Rewriting in P Systems: An Algebraic Approach

Rodica Ceterchi

Faculty of Mathematics and Computer Science, University of Bucharest
Academiei 14, RO-010014, Bucharest, Romania
rceterchi@gmail.com

Summary. We reformulate in algebraic terms the maximal parallel rewriting of symbols which occur inside membranes of a P system.

1 Introduction

We relate to the *formal* definition of a P system with symbol-objects, from Section 3.5 of Chapter 3 of the monograph [2].

We give an equivalent formulation of the symbol-object rewriting rules which occur in one membrane of a P system, reformulation which emphasizes the underlying algebraic structure of commutative monoid with a finite set of generators.

We limit ourselves to systems with only one membrane. Thus no dissolving actions are taken into consideration, and no target indications for rules.

2 An Algebraic Reformulation: Symbol Rewriting in One Membrane

Let $V = \{a_1, \cdots, a_n\}$ be an alphabet. Let us consider $(V^+, +, \lambda)$ the *commutative monoid freely generated by* V.

Denote by $\phi_V : V \to V^+$ the canonical inclusion of V in V^+, $\phi_V(a_i) = a_i$. We recall that V^+ is the unique (up to isomorphism) commutative monoid which includes V, with the following *universality property*: for any commutative monoid M which includes V via $\psi : V \to M$ there exists a unique morphism of monoids $\rho_V : V^+ \to M$ which commutes with ϕ_V and ψ, i.e., for which the equalities $\rho_V(a_i) = \psi(a_i)$ hold in M for all $1 \leq i \leq n$. ρ_V is called the *canonical morphism.*

We will use additive notation for commutative monoids. If $(M, +, 0)$ is such a monoid and $a \in M$, for a natural number n, na stands for $a + a + \cdots + a$ with n occurrences of a.

The elements of $(V^+, +, \lambda)$ will thus be written as $w = \sum_{i=1}^{n} m_i a_i$, for $m_i \in \mathbf{N}$.

Two other commutative monoids, isomorphic to $(V^+, +, \lambda)$ are given by the following:

- $(V^*/\sigma, \cdot, \lambda)$, where V^* is the (non-commutative) free monoid generated by V, σ is the equivalence relation defined by commutation of letters, $\cdot$ is the operation induced on classes by catenation, and λ is the class of the empty word.
- $(\mathbf{N}^n, +, \mathbf{0})$, where $\mathbf{N}^n$ is the n-th powerset of natural numbers, with $+$ componentwise addition.

This is related to several possible representations of multisets which we recall bellow:

- String notation: $w = a_1^{m_1} \cdots a_n^{m_n}$, and any permutation in this string stands for the same multiset.
- Function notation: w is identified with the function $M_w : V \longrightarrow \mathbf{N}$ which associates to each $a_i \in V$ its "multiplicity" in w, $m_i = M_w(a_i) = |w|_{a_i}$. A compact representation is as set of pairs $\{(a_1, m_1), \ldots, (a_n, m_n)\}$.
- Vector notation: w is identified with its Parikh vector $(m_1, \ldots, m_n) \in \mathbf{N}^n$.
- Additive notation: $w = \sum_{i=1}^n m_i a_i$.

In [2] the first three above presentations of multisets are used.

The monoid V^+ is endowed with a natural partial order relation (which is the same order as multiset inclusion) given by:

$$\sum_{i=1}^{n} m_i a_i \leq \sum_{i=1}^{n} m'_i a_i \ \text{ iff } \ m_i \leq m'_i \text{ for all } 1 \leq i \leq n.$$

In the formalism of [2], Section 3.5, *evolution rules* are triples $(\alpha_i, \beta_i, \delta_i)$, where $\alpha_i \in V^+$ stands for the left-hand side, β_i stands for the right-hand side, and is a multiset with target indications, and $\delta_i \in \{\neg\delta, \delta\}$ is a symbol indicating the dissolution or non-dissolution of the membrane. Since we have only one membrane, no target indications and no dissolution, such a rule will be reduced to a pair (α_i, β_i) with $\alpha_i, \beta_i \in V^+$.

Let $\{(\alpha_1, \beta_1), \cdots, (\alpha_n, \beta_n)\}$ be such a set of evolution rules (or symbol rewriting rules), associated to our unique membrane.

Let $\Lambda = \{\alpha_1, \cdots, \alpha_n\} \subset V^+$ be the subset composed of left-hand sides of the set of rules.

Consider $R_\Lambda = \Lambda^+$ the commutative monoid freely generated by Λ in V^+. Consider $\mu_\Lambda : R_\Lambda \longrightarrow V^+$ the canonical morphism given by $\mu(\alpha_i) = \alpha_i$ for all $i = 1, \ldots, n$. Its image $\mu(R_\Lambda) = M_\Lambda$ is the submonoid of V^+ generated by $\Lambda = \{\alpha_1, \cdots, \alpha_n\}$.

Let $w \in V^+$ be a fixed string (the "axiom" of the unique membrane).

Consider in M_Λ the above partial order and its intersection with the interval $[\lambda, w]$ w.r.t. this order. $(M_\Lambda \cap [\lambda, w], \leq_\Lambda)$ is nonempty and finite.

