy* (*resp*. *$\mathcal{S}$-normal*) if, for every $i < p$, we have

$$\forall s{\in}\mathcal{S}\ \forall f{\in}\mathcal{C}\ (s \preccurlyeq fg_ig_{i+1} \Rightarrow s \preccurlyeq fg_i) \tag{1.9}$$

(*resp*. this and, in addition, every entry g_i lies in $\mathcal{S}^\sharp$, defined to be $\mathcal{S}\mathcal{C}^\times \cup \mathcal{C}^\times$).

When using diagrams in which the elements of the category are represented by arrows, we shall indicate that a path $g_1|g_2$ is $\mathcal{S}$-greedy by appending a small arc as in 

.

Example 1.10. Consider the free Abelian monoid M defined by the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{ab} = \mathtt{ba})$ and let $\mathcal{S} = \{\mathtt{a}, \mathtt{b}, \mathtt{ab}\}$, whence $\mathcal{S}^\sharp = \{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}\}$. The $\mathcal{S}$-normal paths in M are precisely the paths $g_1|\cdots|g_q$ that, for some p, p' in $\{0, ..., q\}$, satisfy (i) $g_i = \mathtt{ab}$ for $i = 1, ..., p$; (ii) either $g_i = \mathtt{a}$ for $i = p+1, ..., p'$, or $g_i = \mathtt{b}$ for $i = p+1, ..., p'$; and (iii) $g_i = 1$ for $i = p'+1, ..., q$.

Example 1.11. Consider the wreathed free Abelian monoid $\mathcal{T} = \mathbb{N}^2 \rtimes \mathfrak{S}_2$, where the symmetric group $\mathfrak{S}_2$ acts by permuting the coordinates of $\mathbb{N}^2$. Defining $\mathbf{0} = [0, 0]$ in $\mathbb{N}^2$, we clearly have $\mathcal{T}^\times = \{\mathbf{0}\} \times \mathfrak{S}_2$, that is, there exists a nontrivial invertible element in $\mathcal{T}$.

Here, and in the sequel, we consider the elements $\mathrm{id} = 1_{\mathfrak{S}_2}$ and $\tau = (1\,2)$ of $\mathfrak{S}_2$, the elements $\boldsymbol{v}_1 = [1, 0]$, $\boldsymbol{v}_2 = [0, 1]$ and $\boldsymbol{v}_{12} = [1, 1]$ in $\mathbb{N}^2$, as well as the elements $\mathtt{a} = (\boldsymbol{v}_1, \mathrm{id})$, $\mathtt{b} = (\boldsymbol{v}_2, \mathrm{id})$ and $\mathtt{e} = (\mathbf{0}, \tau)$ in $\mathcal{T}$.

For $\mathcal{A} = \{\mathtt{a}, \mathtt{b}, \mathtt{e}\}$, we have $\mathcal{A}^\sharp = \{\mathbf{0}, \boldsymbol{v}_1, \boldsymbol{v}_2\} \times \mathfrak{S}_2$. For $g_1 = (\boldsymbol{v}_1, \tau)$ and $g_2 = (\boldsymbol{v}_2, \tau)$, the $\mathcal{T}$-path $g_1|g_2$ is $\mathcal{A}$-normal, but the $\mathcal{T}$-paths $g_1|g_1$ and $\mathtt{e}|g_1$ are not. In fact, there is no $\mathcal{A}$-normal path w satisfying $[w] = [g_1|g_1] = (v_{12}, \mathrm{id}) = \mathtt{ab} = \mathtt{ba}$, since one has $(\boldsymbol{v}_1, \sigma) \preccurlyeq \mathtt{ab}$ and $(\boldsymbol{v}_2, \sigma) \preccurlyeq \mathtt{ab}$, but $(\boldsymbol{v}_1, \sigma) \not\preccurlyeq \mathtt{b}$ and $(\boldsymbol{v}_2, \sigma) \not\preccurlyeq \mathtt{a}$ for all σ in $\mathfrak{S}_2$. Finally, the $\mathcal{T}$-path $\mathtt{ab}|\mathtt{ab}$ is $\mathcal{A}$-greedy but not $\mathcal{A}$-normal.

If g is an element of a category $\mathcal{C}$, a path w satisfying $[w] = g$ is called a *decomposition* of g. What we shall be interested in in the sequel are the (possible) $\mathcal{S}$-normal decompositions of the elements of the considered category. One of the interests of such decompositions is that they are essentially unique.

Proposition 1.12. *(Dehornoy et al., 2013b, Proposition 2.11) Assume that $\mathcal{C}$ is a left-cancellative category and $\mathcal{S}$ is included in $\mathcal{C}$. Then any two $\mathcal{S}$-normal decompositions of an element of $\mathcal{C}$ (if any) are $\mathcal{C}^\times$-deformations of one another, where $s_1|\cdots|s_p$ is said to be a $\mathcal{C}^\times$*-deformation *of $t_1|\cdots|t_q$ if there exist invertible elements $e_0, ..., e_m$, with $m = \max(p, q)$, such that e_0, e_m are identity-elements and $t_ie_i = e_{i-1}s_i$ holds for $1 \leqslant i \leqslant m$, where, for $p \neq q$, the shorter path is expanded by identity-elements (see Figure 1).*

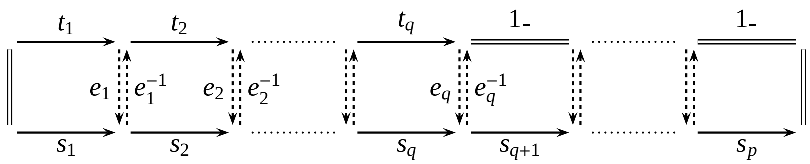


Figure 1: Deformation by invertible elements: invertible elements connect the corresponding entries; if one path is shorter than the other (here we are in the case $q < p$), it is extended by identity-elements.

Note that, if $\mathcal{C}$ contains no nontrivial invertible element, that is, the only invertible elements are the identity-elements, then Proposition 1.12 provides a genuine uniqueness result provided one discards the $\mathcal{S}$-normal paths that finish with an identity-element.

As for the existence of $\mathcal{S}$-normal decompositions, it naturally depends on the family $\mathcal{S}$. Here is where Garside families appear:

Definition 1.13. A subfamily $\mathcal{S}$ of a left-cancellative category C is called a *Garside family* if every element of C admits an $\mathcal{S}$-normal decomposition.

Every left-cancellative category is a Garside family in itself, so every left-cancellative category contains a Garside family. Practically recognizing whether a given family is a Garside family will be one of the main tasks of Sections 2 and 3 below. We shall appeal to the following simple characterization which is valid whenever the ambient category satisfies special assumptions:

Lemma 1.14. *(Dehornoy et al., 2013a, Corollary IV.2.18) Assume that C is a left-cancellative category that is right-Noetherian and admits unique local right-lcms. Then a subfamily $\mathcal{S}$ of C is a Garside family in C if and only if $\mathcal{S}$ generates C and, for all s, t in $\mathcal{S}$ admitting a common right-multiple, $s \vee t$ and $s \setminus t$ lie in $\mathcal{S} \cup \mathbf{1}_C$.*

Also, we shall use the following closure result, which in some sense extends Lemma 1.14, but need not characterize Garside families in general.

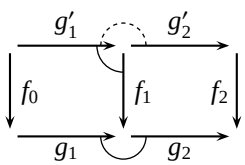
Lemma 1.15. *(Dehornoy et al., 2013b, Proposition 3.9) If $\mathcal{S}$ is a Garside family in a left-cancellative category C and r is a common right-multiple of two elements s, t of $\mathcal{S}^\sharp$, there exists a common right-multiple r' of s and t such that r is a right-multiple of r' and r', together with s' and t' defined by $st' = ts' = r'$, lie in $\mathcal{S}^\sharp$.*

An application of Lemma 1.15 is that every Garside family gives rise to a simple presentation of the ambient category. For our current purpose it will be sufficient to state the result in the particular case when no nontrivial invertible element exists, so that a right-lcm is unique when it exists.

Proposition 1.16. *(Dehornoy et al., 2013a, Proposition IV.3.6) Assume that $\mathcal{S}$ is a Garside family in a left-cancellative category C that contains no nontrivial invertible element and admits local right-lcms. Let $\mathcal{R}$ consist of all relations $s(s \setminus t) = t(t \setminus s)$ for s, t in $\mathcal{S}$ admitting a common right-multiple. Then $(\mathcal{S}; \mathcal{R})$ is a presentation of C.*

Finally, we shall use the following simple diagrammatic rule about $\mathcal{S}$-greedy paths.

Lemma 1.17. *(Dehornoy et al., 2013b, Lemma 3.3) (first domino rule) Assume that C is a left-cancellative category, $\mathcal{S}$ is included in C, and we have a commutative diagram with edges in C as on the right. If $g_1|g_2$ and $g'_1|f_1$ are $\mathcal{S}$-greedy, then $g'_1|g'_2$ is $\mathcal{S}$-greedy as well.*



A second, symmetric domino rule will be mentioned in Lemma 1.32 below.

1.3. Symmetric normal decompositions and strong Garside families

If C is a left-Ore category, then, by Proposition 1.7, it embeds in the groupoid of left-fractions that we shall denote by $\mathcal{E}nv(C)$ (like "enveloping groupoid"). If $\mathcal{S}$ is a Garside family in C, every element of C admits an $\mathcal{S}$-normal decomposition, and it is natural to try to extend the result from C to $\mathcal{E}nv(C)$, that is, to find distinguished decompositions for the elements of $\mathcal{E}nv(C)$ in terms of elements of $\mathcal{S}^\sharp$ and their inverses.

To do it, we extend the notion of an $\mathcal{A}$-path into that of a signed $\mathcal{A}$-path. Formally, if $\mathcal{A}$ is any precategory, we introduce a family $\overline{\mathcal{A}}$ that is disjoint from and in bijection to $\mathcal{A}$ as $\overline{\mathcal{A}} = \{\overline{s} \mid s \in \mathcal{A}\}$, where the source of $\overline{s}$ is the target of s and vice versa. A *signed $\mathcal{A}$-path* is defined to be a $(\mathcal{A} \cup \overline{\mathcal{A}})$-path. We extend the "bar" map to all signed paths by defining $\overline{(\overline{s})} = s$ for s in $\mathcal{A}$ and $\overline{w_1|w_2} = \overline{w_2}|\overline{w_1}$. For $\mathcal{G}$ a groupoid and $\mathcal{A}$ included in $\mathcal{G}$, we extend the notation $[w]$ to signed $\mathcal{A}$-paths by declaring that $[\,\overline{g}\,]$ is g^{-1} for g in $\mathcal{A}$, that is, we use the letters of $\overline{\mathcal{A}}$ to represent the inverses of the elements of $\mathcal{A}$. If w is a signed path and $[w] = g$ holds, we again say that w is a decomposition of g.

When C is a left-Ore category and $\mathcal{S}$ is a Garside family of C, we now look for distinguished decompositions for the elements of the groupoid $\mathcal{E}nv(C)$. As every element of $\mathcal{E}nv(C)$ is a left-fraction, it is natural to consider decompositions that are *negative–positive* $\mathcal{S}^\sharp$-paths, this meaning that every negative entry precedes every positive entry, where the entries in $\mathcal{S}^\sharp$ are called positive and those in $\overline{\mathcal{S}^\sharp}$ are called negative.

Definition 1.18. Assume that C is a left-Ore category.

(i) Two elements f, g of C are called *left-disjoint* if, for all f', g' in C satisfying $f^{-1}g = f'^{-1}g'$ in $\mathcal{E}nv(C)$, there exists h in C satisfying $f' = hf$ and $g' = hg$.

(ii) For $\mathcal{S}$ included in C, a negative–positive path $\overline{g_q}|\cdots|\overline{g_1}|f_1|\cdots|f_p$ is called *symmetric $\mathcal{S}$-greedy* (*resp. symmetric $\mathcal{S}$-normal*) if $g_1|\cdots|g_q$ and $f_1|\cdots|f_p$ are $\mathcal{S}$-greedy (*resp.* $\mathcal{S}$-normal) and, in addition, g_1 and f_1 are left-disjoint.

When using diagrams, we shall indicate that two elements f, g are left-disjoint by appending a small arc as in

. So a generic symmetric $\mathcal{S}$-greedy path pictorially corresponds to a diagram

,

and it is symmetric $\mathcal{S}$-normal if, in addition, all edges correspond to elements of $\mathcal{S}^\sharp$. Note that a positive path is symmetric $\mathcal{S}$-normal if and only if it $\mathcal{S}$-normal: indeed, a positive path is a negative–positive path whose negative part is empty, and, for every g in $\mathcal{S}^\sharp(x, \text{-})$, the elements ε_x and g are (trivially) left-disjoint.

Example 1.19. Consider the free Abelian monoid M and the set $\mathcal{S}$ as in Example 1.10. The unordered pairs of left-disjoint elements of $\mathcal{S}^\sharp$ are precisely $\{1, 1\}$, $\{1, \mathtt{a}\}$, $\{1, \mathtt{b}\}$, $\{1, \mathtt{ab}\}$, and $\{\mathtt{a}, \mathtt{b}\}$, where $1 = 1_x$ for the unique element x of $Obj(M)$.

Like $\mathcal{S}$-normal decompositions in the positive case, symmetric $\mathcal{S}$-normal decompositions turn out to be (nearly) unique when they exist.

Proposition 1.20. *(Dehornoy et al., 2013a, Proposition III.2.16) Assume that C is a left-Ore category and $\mathcal{S}$ is included in C. Then any two symmetric $\mathcal{S}$-normal decompositions of an element of $\mathcal{E}nv(C)$ (if any) are $C^\times$-deformations of one another, this meaning that there exists a commutative diagram as in Figure 2.*

As for existence, symmetric normal decompositions are connected with left-lcms:

Lemma 1.21. *(Dehornoy et al., 2013a, Lemma III.2.19) Assume that $\mathcal{S}$ is a Garside family in a left-Ore category C, that $s_1|\cdots|s_p$ and $t_1|\cdots|t_q$ are $\mathcal{S}$-normal paths, and that $t_1\cdots t_q g = s_1\cdots s_p f$ holds. Then $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ is symmetric $\mathcal{S}$-greedy if and only if $t_1\cdots t_q$ and $s_1\cdots s_p$ are left-disjoint, if and only if $t_1\cdots t_q g$ is a left-lcm of f and g.*

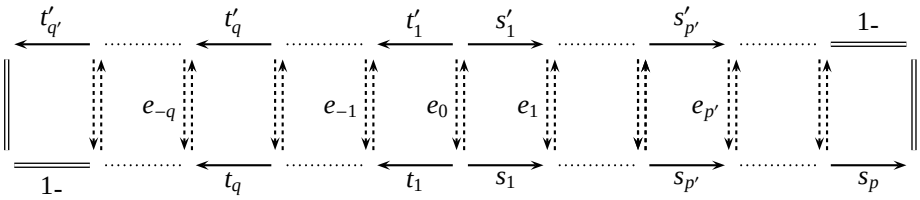


Figure 2: Deformation of a signed path by invertible elements: invertible elements connect the corresponding entries ; if some path is shorter than the other (here we have $q < q'$ and $p' < p$), it is extended by identity-elements.

Lemma 1.21 implies that an element fg^{-1} of $\mathcal{E}nv(C)$ admits a symmetric $\mathcal{S}$-normal decomposition if and only if f and g admit a left-lcm in C, whence:

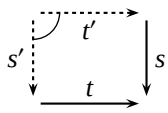
Proposition 1.22. *If $\mathcal{S}$ is a Garside family in a left-Ore category C that admits left-lcms, every element of $\mathcal{E}nv(C)$ that can be expressed as a right-fraction admits a symmetric $\mathcal{S}$-normal decomposition.*

In particular, if C is an Ore category, then every element of $\mathcal{E}nv(C)$ can be expressed as a right-fraction, and we obtain:

Corollary 1.23. *If $\mathcal{S}$ is a Garside family in an Ore category C that admits left-lcms, every element of $\mathcal{E}nv(C)$ admits a symmetric $\mathcal{S}$-normal decomposition.*

When algorithmic questions are involved, it will be convenient to consider special Garside families that we introduce now.

Definition 1.24. Assume that C is a left-Ore category. A Garside family $\mathcal{S}$ of C is called *strong* if, for all s, t in $\mathcal{S}^\sharp$ with the same target, there exist s', t' in $\mathcal{S}^\sharp$ that are left-disjoint and satisfy $s't = t's$.



The interest of introducing the notion of a strong Garside family is to allow for a refined version of Proposition 1.22:

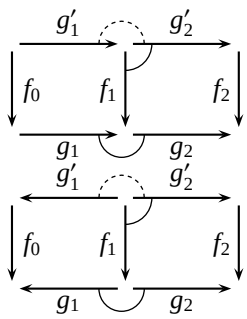
Proposition 1.25. *(Dehornoy et al., 2013a, Proposition III.2.31) If $\mathcal{S}$ is a strong Garside family in a left-Ore category C that admits left-lcms, every element of $\mathcal{E}nv(C)$ that admits a positive–negative $\mathcal{S}^\sharp$-decomposition of length ℓ admits a symmetric $\mathcal{S}$-normal decomposition of length at most ℓ.*

Finally, as in the positive case, we shall appeal to diagrammatic rules involving normal paths.

Lemma 1.26. *(Dehornoy et al., 2013a, Propositions III.2.39 and III.2.42)*

(i) *(third domino rule) Assume that C is a left-cancellative category, $\mathcal{S}$ is included in C, and we have a commutative diagram with edges in C as on the right. If $g_1|g_2$ is $\mathcal{S}$-greedy, and f_1, g'_2 are left-disjoint, then $g'_1|g'_2$ is $\mathcal{S}$-greedy as well.*

(ii) *(fourth domino rule) Assume that C is a left-Ore category, $\mathcal{S}$ is included in C, and we have a commutative diagram with edges in C as on the right. If g_1, g_2 are left-disjoint, and f_1, g'_2 are left-disjoint, then g'_1, g'_2 are left-disjoint as well.*



1.4. Bounded Garside families and Δ-normal decompositions

In many cases, interesting Garside families in a monoid consist of the left-divisors of some maximal element Δ, in which case it is natural to call them bounded by Δ. In a category context, the maximal element has to depend on the source, and it is replaced with a map from the objects to the elements.

Definition 1.27. A Garside family $\mathcal{S}$ in a cancellative category C is called *bounded* if there exists a map Δ from $Obj(C)$ to C satisfying the following conditions:

(i) $s \in \mathcal{S}(x, \text{-})$ implies $s \preccurlyeq \Delta(x)$,

(ii) $\forall y \in Obj(C)\ \exists! z \in Obj(C)\ \forall s \in \mathcal{S}^\sharp(\text{-}, y)\ \exists s' \in \mathcal{S}^\sharp(z, \text{-})\ (s's = \Delta(z))$.

Note that, in Definition 1.27, (i) and (ii) are symmetric, as (i) can be stated as

$$\forall x \in Obj(C)\ \exists! y \in Obj(C)\ \forall s \in \mathcal{S}^\sharp(x, \text{-})\ \exists s' \in \mathcal{S}^\sharp(\text{-}, y)\ (ss' = \Delta(x)),$$

with y the target of $\Delta(x)$. In the above context, for every element s of C, the unique element s' satisfying $ss' = \Delta(x)$ is denoted by $\partial(s)$, whereas the unique element s' satisfying $s's = \Delta(z)$ for some z is denoted by $\widetilde{\partial}(s)$.

Remark 1.28. A Garside family satisfying Definition 1.27(i) only is called *right-bounded* by Δ. The latter notion is natural and useful in the positive case, but it is not sufficient in the signed case and we shall not consider it here.

Proposition 1.29. *(Dehornoy et al., 2013a, Proposition VI.3.11)* (i) *Every cancellative category that admits a bounded Garside family is an Ore category.*

(ii) *Every bounded Garside family is strong.*

The most important technical property implied by the existence of the bounding map Δ is the existence of a derived automorphism of the ambient category.

Lemma 1.30. *(Dehornoy et al., 2013a, Proposition VI.1.11) Assume that $\mathcal{S}$ is a Garside family bounded by Δ in a cancellative category C. Put $\phi(x) = \mathrm{trg}(\Delta(x))$ for x in $Obj(C)$ and $\phi(g) = \partial^2(g)$ for g in C. Then ϕ is an automorphism of C that makes the diagram aside commutative for every g in $C(x, y)$.*

$$\begin{tikzcd} x \arrow[r, "g"] \arrow[d, "\Delta(x)"'] & y \arrow[d, "\Delta(y)"] \arrow[dl, "\partial(g)"'] \\ \phi(x) \arrow[r, "\phi(g)"'] & \phi(y) \end{tikzcd}$$

In terms of normal decompositions, the bounding map implies a simple connection between greediness and left-disjointness that need not be valid in general.

Lemma 1.31. *(Dehornoy et al., 2013a, Proposition VI.1.46) Assume that $\mathcal{S}$ is a Garside family bounded by a map Δ in a cancellative category C. Then, for all s_1, s_2 in $\mathcal{S}^\sharp$, the following are equivalent:*

(i) *$s_1|s_2$ is $\mathcal{S}$-normal;*

(ii) *$\partial(s_1)$ and s_2 are left-disjoint;*

(iii) *$\partial(s_1)$ and s_2 admit no nontrivial common left-divisor.*

A consequence is the following counterpart of the first domino rule.

Lemma 1.32. *(Dehornoy et al., 2013a, Lemma VI.1.32) (second domino rule) Assume that C is a cancellative category and $\mathcal{S}$ is a bounded Garside family in C. Then, when we have a commutative diagram as on the right with edges in $\mathcal{S}^\sharp$ in which $s_1|s_2$ and $t_1|s'_2$ are $\mathcal{S}$-greedy, then $s'_1|s'_2$ is $\mathcal{S}$-greedy as well.*

$$\begin{tikzcd} \cdot \arrow[r, "s_1"] \arrow[d, "t_0"'] & \cdot \arrow[r, "s_2"] \arrow[d, "t_1"'] & \cdot \arrow[d, "t_2"] \\ \cdot \arrow[r, "s'_1"'] & \cdot \arrow[r, "s'_2"'] & \cdot \end{tikzcd}$$

Note that the second domino rule is not an exact counterpart of the first one in that, here, all involved elements are supposed to lie in the reference family $\mathcal{S}^\sharp$.

Also, a second type of distinguished decomposition naturally arises for the elements of the associated groupoid of the considered category: called Δ-*normal*, these decompositions differ from symmetric normal decompositions in that the denominator is demanded to involve a power of the Garside map Δ. In the case of a monoid M, a bound Δ for a Garside family is an element of M, and it makes sense to take powers of Δ. In the case of a general category C, the bound is a map of $Obj(C)$ to C, and the notion of a power has to be adapted.

Notation 1.33. Assume that C is a cancellative category and Δ is a Garside map in C. For n in $\mathbb{Z}$ and x in $Obj(C)$, we put

$$\Delta^{[n]}(x) := \begin{cases} \Delta(x)|\Delta(\phi(x))|\cdots|\Delta(\phi^{n-1}(x)) & \text{for } n > 0, \\ \varepsilon_x & \text{for } n = 0, \\ \overline{\Delta(\phi^{-1}(x))}|\overline{\Delta(\phi^{-2}(x))}|\cdots|\overline{\Delta(\phi^{-|n|}(x))} & \text{for } n < 0, \end{cases} \tag{1.34}$$

and we write $\Delta^{(n)}(x)$ for the element of $\mathcal{E}nv(C)$ represented by $\Delta^{[n]}(x)$.

Note that, in every case, the source of $\Delta^{[n]}(x)$ and $\Delta^{(n)}(x)$ is x and its target is $\phi^n(x)$, and that $\Delta^{(-n)}(x)$ is always the inverse of $\Delta^{(n)}(\phi^{-n}(x))$.

Definition 1.35. Assume that $\mathcal{S}$ is a Garside family of C bounded by a map Δ in an Ore category C.

(i) An element g of $C(x, \text{-})$ is called Δ-*like* if $g =^\times \Delta(x)$ holds.

(ii) A signed $\mathcal{S}$-path is called Δ-*normal* if it has the form $\Delta^{[n]}(\text{-})|s_1|\cdots|s_p$ with n in $\mathbb{Z}$ and $s_1|\cdots|s_p$ an $\mathcal{S}$-normal path such that s_1 is not Δ-like.

