The Inclusion Problem of Context-free Languages: Some Tractable Cases*

Alberto Bertoni[1], Christian Choffrut[2], and Roberto Radicioni[3]

[1] Università degli Studi di Milano
Dipartimento di Scienze dell'Informazione
Via Comelico 39, 20135 Milano, Italy
bertoni@dsi.unimi.it
[2] L.I.A.F.A. (Laboratoire d'Informatique Algorithmique,
Fondements et Applications),
Université Paris VII, 2 Place Jussieu, 75221 Paris - France
cc@liafa.jussieu.fr
[3] Università degli Studi dell'Insubria
Dipartimento di Informatica e Comunicazione
Via Mazzini 5, 21100 Varese, Italy
radicion@dsi.unimi.it

Abstract. We study the problem of testing whether a context-free language is included in a fixed set L_0, where L_0 is the language of words reducing to the empty word in the monoid defined by a complete string rewrite system. We prove that, if the monoid is cancellative, then our inclusion problem is polynomially reducible to the problem of testing equivalence of straight-line programs in the same monoid. As an application, we obtain a polynomial time algorithm for testing if a context-free language is included in a Dyck language (the best previous algorithm for this problem was doubly exponential).

Introduction

In this paper we analyze problems of this kind: given a fixed language L_0, to decide whether the language L_G generated by a context-free grammar is contained in L_0. It is known that he class of context-free grammars generating languages L_0 for which the problem is decidable is not recursive ([7]). This problem is undecidable even for some deterministic context-free language L_0, while it turns out to be decidable if L_0 is superdeterministic ([6]); in this case a doubly exponential time algorithm has been shown.

Since the class of superdeterministic languages includes the AFDL (Abstract Family of Deterministic Languages) closure of Dyck languages, it is decidable if a context-free grammar generates a language which is included in a Dyck language, i.e., the language of well-parenthesized expressions over a fixed but arbitrary subset of parentheses. The case with a single type of parenthesis has

* This work is partially supported by the MIUR PRIN Project "Automata and Formal languages: mathematical and application driven studies".

been studied by Knuth in [8], while a doubly exponential algorithm for the general case can be deduced from an algorithm proposed in [1].

In this work, we exhibit a class of languages L_0 for which the inclusion problem can be solved efficiently. In particular, given a fixed complete string rewrite system, we study the problem of verifying whether a context-free language is included in the set L_0 of words which reduce to the empty word. We prove that, if the string rewrite system is complete and defines a cancellative monoid, then the problem polynomially reduces to testing the compressed equivalence problem on the monoid, i.e., whether two straight-line programs evaluate the same element of the monoid. Straight-line programs are a flexible compressed representation of strings; the compressed equivalence problem has been deeply studied, in particular in the case of 2-homogeneous and 2-monadic string rewrite systems ([11]). In the case of 2-homogeneous string rewrite systems, equivalence is solvable in polynomial time if and only if the rewrite system is N-free, unless P=NP ([11]).

By applying the previous results, we obtain a polynomial algorithm for testing inclusion in the case L_0 belongs to a class of languages that extends that of "Dyck-like" languages. This result also extends to the noncancellative monoids defined by N-free 2-homogeneous string rewrite systems; on the other hand, for 2-homogeneous non N-free rewrite systems, the inclusion problem is at least coNP-hard.

Furthermore, we study the more general case of complete unitary string rewrite systems ([3]), that is, systems for which all rules are of the form $u \to 1$, and prove that the inclusion of a regular language in a regular set over the generated monoid can be tested again in polynomial time.

1 Definitions

Given an alphabet Σ, the free monoid it generates is denoted by Σ^* and the unit, or *empty word* is denoted by 1. A *context-free grammar* is a quadruple $G = (\Sigma, S, P, N)$ where N is the finite alphabet of *nonterminals* which is disjoint from the alphabet Σ of terminal letters, $S \in N$ is the *axiom* and $P \subseteq N \times (N \cup \Sigma)^*$ is the finite set of *productions* or *rules*. A rule is usually written in the form $X \to \alpha$. The smallest relation over $(N \cup \Sigma)^*$ which is invariant under right and left concatenation is denoted by $\Rightarrow$ and its reflexive and transitive closure by $\stackrel{*}{\Rightarrow}$. The *language* generated by the grammar G is the set L_G of all words $w \in \Sigma^*$ such that $S \stackrel{*}{\Rightarrow} w$ holds.

