Applied Soft Computing 38 (2016) 257–268

Contents lists available at ScienceDirect

Applied Soft Computing

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

Simulating non-stationary operators in search algorithms

Adrien Goëffon, Frédéric Lardeux*, Frédéric Saubion

LERIA, University of Angers, France

ARTICLE INFO

Article history:
Received 11 April 2014
Received in revised form 1 September 2015
Accepted 17 September 2015
Available online 8 October 2015

Keywords:
Adaptive operator selection
Operator-based algorithms
Multi-armed bandit
Island models

ABSTRACT

In this paper, we propose new scenarios for simulating search operators whose behaviors often change continuously during the search. In these scenarios, the performance of such operators decreases while they are applied. This is motivated by the fact that operators for optimization problems are often roughly classified into exploitation and exploration operators. Our simulation model is used to compare the performances of operator selection policies and to identify their ability to handle specific non-stationary operators. An experimental study highlights respective behaviors of operator selection policies when faced to such non-stationary search scenarios.

1. Introduction

Using a search algorithm to tackle an optimization problem mainly consists in applying basic solving operators—heuristics—in order to explore and exploit the problem search space for retrieving solutions. Well-known metaheuristics [18,12,23] are based on this principle, as well as sophisticated metaheuristics which have been developed through numerous metaphors [46]. Advanced neighborhood-based or evolutionary algorithms share some common principle according to this notion of basic search operators, especially when they have been combined to get hybrid metaheuristics. The use of appropriate operators is decisive to tackle effectively combinatorial (discrete) optimization problems for which the topology of the search space has to be carefully defined among different possible encoding of solutions. For instance, variation operators (mutation and recombination) are used in evolutionary algorithms, neighborhood-based operators are used in local search (several operators in variable neighborhood search [38]), local improvement operators are used in ant colony optimization [10] or velocity and position update functions, and sometimes local optimization processes, have to be defined in particle swarm optimization [26]. Most of the time, the designer of such algorithms has many choice concerning these search components. Therefore, the study or design of strategies for selecting the most suitable operators in search algorithms is an active research area [13,30]. Usually, the choice of the successive operators along the search process is driven by means of parameters. The improvement of the algorithm performance thus relies on an adequate setting of these parameters. An optimal setting may then be achieved by an optimal operator selection policy. Unfortunately, according to the underlying intuitions provided by the No Free Lunch theorems for optimization [54], this optimal policy may strongly depend on the problem instances to be solved.

Initial parameters setting can be achieved by automated tuning algorithms [25,39]. In this case the set of benchmarks chosen for tuning can greatly impact the algorithm performance [29]. Nevertheless, the values of parameters may require more continuous control [15] and should rather not be fixed during the whole search process. Note that such an adaptive operator selection is strongly related to reinforcement learning problems, and especially to multi-armed bandit problems [16,7]. Finally, managing the famous *exploration vs. exploitation* balance in search heuristics have been greatly investigated in the literature through various methods; see for instance [33,30,48].

1.1. Motivations and related work

The performance of adaptive selection policies depends on the characteristics of the problem's search space, as well as on the specificities of the search operators. Therefore different families of practical problems have been handled (e.g., permutation based problems [51], satisfiability problems [35]), but also more abstract operators models in order to provide more general and comprehensive testing frameworks. Note that abstract frameworks have been widely used in order to experiment and analyze the behavior and the performance of search algorithms, especially in evolutionary computation [37], with the use of Onemax, Royal Road or NK

* Corresponding author. Tel.: +33 241735273.

http://dx.doi.org/10.1016/j.asoc.2015.09.024
1568-4946/

functions [52], or more generally combinatorial fitness landscapes [3].

Focusing now on the adaptive selection of operators, Thierens [48] proposes epoch based scenarios in which rewards, drawn from a constant uniform distribution, are assigned to operators during each epoch. In Costa et al. [7], Boolean scenarios are proposed in order to consider null rewards for some operators and outliers scenarios that produce higher variance in the rewards. Fialho et al. [17] introduced two-values benchmarks in order to consider two possible levels of rewards for operators. Nevertheless, changes of operators efficiency are only considered by means of successive epochs.

