Solving Approximation Problems by Ant Colony Programming

Mariusz Boryczka
University of Silesia, Sosnowiec, Poland
e-mail: boryczka@us.edu.pl

Zbigniew J. Czech
Silesia University of Technology, Gliwice, and
University of Silesia, Sosnowiec, Poland
e-mail: zjc@polsl.gliwice.pl

Abstract

A method of automatic programming, called genetic programming, assumes that the desired program is found by using a genetic algorithm. We propose an idea of ant colony programming in which instead of a genetic algorithm an ant colony algorithm is applied to search for the program. The test results demonstrate that the proposed idea can be used with success to solve the approximation problems.

1 INTRODUCTION

Given a problem one usually builds an appropriate computer program to solve the problem. Automatic programming makes possible to avoid a tedious task of creating such a program. In automatic programming the program is obtained by specifying first the goals which are to be realized by the program. Then, based on this specification, the program is constructed automatically. A method of automatic programming, called genetic programming, was proposed by Koza (Genetic programming: On the programming of computers by natural selection, MIT Press, Cambridge, MA, 1992). In genetic programming a desired program is found by using a genetic algorithm.

2 ANT COLONY PROGRAMMING

This work introduces an idea of ant colony programming in which instead of a genetic algorithm, an ant colony algorithm is applied to search for the program. We consider approximation problems which consist in a choice of an optimum function from some class of functions. Such a function should approximate in a best way another, known function, or some values of an unknown function specified in a finite number of points from its domain. Approximation problems are encountered in analysis of numerical data, modeling physical phenomena, analysis of statistical observations etc.

While solving an approximation problem by ant colony programming we use two approaches. In the first, expression approach, we search for an approximating function in the form of an arithmetic expression represented in the prefix notation. The artificial ants (agents) build the expression as a tree consisted of terminal symbols and functions, communicating with each other through the pheromone trails. In the second, program approach, the desired approximating function is built as a computer program, i.e. a sequence of assignment instructions which evaluates the function. The process of program generation consists in expanding the program by consecutive instructions taken from some predefined set.

To date ant colony programming has not been applied to automatic programming.

3 CONCLUSIONS

The test results demonstrate that the ant colony programming approaches are effective, especially the program approach. There are still some issues which remain to be solved. The most important is the issue of constants which regards the choice of constants which are to be enclosed in the set of terminal symbols. These constants are crucial for the work of the ant colony algorithm. The future work on ant colony programming includes an extension of the solutions to multivariate approximation problems.

Acknowledgments

The work presented in this paper is carried out under the State Committee for Scientific Research (KBN) grant no 7 T11C 021 21.