Russian Mathematics (Iz. VUZ) *Izvestiya VUZ. Matematika*
Vol. 49, No. 11, pp.32–40, 2005 *UDC 519.714*

POSSIBILITIES OF PROBABILISTIC ORDERED k TIMES READING PROGRAMS AND DETERMINISTIC ONCE READING BRANCHING ONES ARE INCOMPARABLE

R.G. Mubarakzyanov

1. Introduction

A branching program is a well known computational model [1]. Let us recall the basic definitions.

A deterministic branching program (BP) P on a set of variables $X = \{x_0, \dots, x_{n-1}\}$ is a directed acyclic graph with one source node and two final nodes, or sinks. One of the latter is labeled by 0 (we call it the rejecting sink, or 0-node), the other one is labeled by 1 (we call it the accepting sink, or 1-node). Each non-final node of this graph is labeled by a certain variable from X. Two arcs outgoing from this node are labeled by 0 and 1. The computation process of a deterministic BP for fixed values of the variables from X implies the following procedure. The computation starts at the root s of the program. If the variable which labels s is equal to $a \in \{0, 1\}$ then the computation moves to a successor of s by an arc labeled by a. The further move depends on the value of the variable which labels the current node. Since the path is uniquely defined according to the values of the variables, one can associate any set of values of variables with a label of the final node which the computation leads to. This label is the value of the computed function.

A branching program is non-deterministic if it contains guessing nodes, i. e., unlabeled ones. A probabilistic branching program has nodes labeled by random variables. These nodes are called random nodes. A random variable takes values 0 and 1 with probability 1/2. A probabilistic branching program P defines a function $c_P : \{0,1\}^n \to [0,1]$, where $c_P(x)$ is the probability of reaching the accepting sink by P with the input x. We call this function the characteristic function of P. The program P computes a certain function h if $c_P(x) > 1/2$ for $h(x) = 1$, and $c_P(x) < 1/2$ for $h(x) = 0$, i. e., the probability that P results in the value of $h(x)$ greater than 1/2 for any values of variables x. If this probability is greater than $1 - \varepsilon$ for some ε, $0 \le \varepsilon < 1/2$, then we say that the computation has a bounded error (ε is the error of the computation).

We define *the complexity* of a program P as the number of its nodes. In order to study a relationship between different complexity classes, one considers restricted classes of branching programs [1]. A *k-times reading* branching program (BPk) is that where each variable is tested at most k times at each computation path. A read-once BP ($BP1$) is called *ordered* (or $OBDD$) if variables are read according to some fixed ordering. $OBDD$ are interesting because of their applications [2]. A program BPk is called a $kOBDD$ if it can be divided into k layers such that the computation process reads variables at each each layer at most once with respect to the same order.

A branching program is called *leveled* if all paths from the root to each non-final node v have the same length. This program can be divided into levels: each level contains non-final nodes equidistant from the root. *The width* of such a program is the maximum number of nodes at a level. A leveled program is called *oblivious* if all nodes of each level are labeled by the same