Thus a Δ-normal path is either a positive $\mathcal{S}$-path beginning with elements in the image of Δ, or an empty path, or a negative–positive $\mathcal{S}$-path whose negative part consists of elements in the image of Δ.

Proposition 1.36. *(Dehornoy et al., 2013a, Proposition VI.3.19) Assume that $\mathcal{S}$ is a Garside family of C bounded by a map Δ in an Ore category C. Then every element of $\mathcal{E}nv(C)$ admits a Δ-normal decomposition, in which the exponent of Δ is uniquely determined and the other entries are unique up to $C^\times$-deformation.*

In the positive case, it is easily seen that Δ-like entries must lie at the beginning of an $\mathcal{S}$-normal path, so a Δ-normal path is simply an $\mathcal{S}$-normal path in which the initial Δ-like entries are not only Δ-like but even lie in the image of Δ. In the signed case, the difference with a symmetric $\mathcal{S}$-normal path is more important: in a Δ-normal path, the numerator and the denominator need not be left-disjoint, the requirement being now that Δ does not left-divide the first positive entry. Actually, Propositions 1.22 and 1.36 are not exactly comparable, as they require different assumptions, namely the existence of left-lcms in the former case, and that of a bounded Garside family in the latter.

2. Recognizing Garside families, case of a presentation

We now begin to investigate the effective methods relevant for Garside structures. In this section as well as the next one, we address the question of recognizing that a given family is a

Garside family in the ambient category, as well as checking that the category is eligible for the Garside approach, that is, it is left-cancellative (*resp*. left-Ore). The question depends in turn on the way the category and the candidate Garside family are specified. In this section, we consider the case when the category is specified using a presentation.

We first define reversing (Subsection 2.1), then address establishing that the ambient category is cancellative (Subsection 2.2), recognizing Garside families (Subsection 2.3), and finally proving further properties like being an Ore category or a strong Garside family (Subsection 2.4).

2.1. The reversing transformation

Before entering the main development, we introduce a technical tool that will be used several times in the sequel, namely a path transformation called reversing (Dehornoy, 1997, 2003, 2011).

Definition 2.1. Assume that $\mathcal{A}$ is a precategory. A *right-complement* on $\mathcal{A}$ is a partial map θ of $\mathcal{A}^2$ to $\mathcal{A}^*$ such that $\theta(s, s)$ is defined and equal to ε_y for every s in $\mathcal{A}(x, y)$ and that, if $\theta(s, t)$ is defined, then s and t have the same source, $\theta(t, s)$ is defined, and both $s|\theta(s, t)$ and $t|\theta(t, s)$ are defined and have the same target. A right-complement is called *short* if, whenever $\theta(s, t)$ is defined, $\theta(s, t)$ has length at most 1, that is, it belongs to $\mathcal{A}$ or is empty.

We note that, if $\mathcal{O}bj(\mathcal{A})$ consists of a single element, the conditions for θ to be a right-complement are simply that $\theta(s, s)$ is empty for every s and that $\theta(s, t)$ is defined if and only if $\theta(t, s)$ is.

Definition of right-reversing

We are now ready to introduce right-reversing. We begin with the special case of a short right-complement and a negative–positive input path, which will be frequently used in the sequel.

Notation 2.2. If $\mathcal{A}$ is a precategory, we write $\varepsilon_{\mathcal{A}}$ for $\{\varepsilon_x \mid x \in \mathcal{O}bj(\mathcal{A})\}$ and $\widehat{\mathcal{A}}$ for $\mathcal{A} \cup \varepsilon_{\mathcal{A}}$. For θ a right-complement on $\mathcal{A}$, we write $\widehat{\theta}$ for the extension of θ to $\widehat{\mathcal{A}}$ obtained by adding $\widehat{\theta}(\varepsilon_x, \varepsilon_x) = \varepsilon_x$ for every object x plus $\widehat{\theta}(s, \varepsilon_x) = \varepsilon_y$ and $\widehat{\theta}(\varepsilon_x, s) = s$ for every s in $\mathcal{A}(x, y)$.

Note that $\widehat{\theta}$ is a right-complement on $\widehat{\mathcal{A}}$. We introduce right-reversing by means of an algorithm working on (certain) signed $\mathcal{A}$-paths.

Algorithm 2.3 (Right-reversing, short case, negative–positive input).

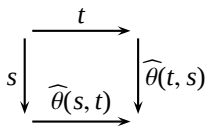
Context: A precategory $\mathcal{A}$, a short right-complement θ on $\mathcal{A}$
Input: A negative–positive $\mathcal{A}$-path $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$
Output: A positive–negative $\mathcal{A}$-path, or `fail`

$t_{i,0} := t_i$ for $i = 1, \ldots, q$
$s_{0,j} := s_j$ for $j = 1, \ldots, p$
for i increasing from 1 to q **do**
  for j increasing from 1 to p **do**
    if $\widehat{\theta}(s_{i-1,j}, t_{i,j-1})$ is defined **then**
      $t_{i,j} := \widehat{\theta}(s_{i-1,j}, t_{i,j-1})$
      $s_{i,j} := \widehat{\theta}(t_{i,j-1}, s_{i-1,j})$
    else
      return `fail`
return $s_{q,1}|\cdots|s_{q,p}|\overline{t_{q,p}}|\cdots|\overline{t_{1,p}}$

Lemma 2.4. *If θ is a short right-complement on a finite precategory $\mathcal{A}$, Algorithm 2.3 running on a pair of $\mathcal{A}$-paths of length at most ℓ terminates in $O(\ell^2)$ steps.*

Proof. The claim is obvious from the pseudocode in Algorithm 2.3. □

Running Algorithm 2.3 amounts to recursively constructing a rectangular grid whose edges are labeled by elements of $\mathcal{A}$ or empty paths, and in which each elementary square has the type shown aside, see Figure 3 for an example.



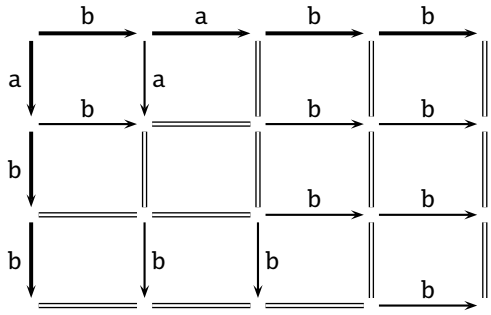


Figure 3: The grid associated with a right-reversing, short case. Here we consider the right-complement θ on $\{\mathsf{a}, \mathsf{b}\}$ defined by $\theta(\mathsf{a}, \mathsf{a}) = \theta(\mathsf{b}, \mathsf{b}) = \varepsilon$, $\theta(\mathsf{a}, \mathsf{b}) = \mathsf{b}$, and $\theta(\mathsf{b}, \mathsf{a}) = \mathsf{a}$ (with one object only), and apply Algorithm 2.3 to the negative–positive word $\overline{\mathsf{b}}|\overline{\mathsf{b}}|\overline{\mathsf{a}}|\mathsf{b}|\mathsf{a}|\mathsf{b}|\mathsf{b}$: the initial word is written on the left (negative part, here $\mathsf{a}|\mathsf{b}|\mathsf{b}$) and the top (positive part, here $\mathsf{b}|\mathsf{a}|\mathsf{b}|\mathsf{b}$), and then the grid is constructed by using the right-complement $\widehat{\theta}$ to recursively complete the squares; we use a double line for ε-labeled arrows. In the current case, the output of the algorithm is the length one word b.

Example 2.5. Recall Example 1.11. We consider a (short) right-complement θ on $\mathcal{A}$, where for s, t in $\mathcal{A}$ we define $\theta(s, t)$ as the entry in row s and column t of the table on the right.

	a	b	e
a	ε	b	e
b	a	ε	e
e	b	a	ε

Right-reversing the negative-positive word $\overline{\mathsf{a}}|\overline{\mathsf{b}}|\overline{\mathsf{e}}|\mathsf{a}|\mathsf{b}|\mathsf{e}|\mathsf{a}$ yields the word a as shown in Figure 4.

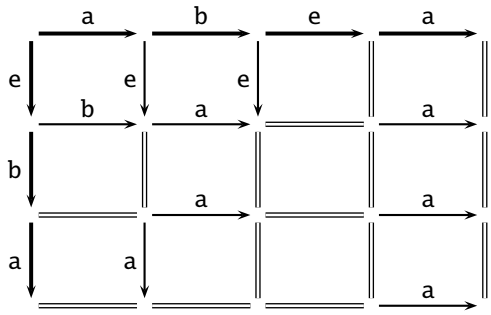


Figure 4: The grid associated with the right-reversing in Example 2.5.

We now turn to the case of an arbitrary right-complement, that is, we no longer assume that $\theta(s,t)$ necessarily has length zero or one. Then we can extend the definition of right-reversing, the only differences being that the constructed grid may involve rectangles whose edges contain more than one arrow when $\theta(s,t)$ has length at least 2. For the description, it is convenient to start from an arbitrary signed path, and not necessarily from a negative–positive one.

Algorithm 2.6 (Right-reversing, general case).

Context: A precategory $\mathcal{A}$, a right-complement θ on $\mathcal{A}$
Input: A signed $\mathcal{A}$-path w
Output: A positive–negative $\mathcal{A}$-path, or `fail`, or no output

while $\exists i < \|w\|$ $(w[i] \in \overline{\mathcal{A}}$ and $w[i+1] \in \mathcal{A})$ **do**
  $j := \min\{i \mid w[i] \in \overline{\mathcal{A}}$ and $w[i+1] \in \mathcal{A}\}$
  $s := \overline{w[j]}$
  $t := w[j+1]$
  if $\widehat{\theta}(s,t)$ is defined **then**
    replace $\overline{s}|t$ in w with $\widehat{\theta}(s,t)|\overline{\widehat{\theta}(t,s)}$
  else
    return `fail`
return w

Definition 2.7. If Algorithm 2.6 terminates successfully, we say that the initial signed $\mathcal{A}$-path w is *right-θ-reversible* to the final path w', and we write $w \curvearrowright_\theta w'$. If u, v are positive $\mathcal{A}$-paths, and there exist positive paths u', v' satisfying $\overline{u}|v \curvearrowright_\theta v'|\overline{u'}$, we define $\theta^*(u,v)$ to be v' and $\theta^*(v,u)$ to be u'.

As in the short case, running Algorithm 2.6 amounts to recursively constructing a grid whose edges are labeled by elements of $\mathcal{A}$ or empty paths, and in which each elementary square has the type shown aside. The difference to Algorithm 2.3 is twofold:

$$\begin{tikzcd} \cdot \arrow[r, "t"] \arrow[d, "s"'] & \cdot \arrow[d, "\widehat{\theta}(t,s)"] \\ \cdot \arrow[r, "\widehat{\theta}(s,t)"'] & \cdot \end{tikzcd}$$

Firstly, if the input is an arbitrary signed path, we do not necessary start with a vertical–horizontal path, but possibly with a staircase in which vertical and horizontal edges alternate; see Figure 5 for an example.

Secondly, if the right-complement θ is not short, the edges of the grid may have different sizes; see Figure 6 for an example.

Remark 2.8. More general versions of right-reversing are possible: at the expense of renouncing to determinism, we can consider multiform right-complements assigning with every pair of letters (s,t) a family of pairs of paths $\{(u_1,v_1),\ldots,(u_n,v_n)\}$ and decide that $\overline{s}|t$ may reverse to any of the paths $v_1|\overline{u_1},\ldots,v_n|\overline{u_n}$ (see Dehornoy, 2003). In such a context, several reversing grids may be associated with an initial path. Although most theoretical results can be adapted, these extended versions are less suitable for algorithms, and we shall not consider them here.

Termination of reversing

It should be clear that, whereas Algorithm 2.3 always terminates (successfully or not, that is, with an output path or with the output "`fail`") in finitely many steps, Algorithm 2.6 may not terminate.

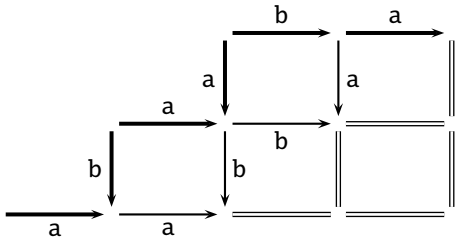


Figure 5: The grid associated with the right-reversing of a signed path. We consider the right-complement from Figure 3, and apply Algorithm 2.6 to the word $\mathtt{a}|\overline{\mathtt{b}}|\mathtt{a}|\overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}$: the initial word is written on the top-left boundary (negative edges vertically and positive edges horizontally), and completing the grid to the bottom-right yields the output word $\mathtt{a}|\mathtt{a}$.

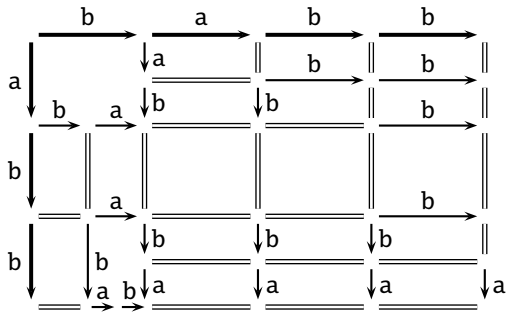


Figure 6: The grid associated with a right-reversing for a right-complement that is not short. Here we consider the right-complement θ on $\{\mathtt{a},\mathtt{b}\}$ defined by $\theta(\mathtt{a},\mathtt{a}) = \theta(\mathtt{b},\mathtt{b}) = \varepsilon$, $\theta(\mathtt{a},\mathtt{b}) = \mathtt{b}|\mathtt{a}$, and $\theta(\mathtt{b},\mathtt{a}) = \mathtt{a}|\mathtt{b}$ (with one object only), and apply Algorithm 2.6 to the negative–positive word $\overline{\mathtt{b}}|\overline{\mathtt{b}}|\overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}|\mathtt{b}|\mathtt{b}$ (the same as in Figure 3): the difference is that, now, edges of variable size occur, so that, a priori, the process need not terminate. In the current case, it terminates, and the output word is $\mathtt{a}|\mathtt{b}|\overline{\mathtt{a}}$.

Example 2.9. Consider the right-complement θ defined on $\{\mathtt{a},\mathtt{b}\}$ by $\theta(\mathtt{a},\mathtt{b}) = \mathtt{b}$ and $\theta(\mathtt{b},\mathtt{a}) = \mathtt{b}|\mathtt{a}$. Let $w = \overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}$. Then w reverses in two steps to $\mathtt{b}|w|\overline{\mathtt{b}}$, hence in $2n$ steps to $\mathtt{b}^n|w|\overline{\mathtt{b}}^n$ for every n, never leading to a positive–negative word.

For our current approach, it will be useful to have a simple termination criterion.

Lemma 2.10. *Assume that θ is a right-complement on a precategory $\mathcal{A}$.*

(i) *Right-θ-reversing terminates successfully for all valid inputs if and only if there exists a family $\mathcal{B}$ of $\mathcal{A}$-paths that includes $\mathcal{A}$ and is such that, for all u, v in $\mathcal{B}$ with the same source, there exist u', v' in $\mathcal{B}$ satisfying $\overline{u}|v \curvearrowright_\theta v'|\overline{u'}$.*

(ii) *If a family $\mathcal{B}$ with the properties as in* (i) *exists and is finite, then, for every signed $\mathcal{A}$-path of length ℓ, the right-θ-reversing of w terminates (successfully) in $O(\ell^2)$ steps and all involved paths have length in $O(\ell)$.*

Proof. (i) If right-θ-reversing is terminating, then $\mathcal{A}^*$ has the expected property.

Conversely, assume that $\mathcal{B}$ satisfies the property of the lemma. Let θ' be the restriction of θ^* to $\mathcal{B}$. Then θ' is a short right-complement on $\mathcal{B}$, so right-θ'-reversing is terminating. Now, assume

that $\overline{u}|v$ is a signed $\mathcal{A}$-path. By assumption, $\overline{u}|v$ is also a signed $\mathcal{B}$-path, and its right-θ'-reversing terminates, so there exists a witnessing θ'-grid. Now a θ'-grid of size $p \times q$ is the juxtaposition of pq θ-grids, whose existence shows that the right-θ-reversing of $\overline{u}|v$ also terminates.

(ii) Right-θ'-reversing terminates in $O(\ell^2)$ steps and all involved $\mathcal{B}$-paths have length in $O(\ell^2)$. As the family $\mathcal{B}$ is finite, there exists a constant that bounds the length of any element of $\mathcal{B}$ considered as an $\mathcal{A}$-path, and the claim follows. □

Corollary 2.11. *If θ is a short right-complement on a finite precategory $\mathcal{A}$, Algorithm 2.6 running on a signed $\mathcal{A}$-path of length ℓ terminates in $O(\ell^2)$ steps.*

Proof. If θ is short, the condition of Lemma 2.10 is satisfied with $\mathcal{B}$ equal to the family $\widehat{\mathcal{A}}$ in Notation 2.2, that is, the union of $\mathcal{A}$ and the empty paths. □

So, starting with an arbitrary right-complement θ, we can possibly show that right-θ-reversing terminates successfully for all valid inputs by applying the following closure method:

Algorithm 2.12 (Termination of right-reversing).

Context: A precategory $\mathcal{A}$
Input: A right-complement θ on $\mathcal{A}$
Output: A (minimal) subfamily of $\mathcal{A}^*$ that includes $\mathcal{A}$ and is closed under θ^*

```
1: A_0 := A
2: repeat
3:     if θ*(s, t) is defined for all s, t ∈ A_i then        ▹ right-reversing may fail to terminate
4:         A_{i+1} := A_i ∪ {θ*(s, t) | s, t ∈ A_i}
5:     else
6:         return fail
7: until A_{i+1} = A_i
8: return A_i
```

Example 2.13. Consider the right-complement of Figure 6 again. Starting with $\mathcal{A}_0 = \{\mathtt{a}, \mathtt{b}\}$ and applying Algorithm 2.12, we find $\mathcal{A}_1 = \mathcal{A}_0 \cup \{\varepsilon, \mathtt{a}|\mathtt{b}, \mathtt{b}|\mathtt{a}\}$, and $\mathcal{A}_2 = \mathcal{A}_1$: here the process terminates in one step, leading to $\{\varepsilon, \mathtt{a}, \mathtt{b}, \mathtt{a}|\mathtt{b}, \mathtt{b}|\mathtt{a}\}$. The existence of this 5 element family that is closed under right-reversing implies that right-reversing is terminating with a quadratic complexity upper bound.

When the closure under θ^* is infinite, the situation is more complicated and there is no general result. Examples are known when right-reversing is always terminating but the time (*resp*. space) complexity is more than quadratic (*resp*. linear): for instance, the right-reversing associated with the right-complement θ defined on $\{\mathtt{a}, \mathtt{b}\}$ by $\theta(\mathtt{a}, \mathtt{b}) = \varepsilon$ and $\theta(\mathtt{b}, \mathtt{a}) = \mathtt{a}(\mathtt{bab})^3\mathtt{ab}$ is always terminating, but the time (*resp*. space) complexity is cubic (*resp*. quadratic) (Dehornoy, 2012, Example 10.3), whereas Dehornoy (2000, Proposition VIII.1.11) displays an example (with an infinite family of generators) where right-reversing is terminating but the only known bound for time complexity is a tower of exponentials of exponential height.

2.2. Establishing left-cancellativity

We now address our main problem, namely investigating a presented category $\langle \mathcal{A} \,|\, \mathcal{R} \rangle^+$ and, in particular, trying to recognize whether it is left-cancellative. Here we consider the problem for presentations of a certain syntactical type. This restriction allows for using right-reversing, and

it is natural in our context as one can show that every Garside family gives rise to a presentation that is eligible for this approach (at least in the extended version alluded to in Remark 2.8).

Definition 2.14. A presentation $(\mathcal{A}; \mathcal{R})$ is called *positive* if all relations of $\mathcal{R}$ are of the form $u = v$ with u, v nonempty; it is called *right-complemented*, associated with the right-complement θ, if $\mathcal{R}$ consists of all relations $s|\theta(s,t) = t|\theta(t,s)$ with (s,t) in the domain of θ.

Note that, by definition, a right-complemented presentation is positive and that, if $(\mathcal{A}; \mathcal{R})$ is a positive presentation, then the category $\langle \mathcal{A} \,|\, \mathcal{R} \rangle^+$ contains no nontrivial invertible element since an empty path and a nonempty path cannot be $\mathcal{R}$-equivalent. Saying that a presentation $(\mathcal{A}; \mathcal{R})$ is right-complemented just means that it is positive and that, for all s, t in $\mathcal{A}$, the family $\mathcal{R}$ contains at most one relation of the form $s... = t...$. The involved right-complement is short if the paths "..." have length 0 or 1, that is, all relations in $\mathcal{R}$ are of the form $u = v$ with u and v of length 1 or 2. For instance, the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{ab} = \mathtt{ba})$ is associated with the short right-complement of Figure 3, whereas the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{aba} = \mathtt{bab})$ is associated with the right-complement of Figure 6 and Example 2.13. By contrast, the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{ab} = \mathtt{ba}, \mathtt{a}^2 = \mathtt{b}^2)$ is not right-complemented since it contains two relations of the form $\mathtt{a}... = \mathtt{b}...$.

When a presentation is right-complemented, it is eligible for the associated right-reversing transformation, leading in good cases to a practical method for recognizing left-cancellativity. The first observation is that right-reversing gives a way to construct common right-multiples in the associated category:

Lemma 2.15. *Assume that $(\mathcal{A}; \mathcal{R})$ is a presentation associated with a right-complement θ. Then, for all $\mathcal{A}$-paths u, v, u', v' satisfying $\overline{u}|v \curvearrowright_\theta v'|\overline{u'}$, the paths $u|v'$ and $v|u'$ are $\mathcal{R}$-equivalent.*

Proof. By definition, each elementary square in the rectangular grid which is associated with Algorithm 2.6 corresponds to a relation of $\mathcal{R}$. □

Lemma 2.15 says in particular that, if u, v are $\mathcal{A}$-paths and $\overline{u}|v$ is right-θ-reversible to an empty path, then u and v are $\mathcal{R}$-equivalent, that is, they represent the same element in the category $\langle \mathcal{A} \,|\, \mathcal{R} \rangle^+$. In our context, right-reversing will be useful only when the previous implication is an equivalence.

Definition 2.16. If $(\mathcal{A}; \mathcal{R})$ is a presentation associated with a right-complement θ, we say that right-reversing is *complete* for $(\mathcal{A}; \mathcal{R})$ if $\overline{u}|v \curvearrowright_\theta \varepsilon$ holds whenever u and v represent the same element in $\langle \mathcal{A} \,|\, \mathcal{R} \rangle^+$.

In other words, right-reversing is complete if it always detects equivalence. The interest of introducing completeness here is the following easy result:

Lemma 2.17. *Assume that $(\mathcal{A}; \mathcal{R})$ is a presentation associated with a right-complement θ and right-reversing is complete for $(\mathcal{A}; \mathcal{R})$. Then the category $\langle \mathcal{A} \,|\, \mathcal{R} \rangle^+$ is left-cancellative.*

Proof. It is sufficient to prove that, if s belongs to $\mathcal{A}$ and u, v are $\mathcal{A}$-paths such that $s|u$ and $s|v$ are $\mathcal{R}$-equivalent, then u and v are $\mathcal{R}$-equivalent. Now, as right-reversing is complete, the hypothesis implies that $\overline{u}|\overline{s}|s|v$ right-θ-reverses to an empty path. Now the first step in the reversing process necessarily consists in deleting $\overline{s}|s$. We deduce that $\overline{u}|v$ must right-θ-reverse to an empty path, which, by Lemma 2.15, implies that u and v are $\mathcal{R}$-equivalent. □

We are thus led to look for completeness criteria.

Definition 2.18. Assume that θ is a right-complement on a precategory $\mathcal{A}$. For r, s, t in $\mathcal{A}$, we say that θ satisfies the *cube condition* at (r, s, t) if either neither of $\theta^*(\theta^*(r, s), \theta^*(r, t))$ and $\theta^*(\theta^*(s, r), \theta^*(s, t))$ is defined, or both are defined and

$$\theta^*(\theta^*(\theta^*(r, s), \theta^*(r, t)), \theta^*(\theta^*(s, r), \theta^*(s, t)))$$

is empty.

