数理解析研究所講究録
第 421 巻 1981 年 15-29

PUSHDOWN AUTOMATA WITH TERMINAL LANGUAGES

Jacques Sakarovitch

Laboratoire d'Informatique Théorique et Programmation
C.N.R.S., Paris

and

Research Institute for Mathematical Sciences(1)
Kyoto University, Kyoto.

Abstract: We propose to modify the definition of the recognition of a word by a pushdown automaton by requiring that not only the automaton will be in a terminal state after the word has been read but also that the content of the store will belong to a fixed language. We show here that this definition does not increase the recognition power of the pushdown automata when the language in the store is chosen to be context free.

---

(1) This paper has been written during the stay of the author at the R.I.M.S.: the supports from the J.S.P.S., the I.N.R.I.A., and the Ministère des Affaires Etrangères that made this stay possible is gratefully acknowledged.

A pushdown automaton A is usually equipped with terminal states and a word f over the input alphabet of A is said to be accepted, or recognized, by A if one of the computations of A with f as input leads A into one of its terminal states. This notion of terminal states however is not truly consistent with the definitions taken in the general theory of (finite or infinite) automata: this theory would lead to define terminal configurations rather than terminal states - and then A would recognize f if one of its computations over f yields a terminal configuration.

It is immediate that a set of terminal configurations is defined by a family of languages over the pushdown store alphabet, which we call a family of terminal languages. We shall show that the recognition power of pushdown automata is not increased by this new definition as long as the terminal languages are chosen to be context free. This result, though it may seem surprising at first sight, appears to be a generalization of the substitution theorem for context free languages. It will be proved here without any construction of grammars or of automata, but as another example of the application of a more algebraic theory of formal languages - especially of rational and context free languages - as developed for instance in [1] and [2].

1. A new definition of recognition by pushdown automata.

Let $A = <Q,X,Y,\delta,q_-,y_->$ be a pushdown automata, where Q is the set of states, X the input alphabet, Y the pushdown store alphabet, δ the transition function, q_- the initial state, and y_- the initial symbol in the pushdown store. The only difference with the classical definition (see for instance [5]) is that we do not distinguish a priori any subset of Q as set of terminal states.

The set $Q \times Y^*$ is called the set of configurations of A and any subset of configurations is then a family of subsets of Y^*, indexed by Q. As usual we note

$$(f, q_-, y_-) \overset{*}{\underset{A}{\vdash}} (1, q, w)$$

if one computation of A, starting with the initial configuration (q_-, y_-) and with f on the input tape, leads A into the configuration (q, w) after f is read completely.

DEFINITION 1: Let $A = <Q,X,Y,\delta,q_-,y_->$ be a pushdown automaton and let $\mathcal{C} = \{L_q | q \in Q \; L_q \subset Y^*\}$ be a family of subsets of Y^*. We call such a $\mathcal{C}$ a family of terminal languages for A and we shall say that

$$L(A,\mathcal{C}) = \{f \in X^* | (f,q_-,y_-) \overset{*}{\underset{A}{\vdash}} (1,q,w) \quad w \in L_q\}$$

is the language recognized by A and

This definition is of course a mere generalization of the classical notion of acceptance by pushdown automata: for any subset T of Q let $\mathcal{C}_T$ be the family of terminal languages such that $L_q = Y^*$ if q is in T and $L_q = \phi$ otherwise. Then $L(A,\mathcal{C}_T)$ is exactly the language recognized by A with T as set of terminal states. If $\mathcal{C}$ is such that $L_q = 1$[1] for every q then $L(A,\mathcal{C})$ is the so-called language accepted by A by empty store.

Along the same line we have the following proposition, which belongs to folklore:

PROPOSITION 1: Let A be a pushdown automaton and let $\mathcal{C}$ be a family of rational terminal languages for A. Then $L(A,\mathcal{C})$ is a context free language.

The result we have announced in the introduction is then stated as:

THEOREM 1: Let A be a pushdown automaton and let $\mathcal{C}$ be a family of context free terminal languages for A. Then $L(A,\mathcal{C})$ is a context free language.

What may be surprising in that result is that two distinct pushdown automata seem to be necessary in order to recognize $L(A,\mathcal{C})$ when $\mathcal{C}$ is a family of context free terminal languages (for the sequel of the discussion we suppose that

---

(1) We denote by 1 the empty word of any free monoid.

only one terminal language - say L_p - is non empty; for context free languages are closed under finite union this assumption causes no loss of generality): the first pushdown automaton is A and the second one is B, a pushdown automaton which recognizes L_p. And the problem comes from the fact that with two pushdown automata working in the same recognition process it is possible to recognize languages which are not context free. Indeed, because those two pushdown automata are used one after the other, it is possible to simulate them with only one pushdown automaton C, and this explains the result.

