Expert Systems With Applications 63 (2016) 344–374

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An agent based approach for the implementation of cooperative proactive S-Metaheuristics

Mailyn Moreno[a], Alejandro Rosete[a], Juan Pavón[b,*]

[a] Facultad de Ingeniería Informática, Instituto Superior Politécnico "José Antonio Echeverría" (CUJAE), Marianao 19390, La Habana, Cuba
[b] Dep. Ingeniería del Software e Inteligencia Artificial, Universidad Complutense de Madrid, Spain

ARTICLE INFO

Article history:
Received 2 November 2013
Revised 7 July 2016
Accepted 8 July 2016
Available online 9 July 2016

Keywords:
Metaheuristics
Agents
Proactivity
Cooperation
Local search
Threshold accepting
Great deluge algorithm
Record-to-record travel
Fitness distance correlation

ABSTRACT

This paper introduces several cooperative proactive S-Metaheuristics, i.e. single-solution based metaheuristics, which are implemented taking advantage of two singular characteristics of the agent paradigm: proactivity and cooperation. Proactivity is applied to improve traditional versions of Threshold Accepting and Great Deluge Algorithm metaheuristics. This approach follows previous work for the definition of proactive versions of the Record-to-Record Travel and Local Search metaheuristics. Proactive metaheuristics are implemented as agents that cooperate in the environment of the optimization process with the goal of avoiding stagnation in local optima by adjusting their parameters. Based on the environmental information about previous solutions, the proactive adjustment of the parameters is focused on keeping a minimal level of acceptance for the new solutions. In addition, simple forms of cooperation by competition are used to develop cooperative metaheuristics based on the combination of the four proactive metaheuristics. The proposed metaheuristics have been validated through experimentation with 28 benchmark functions on binary strings, and several instances of knapsack problems and travelling salesman problems.

1. Introduction

Metaheuristics are optimization methods for finding good solutions (not necessarily optimal) to complex optimization problems in different domains (Talbi, 2009; Torres-Jiménez & Pavón, 2014). Examples of applications of metaheuristics vary across many fields, such as health care (Logeswari & Karnan, 2010), genome sequencing (Luque & Alba, 2005), manufacturing (Ghosha, Senguptaa, Chattopadhyayb, & Dana, 2011), transportation problems (Baptista-Pereira & Tavares, 2009; Brito, Martínez, Moreno, & Verdegay, 2009; Pillac, Gendreau, Guéret, & Medaglia, 2013; Tas, Dellaert, Woensel, & Kok, 2013), energy (Kallrath, Pardalos, Rebennack, & Scheidt, 2009; Yan et al., 2010), face recognition (Chand, 2010), timetabling (Özcan, Mısır, Ochoa, & Burke, 2010), video compression (Luis, Molina, & Patricio, 2011), service systems (Cheng, Lai, Yang, & Zhu, 2016), and civil engineering (Hamm, Beißert, & König, 2009). Even in computer science and software engineering, metaheuristics have been extensively applied, for example in software design (Aleti & Moser, 2015; Räihä, 2010; Räihä, Mäkinen, & Poranen, 2009), data analysis and processing (Marwala, 2009; Yan, Zhang, & Zhang, 2009), communication networks (Ebrahimi, ShafieiBavani, Wong, Fong, & Fiaidhi, 2015; Yang, Cheng, & Wang, 2010), software testing (Arcuri, 2011; Arcuri & Yao, 2008), quality prediction (Azar & Vybihal, 2011), and requirements engineering (Tonella, Susi, & Palma, 2013; Zhang, Harman, & Lim, 2013). The aim of the previous list is not to survey all the possible applications of the metaheuristics, but only to give an overall view of the wide range of relevant applications of these modern optimization techniques.

The list of available metaheuristics is also large, including diverse techniques such as genetic algorithms, ant colonies, simulated annealing, memetic algorithms, great deluge algorithm, and variable neighborhood search (Talbi, 2009). It is worth noting that all metaheuristics have a similar overall performance when considering the average across all possible problems according to the No Free Lunch theorem (Wolpert & Macready, 1997). Indeed, the characterization of the problems and the identification of the characteristics that make each problem harder or easier to address with specific metaheuristics is currently a relevant issue (Degroote & Causmaecker, 2015; Smith-Miles & Lopes, 2012; Smith-Miles, Baatar, Wreford, & Lewis, 2014). In consequence, the development of new kind of metaheuristics and the identification of problems characteristics that affect their performance is an open field of research (Aleti & Moser, 2015; Muñoz, Sun, Kirley, & Halgamuge, 2015).

* Corresponding author.
E-mail addresses: my@ceis.cujae.edu.cu (M. Moreno), rosete@ceis.cujae.edu.cu (A. Rosete), jpavon@fdi.ucm.es (J. Pavón).

http://dx.doi.org/10.1016/j.eswa.2016.07.013
0957-4174/

The inspiration for defining new kind of metaheuristics may come from different fields. Recently, the agent paradigm is being applied to design new metaheuristics, for example Aydin (2012), González, Cruz, Amo, and Pelta (2011), Abraham, Grosan, and Ramos (2006), Barbucha (2012), Barbucha, Czarnowski, Jędrzejowicz, Ratajczak-Ropel, and Wierzbowska (2013), Ebrahimi et al. (2015), Lepagnot, Nakib, Oulhadj, and Siarry (2010), Li, Yu, Shen, and Miao (2009), Malek (2009), and Turek et al. (2016), which are mainly based in the social ability of the agents to interchange information to guide the metaheuristics. Proactivity is a singular characteristic of agents that, however, has received less attention for the design of metaheuristics (Beck & Wilson, 2007; Moreno, Rosete, & Pavón, 2013; Reyes-Badillo, Ruiz, Cotta, & Fernández-Leiva, 2013; Wang, Liu, Wang, & Jin, 2015), and only recently has been applied in two metaheuristics: Record-to-Record Travel and Local Search (Moreno et al., 2013). The idea was to implement a proactive adjustment of the parameters and operators to avoid stagnation in local optima, by driving the process by a goal model.