A grammar is in *Chomsky's normal form* if its rules are of the form $X \to YZ$ or of the form $X \to a$ where X, Y, Z are nonterminals and a is a terminal. Such a grammar generates a context-free language which does not contain the empty word. Since a language $L \subseteq \Sigma^*$ is context-free if and only if $L \setminus \{1\}$ so is, there is no loss of generality to restrict the study to context-free languages not containing the empty word (also called ϵ-free in the literature). In other words, the expression "Given a context-free language" means in our context that the language is given by some grammar in Chomsky's normal form. It is convenient to enumerate the nonterminals $X_0, \ldots, X_n$ and to assume that X_0 is the axiom. Furthermore, as usual, we assume that all nonterminals appear in some derivation of a word of the language, i.e., for all $X_i \in N$ there exist a word $w \in L_G$ and a derivation of the form $X_0 \overset{*}{\Rightarrow} \alpha X_i \beta \overset{*}{\Rightarrow} w$.

A *straight-line program* (SLP) is a restricted context-free grammar $G = (\Sigma, S, P, N)$ such that:

- for every $X \in N$, there exists exactly one production of the form $(X, \alpha) \in P$ for $\alpha \in (N \cup \Sigma)^*$, and
- there exists a linear order $\prec$ on the set of nonterminals N such that $X \prec Y$ whenever there exists a production of the form $(X, \alpha) \in P$ for $\alpha \in (N \cup \Sigma)^* Y (N \cup \Sigma)^*$.

Every SLP S obviously generates a single word, denoted by $\text{eval}(S)$. In our work, we will always use SLPs in Chomsky's normal form. The size $g(S)$ of a SLP S is defined as the number of nonterminal symbols appearing in it. Since $g(S)$ could be logarithmic with respect to the length of $\text{eval}(S)$, SLPs are considered as a compression encoding comparable to Lempel-Ziv encoding ([14]). For several well known string problems solvable in polynomial time, the compressed version has been studied, that is, the complexity of the same problem was analyzed when the input is given in a compressed form ([5, 9, 12–14]).

Let Σ be a finite alphabet. A *semi-Thue system* (or *string rewrite system*) is a pair of *generators* and *relators* $\langle \Sigma; R \rangle$ where R is a finite subset of $\Sigma^* \times \Sigma^*$. The pair $\langle \Sigma; R \rangle$ is a *monoid presentation*; it defines the monoid $M(R)$ which is the quotient of Σ^* by the congruence $\sim_R$ generated by the relators R. The *canonical* morphism ϕ_R maps Σ^* onto $M(R)$ by assigning to every word its congruence class. We use the same notation when we apply it to subsets and, more generally, to n-tuples of subsets.

A pair $(u, v) \in R$ is also denoted by $u = v$. The rewrite theory developed by Knuth in the seventies consists, whenever possible, to define a ordering of the free monoid which is invariant under right and left concatenation and to orient each relator (u, v) as $u \to v$ if u is greater than v. Once oriented, such a pair can be considered as a *rewrite rule*, also called *reduction rule* with the purpose that each occurrence of u in a word can be replaced by an occurrence of v. A word containing no occurrence of a left handside is called *reduced*, otherwise it is *reducible*. We still denote by $\to$ the closure of the set of reduction rules relative to right and left concatenation and by $\to_R^*$ its transitive closure. If the ordering thus defined is *complete*, i.e., it has the property of finite termination and of confluence, [4], each word x is equivalent to a unique reduced word denoted by $\text{Red}_R(x)$, also simply written $\text{Red}(x)$ when R is understood. Furthermore this reduced word depends on the equivalence class only, and we say that it is its *normal form*. In this case, two words $x, y \in \Sigma^*$ are congruent, i.e., $x \sim_R y$, if and only if $\text{Red}(x) = \text{Red}(y)$.