Here is, roughly described, a construction for such an automaton C. The pushdown store of C is made of the pushdown store of A (the A-part of the store) standing on the pushdown store of B (the B-part). When C reads a word f of X^* it has the same behaviour of A as long as the length of the A-part in its store is greater than one. When there is only one letter y in the A-part, C has to guess whether this letter will remain in the store of A during all the sequel of the reading of f and whether A will stop in the state p. If C guesses no, it goes on with its simulation of A; but if it guesses yes, it switches to the behaviour of B, considering y as an input letter and computing with the B-part of the store which is then accesible since the A-part is empty.

Of course the formal construction of such an automaton C would be rather heavy, and the proofs on it even heavier. As we said our method is completely different and makes use of already known results on context free languages.

2. Rational relations

Rational relations from one free monoid into another one are the multivalued mappings that can be realized by the so-called a-transducers (cf. [5]). They have proved to be a very basic tool for the study of context free languages (see [1]). We just recall here the definitions and the properties we shall use in the sequel.

Let M be a monoid. The family of rational sets of M, which is denoted by Rat M, is the smallest family of subsets of M which contains the finite subsets and which is closed under the operations of union, product, and "star" $(R^* = \bigcup_{n=0}^{\infty} R^n)$.

DEFINITION 2: A relation τ from X^* into Y^* is a rational relation if its graph $\hat{\tau}$ is a rational subset of the monoid $X^* \times Y^*$.

If f is in X^* $f\tau = \{w \in Y^* \mid (f,w) \in \hat{\tau}\}$ by definition of the graph of a relation and if L is a subset of X^* the image of L by τ is $L\tau = \bigcup_{f \in L} f\tau$. The inverse of τ, denoted by τ^{-1}, is defined by $w\tau^{-1} = \{f \in X^* \mid (f,w) \in \hat{\tau}\}$ for any w in Y^*.

PROPERTY 1: The inverse of a rational relation is a rational relation.

PROPERTY 2: The image of a context free language by a rational relation is a context free language.

DEFINITION 3: A rational transducer from X^* into Y^* is a triple (λ,μ,ν) where μ is a homomorphism from X^* into $(\text{Rat } Y^*)^{N\times N}$, the monoid of square matrices of dimension N the entries of which are rational subsets of Y^* and where λ (resp. ν) is a row-vector (resp. a column-vector) of dimension N the entries of which are in $\text{Rat } Y^*$ as well. The transducer (λ,μ,ν) realizes the relation τ from X^* into Y^* which is defined by

$$\forall f \in X^* \qquad f\tau = \lambda . f\mu . \nu$$

THEOREM 2 (Kleene-Schützenberger): A relation τ is rational if, and only if, there exists a rational transducer which realizes τ.

3. Algebraic substitutions

The following definition and result go back to the very beginning of the study of context free languages (cf. [5]).

DEFINITION 4: A relation τ from X^* into Y^* is a substitution if τ is a homomorphism (of monoids) from X^* into $\mathcal{P}(Y^*)$, i.e. for every f in X^*, $f=f_1f_2\cdots f_n$ with f_i in X, $f\tau=(f_1\tau)(f_2\tau)\cdots(f_n\tau)$. We say that a substitution τ is algebraic if for every x in X, $x\tau$ is a context free language of Y^*.

THEOREM 3 (Bar Hillel-Perles-Shamir): The image of a context free language by an algebraic substitution is a context free language.

4. The pushdown reduction.

Let Y be any alphabet; let us denote by $\bar{Y}$ a copy of Y, disjoint of Y, and by $\tilde{Y}$ the union of Y and $\bar{Y}$. For every letter z in Y let $\bar{z}$ be the letter of $\bar{Y}$ which corresponds to z in the canonical bijection between Y and $\bar{Y}$.

Let σ be the mapping from $\tilde{Y}^*$ into itself inductively defined by:

$$1\sigma = 1 ;$$

$(wz)\sigma = u$ if $w\sigma = uy$ and $z = \bar{y}$ with y in Y;

$(wz)\sigma = (w\sigma)z$ otherwise.

