Applied Soft Computing 28 (2015) 312–326

Contents lists available at ScienceDirect

Applied Soft Computing

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

# Choice function based hyper-heuristics for multi-objective optimization

Mashael Maashi*, Graham Kendall[1], Ender Özcan

*Automated Scheduling, Optimisation and Planning Research Group, University of Nottingham, School of Computer Science, Jubilee Campus, Wollaton Road, Nottingham NG8 1BB, UK*

ARTICLE INFO

*Article history:*
Received 16 January 2014
Received in revised form 1 December 2014
Accepted 2 December 2014
Available online 17 December 2014

*Keywords:*
Hyper-heuristic
Metaheuristic
Great deluge
Late acceptance
Multi-objective optimization

ABSTRACT

A selection hyper-heuristic is a high level search methodology which operates over a fixed set of low level heuristics. During the iterative search process, a heuristic is selected and applied to a candidate solution in hand, producing a new solution which is then accepted or rejected at each step. Selection hyper-heuristics have been increasingly, and successfully, applied to single-objective optimization problems, while work on multi-objective selection hyper-heuristics is limited. This work presents one of the initial studies on selection hyper-heuristics combining a choice function heuristic selection methodology with great deluge and late acceptance as non-deterministic move acceptance methods for multi-objective optimization. A well-known hypervolume metric is integrated into the move acceptance methods to enable the approaches to deal with multi-objective problems. The performance of the proposed hyper-heuristics is investigated on the Walking Fish Group test suite which is a common benchmark for multi-objective optimization. Additionally, they are applied to the vehicle crashworthiness design problem as a real-world multi-objective problem. The experimental results demonstrate the effectiveness of the non-deterministic move acceptance, particularly great deluge when used as a component of a choice function based selection hyper-heuristic.



## 1. Introduction

Hyper-heuristics perform a search over the space of heuristics when solving problems. In a hyper-heuristic approach, different heuristics or heuristic components can be selected, generated or combined to solve a given computationally difficult optimization problem in an efficient and effective way. A selection hyper-heuristic, which is the focus of this study, manages a predetermined set of low level heuristics with the goal of choosing the best one at any given time using a performance measure maintained for each low level heuristic. This type of hyper-heuristic comprises two main stages: *heuristic selection* and *move acceptance* strategy. A selection hyper-heuristic is often described as *heuristic selection-move acceptance*. Hyper-heuristics are sufficiently general and modular search methods enabling reuse of their components for solving problems from different domains [1]. The task of heuristic selection, also referred to as the *high level strategy*, is to guide the search intelligently and adapt taking into account the success/failure of the low level heuristics or combinations of heuristic components during the search process.

The low level heuristics in a selection hyper-heuristic framework are in general human designed heuristics which are fixed before the search starts. An initial solution (or a set of initial solutions) is iteratively improved using the low level heuristics until some termination criteria are satisfied. During each iteration, the heuristic selection decides which low level heuristic will be employed next. After the selected heuristic is applied to the current solution(s), a decision is made whether to accept the new solution(s) or not using an acceptance criteria. Usually, in a selection hyper-heuristic framework, there is a clear separation between the high level strategy and the set of low-level heuristics or heuristic components. It is assumed that there is a domain barrier between them [2]. The purpose of domain barrier is increase the level of the generality of hyper-heuristics by being able to apply it to a new of problem without changing the framework. Only a new set of problem-related low-level heuristics need to be supplied. The barrier allows only problem domain independent information to flow

* Corresponding author at: Department of Computer Science, Faculty of Computer and Information Technology, University of Tabuk, Saudi Arabia.
Tel.: +966506620227.
*E-mail addresses:* m.maashi@gmail.com, m.maashi@ut.edu.sa (M. Maashi), graham.kendall@nottingham.edu.my (G. Kendall), ender.ozcan@nottingham.ac.uk (E. Özcan).
[1] Address: The University of Nottingham Malaysia Campus, Malaysia.

http://dx.doi.org/10.1016/j.asoc.2014.12.012
1568-4946/

from the low level to the high level, such as the fitness/cost/penalty value measured by an evaluation function, indicating the quality of a solution [3]. Low level heuristics, or heuristic components, are the problem domain specific elements of a hyper-heuristic framework. Hence they have access to any relevant information, such as candidate solution(s).

