International Journal of Foundations of Computer Science Vol. 3 No. 2 (1992) 181–191

HOLONOMIC GENERATING FUNCTIONS AND CONTEXT FREE LANGUAGES

A. BERTONI, P. MASSAZZA and N. SABADINI
Dipartimento di Scienze dell'Informazione, Università degli Studi di Milano
Via Comelico 39/41, 20135, Milano, Italy

Received 11 November 1990
Revised 10 June 1992
Communicated by D. Bovet

ABSTRACT

In this paper we give some undecidability and decidability results about context-free languages. First, we prove that the problem of deciding whether a context-free language which admits a holonomic generating function is Turing equivalent to the finiteness question for r.e. sets. Second, we show that the Equivalence Problem is decidable for a suitable class of languages, called LCL_R.

Keywords: Context-free languages, generating functions, decidability, differential equations.

1. Introduction

In this paper, we consider some problems on generating functions of context-free languages. In many cases generating functions techniques give an elegant method for solving problems about languages: for example, it is known from a result of Chomsky and Schuetzenberger[1] that an unambiguous context-free language has an algebraic generating function. This result leads to an analytic technique used in certain cases to solve the ambiguity problem for languages: if the generating function of a context-free language L is not algebraic, then L is inherently ambiguous. For instance, this technique is used by Flajolet[2] to solve a conjecture of Autebert *et al.*[3] In the same paper, some "transcendence criteria" for determining the inherent ambiguity of context-free languages are presented. In this framework, it is interesting to study the class of the holonomic functions as an immediate extension of the class of the algebraic functions: this class has been introduced by Bernstein[4,5] and investigated by Stanley,[6] Lipshitz[7] and Zeilberger.[8]

In this paper we consider two problems concerning holonomic functions. First, we analyze the problem of deciding the holonomicity of the generating function of a context-free language. We show that this problem is Turing equivalent to the finiteness question for r.e. sets; hence, by a result of Reedy–Savitch,[9] it is Turing equivalent to the inherent ambiguity problem for context-free languages.

Furthermore, the method we use gives an independent and simple proof of the Reedy–Savitch result.

Second, we show that there exists a decision algorithm for the Equivalence Problem for some classes of languages, based on the property of holonomicity. In particular we prove that the Equivalence Problem is decidable for a class C of languages if the following conditions are satisfied: C is closed under intersection, every language in C is recursive and admits a holonomic generating function. As an example, we study the Equivalence Problem for the class $\mathrm{LCL_R}$; informally we say that a language L belongs to the class $\mathrm{LCL_R}$ iff $L = A \cap B$, where A is a regular language and B is the set of words that satisfy a finite system of linear constraints on the number of occurrences of symbols. We prove that for this class the Equivalence Problem is decidable.

2. Turing Machines Computations and Context-Free Languages

In this section we recall some results that relate Turing Machines computations with context-free languages. We consider a standard model of deterministic Turing Machine (DTM) with a single one way tape infinite to the right (for a formal presentation and for standard notions on formal languages the reader is referred to Ref. 10). Let M be the set of DTM's; given a machine $\mathrm{M} \in M$, let S be the set of all initial configurations of M and H the set of all its accepting configurations. We denote by $\Rightarrow_{\mathrm{M}}$ (or simply $\Rightarrow$) the transition relation between two successive configurations of M, and by $\Rightarrow *$ its transitive closure. We consider the following languages, introduced by Hartmanis[11]:

Definition 2.1. Given a machine M, let it be such that:

$$L_o(\mathrm{M}) = m\{y_k m y_j^r m \mid y_k \Rightarrow y_j\}^* m\,,$$
$$L_E(\mathrm{M}) = mSm\{y_j^r m y_k m \mid y_j \Rightarrow y_k\}^* H^r mm\,,$$

where m is not a tape or state symbol of M and w^r denotes the reversal of a string w.

The following theorem, proved in Ref. 11, gives a useful relation between context-free languages and accepting computations of a Turing Machine:

Theorem 2.1. For every machine $\mathrm{M} \in M$, $L_o(\mathrm{M})$ and $L_E(\mathrm{M})$ are unambiguous context-free languages and $L_o(\mathrm{M}) \cap L_E(\mathrm{M})$ is the set of all accepting computations of M.

