Formalization of Basic Combinatorics on Words

Anonymous Author(s)*

Abstract

Combinatorics on Words is a rather young domain encompassing the study of words and formal languages. An archetypal example of a task in Combinatorics on Words is to solve the equation $x \cdot y = y \cdot x$, i.e., to describe words that commute. The choice of the multiplication $\cdot$ for the concatenation operation suggests that we tend to see the set of words as an algebraic structure, namely a monoid.

This paper provides a sample of our project devoted to formalization of Combinatorics on Words, starting from basic facts, and focusing mainly on equations over words (and therefore on finite words). Our work is set up on existing tools in Isabelle/HOL, namely on the ubiquitous and well-developed datatype of lists. From the point of view of equations, nevertheless, the standard library does not reach much beyond the solution of the above mentioned commutation.

This contribution contains formalization of two moderately advanced topics, namely i) the solution of the famous equation $x^a \cdot y^b = z^c$ with $2 \leq a, b, c$, known as the Lyndon-Schützenberger Equation (LSE); and ii) an important result known as the Graph Lemma (GL), which is closely related to the Defect Theorem (DT), namely, it yields a generic upper bound on the rank of a solution of a system of equations.

The LSE represents the more combinatorial aspect of the field, and uses the (weak version) of the basic result about periods, the Periodicity Lemma (also known as the Fine and Wilf theorem). On the other hand, GL has a more algebraic flavour and uses the concept of the free hull of a given set of words as the main ingredient.

Finally, the submission is accompanied by an evolving toolkit of several hundreds auxiliary results which provide for a relatively smooth reasoning within more complex tasks.

1 Introduction

Combinatorics on Words usually dates its beginning (cf. [2]) back to the works of Axel Thue on repetitions in infinite words published more than hundred years ago [19, 20]. Nevertheless, the first monograph on the subject was published only in 1983 [11], and it is listed in the Mathematics Subject Classification since 2009 (as 68R15). In this paper, we are interested in the part of the field dealing with finite (rather than infinite) words, which in particular includes solving word equations. Solving general word equations is a difficult algorithmic task. Once believed to be undecidable, the first algorithm was described by Makanin in 1977 [14] (see [12] for a survey). Currently, the approach of *recompression* introduced by Arthur Jeż [9] is the most efficient one with PSPACE complexity. While the problem is NP hard, it remains a challenging open question whether it is NP complete.

We believe that combinatorics of (finite) words is an area where computer assisted formalization may be very helpful. Proofs of even fairly simple results tend to be tedious and repetitive, featuring complicated analysis of cases, which makes them hard (both for referees and readers) to verify. Moreover, despite the short history of the field, basic auxiliary results are sometimes forgotten and rediscovered, or simply repeatedly proven in many papers. Some easily stated problems, like the solution of equations in three unknowns [15], or the characterization of binary equality languages [8], are vast classification tasks resembling much more prominent projects like the classification of finite groups [4], four-colour problem [16] or Kepler's conjecture [6].

In this paper, we present two moderately advanced results, which together reveal main features of the general project of formalization of word equations. The first result is the solution of the equation $x^a \cdot y^b = z^c$ with $2 \leq a, b, c$, namely, a proof of the fact that this equation admits trivial solutions only, that is, solutions where all three unknown words are powers of a common root. This was first proven by Lyndon and Schützenberger [13] in a more general setting of free groups, and represents a historically first nontrivial result for equations with three unknowns. The proof is not obvious even in free monoids, and we present here its formalization in Isabelle/HOL.

The solution of the Lyndon-Schützenberger Equation (LSE) is mainly combinatorial. One of its main ingredients is the Periodicity Lemma (PL), also known as the theorem of Fine and Wilf [5]. Although neither the PL is trivial, it shares the status of an auxiliary result with several hundreds other results of our formalization, providing a background for more advanced results.

The need to deal with equations like the LSE in an *ad hoc* manner is tightly related to the fact that word equations are rather immune against the so called defect effect. To understand what this means, consider linear equations. Each

CPP 2021, 18–19 January 2021, virtual
© 2020 Association for Computing Machinery.
ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

new independent linear equation decreases the degree of freedom of a solution of the corresponding system, so that n independent equations over n unknowns admit only the trivial solution. In contrast, there is no known upper bound on the size of an independent systems of word equations over $n \geq 4$ unknowns.