Given a fixed complete string rewrite system, we are interested in the following decision problem

Inclusion Problem for $\langle \Sigma; R \rangle$

Instance: a grammar G in Chomsky normal form.

Question: does $L_G \subseteq \phi_R^{-1}(1)$ hold?

In this work, we show that, if the monoid $M(R)$ is cancellative, this problem is polynomially reducible to the compressed version of an equality problem:

Compressed-Equality Problem for $\langle \Sigma; R \rangle$

Instance: two SLPs S_1 and S_2 with $\mathrm{eval}(S_1), \mathrm{eval}(S_2) \in \Sigma^*$.

Question: does $\phi_R(\mathrm{eval}(S_1)) = \phi_R(\mathrm{eval}(S_2))$ hold?

2 Rewrite Systems Defining Cancellative Monoids

A monoid M is said to be *cancellative* if $xy = xz$ implies $y = z$ and $yx = zx$ implies $y = z$, for every $x, y, z \in M$. In this section, we prove that, if $\langle \Sigma; R \rangle$ defines a cancellative monoid $M(R)$, then Inclusion Problem for $\langle \Sigma; R \rangle$ is polynomially reducible to Compressed-Equality Problem for $\langle \Sigma; R \rangle$.

Now, let $m = |\Sigma|$ and consider a system of equations whose solutions are n-vectors of elements in the semiring $\langle 2^{\Sigma^*}, \cup, \cdot \rangle$ or in the semiring $\langle 2^M, \cup, \cdot \rangle$ accordingly

$$X_i = \bigcup_{0 \leq j,k \leq n-1} \alpha_{j,k}^{(i)} X_j X_k \cup \bigcup_{1 \leq \ell \leq m} \beta_\ell^{(i)} a_\ell \tag{1}$$

where a_ℓ are constant singletons and $\alpha_{j,k}^{(i)}$ and $\beta_\ell^{(i)}$ have values $\emptyset$ or 1.

When the X_i's are interpreted as subsets in Σ^*, the component X_0 of the least fixed point of (1) is the context-free language generated by the grammar whose productions are $X_i \to X_j X_\ell$ with $\alpha_{jk}^{(i)} \neq \emptyset$ and $X_i \to a_\ell$ with $\beta_\ell^{(i)} \neq \emptyset$. Therefore, we will not distinguish between systems of equations and systems of productions.

Lemma 1. *Consider a grammar G in Chomsky normal form as in (1) and a cancellative rewrite system $\langle \Sigma, R \rangle$. Assume that $\phi_R(L_G) = \{1\}$. Then, the images $\phi_R(X_i)$ of all components are singletons and satisfy Equation (1) interpreted in the quotient monoid $M(R)$. Conversely, if there exist elements $y_i \in \phi_R(X_i)$ such that the n-tuple $(y_0, y_1, \ldots, y_{n-1})$ with $y_0 = 1$ satisfies the system (1) interpreted in $M(R)$, then $\phi_R(L_G) = \{1\}$.*

Proof. Consider the graph whose vertices are the nonterminal symbols of G and there is an edge (X, Y) where $X \to YZ$ or $X \to ZY$. We perform a breadth-first visit starting from the axiom and verify by induction that, when visiting (X, Y), we have that $\phi_R(Y)$ is a singleton.

Indeed, this is true if X is the axiom, by definition. Now, consider the rule $X_i \to X_j X_k$ and assume without loss of generality that the visited edge is (X_i, X_j). Since $\phi_R(X_i) \supseteq \phi_R(X_j)\phi_R(X_k)$ holds and since $M(R)$ is cancellative, the induction hypothesis that $\phi_R(X_i)$ is a singleton implies that so is $\phi_R(X_j)$.