We are interested in a particular class of Turing Machines; in order to define this class, we introduce two functions C_{M} and T_{M} as follows:

Definition 2.2. Given a DTM M, let $y_0 \ldots y_i \ldots y_n$ denote a computation of M and let $|y|$ be the length of the computation y; then we have:

$C_{\mathrm{M}}(x) =$ **if** there is an accepting computation $x \dots y_i \dots y_n$ of M starting with x
then $mxmy_1m \dots my_nmm$
else undefined,

$T_{\mathrm{M}}(x) =$ **if** $C_{\mathrm{M}}(x)$ is undefined
then ∞
else $|C_{\mathrm{M}}(x)|$.

By observing that $\{T_{\mathrm{M}}\}$ is a Blum's complexity measure for M^{12} it is possible to find a subset B of M that satisfies the following conditions:

(1) for every recursively enumerable language L there is a machine $\mathrm{M} \in B$ that accepts L;
(2) if $\mathrm{M} \in B$ then $T_{\mathrm{M}}(x) \geq 2^{2^{|x|}}$,
(3) B is a recursive subset of M.

These conditions ensure that B contains programs for all r.e. sets; moreover every program in B has a "sufficiently high" complexity.

3. Generating Functions of Languages

In this section, we recall the notion of generating function of a language. Given a language $L \subseteq \sum^*$, the *enumeration sequence* of L is the sequence $\{S_n\}$, defined as

$$S_n = \#\{x | x \in L \text{ and } |x| = n\}$$

where $|x|$ denotes the length of x and $\#X$ is the cardinality of the set X.

We associate with this sequence a formal power series, called the *generating function* $F_L(z)$ of the language L, defined as

$$F_L(z) = \sum_0^\infty S_n z^n \,.$$

This series can be interpreted as a function in a complex variable; we observe that it is an analytic function in a neighbourhood of the origin, with a convergence radius ρ such that $\frac{1}{\#\sum} \leq \rho \leq 1$.

In this paper, we consider algebraic and holonomic functions; we recall that a function $f(z)$ is said to be *algebraic* iff there exist polynomials $P_0(z), \dots, P_d(z)$ such that

$$\sum_{k=0}^{\infty} P_k(z) f^k(z) = 0 \,.$$

The least d for which this relation holds is called the degree of f. A useful necessary condition for a function to be algebraic is given by a result of Comtet[14]:

Theorem 3.1. Let $f(z)$ be an algebraic function of degree d,

$$f(z) = \sum_{k=0}^{\infty} N_k z^k \,,$$

then there exist a positive integer q and polynomials $P_0'(k), \ldots, P_q'(k)$ such that:

(1) $P'q(k) \neq 0$,
(2) $\deg(P_j'(k)) < d\,(0 \leq j \leq q)$,
(3) for all k sufficiently large, it holds: $P'q(k)N_{k+q} + \cdots + P_0'(k)N_k = 0$.

We recall a well-known theorem of Chomsky–Schuetzenberger[1] that states that every unambiguous context-free language admits an algebraic generating function; we will use this fact in the following.

In this paper, we consider also the class of the holonomic functions, introduced by Bernstein in the early 70's (see Refs. 4, 5); we recall that a function $f(x)$ is said to be ***holonomic*** iff it satisfies a linear homogeneous differential equation with polynomial coefficients, that is $\sum_{k=0,d} P_k(z)f^{(k)}(z) = 0$. More formally, the general definition in the multivariate case can be stated as follows.

Definition 3.1. A function $f(x_1, \ldots, x_k)$ is said to be holonomic iff it satisfies a system of linear partial differential equations (one for each $i = 1, \ldots, k$) of the form

$$\sum_{j=0}^{d_i} p_{ij}(x_1, \ldots, x_k)\left(\frac{\partial}{\partial x_i}\right)^j f(x_1, \ldots, x_k) = 0\,,$$

where d_i are integers, $1 \leq i \leq k$, and $p_{ij}(x_1, \ldots, x_k)$, $1 \leq i \leq k$, $0 \leq j \leq d_i$, are polynomials with coefficients in the complex field $\mathbb{C}$.