The best general form of the Defect Theorem (DT) for word equations is provided by the Graph Lemma (GL), which is the second main result presented and formalized in this paper. We adopt the approach to GL which exploits the algebraic concept of the *free hull* of a solution, and of its rank, that is, of the cardinality of its basis. The corresponding background theory represents the second, algebraic pillar of our development, which deals with sets of words closed under concatenation, that is, with submonoids of the underlying free monoid of words. It is immediate that (unlike in the free group case) submonoids of the free monoid are not always free. Nevertheless, each submonoid M has a free hull $\langle M \rangle_{\mathrm{F}}$, the unique smallest free submonoid containing M. If we see a solution of a given system of word equations as a basis of a monoid M, then GL limits the size of the basis of $\langle M \rangle_{\mathrm{F}}$, which is in particular always less than the number of unknowns (if the system is nontrivial).

Our formalization of the above mentioned results in Isabelle/HOL is based on the fundamental and well developed datatype of lists. Nevertheless, from the point of view of word equations, the main library contains only the solution of the easiest nontrivial word equation, namely $x \cdot y = y \cdot x$, showing that commuting words x and y are always powers of the same (shorter) word. Note, in this respect, that GL in particular implies that *all* nontrivial equations over two unknowns have this property. In fact, the main library does not provide any support for seeing lists as a free monoid. From this point of view the corresponding algebraic approach has to be built from scratch. On the other hand, we note that the “algebraic” point of view remains sufficiently close to the “combinatorial” one so that the interplay is fairly smooth. This is one of the facts illustrated by this paper.

2 Presented results

2.1 Preliminaries

We shall assume that all words are over some fixed alphabet Σ. The concatenation of u and v is denoted as $u \cdot v$, or simply as uv. The length of a word w is $|w|$. Let $w[i]$, $0 < i < |w|$, be the $(i+1)$th letter of w. That is,

$$w = w[0] \cdot w[1] \cdots w[n-1],$$

where $n = |w|$.

A word w has a period $1 \leq p$ if $w[i] = w[i+p]$ for each $0 \leq i < |w| - p$. We allow (trivial) periods $p \geq |w|$. It is useful to note that w has a period p if and only if w is a prefix of $u \cdot w$, where u is a word of length p, called a *period root* of w. Which, in turn, is equivalent to w being a prefix of u^ω with $u^\omega = uuu\ldots$.

The Periodicity Lemma (PL) claims that if a word w of length at least $p + q - \gcd(p, q)$ has periods p and q, then it also has a period $\gcd(p, q)$. It is an exercise to see that the PL holds if the length of w is at least $p + q$, which is often sufficient in applications.

The very first result in the basic course of Combinatorics od Words is the Commutation Lemma which says, that $xy = yx$ implies the existence of a word t such that $x \in t^*$ and $y \in t^*$. Here t^* denotes the set $\{t^n \mid 0 \leq n\}$, as is common in regular expressions. The Commutation Lemma is easy to prove directly, but it can be also noted that the word $w = uv = vu$ has periods $|u|$ and $|v|$, and the claim follows from the PL.

The Kleene star used in the expression t^* is commonly used even for sets as, for example, in $\{u, v\}^*$. However, this allows a certain confusion. If G is a set of words over Σ, then G^* should denote all words over Σ generated by G. On the other hand, Σ^* denotes all words over the alphabet Σ, and the difference between the alphabet Σ and the set of words G has to be kept in mind. Strictly speaking, Σ^* is not generated by the alphabet Σ, but rather by the set of singletons, that is, words of length one. While the subtle difference between letters and singletons is typically ignored in the literature without any significant harm, for the formalization, the difference between a letter a, and the list $[a]$ must obviously be kept in mind. We therefore prefer to denote $\langle G \rangle$ the submonoid of Σ^* generated by a set $G \subset \Sigma^*$. We also call it the *hull* of G. The expression t^* above is therefore an abbreviation for $\langle \{t\} \rangle$.

2.2 The theorem of Lyndon and Schützenberger

We present a concise proof of the theorem of Lyndon and Schützenberger. Our proof is similar to the one given in [11, Section 9.2], however, the core case $c = 3$ is significantly simplified. The proof is rather dense, and relies on intuition at several places. A proof of this kind is standard in the literature, and should be easily comprehensible for a reader with some experience in Combinatorics on Words. At the same time, the proof should document that a verified formalization is desirable already on this level.