Proposition 2.19. *(Dehornoy et al., 2013a, Proposition II.4.11) Assume that $(\mathcal{A}; \mathcal{R})$ is a presentation associated with a right-complement θ and that at least one of the following holds:*

(i) *the right-complement θ is short*

(ii) *the presentation $(\mathcal{A}; \mathcal{R})$ is right-Noetherian.*

Then right-reversing is complete for $(\mathcal{A}; \mathcal{R})$ if and only if, for all pairwise distinct s, t, r in $\mathcal{A}$ with the same source, the cube condition for (r, s, t) is satisfied.

Corollary 2.20. *If the equivalent conditions of Proposition 2.19 are satisfied, then the category $\langle \mathcal{A} \mid \mathcal{R} \rangle^+$ is left-cancellative.*

Example 2.21. Both presentations $(\mathtt{a}, \mathtt{b}; \mathtt{ab} = \mathtt{ba})$ and $(\mathtt{a}, \mathtt{b}; \mathtt{aba} = \mathtt{bab})$ are eligible for Proposition 2.19 and Corollary 2.20. Indeed, the former is associated with a short right-complement, whereas the latter is associated with a right-complement that is not short, but defining $\lambda(w)$ to be the length of w and noting that the (unique) relation of the presentation consists of two words with the same length, Lemma 1.6 yields that the presentation is right-Noetherian. Then, in order to apply Proposition 2.19 and Corollary 2.20, we need to check that the cube condition is satisfied for all triples of pairwise distinct elements of $\{\mathtt{a}, \mathtt{b}\}$, which is vacuously true. We conclude that both monoids $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{ab} = \mathtt{ba} \rangle^+$ (the free Abelian monoid $\mathbb{N}^2$) and $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{aba} = \mathtt{bab} \rangle^+$ (the 3-strand braid monoid B_3^+) are left-cancellative.

Example 2.22. Recall Example 2.5 and the short right-complement θ defined there. It is routine to check that the cube condition is satisfied for all r, s, t in $\mathcal{A}$. Thus, by Proposition 2.19 and Corollary 2.20, the category $\langle \mathcal{A} \mid \mathcal{R} \rangle^+$ defined by the presentation $(\mathcal{A}; \mathcal{R})$, where $\mathcal{R}$ is the set of relations given by θ, is left-cancellative. Note, however, that $\langle \mathcal{A} \mid \mathcal{R} \rangle^+$ is not isomorphic to the wreathed free Abelian monoid $\mathcal{T}$, as the relation $\mathtt{e}|\mathtt{e} = 1_T$ which holds in $\mathcal{T}$ is not described by θ, and hence not included in $\mathcal{R}$. In fact, $\mathcal{T}$ is the quotient of $\langle \mathcal{A} \mid \mathcal{R} \rangle^+$ by the relation $\mathtt{e}|\mathtt{e} = 1$.

Remark 2.23. Proposition 2.19 does not exhaust all known types of presentations for which right-reversing is complete. For instance, the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{a} = \mathtt{bab})$ of the Klein bottle monoid is not eligible for Proposition 2.19, since the associated right-complement θ is neither short nor right-Noetherian since no map λ may satisfy $\lambda(\mathtt{a}) = \lambda(\mathtt{bab}) > \lambda(\mathtt{ab})$. However right-reversing is complete and terminating for this presentation, as well as for a number of similar presentations (Dehornoy, 2012).

2.3. Recognizing Garside

Assuming that $(\mathcal{A}; \mathcal{R})$ is a right-complemented presentation and the category $\langle \mathcal{A} \mid \mathcal{R} \rangle^+$ has been shown to be left-cancellative using the method explained in Subsection 2.2, our next task is to recognize that some subfamily is possibly a Garside family. Once again the task will turn out to be easy whenever right-reversing is complete for the considered presentation.

The main observation is that, in the above context, the category $\langle \mathcal{A} \mid \mathcal{R} \rangle^+$ admits local right-lcms, that is, any two elements that admit a common right-multiple admit a right-lcm.

Lemma 2.24. *Assume that $(\mathcal{A};\mathcal{R})$ is a presentation associated with a right-complement θ and right-reversing is complete for it. Then, for all $\mathcal{A}$-paths u, v with the same source, the elements $[u], [v]$ admit a common right-multiple in $\langle \mathcal{A} \,|\, \mathcal{R}\rangle^+$ if and only if the right-reversing of $\overline{u}|v$ terminates, in which case we have $[u] \vee [v] = [u|\theta^*(u,v)]$ and $[u] \setminus [v] = [\theta^*(u,v)]$.*

Proof. Assume that $[u]$ and $[v]$ admit a common right-multiple in the category $\langle \mathcal{A} \,|\, \mathcal{R}\rangle^+$. This means that there exist $\mathcal{A}$-paths u', v' such that $u|v'$ and $v|u'$ are $\mathcal{R}$-equivalent. As right-reversing is complete, this implies that $\overline{v'}|\overline{u}|v|u'$ right-reverses to an empty path. Decompose the associated reversing grid as shown aside. The assumption that the reversing of $\overline{v'}|\overline{u}|v|u'$ successfully terminates implies in particular that the reversing of $\overline{u}|v$ successfully terminates, that is, the paths $\theta^*(u,v)$ and $\theta^*(v,u)$ are defined.

$$\begin{tikzcd}
\cdot \arrow[r, "v"] \arrow[d, "u"'] & \cdot \arrow[r, "u'"] \arrow[d, "\theta^*(v,u)"] & \cdot \arrow[d, equal] \\
\cdot \arrow[r, "\theta^*(u,v)"] \arrow[d, "v'"'] & \cdot \arrow[r] \arrow[d] & \cdot \arrow[d, equal] \\
\cdot \arrow[r, equal] & \cdot \arrow[r, equal] & \cdot
\end{tikzcd}$$

The diagram then shows that $[u|v']$ is a right-multiple of $[u|\theta^*(u,v)]$ in $\langle \mathcal{A} \,|\, \mathcal{R}\rangle^+$, so the latter, which depends only on $[u]$ and $[v]$, is a right-lcm of these elements. □

Applying the criterion of Lemma 1.14, we immediately deduce a method for recognizing Garside families in the right-Noetherian case. Actually, we obtain more: instead of just a YES/NO answer for a candidate-subfamily, we obtain the existence and a characterization of the smallest Garside family that includes the given family. Hereafter, for $\mathcal{B} \subseteq \mathcal{A}^*$, we write $[\mathcal{B}]$ for $\{[w] \mid w \in \mathcal{B}\}$.

Algorithm 2.25 (Smallest Garside family).

Context: A right-Noetherian right-complemented presentation $(\mathcal{A};\mathcal{R})$ for which right-reversing is complete

Input: A finite subfamily $\mathcal{B}$ of $\mathcal{A}^*$ that includes $\mathcal{A}$

Output: A subfamily of $\mathcal{A}^*$ that represents the smallest Garside family of $\langle \mathcal{A} \,|\, \mathcal{R}\rangle^+$ that includes $\mathcal{B} \cup \mathbf{1}_{\mathcal{A}}$ if the algorithm terminates successfully

```
enumerate B as {w_1, ..., w_n} and set B̂ := [w_1, ..., w_n]
i := 2
while i ⩽ |B̂| do
    for j := 1 to i − 1 do
        if θ*(w_i, w_j) is defined then          ▹ right-reversing may fail to terminate
            Include(B̂, θ*(w_i, w_j))
            Include(B̂, θ*(w_j, w_i))
            Include(B̂, w_i|θ*(w_i, w_j))
            Include(B̂, w_j|θ*(w_j, w_i))
        else
            return fail
    i := i + 1
return B̂ ∪ 1_A

procedure Include(B̂, w)          ▹ append w to B̂, unless B̂ contains a path equivalent to w.
    if ∄i ∈ {1, ..., |B̂|} with θ*(w, w_i) = θ*(w_i, w) = ε_− then
        append w to B̂
```

Proposition 2.26. *Assume that $(\mathcal{A}; \mathcal{R})$ is a finite right-Noetherian right-complemented presentation for which right-reversing is complete and $\mathcal{B}$ is a finite subfamily of $\mathcal{A}^*$ that includes $\mathcal{A}$. Then there exists a smallest finite Garside family of C including $[\mathcal{B}] \cup \mathbf{1}_C$ if and only if Algorithm 2.25 successfully terminates, in which case the returned family of paths represents that Garside family.*

Proof. By Lemma 2.24, any two elements of $\langle \mathcal{A} \,|\, \mathcal{R} \rangle^+$ that admit a common right-multiple admit a right-lcm, so $\langle \mathcal{A} \,|\, \mathcal{R} \rangle^+$ is eligible for Lemma 1.14. Hence a subfamily $\mathcal{S}$ that includes $\mathcal{A} \cup \mathbf{1}_{\mathcal{A}}$ is a Garside family if and only if, for all s, t in $\mathcal{S}$ with a common right-multiple, $s \vee t$ and $s \setminus t$ belong to $\mathcal{S}$. So, by Lemma 2.24 again, a family $\widehat{\mathcal{B}}$ of $\mathcal{A}$-paths represents a Garside family if and only if, for all u, v in $\widehat{\mathcal{B}}$, the paths $u|\theta^*(u, v)$ and $\theta^*(u, v)$ are $\mathcal{R}$-equivalent to at least one element of $\widehat{\mathcal{B}}$. It follows that there exists a smallest Garside family that includes $[\mathcal{B}]$, namely the smallest family of words that includes $\mathcal{B}$ and is such that, for all u, v in $\widehat{\mathcal{B}}$, the paths $u|\theta^*(u, v)$ and $\theta^*(u, v)$ are $\mathcal{R}$-equivalent to elements of $\widehat{\mathcal{B}}$.

That smallest subfamily $\widehat{\mathcal{B}}$ is precisely what Algorithm 2.25 computes. Indeed, what the latter does is to consider systematically all pairs (w_i, w_j) and, for each of them, test whether $\theta^*(w_i, w_j)$ (lines 6 and 7) as well as $w_i|\theta^*(w_i, w_j)$ (lines 8 and 9) are $\mathcal{R}$-equivalent to some existing path w_i of the list and, if not, append the missing paths to the list. Note that, as right-reversing is complete, two paths w, w' are equivalent if and only if $\theta^*(w, w')$ and $\theta^*(w', w)$ exist and are empty (line 15). □

Example 2.27. Consider again the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{aba} = \mathtt{bab})$. Running Algorithm 2.25 on the family $\{\mathtt{a}, \mathtt{b}\}$ yields $\{\varepsilon, \mathtt{a}, \mathtt{b}, \mathtt{a}|\mathtt{b}, \mathtt{b}|\mathtt{a}, \mathtt{a}|\mathtt{b}|\mathtt{a}\}$, a family of words representing the well known smallest Garside family $\{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ of the braid monoid B_3^+ that includes 1; here and everywhere in the sequel, we use Δ for $\mathtt{aba}$.

Remark 2.28. If right-reversing is not always terminating, that is, if some elements of the considered category have no common right-multiple although they share the same source, Algorithm 2.25 may never terminate. Even in the case when right-reversing is always terminating, it might happen that Algorithm 2.25 does not terminate: for instance, in the case of $(\mathtt{a}, \mathtt{b}; \mathtt{ab}^2 = \mathtt{ba})$, starting from $\{\mathtt{a}, \mathtt{b}\}$, even Algorithm 2.25 does not terminate in finite time: indeed, we have then $\theta^*(\mathtt{a}, \mathtt{b}^n) = \mathtt{b}^{2n}$, and the family $\widehat{\{\mathtt{a}, \mathtt{b}\}}$ is the infinite family $\{\varepsilon, \mathtt{a}\} \cup \{\mathtt{b}^{2^n} \mid n \geqslant 0\}$. However, it can be shown (Dehornoy, 2002, Lemme 1.9) that, when the closure under $\setminus$ is finite, then the closure under $\setminus$ and $\vee$ is necessarily finite.

2.4. Further questions

In the previous subsections, we showed how to establish left-cancellativity and to recognize Garside families starting from a presentation. We now briefly address further relevant questions, namely recognizing Ore categories, establishing the existence of lcms, and recognizing strong and bounded Garside families.

Establishing right-cancellativity

Right-reversing is not suitable here, and no practical method is known for establishing right-cancellativity directly. However an obvious solution is to consider the opposite category and the opposite presentation, that is, to switch left and right everywhere, and apply the previous methods. Equivalently, we can work with the initial presentation and apply the symmetric counterpart of right-reversing, naturally called *left-reversing*: whereas right-reversing consists in iteratively

replacing subpaths of the form $\overline{s}|t$ with $\theta(s,t)|\overline{\theta(t,s)}$, left-reversing consists in replacing subpaths of the form $t|\overline{s}$ with $\overline{\widetilde{\theta}(s,t)}|\widetilde{\theta}(t,s)$ when $\widetilde{\theta}$ is a *left-complement* on the considered precategory $\mathcal{A}$, namely a partial map of $\mathcal{A}^2$ to $\mathcal{A}^*$ such that $\widetilde{\theta}(s,s)$ is empty for every s in $\mathcal{A}$ and that, if $\widetilde{\theta}(s,t)$ is defined, then s and t have the same target, $\widetilde{\theta}(t,s)$ is defined, and both $\widetilde{\theta}(s,t)|t$ and $\widetilde{\theta}(t,s)|s$ are defined and have the same source. In terms of diagrams, this corresponds to constructing a grid starting from the bottom and the right, instead of from the top and the left. Then the counterpart of Corollary 2.20 gives a criterion for establishing that the category $\langle\mathcal{A}\,|\,\mathcal{R}\rangle^+$ defined by a left-complemented presentation $(\mathcal{A};\mathcal{R})$ is right-cancellative.

Establishing the existence of common multiples

Here two different methods can be used. If $(\mathcal{A};\mathcal{R})$ is a presentation associated with a right-complement θ, then the existence of common right-multiples in $\langle\mathcal{A}\,|\,\mathcal{R}\rangle^+$ is directly connected with the termination of right-reversing since, as proved in Lemma 2.24, two elements $[u]$ and $[v]$ admit a common right-multiple (and even a right-lcm) in $\langle\mathcal{A}\,|\,\mathcal{R}\rangle^+$ if and only if the right-θ-reversing of $\overline{u}|v$ successfully terminates in finite time. We deduce the following sufficient condition:

Proposition 2.29. *If $(\mathcal{A};\mathcal{R})$ is a presentation associated with a right-complement θ and Algorithm 2.12 running on $(\mathcal{A};\mathcal{R})$ succeeds, any two elements of $\langle\mathcal{A}\,|\,\mathcal{R}\rangle^+$ with the same source admit a common right-multiple.*

Another approach can be used once a Garside family $\mathcal{S}$ is known. It is based on the following result, which reduces the existence of common multiples for arbitrary elements to the existence of common multiples inside the Garside family.

Proposition 2.30. *Assume that $\mathcal{S}$ is a Garside family in a left-cancellative category C. Then any two elements of C with the same source admit a common right-multiple if and only if any two elements of $\mathcal{S}$ with the same source admit one.*

Proof. Obviously the condition is necessary. On the other hand, assume that any two elements of $\mathcal{S}$ with the same source admit a common right-multiple. By Lemma 1.15, for all s,t in $\mathcal{S}^\sharp$ with the same source, there exist s',t' in $\mathcal{S}^\sharp$ satisfying $st'=ts'$. Now, consider arbitrary elements f,g of C with the same source. As $\mathcal{S}^\sharp$ generates C, there exist $s_1,\dots,s_p$ and $t_1,\dots,t_q$ in $\mathcal{S}^\sharp$ satisfying $f=s_1\cdots s_p$ and $g=t_1\cdots t_q$. Using the result above, one inductively constructs a $p\times q$ rectangular grid based on $s_1,\dots,s_p$ and $t_1,\dots,t_q$ with edges in $\mathcal{S}^\sharp$, and the diagonal of the grid (as well as any path from the top–left corner to the bottom–right corner) represents a common right-multiple of f and g. □

Note that the construction of a grid in the proof of Proposition 2.30 is directly reminiscent of a right-reversing process—more exactly, of its non-deterministic extension alluded to in Remark 2.8 as there is no uniqueness of the elements called s',t' in general.

It should be clear that Proposition 2.30 directly leads to an effective method for deciding the existence of common right-multiples in the case of a finite Garside family, see Example 2.33 below.

A symmetric argument is possible for common left-multiples. However, as the definition of a Garside family is not invariant under exchanging left and right, the result takes a different form. In particular, it only gives a sufficient condition that need not be necessary in general.

Proposition 2.31. *Assume $\mathcal{S}$ is a Garside family in a left-cancellative category C and, for all s,t in $\mathcal{S}^\sharp$ with the same target, there exist s',t' in $\mathcal{S}^\sharp$ satisfying $s't=t's$. Then any two elements of C with the same target admit a common left-multiple.*

The proof is symmetric to that of Proposition 2.30, the inductive step consisting now in constructing a rectangular grid starting from the bottom and the right. Again Proposition 2.31 leads to an effective method for deciding the existence of common left-multiples in the case of a finite Garside family, see Example 2.33.

Establishing the existence of lcms

As for the existence of right-lcms (their computation will be addressed in Subsection 4.4), we shall just mention an algorithmically important consequence of Lemma 1.15:

Proposition 2.32. *Assume that $\mathcal{S}$ is a Garside family in a left-cancellative category C. Then C admits right-lcms (resp. local right-lcms) if and only if any two elements s, t of $\mathcal{S}^\sharp$ with the same source (resp. that admit a common right-multiple in $\mathcal{S}^\sharp$) admit a right-lcm r inside $\mathcal{S}^\sharp$, this meaning that r belongs to $\mathcal{S}^\sharp$ and every common right-multiple of s and t that lies in $\mathcal{S}^\sharp$ is a right-multiple of r.*

Proof. By Lemma 1.15, a common right-multiple of two elements of $\mathcal{S}^\sharp$ must be a right-multiple of some common right-multiple that lies in $\mathcal{S}^\sharp$. So it is enough to consider right-multiples lying in $\mathcal{S}^\sharp$, and we deduce that two elements of $\mathcal{S}^\sharp$ admit a right-lcm in C if and only if they admit one inside $\mathcal{S}^\sharp$. Then (the proof of) Proposition 2.30 enables one to go from $\mathcal{S}^\sharp$ to products of elements of $\mathcal{S}^\sharp$, that is, to arbitrary elements of C. □

We can thus establish the possible existence of right-lcms by exclusively inspecting right-multiples inside the considered Garside family.

Recognizing strong and bounded Garside families

So far, we have obtained methods for possibly establishing that a category is left- or right-Ore. Deciding whether a given Garside family $\mathcal{S}$ is strong is then easy when $\mathcal{S}$ is finite and an effective method is available for deciding, for all f, g, h, whether h is a left-lcm of f and g. Note that, by the counterpart of Lemma 2.24, such a method exists whenever the presentation is associated with a left-complement for which left-reversing is complete. Finally, deciding whether a finite Garside family is bounded is easy, as it only requires testing divisibility relations.

Example 2.33. We continue with the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{aba} = \mathtt{bab})$ of the monoid B_3^+. In Example 2.27, we identified the Garside family $\mathcal{S} = \{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ (where we recall Δ is $\mathtt{aba}$). It is clear that Δ is a right-multiple of every element of $\mathcal{S}$. By Proposition 2.30, we conclude that any two elements of B_3^+ admit a common right-multiple. Next, we easily check the existence of right-lcms inside $\mathcal{S}$ and, by Proposition 2.32, conclude that B_3^+ admits right-lcms (alternatively one can invoke Lemma 2.24 here as the monoid contains no nontrivial invertible element).

Similarly, Δ is also a common left-multiple of each element of $\mathcal{S}^\sharp$, which is $\mathcal{S}$, so, by Proposition 2.31, we deduce that any two elements of B_3^+ admit a common left-multiple—whence a left-lcm by the symmetric counterpart of Lemma 2.24.

Finally, as every element of $\mathcal{S}$ is both a left- and a right-divisor of Δ, we deduce that the Garside family $\mathcal{S}$ is bounded by Δ. So, in particular, it is strong.

Example 2.34. Recall the wreathed free Abelian monoid $\mathcal{T}$ from Example 1.11. We will see in Example 3.21 that $\mathcal{S} = \{\mathtt{a}, \mathtt{b}, \Delta\}$, where $\Delta = \mathtt{ab} = \mathtt{ba}$, is a Garside family in $\mathcal{T}$.

Proposition 2.30 yields that any two elements of $\mathcal{T}$ admit a common right-multiple, as Δ is a right-multiple of every element of $\mathcal{S}$. Moreover, we have $\mathcal{S}^\sharp = \{1, \mathtt{a}, \mathtt{b}, \Delta, \mathtt{e}, \mathtt{ae}, \mathtt{be}, \Delta\mathtt{e}\}$, and

it is easily verified that for any s, t in $\mathcal{S}^\sharp$ there exist s', t' in $\mathcal{S}^\sharp$ satisfying $s't = t's$, as well as a right-lcm r of s and t inside $\mathcal{S}^\sharp$. Hence, any two elements of $\mathcal{T}$ admit a common left-multiple and a right-lcm by Proposition 2.31 respectively Proposition 2.32. As every element of $\mathcal{S}$ is both a left- and a right-divisor of Δ, the Garside family $\mathcal{S}$ is bounded by Δ, and thus strong.

3. Recognizing Garside families, case of a germ

We now consider the same questions as in Section 2, namely establishing left-cancellativity and recognizing Garside families, when the ambient category is specified by giving either the complete multiplication table (Subsection 3.1), or a germ, defined to be a fragment of the multiplication table that contains enough information to determine the latter unambiguously (Subsections 3.2 and 3.3). In this case, one obtains a compact method that enables one to treat both questions (left-cancellativity and Garside family) simultaneously.

3.1. Using the complete multiplication table

We first quickly consider the case of a finite category. Such a category can be specified by an exhaustive enumeration of its elements, its source and map functions, and its multiplication table, which all are finite data. All questions are then easy.

First, for a finite category with an explicit multiplication table, left-cancellativity can be decided by an exhaustive inspection. Next, the left-divisibility relation is finite, so checking the condition (1.9) is easy and, for every subfamily $\mathcal{S}$ of C, one can construct a list of all $\mathcal{S}$-greedy paths of length two. *A priori*, it is not clear that this is sufficient to recognize a Garside family, since the definition of the latter mentions no upper bound on the length of the considered decompositions. However, such a bound exists, which limits both the elements to be considered and the length of the candidate-decompositions.

Lemma 3.1. *(Dehornoy et al., 2013b, Proposition 3.1) A subfamily $\mathcal{S}$ of a left-cancellative category C is a Garside family if and only if $\mathcal{S}^\sharp$ generates C and every element of $(\mathcal{S}^\sharp)^2$ admits an $\mathcal{S}$-normal decomposition of length two.*

We deduce

Proposition 3.2. *Assume that C is a finite category C with n elements, and that the source and target maps on C and the multiplication table of C are given. Then the following hold:*

(i) *It is decidable in time $O(n^2)$ whether C is left-cancellative.*

(ii) *The left-divisibility relation can be computed in time $O(n^2)$.*

(iii) *If C is left-cancellative and $\mathcal{S}$ is a subfamily of C, then it is decidable in time $O(n^6)$ whether $\mathcal{S}$ is a Garside family in C.*

Proof. We can test whether C is left-cancellative and compute the left-divisibility relation as follows: For each f in C, iterate over all g in C, keeping track of the products fg that occur; C is left-cancellative if and only if no such product occurs more than once, and the occurring products are precisely the right-multiples of f. Hence (i) and (ii) hold.

Comparing fg to 1_x, where x is the source of f for all f, g in C is sufficient to determine $C^\times$, and then $\mathcal{S}^\sharp$ can be obtained by computing all products sg for s in $\mathcal{S}$ and g in $C^\times$. Thus computing $\mathcal{S}^\sharp$ takes time $O(n^2)$. To verify that $\mathcal{S}^\sharp$ generates C, one computes, for $i = 2, 3, ...$, the set $(\mathcal{S}^\sharp)^i$ by considering all products gs with g in $(\mathcal{S}^\sharp)^{i-1}$ and s in $\mathcal{S}^\sharp$ until this sequence stabilizes, which happens after at most n steps. Thus verifying that $\mathcal{S}^\sharp$ generates C takes time $O(n^3)$. Finally, by

Lemma 3.1, one can decide whether a subfamily $\mathcal{S}$ of C is a Garside family by checking for each element of $(\mathcal{S}^\sharp)^2$ all possible $\mathcal{S}$-paths of length two against condition (1.9); as each test of condition (1.9) involves $O(n^2)$ operations, this takes time $O(n^6)$, completing the proof of (iii). $\square$

Note that, if C is infinite, $\mathcal{S}$ being finite does not make recognizing $\mathcal{S}$-greediness decidable in general, as (1.9) contains a universal quantification over an arbitrary element f of C and, so, in that case, Lemma 3.1 is of no use.