An extensive study of such functions (also called D-finite) has been carried out by Lipshitz.[7]

We denote by $\mathbb{C}[[X]]$ the set of formal series on a set of commutative variables X with coefficients in the complex field $\mathbb{C}$. On the set $\mathbb{C}[[X]]$ we consider the usual operations of sum and Cauchy product together with the Hadamard product ($\odot$) defined as

$$f_1 \odot f_2 = \sum_{\underline{k} \geq 0} a_{\underline{k}}\underline{x}^{\underline{k}} \odot \sum_{\underline{k} \geq 0} b_{\underline{k}}\underline{x}^{\underline{k}} = \sum_{\underline{k} \geq 0} a_{\underline{k}}b_{\underline{k}}\underline{x}^{\underline{k}}\,.$$

The following theorem summarizes some closure results of the class of the holonomic series $\mathbb{C}[[X]]_h$ with respect to the usual operators on formal series:

Theorem 3.2. The class $\mathbb{C}[[X]]_h$ of the holonomic series on X with coefficients in $\mathbb{C}$ is closed with respect to the operators of sum, Cauchy product, Hadamard product.

Example 3.1. Let us consider the rational series

$$\varphi_1 = \sum_{n=0}^{\infty} a^n b^n c^n = \frac{1}{1 - abc}$$

and

$$\varphi_2 = \sum_{\substack{i \geq 0 \\ j \geq 0 \\ k \geq 0}}^{\infty} \binom{i+j+k}{i,\, j,\, k} a^i b^j c^k = \frac{1}{1 - (a+b+c)}\,.$$

The Hadamard product of φ_1 and φ_2 is the series

$$\varphi_1 \odot \varphi_2 = \sum_{n=0}^{\infty} \frac{(3n)!}{(n!)^3} a^n b^n c^n$$

that is neither rational nor algebraic but holonomic (a proof that $\varphi_1 \odot \varphi_2$ is transcendental can be found in Refs. 2, 15).

We observe that it is possible to give a constructive proof of the previous theorem: in particular in Ref. 8 a detailed analysis of the closure properties of the class $\mathbb{C}[[X]]_h$ is carried out in order to give an algorithm for verifying some combinatorial identities. We recall here another theorem that is used in the sequel to prove our results (the proof is straightforward, see for instance Refs. 15, 16).

Theorem 3.3. Let $F_L(z) = \sum_{n=0,\infty} S_n z^n$ be the generating function of a language L. If $F_L(z)$ satisfies a linear differential equation with polynomial coefficients, then the sequence $\{S_n\}$ verifies a linear recurrence equation with polynomial coefficients, and vice versa.

It is easy to show that an algebraic function also satisfies a linear differential equation with polynomial coefficients[6,16]; hence, the class of the algebraic generating functions is a proper subclass of the class of the holonomic generating functions. In this setting, the previous theorem gives us a simple proof of Comtet's result.

4. The Holonomicity Question for Generating Functions of Context-Free Languages

We know that there exist context-free languages with non-algebraic generating functions; for instance the generating function of the language

$$L = \{w \in \{a, b, c\}^* | \#_a(w) \neq \#_b(w) \text{ or } \#_b(w) \neq \#_c(w)\},$$

where $\#_\sigma(w)$ is the number of occurrences of the symbol σ in the string w, is

$$\varphi_L(z) = \frac{1}{1-3z} - \sum_{n\geq 0} \frac{(3n)!}{(n!)^3} z^n,$$

which is not algebraic[2] but holonomic.

So, it is interesting to study the holonomicity question for generating functions of context-free languages. In this section it is proved that this question is Turing equivalent to the finiteness question for r.e. sets.

First of all, by recalling the definition of the languages $L_o(\mathrm{M})$ and $L_E(\mathrm{M})$, we are able to give a result which relates the holonomicity of generating functions of context-free languages to the holonomicity of generating functions of accepting computations of Turing Machines. In fact, we can prove the following:

Lemma 4.1. $F_{L_o(\mathrm{M})\cup L_E(\mathrm{M})}$ is holonomic iff $F_{L_o(\mathrm{M})\cap L_E(\mathrm{M})}$ is holonomic.