In this paper, we propose a new model for simulating search operators whose behavior often change continuously during the search. In these scenarios, the more an operator is applied during a period, the more its performance will decrease. At the contrary, performance of operators can increase when other operators are applied. This is motivated by the fact that operators for optimization problems are often roughly classified into exploitation and exploration operators. Exploitation operators aim at focusing on the evaluation of the visited solutions (configurations) of the search space in order to converge quickly to a local optimum. Exploration operators aim at diversifying the search trajectory by visiting sparse areas of the search space. Unfortunately, it is not possible to always exploit nor explore the search space. For instance, it is unlikely that an exploitation operator will always improve a configuration and find directly an optimal solution (except for simple problems). Therefore, decreasing performance may be observed along the search as well as changing behaviors of operators, according to the current state of the search and to possible interactions between operators. For example, the relative improvements or expected improvements of an intensification operator are likely to decrease when approaching a local optimum. We already observed such behaviors while studying adaptive operator selection mechanisms on OneMax functions within an evolutionary process [6], as well as for permutation problems with different neighborhoods in local search algorithms [51,8]. Similarly, Ochoa et al. [41] observed that the crossover role may change during the search with regards to the notion of exploitation and exploration.

1.2. Contributions

While metaheuristics algorithms have been widely developed using different formulations [46], we choose here to focus on a simple common scheme. Hence, the general description of operator based search algorithms proposed in the following may be helpful in this design process when the user has to precisely identify the components and performance criteria that are used in the adaptive process.

To compare operator selection policies, we propose the use of specific non-stationary operators modeled by an original gain function where previous actions affect gain distributions. Such a simulation model can be used as a surrogate operator model when designing new adaptive search algorithms, since it makes possible to identify the ability of operator selection policies to manage exploitation and exploration operators. The aim is to provide abstract scenarios for modeling sets of operators whose behaviors may be subjected to progressive changes and interactions that require more intricate combinations when being applied and to evaluate different operator selection policies on these scenarios. Note that such adaptive policies for selecting operators may easily be introduced in various solving algorithms in order to schedule dynamically their basic search heuristics, as it has been already investigated for solving different optimization and combinatorial problems with evolutionary algorithms [9,20], local search [50] or constraint solvers [31].

More precisely, our scenarios consider operators whose gains decrease if they are too much applied, which corresponds, as mentioned above, to intensification operators approaching a local optimum. Of course two different operators with such decreasing gain may use different heuristics as thus should be alternated in order to improve their performance. For instance, local search intensification operators that use different neighborhood may be used complementary in order to avoid getting stuck into a local optima (since may have different local optima with regards to their neighborhood). Moreover, we also consider operators with null gain that may be used for diversification or that correspond to inefficient operators for the current instance of the problem. This may be very useful when considering generic solvers which include numerous possible solving strategies. Therefore, the basic model of scenarios described here—and more sophisticated variants which can easily be designed—allows the simulation of very different type of search situations, focusing on the operator management.

The experimental study gives information on the respective behaviors of operator selection policies when faced to such non-stationary search scenarios. We show that none of the selection policies achieve the best performance in every situation but results highlight that their respective performances rather depend on the specificities of the operators. Therefore, such comparison may help a user to determine the most appropriate selection policy according to her/his problem at hand.

At last, considered as a multi-armed bandit problem, the adaptive *non-stationary* operator selection problem corresponds to a specific restless bandit problem that could be used to model different real applications as soon as the efficiency of a given action decreases according to successive frequent uses. For instance, such reinforcement learning techniques are now widely used for recommendation on the web [28] in order to manage adaptive content. Our model could be pertinent in this context since it may be clear that the relevance of an advertisement decreases if it is too much shown to the same user. Other cases of such repeated decreasing actions may actually be observed in various application domains.

1.3. Organization of the paper

In Section 2, we formally describe optimization algorithms that are based on applications of basic search operators. We also define the problem of designing the best possible operator selection policy and show its relationship with multi-armed bandit problems. Section 3 is dedicated to review different operator selection policies. Section 4 presents our model for simulating non-stationary operators. Experiments are presented in Section 5.

2. Operator based search algorithms for optimization problems

In this section, we propose to precisely define the components of a search algorithm in the context of solving optimization problems, in order to be able to manage their behavior. We focus on algorithms that use operators—also commonly called heuristics—which aims at determining pertinent solutions in the search space. Our purpose is to progressively introduce and discuss the different aspects that must be taken into account when one wants to improve search algorithms.

We first propose a generic algorithmic scheme that allows us to precisely define the main components of an operator based search algorithm. We discuss the notion of performance with regards to the operational semantics of the algorithm and recall two general methodologies for improving this performance, namely tuning and control. Focusing on the dynamic control of search algorithms, we focus then on the notion of policy, that defines how