Applied Soft Computing 10 (2010) 711–718

Contents lists available at ScienceDirect

Applied Soft Computing

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

Assessing solution quality of biobjective 0-1 knapsack problem using evolutionary and heuristic algorithms

Rajeev Kumar *, P.K. Singh

Department of Computer Science and Engineering, Indian Institute of Technology, CSE Building, Kharagpur, WB 721 302, India

ARTICLE INFO

Article history:
Received 4 August 2009
Accepted 28 August 2009
Available online 4 September 2009

Multiobjective problem
Combinatorial optimization
Evolutionary algorithm
Heuristics
Pareto-front
0-1 knapsack

ABSTRACT

Multiobjective 0-1 knapsack problem involving multiple knapsacks is a widely studied problem. In this paper, we consider a formulation of the biobjective 0-1 knapsack problem which involves a *single* knapsack; this formulation is more *realistic* and has many industrial applications. Though it is formulated using simple linear functions, it is an NP-hard problem. We consider three different types of knapsack instances, where the weight and profit of an item is (i) uncorrelated, (ii) weakly correlated, and (iii) strongly correlated, to obtain generalized results. First, we solve this problem using three well-known multiobjective evolutionary algorithms (MOEAs) and quantify the obtained solution-fronts to observe that they show good diversity and (local) convergence. Then, we consider two heuristics and observe that the quality of solutions obtained by MOEAs is much inferior in terms of the extent of the solution space. Interestingly, none of the MOEAs could yield the entire coverage of the Pareto-front. Therefore, based on the knowledge of the Pareto-front obtained from the heuristics, we incorporate problem-specific knowledge in the initial population and obtain good quality solutions using MOEAs too. We quantify the obtained solution fronts for comparison.

The main point we stress with this work is that, for real world applications of *unknown* nature, it is indeed difficult to realize how good/bad is the quality of the solutions obtained. Conversely, if we know the solution space, it is trivial to obtain the desired set of solutions using MOEAs, which is a paradox in itself.

1. Introduction

The 0-1 knapsack problem is a well-studied combinatorial optimization problem and much research has been performed on many variants of the problem [1,28]. There are single and multiobjective versions of the problem involving one and m-dimensional knapsacks [9,15]. Even the single objective case has been proven to be NP-hard. Much research for the single objective case has been performed over the decades and the problem continues to be a challenging area of research.

There are several effective approximation heuristics for solving knapsack problems. Ibarra and Kim [12] proved the existence of a fully polynomial time approximation scheme (FPTAS) for the 0-1 knapsack problem. For the single objective m-dimensional knapsack problem a polynomial-time approximation scheme (PTAS) was presented by Frieze and Clarke [10]. Their algorithm makes use of the fact that linear programs can be solved in polynomial time. Erlebach et al. [9] described a practical FPTAS for the multiobjective one-dimensional knapsack problem. They described a PTAS based on linear programming for the m-dimensional knapsack problem also.

In general, the multiobjective variant of the problem is *harder* than the single objective case. The multiobjective optimizer is expected to obtain a set of all *representative* equivalent and diverse solutions [4,8]. The set of all optimal solutions is called the Pareto-front. Objectives to be *simultaneously* optimized may be mutually conflicting. Additionally, achieving proper diversity in the solutions while approaching convergence is another challenge in multiobjective optimization, especially for *unknown* problems in black-box optimization. Moreover, the size of the obtained Pareto-front may be exponentially large.

Evolutionary algorithms (EAs) are emerging as a powerful black-box tool to solve combinatorial optimization problems. EAs use a randomized search technique with a *population* of individuals. The genetic operators used by EAs, in general, do not apply any problem-specific knowledge. However, special genetic operators may be designed by incorporating domain knowledge to expedite the search in certain applications. In multiobjective scenario, we aim to effectively obtain a set of diverse and mutually competitive solutions. Some results for solving computationally hard problems empirically using multiobjective EAs (MOEAs) are available in the literature, e.g., m-dimensional knapsack [35], minimum spanning

* Corresponding author.
E-mail address: rkumar@cse.iitkgp.ernet.in (R. Kumar).

1568-4946/$ – see front matter © 2009 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2009.08.037

tree [24], partitioning of high-dimensional patterns spaces [22], code-book design [26], communication network topology design [20], and network design [18].