Proof. The generating function $F_{L_o(\mathrm{M})\cup L_E(\mathrm{M})}(z)$ satisfies the relation

$$F_{L_o(\mathrm{M})\cup L_E(\mathrm{M})}(z) = F_{L_o(\mathrm{M})}(z) + F_{L_E(\mathrm{M})}(z) - F_{L_o(\mathrm{M})\cap L_E(\mathrm{M})}(z).$$

Since $L_o(\mathrm{M})$ and $L_E(\mathrm{M})$ are unambiguous, their generating functions are algebraic, hence holonomic. The result follows from the closure properties of the class $\mathbb{C}[[X]]_h$.

□

The following result gives a reduction from the finiteness question for r.e. sets to the holonomicity question. Let B be the set defined at the end of Sec. 2.

Lemma 4.2. Given a DTM $\mathrm{M} \in B$, then $F_{L_o(\mathrm{M}) \cup L_E(\mathrm{M})}$ is holonomic (algebraic, rational) iff the language accepted by M is finite.

Proof. We prove the result in the holonomic case (the proof in the algebraic or rational case is similar).

From Lemma 4.1 it follows that $F_{L_o(\mathrm{M}) \cup L_E(\mathrm{M})}$ is holonomic iff $F_{L_o(\mathrm{M}) \cap L_E(\mathrm{M})}$ is holonomic; so, we can consider this last function. Let $c_n = \#\{x | T_{\mathrm{M}}(x) = n\}$; from Theorem 2.1 and by observing that different initial configurations of M produce different sequences of computations, we have:

$$F_{L_o(\mathrm{M}) \cap L_E(\mathrm{M})} = \sum_{n=0,\infty} c_n z^n \,.$$

Now, given $\mathrm{M} \in B$, we have two possibilities:

(1) the language accepted by M is finite; in this case $F_{L_o(\mathrm{M}) \cap L_E(\mathrm{M})}(z)$ is a polynomial, i.e. an algebraic function. From Lemma 4.1 and by recalling that an algebraic function is holonomic, we can conclude that $F_{L_o(\mathrm{M}) \cup L_E(\mathrm{M})}$ is holonomic too;

(2) the language accepted by M is infinite; let us suppose that the generating function $F_{L_o(\mathrm{M}) \cap L_E(\mathrm{M})} = \sum_{n=0,\infty} c_n z^n$ is holonomic; by Theorem 3.3 we know that the sequence $\{c_n\}$ verifies a linear recurrence equation with polynomial coefficients. We can find two integers N, $q > 0$ such that, for all $n \geq N$, at least one of the following coefficients $c_n, c_{n+1}, \ldots, c_{n+q}$ is different from 0 (if all the coefficients $c_n, c_{n+1}, \ldots, c_{n+q}$ were equal to zero then we would have $c_n = 0$ for all $n \geq N$; in this case $F_{L_o(\mathrm{M}) \cap L_E(\mathrm{M})}$ would be a polynomial and the set of accepting computations of M would be finite against the hypothesis that the language accepted by M is infinite).

Then, for all $n \geq N$, we have:

$$\#\{x | T_{\mathrm{M}}(x) \leq n\} \geq \lfloor (n-N)/q \rfloor \,.$$

Since $\mathrm{M} \in B$, it holds that $T_{\mathrm{M}}(x) \geq 2^{2^{|x|}}$ and

$$\#\{x | 2^{2^{|x|}} \leq n\} \geq \lfloor (n-N)/q \rfloor \,.$$

Denoting by k the cardinality of the alphabet of M, let $f(n)$ be the function

$$f(n) = \#\{x | 2^{2^{|x|}} \leq n\} = \#\{x \,|\, |x| \leq \log\log n\} \,.$$

We obtain:

$$f(n) \leq g(n) = \sum_{i=0,\lg\lg n} k^i = o(n) \,.$$

So, informally, we have the contradiction $o(n) \geq \lfloor (n-N)/q \rfloor$.

We can conclude that, in this case, $F_{L_o(\mathrm{M})\cap L_E(\mathrm{M})}$ is not holonomic, hence $F_{L_o(\mathrm{M})\cup L_E(\mathrm{M})}$ is not holonomic too. □

We give now an outline of the converse reduction.