Many real-world optimization problems are multi-objective requiring improvement of more than one objective, simultaneously. Often, there is some trade-off between multiple conflicting objectives [4–7]. Hence, the multi-objective approaches provide a set of improved solutions (not a single solution as in single objective optimization) capturing the trade-off between those objectives for a given problem at the end of the search process. There is a variety of population based approaches for multi-objective optimization in the scientific literature, such as NSGAII [8], SPEA2 [9], and MOGA [10]. However, there are a few studies on multi-objective selection hyper-heuristics. To the best of the authors' knowledge, this paper is one of the first studies that investigate the influence of the move acceptance component on the performance of a selection hyper-heuristic for multi-objective optimization. In this study, we extend our previous work in [11] which describes a HHMO_CF_AM multi-objective hyper-heuristic controlling a set of low level (meta-)heuristics (NSGAII [8], SPEA2 [9], and MOGA [10]). We have adopted the great deluge algorithm (GDA) and late acceptance (LA) separately as a non-deterministic move acceptance component of a selection hyper-heuristic for multi-objective optimization and we have tested the performance of the overall algorithm using the same set of low level heuristics as in our previous study on the well-known Walking Fish Group (WFG) benchmark instances [12]. Moreover, we have applied the proposed selection hyper-heuristics with embedded GDA and LA, on a multi-objective real-world problem of *vehicle crashworthiness* [13] for which a solution aims to provide a vehicle design satisfying multiple objectives reducing different types of injuries as much as possible for the passengers within the vehicle during a crash. The empirical results are aligned with the previous observations for single objective optimization [14] that different combinations of heuristic selection and move acceptance under a selection hyper-heuristic framework yield different performances. Move acceptance components could be extremely influential on the overall performance of a selection hyper-heuristic. Moreover, the proposed multi-objective hyper-heuristic, embedding GDA, turns out to be an effective, reusable and general approach for multi-objective optimization. The empirical results show that it is the best option as a multi-objective selection hyper-heuristic move acceptance component, outperforming each individual low level (meta-)heuristic run on their own for the WFG instances and NSGA II for the vehicle crashworthiness design problem.

The rest of the paper is organized as follows. In Section 2, a broad overview of the scientific literature on move acceptance methods, in particular the great deluge and late acceptance algorithms, is provided. An overview of existing studies on multi-objective selection hyper-heuristics and a selection hyper-heuristic framework supporting the use of great deluge and late acceptance move acceptance methods for multi-objective optimization are covered in Section 3. The experimental results for the proposed hyper-heuristics to the WFG benchmark and vehicle crashworthiness problem instances are provided in Sections 4 and 5, respectively. Finally, the conclusions are presented in Section 6.

## 2. Move acceptance methods

The choice of heuristic selection and move acceptance methods in selection hyper-heuristics influences the performance of a hyper-heuristic [14]. A move acceptance criterion can be *deterministic* or *non-deterministic*. A deterministic move acceptance criterion produces the same result given the same initial solutions. A non-deterministic move acceptance criteria may generate a different result even when the same solutions are used. This could be because the move acceptance criterion depends on time or it might have a stochastic component while making the accept/reject decision. Examples of deterministic move acceptance criteria are All-Moves, Only-Improving and Improving & Equal. In All-Moves, the candidate solution is always accepted whether a move worsens or improves the solution quality. The candidate solution in Only-Improving criteria is accepted only if it improves the solution quality, while in Improving & Equal criteria, the candidate solution is accepted only if it improves or it is equal to the current solution. For a non-deterministic move acceptance criteria, the candidate solution is always accepted if it improves the solution quality, while the worsening solution can be accepted based on an acceptance function some of which include the great deluge algorithm [15], simulated annealing [16] and Monte Carlo [17].

The *choice function* (CF) is introduced as a heuristic selection method as part of a selection hyper-heuristic in Cowling et al. [18]. The choice function maintains a score for each low level heuristic and chooses the one with the highest score at each decision point during the search process. A low level heuristic's score depends on whether or not the heuristic generates improvement when used individually, when used in cooperation with another heuristic and how much time has been passed since its last invocation. This initial study has been followed by many other studies indicating the success of choice function based hyper-heuristics using different move acceptance methods on different problems. Cowling et al. [19] developed an approach using several proposed hyper-heuristic components in order to solve a real-world scheduling problem; namely project presentations. The approach employed deterministic move acceptance strategies {All-Moves, Only-Improvements} and seven heuristic selection methods {Simple Random, Random Gradient, Random Permutation, Random Permutation-Gradient, Greedy, Reinforcement Learning, Choice Function}. The experimental results show that choice function all-moves performs better than simple random moves over the given problems, and produced better solutions than those produced by humans.

There are a few comparative studies which evaluate the performances of different heuristic selection and move acceptance methods. A set of seven different heuristic selection strategies (Simple Random, Random Descent, Random Permutation, Random Permutation Descent, Greedy, Choice Function, Tabu Search) are combined with a set of five acceptance strategies {All-Moves, Only-Improving, Improving & Equal, Exponential Monte Carlo with Counter, GDA}. The combination set is tested on fourteen benchmark functions against genetic and mimetic algorithms. Choice Function-Improving & Equal performs the best [14]. Another study was conducted by Bilgin et al. [20] using a set of eight heuristic selection strategies {Simple Random, Random Gradient, Random Permutation, Random Permutation Gradient, Greedy, Choice Function, Reinforcement Learning, Tabu Search} and five move acceptance strategies {All-Moves, Only-Improving, Improving &Equal, GDA, EMCQ} which were tested on different timetabling benchmark problems. The study showed that there is no one strategy that dominates. In the scientific literature, a wide variety of hyper-heuristics have been proposed that use different heuristic selection and acceptance strategies in different domains: packing, vehicle routing, timetabling, channel assignment, component placement, personnel scheduling, planning and shelf space allocation (see Ref. [21]). The choice function simulated annealing hyper-heuristic performed better than a simple random great deluge hyper-heuristic over a set of examination timetabling problems as presented in [20]. In [22] different heuristic selection methods {Simple Random, Greedy, Reinforcement Learning, Reinforcement,