Theorem 2.1. *If $x^a y^b = z^c$ and $a, b, c \geq 2$, then the words x, y a z commute.*

Proof. By symmetry, assume $|x^a| \geq |y^b|$.

The word x^a has periods $|x|$ a $|z|$. If $|x^a| \geq |z| + |x|$, then the Periodicity lemma implies that x and z have a period dividing $|x|$ a $|z|$, which easily yields that they commute. Similarly if $|y^b| \geq |z| + |y|$.

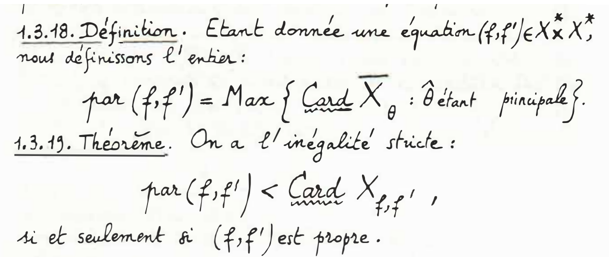
Therefore, suppose that x^{n-1} is a proper prefix of z and y^{m-1} a proper suffix of z. Then $|x^a| < 2|z|$ and $|y^b| < 2|z|$, hence $c < 4$.

Let $c = 3$. If $a \geq 3$, then $|x^2| < |z|$ implies $|x^3| < \frac{3}{2}|z|$, contradicting the assumption $|x^a| \geq |y^b|$. Therefore $a = 2$ and $|x| \geq |y|$. There are words u, v, w such that $x = uw = wv$, $z = xu = wvu$ and $y^b = vuwvu$. From $uw = wv$ we deduce that uwv has a period $|u|$. Moreover, uwv is a factor of y^b which implies that it has a period $|y|$. Since $|y| + |u| \leq |uwv|$, the PL implies that $d = \gcd(|u|, |y|)$ is a period of uwv. It is easy to see that d divides also $|v|$ and $|w|$, which implies that words u, v and w commute. Therefore also x, y and z commute.

The case $c = 2$ remains. We have $z = x^{a-1}u = wy^b$, where $uw = x$. Then $wz = (wu)^a = w^2y^b$, where wu is shorter than z. The proof is completed by induction. □

2.3 Graph lemma

The DT for word equations states that any solution of a nontrivial equation has rank less than the number of unknowns. It was probably for the first time proved in the legendary hand-written book by Lentin [10]:



1.3.18. Définition. Etant donnée une équation $(f, f') \in X^* \times X^*$, nous définissons l'entier :

$$\text{par}\,(f, f') = \text{Max}\left\{ \text{Card}\,\overline{X}_{\theta} : \hat{\theta} \text{ étant principale} \right\}.$$

1.3.19. Théorème. On a l'inégalité stricte :

$$\text{par}\,(f, f') < \text{Card}\, X_{f,f'} \,,$$

si et seulement si (f, f') est propre.

which is, admittedly, not a fully formalized format by today standards. The GL is a stronger version of the claim, generalized for systems of equations. It owes its name to the formulation in [7], where the rank is described using connected components of a graph related to a system of equations. The connected components are in fact equivalence classes of unknowns which must share the first element in the decomposition into the free basis as explained below. The crucial fact yielding the GL has a nice proof given already in [3], which is the one we formalize.

Every submonoid of Σ^* has a unique smallest generating set, called *basis*. It is simply the set of indecomposable nonempty elements, that is, elements that cannot be nontrivially factorized. The basis exists since any factorization decreases length. For example, the set $\{a, ab, ba\}$ is the basis of the monoid $\langle\{a, ab, ba\}\rangle$. However, the latter monoid is not free, since aba has two distinct factorizations into elements of the basis $a \cdot ba = ab \cdot a$.

As already mentioned in the Introduction, for any set $X \subseteq \Sigma^*$, there is a unique smallest (with respect to the inclusion) free monoid $\langle X \rangle_{\mathrm{F}}$ containing X as a subset. This follows from the fact that a monoid M is free if it is equidivisible, that is, if $x \cdot y = u \cdot v$, where $x, y, u, v \in M$ and u is shorter than x, implies that $u^{-1}v$ is also in M. This is easily seen to be equivalent to the usual "unique factorization" or "no nontrivial relation" definition of freeness. Another formulation of the same fact is the *stability condition*:

$$p, pw, wq, q \in M \Longrightarrow w \in M\,.$$

Note that the link to the equidivisibility is given by $p \cdot wq = pw \cdot q$. Since the stability condition is obviously closed under intersection, we obtain

$$\langle X \rangle_{\mathrm{F}} = \bigcap \{M \mid X \subset M,\ M \text{ free}\}.$$