Lemma 4.3. The holonomicity problem for context-free languages is Turing reducible to the finiteness problem for r.e. sets.

Proof. We recall that the generating function of a language L is holonomic iff the sequence $\{a_n\}$ of the number of words of length n in L verifies a linear difference equation with polynomial coefficients. Let us consider the pair $\langle E, C\rangle$, where E is a linear difference equation of finite order with polynomial coefficients, and C is a finite vector of integers (initial conditions): this pair uniquely identifies a function $f : N \to N$. Let $\{E_k\}$ be an enumeration of such pairs, and f_j be the function identified by E_k. Moreover, let $\{G_j\}$ be an enumeration of context free grammars and L_j be the language generated by the grammar of index j.

Since the function $f(k, n) = f_k(n)$ is a total recursive function, the relation $R(j, k, n)$ defined as

$$R(j, k, n) = \textbf{if } \#\{w \mid w \in L_j,\ |w| = n\} \neq f(k, n) \textbf{ then } 1 \textbf{ else } 0\,,$$

is a recursive relation.

Let T be the class of (indices of) context free grammars that generates languages with non-holonomic generating functions; then we have $T = \{j \mid \forall k \exists n\ R(j, k, n)\}$, with R recursive; now, the reduction follows from standard results.[13] Analogous considerations are valid in the algebraic and rational cases. □

As a conclusion, we can state our main result:

Theorem 4.1. The problem of deciding the holonomicity (algebricity, rationality) of the generating function of a context-free language is Turing equivalent to the finiteness question for r.e. sets.

By recalling the theorem of Chomsky–Schuetzenberger on the algebricity of the generating function of an unambiguous context-free language, we obtain as an interesting application of the previous theorem, the possibility of proving, without using combinatorial lemmas, the main result of Reedy–Savitch[9]:

Theorem 4.2. Given a DTM $\mathrm{M} \in B$, then $L_o(\mathrm{M}) \cup L_E(\mathrm{M})$ is inherently ambiguous iff the language accepted by M is infinite.

Proof. If the language accepted by M is finite, also $L_o(\mathrm{M}) \cap L_E(\mathrm{M})$ is finite, hence $L_o(\mathrm{M}) \cup L_E(\mathrm{M})$ is unambiguous. If the language accepted by M is infinite, by Lemma 4.2 the function $F_{L_o(\mathrm{M})\cup L_E(\mathrm{M})}$ is not algebraic, so $L_o(\mathrm{M}) \cup L_E(\mathrm{M})$ is inherently ambiguous. □

5. The Equivalence Problem for Languages: The Class $\mathbf{LCL_R}$

In this section we consider the Equivalence Problem for languages. It is well known that the problem is decidable for the class of regular languages; on the other hand, an undecidability result holds for context-free languages.

To state the Equivalence Problem in this general setting we need the notion of specification of a class:

Definition 5.1. A *specification* for a class C of recursive languages on the alphabet $\sum$ is a pair $\langle S, [\ ]\rangle$, where S is a recursive subset of $\{0, 1\}^*$ and $[\ ]$ is a surjective function $[\ ] : S \to C$ such that the relation $\{(x, s) | x \in \sum^*, s \in S, x \in [s]\}$ is recursive.

Given a specification $\langle S, [\ ]\rangle$ for a class of languages C, the Equivalence Problem for C can be stated as follows:

Problem (Equivalence for C).
Instance: $s_1, s_2 \in S$.
Question: $[s_1] = [s_2]$?

In the following we state a sufficient condition for which the previous problem is decidable. Let us denote by Eq the set of the linear differential equations having polynomial coefficients. We give the following:

Definition 5.2. A class C of languages is said to be *c-holonomic* if there exists a specification $\langle S, [\ ]\rangle$ for C and a total computable function $f : S \to \text{Eq}$ such that $\forall s \in SF_{[s]}(z)$ verifies the differential equation $f(s)$; moreover, C is said to be closed (constructively) under intersection (briefly *c-closed*) if there exists a total computable function $g : S \times S\,\text{s.t.} : \forall s_1, s_2 \in S(g(s_1, s_2) = s_3 \Rightarrow [s_1] \cap [s_2] = [s_3])$.

