A Binary Cuckoo Search Algorithm for Graph Coloring Problem

Halima Djelloul, MISC Laboratory, Constantine 2 University, Constantine, Algeria
Abdesslem Layeb, MISC Laboratory, Constantine 2 University, Constantine, Algeria
Salim Chikhi, MISC Laboratory, Constantine 2 University, Constantine, Algeria

ABSTRACT

The Graph Coloring Problem (GCP) is one of the most interesting, studied, and difficult combinatorial optimization problems. That is why several approaches were developed for solving this problem, including exact approaches, heuristic approaches, metaheuristics, and hybrid approaches. This paper tries to solve the graph coloring problem using a discrete binary version of cuckoo search algorithm. To show the feasibility and the effectiveness of the algorithm, it has used the standard DIMACS benchmark, and the obtained results are very encouraging.

Keywords: Binary Cuckoo Search, Cuckoo, Evolutionary Computation, Graph Coloring Problem, Heuristics, Optimization, Search Algorithm

1. INTRODUCTION

The Graph Coloring Problem (GCP) is one of the most interesting, studied and difficult combinatorial optimization problems. The GCP consists in coloring each vertex of a given graph by using a minimum number of colors called the chromatic number (Matula et al., 1972), so that no two adjacent vertices are colored with the same color. Unfortunately, GCP has been shown to be NP-hard (Garey & Jonhson, 1979), hence, several approaches were developed to handle this problem, that we can classify into three classes: exact approaches, heuristic approaches and metaheuristics approaches. There exist quite a few of exact approaches for this problem; they are generally based on the implicit enumeration algorithms (Kubale & Jackowski, 1985), as well as branch- and- bound and its variants (Méndez-Díaz & Zabala, 2006; Mehrotra & Trick, 1996). On the other hand, the constructive approaches have been widely proposed to resolve the graph coloring problem. The constructive approaches progressively build the solution, in this category we can mention the algorithm developed by Welsh and Powell (Welsh & Powell, 1967), the degree of saturation (DSATUR) (Brélaz, 1979) and the recursive largest first algorithm (RLF) (Leighton, 1979). Moreover, many kinds of metaheuristics and
their hybridizations have been used to solve GCP like tabu Search (Hertz & de werra, 1987), Simulated annealing (Johnson et al., 1991), genetic algorithm (Fleurent & Ferland, 1996), ant colony (Dowsland & Thompson, 2008), variable neighborhood search VNS (Avanthay et al., 2003), Variable Search Space (VSP) that is an extension of the VNS (Hertz et al., 2008), a memetic algorithm (Zhipeng & Jin-Kao, 2010), a hybrid Artificial Bee Colony Algorithm for Graph 3-Coloring (Fister, I., Jr., Fister, I., & Brest, 2012), etc.

Far from the graph coloring problem, Cuckoo Search (CS) is an optimization algorithm developed by Xin-She Yang and Suash Deb (Yang & Deb, 2010). It was inspired by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of other host birds (of other species). Some bird’s host can involve direct conflicts with the invading cuckoos. For example, if a bird’s host discovers that the eggs are strange eggs, it will either throw these alien eggs away or simply abandon its nest and build a new nest elsewhere (Barthelemy et al., 2008).

The cuckoo’s behavior and the mechanism of Lévy flights (Payne et al., 2005; Pavlyukevich, 2007) have led to the design of an efficient inspired algorithm performing optimization search. The recent applications of Cuckoo Search for optimization problems have shown its promising effectiveness. Moreover, a promising Binary Cuckoo Search algorithm (BCS) is recently proposed to deal with discrete problems (Gherboudj et al., 2012) which used a sigmoid function similar to that used in the binary particle swarm optimization algorithm to obtain a binary solution.

The present study was designed to investigate the use of the BCS algorithm to deal with the graph coloring problem. The main features of the proposed approach consist in adopting a binary representation of the search space and using a sigmoid function and probability model in order to generate binary values (Gherboudj et al., 2012). Moreover, we have modified the BSR algorithm proposed in (Gherboudj et al., 2012) to avoid the infeasible solution like having an uncolored node. Finally, we have tested our algorithm on some DIMACS instances taken from (http://mat.gsia.cmu.edu/ COLOR/instances.html) and the results found are promising.

The reminder of the paper is organized as follows. In section 2, a formulation of the tackled problem is given. In section 3, an overview of cuckoo search is presented. In section 4, the proposed method is described in section 5. Experimental results are discussed in section 6. Finally, conclusions and future work are drawn.

2. PROBLEM FORMULATION

Graph Coloring Problem (GCP) is a well-known combinatorial problem, and important task in solving many real problems such as the frequency assignment problem (Smith et al., 1998), crew scheduling (Gamache et al., 2007), register allocation (de Werra et al., 1999), etc. graph is k-colorable if and only if it can be colored using k colors. Formally, a k-coloring will be represented by a set $H = \{C(v_1), C(v_2), \ldots, C(v_n)\}$ such as $C(v)$ is the color assigned to the vertex $v$. If for all $\{u, v\} \in E, C(u) \neq C(v)$, then $H$ is a legal coloring; otherwise, $H$ is an unfeasible k-coloring. In the optimization version of the GCP, the principal objective is to minimize the total number of colors used to color a given graph.

Formally, the graph coloring problem can be formulated as follows:

Given a k-coloring $H = \{C(v_1), C(v_2), \ldots, C(v_n)\}$ with the set $V = \{v_1, \ldots, v_n\}$ of vertices, the evaluation function $f$ counts the number of conflicting vertices produced by $H$ such that:

$$f(H) = \sum_{\{u, v\} \in E} \delta_{uv}$$

Where:

$$\delta_{uv} = \begin{cases} 1, & \text{if } C(u) = C(v) \\ 0, & \text{otherwise.} \end{cases}$$

By consequent, a coloring $H$ with $f(H) = 0$ corresponds to a feasible k-coloring.

Copyright © 2014, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.
Related Content

Web Distributed Computing Systems: Implementation and Modeling
[www.igi-global.com/chapter/web-distributed-computing-systems/63581?camid=4v1a](www.igi-global.com/chapter/web-distributed-computing-systems/63581?camid=4v1a)

Dynamic Genetic Algorithms with Hyper-Mutation Schemes for Dynamic Shortest Path Routing Problem in Mobile Ad Hoc Networks
[www.igi-global.com/article/dynamic-genetic-algorithms-hyper-mutation/62836?camid=4v1a](www.igi-global.com/article/dynamic-genetic-algorithms-hyper-mutation/62836?camid=4v1a)

Reflections of Spiral Complexity on Art
[www.igi-global.com/chapter/reflections-spiral-complexity-art/28385?camid=4v1a](www.igi-global.com/chapter/reflections-spiral-complexity-art/28385?camid=4v1a)
Three Novel Methods to Predict Traffic Time Series in Reconstructed State Spaces
[www.igi-global.com/article/three-novel-methods-predict-traffic/40902?camid=4v1a](www.igi-global.com/article/three-novel-methods-predict-traffic/40902?camid=4v1a)