Remark 3.3. The above analysis partly extends to the case when the category C is infinite but all operations are computable. For instance, if multiplication, viewed as a partial function of C^2 to C, is computable, then left-cancellativity is a Π^1_1 condition: by enumerating all triples (f, g, g') with $g \neq g'$ and checking whether fg and fg' are equal, one finds a counter-example in finite time if one exists, but one never obtains an answer when the category is left-cancellative. Similarly, left-divisibility is a Σ^1_1 condition and, therefore, $\mathcal{S}$-greediness, which entails an additional existential quantification over an arbitrary element f of C, is a Π^1_2 condition.

3.2. Germs

For an infinite category, it is impossible to exhaustively enumerate the multiplication table. However, in good cases, it may happen that some finite fragment of the latter determines the category and provides methods for establishing properties of the latter. This is the germ approach that we now introduce.

Definition 3.4. A *germ* is a triple $(\mathcal{S}, \mathbf{1}_\mathcal{S}, \bullet)$ where $\mathcal{S}$ is a precategory, $\mathbf{1}_\mathcal{S}$ is a subfamily of $\mathcal{S}$ consisting of an element 1_x with source and target x for each object x, and $\bullet$ is a partial map of $\mathcal{S}^{[2]}$ into $\mathcal{S}$ that satisfies

$$\text{if } s \bullet t \text{ is defined, its source is the source of } s \text{ and its target is the target of } t, \tag{3.5}$$

$$1_x \bullet s = s = s \bullet 1_y \text{ hold for each } s \text{ in } \mathcal{S}(x, y), \tag{3.6}$$

$$\text{if } r \bullet s \text{ and } s \bullet t \text{ are defined, then } (r \bullet s) \bullet t \text{ is defined if and only if } r \bullet (s \bullet t) \text{ is, in which case they are equal.} \tag{3.7}$$

The germ is called *left-associative* if, for all r, s, t in $\mathcal{S}$, it satisfies

$$\text{if } (r \bullet s) \bullet t \text{ is defined, then } s \bullet t \text{ is defined,} \tag{3.8}$$

and it is called *left-cancellative* if, for all s, t, t' in $\mathcal{S}$, it satisfies

$$\text{if } s \bullet t \text{ and } s \bullet t' \text{ are defined and equal, then } t = t' \text{ holds.} \tag{3.9}$$

We shall usually write $\underline{\mathcal{S}}$ for a germ whose domain is $\mathcal{S}$. Whenever $\mathcal{S}$ is a subfamily of a category C, one obtains a germ by considering the restriction of the product of C to $\mathcal{S}$, that is, the partial binary operation $\bullet$ on $\mathcal{S}$ defined by $s \bullet t = st$ whenever st is defined in C and it belongs to $\mathcal{S}$. In the other direction, starting with a germ, we can always construct a category.

Definition 3.10. If $\underline{\mathcal{S}}$ is a germ, we denote by $Cat(\underline{\mathcal{S}})$ the category $\langle \mathcal{S} \mid \mathcal{R}_\bullet \rangle^+$, where $\mathcal{R}_\bullet$ is the family of all relations $s|t = s \bullet t$ with s, t in $\mathcal{S}$ and $s \bullet t$ defined.

When we start with a category C and a subfamily $\mathcal{S}$ of C, it may or may not be the case that the induced germ on $\mathcal{S}$ contains enough information to reconstruct the initial category C.

Example 3.11. Let us consider the braid monoid B_3^+ again. If we take $\mathcal{A} = \{1, \mathtt{a}, \mathtt{b}\}$, the table of the induced germ is shown aside, and the derived monoid is the free monoid based on $\mathtt{a}, \mathtt{b}$, hence one that is not isomorphic to B_3^+.

$\bullet$	1	a	b
1	1	a	b
a	a		
b	b		

By contrast, when we start with $\mathcal{S} = \{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$, the table of the induced germ is less hollow, and the derived monoid is indeed isomorphic to B_3^+.

$\bullet$	1	a	b	ab	ba	Δ
1	1	a	b	ab	ba	Δ
a	a		ab		Δ	
b	b	ba		Δ		
ab	ab	Δ				
ba	ba		Δ			
Δ	Δ					

The good point is that, as in the above example, the germ induced by a Garside family always contains enough information to reconstruct the initial category:

Lemma 3.12. *(Dehornoy et al., 2013b, Proposition 4.8) If $\mathcal{S}$ is a Garside family in a left-cancellative category C and $\underline{\mathcal{S}}$ is the germ induced on $\mathcal{S}$, the category $\mathit{Cat}(\underline{\mathcal{S}})$ is isomorphic to C.*

3.3. Establishing left-cancellativity and recognizing Garside

Lemma 3.12 precisely shows that using a germ to specify a category is relevant here: if we start with a good candidate for a Garside family, the germ will indeed define the category.

Definition 3.13. A germ $\underline{\mathcal{S}}$ is said to be a *Garside germ* if $\mathcal{S}$ embeds in $\mathit{Cat}(\underline{\mathcal{S}})$, the latter is left-cancellative, and (the image of) $\mathcal{S}$ is a Garside family in that category.

So, for instance, the germs in Example 3.11 are Garside germs. By contrast, below is an example of a germ that is not a Garside germ although it defines the ambient monoid.

Example 3.14. Let $M = \langle \mathtt{a}, \mathtt{b} \mid \mathtt{ab} = \mathtt{ba}, \mathtt{a}^2 = \mathtt{b}^2 \rangle^+$, and $\mathcal{S}$ consist of $1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{a}^2$. The germ $\underline{\mathcal{S}}$ induced on $\mathcal{S}$ is shown aside. It is left-associative and left-cancellative. The category (here the monoid) $\mathit{Cat}(\underline{\mathcal{S}})$ is (isomorphic to) M, as the relations $\mathtt{a}|\mathtt{a} = \mathtt{a}^2 = \mathtt{b}|\mathtt{b}$ and $\mathtt{a}|\mathtt{b} = \mathtt{ab} = \mathtt{b}|\mathtt{a}$ belong to the family $\mathcal{R}_\bullet$. However $\mathcal{S}$ is not a Garside family in M, as $\mathtt{a}^3$ admits no $\mathcal{S}$-normal decomposition: $\mathtt{a}^2|\mathtt{a}$ is not $\mathcal{S}$-greedy as $\mathtt{ab}$ left-divides $\mathtt{a}^3$ but not $\mathtt{a}^2$, and $\mathtt{ab}|\mathtt{b}$ is not $\mathcal{S}$-greedy as $\mathtt{a}^2$ left-divides $\mathtt{a}^3$ but not $\mathtt{ab}$.

$\bullet$	1	a	b	$\mathtt{a}^2$	ab
1	1	a	b	$\mathtt{a}^2$	ab
a	a	$\mathtt{a}^2$	ab		
b	b	ab	$\mathtt{a}^2$		
$\mathtt{a}^2$	$\mathtt{a}^2$				
ab	ab				

What we do below is to develop algorithms to decide whether a (finite) germ is a Garside germ. Note that this includes establishing that the defined category is left-cancellative. Our method is based on a result by Dehornoy et al. (2013b).

Definition 3.15. Assume that $\underline{\mathcal{S}}$ is a germ.

(i) We define the *local left-divisibility* relation $\preccurlyeq_{\mathcal{S}}$ of $\underline{\mathcal{S}}$ by saying that $s \preccurlyeq_{\mathcal{S}} t$ holds if and only if there exists t' in $\mathcal{S}$ satisfying $t = st'$. We write $s \prec_{\mathcal{S}} t$ for the conjunction of $s \preccurlyeq_{\mathcal{S}} t$ and $t \not\preccurlyeq_{\mathcal{S}} s$, and we call a sequence $s_1, \ldots, s_n$ in $\mathcal{S}$ *non-ascending* if $s_i \not\prec_{\mathcal{S}} s_j$ holds for $1 \leqslant i < j \leqslant n$.

(ii) For $s_1|s_2$ in $\mathcal{S}^{[2]}$, we put $\mathcal{J}_{\mathcal{S}}(s_1, s_2) = \{t \in \mathcal{S} \mid s_1 \bullet t \text{ is defined and } t \preccurlyeq_{\mathcal{S}} s_2\}$.

Proposition 3.16. *(Dehornoy et al., 2013b, Proposition 5.9) A germ $\underline{\mathcal{S}}$ is a Garside germ if and only if it is left-associative, left-cancellative, and if, for any s_1, s_2 in $\mathcal{S}$ there exists a $\preccurlyeq_{\mathcal{S}}$-greatest element in $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$ (that is, an element r in $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$ such that $t \preccurlyeq_{\mathcal{S}} r$ holds for all t in $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$).*

Corollary 3.17. *If $\mathcal{S}$ is a Garside germ, if $s_1|s_2$ is in $\mathcal{S}^{[2]}$, if t is a $\preccurlyeq_{\mathcal{S}}$-greatest element in the set $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$, and if r is an element of $\mathcal{S}$ satisfying $s_2 = t \bullet r$, then the path $s_1 t|r$ is an $\mathcal{S}$-normal decomposition of $s_1 s_2$ in $\mathcal{C}at(\underline{\mathcal{S}})$.*

Algorithm 3.18 (Recognizing a Garside germ).

Input: A finite germ $\underline{\mathcal{S}}$
Output: `true` if $\underline{\mathcal{S}}$ is a Garside germ, and `false` otherwise

```
isLeftCancellative, ≼_S := LeftDivisibility(S)
if not isLeftCancellative or not IsLeftAssociative(S) then
    return false
S' := NonAscending(S, ≼_S)
for s1|s2 in S^[2] do
    if not JHasGreatestElement(S', s1, s2, ≼_S) then
        return false
return true

function IsLeftAssociative(S)
    for r|s|t in S^[3] do
        if (r • s) • t↓ and not s • t↓ then
            return false
    return true

function LeftDivisibility(S)    ▷ whether S is left-cancellative and, if it is, a table (denoted
    by ≼_S) with the truth values [[s ≼_S t]] of s ≼_S t for s, t in S
    for s ∈ S do
        for t ∈ S do
            [[s ≼_S t]] := false
        for t ∈ S do
            if s • t↓ then
                if [[s ≼_S s • t]] then
                    return false
                else
                    [[s ≼_S s • t]] := true
    return true, ≼_S

function NonAscending(S, ≼_S)                    ▷ S as a non-ascending sequence
    S' := [ ]
    for s ∈ S do
        if ∃i ∈ {1, ..., |S'|} with S'[i] ≺_S s then
            insert s into S at position min({i ∈ {1, ..., |S'|} | S'[i] ≺_S s})
        else
            append s to S'
    return S'
```

```
function JHasGreatestElement(𝒮′, s₁, s₂, ≼_𝒮)                    ▷ 𝒮′ non-ascending
    r := ⊥
    for i := 1 to |𝒮′| do
        if s₁ • 𝒮′[i]↓ and 𝒮′[i] ≼_𝒮 s₂ then                      ▷ 𝒮′[i] ∈ 𝒥_𝒮(s₁, s₂)
            if r = ⊥ then
                r := 𝒮′[i]
            else if not 𝒮′[i] ≼_𝒮 r then
                return false
    return true
```

Proposition 3.19. *Assume that $\underline{\mathcal{S}}$ is a finite germ with $|\mathcal{S}| = n$ and that the partial binary operation $\bullet$ can be computed in time $O(1)$. Then the following hold; cf. Algorithm 3.18.*

(i) *The function* `IsLeftAssociative` *decides in time $O(n^3)$ whether $\underline{\mathcal{S}}$ is left-associative.*

(ii) *The function* `LeftDivisibility` *decides whether $\underline{\mathcal{S}}$ is left-cancellative and, if it is, computes the left-divisibility relation $\preccurlyeq_{\mathcal{S}}$ on $\mathcal{S}$ with respect to $\bullet$ in time $O(n^2)$.*

(iii) *Given the left-divisibility relation $\preccurlyeq_{\mathcal{S}}$, the function* `NonAscending` *computes a non-ascending sequence containing the elements of $\mathcal{S}$ in time $O(n^2)$.*

(iv) *Given $s_1|s_2$ in $\mathcal{S}^{[2]}$, a non-ascending sequence $\mathcal{S}'$ containing the elements of $\mathcal{S}$, and the left-divisibility relation $\preccurlyeq_{\mathcal{S}}$, the function* `JHasGreatestElement` *decides in time $O(n)$ whether $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$ has a $\preccurlyeq_{\mathcal{S}}$-greatest element.*

(v) *It is decidable in time $O(n^3)$ whether $\mathcal{S}$ is a Garside germ.*

Proof. Claims (i) and (ii) are obvious from the pseudocode in Algorithm 3.18 and the definitions.

The sequence $\mathcal{S}'$ constructed in the function `NonAscending` is non-ascending at any time by induction: If there is no i satisfying $\mathcal{S}'[k] \prec_{\mathcal{S}} s$, the induction step is trivial. Otherwise, one has $\mathcal{S}'[j] \not\prec_{\mathcal{S}} s$ for all $j < i$ by the choice of i, and $s \not\prec_{\mathcal{S}} \mathcal{S}'[j]$ for $j \geqslant i$, as $\mathcal{S}'[k] \prec_{\mathcal{S}} s \prec_{\mathcal{S}} \mathcal{S}'[j]$ would contradict the induction hypothesis. It is clear from the pseudocode that the time complexity is $O(n^2)$, so claim (iii) holds.

The function `JHasGreatestElement` tests the elements of $\mathcal{S}$ for membership in $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$ in non-ascending order. Moreover, the set $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$ is non-empty, as it contains ε_x, where x is the target of s_1. Hence, the set $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$ has a $\preccurlyeq_{\mathcal{S}}$-greatest element if and only if the *first* encountered element is an upper bound; this is what the function `JHasGreatestElement` tests. It is clear from the pseudocode that the time complexity is $O(n)$, so claim (iv) holds.

Claim (v) then follows with Proposition 3.16. □

Example 3.20. We apply Algorithm 3.18 to the second germ of Example 3.11, that is, we have $\mathcal{S} = \{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ and $\bullet$ is the partial binary operation on $\mathcal{S}$ induced by the multiplication in B_3^+.

One readily verifies that the germ is left-associative and left-cancellative, and computes the left-divisibility relation $\preccurlyeq_{\mathcal{S}}$ which is given in the left table below.

We obtain $\mathcal{S}' = (\Delta, \mathtt{ab}, \mathtt{ba}, \mathtt{b}, \mathtt{a}, 1)$ and the non-ascending sequences describing the sets $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$ given in the right table. For each set $\mathcal{J}_{\mathcal{S}}(s_1, s_2)$, the first listed element is a maximum, showing that $\mathcal{S}$ is a Garside germ.

$\preccurlyeq_{\mathcal{S}}$	1	a	b	ab	ba	Δ
1	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$
a		$\preccurlyeq$		$\preccurlyeq$		$\preccurlyeq$
b			$\preccurlyeq$		$\preccurlyeq$	$\preccurlyeq$
ab				$\preccurlyeq$		$\preccurlyeq$
ba					$\preccurlyeq$	$\preccurlyeq$
Δ						$\preccurlyeq$

$\mathcal{J}_{\mathcal{S}}$	1	a	b	ab	ba	Δ
1	(1)	(a, 1)	(b, 1)	(ab, a, 1)	(ba, b, 1)	$\mathcal{S}'$
a	(1)	(1)	(b, 1)	(1)	(ba, b, 1)	(ba, b, 1)
b	(1)	(a, 1)	(1)	(ab, a, 1)	(1)	(ab, a, 1)
ab	(1)	(a, 1)	(1)	(a, 1)	(1)	(a, 1)
ba	(1)	(1)	(b, 1)	(1)	(b, 1)	(b, 1)
Δ	(1)	(1)	(1)	(1)	(1)	(1)

Example 3.21. Consider again the wreathed free Abelian monoid $\mathcal{T}$ from Example 1.11, the family $\mathcal{S} = \{\mathsf{a}, \mathsf{b}, \Delta\}$ in $\mathcal{T}$, where $\Delta = \mathsf{ab}$, and the germ $\underline{\mathcal{S}^\sharp}$ which is induced on the set $\mathcal{S}^\sharp = \{1, \mathsf{a}, \mathsf{b}, \Delta, \mathsf{e}, \mathsf{ae}, \mathsf{be}, \Delta\mathsf{e}\}$. The germ $\underline{\mathcal{S}^\sharp}$ is shown on the right.

$\bullet$	1	a	b	Δ	e	ae	be	Δe
1	1	a	b	Δ	e	ae	be	Δe
a	a		Δ		ae		Δe	
b	b	Δ			be	Δe		
Δ	Δ				Δe			
e	e	be	ae	Δe	1	b	a	Δ
ae	ae	Δe			a	Δ		
be	be		Δe		b		Δ	
Δe	Δe			Δ				

The category (here the monoid) $\mathit{Cat}(\underline{\mathcal{S}^\sharp})$ is (isomorphic to) $\mathcal{T}$, as the relations $\mathsf{a}|\mathsf{b} = \Delta = \mathsf{b}|\mathsf{a}$, $\mathsf{e}|\mathsf{a} = \mathsf{be} = \mathsf{b}|\mathsf{e}$ and $\mathsf{e}|\mathsf{e} = 1$ belong to the family $\mathcal{R}_\bullet$. We apply Algorithm 3.18.

One readily verifies that the germ is left-associative and left-cancellative, computes the left-divisibility relation $\preccurlyeq_{\mathcal{S}^\sharp}$ as well as the sets $\mathcal{J}_{\mathcal{S}^\sharp}(s_1, s_2)$ for s_1, s_2 in $\mathcal{S}^\sharp$, and verifies that each set $\mathcal{J}_{\mathcal{S}^\sharp}(s_1, s_2)$ has a maximal element. The left divisibility relation is given in the left table below. The entry in row s_1 and column s_2 of the right table shows a maximal element of $\mathcal{J}_{\mathcal{S}^\sharp}(s_1, s_2)$. (The maximal elements are only unique up to multiplication by the invertible element e.)

$\preccurlyeq_{\mathcal{S}^\sharp}$	1	a	b	Δ	e	ae	be	Δe
1	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$
a		$\preccurlyeq$		$\preccurlyeq$		$\preccurlyeq$		$\preccurlyeq$
b			$\preccurlyeq$	$\preccurlyeq$			$\preccurlyeq$	$\preccurlyeq$
Δ				$\preccurlyeq$				$\preccurlyeq$
e	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$	$\preccurlyeq$
ae		$\preccurlyeq$		$\preccurlyeq$		$\preccurlyeq$		$\preccurlyeq$
be			$\preccurlyeq$	$\preccurlyeq$			$\preccurlyeq$	$\preccurlyeq$
Δe				$\preccurlyeq$				$\preccurlyeq$

$\max \mathcal{J}_{\mathcal{S}^\sharp}$	1	a	b	Δ	e	ae	be	Δe
1	1	a	b	Δ	1	a	b	Δ
a	1	1	b	b	1	1	b	b
b	1	a	1	a	1	a	1	a
Δ	1	1	1	1	1	1	1	1
e	1	a	b	Δ	1	a	b	Δ
ae	1	a	1	a	1	a	1	a
be	1	1	b	b	1	1	b	b
Δe	1	1	1	1	1	1	1	1

In particular, $\underline{\mathcal{S}^\sharp}$ is a Garside germ, that is, $\mathcal{S}^\sharp$ is a Garside family in $\mathcal{T}$. By Lemma 3.1, $\mathcal{S}$ is a Garside family if and only if $\mathcal{S}^\sharp$ is, so we can conclude that $\mathcal{S} = \{\mathsf{a}, \mathsf{b}, \Delta\}$ is a Garside family in $\mathcal{T}$.

3.4. Further questions

From that point, results are similar to those of Subsection 2.4: once a germ has been shown to be a Garside germ, it is known to be a Garside family in its ambient category, and, for instance, Propositions 2.30 and 2.31 directly apply for the existence of common multiples. The only point worth mentioning here is right-cancellativity: let us observe that a germ is a symmetric structure, so applying the method of Subsection 3.3 to the opposite of the considered germ $\underline{\mathcal{S}}$, that is, $(\mathcal{S}, \tilde{\bullet})$ with $\tilde{\bullet}$ defined by $s \,\tilde{\bullet}\, t = t \bullet s$, directly leads to a right-cancellativity criterion.

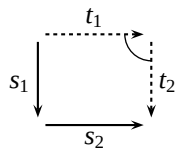
4. Computing with a Garside family, positive case

We now turn to a different series of questions: here we no longer aim at deciding whether the ambient category and a candidate family are relevant for the Garside approach, but we assume that they are and we aim at computing in the context so obtained. In this section, we consider computations that take place inside the considered category, and postpone to the next section the computations that take place in the groupoid of fractions. We shall successively consider computing normal decompositions (Subsections 4.1 and 4.2), solving the Word Problem (Subsection 4.3) and computing least common multiples (Subsection 4.4).

4.1. Computing normal decompositions from the left

By definition, a Garside family gives rise to distinguished decompositions for the elements of the ambient category, namely $\mathcal{S}$-normal decompositions. The first natural algorithmic question is to determine an $\mathcal{S}$-normal decomposition of an element g specified by an arbitrary decomposition in terms of $\mathcal{S}^\sharp$. So the question is, starting from an $\mathcal{S}^\sharp$-path w, to find an equivalent $\mathcal{S}$-normal path. Here we shall use an incremental approach, starting from a solution in the case of length two paths.

Definition 4.1. A subfamily $\mathcal{A}$ of a left-cancellative category $\mathcal{C}$ *satisfies Property* $\square$ if, for every $s_1|s_2$ in $\mathcal{A}^{[2]}$, there exists an $\mathcal{A}$-greedy decomposition $t_1|t_2$ of s_1s_2 with t_1 and t_2 in $\mathcal{A}$. In this case, a map W that chooses, for every $s_1|s_2$ in $\mathcal{A}^{[2]}$, a pair (t_1, t_2) as above is called a $\square$-*witness* on $\mathcal{A}$.



If $\mathcal{S}$ is a Garside family, then one easily shows that there exists a $\square$-witness on $\mathcal{S}^\sharp$, that is, every element of $(\mathcal{S}^\sharp)^2$ admits an $\mathcal{S}$-normal decomposition of length 2. Moreover, if the ambient category admits no nontrivial invertible element, then, by Proposition 1.12, the $\square$-witness on $\mathcal{S}^\sharp$ is unique. Note that, if the Garside family was initially specified as a germ, then Corollary 3.17 directly provides a $\square$-witness; see Table 1 for an example.

	ε	a	b	ab	ba	Δ
ε	(1, 1)	(a, 1)	(b, 1)	(ab, 1)	(ba, 1)	(Δ, 1)
a	(a, 1)	(a, a)	(ab, 1)	(a, ab)	(Δ, 1)	(Δ, b)
b	(b, 1)	(ba, 1)	(b, b)	(Δ, 1)	(b, ba)	(Δ, a)
ab	(ab, 1)	(Δ)	(ab, b)	(Δ, b)	(ab, ba)	(Δ, ba)
ba	(ba, 1)	(ba, a)	(Δ, 1)	(ba, ab)	(Δ, a)	(Δ, ab)
Δ	(Δ, 1)	(Δ, a)	(Δ, b)	(Δ, ab)	(Δ, ba)	(Δ, Δ)

Table 1: The $\square$-witness on the Garside family $\{1, \mathrm{a}, \mathrm{b}, \mathrm{ab}, \mathrm{ba}, \Delta\}$ in the braid monoid B_3^+: for every pair of elements (s_1, s_2), it specifies the unique normal decomposition of s_1s_2 of length 2. The values directly follow from considering $\mathcal{S}$ as a germ and using the maximal $\mathcal{J}$-function of Example 3.20.

Example 4.2. Consider the Garside germ $\underline{\mathcal{S}^\sharp}$ for the wreathed free Abelian monoid $\mathcal{T}$ from Example 3.21. By the calculations done there, a $\square$-witness on $\mathcal{S}^\sharp$ is given by Table 2.

Algorithm 4.3 (Left-multiplication—see Figure 7).