Informally a class C is c-holonomic if for any language L in C it is possible to compute the differential equation satisfied by $F_L(z)$; furthermore, C is c-closed if, given two representations $s_1, s_2 \in S$, it is possible to compute a representation for the language $[s_1] \cap [s_2]$.

Now, we prove that if C is a class c-holonomic and c-closed of recursive languages, then the Equivalence Problem for C is decidable. Formally we have:

Theorem 5.1. Let C be a class of recursive languages such that C is c-closed and c-holonomic, then the Equivalence Problem for C is decidable.

Proof. Consider two arbitrary elements s_1, s_2 in S. Since C is c-closed we can compute $s_3 = g(s_1, s_2)$ such that $[s_3] = [s_1] \cap [s_2]$.

It is easy to show that $[s_1] = [s_2]$ iff it holds

$$F_{[s_1]}(z) = F_{[s_2]}(z) = F_{[z_3]}\,.$$

Since C is c-holonomic we can compute the linear differential equations (with polynomial coefficients) satisfied by the functions $F_{[s_i]}(i = 1, 2, 3)$ and also, by Theorem 3.2, the linear differential equations satisfied by $F_{[s_3]}(z) - F_{[s_1]}(z)$ and $F_{[s_3]}(z) - F_{[s_2]}(z)$.

Let us consider $F_{[s_3]}(z) - F_{[s_1]}(z)$ and $F_{[s_3]}(z) - F_{[s_2]}(z)$: we denote by $\{a_n\}$ and $\{b_n\}$ the successions of the coefficients in the respective Taylor's expansions; by Theorem 3.3 we can compute two integers q_1, q_2 such that $\{a_n\}$ and $\{b_n\}$ satisfy two linear difference equations (with polynomial coefficients)

$$\sum_{k=0}^{q_1} q_k(n)a_{n+k} = 0\,,$$
$$\sum_{k=0}^{q_2} p_k(n)b_{n+k} = 0\,.$$

We have that $[s_1] = [s_2]$ iff $a_0 = a_1 = \cdots = a_{q_1} = 0$ and $b_0 = b_1 = \cdots = b_{q_2} = 0$. These last conditions are satisfied iff

$$\#\{w \mid w \in [s_1],\, |w| = k\} = \#\{w \mid w \in [s_3],\, |w| = k\} = \#\{w \mid w \in [s_2],\, |w| = k\}$$

for all k, $0 \le k \le \mathrm{Max}\,\{q_1, q_2\}$.

Since the relation $w \in [s]$ is recursive we conclude that the Equivalence Problem for C is decidable. □

As an immediate application, we observe that the decidability of the Equivalence Problem for the class R of the regular languages follows directly since R is closed under intersection and the generating function of a regular language is rational (hence holonomic).

As a non-trivial example, we study the Equivalence Problem for the class $\mathrm{LCL_R}$ (Linearly Constrained Languages-Rational). In order to define this class, we introduce the notion of linear constraint; a *linear constraint* is a predicate of the type

$$\sum_{\sigma \in \Sigma} \lambda_\sigma \#_\sigma(w)\ rel\ k$$

where *rel* belongs to the set $\{=, \neq, <, >, \le, \ge\}$ and k is an integer. Informally we say that a language L belongs to the class $\mathrm{LCL_R}$ iff $L = A \cap B$, where A is a regular language and B is the set of words that satisfy a finite system of linear constraints. This class is an extension of the class of the regular languages and it is a proper subclass of the class LCL defined in Ref. 17. It also contains some languages that are not context-free: for example, the language $L_1 = \{a^n b^n c^n \mid n \ge 0\}$ is in $\mathrm{LCL_R}$ since $L_1 = A_1 \cap B_1$ where $A_1 = a^*b^*c^*$ and $B_1 = \{w \in \{a, b, c\}^* \mid \#_a(w) = \#_b(w),\, \#_b(w) = \#_c(w)\}$.

The main interest in this class is due to the fact that it satisfies the hypothesis of Theorem 5.1. More precisely, by the results contained in Ref. 17 the following lemma is easily proved.

Lemma 5.2. The class $\mathrm{LCL_R}$ is c-holonomic.

Furthermore, we have:

Lemma 5.3. The class $\mathrm{LCL_R}$ does contain recursive languages only.

