Journal of Automata, Languages and Combinatorics **7** (2002) 4, 505–518

FINITE CODES OVER FREE BINOIDS [1]

KOSABURO HASHIGUCHI
Department of Information Technology, Faculty of Engineering, Okayama University
Tsushima, Okayanma, 700-0082, Japan
e-mail: hasiguti@kiso.it.okayama-u.ac.jp

TAKAHIRO KUNAI and SHUJI JIMBO
Department of Information Technology, Faculty of Engineering
Tsushima, Okayama, 700-0082, Japan
e-mail: jimbo@kiso.it.okayama-u.ac.jp

ABSTRACT

A binoid is an algebra which has two associative operations and the same identity to both operations. For any finite alphabet Σ, $\Sigma^*(\circ, \bullet)$ denotes the free binoid generated by Σ with two independent associative operations $\circ$ and $\bullet$ and the identity λ. We introduce the notion of two types of finite codes ($\circ$-codes and $\bullet$-codes) over free binoids and show that for any given finite subset X of $\Sigma^*(\circ, \bullet)$ and $\times \in \{\circ, \bullet\}$, one can decide effectively whether X is a $\times$-code or not.

Keywords: Free binoid, binoid language, finite codes, algorithm

1. Introduction

In [3], we introduce three algebraic systems, bisemigroups, bimonoids and binoids. A bisemigroup consists of a set of elements and two associative operations. A bimonid is a bisemigroup which has an identity to each associative operation. A binoid is a bimonoid which has the same identity to the two associative operations. In [3], as for the Chomsky hierarchy, we introduce five types of grammars, phrase structure B-grammars (a B-grammar means a binoid grammar), context sensitive B-grammars, context free B-grammars, right linear B-grammars and left linear B-grammars. In any free binoid X, generally we use two symbols $\circ$ and $\bullet$ for denoting two associative operators in X, and the free binoid generated by a finite alphabet Σ will be denoted by $\Sigma^*(\circ, \bullet)$. This is a binoid with two associative operations $\circ$ and $\bullet$, and is "free" in the sense that in this binoid, only laws over elements are two associative laws: for any $x, y, z \in \Sigma^*(\circ, \bullet)$ and $\times \in \{\circ, \bullet\}$, $(x \times y) \times z = x \times (y \times z)$.

We call any subset of a free binoid a binoid language (or a B-language). Sometimes any subset of a free monoid is called a monoid language. We say that an automaton

[1] Full version of a submission presented at the Third International Workshop on *Descriptional Complexity of Automata, Grammars and Related Structures* (Vienna, Austria, July 20 – 22, 2001).