The basis $\mathcal{B}_{\mathrm{F}}(X)$ of $\langle X \rangle_{\mathrm{F}}$ is the *free basis* of X, and its cardinality is the *free rank* of X. The DT states that the free rank of X is at most the cardinality of X, and it is strictly smaller unless X is its own free basis, that is, unless it is a *code*. Note that the free basis has a smaller cardinality than the (ordinary) basis, although the monoid it generates is larger.

Very little can be said in general about the actual degree of the defect, that is, about the actual value of the free rank for a set that is not a code, and the GL is the best general bound available. Note that any element of $\langle X \rangle_{\mathrm{F}}$ has a unique factorization into elements of $\mathcal{B}_{\mathrm{F}}(X)$. For $x \in \mathcal{B}_{\mathrm{F}}(X)$, let $\mathrm{hd}_{\mathrm{F}}(x)$ denote the *head*, that is, the first factor, of such a decomposition. The crucial fact mentioned above yielding the GL is the following one (see [3]):

Theorem 2.2. $\mathcal{B}_{\mathrm{F}}(X) = \{\mathrm{hd}_{\mathrm{F}}(x) \mid x \in X\}$.

The nontrivial inclusion is to show that each element of $\mathcal{B}_{\mathrm{F}}(X)$ must be a head of some element from X. The proof is based on the following simple observation:

Lemma 2.3. *Let C be a code, and let $b \in C$. Then*

$$C' = \{zb^k \mid k \geq 0,\ z \in C,\ z \neq b\}$$

is also a code.

Now, if $b \in \mathcal{B}_{\mathrm{F}}(X)$ is not a head, then X is contained in $\langle C' \rangle$ where C' is as in the lemma for $C = \mathcal{B}_{\mathrm{F}}(X)$. Since $\langle C' \rangle$ does not contain b, we have $\langle C' \rangle \subsetneq \langle X \rangle_{\mathrm{F}}$, a contradiction with the minimality of $\langle X \rangle_{\mathrm{F}}$.

3 Remarks on the formalization in Isabelle/HOL

Formalization described in this paper consists of four theories. Two background theories

- **CoWBasic**: defines basic concepts, and contains about three hundred auxiliary lemmas (not all of them needed for the two main presented results).
- **CoWSubmonoids**: defines submonoids, and contains fundamental properties of bases, codes and free hulls.

and two main results:

- **CoWLyndonSchutzenberger**: of Theorem 2.1 we prove only that x and y commute. Commutation of z follows easily, the only reason for this choice is that there is no elegant formulation of the claim that three words commute.

- **CoWGraphLemma**: proves Lemma 2.3 and Theorem 2.2.

We highlight some details from these theories.

3.1 Lists

The choice of the datatype of lists to represent words is an obvious one. The underlying alphabet is an unspecified datatype represented by the type variable ′a. We use the abbreviation · for the Isabelle's concatenation symbol , and ε for the empty list (Nil or [] in Isabelle). We also introduce notation **|**w**|** for length w, and ∈n for an nonempty element. Moreover, the set $G \setminus \{\varepsilon\}$ can be written as G_+.
NB: Whenever we speak about Isabelle, we have in mind the Main library of Isabelle/HOL.

3.2 Monoids and powers

The choice of the symbol · for concatenation underscores the importance of the fact that lists form a monoid. This is a trivial fact (associativity is the most natural, almost invisible property of concatenation) which would deserve no discussion, if not for the need of using the power. Since Isabelle does not instantiate the class power to lists, we do not have a direct approach to such basic facts as $x^{a+b} = x^a \cdot x^b$. There are several options how to solve this: we could instantiate the class power ourselves, or we could interpret lists as a sublocale of monoid_mult. None of these solutions being optimal, we define list_power

```
primrec list-power :: ′a list ⇒ nat ⇒ ′a list (infixr @)
where
  power-zero: u@0 = ε |
  power-Suc-list: u@(Suc n) = u · u@n
```

and prove corresponding lemmas afresh. The overhead is minimal. Since we then cannot use the usual $u^\wedge n$, we write $u^{@}n$, as a kind of tribute to the original notation for concatenation.