Let us prove the converse. First observe that the image in the morphism ϕ_R of the least (in the subset ordering) solution of the system (1) in the power set of Σ^* is the least solution of the system when interpreted in the power set of $M(R)$. Indeed, let $\tau : \left(2^{\Sigma^*}\right)^n \to \left(2^{\Sigma^*}\right)^n$ and $\sigma : \left(2^{M}\right)^n \to \left(2^{M}\right)^n$ be the transformations defined by the system (1) when interpreted in $\left(2^{\Sigma^*}\right)^n$ and $\left(2^{M}\right)^n$ respectively and observe that these transformations are monotone and continuous. The least solution of the system in $\left(2^{\Sigma^*}\right)^n$ is the subset $\bigcup_{k\to\infty} \tau^k(\emptyset)$. Now we have

$$\phi_R\left(\bigcup_{k\to\infty} \tau^k(\emptyset)\right) = \bigcup_{k\to\infty} \phi_R(\tau^k(\emptyset))$$

Because of equality $\phi_R\tau = \sigma\phi_R$ between the two composed mappings and of the equality $\phi_R(\emptyset) = \emptyset$ we finally obtain

$$\phi_R\left(\bigcup_{k\to\infty} \tau^k(\emptyset)\right) = \bigcup_{k\to\infty} \sigma^k(\phi_R(\emptyset)) = \bigcup_{k\to\infty} \sigma^k(\emptyset)$$

as claimed.

The previous Lemma has the following interesting consequence.

Theorem 1. *In the case of a cancellative complete rewrite system $\langle\Sigma, R\rangle$,* Inclusion Problem for $\langle\Sigma, R\rangle$ *is polynomially reducible to* Compressed-Equality problem for $\langle\Sigma, R\rangle$.

Proof. (Outline) In view of the previous Lemma, the condition $\phi_R(L_G) = \{1\}$ can be tested in two steps.

First step: We assign to every nonterminal X_i of G a word w_i which it generates and chosen as explained below.

Second step: We check whether or not the n-tuple $(w_0, \ldots, w_{n-1})$ satisfies the following conditions:
- $w_0 \sim_R 1$;
- for all productions $X_i \to X_jX_k$ (resp. $X_i \to a_\ell$), $w_i \sim_R w_jw_k$ (resp. $w_i \sim_R a_\ell$)). These conditions are not verified on the words, but through SLP's representing the words.

We now explain how we choose the words w_i. Denote by $h(X_i)$ the minimal height of a derivation tree with root labelled by X_i. If $h(X_i) = 1$ then take arbitrarily a word w_i where $X_i \to w_i$ is an X_i-production of the grammar G. The SLP associated to w_i is then reduced to the production $X_i \to w_i$. More generally, if $h(X_i) > 1$, there exist two non-terminals X_j, X_k such that $h(X_i) > h(X_j), h(X_k)$ and $X_i \to X_jX_k$ is a production of G. Then recursively choose w_jw_k as the word w_i represented by the SLP obtained by considering the production $X_i \to X_jX_k$ along with the productions of the two SLP's representing w_j and w_k.

From a complexity viewpoint, considering the grammar size as the size of the input, the first step can be executed in polynomial time, while the second requires to call the SLPs equality test in $M(R)$ a polynomial number of times. This proves the result.

This result can be applied to obtain an upper bound to the Inclusion Problem complexity for those cancellative monoids whose Compressed-Equality Problem complexity is known. This is the case of 2-homogeneous rewrite systems, introduced in [2], that will be discussed in the next section.

3 Unitary Confluent Systems

A presentation $\langle \Sigma; R\rangle$ is *unitary* if R is composed of elements of the form $(u, 1)$. We study the case where R can be oriented as a confluent and therefore complete system.

First of all, observe that for all unitary rewrite systems, the set of words which can be reduced to the empty word is generated by a context-free grammar with a unique nonterminal S whose productions are of the form $S \to 1$ and $S \to Sa_1Sa_2 \cdots Sa_pS$ where $(a_1a_2 \cdots a_p, 1) \in R$. Hence, Inclusion Problem for unitary rewrite systems is a special case of the general inclusion problem for context-free grammars (see for example [7, 6]).

Before studying more specifically the so-called 2-homogeneous case, we prove a result for the general case. Consider the following problem:

Regular Set Inclusion Problem

Input: a regular language $L \subseteq \Sigma^*$, a unitary rewrite system $\langle \Sigma, R\rangle$ and a regular subset $X \subseteq M(R)$.

Question: does $\phi_R(L) \subseteq X$ hold?