The inverse image of 1 by σ, which we denote by P_Y, is called the <u>restricted Dyck set</u> (or "semi-Dyck set") and is a context free language cf.[1,5]. The image of $\tilde{Y}^*$ by σ, $\tilde{Y}^*\sigma=\tilde{Y}^*\backslash\tilde{Y}^*\{y\bar{y} \mid y\in Y\}\tilde{Y}^*$, is a rational set (the "set of reduced words") which will be denoted by R.

The partial mapping ρ from $\tilde{Y}^*$ into itself, which we shall call the <u>pushdown reduction</u> is then defined by

$w\rho = w\sigma$ if $w\sigma$ is in $K = \bar{Y}^*Y^*$

$w\rho$ is not defined otherwise.

Thus $\rho=\sigma\iota_K$ where ι_K denotes the intersection with the (rational) set K.

PROPOSITION 2 [3]: <u>The image of a context free language of</u> $\tilde{Y}^*$ <u>by the inverse of the mapping</u> σ (<u>resp. of the mapping</u> ρ) <u>is a context free language.</u>

Proof: Let τ be the algebraic substitution from $\tilde{Y}^*$ into itself defined by $z\tau=P_YzP_Y$ for every z in $\tilde{Y}$. Let $w=z_1z_2\cdots z_n$ be a word of $\tilde{Y}^*$, with z_i in $\tilde{Y}$; from the properties of σ (cf. [1,5]) it is easy to verify that

$$w\sigma^{-1} = P_Y z_1 P_Y z_2 P_Y \cdots P_Y z_n P_Y \quad \text{if} \quad w \in R$$

$$w\sigma^{-1} = \phi \quad \text{otherwise}$$

Thus, since $P_Y P_Y = P_Y$, $\sigma^{-1} = \iota_R \cdot \tau$; similarly $\rho = \iota_K \cdot \tau$. And the conclusion follows from theorem 3 and the fact that the intersection of a context free language with a rational language is a context free language.■

5. Representation theorem for pushdown automata

Let $A = \langle Q, X, Y, \delta, q_-, y_- \rangle$ be a pushdown automaton. The following notations proved to be convenient: for every q in Q let θ_q be the mapping from X^* into Y^* defined by

$$f\theta_q = \{w \mid (f, q_-, y_-) \overset{*}{\underset{A}{\vdash}} (1, q, w)\}$$

For every q in Q we also denote by ν_q the boolean vecter of dimension Q all the entries of which are 0 but the q-th one (which in 1).

The following theorem of representation of pushdown automata is due to Nivat [6,7]; it is the analogous of the representation theorem of context free grammars of Shamir [11].

THEOREM 4 (Shamir-Nivat): Let $A = \langle Q, X, Y, \delta, q_-, y_- \rangle$

be a pushdown automaton and let ρ be the pushdown reduction on $\tilde{Y}^*$. There exists a rational transducer $(\lambda,\mu,.)$ from X^* into Y^*, of dimension Q, such that for every q in Q, and every f in X^* one has:

$$f\theta_q = (\lambda . f\mu . \nu_q)\rho$$

Outline of the proof: The transitions of A are defined by the mapping

$$\delta: (X \cup \{1\}) \times Q \times Y \to (Q \times Y^*)$$

From δ one defines η, element of $(\text{Rat } \tilde{Y}^*)^{Q\times Q}$, and α, homomorphism from X^* into $(\text{Rat } \tilde{Y}^*)^{Q\times Q}$, by the following:

$$\forall q,q' \in Q \quad \eta_{q,q'} = \{\bar{y}v \mid (q,v) \in \delta(1,q',y)\}$$

$$\forall a \in X \quad \forall q,q' \in Q \quad a\alpha_{q,q'} = \{\bar{y}v \mid (q,v) \in \delta(a,q',y)\}$$

From the definition of a pushdown automaton, every entry of these matrices is a finite subset of $\tilde{Y}^*$.

Let ζ be the row-vector of dimension Q, all the coordinates of which are zero, but the q_--th coordinate which is y_-. Let then $\lambda=\zeta.\eta^*$ and for every a in X $a\mu=a\alpha.\eta^*$. The entries of the matrix η^* are rational subsets of $\tilde{Y}^*$ (cf. [2]) and so are those of λ and $a\mu$. For every q the transducer (λ,μ,ν_q) is thus a rational one and the equality $f\theta_q=(\lambda.f\mu.\nu_q)\rho$ is then easily checked by induction on the length of f. ∎

Now we are able to explain why we called ρ the pushdown reduction. We interprete a word of $\tilde{Y}^*$ as a sequence of actions on the pushdown store as follows: the meaning of a letter y in Y is: 'push the letter y in the store', while the meaning of the letter $\bar{y}$ is: 'pop the letter y up from the store'. And $w\rho$ is the shortest sequence of elementary actions which has the same effect than w on any store. Thus for instance $y\bar{y}$ is equivalent to no action and $y\bar{y}\rho=1_{\tilde{Y}*}$; the word $y\bar{z}$, with z different from y, leads to an impossibility since it is not possible to pop the letter z immediately after having pushed a different letter y in the store; this is represented by the empty set, i.e. the zero for the multiplication of subsets.