3.3 Elementary equations on words

As mentioned above, the solution of the most elementary non-trivial equation on words $x \cdot y = y \cdot x$ is provided by Main's theory List as comm-append-are-replicate:

```
lemma comm-append-are-replicate:
  [[ xs ≠ []; ys ≠ []; xs @ ys = ys @ xs ]]
  ⟹ ∃m n zs. concat (replicate m zs) = xs ∧ concat
(replicate n zs) = ys
```

We can see here the original Isabelle's notation, and also the way how it deals with the power: zs^m is obtained as concat (replicate m zs). More significantly, the claim is unnecessarily weak, since the conclusion holds even for empty lists. We can therefore straightforwardly generalize to:

```
theorem comm:  x · y = y · x  ⟹  ∃ t m k. x = t@k ∧
y = t@m
```

or even to

```
corollary comm-root:  x · y = y · x  ⟷  (∃ t. x ∈ t*
∧ y ∈ t*)
```

A slightly more elaborate equation $x \cdot z = z \cdot y$, which is in fact the relation of x and y being conjugated by the word z, is, expectedly, not treated in the Isabelle's Main library. The solution of this equation is as follows:

```
theorem conjug: assumes x·z = z·y and x ≠ ε
  shows ∃ u v k. x = u · v ∧ y = v · u ∧ z = (u · v)@k · u
```

It is interesting to remark that, in light of our formalization, it is more natural to see the equality $x \cdot z = z \cdot y$ not as a property of x and y (namely of their being conjugated) but rather as a property of z, namely of its having a period root x, written as $z \leq_{\mathrm{p}} x^\omega$, where $\leq_{\mathrm{p}}$ is the prefix relation:

```
definition period-root :: ′a list ⇒ ′a list ⇒ bool (- ≤p -ω )
  where period-root z x = (z ≤p x · z ∧ x ≠ ε)
```

3.4 The theorem of Lyndon and Schützenberger

We have seen above that the solution of the LSE naturally splits into several cases. Two of them are proven separately in a locale:

```
locale LS =
  fixes x a y b z c
  assumes a:  2 ≤ a and b: 2 ≤ b and c: 2 ≤ c   and eq:
x@a · y@b = z@c
```

Namely, the cased solved by the PL,

```
lemma per-lemma-case:
  assumes |z| + |x| ≤ |x@a| and x ≠ ε
  shows x·y=y·x
```

and the core case $c = 3$.

```
lemma core-case:
  assumes
    c = 3 and
    b∗|y| ≤ a∗|x| and x ≠ ε and y ≠ ε and
    lenx: a∗|x| < |z| + |x| and
    leny: b∗|y| < |z| + |y|
  shows x·y = y·x
```

It would seem natural to solve even the remaining case $c = 2$ separately, and then simply put the three cases together. However, this is not possible, since the induction, abruptly announced on the last line of the human proof, actually governs the whole proof since it covers the first two cases as well. (This is one of the typical backtracking moments of the development.) The main proof of the Theorem of Lyndon and Schützenberger, Theorem 2.1, therefore has the following structure.

```
theorem Lyndon-Schutzenberger:
  [[ x@a·y@b = z@c;  2 ≤ a;  2 ≤ b; 2 ≤ c ]]
  ⟹ x·y = y·x
proof (induction |z| + b∗|y| arbitrary: x y z a b c  rule:
nat-less-induct)
qed
```

Note that the induction is on $|z| + b|y|$. This curious choice avoids (in a hopefully elegant way), another typical pitfall of the formalization, namely the humanly generous "by symmetry" from the first line of the proof (which, by the way, is still more precise than frequent and even more generous "wlog"). The point is that the introduced locale LS allows two interpretations (within the proof), one for each of the two symmetric situations (the first interpretation needs no name, the second one is called LSrev):

```
interpret  LS x a y b z c
interpret  LSrev: LS rev y b rev x a rev z c
```

Now, if $|x^a| < |y^b|$, then the symmetric case is solved immediately by induction.

3.5 Submonoids, free hull and decompositions

The set $\langle G \rangle$ can be seen (and defined) in two different ways:

- it is the smallest set closed under concatenation containing G; and/or
- it is the set of all words that can be obtained by concatenation of lists of words from G (note that we deal with lists of lists here).