Following the idea presented in Moreno et al. (2013), a *proactive metaheuristic* can be defined as a metaheuristic that adjusts its performance in a proactive way, in order to satisfy the user goals. The focus was on S-Metaheuristics with the goal of avoiding local optima. To satisfy this goal, a proactive metaheuristic configures the parameters that control the performance of the metaheuristic with values that are supposed to be good for the next stages of the optimization, taking into account information from the current stage. The advantage of the approach presented in Moreno et al. (2013) for Record-to-Record Travel and Local Search is the possibility to delegate to the metaheuristic the adjustment of the parameter Deviation and the selection of the mutation operator, thus increasing the autonomy of the metaheuristic in order to satisfy the goal for the human optimizer of avoiding local optima. Consequently, each metaheuristic can be considered as an agent that operates in the search space with the goal of finding the best solution to the optimization problem.

This paper extends these previous works on the use of the agent paradigm for defining new metaheuristics that exploit the proactiveness characteristic of agents. As a result, new proactive metaheuristics are defined as versions of two single-solution based metaheuristics: Threshold Accepting (TA) and Great Deluge Algorithms (GDA). In addition, these proactive metaheuristics are also combined with the proactive metaheuristics presented in Moreno et al. (2013) to obtain new variants of *cooperative proactive metaheuristics*. The cooperation between the metaheuristic agents is based on a simple form of competition where the metaheuristic agent with the best performance is allowed to consume more evaluations of the fitness function. In addition, this paper illustrates how these cooperative and proactive metaheuristics can be designed by using an agent-based methodology, i* (Yu, 2009). This methodology has been chosen as it strongly relies on the concept of *goal*, which is relevant for the definition of proactive behaviors. The use of a methodology allows for a more systematic and robust development of the metaheuristics.

Section 2 explains the main concepts of agents and metaheuristics that are relevant to this paper. Section 3 presents the analysis and design of the system model with the i∗ methodology. This section explains that the soft goal of avoiding local optima can be delegated to the metaheuristic by defining plans for adjusting the parameters and operators, in order to satisfy the goal of the optimizer. Section 4 presents an experimental validation of the proactive metaheuristics with 28 benchmark functions on 100-bits strings. Section 5 extends the experimental study to two known NP-Complete Problems: Knapsack problems and Travelling Salesman Problems. The results are analyzed using known non-parametric statistical tests, showing that the good performance of the proactive metaheuristics. Also, some arguments are included for the characterization of the problems where each metaheuristic get better or worse performance. Section 6 presents the conclusions and discusses possible extensions to this approach.

2. S-Metaheuristics and agents

2.1. S-Metaheuristics performance

In their process of optimization, metaheuristics get solutions, which can be the base to look for better solutions. When they use the current (single) solution as a reference for generating potential new solutions, they are called S-Metaheuristics, i.e., single-solution based metaheuristics (Talbi, 2009). The performance of metaheuristics depends highly on the balance between two factors: exploration of the search space (*diversification*) and exploitation of the best solutions found (*intensification*) (Talbi, 2009).

One extreme of diversification is Random Search (RS), because every new solution is generated randomly, taking any potential solution in the search space, without any considerations of the previously generated solutions. On the other end, Local Search (LS) generates new solutions as modifications of the best previous solutions, thus limiting the search space by using certain operators. Low diversification has the risk of converging to local (not global) optima, where the optimization is stagnated.

In order to overcome this issue, many S-Metaheuristics relax the acceptance criterion, and can consider some worse solutions as new references. For example, in Random Walk (RW) every new solution (worse or better) is accepted as reference. RW introduces a trivial relaxation of the acceptance criterion that avoids stagnation in local optima. However, RW is imbalanced to the exploration, because the quality of the solutions is not taken into account to be used as references. Other S-Metaheuristics, such as Record-to-Record Travel (RRT), Threshold Accepting (TA) and Great Deluge Algorithms (GDA) use a moderated acceptance criterion. They accept some worse solutions taking into account the quality of the new solution, and some other aspects and parameters. For instance, RRT accepts worse solutions which are not much worse than the best solution in a certain parameter (Deviation). TA accepts worse solutions which are not much worse than the current solution in a certain parameter (Threshold). GDA accepts worse solutions which are above a certain parameter (Water level) which is systematically updated based on another parameter (Rain). Each parameter directly affects the performance of each algorithm, because it controls the balance between exploitation and exploration. For example, RRT with a very high value of deviation is similar to RW. The same can be said about TA with a very high value of Threshold and GDA with very low value of water level. It is important to note that these three metaheuristics (TA, RRT, and GDA) share a common approach to avoid stagnation by accepting worse solution in a deterministic way. Indeed, Talbi (2009) classified them as similar versions of simulated annealing.

An alternative to the modification of the acceptance criterion, in order to avoid local optima is to modify the neighborhood. This approach is used by Variable Neighborhood Search (VNS). It is important to note that local optima are also consequences of the operators (Jones & Forrest, 1995). As the neighborhood changes, a solution that is a local optimum in a neighborhood is not necessarily a local optimum in other neighborhood. The underlying idea is that the best solution at the end of the search may be a global optimum because it has been a local optimum in many neighborhood structures. The operators and the criteria to change them affect the performance of VNS.

Unless some general guidelines are available to adjust all these S-Metaheuristics (Birattari, 2009; Talbi, 2009), the best values