Proof. If a language L belongs to $\mathrm{LCL_R}$ then $L = A \cap B$, where A is a regular language and B is identified by a finite set of linear constraints. Hence, given a word

w, we have that the relation $w \in L$ is recursive since $w \in A$ and $w \in B$ are recursive (for the second relation it is sufficient to count the number of occurrences of each symbol and to test whether the finite system of linear constraints is verified). $\square$

At last, it holds:

Lemma 5.4. The class $\mathrm{LCL}_\mathbf{R}$ is c-closed.

Proof(outline). Given two languages $L_1 = A_1 \cap B_1$, $L_2 = A_2 \cap B_2$, in $\mathrm{LCL}_\mathbf{R}$, their intersection is the language $L_3 = L_1 \cap L_2 = A_1 \cap B_1 \cap A_2 \cap B_2 = A_3 \cap B_3$ where $A_3 = A_1 \cap A_2$ is a regular language and $B_3 = B_1 \cap B_2$ is specified by the finite system of linear constraints obtained by joining B_1 and B_2. Hence, by definition, L_3 is in $\mathrm{LCL}_\mathbf{R}$. $\square$

Hence, by applying Theorem 5.1, we can conclude by stating the following theorem:

Theorem 5.5. The Equivalence Problem for the class $\mathrm{LCL}_\mathbf{R}$ is decidable.

Acknowledgements

This research has been supported by the Italian Ministry of University and Scientific Research Funds (40%) and by CEC under Esprit B.R.A. Working group n.3166-ASMICS.

References

1. N. Chomsky and M. P. Schuetzenberger, The algebraic theory of context-free languages, *Computer Programming and Formal Systems* (North Holland, 1963) pp. 118–161.
2. P. Flajolet, Analytic models and ambiguity of context-free languages, *Theor. Comput. Sci.* **49** (1987) 283–309.
3. J. M. Autebert, J. Beauquier, L. Boasson and M. Nivat, Quelques problemes ouverts en theorie des languages algebriques, *R.A.I.R.O. Inf. Theor.* **13** (1979) 363–379.
4. I. N. Bernstein, Modules over the ring of differential operators. A study of the fundamental solutions of equations with constant coefficients, *Functional Analysis and Its Applications* **5**, 2 (1971) 89–101.
5. I. N. Bernstein, The analytic continuation of generalized functions with respect to a parameter, *Functional Analysis and Its Applications* **6**, 4 (1972) 273–285.
6. R. P. Stanley, Differentiably finite power series, *European Journal of Combinatorics* **1** (1980) 175–188.
7. L. Lipshitz, The diagonal of D-finite power series is D-finite, *Journal of Algebra* **113** (1988) 373–378.
8. D. Zeilberger, A holonomic systems approach to special functions identities, *J. Comput. Appl. Math.* **32** (1988) 321–368.
9. A. Reedy and W. Savitch, The Turing degree of the inherent ambiguity problem for context-free languages, *Theor. Comput. Sci.* **1** (1975) 77–91.
10. J. E. Hopcroft and J. D. Ullman, *Formal Languages and Their Relation to Automata* (Addison-Wesley, 1969).
11. J. Hartmanis, Context-free languages and Turing machines computations, *Proc. Symp. on Appl. Mathematics*, 1967, pp. 42–51.

12. M. Blum, Machine independent theory of the complexity of recursive functions, *J. ACM* **14** (1967).
13. H. Rogers, Jr., *Theory of Recursive Functions and Effective Computability* (McGraw-Hill, New York, 1967).
14. L. Comtet, Calcul pratique des coefficients de Taylor d'une fonction algebrique, *Einsegnement Math.*, (1964) 267–270.
15. P. Massazza and N. Sabadini, Some applications and techniques for generating functions, *Proc. CAAP, Lecture Notes in Computer Science* **351** (Springer-Verlag, 1989) pp. 321–356.
16. P. Massazza, Problemi conteggio e funzioni generatrici olonomiche, Tesi di Dottorato in Informatica, Università di Milano, 1990.
17. P. Massazza, Holonomic functions and their relation to linearly constrained languages, *R.A.I.R.O. Inf. Theor.*, to appear.