Theorem 2. Regular Set Inclusion Problem *is solvable in polynomial time.*

Proof. We briefly sketch how one can check the inclusion $\phi_R(L) \subseteq X$ efficiently. Since each equivalence class has a unique normal form, the previous inclusion is equivalent to saying that the set of normal forms of all elements in L is included in the set of normal forms which are representatives of an element of X. By definition of a regular set in the quotient $M(R)$, we may suppose that X is given by a finite automaton which recognizes for each element $x \in X$ and only for elements in X, a word representing this element. It thus suffices to show that the set of normal forms is an effective regular language. Indeed, if L is a regular language recognized by a finite automaton, augment it by adding, as long as possible, a transition from state q to state p labeled by the empty word whenever there exists a path from q to p labeled by a left handside of a rule. This procedure can be easily executed in polynomial time with respect to the size of the automaton and the size of the presentation of the monoid. The set of its normal forms is the intersection of the language recognized by this augmented automaton with the regular set of words containing no occurrence of a left handside of a rewrite rule. The inclusion problem consists thus of checking the inclusion $\mathrm{Red}_R(L) = \mathrm{Red}_R(\phi^{-1}(X))$.

3.1 2-Homogeneous Rewrite Systems

A unitary rewrite system is said to be *2-homogeneous* if every left-hand member of R is in Σ^2. In [2] it is shown that every 2-homogeneous rewrite system is equivalent to a 2-homogeneous confluent rewrite system $\langle \Theta, R \rangle$. In addition, Lohrey ([10]) proved that, for such a presentation, it is always possible to define a partition $\Theta = \Sigma \uplus \Delta \uplus \Gamma$ and an involution $^{-} : \Sigma \to \Sigma$ with

$$\{(a\bar{a}, 1) \mid a \in \Sigma\} \;\subseteq\; R \;\subseteq\; \{(a\bar{a}, 1) \mid a \in \Sigma\} \cup \{(ab, 1) \mid a \in \Delta, b \in \Gamma\}.$$

A rewrite system is called *N-free* if it is 2-homogeneous and $ac, ad, bc \in \mathrm{dom}(R)$ implies that $bd \in \mathrm{dom}(R)$, for every $a, b \in \Delta$ and $c, d \in \Gamma$. This means that the graph determined by nodes in $\Delta \cup \Gamma$ and edges $\{(a, b) \in \Delta \times \Gamma \mid ab \in \mathrm{dom}(R)\}$ is the disjoint union of complete bipartite graphs. In this case, we can rewrite Θ as

$$\Theta = \Sigma \cup \bigcup_{1 \le i \le k} \Delta_i \cup \bigcup_{1 \le i \le k} \Gamma_i \tag{2}$$

and R as

$$R \;=\; \{(a\bar{a}, 1) \mid a \in \Sigma\} \;\cup\; \{(ab, 1) \mid a \in \Delta_i, b \in \Gamma_i,\; 1 \le i \le k\}. \tag{3}$$

In this construction, for each i, $\Delta_i \cup \Gamma_i$ are the nodes of a complete bipartite graph. Moreover, if the sets Δ_i and Γ_i are all singletons, then the rewrite system is cancellative.

Particularly important cases are:

– The free group reduction

$$\left\langle a_1, \ldots, a_m, a_1^{-1}, \ldots, a_m^{-1}; \bigcup_{1 \le i \le m} (a_i a_i^{-1}, 1) \cup (a_i^{-1} a_i, 1) \right\rangle.$$

– The Dyck reduction ([1])

$$\left\langle a_1, \ldots, a_m, \bar{a}_1, \ldots, \bar{a}_m; \bigcup_{1 \le i \le m} (a_i \bar{a}_i, 1) \right\rangle;$$

in this case, $\phi_R^{-1}(1)$ is the Dyck set on $\{a_1, \ldots, a_m, \bar{a}_1, \ldots, \bar{a}_m\}$.

