Comparative Study on Discrete SI Approaches to the Graph Coloring Problem

Claus Aranha
University of Tsukuba, Faculty of Engineering, Information and Systems
caranha@cs.tsukuba.ac.jp

Jair Pereira Junior
Federal University of ABC
jair.pereira@aluno.ufabc.edu.br

Hitoshi Kanoh
University of Tsukuba, Faculty of Engineering, Information and Systems
kanoh@cs.tsukuba.ac.jp

ABSTRACT

The Graph Coloring Problem is an important benchmark problem for decision and discrete optimization problems. In this work, we perform a comparative experimental study of four algorithms based on Swarm Intelligence for the 3-Graph Coloring Problem: Particle Swarm Optimization (PSO), Artificial Bee Colonies (ABC), Cuckoo Search (CS) and FireFly Algorithm (FFA). For each algorithm, we test parameter settings published in the literature, as well as parameters found by an automated tuning methodology (irace). This comparison may shed some light at the strengths and weaknesses of each algorithm, as well as their dependence on parameter values.

CCS CONCEPTS

• **Computing methodologies → Genetic algorithms**;

KEYWORDS

Swarm Intelligence, Graph Coloring Problem, Parameter Tuning

ACM Reference Format:

Claus Aranha, Jair Pereira Junior, and Hitoshi Kanoh. 2018. Comparative Study on Discrete SI Approaches to the Graph Coloring Problem. In *GECCO '18 Companion: Genetic and Evolutionary Computation Conference Companion, July 15–19, 2018, Kyoto, Japan.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3205651.3205664

1 MOTIVATION

We present an experimental comparative study of discrete Swarm Intelligence (SI) approaches to the Graph Coloring Problem (GCP). The GCP consists of assigning a set of colors (labels) to the vertices of a graph, so that no two adjacent vertices share the same color. Algorithms that can solve the GCP can also be applied to real world assignment problems, such as memory registers to variables, hospital staff to shifts, and classrooms to school courses.

Many meta-heuristic search algorithms based on the concepts of Swarm Intelligence have been proposed to solve the GCP. From more recent works, we highlight Fister et al. proposing a hybrid algorithm that used Firefly algorithm to calculate the initial weights of a combinatorial solver [4], Aoki et al. [1], who described a Particle Swarm Optimization using hamming distance to calculate the swarm speed in the discrete space [1], Toda and Aranha, who proposed a discrete version of the Levy mutation operator to implement a Cuckoo Search strategy [2], and Kui et al., who proposed a discrete version of the Artificial Bee Colony algorithm [3].

2 EXPERIMENT

We implement the four algorithms above from scratch using the R language, following the descriptions on the respective papers. The implementation of each algorithm, as well as the scripts and data needed to replicate the experiments presented here as well as a few others, are available at: https://github.com/caranha/EvoGCP.

For each algorithm, we use *irace* [5] to fine tune the parameters. Parameter tuning is performed on 10 sets of 50 graphs with 90 nodes and 2.5 edge density. Figure 2 shows the progress of the tuning. Table 1 shows the tuned and literature parameters.

Using the parameters from this table, we applied the four methods to 12 sets of 3-GCP problems. Each set has 100 graphs with 90, 120, 150 and 180 nodes, and 2, 2.5 and 3 edge density. Each algorithm is allowed 2×10^7 function evaluations to solve each graph in the problem set. Figure 1 shows the results for the graph sets with 180 nodes.

Fine tuning the parameters clearly improved the results for D-CS. For HD-PSO the fine-tuned parameters largely agreed with the parameters suggested by the literature. On the other hand, parameter tuning worsened the results for D-ABC. This may be related to the fact that ABC literature usually keeps the same values for Employed and Onlooker bees, while we allowed the fine-tuning script to use different values, largely increasing the parameter search space. For M-FFA, the local search operator used the vast majority of the computational effort, sometimes never entering the FFA main loop.

REFERENCES

[1] Takuya Aoki, Claus Aranha, and Hitoshi Kanoh. 2015. PSO Algorithm with Transition Probability Based on Hamming Distance for Graph Coloring Problem. In *Proceedings of the 2015 IEEE International Conference on Systems, Man and Cybernetics*. IEEE.

[2] Claus Aranha, Keita Toda, and Hitoshi Kanoh. 2017. Solving the Graph Coloring Problem using Cuckoo Search. In *Advances in Swarm Intelligence*. Fukuoka, Japan, 552–560.

[3] Kui Chen and Hitoshi Kanoh. 2016. A Discrete Artificial Bee Colony Algorithm Based on Similarity for Graph Coloring Problems. In *Theory and Practice of Natural Computing*, Carlos Martín-Vide, Takaaki Mizuki, and Miguel A Vega-Rodríguez (Eds.). Springer International Publishing, Cham, 73–84.

[4] Iztok Fister Jr., Iztok Fister, Janez Brest, and Xin-She Yang. 2012. Memetic firefly algorithm for combinatorial optimization. In *Bioinspired optimization methods and their applications : proceedings of the Fifth International Conference on Bioinspired Optimization Methods and their Applications - BIOMA 2012*. 75–86.