Context: A Garside family $\mathcal{S}$ in a left-cancellative category $\mathcal{C}$, a $\square$-witness W for $\mathcal{S}^\sharp$

	1	a	b	Δ	e	ae	be	Δe
1	(1, 1)	(a, 1)	(b, 1)	(Δ, 1)	(1, e)	(a, e)	(b, e)	(Δ, e)
a	(a, 1)	(a, a)	(Δ, 1)	(Δ, a)	(a, e)	(a, ae)	(Δ, e)	(Δ, ae)
b	(b, 1)	(Δ, 1)	(b, b)	(Δ, b)	(b, e)	(Δ, e)	(b, be)	(Δ, be)
Δ	(Δ, 1)	(Δ, a)	(Δ, b)	(Δ, Δ)	(Δ, e)	(Δ, ae)	(Δ, be)	(Δ, Δe)
e	(e, 1)	(be, 1)	(ae, 1)	(Δe, 1)	(e, e)	(be, e)	(ae, e)	(Δe, e)
ae	(ae, 1)	(Δe, 1)	(ae, b)	(Δe, b)	(ae, e)	(Δe, e)	(ae, be)	(Δe, be)
be	(be, 1)	(be, a)	(Δe, 1)	(Δe, a)	(be, e)	(be, ae)	(Δe, e)	(Δe, ae)
Δe	(Δe, 1)	(Δe, a)	(Δe, b)	(Δe, Δ)	(Δe, e)	(Δe, ae)	(Δe, be)	(Δe, Δe)

Table 2: The □-witness on the Garside family $\{1, \mathsf{a}, \mathsf{b}, \Delta, \mathsf{e}, \mathsf{ae}, \mathsf{be}, \Delta\mathsf{e}\}$ in the wreathed free Abelian monoid $\mathcal{T}$: for every pair of elements (s_1, s_2), it specifies the unique normal decomposition of $s_1 s_2$ of length 2. The values directly follow from considering $\mathcal{S}$ as a germ and using the maximal $\mathcal{J}$-function of Example 3.21.

Input: An element t of $\mathcal{S}^\sharp$ and an $\mathcal{S}$-normal decomposition $s_1|\cdots|s_p$ of an element g of C such that tg exists

Output: An $\mathcal{S}$-normal decomposition of tg

1: $t_0 := t$
2: **for** i increasing from 1 to p **do**
3: $\quad (s'_i, t_i) := W(t_{i-1}, s_i)$
4: **return** $s'_1|\cdots|s'_p|t_p$

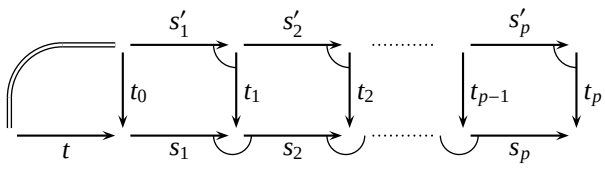


Figure 7: Algorithm 4.3: starting from t in $\mathcal{S}^\sharp$ and an $\mathcal{S}$-normal decomposition of g, it returns an $\mathcal{S}$-normal decomposition of tg.

Proposition 4.4. *Assume that $\mathcal{S}$ is a Garside family in a left-cancellative category C, and W is a □-witness for $\mathcal{S}^\sharp$. Then Algorithm 4.3 running on t and an $\mathcal{S}$-normal decomposition $s_1|\cdots|s_p$ of g returns an $\mathcal{S}$-normal decomposition of tg. The function W is called p times.*

Proof. By construction, the diagram of Figure 7 is commutative. By assumption, $s_i|s_{i+1}$ is $\mathcal{S}$-greedy for every i, and, by the defining property of a □-witness, $s'_i|t_i$ is $\mathcal{S}$-normal for every i. Then the first domino rule (Lemma 1.17) implies that $s'_i|s'_{i+1}$ is $\mathcal{S}$-normal. It is obvious from Algorithm 4.3 that the function W is invoked p times. □

We can now compute $\mathcal{S}$-normal decompositions for an arbitrary element of C specified by an arbitrary $\mathcal{S}^\sharp$-decomposition by iterating Algorithm 4.3.

Algorithm 4.5 (Normal decomposition).

Context: A Garside family $\mathcal{S}$ in a left-cancellative category C, a $\square$-witness W for $\mathcal{S}^\sharp$
Input: An $\mathcal{S}^\sharp$-decomposition $t_1|\cdots|t_p$ of an element g of C
Output: An $\mathcal{S}$-normal decomposition of g

1: $t_{0,i} := t_i$ for $1 \leqslant i \leqslant p$
2: **for** j decreasing from p to 1 **do**
3: **for** i increasing from 1 to $p - j$ (if any) **do**
4: $(s_{i,j-1}, t_{i,j}) := W(t_{i-1,j}, s_{i,j})$
5: $s_{p-j+1,j-1} := t_{p-j,j}$
6: **return** $s_{1,0}|\cdots|s_{p,0}$

Proposition 4.6. *Assume that $\mathcal{S}$ is a Garside family in a left-cancellative category C and W is a $\square$-witness for $\mathcal{S}^\sharp$. Then Algorithm 4.5 running on an $\mathcal{S}^\sharp$-decomposition $t_1|\cdots|t_p$ of an element g of C returns an $\mathcal{S}$-normal decomposition of g. The map W is appealed to $p(p-1)/2$ times.*

Proof. Consider the **for**-loop for j and assume that before the execution of its body, $s_{1,j}|\cdots|s_{p-j,j}$ is an $\mathcal{S}$-normal decomposition of $t_{0,j+1}\cdots t_{0,p}$, which is also $t_{j+1}\cdots t_p$; this condition is trivially satisfied for $j = p$.

According to Proposition 4.4, the execution of lines 3-5 involves $p - j$ invocations of the map W and produces an $\mathcal{S}$-normal decomposition $s_{1,j-1}|\cdots|s_{p-j+1,j-1}$ of $t_{0,j}\cdots t_{0,p}$ that is, of $t_j\cdots t_p$. The claim then follows by induction. $\square$

Example 4.7. Consider the Garside family $\mathcal{S} = \{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ and the element $g = \mathtt{bab}^2$. We begin with the empty word, $\mathcal{S}$-normal decomposition of 1, and, starting from the right, we multiply by the successive letters on the left using Algorithm 4.3 and the $\square$-witness of Table 1, removing factors equal to 1 after every step, thus finding $\mathtt{b}$, $\mathtt{b}|\mathtt{b}$, $\mathtt{ab}|\mathtt{b}$ and, finally, $\Delta|\mathtt{b}$ for the (unique) $\mathcal{S}$-normal decompositions of $\mathtt{b}$, $\mathtt{b}^2$, $\mathtt{ab}^2$, and g.

Example 4.8. In the wreathed free Abelian monoid $\mathcal{T}$, consider the Garside family $\mathcal{S}^\sharp$ from Example 3.21 and the $\square$-witness from Example 4.2. In order to compute an $\mathcal{S}$-normal decomposition of the element $g = \mathtt{be\ ae\ e\ b}$, we begin with the empty word as an $\mathcal{S}$-normal decomposition of the element 1 and repeatedly use Algorithm 4.3 to compute $\mathcal{S}$-normal decompositions of the elements $\mathtt{b}$, $\mathtt{e\ b}$, $\mathtt{ae\ e\ b}$ and g, removing factors equal to 1 after every step. We obtain $\mathtt{b}$, $\mathtt{ae}$, $\Delta|\mathtt{e}$ and $\Delta|\mathtt{ae}|\mathtt{e}$.

4.2. Computing normal decompositions using right-multiplication

Algorithm 4.3 is not symmetric: the construction starts from the left and moves to the right. It is natural to wonder whether a symmetric right–to–left version exists. The answer depends on the considered Garside family $\mathcal{S}$: if the latter satisfies the second domino rule (Lemma 1.32), the previous constructions have counterparts based on right-multiplication.

Algorithm 4.9 (Right-multiplication—see Figure 8).

Context: A Garside family $\mathcal{S}$ satisfying the second domino rule in a left-cancellative category C, a $\square$-witness W for $\mathcal{S}^\sharp$
Input: An element t of $\mathcal{S}^\sharp$, an $\mathcal{S}$-normal decomposition $s_1|\cdots|s_p$ of an element g of C such that gt is defined
Output: An $\mathcal{S}$-normal decomposition of gt

1: $t_p := t$
2: **for** i decreasing from p to 1 **do**
3: $\quad (t_{i-1}, s'_i) := W(s_i, t_i)$
4: **return** $t_0|s'_1|\cdots|s'_p$

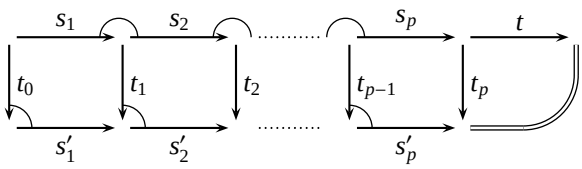


Figure 8: Algorithm 4.9: starting from an $\mathcal{S}$-normal decomposition of g and t, it returns an $\mathcal{S}$-normal decomposition of gt.

Proposition 4.10 (**right-multiplication**). *Assume that $\mathcal{S}$ is a Garside family satisfying the second domino rule in a left-cancellative category C and W is a $\square$-witness for $\mathcal{S}^\sharp$. Then Algorithm 4.9 running on t in $\mathcal{S}^\sharp$ and an $\mathcal{S}$-normal decomposition $s_1|\cdots|s_p$ of g in C returns an $\mathcal{S}$-normal decomposition of gt, if the latter is defined. The function W is invoked p times.*

Proof. The commutativity of the diagram gives $t_0 s'_1 \cdots s'_p = s_1 \cdots s_p t$. Applying the second domino rule to each two-square subdiagram of the diagram of Figure 8 starting from the right, we see that the sequence $t_0|s'_1|\cdots|s'_q$ is $\mathcal{S}$-greedy. As all entries lie in $\mathcal{S}^\sharp$, the sequence is $\mathcal{S}$-normal. $\square$

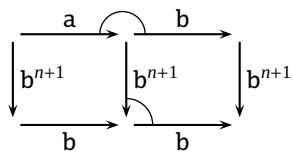
By iterating Algorithm 4.9 (when applicable), we easily obtain a symmetric version of Algorithm 4.5 and Proposition 4.6, that we shall not explicitly state.

Remark 4.11. The effect of the second domino rule is to shorten the computation of certain normal decompositions. Indeed, assume that $t_1|\cdots|t_q$ and $s_1|\cdots|s_p$ are $\mathcal{S}$-normal paths and $t_q s_1$ is defined. By applying Proposition 4.10, we can compute an $\mathcal{S}$-normal decomposition of the product $t_1 \cdots t_q s_1 \cdots s_p$ by filling a diagram as in Figure 9. When valid, the second domino rule guarantees that the path consisting of the first p top edges followed by q vertical edges is $\mathcal{S}$-normal, that is, the triangular part of the diagram may be forgotten.

We conclude with an example of a Garside family that does not satisfy the second domino rule: its existence shows that no uniform result for multiplication on the right is possible.

Example 4.12. For $n \geqslant 2$, let $M_n = \langle \mathtt{a}, \mathtt{b} \mid \mathtt{ab}^n = \mathtt{b}^{n+1} \rangle^+$. The method of Subsection 2.2 shows that M_n is left-cancellative, and that any two elements of M_n that admit a common right-multiple admit a right-lcm. On the other hand, M_n is not right-cancellative since we have $\mathtt{ab}^{n-1} \neq \mathtt{b}^n$ and $\mathtt{ab}^n = \mathtt{b}^{n+1}$. Let $S_n = \{1, \mathtt{a}, \mathtt{b}, \mathtt{b}^2, \ldots, \mathtt{b}^{n+1}\}$, a subset of M_n with $n+3$ elements. Using Lemma 3.1, one can check that S_n is a Garside family in M_n.

Now the second domino rule is not valid for S_n in M_n. Indeed, the diagram aside is commutative, the paths $\mathtt{a}|\mathtt{b}$ and $\mathtt{b}^{n+1}|\mathtt{b}$ are S_n-greedy, and all edges corresponds to elements of S_n. However $\mathtt{b}|\mathtt{b}$ is not S_n-greedy since $\mathtt{b}^2$ lies in S_n.



4.3. Solving the Word Problem

We shall describe two solutions of the Word Problem for a left-cancellative category equipped with a Garside family, one based on normal decompositions, and one based on reversing.

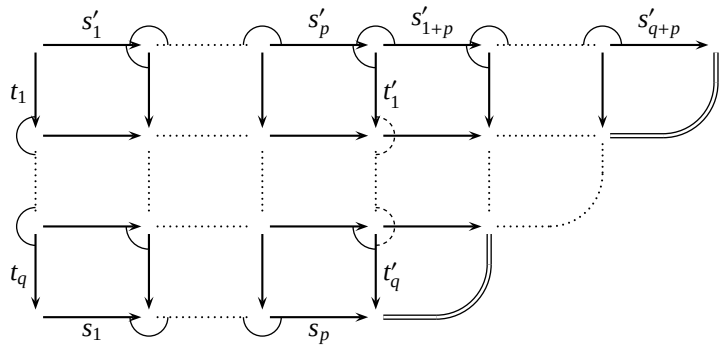


Figure 9: Finding an $\mathcal{S}$-normal decomposition of $t_1 \cdots t_q s_1 \cdots s_p$ when $t_1|\cdots|t_q$ and $s_1|\cdots|s_p$ are $\mathcal{S}$-normal: using Proposition 4.4, hence the first domino rule only, one determines the $\mathcal{S}$-normal sequence $s'_1|\cdots|s'_{q+p}$ in $pq+q(q-1)/2$ steps; if the second domino rule is valid, the sequence $t'_1|\cdots|t'_q$ is already $\mathcal{S}$-normal, and $s'_1|\cdots|s'_p|t'_1|\cdots|t'_q$ is an $\mathcal{S}$-normal decomposition of $t_1 \cdots t_q s_1 \cdots s_p$.

Using $\mathcal{S}$-normal decompositions

Whenever an effective method determining an $\mathcal{S}$-normal decomposition is available, a solution to the Word Problem is very close. However, due to the possible existence of nontrivial invertible elements, $\mathcal{S}$-normal decompositions need not be readily unique, so a final comparison step is needed.

Definition 4.13. Assume that C is a left-cancellative category and $\mathcal{A}$ is included in C. An $=^\times$*-test* on $\mathcal{A}$ is a map E of $\mathcal{A}^2$ to $C^\times \cup \{\bot\}$ satisfying $sE(s,t) = t$ whenever s, t are $=^\times$-equivalent, and $E(s,t) = \bot$ otherwise.

Algorithm 4.14 (Word Problem, positive case I).

Context: A left-cancellative category C, a Garside subfamily $\mathcal{S}$ of C, a $\square$-witness W on $\mathcal{S}^\sharp$, an $=^\times$-test E on $\mathcal{S}^\sharp$

Input: Two $\mathcal{S}^\sharp$-paths u, v

Output: `true` if u, v represent the same element of C, and `false` otherwise

if $\mathrm{src}(u) \neq \mathrm{src}(v)$ or $\mathrm{trg}(u) \neq \mathrm{trg}(v)$ **then**
    return `false`
else
    use Algorithm 4.5 with W to find an $\mathcal{S}$-normal path $s_1|\cdots|s_p$ representing $[u]$
    use Algorithm 4.5 with W to find an $\mathcal{S}$-normal path $t_1|\cdots|t_q$ representing $[v]$
    return the value of CompareNormalPaths($s_1|\cdots|s_p$, $t_1|\cdots|t_q$)

function CompareNormalPaths($s_1|\cdots|s_p$, $t_1|\cdots|t_q$)
    ▹ the paths should have the same source and the same target
    $x :=$ source of t_1 ; $y :=$ target of t_q
    $e_0 := 1_x$; $t_i := 1_y$ for $q < i \leqslant \max(q,p)$; $s_j := 1_y$ for $p < j \leqslant \max(q,p)$
    for $k := 1$ to $\max(q,p)$ **do**
        if $E(t_i, e_{i-1}s_i) \neq \bot$ **then**
            $e_i := E(t_i, e_{i-1}s_i)$
        else
            return `false`
    return $[\![e_i = 1_y]\!]$

Proposition 4.15. *Assume that $\mathcal{S}$ is a Garside family in a left-cancellative category C, W is a $\square$-witness on $\mathcal{S}^\sharp$, and E is an $=^\times$-test on $\mathcal{S}^\sharp$.*

(i) *Given two $\mathcal{S}$-normal paths u and v of length at most ℓ, the function* CompareNormalPaths *in Algorithm 4.14 decides in time $O(\ell)$ whether $[u] = [v]$ holds.*

(ii) *Given two $\mathcal{S}^\sharp$-paths u and v of length at most ℓ, Algorithm 4.14 decides in time $O(\ell^2)$ whether $[u] = [v]$ holds.*

Proof. (i) By Proposition 1.12, two $\mathcal{S}$-normal paths u and v satisfy $[u] = [v]$ if and only if they are $C^\times$-deformations of one another. The latter in turn is the case if and only if $E(t_i, e_{i-1}s_i) \neq \perp$ for all $i = 1, ..., \max(p, q)$ and $e_{\max(p,q)} = 1_y$ hold in lines 10-14 of the function CompareNormalPaths. It is obvious that this test takes time $O(\ell)$.

(ii) If u and v have different sources or different targets, clearly we have $[u] \neq [v]$. Otherwise, one has $[u] = [v]$ if and only if the paths computed in lines 4 and 5 of Algorithm 4.14 represent the same element. By Proposition 4.6, lines 4 and 5 of Algorithm 4.14 have a cost of $O(\ell^2)$ and the lengths of the produced paths is at most ℓ. The claim then follows with (i). $\square$

Example 4.16. Consider once again the 3-strand braid monoid B_3^+. Let $\mathcal{S}$ be $\{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$. We saw in Example 2.27 that $\mathcal{S}$ is a (minimal) Garside family in B_3^+ containing 1, and determined in Table 1 a $\square$-witness on $\mathcal{S}$. As B_3^+ contains no nontrivial invertible element, testing for $=^\times$ reduces to testing equality.

As in Figure 6, consider $u = \mathtt{a}|\mathtt{b}|\mathtt{b}$ and $v = \mathtt{b}|\mathtt{a}|\mathtt{b}|\mathtt{b}$. Applying Algorithm 4.5, we obtain the normal decomposition $\mathtt{ab}|\mathtt{b}$ of $[u]$ and the normal decomposition $\Delta|\mathtt{b}$ of $[v]$. As $\mathtt{ab}$ and Δ are not equal, hence not $=^\times$-equivalent, Algorithm 4.14 returns `false`, that is, one has $[u] \neq [v]$.

Example 4.17. Consider again the wreathed free Abelian monoid $\mathcal{T}$ from Example 4.8. An $=^\times$-test on $\mathcal{S}^\sharp$ is given by the map E defined on $(\mathcal{S}^\sharp)^2$ by $E(s, s) = 1$ for s in $\mathcal{S}^\sharp$, $E(s\mathtt{e}, s) = E(s, s\mathtt{e}) = \mathtt{e}$ for s in $\{1, \mathtt{a}, \mathtt{b}, \Delta\}$, and $E(s, t) = \perp$ for all other pairs (s, t) in $(\mathcal{S}^\sharp)^2$.

In Example 4.8, we obtained the $\mathcal{S}$-normal decomposition $\Delta\mathtt{e}|\mathtt{ae}|\mathtt{e}$ of the element $g = \mathtt{be\,ae\,e\,b}$. Similarly, Algorithm 4.5 yields the $\mathcal{S}$-normal decomposition $\Delta|\mathtt{be}$ of the element $g' = \mathtt{a\,be\,a}$.

Using the function CompareNormalPaths in Algorithm 4.14, we obtain $e_1 = E(\Delta, \Delta\mathtt{e}) = \mathtt{e}$, $e_2 = E(\mathtt{be}, \mathtt{eae}) = E(\mathtt{be}, \mathtt{b}) = \mathtt{e}$, and $e_3 = E(1, \mathtt{ee}) = 1$, and thus conclude that $[g] = [g']$ holds.

Using reversing

An alternative solution of the Word Problem that does not require computing a normal form can be given using the reversing method of Subsection 2.1. By definition, if the investigated category C is specified using a right-complemented presentation $(\mathcal{A}; \mathcal{R})$ for which right-reversing is complete, then two $\mathcal{A}$-paths u, v represent the same element of C if and only if the signed path $\overline{u}|v$ reverses to an empty path, that is, if and only if Algorithm 2.6 running on $\overline{u}|v$ returns an empty path. This however provides a solution to the Word Problem only if right-reversing is known to terminate in finite time. By Lemma 2.10, this happens whenever any two elements with the same source admit a common right-multiple in the considered category, but nothing can be said in more general cases.

Now the termination problem does not arise when one considers a presentation that is associated with a short right-complement, in which case the Word Problem can be solved using reversing. The point here is that every Garside family gives rise to such a presentation:

Lemma 4.18. *Under the assumptions of Proposition 1.16, the obtained presentation* $(\mathcal{S};\mathcal{R})$ *of* C *is associated with a short right-complement, and right-reversing is complete for this presentation.*

Proof. Let θ be the partial map defined on $\mathcal{S}$ by $\theta(s,t) = s \setminus t$. By definition, θ is a right-complement on $\mathcal{S}$ and, by Lemma 1.15, it is short. Next, the associativity of the right-lcm operation implies that θ satisfies the cube condition for every triple of elements of $\mathcal{S}$. Then Proposition 2.19, implies that right-reversing is complete for $(\mathcal{S};\mathcal{R})$. □

It follows that the presentation $(\mathcal{S};\mathcal{R})$ is eligible for right-reversing and that the latter solves the Word Problem.

Algorithm 4.19 (Word Problem, positive case II).

Context: A Garside family $\mathcal{S}$ in a left-cancellative category C that admits unique local right-lcms, the right-lcm selector θ on $\mathcal{S} \cup \mathbf{1}_C$
Input: Two $\mathcal{S}$-paths u, v
Output: `true` if u, v represent the same element of C, and `false` otherwise
1: `ret` := the return value of Algorithm 2.3 for right-complement θ and input $\overline{u}|v$
2: **return** $[\![\texttt{ret} = \varepsilon_{-}]\!]$ ▹ `ret` is either a positive–negative path or equal to `fail`

Lemmas 4.18 and 2.4 immediately imply:

Proposition 4.20. *Assume that* $\mathcal{S}$ *is a Garside family in a left-cancellative category* C *that admits unique local right-lcms.*
(i) *Algorithm 4.19 solves the Word Problem of* C *with respect to* $\mathcal{S}$.
(ii) *If* $\mathcal{S}$ *is finite, the complexity of Algorithm 4.19 is quadratic in the length of the input paths.*

Example 4.21. Consider once again the 3-strand braid monoid B_3^+. As seen in Example 2.27, the five elements family $\{\mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ is a (minimal) Garside family in B_3^+. The resulting presentation of B_3^+ is $(\mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta; \mathcal{R})$, where $\mathcal{R}$ consists of $\binom{5}{2}$ relations (in which we leave the concatenation sign to avoid ambiguities) $\mathtt{a}|\mathtt{ba} = \mathtt{b}|\mathtt{ab}$, $\mathtt{a}|\mathtt{b} = \mathtt{ab}$, $\mathtt{a}|\mathtt{ba} = \mathtt{ba}|\mathtt{b}$, $\mathtt{a}|\mathtt{ba} = \Delta$, $\mathtt{b}|\mathtt{ab} = \mathtt{ab}|\mathtt{a}$, $\mathtt{b}|\mathtt{a} = \mathtt{ba}$, $\mathtt{b}|\mathtt{ab} = \Delta$, $\mathtt{ab}|\mathtt{a} = \mathtt{ba}|\mathtt{b}$, $\mathtt{ab}|\mathtt{a} = \Delta$, $\mathtt{ba}|\mathtt{b} = \Delta$. Starting, as in Example 4.16, from the words $u = \mathtt{a}|\mathtt{b}|\mathtt{b}$ and $v = \mathtt{b}|\mathtt{a}|\mathtt{b}|\mathtt{b}$, we apply Algorithm 2.3 for the input $\overline{u}|v$, obtaining $w = \mathtt{ab}|\overline{\mathtt{a}}$ (see Figure 10) and thus, as w is not empty, we conclude once again that u and v are not equivalent.