6. Proof of theorem 1

Let A be a pushdown automaton, and $\mathcal{C}$ be a family of terminal languages for A. With the notation of the previous paragrah we have

$$L(A, \mathcal{C}) = \bigcup_{q \in Q} L_q \theta_q^{-1}$$

If for every q in Q we denote by τ_q the rational relation from X^* into $\tilde{Y}^*$ which is realized by the rational transducer (λ,μ,ν_q) given by theorem 4 we have

$$L(A, \mathcal{C}) = \bigcup_{q \in Q} (L_q \rho^{-1}) \tau_q^{-1}$$

and conclusion follows from proposition 2, properties 1 and 2, and the fact that a finite union of context free languages is a context free language. Q. E. D.

7. Further and related results

The representation theorem above is the very basic step for building an algebraic theory of pushdown automata. The result we have presented here is the very first one which can be obtained within that theory (and it was indeed inspired by it) and which can be stated and proved with not much preparation.

It turned out that this theory is especially fitted to the study of deterministic pushdown automata, but for this case we must consider formal power series and $\mathbb{N}$-relations rather than languages and relations. Within this framework it is possible to give a representation theorem for deterministic pushdown automata similar to theorem 4 (cf. [10]) and to get then results which can be stated in quite a classical setting. For instance if A is a deterministic pushdown automaton we note

$$(f, q_-, y_-) \overset{*t}{\underset{A}{\vdash}} (1, q, w)$$

a terminal computation of A (i.e. there is no ε-move possible in the configuration (q,w)). If $\mathcal{C}$ is a family of terminal languages for A we define

$$Lt(A, \mathcal{T}) = \{f \mid (f, q_-, y_-) \overset{*t}{\underset{A}{\vdash}} (1, q, w) \; w \quad L_q\}$$

and we have:

THEOREM 5 [10]: Let A be a deterministic pushdown automaton and let $\mathcal{T}$ be a family of unambiguous context free terminal languages for A. Then $Lt(A, \mathcal{T})$ is an unambiguous context free language.

Together with the same representation theorem for deterministic pushdown automata a precise study of the pushdown reduction allows to prove:

THEOREM 6 [9]: For any deterministic context free language L of X^* there exists a right regular equivalence ω on X^* such that:

i) L is a union of classes modulo ω

ii) there exists a rational set R of representatives of the classes modulo ω contained in L.

And it is shown [9,10] that this theorem plays a rôle similar to the one of the classical iteration theorems for deterministic context free languages (see [4,5,8]).

REFERENCES

[1] J. Berstel - Transductions and context free languages, Teubner, 1978.

[2] S. Eilenberg - Automata, Languages, and Machines, Vol. A, Academic Press, 1974.

[3] M. Fliess - Deux applications de la représentation matricielle d'une série rationnelle non commutative, J. of Algebra 19, 1971, 344-353.

[4] S. Ginsburg and S. Greibach - Deterministic context free languages, Inform. and Control 9, 1966, 620-648.

[5] M. Harrison - Introduction to Formal Language Theory, Addison-Wesley, 1978.

[6] M. Nivat - Transductions des langages de Chomsky, Ann. Inst. Fourier 18, 1968, 339-445.

[7] M. Nivat - Sur les automates à mémoire pile, in Proceedings of International Computing Symposium 70 (W. Itzfeld ed.), 1973, 655-663.

[8] W. Ogden - Intercalation theorems for pushdown store and stack languages, Ph. D. Thesis, Standford, 1968.

[9] J. Sakarovitch - Un théorème de transversale rationnelle pour les automates à pile déterministes, in Proceedings of the 4th GI Conference on Theoretical Computer Science (K. Weihrauch, ed.), Lecture Notes in Computer Science n° 67, Springer, 1979, 276-285.

[10] J. Sakarovitch - Syntaxe des langages de Chomsky, Th. Sci. Univ. Paris 7, Paris, 1979.

[11] E. Shamir - A representation theorem for algebraic and context-free power series in non commuting variables, Inform. and Control 11, 1967, 239-254.