[5] Manuel López-Ibáñez, Jérémie Dubois-Lacoste, Leslie Pérez Cáceres, Mauro Birattari, and Thomas Stützle. 2016. The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives* 3 (2016), 43–58.

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07.
https://doi.org/10.1145/3205651.3205664

Table 1: Results of the parameter tuning

Method	Parameters
HD-PSO - literature	Pop. size = 10, w = 0.05, c1 = 7, c2 = 0.03
HD-PSO - tuning	Pop. size = 10, w = 0.0596, c1 = 5.209, c2 = 0.105
D-ABC - literature	Employed = 200, Onlooker = 200, Scout = 2, c = 2, limit = 90
D-ABC - tuning	Employed = 100, Onlooker = 454, Scout = 20, c = 12, limit = 13
D-CS - literature	Pop. size = 10, p_a = 0.0001, parasitism policy = "levy", α = 1, β = 1.5, compare = F
D-CS - tuning	Pop. size = 5, p_a = 0.0035, parasitism policy = "fixed", E = 2, α = 0.381, β = 1.4799, compare = F
M-FFA - literature	Pop. size = 500, lb = 0, ub = 1, α = 0.1, β = 0.1, γ = 0.8
M-FFA - tuning	Implementation turned out to spend almost all effort on local search. Tuning not used

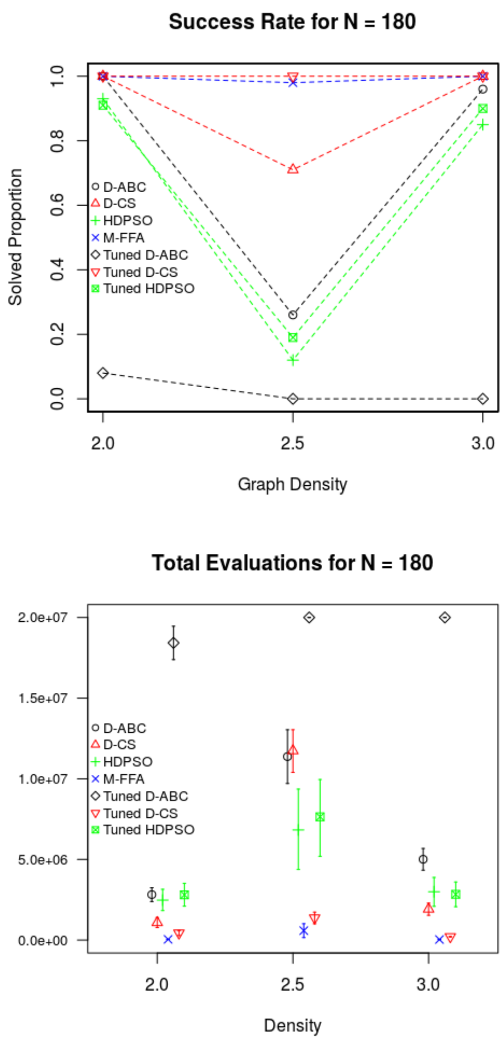

Figure 1: Results for the N=180 graph set. Top image shows the proportion of solved graphs, while the bottom image shows the mean number of function evaluations necessary, with a 95% confidence bars.

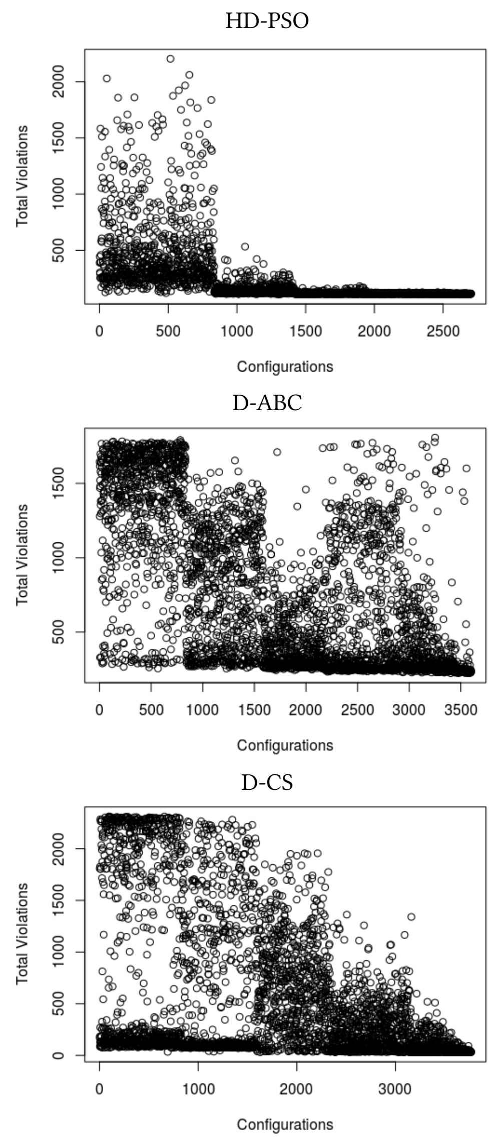

Figure 2: Details of the parameter tuning process. Each dot represent one parameter configuration tested, and lower vertical position represent better results.