One of the benefits of the reversing approach is to provide at the same time a decision method for the left-divisibility relation, as Lemma 4.18 also implies:

Proposition 4.22. *If line 2 of Algorithm 4.19 is modified so that the algorithm returns* `true` *if* `ret` *is a positive path, and* `false` *otherwise, the modified algorithm returns* `true` *if and only if* $[u]$ *left-divides* $[v]$ *in* C.

Remark 4.23. The above results extend to every left-cancellative category, possibly containing nontrivial invertible elements and not admitting local right-lcms, at the expense of considering a more general, non-deterministic version of reversing. But, as said in Remark 2.8, we do not consider such extensions here.

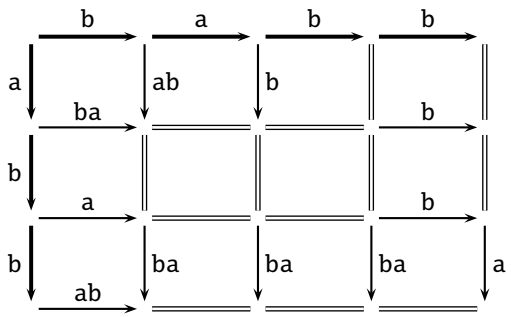


Figure 10: The grid associated with the right-reversing of $\overline{u}|v$ in Example 4.21; compare with the grid of Figure 6: the final paths are equivalent, but not equal, which is not contradictory as they are obtained in different ways.

4.4. Computing lcms

Right-lcms

It follows from Lemma 2.24 that, whenever a right-complemented presentation $(\mathcal{A};\mathcal{R})$ is given for a left-cancellative category C and right-reversing is complete for that presentation, then right-reversing provides a direct algorithm for computing right-lcms when they exist.

Example 4.24. Consider again the presentation $(\mathtt{a},\mathtt{b};\mathtt{aba}=\mathtt{bab})$ of the braid monoid B_3^+ and the elements $\mathtt{ab}^2$ and $\mathtt{bab}^2$. We saw in Example 2.21 that right-reversing is complete for this presentation. Now, we saw in Figure 6 that the word $\overline{\mathtt{b}}|\overline{\mathtt{b}}|\overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}|\mathtt{b}|\mathtt{b}$ is right-reversible to $\mathtt{a}|\mathtt{b}|\overline{\mathtt{a}}$ and, therefore, we conclude that the right-lcm of $\mathtt{ab}^2$ and $\mathtt{bab}^2$ is $\mathtt{ab}^2\cdot\mathtt{ab}$ (which is also $\mathtt{bab}^2\cdot\mathtt{a}$).

Now, independently of the way the considered category C was initially specified, if one happens to know a right-lcm selector θ on a generating subfamily $\mathcal{A}$, we can use the latter, which by definition is a right-complement, to right-reverse $\mathcal{A}$-paths. If the right-selector is not short, termination is not guaranteed but, if it is short, we immediately obtain:

Proposition 4.25. *Assume that θ is a short right-lcm selector on some generating family $\mathcal{A}$ in a left-cancellative category C that admits local right-lcms. For all $\mathcal{A}$-paths u, v, if Algorithm 2.3 running on $\overline{u}|v$ returns a positive–negative path $v'|\overline{u'}$, then $[u|v']$ is a right-lcm of $[u]$ and $[v]$ in C; if it returns* `fail`*, then $[u]$ and $[v]$ admit no right-lcm in C. The complexity of the computation is in $O((\|u\|+\|v\|)^2)$*

Proof. As observed in the proof of Lemma 4.18, right-θ-reversing must be complete in this case, and Lemma 2.24 then gives the expected result. Alternatively, we can directly observe that, if g_1f' is a right-lcm of f and g_1 and g_2f'' is a right-lcm of f' and g_2, then g_1g_2f'' is a right-lcm of f and g_1g_2, which inductively implies that the diagonal of every rectangle in a right-reversing grid represents the right-lcm of the left and top edges. □

The point here is that, by Lemma 4.18, every Garside family in a left-cancellative category that admits right-lcms and does not contain any nontrivial invertible elements, gives rise to a short right-lcm selector, and therefore is eligible for Proposition 4.25. Moreover, by Proposition 2.32, the existence of right-lcms in C reduces to the existence of right-lcms inside $\mathcal{S}$. It follows that, when $\mathcal{S}^\sharp$ is finite, one can effectively decide the existence of right-lcms for the elements of $\mathcal{S}^\sharp$ and, in addition, obtain a right-lcm selector.

Example 4.26. Starting with the Garside family $\{\mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ in B_3^+, the right-lcm selector θ on $\mathcal{S}$ leads to the presentation of Example 4.21. Considering as in Example 4.24 the elements $\mathtt{ab}^2$ and $\mathtt{bab}^2$, we find their right-lcm by right-θ-reversing $\overline{\mathtt{b}}|\overline{\mathtt{b}}|\overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}|\mathtt{b}|\mathtt{b}$. As seen in Figure 10, the latter word reverses to $\mathtt{ab}|\overline{\mathtt{a}}$, and we conclude once again that the right-lcm of $\mathtt{ab}^2$ and $\mathtt{bab}^2$ is $\mathtt{ab}^2 \cdot \mathtt{ab}$.

Remark 4.27. The above approach consisting in using the right-reversing transformation associated with a (short or non-short) right-lcm selector θ on some subfamily $\mathcal{A}$ may work even when $\mathcal{A}$ is not a Garside family, provided $\mathcal{A}$ generates the ambient category $\mathcal{C}$ and the family of all relations $s\theta(s,t) = t\theta(t,s)$ with s, t in $\mathcal{A}$ makes a presentation of $\mathcal{C}$. Interestingly, this is always the case when $\mathcal{C}$ is right-Noetherian, but the example of the Dubrovina-Dubrovin 4-strand braid monoid (Dehornoy, 2012) shows that the property may fail in the general case.

Left-lcms

The case of left-lcms is symmetric, and we shall not say much: by the counterpart of Proposition 4.25, if there exists a short left-lcm-selector $\widetilde{\theta}$ on a generating family $\mathcal{A}$, then left-$\widetilde{\theta}$-reversing a positive–negative path $v|\overline{u}$ possibly leads to a negative–positive path $\overline{u'}|v'$ such that $u'|v$ and $v'|u$ represent the left-lcm of $[u]$ and $[v]$.

Remark 4.28. Even in the presence of nontrivial invertible elements, it may be possible to apply Proposition 4.25, respectively its counterpart, and compute right-lcms by right-reversing and left-lcms by left-reversing. We illustrate this remark by the following example.

Example 4.29. Recall the wreathed free Abelian monoid defined in Example 1.11. It is easily verified from the multiplication table (cf. Example 3.21) that a short right-lcm selector θ on $\mathcal{S}^\sharp$ is given by the left hand table below, and that a short left-lcm selector $\widetilde{\theta}$ on $\mathcal{S}^\sharp$ is given by the right hand table below.

θ	1	a	b	Δ	e	ae	be	Δe
1	1	a	b	Δ	1	a	b	Δ
a	1	1	b	b	1	1	b	b
b	1	a	1	a	1	a	1	a
Δ	1	1	1	1	1	1	1	1
e	e	be	ae	Δe	e	be	ae	Δe
ae	e	e	ae	ae	e	e	ae	ae
be	e	be	e	be	e	be	e	be
Δe	e	e	e	e	e	e	e	e

$\widetilde{\theta}$	1	a	b	Δ	e	ae	be	Δe
1	1	1	1	1	e	e	e	e
a	a	1	a	1	ae	ae	e	e
b	b	b	1	1	be	e	be	e
Δ	Δ	b	a	1	Δe	ae	be	e
e	1	1	1	1	e	e	e	e
ae	b	b	1	1	be	e	be	e
be	a	1	a	1	ae	ae	e	e
Δe	Δ	b	a	1	Δe	ae	be	e

Algorithm 2.3, respectively its counterpart, can be used to compute right-lcms and left-lcms in $\mathcal{T}$. For example, using θ to right-reverse the negative-positive path $\overline{\mathtt{a}}|\overline{\mathtt{be}}|\overline{\mathtt{be}}|\mathtt{ae}|\mathtt{a}|\Delta\mathtt{e}|\mathtt{a}$, removing trivial factors when they occur, yields the positive-negative path $\mathtt{e}|\mathtt{ae}|\mathtt{ae}|\overline{\mathtt{e}}$. Hence, a right-lcm of $u = \mathtt{ae\,a}\,\Delta\mathtt{e\,a}$ and $v = \mathtt{be\,be\,a}$ is represented by both $u|\mathtt{e}$ and $v|\mathtt{e}|\mathtt{ae}|\mathtt{ae}$. In fact, as the element $\mathtt{e}$ is invertible, we have $v \preccurlyeq u$.

5. Computing with a strong Garside family, signed case

In this final section, we show how to use (strong) Garside families to compute in groupoids of fractions. The successive problems we address are finding symmetric normal decompositions

(Subsection 5.1), finding Δ-normal decompositions (Subsection 5.2), solving the Word Problem (Subsection 5.3), computing inverses (Subsection 5.4) and, finally, computing lower and upper bounds (Subsection 5.5).

5.1. Computing symmetric normal decompositions

According to Proposition 1.22, if $\mathcal{S}$ is a Garside family in a left-Ore category C, then every element of $\mathcal{E}nv(C)$, the groupoid of fractions of C, that can be represented as a right-fraction admits a symmetric $\mathcal{S}$-normal decomposition. Here we address the question of algorithmically computing such a decomposition.

Starting from a right fraction

Owing to the above recalled restriction on the considered elements (namely that of being expressible as a right-fraction), it is natural to first start with a positive–negative path. Then, Lemma 1.21 reduces the computation of a symmetric normal decomposition to the determination of a left-lcm:

Algorithm 5.1 (Symmetric normal decomposition, positive–negative input).

Context: A Garside family $\mathcal{S}$ in a left-Ore category C that admits left-lcms, a $\square$-witness W on $\mathcal{S}^\sharp$, a procedure computing left-lcms in C

Input: A positive–negative $\mathcal{S}^\sharp$-path $v|\overline{u}$

Output: A symmetric $\mathcal{S}$-normal decomposition of the element $[v][u]^{-1}$ of $\mathcal{E}nv(C)$

1: find u', v' such that $u'|v$ and $v'|u$ represent a left-lcm of $[u]$ and $[v]$
2: find an $\mathcal{S}$-normal path $t_1|\cdots|t_q$ equivalent to u' using Algorithm 4.5 with W
3: find an $\mathcal{S}$-normal path $s_1|\cdots|s_p$ equivalent to v' using Algorithm 4.5 with W
4: **return** $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$

Proposition 5.2. *Assume that C is a left-Ore category that admits left-lcms, $\mathcal{S}$ is a Garside family in C, W is a $\square$-witness on $\mathcal{S}^\sharp$, and that left-lcms in C can be computed effectively. Then Algorithm 5.1 returns a symmetric $\mathcal{S}$-normal decomposition of* $[v][u]^{-1}$.

Proof. By construction, the paths $s_1|\cdots|s_p$ and $t_1|\cdots|t_q$ are $\mathcal{S}$-normal, and $t_1\cdots t_q v$, which is also $s_1\cdots s_p u$, is a left-lcm of u and v. By Lemma 1.21, this implies that $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ is symmetric $\mathcal{S}$-normal. $\square$

Example 5.3. Again in the case of the braid monoid B_3^+ given by the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{aba} = \mathtt{bab})$ and the Garside family $\mathcal{S} = \{\mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$, let us consider the element $g = \mathtt{a}^2\mathtt{b}\mathtt{a}^{-1}\mathtt{b}^{-1}$. A decomposition of g as a right-fraction is the signed $\mathcal{A}$-path $\mathtt{a}|\mathtt{a}|\mathtt{b}|\overline{\mathtt{a}}|\overline{\mathtt{b}}$. By a symmetric argument to that used in Example 2.21, the presentation $(\mathtt{a}, \mathtt{b}; \mathtt{aba} = \mathtt{bab})$ is left-complemented and left-reversing is complete for it. Hence left-reversing computes left-lcms. Here, left-reversing $\mathtt{a}|\mathtt{a}|\mathtt{b}|\overline{\mathtt{a}}|\overline{\mathtt{b}}$ leads to $\overline{\mathtt{b}}|\overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}|\mathtt{a}$ (see Figure 11, left), so $\mathtt{ab} \cdot \mathtt{a}^2\mathtt{b}$, which is also $\mathtt{ba}^2 \cdot \mathtt{ba}$, is a left-lcm of $\mathtt{a}^2\mathtt{b}$ and $\mathtt{ba}$. The $\mathcal{S}$-normal decompositions of $\mathtt{ab}$ and $\mathtt{ba}^2$ respectively are $\mathtt{ab}$ and $\mathtt{ba}|\mathtt{a}$. We conclude that $\overline{\mathtt{ab}}|\mathtt{ba}|\mathtt{a}$ is a symmetric $\mathcal{S}$-normal decomposition of g.

Algorithm 5.1 is not fully satisfactory in that, as explained in Subsection 4.4, there exists no general solution to the question of computing left-lcms. It may be that the ambient category has been specified by a left-complemented presentation $(\mathcal{A}; \mathcal{R})$ for which left-reversing is complete,

in which case left-reversing effectively computes left-lcms, but this need not be the case in general. Now, by (the counterpart of) Proposition 4.25, the natural context for computing left-lcms effectively is when a short left-lcm selector exists.

Lemma 5.4. *Assume that $\mathcal{S}$ is a Garside family in a left-Ore category C that admits left-lcms. Then there exists a short left-lcm selector on $\mathcal{S}^\sharp$ if and only if $\mathcal{S}$ is strong.*

Proof. For $\mathcal{S}$ to be strong means that, for all s, t with the same target in $\mathcal{S}^\sharp$, there exist s', t' in $\mathcal{S}^\sharp$ satisfying $s't = t's$ and such that s' and t' are left-disjoint. By Lemma 1.21, $s't$ is a left-lcm of s and t. □

We are thus led to restate Algorithm 5.1 in the case of a strong Garside family.

Algorithm 5.5 (Symmetric normal decomposition, positive–negative input).

Context: A strong Garside family $\mathcal{S}$ in a left-Ore category C that admits left-lcms, a short lcm-selector $\widetilde{\theta}$ and a □-witness W on $\mathcal{S}^\sharp$
Input: A positive–negative $\mathcal{S}^\sharp$-path $v|\overline{u}$
Output: A symmetric $\mathcal{S}$-normal decomposition of the element $[v][u]^{-1}$ of $\mathcal{E}nv(C)$

1: left-$\widetilde{\theta}$-reverse $v|\overline{u}$ into a negative–positive path $\overline{u'}|v'$
2: find an $\mathcal{S}$-normal path $t_1|\cdots|t_q$ equivalent to u' using Algorithm 4.5 with W
3: find an $\mathcal{S}$-normal path $s_1|\cdots|s_p$ equivalent to v' using Algorithm 4.5 with W
4: **return** $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$

Proposition 5.6. *Assume that $\mathcal{S}$ is a strong Garside family in a left-Ore category C that admits left-lcms, W is a □-witness on $\mathcal{S}^\sharp$, and $\widetilde{\theta}$ is a short left-lcm selector on $\mathcal{S}^\sharp$. Then Algorithm 5.5 running on a positive–negative $\mathcal{S}^\sharp$-path $v|\overline{u}$ returns a symmetric $\mathcal{S}$-normal decomposition of $[v][u]^{-1}$ in time $O((\|u\| + \|v\|)^2)$.*

We skip the proof, which is similar to that of Proposition 5.2.

Example 5.7. With the notation of Example 5.3, the difference is that, in Algorithm 5.5, we use the left-reversing associated with the presentation $(\mathcal{S}; \widetilde{\mathcal{R}})$ deduced from $\widetilde{\theta}$ rather than the initial presentation $(\mathcal{A}; \mathcal{R})$. The result of left-reversing $\mathtt{a}|\mathtt{a}|\mathtt{b}|\overline{\mathtt{a}}|\overline{\mathtt{b}}$ is now the length 3 word $\overline{\mathtt{ab}}|\mathtt{ba}|\mathtt{a}$ (see Figure 11, right), which is equivalent to $\overline{\mathtt{b}}|\overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}|\mathtt{a}$ as obtained previously (see Figure 11, left). As $\mathtt{ab}$ and $\mathtt{ba}|\mathtt{a}$ are $\mathcal{S}$-normal, the subsequent normalization steps change nothing, and we conclude again that $\overline{\mathtt{ab}}|\mathtt{ba}|\mathtt{a}$ is a symmetric $\mathcal{S}$-normal decomposition of g.

Example 5.8. Consider again the wreathed free Abelian monoid $\mathcal{T}$. We found a □-witness W in Example 4.2 and a short left-lcm selector $\widetilde{\theta}$ in Example 4.29. We apply Algorithm 5.5 to compute a symmetric $\mathcal{S}$-normal decomposition of the positive-negative path $w = \mathtt{ae}|\mathtt{a}|\Delta|\mathtt{a}|\overline{\mathtt{a}}|\overline{\mathtt{be}}|\overline{\mathtt{be}}$.

Left-$\widetilde{\theta}$-reversing of w and removing trivial factors yields the negative-positive path $\overline{u'}|v'$ with $u' = \mathtt{e}$ and $v' = \mathtt{b}|\mathtt{a}|\mathtt{e}|\mathtt{e}$, and applying Algorithm 4.5 with the □-witness W to u' and v', again removing trivial factors, we obtain the $\mathcal{S}$-normal decompositions $\mathtt{e}$ of u' respectively $\Delta|\mathtt{e}|\mathtt{e}$ of v'. Thus, $\overline{\mathtt{e}}|\Delta|\mathtt{e}|\mathtt{e}$ is a symmetric $\mathcal{S}$-normal decomposition of w.

Remark 5.9. Algorithm 5.5 (as well as Algorithm 5.1) obeys the scheme "left-reverse, then normalize". We could as well use switch the operations into "normalize, then left-reverse", as suggested in Figure 12. The property that guarantees the correctness of the switched version is

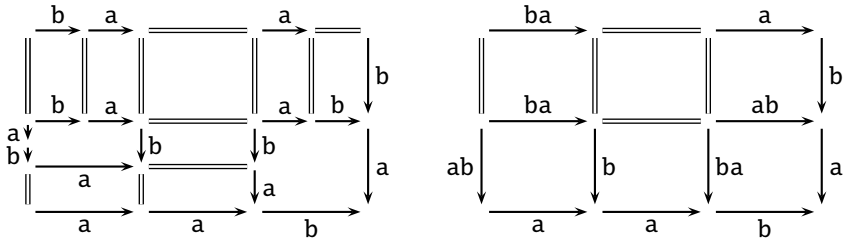
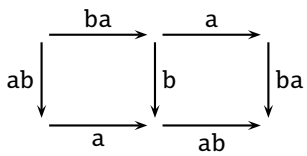


Figure 11: Left-reversing of the word $\mathtt{a}|\mathtt{a}|\mathtt{b}|\overline{\mathtt{a}}|\overline{\mathtt{b}}$ using the left-complement on $\{\mathtt{a}, \mathtt{b}\}$ (left) and the left-lcm selector on the Garside family $\{\mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ (right). Here the right diagram corresponds to gathering pieces of the left diagram, but this would not necessarily be the case for more complicated examples.

the third domino rule (Lemma 1.26): left-reversing $\mathcal{S}$-normal paths always produces $\mathcal{S}$-normal paths.

For instance, in the context of Example 5.7, we could first find $\mathcal{S}$-normal words equivalent to $\mathtt{a}|\mathtt{a}|\mathtt{b}$ and $\mathtt{b}|\mathtt{a}$, namely $\mathtt{a}|\mathtt{ab}$ and $\mathtt{ba}$, and then left-reverse the (length 3) signed word $\mathtt{a}|\mathtt{ab}|\overline{\mathtt{ba}}$. Of course, the result is $\overline{\mathtt{ab}}|\mathtt{ba}|\mathtt{a}$ once more, but the involved grid is now the one given on the right.



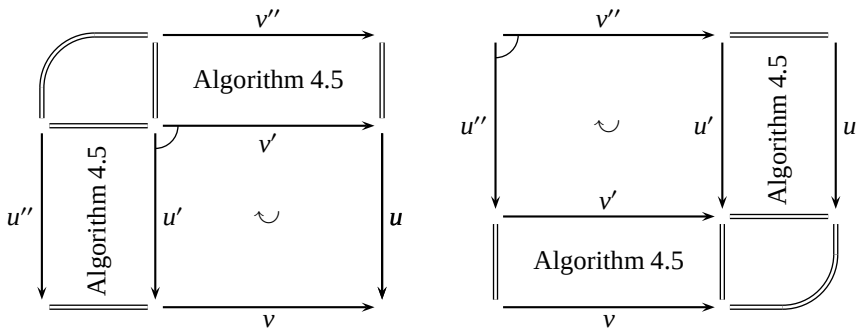


Figure 12: Switching normalization and left-reversing in Algorithm 5.5: on the left, we first left-reverse $v|\overline{u}$ into $\overline{u'}|v'$ and then normalize u' into u'' and v' into v''; on the right, we first normalize u into u' and v into v' and then left-reverse $v'|\overline{u'}$ into $\overline{u''}|v''$.

Starting from an arbitrary path

Extending Algorithm 5.5 to start with an arbitrary path rather than with a positive–negative path is easy. Indeed, we saw that every Garside family provides a short right-lcm selector and using the associated right-reversing transforms an arbitrary path into an equivalent positive–negative path, if some exists.

Algorithm 5.10 (Symmetric normal decomposition, general input).

Context: A strong Garside family $\mathcal{S}$ in a left-Ore category C that admits left-lcms, a short right-lcm selector θ, a $\square$-witness W, and a short left-lcm selector $\widetilde{\theta}$ on $\mathcal{S}^\sharp$

Input: A signed $\mathcal{S}^\sharp$-path w

Output: A symmetric $\mathcal{S}^\sharp$-normal decomposition of $[w]$

```
O := the return value of Algorithm 2.6 with θ running on w
if O is a positive–negative path v|ū then
    compute a symmetric S-normal decomposition ū'|v' of v|ū (Algorithm 5.5)
    return ū'|v'
else                                                    ▷ O is fail
    return fail                  ▷ [w] has no symmetric S-normal decomposition
```

Proposition 5.11. *Assume that $\mathcal{S}$ is a strong Garside family in an Ore category C, W is a $\square$-witness, θ is a short right-lcm selector on $\mathcal{S}^\sharp$, and $\widetilde{\theta}$ is a short left-lcm selector on $\mathcal{S}^\sharp$. Then Algorithm 5.10 running on a signed $\mathcal{S}^\sharp$-path w returns a symmetric $\mathcal{S}$-normal decomposition of $[w]$ in time $O(\|w\|^2)$.*

Proof. By Proposition 5.6, the final sequence $\overline{u'}|v'$ is a symmetric $\mathcal{S}$-normal decomposition of the element $[v|\overline{u}]$ of $\mathcal{E}nv(C)$. Now, by construction, $v|\overline{u}$ represents the same element of $\mathcal{E}nv(C)$ as the initial path w.

The complexity follows from Lemma 2.10 and Propositions 4.6 and 5.6. $\square$

Example 5.12. Again in the B_3^+ context, let w be the signed word $w = \overline{\mathtt{b}}|\overline{\mathtt{a}}|\mathtt{b}|\mathtt{a}|\mathtt{a}$. Right-reversing w using the right-lcm selector on $\{1, \mathtt{a}, \mathtt{b}, \mathtt{ab}, \mathtt{ba}, \Delta\}$ yields the positive–negative word $\mathtt{a}|\mathtt{ab}|\overline{\mathtt{ba}}$. Left-reversing the latter using the left-lcm selector yields $\overline{\mathtt{ab}}|\mathtt{ba}|\mathtt{a}$. Normalizing $\mathtt{ab}$ respectively $\mathtt{ba}|\mathtt{a}$ do not change these words, and we conclude (once again) that w is symmetric normal.

Example 5.13. Consider again the wreathed free Abelian monoid $\mathcal{T}$ with the $\square$-witness W from Example 4.2 and the short right-lcm selector θ as well as the short left-lcm selector $\widetilde{\theta}$ from Example 4.29. We use Algorithm 5.10 to compute a symmetric $\mathcal{S}$-normal decomposition of the signed path $w = \overline{\Delta}|\overline{\mathtt{ae}}|\overline{\mathtt{be}}|\mathtt{ae}|\mathtt{b}|\Delta|\mathtt{a}$, removing trivial factors in every step.