Let $C_{\Lambda,w}$ denote the set of the maximal elements of $(M_\Lambda \cap [\lambda, w], \leq_\Lambda)$. It is nonempty.

Let $R_{\Lambda,w} = {\mu_\Lambda}^{-1}(C_{\Lambda,w})$. It is nonempty and finite.

Definition 1. *A* one step computation *starting with axiom* w *and applying symbol evolution rules* $(\alpha_1, \beta_1), \cdots, (\alpha_n, \beta_n)$ *is:*

(i) If $R_{\Lambda,w} = \{\lambda\}$*, then nothing happens.*
(ii) If $R_{\Lambda,w} \neq \{\lambda\}$*, then an element* $m_1\alpha_1 + m_2\alpha_2 + \cdots + m_n\alpha_n \in R_{\Lambda,w}$ *is chosen at random.*
(iii) Replace w *by* z*, obtained from* w *as*

$$z = w - \sum_{i=1}^{n} m_i\alpha_i + \sum_{i=1}^{n} m_i\beta_i.$$

Lemma 1. *The one step computation defined above is the same as the one step transition in a P system with one membrane containing* w *and symbol evolution rules* $\{(\alpha_1, \beta_1), \cdots, (\alpha_n, \beta_n)\}$.

Proof. Case (i) corresponds to the situation that the rules are not applicable. The strings z obtained by cases (ii) and (iii) of the above are precisely the strings obtained in one membrane, with initial content w, by the maximal parallel application of the set of rewriting rules $(\alpha_1, \beta_1), \cdots, (\alpha_n, \beta_n)$, after one step of (nondeterministic) computation (see [2]). □

3 The Case of String Rewriting

We want to see to what extent, and how, the construction of the previous section can be extended to cover the case of string rewriting systems.

This reopens the discussion on *several types of parallel processing features* present in a membrane system. According to [2], we have three levels of parallelism; two of them are shared by the systems with symbols and those with strings, and a third one, specific to string-rewriting systems:

- Parallel processing *inside one membrane*: the rules are applied to *all* symbols or strings inside a membrane.
- Parallel processing *at the level of the system*: processing occurs simultaneously in all membranes.
- Parallel processing *of each string*: this would mean parallel rewriting of each string, and would be applicable only to string-rewriting systems.

This third type of parallelism, specific to Lindenmeyer systems, is *not* used by P systems: the rules rewrite each string in only one place.

If we want to consider this third type of parallelism, i.e. the parallel rewriting of each string, and only one string is present, in one membrane, then we have a non-commutative version of the previous construction. A good candidate for the order relation in terms of which to express maximality is the *scattered subwords* order. More precisely, for $V = \{a_1, \cdots, a_n\}$ an alphabet, consider $(V^*, \cdot, \lambda)$ the (noncommutative) free monoid generated by V. We consider the partial order relation on V^* given by the *scattered subwords* of a word:

u is a *scattered subword* of v, and we write $u \leq v$, iff there exist a decomposition of u as $u = u_1 \cdots u_n$ and strings $z_0, z_1, \cdots, z_n \in A^*$ such that v can be decomposed as $v = z_0 u_1 z_1 u_2 z_2 \cdots u_n z_n$.

For an axiom-string $w \in V^*$, and a set of string rewriting rules $(\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n)$, $\alpha_i, \beta_i \in V^*$, one can repeat the construction from the previous section, using the universality property of freely generated non-commutative monoids, and the above partial order, and define, in algebraic terms, the one-step parallel rewriting of one string.

If we want to consider the string rewriting systems which are present in the literature, and which have only the first two features of parallelism above, i.e., for which each string is rewritten in only one place, then we will have a different construction. This is a matter of ongoing research.

4 Conclusions and Open Problems

Definition 1 is given in purely algebraic terms. Some more intricate notions from Section 3.5 of [2] can be recaptured in this formalism: for instance, $(m_1, \ldots, m_n)$ is an *applicability vector* (see Definition 3.5.9 of [2]).

The formalism can be extended in several directions. First, to deal with several membranes, and next, with the features added by *communication* between them. Communication in the form of symport/antiport rules can (at a first glance) be more easily adapted to this formalism. Thus ECP systems introduced in [1] seem good candidates. Second, as outlined in Section 3, to deal with string rewriting systems. In the unifying framework of this formalism, the gap between the case of symbol rewriting and the case of string rewriting could be bridged.

The concept could also be useful for considering fuzzy extensions.

Acknowledgements. The author thanks her colleague Paul Flondor for suggesting the present approach, and for many discussions on this topic.

She also thanks the Research Group on Natural Computing, Department of Computer Science and Artificial Intelligence of the University of Sevilla, for a stimulating working environment. This was made possible by project TIN2005-09345-C04-01 of the Ministerio de Educación y Ciencia of Spain, cofinanced by FEDER funds.

References

1. M. Cavaliere: Evolution-Communication P Systems. In *Membrane Computing, International Workshop, WMC-CdeA 2002, Curtea de Argeş, Romania, August 19-23*, LNCS 2597, Springer, 2003.
2. Gh. Păun: *Membrane Computing. An Introduction.* Springer, Berlin, 2002.