We use the first definition:

```
inductive-set hull :: 'a list set ⇒ 'a list set (⟨-⟩)
  for G where
    ε ∈ ⟨G⟩
  | gen-in: w ∈ G ⟹ w ∈ ⟨G⟩
  | w_1 ∈ ⟨G⟩ ⟹ w_2 ∈ ⟨G⟩ ⟹ w_1 · w_2 ∈ ⟨G⟩
```

and prove its equivalence to the latter:

```
lemma hull-concat-lists: ⟨G⟩ = concat ` lists G
```

The term $\mathrm{Dec}\ G\ u$ represents SOME decomposition of the word u into elements of G. It returns a list of words, i.e., of type 'a list list.

```
fun decompose :: 'a list set  ⇒ 'a list ⇒ 'a list list (Dec
- - ) where
  decompose G u = (SOME us. us ∈ lists G_+ ∧ u = concat
  us)
```

The output of the function makes no good sense if the second argument is not in $\langle G \rangle$. Nevertheless, even for elements of $\langle G \rangle$ the list is an unspecified choice among all possible factorizations. For example, if $G = \{a, ab, ba\}$ and $u = aba$, then $\mathrm{Dec}\ G\ u$ is either $[a, ba]$ or $[ba, a]$. This in particular implies that we cannot prove $\mathrm{Dec}\ G\ (u{\cdot}v) = \mathrm{Dec}\ G\ u{\cdot}\mathrm{Dec}\ G\ v$.

These difficulties disappear in the free hull, where the decomposition is unique. In particular, the term $\mathrm{Dec}\ \mathcal{B}_{\mathrm{F}}\ (X)\ \ x$, which plays the crucial role in the GL, has a definite meaning.

The definition of the free hull is a natural extension of the inductive definition of the (ordinary) hull by the stability condition:

```
inductive-set free-hull :: 'a list set ⇒ 'a list set (⟨-⟩_F)
  for G where
    ε ∈ ⟨G⟩_F
  | free-gen-in: w ∈ G ⟹ w ∈ ⟨G⟩_F
  | w_1 ∈ ⟨G⟩_F ⟹ w_2 ∈ ⟨G⟩_F ⟹ w_1 · w_2 ∈ ⟨G⟩_F
  | p ∈ ⟨G⟩_F ⟹ q ∈ ⟨G⟩_F ⟹ p · w ∈ ⟨G⟩_F ⟹ w · q ∈
⟨G⟩_F ⟹ w ∈ ⟨G⟩_F
```

3.6 The Graph Lemma

The theory behind the proof of the GL relies on two inductive sets. The first one is the set C' of Lemma 2.3:

```
inductive-set no-head-gen :: 'a list set ⇒ 'a list ⇒ 'a
list set
  for C b where
    u ∈ C ⟹ u ≠ b ⟹ u ∈ no-head-gen C b
  | u ∈ no-head-gen C b ⟹  u · b ∈ no-head-gen C b
```

The second one is the set of all elements in $\langle C \rangle$ whose factorization into elements of C does not start with b.

```
inductive-set no-head :: 'a list set ⇒ 'a list ⇒ 'a list
set
  for C b where
    ε ∈ no-head C b
  | u ∈ C ⟹ u ≠ b ⟹ u ∈ no-head C b
  | u ∈n no-head C b ⟹ v ∈ ⟨C⟩ ⟹ u · v ∈ no-head C b
```

The core of the proof is to show that no-head-gen C b generates no-head C b, and, most importantly, that no-head-gen C b is a code:

```
theorem no-head-gen-code:
  assumes code C and b ∈ C
  shows code {z · b@k | z k. z ∈ C ∧ z ≠ b}
```

With those ingredients, the proof of Theorem 2.2:

```
theorem graph-lemma: 𝔅_F X = {hd (Dec (𝔅_F X) x) |
x. x ∈n X}
```

is not difficult.

4 Conclusion

The aim of this paper to introduce an ongoing formalization of Combinatorics on Words. The next step after the Lyndon-Schützenberger theorem is its natural extension obtained independently by J.-P. Spehner [18], and by E. Barbin-Le Rest, M. Le Rest [1] which claims that $x^i y$ is the only non-trivial way (up to symmetry and conjugation) how two non-commuting words can form an imprimitive word (like z^c). The history of this result is another good motivation for our formalization project. The result, while very natural and important, has been almost forgotten (it was cited only six times before 2015). A weaker form of this result was even rediscovered in 1994 [17], and started to be referenced. One reason for this is that already this relatively simple result is very technical and difficult to read. Moreover, the paper contains several minor inaccuracies which may further discourage the reader. This is by no means an exceptional situation in Combinatorics on words, which testifies for a strong need of formally verified proofs in the field.