Right-θ-reversing of the path w using Algorithm 2.6 yields the positive-negative path $v|\overline{u}$ with $u = \mathtt{a}$ and $v = \mathtt{ae}|\mathtt{be}|\mathtt{e}$. Subsequent application of Algorithm 5.5 to the positive-negative path $v|\overline{u}$ yields the symmetric $\mathcal{S}$-normal decomposition $\overline{u'}|v'$ of w with $u' = \mathtt{ae}$ and $v' = \mathtt{be}|\mathtt{ae}$.

Remark 5.14. The Baumslag–Solitar presentation $(\mathtt{a}, \mathtt{b}; \mathtt{a} = \mathtt{b}^2\mathtt{ab})$ gives an example of a monoid that is left-Ore with left-lcms but not Ore. For instance, as follows from Example 2.9, the elements $\mathtt{a}$ and $\mathtt{ba}$ admit no common right-multiple. As a consequence, the element $\mathtt{a}^{-1}\mathtt{ba}$ of the group $\langle \mathtt{a}, \mathtt{b} \mid \mathtt{a} = \mathtt{b}^2\mathtt{ab} \rangle$ admits no expression as a right-fraction, and no expression as a left-fraction $f^{-1}g$ where f, g are left-disjoint.

Incremental method

Besides Algorithms 5.5 and 5.10, which are global, we can also look for local methods in the spirit of Algorithms 4.3 and 4.5, that is, methods for computing, say, a symmetric normal decomposition of rg from a symmetric normal decomposition of g when r lies in the reference Garside family. This is easy.

Algorithm 5.15 (Left-multiplication—see Figure 13).

Context: A strong Garside family $\mathcal{S}$ in a left-Ore category C that admits left-lcms, a short left-lcm selector $\widetilde{\theta}$ and a $\square$-witness W on $\mathcal{S}^\sharp$

Input: A symmetric $\mathcal{S}$-normal decomposition $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ of an element g of $\mathcal{E}nv(C)$ and an element r of $\mathcal{S}^\sharp$

Output: A symmetric $\mathcal{S}$-normal decomposition of rg, if the latter is defined

1: $r_{-q} := r$
2: **for** i decreasing from q to 1 **do**
3: $\quad t'_i := \widetilde{\theta}(r_{-i}, t_i)\,;\, r_{-i+1} := \widetilde{\theta}(t_i, r_{-i})$
4: **for** i increasing from 1 to p **do**
5: $\quad (s'_i, r_i) := W(r_{i-1}, s_i)$
6: **return** $\overline{t'_q}|\cdots|\overline{t'_1}|s'_1|\cdots|s'_p|r_p$

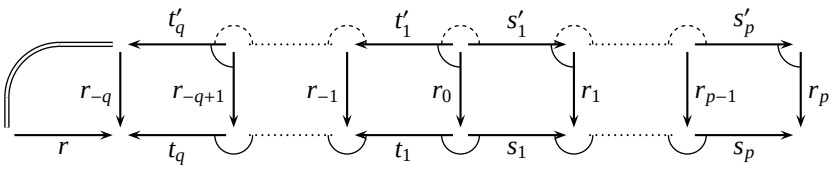


Figure 13: Algorithm 5.15: starting from r in $\mathcal{S}^\sharp$ and a symmetric $\mathcal{S}$-normal decomposition $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ of g, one obtains a symmetric $\mathcal{S}$-normal decomposition of rg.

Example 5.16. Consider again the wreathed free Abelian monoid $\mathcal{T}$. We saw in Example 5.13 that the path $w = \overline{\mathsf{ae}}|\mathsf{be}|\mathsf{ae}$ is a symmetric $\mathcal{S}$-normal decomposition of the element $g = [w]$.

Applying Algorithm 5.15 with the $\square$-witness W from Example 4.2 and the short left-lcm selector $\widetilde{\theta}$ from Example 4.29, we can conclude that an $\mathcal{S}$-normal decomposition of the element $\mathsf{a}g$ is given by $\overline{\mathsf{b}}|\mathsf{ae}|\mathsf{be}|\mathsf{ae}$.

Proposition 5.17. *Assume that $\mathcal{S}$ is a strong Garside family in a left-Ore category C that admits left-lcms, W is a $\square$-witness on $\mathcal{S}^\sharp$, and $\widetilde{\theta}$ is a short left-lcm selector on $\mathcal{S}^\sharp$. Then Algorithm 5.15 running on a symmetric $\mathcal{S}$-normal decomposition of an element g of $\mathcal{E}nv(C)$ and r in $\mathcal{S}^\sharp$ returns a symmetric $\mathcal{S}$-normal decomposition of rg if the latter is defined. The map $\widetilde{\theta}$ is invoked $2q$ times and the map W is invoked p times.*

Proof. The third domino rule (Lemma 1.26) implies that $t'_1|\cdots|t'_q$ is $\mathcal{S}$-greedy. The fourth domino rule implies that t'_1 and s'_1 are left-disjoint. Next, the first domino rule implies that $s'_1|\cdots|s'_p$ is $\mathcal{S}$-normal. Finally, $s'_p|r_p$ is $\mathcal{S}$-normal by construction. Hence $\overline{t'_q}|\cdots|\overline{t'_1}|s'_1|\cdots|s'_p|r_p$ is symmetric $\mathcal{S}$-normal. The commutativity of the diagram in Figure 13 then guarantees that the latter sequence is a decomposition of $rt_q^{-1}\cdots t_1^{-1}s_1\cdots s_p$, hence of rg. $\square$

The symmetry of the definition of symmetric normal paths implies that there exists a counterpart of Algorithm 5.15 computing a symmetric normal decomposition of gr^{-1} from one of g for r in the reference Garside family $\mathcal{S}$. By contrast, there exists no simple general method for left-division (or right-multiplication). However, when $\mathcal{S}$ happens to be bounded, such a method exists, because left-dividing by an element $\Delta(x)$ is easy. Now, if $\mathcal{S}$ is bounded by Δ and s belongs to $\mathcal{S}(x, \text{-})$, then $s\partial(s) = \Delta(x)$ holds, so that left-dividing by s can be decomposed into left-dividing

by $\Delta(x)$ and then left-multiplying by $\partial(s)$. Hereafter, in order to simplify notation, we shall often skip the source of the considered Δ-element, thus writing $\Delta(\text{-})$ instead of $\Delta(x)$ when mentioning x explicitly is not necessary.

Algorithm 5.18 (Left-division—see Figure 14).

Context: A Garside family $\mathcal{S}$ bounded by a map Δ in an Ore category C, a $\square$-witness W for $\mathcal{S}^\sharp$
Input: A symmetric $\mathcal{S}$-normal decomposition $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ of g in $\mathcal{E}nv(C)$, an element r of $\mathcal{S}^\sharp$
Output: A symmetric $\mathcal{S}$-normal decomposition of $r^{-1}g$, if the latter is defined

$r_{-q} := r$
for i decreasing from q to 1 **do**
  $(r_{-i+1}, t'_i) := W(t_i, r_{-i})$
$r'_0 := \widetilde{\partial}(r_0)$
for i increasing from 1 to p **do**
  $(s'_i, r'_i) := W(r'_{i-1}, s_i)$
$t'_0 := \partial(s'_1)$
return $\overline{t'_q}|\cdots|\overline{t'_1}|\overline{t'_0}|s'_2|\cdots|s'_p|r'_p$

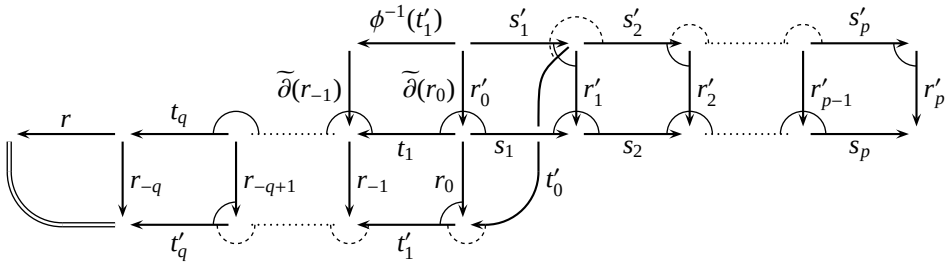


Figure 14: Left-division by an element of $\mathcal{S}^\sharp$ starting from a symmetric $\mathcal{S}$-normal path $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$.

Proposition 5.19. *Assume that $\mathcal{S}$ is a Garside family that is bounded by a map Δ in an Ore category C. Then Algorithm 5.18 running on a symmetric $\mathcal{S}$-normal decomposition of an element g of $\mathcal{E}nv(C)$ and r in $\mathcal{S}^\sharp$ returns a symmetric $\mathcal{S}$-normal decomposition of $r^{-1}g$ if the latter is defined.*

Proof. By construction, the diagram of Figure 14 is commutative, so the returned path is equivalent to $\overline{r}|\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$, hence it is a decomposition of $r^{-1}g$, and its entries are elements of $\mathcal{S}^\sharp$. So the point is to check that the path is symmetric $\mathcal{S}$-greedy.

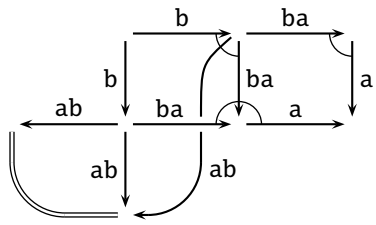
First, as $t_i|t_{i+1}$ is $\mathcal{S}$-normal for $i = 1, \ldots, q-1$, the second domino rule (Lemma 1.32) implies that $t'_i|t'_{i+1}$ is $\mathcal{S}$-normal as well. Similarly, as $s_i|s_{i+1}$ is $\mathcal{S}$-normal for $i = 1, \ldots, p-1$, the first domino rule (Lemma 1.17) implies that $s'_i|s'_{i+1}$ is $\mathcal{S}$-normal as well.

So it remains to check that $t'_0|t'_1$ is $\mathcal{S}$-normal and that t'_0 and s'_2 are left-disjoint. Consider $t'_0|t'_1$. Assume that s is an element of $\mathcal{S}^\sharp$ satisfying $s \preccurlyeq \phi^{-1}(t'_1)$ and $s \preccurlyeq s'_1$. A fortiori, we have $s \preccurlyeq \phi^{-1}(t'_1)\widetilde{\partial}(r_{-1})$ and $s \preccurlyeq s'_1 r'_1$, that is, $s \preccurlyeq r'_0 t_1$ and $s \preccurlyeq r'_0 s_1$. By assumption, t_1 and s_1 are left-disjoint, hence we deduce $s \preccurlyeq r'_0$. Now, by assumption, $r_0|t'_1$ is $\mathcal{S}$-normal, hence, by Lemma 1.31, $\partial(r_0)$ and t'_1 have no nontrivial common left-divisor. As ϕ and, therefore, ϕ^{-1} is an automorphism (Lemma 1.30), it follows that $\phi^{-1}(\partial(r_0))$ and $\phi^{-1}(t'_1)$ have no nontrivial common left-divisor. Now, by definition, $\phi^{-1}(\partial(r_0))$ is r'_0. As we have $s \preccurlyeq \phi^{-1}(t'_1)$ and $s \preccurlyeq r'_0$, we deduce that s is invertible and, therefore, $\phi^{-1}(t'_1)$ and s'_1 have no nontrivial common left-divisor. Applying the

automorphism ϕ, we deduce that t'_1 and $\phi(s'_1)$, that is, t'_1 and $\partial(t'_0)$, have no nontrivial common left-divisor. By Lemma 1.31 again, this implies that $t'_0|t'_1$ is $\mathcal{S}$-normal.

Finally, again by Lemma 1.31, the assumption that $s'_1|s'_2$ is $\mathcal{S}$-normal implies that $\partial(s'_1)$ and s'_2 are left-disjoint: by definition, $\partial(s'_1)$ is t'_0, so t'_0 and s'_2 are left-disjoint. So $\overline{t'_q}|\cdots|\overline{t'_1}|\overline{t'_0}|s'_2|\cdots|s'_p|r'_p$ is indeed a symmetric $\mathcal{S}$-normal path. □

Example 5.20. Consider $(\mathsf{ab})^{-1}(\mathsf{ba}^2)$ in B_3^+ once more. Suppose that we found that $\mathsf{ba}|\mathsf{a}$ is an $\mathcal{S}$-normal decomposition of ba^2, and we wish to find a (the) $\mathcal{S}$-normal decomposition of $(\mathsf{ab})^{-1}(\mathsf{ba}^2)$. Then applying Algorithm 5.18 amounts to completing the diagram on the right, in which of course we read the expected symmetric normal decomposition $\overline{\mathsf{ab}}|\mathsf{ba}|\mathsf{a}$.



Example 5.21. Consider again the wreathed free Abelian monoid $\mathcal{T}$. We saw in Example 5.13 that the path $w = \overline{\mathsf{ae}}|\mathsf{be}|\mathsf{ae}$ is a symmetric $\mathcal{S}$-normal decomposition of the element $g = [w]$.

Applying Algorithm 5.18 with the □-witness W from Example 4.2 and noting $\widetilde{\partial}(\Delta\mathsf{e}) = \mathsf{e}$ and $\partial(\mathsf{ae}) = \mathsf{ae}$, we can conclude that a symmetric $\mathcal{S}$-normal decomposition of the element $\mathsf{a}^{-1}g$ is given by $\overline{\mathsf{ae}}|\mathsf{be}|\mathsf{e}$.

If, in Algorithm 5.18, r is $\Delta(x)$, then each element r_{-i} is of the form $\Delta(\text{-})$, so that t'_i is simply $\phi(t_i)$, so that r'_0 is trivial, and so are all elements r_i with $i \geqslant 0$. We immediately deduce

Corollary 5.22. *Assume that $\mathcal{S}$ is a Garside family that is bounded by a map Δ in an Ore category C. If $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ is a symmetric $\mathcal{S}$-normal decomposition of g, then a symmetric $\mathcal{S}$-normal decomposition of $\Delta(x)^{-1}g$ (where x is the source of g) is $\overline{\phi(t_q)}|\cdots|\overline{\phi(t_1)}|\overline{\partial(s_1)}|s_2|\cdots s_p$.*

5.2. Computing Δ-normal decompositions

When a Garside family is bounded, alternative distinguished decompositions for the elements of the ambient category and its groupoid of fractions arise, namely the Δ-normal decompositions. We now describe incremental methods for computing such decompositions, namely algorithms that, when starting from a Δ-normal decomposition of an element g and an element r of the considered Garside family, return Δ-normal decompositions of rg and $r^{-1}g$, respectively.

Algorithm 5.23 (Left-multiplication for Δ-normal—see Figure 15).

Context: A Garside family $\mathcal{S}$ that is bounded by a map Δ in an Ore category C, a □-witness W for $\mathcal{S}^\sharp$, an $=^\times$-test E in C

Input: A Δ-normal decomposition $\Delta^{[n]}(x)|s_1|\cdots|s_p$ of an element g of $\mathcal{E}nv(C)$, an element r of $\mathcal{S}^\sharp$

Output: A Δ-normal decomposition of rg, if the latter is defined

```
r_0 := φ^n(r)
for i increasing from 1 to p do
    (s'_i, r_i) := W(r_{i-1}, s_i)
if E(Δ(-), s'_1) ≠ ⊥ then                    ▷ s'_1 is Δ-like
    s'_2 := E(Δ(-), s'_1)s'_2
    return Δ^[n+1](-)|s'_2|···|s'_p|r_p
else                                         ▷ s'_1 is not Δ-like
    return Δ^[n](-)|s'_1|···|s'_p|r_p
```

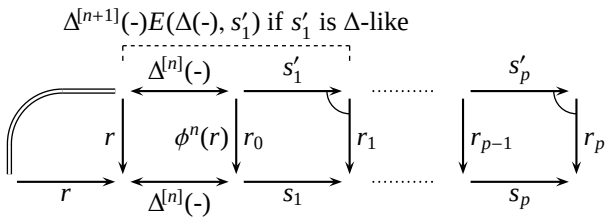


Figure 15: Computing a Δ-normal decomposition of rg starting from a Δ-normal decomposition of g: the $|n|$ entries of the type $\Delta(\text{-})$ are treated directly using ϕ^n, and then it just remains to left-multiply a (positive) Δ-normal sequence by an element of $\mathcal{S}$ using the standard method. Note that, depending on the sign of n, the diagrams for $\Delta^{(n)}(\text{-})$ consists either of n right-oriented arrows (case $n \geqslant 0$) or of $|n|$ left-oriented arrows (case $n \leqslant 0$).

Example 5.24. Consider again the wreathed free Abelian monoid $\mathcal{T}$. If follows from Example 5.13 that the path $w = \mathtt{be}|\mathtt{ae}$ is a Δ-normal decomposition of the element $g = [w]$.

Applying Algorithm 5.23 with the $\square$-witness W from Example 4.2 and the $\overset{\times}{=}$-test E on $\mathcal{S}^\sharp$ from Example 4.17, we can conclude that $\Delta|\mathtt{be}|\mathtt{e}$ is a Δ-normal decomposition of the element $\mathtt{a}g$.

Algorithm 5.25 (Left-division for Δ-normal—see Figure 16).

Context: A Garside family $\mathcal{S}$ that is bounded by a map Δ in an Ore category C, a $\square$-witness W for $\mathcal{S}^\sharp$, an $\overset{\times}{=}$-test E in C

Input: A Δ-normal decomposition $\Delta^{[n]}(x)|s_1|\cdots|s_p$ for an element r of $\mathcal{E}nv(C)$ and an element r of $\mathcal{S}^\sharp$ with source x

Output: An Δ-normal decomposition of $r^{-1}g$

$r_0 := \phi^n(\widetilde{\partial}(r))$
for i increasing from 1 to p **do**
  $(s'_i, r_i) := W(r_{i-1}, s_i)$
if $E(\Delta(\text{-}), s'_1) \neq \perp$ **then** ▷ s'_1 is Δ-like
  $s'_2 := E(\Delta(\text{-}), s'_1)s'_2$
  return $\Delta^{[n]}(\text{-})|s'_2|\cdots|s'_p|r_p$
else ▷ s'_1 is not Δ-like
  return $\Delta^{[n-1]}(\text{-})|s'_1|\cdots|s'_p|r_p$

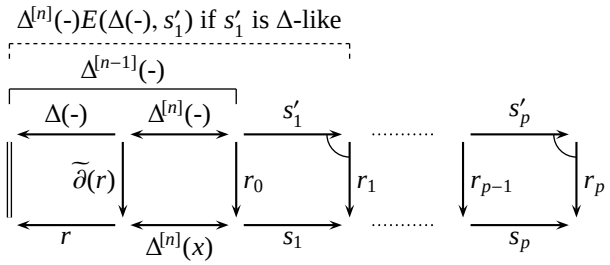


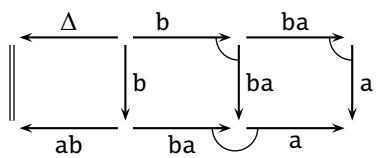
Figure 16: Algorithm 5.25 : Computing a Δ-normal decomposition for $r^{-1}g$ from a Δ-normal decomposition of g.

Proposition 5.26. *Assume that $\mathcal{S}$ is a Garside family that is bounded by a map Δ in an Ore category C, W is a $\square$-witness on $\mathcal{S}$, and E is an $\overset{\times}{=}$-test in C. Then Algorithm 5.23 (resp. Algorithm 5.25) running on a Δ-normal decomposition of an element g of $\mathcal{E}nv(C)$ and r in $\mathcal{S}^\sharp$ returns a Δ-normal decomposition of rg (resp. $r^{-1}g$) (when the latter is defined).*

Proof. Consider first Algorithm 5.23. Then the diagram of Figure 15 is commutative as, in particular, $r\Delta^{(n)}(\text{-}) = \Delta^{(n)}(u)\phi^n(r)$ holds whenever defined. As usual, the first domino rule implies that $s'_1|\cdots|s'_p$ is $\mathcal{S}$-normal and, by construction, $s'_p|r_p$ is $\mathcal{S}$-normal. So it just remains to consider the relation between $\Delta^{[n]}(\text{-})$ and s'_1. Now, if s'_1 is not Δ-like, $\Delta^{[n]}(\text{-})|s'_1$ is Δ-normal. On the other hand, if s'_1 is Δ-like, then, by definition, we have $s'_1 = \Delta(\text{-})E(\Delta(\text{-}), s'_1)$, so that $\Delta(\text{-})$ can be incorporated to $\Delta^{[n]}(\text{-})$ to form $\Delta^{[n+1]}(\text{-})$ and $E(\Delta(\text{-}), s'_1)$ can be incorporated to s'_2 to form a new element of $\mathcal{S}^\sharp$ as $C^\times\mathcal{S}^\sharp \subseteq \mathcal{S}^\sharp$ holds. Then $\Delta^{[n+1]}(\text{-})|s'_2$ must be Δ-normal because the new element s'_2 cannot be Δ-like. Indeed, $\Delta(\text{-}) \preccurlyeq s'_2$ would imply $\Delta^{(2)}(\text{-}) \preccurlyeq s'_1s'_2$, whence $\Delta^{(2)}(\text{-}) \preccurlyeq s'_1s'_2r_2 = r_0s_1s_2$. Now, as r_0 lies in $\mathcal{S}^\sharp$, we can write $\Delta^{(2)}(\text{-}) = r_0\partial(r_0)\Delta(\text{-}) = r_0\Delta(\text{-})\phi(\partial(r_0))$, so $\Delta^{(2)}(\text{-}) \preccurlyeq r_0s_1s_2$ implies $r_0\Delta(\text{-})\phi(\partial(r_0)) \preccurlyeq r_0s_1s_2$, whence $\Delta(\text{-}) \preccurlyeq s_1s_2$, contradicting the assumption that $\Delta^{[n]}(\text{-})|s_1$ is Δ-normal. So no cascade may occur here.

The argument for Algorithm 5.25 is entirely similar. The only new point is that, in the diagram of Figure 16, the left square is commutative because $\widetilde{\partial}(r)r = \Delta(\text{-})$ holds for every r in $\mathcal{S}^\sharp$. □

Example 5.27. As in Example 5.20, we assume that we know the Δ-normal decomposition $\mathtt{ba}|\mathtt{a}$ of $\mathtt{b}^2\mathtt{a}$, and that we would like to find a (the) Δ-normal decomposition of $(\mathtt{ab})^{-1}(\mathtt{b}^2\mathtt{a})$. Applying Algorithm 5.25 amounts to completing the diagram on the right, and we read that the expected decomposition is $\Delta^{[-1]}|\mathtt{b}|\mathtt{ba}|\mathtt{a}$.



Example 5.28. Consider again the wreathed free Abelian monoid $\mathcal{T}$. If follows from Example 5.13 that the path $w = \mathtt{be}|\mathtt{ae}$ is a Δ-normal decomposition of the element $g = [w]$.

Applying Algorithm 5.23 with the □-witness W from Example 4.2 and the $=^\times$-test E on $\mathcal{S}^\sharp$ from Example 4.17, we can conclude that a Δ-normal decomposition of the element $\mathtt{a}^{-1}g$ is given by $\Delta^{[-1]}|\mathtt{b}|\mathtt{be}|\mathtt{ae}$.

5.3. Solving the Word Problem

As in the positive case, we shall describe several solutions for the Word Problem in the involved groupoid of fractions

Using symmetric normal decompositions

First, symmetric normal decompositions give a direct solution extending Algorithm 4.14 to the signed case.

Algorithm 5.29 (Word Problem, general case I).

Context: A strong Garside family $\mathcal{S}$ in an Ore category C that admits left-lcms, a □-witness W, a short left-lcm selector $\widetilde{\theta}$, and an $=^\times$-test E on $\mathcal{S}^\sharp$

Input: A signed $\mathcal{S}^\sharp$-path w

Output: `true` if w represents an element 1_x in $\mathcal{E}nv(C)$, and `false` otherwise

```
x := src(w) ; y := trg(w)
if x ≠ y then
    return false
else
    use Algorithm 5.10 to find a symmetric S-normal path ū|v that represents [w]
    return the value of CompareNormalPaths(u, v)    (Algorithm 4.14)
```

Proposition 5.30. *Assume that $\mathcal{S}$ is a strong Garside family $\mathcal{S}$ in an Ore category C that admits left-lcms, W is a $\square$-witness on $\mathcal{S}$, $\widetilde{\theta}$ is a short left-lcm selector on $\mathcal{S}^\sharp$, and E is an $=^\times$-test on $\mathcal{S}^\sharp$. Then Algorithm 5.29 running on a signed $\mathcal{S}^\sharp$-path w of length at most ℓ decides in time $O(\ell^2)$ whether $[w]$ is an identity-element in $\mathcal{E}nv(C)$.*

Proof. By Proposition 5.11, line 5 of Algorithm 5.29 computes a symmetric $\mathcal{S}$-normal path $\overline{u}|v$ representing $[w]$ in time $O(\ell^2)$. The signed path w represents an identity-element in $\mathcal{E}nv(C)$ if and only if the (positive) paths u and v represent the same (invertible) element in C. The claim then follows with Proposition 4.15. $\square$

Example 5.31. Consider again the wreathed free Abelian monoid $\mathcal{T}$ with the $\square$-witness W from Example 4.2, the $=^\times$-test E on $\mathcal{S}^\sharp$ from Example 4.17, and the short right-lcm selector θ as well as the short left-lcm selector $\widetilde{\theta}$ from Example 4.29.