Lemma 2. *Let $\langle \Theta, R \rangle$ be a N-free rewrite system having $|\Delta_i| = |\Gamma_i| = 1$ for $1 \le i \le n$. Then,* Inclusion Problem for $\langle \Theta, R \rangle$ *is solvable in polynomial time.*

Proof. It is known that, in the case of a N-free rewrite system $\langle \Sigma, R \rangle$, Compressed Equality Problem for $\langle \Sigma, R \rangle$ is solvable in polynomial time ([11]). Hence, by Theorem 1, the result is straightforwardly proved.

We now show how to extend this result to the general case of N-free rewrite systems. Consider a presentation $\langle \Theta; R\rangle$ defined by the conditions (2) and (3). We introduce two symbols δ_i and γ_i for each Δ_i and Γ_i, respectively, and consider the alphabet

$$\Theta' = \Sigma \cup \bigcup_{1\le i\le k} \{\delta_i, \gamma_i\} \tag{4}$$

and the relation R' over Θ' defined by the same relators as those of Eq. (3) for Σ and the relators

$$\delta_i\gamma_i = 1, \quad \text{for } 1 \le i \le k.$$

We denote by $\sim_{R'}$ the congruence induced by R' on Θ'^* and by π the morphism from $\Theta^*/\sim_R$ to $\Theta'^*/\sim_{R'}$ leaving every element of Σ invariant and mapping all elements of Δ_i to δ_i and all elements of Γ_i to γ_i.

Theorem 3. Inclusion Problem for $\langle \Sigma, R\rangle$ *is solvable in polynomial time whenever* $\langle \Sigma, R\rangle$ *is* N*-free.*

Proof. Let the input of the problem be the context-free grammar G in Chomsky's normal form. Then, it suffices to prove that $\phi(L_G) = 1$ if and only if $\phi(\pi(L_G)) = 1$.

We prove by induction on the length of the words that $w \sim_R 1$ implies $\pi(w) \sim_{R'} 1$. As the system of rewriting rules is confluent, the relation $w \sim_R 1$ implies that we may write $w = w_1uw_2$ where $u = a\overline{a}$ or $u = \overline{a}a$ with $a \in \Sigma$ or $u = \delta\gamma$ with $\delta \in \Delta_i$ and $\gamma \in \Gamma_i$, for some i. The former case is trivial, so we assume the latter holds. Take the image by π:

$$\begin{aligned}\pi(w) = \pi(w_1)\pi(\delta)\pi(\gamma)\pi(w_2) = \pi(w_1)\delta_i\gamma_i\pi(w_2) \sim_{R'}\\ \pi(w_1)\pi(w_2) = \pi(w_1w_2) \sim_{R'} 1.\end{aligned}$$

Conversely if $\pi(w) \sim_{R'} 1$, then $\pi(w) = \pi(w_1)v\pi(w_2)$ with $v = a\overline{a}$ or $v = \overline{a}a$ with $a \in \Sigma$ or $v = \delta_i\gamma_i$ for some i. In this case, $v = \pi(u)$, where $u = \delta\gamma$ with $\delta \in \Delta_i$ and $\gamma \in \Gamma_i$ for some i. Hence, $w \sim_R w_1w_2$, which completes the proof.

Now we turn to the simplest non N-free presentation, to wit the presentation $\langle \Sigma; R\rangle$ where $\Sigma = \{a, b, c, d\}$ and R is defined by the relators

$$ac \to 1, bc \to 1, \text{ and } ad \to 1, \tag{5}$$

while $bd \notin \mathrm{dom}(R)$. In order to give a lower bound to the complexity of Inclusion Problem for $\langle \Sigma; R\rangle$ in the case $M(R)$ is non N-free, we consider the following compressed string problem

Compressed 1-Equality Problem for $\langle \Sigma; R\rangle$

Input: an SLP S with $\mathrm{eval}(S) \in \Sigma^*$.

Question: does $\phi_R(\mathrm{eval}(S)) = 1$ hold?

Since a SLP is a special case of context-free grammar, such a problem is obviously a special case of INCLUSION PROBLEM FOR $\langle \Sigma; R\rangle$. This means that the complexity of INCLUSION PROBLEM FOR $\langle \Sigma; R\rangle$ lies between the complexity of COMPRESSED 1-EQUALITY PROBLEM and that of COMPRESSED EQUALITY PROBLEM for the same rewrite system.