Acknowledgments

References

[1] Evelyne Barbin-Le Rest and Michel Le Rest. 1985. Sur la Combinatoire des Codes à Deux Mots. *Theor. Comput. Sci.* 41 (1985), 61–80.

[2] Jean Berstel and Dominique Perrin. 2007. The origins of combinatorics on words. *European Journal of Combinatorics* 28, 3 (2007), 996 – 1022. https://doi.org/10.1016/j.ejc.2005.07.019

[3] J Berstel, D Perrin, J.F Perrot, and A Restivo. 1979. Sur le théorème du défaut. *Journal of Algebra* 60, 1 (1979), 169 – 180. https://doi.org/10.1016/0021-8693(79)90113-3

[4] Georges Gonthier et al. 2013. A Machine-Checked Proof of the Odd Order Theorem. In *ITP (Lecture Notes in Computer Science, Vol. 7998)*. Springer, 163–179.

[5] N. J. Fine and H. S. Wilf. 1965. Uniqueness theorems for periodic functions. *Proc. Amer. Math. Soc.* 16, 1 (jan 1965), 109–109. https://doi.org/10.1090/S0002-9939-1965-0174934-9

[6] Thomas Hales et al. 2017. A FORMAL PROOF OF THE KEPLER CONJECTURE. *Forum of Mathematics, Pi* 5 (2017), e2. https://doi.org/10.1017/fmp.2017.1

[7] T. Harju and J. Karhumäki. 1986. On the defect theorem and simplifiability. *Semigroup Forum* 33 (1986), 199–217.

[8] Štěpán Holub. 2017. Commutation and Beyond. In *Combinatorics on Words*, Srečko Brlek, Francesco Dolce, Christophe Reutenauer, and Élise Vandomme (Eds.). Springer International Publishing, Cham, 1–5.

[9] Artur Jez. 2016. Recompression: A Simple and Powerful Technique for Word Equations. *J. ACM* 63, 1 (2016), 4:1–4:51. https://doi.org/10.1145/2743014

[10] A. Lentin. 1972. *Equations dans les monoides libres*. De Gruyter Mouton. https://doi.org/10.1515/9783111544526

[11] M. Lothaire. 1997. *Combinatorics on words*. Cambridge University Press, Cambridge. xviii+238 pages. https://doi.org/10.1017/CBO9780511566097

[12] M. Lothaire. 2002. *Makanin's Algorithm*. Cambridge University Press, 387–442. https://doi.org/10.1017/CBO9781107326019.013

[13] R. C. Lyndon and M. P. Schützenberger. 1962. The equation $a^M = b^N c^P$ in a free group. *Michigan Math. J.* 9, 4 (12 1962), 289–298. https://doi.org/10.1307/mmj/1028998766

[14] Gennadiy Semenovich Makanin. 1977. The problem of solvability of equations in a free semigroup. *Matematicheskii Sbornik* 145, 2 (1977), 147–236.

[15] Dirk Nowotka and Aleksi Saarela. 2018. One-Variable Word Equations and Three-Variable Constant-Free Word Equations. *Int. J. Found. Comput. Sci.* 29, 5 (2018), 935–950. https://doi.org/10.1142/S0129054118420121

[16] Neil Robertson, Daniel Sanders, Paul Seymour, and Robin Thomas. 1997. The Four-Colour Theorem. *Journal of Combinatorial Theory, Series B* 70, 1 (1997), 2 – 44. https://doi.org/10.1006/jctb.1997.1750

[17] H.J. Shyr and S.S. Yu. 1994. Non-primitive words in the language p^+q^+. *Soochow Journal of Mathematics* 20 (01 1994).

[18] J.-P. Spehner. 1976. *Quelques problèmes d'extension, de conjugaison et de presentation des sous-monoïdes d'un monoïde libre*. Ph.D. Dissertation. Université Paris VII, Paris.

[19] Axel Thue. 1906. Über unendliche Zeichenreichen. *Skrifter: Matematisk-Naturvidenskapelig Klasse* (1906).

[20] Axel Thue. 1912. Uber die gegenseitige Lage gleicher Teile gewisser Zeichenreihen. *Kra. Vidensk. Selsk. Skrifer, I. Mat. Nat. Kl.* (1912), 1–67.