Running Algorithm 5.10 with $\square$, θ and $\widetilde{\theta}$ on the signed $\mathcal{S}^\sharp$-path $w = \overline{\Delta}|\overline{\mathtt{a}}|\overline{\mathtt{b}}|\overline{\mathtt{a}}|\mathtt{ae}|\mathtt{b}|\Delta|\mathtt{ae}$, and removing trivial factors in every step, yields the symmetric $\mathcal{S}$-normal decomposition $\overline{u}|v$ of $[w]$ with $u = \mathtt{e}|\mathtt{e}$ and $v = \varepsilon$. Subsequent application of the function CompareNormalPaths from Algorithm 4.14 using E establishes that u and v are equivalent, so we conclude $[w] = 1$.

Using Δ-normal decompositions

According to Proposition 1.36, when they exist, Δ-normal decompositions enjoy the same uniqueness property as symmetric normal decompositions. Therefore, in the case when the reference Garside family is not only strong but even bounded, Algorithm 5.29 and Proposition 5.30 admit exact counterparts where symmetric normal is replaced with Δ-normal. We skip the details.

Using reversing

As in the positive case, at least when the ambient category contains no nontrivial invertible element and admits lcms, we can also solve the Word Problem by using reversing and thus avoiding to compute distinguished decompositions. To stick to the context described in Subsection 2.1 we assume here that the ambient category admits left- and right-lcms and contains no nontrivial invertible element, which amounts to requiring that the lcms are unique.

Algorithm 5.32 (Word Problem, general case II).

Context: A strong Garside family $\mathcal{S}$ in an Ore category C that admits unique right- and left-lcms, the right-lcm selector θ on $\mathcal{S} \cup \mathbf{1}_C$ and the left-lcm selector $\widetilde{\theta}$ on $\mathcal{S} \cup \mathbf{1}_C$

Input: A signed $\mathcal{S}$-path w

Output: `true` if w represents an element 1_x in $\mathcal{E}nv(C)$, and `false` otherwise

1: right-θ-reverse w into a positive–negative path $u|\overline{v}$ using Algorithm 2.6
2: left-$\widetilde{\theta}$-reverse $u|\overline{v}$ into a negative–positive path $\overline{u'}|v'$ using (the left counterpart of) Algorithm 2.3
3: **return** $[\![u' = v' = \varepsilon_-]\!]$

Lemma 2.10, Proposition 1.16, and their left counterparts imply:

Proposition 5.33. *Assume that $\mathcal{S}$ is a strong Garside family $\mathcal{S}$ in an Ore category C that admits unique right- and left-lcms.*

(i) *Algorithm 5.32 solves the Word Problem of $\mathcal{E}nv(C)$ with respect to $\mathcal{S}$.*

(ii) *If $\mathcal{S}$ is finite, the complexity of Algorithm 5.32 is quadratic in the length of the input path.*

Appealing to a right- and a left-reversing, Algorithm 5.32 is nicely symmetric. However, it requires the existence of right-lcms, which is not guaranteed in every Ore category that admits a strong Garside family. Actually, this assumption is superfluous, since a double left-reversing can be used instead.

Algorithm 5.34 (Word Problem, general case III).

Context: A strong Garside family $\mathcal{S}$ in an Ore category C that admits unique left-lcms, the left-lcm selector $\widetilde{\theta}$ on $\mathcal{S} \cup \mathbf{1}_C$

Input: A signed $\mathcal{S}$-path w

Output: `true` if w represent an element 1_x in $\mathcal{E}nv(C)$, and `false` otherwise

1: left-$\widetilde{\theta}$-reverse w into a negative–positive $\overline{u}|v$ using (the left counterpart of) Algorithm 2.6

2: left-$\widetilde{\theta}$-reverse $v|\overline{u}$ into a negative–positive path $\overline{u'}|v'$ using (the left counterpart of) Algorithm 2.3

3: **return** $[\![u' = v' = \varepsilon_{-}]\!]$

Proposition 5.35. *Assume that $\mathcal{S}$ is a strong Garside family in an Ore category C that admits unique left-lcms.*

(i) *Algorithm 5.34 solves the Word Problem of $\mathcal{E}nv(C)$ with respect to $\mathcal{S}$.*

(ii) *If $\mathcal{S}$ is finite, the complexity of Algorithm 5.34 is quadratic in the length of the input path.*

Proof. The signed path w represents an identity-element in $\mathcal{E}nv(C)$ if and only if the paths u and v represent the same element in C. By the counterpart of Proposition 1.16, left-$\widetilde{\theta}$-reversing is complete and terminating, so u and v represent the same element of C if and only if $v|\overline{u}$ is left-$\widetilde{\theta}$-reversible to an empty path. By Lemma 2.10, the complexity of Algorithm 5.34 is in $O(\|w\|^2)$. □

Example 5.36. Running on the signed word $w = \overline{\mathtt{ab}}|\mathtt{ba}|\mathtt{a}$, Algorithm 5.32 right-reverses w into $\mathtt{a}|\mathtt{ab}|\overline{\mathtt{ba}}$, and then left-reverses the latter words back to $\overline{\mathtt{ab}}|\mathtt{ba}|\mathtt{a}$. The final word is not empty, hence w does not represent 1 in the group B_3.

Running on w, Algorithm 5.34 first left-reverses w into itself (since w is a negative–positive word), then switches the numerator and the denominator into $\mathtt{ba}|\mathtt{a}|\overline{\mathtt{ab}}$, and finally left-reverses the latter word. One finds now $\overline{\mathtt{a}}|\mathtt{b}|\mathtt{b}$, a nonempty word, and one concludes again that w does not represent 1.

Remark 5.37. As in Section 4, we could also state a result referring to an arbitrary generating family $\mathcal{A}$ that satisfies convenient properties but that is not necessarily a strong Garside family.

5.4. Computing decompositions for an inverse

We complete the analysis of our distinguished decompositions with explicit methods for finding a decomposition of g^{-1} from one of g.

In the case of symmetric normal decompositions, the result is trivial:

Proposition 5.38. *Assume that C is a left-Ore category and that $\mathcal{S}$ is a Garside family in C. If $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ is a symmetric $\mathcal{S}$-normal decomposition for an element g of $\mathcal{E}nv(C)$, then $\overline{s_p}|\cdots|\overline{s_1}|t_1|\cdots|t_q$ is a symmetric $\mathcal{S}$-normal decomposition for g^{-1}.*

Proof. The definition makes it obvious that $\overline{s_p}|\cdots|\overline{s_1}|t_1|\cdots|t_q$ is symmetric $\mathcal{S}$-normal whenever $\overline{t_q}|\cdots|\overline{t_1}|s_1|\cdots|s_p$ is. On the other hand, $g = t_q^{-1}\cdots t_1^{-1}s_1\cdots s_p$ implies $g^{-1} = s_p^{-1}\cdots s_1^{-1}t_1\cdots t_q$ in $\mathcal{E}nv(C)$. □

The case of Δ-normal decompositions is more complicated, as the definition is not symmetric. However, finding an explicit formula is not difficult. The following result is an adaptation of the well known analogous result for braid groups respectively Garside groups (see, for instance, El-Rifai and Morton, 1994, proof of Proposition 4.5).

Proposition 5.39. *Assume that $\mathcal{S}$ is a Garside family that is bounded by Δ in a cancellative category C. If $\Delta^{(n)}|s_1|\cdots|s_p$ is a Δ-normal decomposition for an element g of $\mathcal{E}nv(C)$, then $\Delta^{(-n-p)}|\partial(\phi^{-n-p}(s_p))|\cdots|\partial(\phi^{-n-1}(s_1))$ is a Δ-normal decomposition for g^{-1}.*

Proof. We first check that the sequence mentioned in the statement is Δ-normal. So let $i < p$. By hypothesis, $s_i|s_{i+1}$ is $\mathcal{S}$-normal. As ϕ is an automorphism of $\mathcal{S}$, the path $\phi^{-n-i}(s_i)|\phi^{-n-i}(s_{i+1})$ is also $\mathcal{S}$-normal. By Lemma 1.31, this implies that $\partial(\phi^{-n-i}(s_i))$ and $\phi^{-n-i}(s_{i+1})$ are left-disjoint. The latter element is $\phi(\phi^{-n-i-1}(s_{i+1}))$, hence it is $\partial(\partial(\phi^{-n-i-1}(s_{i+1})))$. Reading the above disjointness result in a symmetric way and applying Lemma 1.31 once more, we can deduce that $\partial(\phi^{-n-i-1}(s_{i+1}))|\partial(\phi^{-n-i}(s_i))$ is $\mathcal{S}$-normal. Hence $\partial(\phi^{-n-p}(s_p))|\cdots|\partial(\phi^{-n-1}(s_1))$ is $\mathcal{S}$-normal.

Moreover, the hypothesis that s_1 is not Δ-like implies that $\phi^{-n-1}(s_1)$ is not Δ-like either, and, therefore, $\partial(\phi^{-n-1}(s_1))$ is not invertible. On the other hand, the hypothesis that s_p is not invertible implies that $\phi^{-n-p}(s_p)$ is not invertible either, and, therefore, $\partial(\phi^{-n-p}(s_p))$ is not Δ-like. Hence, we conclude that the path $\Delta^{(-n-p)}|\partial(\phi^{-n-p}(s_p))|\cdots|\partial(\phi^{-n-1}(s_1))$ is Δ-normal.

It remains to check that the latter Δ-normal path is a decomposition of g^{-1}. Now, by construction, the maps ϕ and ∂ commute and, pushing the factor $\Delta^{(n)}$ to the left, we obtain the equality

$$\begin{aligned}\Delta^{(-n-p)}(\text{-})\,\partial(\phi^{-n-p}(s_p))\cdots\partial(\phi^{-n-1}(s_1))\cdot\Delta^{(n)}(\text{-})\,s_1\cdots s_p\\ = \Delta^{(-n-p)}(\text{-})\,\Delta^{(n)}(\text{-})\,\partial(\phi^{-p}(s_p))\cdots\partial(\phi^{-1}(s_1))\,s_1\cdots s_p,\end{aligned}\tag{5.40}$$

and our goal is to prove that this element is 1_y. Let us call the right term of (5.40) $E(n, s_1, ..., s_p)$. We shall prove using induction on $p \geqslant 0$ that an expression of the form $E(n, s_1, ..., s_p)$ equals 1_y, where y is the target of s_p.

Assume first $p = 0$. What remains for $E(n)$ is then $\Delta^{(-n)}(\text{-})\,\Delta^{(n)}(\text{-})$, which, by the remark following Notation 1.33, is 1_y. Assume $p \geqslant 1$. The above result enables us to gather the first two entries of E into $\Delta^{(-p)}(\text{-})$. Then $\partial(\phi^{-1}(s_1))\,s_1$, which is also $\widetilde{\partial}(s_1)\,s_1$, equals $\Delta(z)$, where z is the source of $\widetilde{\partial}(s_1)$. But, then, we can push this $\Delta(\text{-})$-factor to the left through the $\partial(\phi^{-i}(s_i))$ factors with the effect of diminishing the exponents of ϕ by one. In this way, $E(n, s_1, ..., s_p)$ becomes

$$\Delta^{(-p)}(\text{-})\,\Delta(\text{-})\,\partial(\phi^{-p+1}(s_p))\cdots\partial(\phi^{-1}(s_2))\,s_2\cdots s_p,$$

which is $E(1, s_2, ..., s_p)$. By the induction hypothesis, its value is 1_y. □

5.5. Computing upper and lower bounds

If C is an Ore category, the left-divisibility relation of C naturally extends into a relation on $\mathcal{E}nv(C)$, namely the relation, still denoted by $\preccurlyeq$, such that $f \preccurlyeq g$ holds if there exists h in C satisfying $fh = g$. This relation is a partial preordering on $\mathcal{E}nv(C)$, and it is a partial ordering if C admits no nontrivial invertible element. It directly follows from the definition that, if f, g are elements of C, then a least common upper bound (*resp.* a greatest common lower bound) of f and g with respect to $\preccurlyeq$ is (when it exists) a right-lcm (*resp.* a left-gcd) of f and g.

Lemma 5.41. *Assume that C is an Ore category. Then any two elements of $\mathcal{E}nv(C)$ with the same source admit a least common upper bound (resp. a greatest common lower bound) with respect to $\preccurlyeq$ if and only if any two elements of C with the same source admit a right-lcm (resp. a left-gcd).*

We skip the easy proof. Then a natural computation problem arises whenever the above bounds exist. The solution is easy.

Algorithm 5.42 (Least common upper bound with respect to $\preccurlyeq$).

Context: A strong Garside family $\mathcal{S}$ in an Ore category C that admits unique left- and right-lcms, the right-lcm selector θ on $\mathcal{S} \cup \mathbf{1}_C$ and the left-lcm selector $\widetilde{\theta}$ on $\mathcal{S} \cup \mathbf{1}_C$
Input: Two signed $\mathcal{S}$-paths w_1, w_2 with the same source
Output: A signed $\mathcal{S}$-path representing the least common upper bound of $[w_1]$ and $[w_2]$

1: left-$\widetilde{\theta}$-reverse $\overline{w_1}|w_2$ into a negative–positive path $\overline{u}|v$ using (the left counterpart of) Algorithm 2.6
2: right-θ-reverse $\overline{u}|v$ into a positive–negative path $v'|\overline{u'}$ using Algorithm 2.3
3: **return** $w_1|v'$

The solution for greatest lower bound is entirely symmetric.

Algorithm 5.43 (Greatest lower bound with respect to $\preccurlyeq$).

Context: A strong Garside family $\mathcal{S}$ in an Ore category C that admits unique left- and right-lcms, the right-lcm selector θ on $\mathcal{S} \cup \mathbf{1}_C$ and the left-lcm selector $\widetilde{\theta}$ on $\mathcal{S} \cup \mathbf{1}_C$
Input: Two signed $\mathcal{S}$-paths w_1, w_2 with the same source
Output: A signed $\mathcal{S}$-path representing the greatest lower bound of $[w_1]$ and $[w_2]$

1: right-θ-reverse $\overline{w_1}|w_2$ into a positive–negative path $u|\overline{v}$ using Algorithm 2.6
2: left-$\widetilde{\theta}$-reverse $u|\overline{v}$ into a negative–positive path $\overline{v'}|u'$ using (the left counterpart of) Algorithm 2.3
3: **return** $w_1|\overline{v'}$

Proposition 5.44. *Assume that $\mathcal{S}$ is a strong Garside family in an Ore category C that admits unique left- and right-lcms. Then Algorithm 5.42 (resp. Algorithm 5.43) running on two signed $\mathcal{S}$-paths w_1 and w_2 of length at most ℓ returns the least upper bound (resp. the greatest lower bound) of $[w_1]$ and $[w_2]$ with respect to $\preccurlyeq$ in time $O(\ell^2)$.*

Proof. First consider Algorithm 5.42. By the left counterpart of Lemma 2.10, the left-$\widetilde{\theta}$-reversing in line 1 takes time $O(\ell^2)$ and produces two $\mathcal{S}^\sharp$-paths u, v of length $O(\ell)$ such that, in $\mathcal{E}nv(C)$, we have $[\overline{u}|v] = [\overline{w_1}|w_2]$, hence $[w_1] = g[u]$ and $[w_2] = g[v]$ for some g, namely the common class of the signed paths $w_1|\overline{u}$ and $w_2|\overline{v}$. By Proposition 4.25, the right-θ-reversing in line 2 takes time $O(\ell^2)$ and produces two $\mathcal{S}$-paths u', v' such that $u|v'$ and $v|u'$ both represent the right-lcm of $[u]$ and $[v]$, hence their least upper bound with respect to $\preccurlyeq$. As the partial ordering $\preccurlyeq$ is invariant under left-multiplication, the claim follows.

The claim for Algorithm 5.43 follows in an analogous way from Lemma 2.10 and the left counterpart of Proposition 4.25, noting that, for all f, g in $\mathcal{E}nv(C)$ and h in C, the conditions $fh = g$ and $f^{-1} = hg^{-1}$ are equivalent. □

When we apply Algorithms 5.42 and 5.43 to a pair of positive paths, we obtain algorithms that determine right-lcms and left-gcds. In this case, Algorithm 5.42 is simply Algorithm 2.3, since the left-reversing step is trivial: the initial path $\overline{w_1}|w_2$ is directly negative–positive. Algorithm 5.43, however, has no equivalent in Section 4. Its output is a path representing the left-gcd but, in general, it need not be a positive path, although it must be equivalent to a positive path. In order to obtain a positive output, a third reversing step can be added.

Algorithm 5.45 (Left-gcd).

Context: A strong Garside family $\mathcal{S}$ in an Ore category C that admits unique left- and right-lcms, the right-lcm selector θ on $\mathcal{S} \cup \mathbf{1}_C$ and the left-lcm selector $\widetilde{\theta}$ on $\mathcal{S} \cup \mathbf{1}_C$

Input: Two $\mathcal{S}$-paths u, v with the same source

Output: An $\mathcal{S}$-path that represents the left-gcd of $[u]$ and $[v]$ in C

1: right-θ-reverse $\overline{u}|v$ into a positive–negative path $v'|\overline{u'}$ using Algorithm 2.3
2: left-$\widetilde{\theta}$-reverse $v'|\overline{u'}$ into a negative–positive path $\overline{u''}|v''$ using (the left counterpart of) Algorithm 2.3
3: left-$\widetilde{\theta}$-reverse $u|\overline{u''}$ into a positive path w using (the left counterpart of) Algorithm 2.3
4: **return** w

Proposition 5.46. *Assume that $\mathcal{S}$ is a strong Garside family in an Ore-category that admits unique left- and right-lcms. Then Algorithm 5.45 running on two $\mathcal{S}$-paths u, v of length at most ℓ returns the left-gcd of $[u]$ and $[v]$ in time $O(\ell^2)$.*

Proof. It follows from Proposition 5.44 that, for u'' as computed in line 2, the path $u|\overline{u''}$ represents the left-gcd of $[u]$ and $[v]$. As u and v are both positive, this path $u|\overline{u''}$ must be equivalent to some positive path, say u_0. Then the signed path $u|\overline{u''}|\overline{u_0}$ represents an identity-element in $\mathcal{E}nv(C)$, so the positive paths u and u_0u'' are equivalent. Since, by the counterpart of Proposition 1.16, left-reversing is complete for the considered presentation, the path $u|\overline{u''}|\overline{u_0}$ is left-reversible to an empty path, that is, the left-reversing grid constructed from $u|\overline{u''}|\overline{u_0}$ has empty arrows everywhere on the left and on the top. This implies in particular that the subgrid corresponding to left-reversing $u|\overline{u''}$ has empty arrows on the left, that is, with our current notation, that the word w computed at line 5 is positive (and equivalent to u_0).

By Proposition 5.44, the computation time of w lies in $O(\ell^2)$ since, by the left counterpart of Lemma 2.4, the left-$\widetilde{\theta}$-reversing in line 3 takes time $O(\ell^2)$. □

Example 5.47. Let us consider a last time the monoid B_3^+ and the elements represented by $u = \mathsf{a}|\mathsf{b}|\mathsf{b}$ and $v = \mathsf{b}|\mathsf{a}|\mathsf{b}|\mathsf{b}$. As seen in Figure 10, right-reversing $\overline{u}|v$ using the right-lcm selector leads to the positive–negative word $\mathsf{ab}|\overline{\mathsf{a}}$. Then left-reversing the latter word using the left-lcm selector leads to the negative–positive word $\overline{\mathsf{b}}|\mathsf{ab}$. Finally, left-reversing $\mathsf{a}|\mathsf{b}|\mathsf{b}|\overline{\mathsf{b}}$ gives the positive word $\mathsf{a}|\mathsf{b}$, and we conclude that the left-gcd of $[u]$ and $[v]$ in B_3^+ is ab. Note that, instead of using $\mathcal{S}$-words and the associated lcm selector, we could instead use $\mathcal{A}$-words and the initial right-complement associated with the Artin presentation: the uniqueness of the final result guarantees that the successive involved words must be pairwise equivalent.

References

Adyan, S.I., 1984. Fragments of the word Δ in the braid group. Mat. Zametki 36, 25–34.

Artin, E., 1947. Theory of braids. Ann. of Math. (2) 48, 101–126.

Bessis, D., 2006a. Finite complex reflection arrangements are $K(\pi, 1)$. arXiv:math/0610777.

Bessis, D., 2006b. Garside categories, periodic loops and cyclic sets. arXiv:math/0610778.

Brieskorn, E., Saito, K., 1972. Artin-Gruppen und Coxeter-Gruppen. Invent. Math. 17, 245–271.

Charney, R., 1992. Artin groups of finite type are biautomatic. Math. Ann. 292, 671–683.

Clifford, A.H., Preston, G.B., 1961. The algebraic theory of semigroups. Vol. I. Mathematical Surveys, No. 7, American Mathematical Society, Providence, R.I.

Dehornoy, P. with Digne, F., Godelle, E., Krammer, D., Michel, J., 2013a. Foundations of Garside theory. URL: `http://www.math.unicaen.fr/~garside/Garside.pdf`.

Dehornoy, P., 1997. Groups with a complemented presentation. J. Pure Appl. Algebra 116, 115–137. Special volume on the occasion of the 60th birthday of Professor Peter J. Freyd.

Dehornoy, P., 2000. Braids and self-distributivity. volume 192 of *Progress in Mathematics*. Birkhäuser Verlag, Basel.
Dehornoy, P., 2002. Groupes de Garside. Ann. Sci. École Norm. Sup. (4) 35, 267–306.
Dehornoy, P., 2003. Complete positive group presentations. J. Algebra 268, 156–197.
Dehornoy, P., 2008. Efficient solutions to the braid isotopy problem. Discrete Appl. Math. 156, 3091–3112.
Dehornoy, P., 2011. The subword reversing method. Internat. J. Algebra Comput. 21, 71–118.
Dehornoy, P., 2012. Monoids of O-type, subword reversing, and ordered groups. J. of Group Theory, to appear; arXiv:1204.3211.
Dehornoy, P., Digne, F., Michel, J., 2013b. Garside families and Garside germs. J. Algebra 380, 109–145.
Dehornoy, P., Paris, L., 1999. Gaussian groups and Garside groups, two generalisations of Artin groups. Proc. London Math. Soc. (3) 79, 569–604.
Deligne, P., 1972. Les immeubles des groupes de tresses généralisés. Invent. Math. 17, 273–302.
Digne, F., Michel, J., 2006. Garside and locally garside categories. arXiv:math/0612652.
El-Rifai, E.A., Morton, H.R., 1994. Algorithms for positive braids. Quart. J. Math. Oxford Ser. (2) 45, 479–497.
Epstein, D.B.A., Cannon, J.W., Holt, D.F., Levy, S.V.F., Paterson, M.S., Thurston, W.P., 1992. Word processing in groups. Jones and Bartlett Publishers, Boston, MA.
Garside, F.A., 1969. The braid group and other groups. Quart. J. Math. Oxford Ser. (2) 20, 235–254.
Gebhardt, V., 2005. A new approach to the conjugacy problem in Garside groups. J. Algebra 292, 282–302.
Gebhardt, V., González-Meneses, J., 2010. Solving the conjugacy problem in Garside groups by cyclic sliding. J. Symbolic Comput. 45, 629–656.
Krammer, D., 2008. A class of Garside groupoid structures on the pure braid group. Trans. Amer. Math. Soc. 360, 4029–4061.
Lévy, A., 1979. Basic set theory. Springer-Verlag, Berlin.
Thurston, W., 1988. Finite state algorithms for the braid group. Circulated notes.