QUANTUM MODEL OF COMPUTATIONS:
UNDERLYING PRINCIPLES AND ACHIEVEMENTS

A. V. Anisimov and S. V. Danil’chenko

UDC 519.682

A quantum Turing machine is considered. A review of basic methodological principles and achievements in the field of quantum computations is given. Some problems of construction of correct quantum computations and their complexity are considered. The result of P. Shor concerning the solution of the problems of taking discrete logarithms in polynomial time relative to the length of numbers is considered in detail.

Keywords: computational processes, models of computations, deterministic Turing machine, probabilistic Turing machine, quantum Turing machine, distinctive features of quantum systems, functioning of quantum machines, realization and application of quantum machines, quantum cryptography.

INTRODUCTION

Simulation of the outside world on the basis of some algorithmic means or other plays an important role in many aspects of human activity. Simulation models and languages can vary within a wide range from mathematical equations to varieties of arts languages. In any case, the correctness of a model is determined by its adequacy to objective laws and the intrinsic logic of the outside world. If we restrict ourselves to the domain of formalized processing of algorithmic information, then there emerged a vital necessity of conducting a methodological research in the paradigm of the notion of a computational process [1, 2].

The modem theory of complexity is based on the classical model of computations, which involves the estimation of problems with respect to their decision algorithms based on devices that use the classical physical principles. As a rule, a Turing machine is used as a standard to introduce a classification of computational problems, and only estimates that are obtained on deterministic and probabilistic Turing machines (in what follows, DTM and PTM, respectively) are considered as actual (computable on modern hardware) ones. Problems that have polynomial time (and, respectively, spatial) estimates on PTM (this class is denoted by BPP as an abbreviation for “bounded probabilistic polynomial”) relative to the size of input data are considered as effectively decidable. Another well-known class of problems P decidable on DTM with a polynomial time estimate is a subclass of BPP and, therefore, speaking of effective decidability, we will mean the BPP class.

Thus, the notion of efficiency specifies an “available” model of computations (computational models operating on words of arbitrary dimension or real numbers with an unlimited accuracy per unit time are not “available” models for technical devices). Nevertheless, the standard PTM is not an adequate model for all physical devices since it is based on the classical model of the world, whereas the modern physical theory assumes a quantum model. However, it is interesting to know whether the use of principles of quantum physics in computing devices gives any advantages or the quantum model can be efficiently simulated on modern computers. The first suppositions that quantum devices can be more powerful in the computational aspect than classical ones appeared as long ago as in the eighties (R. Feynman, the well-known physicist, has pointed out in [3] that it is impossible to realize a general quantum physical system on PTM without an exponential increase in temporal resources).
In [4], physical principles of constructing a quantum computing device were formulated with a quantum Turing machine as its model (in what follows, a QTM). It is a quantum analogue of a probabilistic machine (PTM), i.e., it has an infinite tape (memory) with a finite alphabet specifying possible contents of tape cells and a local control device (processor) with a finite number of states, moving along the tape. Operations of the machine at a successive instant of time (clock cycle) are completely determined by the state of the control device and local contents of the tape (i.e., by the contents observed by the device at a current instant of time). In general, after obtaining input data, the quantum machine produces a random output sequence with a certain probability (of course, the sum of probabilities of all possible output sequences must be equal to unity).

One more important step in the development of the quantum computer was a generalization and systematization of available notions in [5], where the principles of construction of the universal quantum machine are downloaded, and also some results are obtained, from which the advantage of the quantum machine over the classical one is easily seen [5, 6].

Finally, the best known result is that of P. Shor [7], who has shown the decidability of the problem of taking the discrete logarithms of numbers in multiplicative integral groups (mod p) and the problem of decomposition of an integer into multipliers in polynomial time (relative to the length of the representation of a number) on quantum machines. Although the NP-completeness of these problems is not proved, their complexity is characterized by the absence of an effective algorithm for solving them, which is the basis of many modern cryptoalgorithms.

At present, there exists no physical realization of a quantum machine, but such a realization is theoretically possible. An important theoretical step in this direction is the construction of efficiently interpreting (i.e., with at the most polynomial decrease in time and spatial estimates) universal quantum machines [5, 8]. Practical works on the creation of quantum-mechanical computing devices are carried out within the framework of quantum cryptography.

This article is a review of fundamental present-day results available in the field of the theory of quantum computations, namely, basic concepts are introduced, primary theorems concerning structures of quantum machines are given, the rules of their construction and research are justified, and one of results of P. Shor is presented [7].

PRINCIPLES OF CONSTRUCTION OF A QUANTUM MODEL OF COMPUTATIONS

We will first describe a standard representation of a centralized controlled computational process (on a physical or abstract device). Let us consider the following two basic components of this process:

— an information environment is a certain domain used as a storage of data (in Turing machines, it is an infinite tape, and an elementary information unit is a symbol from a finite alphabet; in automata, it is a stack; in modern computers, it is a matrix of on-line storage whose information unit is the bit (assuming only two values, namely, 0 or 1), etc.). In different computational models, information environments differ from each other by their physical structures, access methods, and methods of information storage but, as a whole, all of them are characterized by a set of data collections that can be specified by them;

— a control device (processor) is a certain mechanism that realizes a process of computations by transformation of the information environment (it is the control head in Turing machines and the central processor together with control peripheral devices in modern computers).

For the majority of computational processes running on physical devices, it is usual to distinguish between the following characteristic properties:

*discreteness* means that any computation process is not continuous in time and can be divided into sequential transforming steps, between which the information environment and the state of the control device of the process remain invariable (these steps can be called clock cycles). States of the information environment and control device between these steps are usually called the configuration of the computational process;

*localness* means that, for one clock cycle, the control device can change only a fixed part (whose size is restricted by a constant) of its information environment (for example, the processors of modern computers cannot change the contents of an arbitrary number of storage bits);

*boundedness of control commands* means that the number of different transforming operations executed by the control device is always bounded by a constant.

Physical computing devices also have the property of determinacy, which means that, at each step of functioning, the configuration of the process before executing this step uniquely specify the configuration of the process after its execution.