

Chapter 2

Phylogenetic Differential Evolution

Vinicius Veloso de Melo
University of São Paulo, Brazil

Danilo Vasconcellos Vargas
University of São Paulo, Brazil

Marcio Kassouf Crocorno
University of São Paulo, Brazil

ABSTRACT

This paper presents a new technique for optimizing binary problems with building blocks. The authors have developed a different approach to existing Estimation of Distribution Algorithms (EDAs). Our technique, called Phylogenetic Differential Evolution (PhyDE), combines the Phylogenetic Algorithm and the Differential Evolution Algorithm. The first one is employed to identify the building blocks and to generate metavariables. The second one is used to find the best instance of each metavariable. In contrast to existing EDAs that identify the related variables at each iteration, the presented technique finds the related variables only once at the beginning of the algorithm, and not through the generations. This paper shows that the proposed technique is more efficient than the well known EDA called Extended Compact Genetic Algorithm (ECGA), especially for large-scale systems which are commonly found in real world problems.

1. INTRODUCTION

The resolution of several black-box optimization problems has been possible due to the development of metaheuristics (Larrañaga & Lozano, 2001; Goldberg, 2002; Chakraborty, 2008). Also,

engineering design problems have been solved by those same algorithms (Ruela, Cabral, & Aquino, 2010; Taroco, Carrano, & Neto, 2010). However, when the problem has strong correlated variables - as the deceptive problems (Goldberg, 2002) - classical metaheuristics may not have a high

DOI: 10.4018/978-1-4666-4253-9.ch002

probability of success. The Evolutionary Compact Genetic Algorithm (ECGA) (Harik, Lobo, & Sastry, 2006) - based on the Compact Genetic Algorithm (Harik, Lobo, & Goldberg, 1999) - is an Estimation of Distribution Algorithm (EDA) (Larrañaga & Lozano, 2001) that identifies the correlated variables, called Building Blocks (BBs), and uses a recombination operator to combine the best found partial solutions of each BB in order to generate a new full solution of the problem.

This paper proposes the use of the Phylogenetic Differential Evolution (PhyDE) to identify the BBs of a deceptive problem. This technique employs a clustering algorithm named Neighbor Joining (Saitou & Nei, 1987), commonly used in Biology to create Phylogenetic Trees (Felsenstein, 2003).

While most EDAs, including ECGA, aim to find the BBs in each generation, Phylogenetic Algorithms (Vargas, Delbem, & de Melo 2010) first applies a process to identify the BBs of a problem, and soon after the BBs have been known, a local search algorithm, usually an exhaustive search, is applied to each BB in order to find its optimum configuration.

However, an exhaustive search must be avoided when a BB is composed of several bits, due to the exponential complexity of the problem. Thus, a better alternative to locate the optimum configuration must be used. The idea is to replace the exhaustive search with metaheuristics, and more than that, to employ an efficient metaheuristic incapable of solving large deceptive problems. To achieve this goal, a type conversion from binary to integer is required. Since the Differential Evolution (DE) algorithm presents one of the best results in the resolution of several problems (García, Molina, Lozano, & Herrera, 2009), it was chosen as the optimization algorithm to be used in this project.

Hard problems solved by EDAs are usually characterized by the number of variables, number of BBs that compose the problem, and their size. While the problem size is appropriately treated by existing algorithms, the size of the BBs is not. This is the main contribution of the proposed PhyDE. EDAs solve these problems using large number of

evaluations (Harik, Lobo, & Sastry, 2006; Pelikan, Goldberg, & Cantú-Paz, 1999). By contrast, the proposed method is capable of solving problems with larger BBs in as few as half the number of evaluations. In general, results of EDAs are presented for BBs of size four. The experiments presented in this paper show a comparison using BBs composed of eight variables.

This paper is organized as followed: the next section introduces the concept of BBs and deceptive functions which are commonly used to test the performance of different EDAs. After that, we briefly present the way other algorithms find BBs, and then, we present methods of phylogenetic reconstruction used to identify BBs. Following this, two sections introduce the Phylogenetic Algorithms and the proposed PhyDE algorithm, which is composed of three sections detailing its steps: the construction of the Distance Matrix, the Neighbor Joining method and the construction of metavariables, and the Differential Evolution algorithm, used by the PhyDE to locate the optimum configuration for the identified BBs. Finally, the last section contains the explanation of the conducted experiments, and the results that show its efficiency. The results have shown that the technique proposed in this paper is approximately twice more efficient than the known ECGA.

2. BUILDING BLOCKS AND DECEPTIVE FUNCTIONS

It is usual to evaluate the size of its search space to determine the difficulty of an optimization problem. However, two other factors (Goldberg, 2002) that are also important in determining difficulty are the number of Building Blocks (BBs) and the number of variables in each BB.

Consider the deceptive problem described by Equation 1. This problem accepts an 8-bit string as input and calculates the fitness based on the number of bits '1'. To illustrate a deceptive function, consider the input 00001111. This string has 4 bits marked as '1'. Applying $u = 4$ to the equa-

17 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/phylogenetic-differential-evolution/80055

Related Content

Analytics on Fireworks Algorithm Solving Problems with Shifts in the Decision Space and Objective Space

Shi Cheng, Quande Qin, Junfeng Chen, Yuhui Shi and Qingyu Zhang (2017). *Nature-Inspired Computing: Concepts, Methodologies, Tools, and Applications* (pp. 1008-1038).

www.irma-international.org/chapter/analytics-on-fireworks-algorithm-solving-problems-with-shifts-in-the-decision-space-and-objective-space/161060

Swarm Intelligence-Empowered Bug Prediction Strategy for Decision Support in Software Defect Prediction

Medhunhashini D. R. and Jeen Marseline K. S. (2024). *Intelligent Decision Making Through Bio-Inspired Optimization* (pp. 18-28).

www.irma-international.org/chapter/swarm-intelligence-empowered-bug-prediction-strategy-for-decision-support-in-software-defect-prediction/344560

Artificial Life Optimization Algorithm and Applications

Bo-Suk Yang (2009). *Handbook of Research on Artificial Immune Systems and Natural Computing: Applying Complex Adaptive Technologies* (pp. 423-449).

www.irma-international.org/chapter/artificial-life-optimization-algorithm-applications/19656

Combining Forecasts: A Genetic Programming Approach

Adriano S. Koshiyama, Tatiana Escovedo, Douglas M. Dias, Marley M. B. R. Vellasco and Marco A. C. Pacheco (2012). *International Journal of Natural Computing Research* (pp. 41-58).

www.irma-international.org/article/combining-forecasts-genetic-programming-approach/76376

Evolutionary Algorithms

P. Collet (2007). *Handbook of Research on Nature-Inspired Computing for Economics and Management* (pp. 45-58).

www.irma-international.org/chapter/evolutionary-algorithms/21119