In [11, Theorems 5.2, 5.4], COMPRESSED 1-EQUALITY PROBLEM for non N-free is proved to be coNP-complete. Hence, the following straightforwardly follows

Remark 1. If $\langle \Sigma; R\rangle$ is non N-free, then INCLUSION PROBLEM FOR $\langle \Sigma; R\rangle$ is at least coNP-Hard.

4 Conclusions

We studied some conditions for which the inclusion problem for context-free languages is decidable in polynomial time. We showed that verifying whether all the words a context-free language are mapped to the empty word in a cancellative monoid can be reduced to testing the equality of two SLPs in the same monoid. This result solves the inclusion problem for the 2-homogeneous rewrite systems, for which there exists a polynomial solution if the system is N-free, while, in the other case, the problem is at least coNP hard.

In the more general case of complete unitary rewrite systems, we proved that the inclusion of a regular language in a regular set over the generated monoid is solvable in polynomial time.

An open problem on 2-homogeneous rewrite systems is to study the precise complexity of INCLUSION PROBLEM in the non N-free case; moreover, it would be interesting to identify complete unitary rewrite systems for which INCLUSION PROBLEM is solvable in polynomial time.

References

1. Jean Berstel and Luc Boasson. Formal properties of XML grammars and languages. *Acta Inform.*, 38(9):649–671, 2002.
2. Ronald V. Book. Homogeneous Thue systems and the Church-Rosser property. *Discrete Math.*, 48(2-3):137–145, 1984.
3. Y. Cochet. Church-Rosser congruences on free semigroups. In *Algebraic theory of semigroups (Proc. Sixth Algebraic Conf., Szeged, 1976)*, volume 20 of *Colloq. Math. Soc. János Bolyai*, pages 51–60. North-Holland, Amsterdam, 1979.
4. Nachum Dershowitz and Jean-Pierre Jouannaud. Rewrite systems. In *Handbook of theoretical computer science, Vol. B*, pages 243–320. Elsevier, Amsterdam, 1990.
5. M. Farach and M. Thorup. String matching in Lempel-Ziv compressed strings. *Algorithmica*, 20(4):388–404, 1998.
6. S. A. Greibach and E. P. Friedman. Superdeterministic PDAs: a subcase with a decidable inclusion problem. *J. Assoc. Comput. Mach.*, 27(4):675–700, 1980.
7. J. E. Hopcroft. On the equivalence and containment problems for context-free languages. *Math. Systems Theory*, 3:119–124, 1969.

8. D. E. Knuth. A characterization of parenthesis languages. *Inform. Control*, 11(3):269–289, 1967.
9. Yury Lifshits. Processing compressed texts: A tractability border. In *Combinatorial Pattern Matching*, volume 4580 of *Lecture Notes in Comput. Sci.*, pages 228–240. Springer, Berlin, 2007.
10. Markus Lohrey. Word problems for 2-homogeneous monoids and symmetric logspace. In *Mathematical foundations of computer science, 2001 (Mariánské Lázně)*, volume 2136 of *Lecture Notes in Comput. Sci.*, pages 500–511. Springer, Berlin, 2001.
11. Markus Lohrey. Word problems and membership problems on compressed words. *SIAM J. Comput.*, 35(5):1210–1240 (electronic), 2006.
12. Masamichi Miyazaki, Ayumi Shinohara, and Masayuki Takeda. An improved pattern matching algorithm for strings in terms of straight-line programs. *J. Discrete Algorithms (Oxf.)*, 1(1):187–204, 2000.
13. Wojciech Plandowski. Testing equivalence of morphisms on context-free languages. In *Algorithms—ESA '94 (Utrecht)*, volume 855 of *Lecture Notes in Comput. Sci.*, pages 460–470. Springer, Berlin, 1994.
14. Wojciech Rytter. Algorithms on compressed strings and arrays. In *SOFSEM'99: theory and practice of informatics (Milovy)*, volume 1725 of *Lecture Notes in Comput. Sci.*, pages 48–65. Springer, Berlin, 1999.