Zitzler and Thiele [35] pioneered the work of solving multiobjective 0-1 knapsack problem using EAs. They formulated the problem using m knapsacks and maximized the profits *simultaneously* for all the m knapsacks within weight constraints. Subsequently, many other researchers (e.g., [29,14,16]) attempted to solve the same problem-formulation using many other variants of EAs. In this work, we consider another formulation of 0-1 knapsack problem which uses a *single* knapsack. The considered multiobjective formulation, in this paper, is natural and more realistic, and has many industrial applications manifested by bin-packaging of a single set of items. Though it is expressed using linear equations, it is an NP-hard problem. We solve this problem-formulation using well-known MOEAs and obtain interesting insight in to the EA problem solving strategy for real-world applications whose solution space is not known *a priori*.

For comparison of results obtained using MOEAs, we consider two heuristics and observe that the solutions obtained by the heuristics are much superior for larger problem instances than those obtained by MOEAs. Then, with the aim to get equally good solutions from MOEAs, we take a deeper look into the dynamics of population evolutions and infer that the genetic operators are not strong enough to extend the coverage of the solution-front as obtained by the heuristics. Therefore, we apply this knowledge of the problem domain to improve upon and obtain equally good solutions by MOEAs too.

Just obtaining quality solutions for a real-world application using MOEAs is not the main point we wish to highlight through this research monogram. The important point which we stress is that it is difficult to assess the quality of solutions obtained using EAs for the problems whose solution spaces are *unknown* as the available metrics to measure the quality of solutions in terms of convergence, diversity, and extent for such problems are inadequate in absence of any reference. (An initial version of this work appeared in web proceedings of a conference [25].)

The rest of the paper is organized as follows. We describe, in Section 2, a brief review of the issues to be addressed for achieving quality solutions in the context of a MOEA. In Section 3, we include a few definitions pertaining to multiobjective optimization, and formulate the 0-1 knapsack problem which we address in this paper. We include, in Section 4, two fast heuristics which we use to compare the solutions obtained by MOEAs. We present results obtained by the heuristics and MOEAs for uncorrelated knapsack instances in Section 5 and for weakly/strongly correlated knapsack instances in Section 6. Finally, we draw conclusions in Section 7.

2. Multiobjective evolutionary algorithms: an overview

EAs have emerged as powerful black-box optimization tools to approximate solutions for NP-hard combinatorial optimization problems. In multiobjective scenario, EAs often find effectively a set of mutually competitive solutions without applying much problem-specific information. However, achieving proper diversity in the solution-set while approaching convergence is a challenge in multiobjective optimization, especially for *unknown* problems.

2.1. Achieving diversity

The commonly used techniques for preventing genetic drift and promoting diversity are: sharing, mating restrictions, density count (crowding), clustering, and pre-selection operators. These approaches can be grouped into two classes: parameter-based sharing and parameter-less sharing. The niching/sharing techniques have been commonly employed to obtain a diverse set of solutions although such techniques work best when one has *a priori* knowledge of the solution. It is the experience of almost all researchers that proper tuning of sharing parameters is necessary for effective performance.

In recent years, much work has been done on parameter-less diversity preserving approaches. Most of the newer MOEAs (e.g., NSGA-II [6] and SPEA2 [34]) have now dispensed away or some MOEAs (e.g., PCGA [23]) do not use explicit parameters for diversity preserving. (The NSGA-II and SPEA2 use parameter-less crowding and clustering, respectively.)

2.2. Monitoring convergence

A common metric used for convergence is the distance metric which finds distance of the obtained solution front from the true Pareto-front; this is trivially done for known problems. Such a metric is based on a reference. In real-world search problems, location of the true Pareto-front, by definition, is unknown. A commonly practiced approach to determine the reference for unknown problems is to extract the reference from the best solutions obtained so far, which is incrementally updated with every generation in iterative refinement based algorithms.

Some other studies have been done on combining convergence with diversity. Laumanns et al. [27] proposed an ϵ-dominance for getting an ϵ-approximate Pareto-front for problems whose optimal Pareto-set is *known*. As to on-line convergence, some authors have proposed running performance metrics for convergence, e.g., Deb and Jain [5] and Tan et al. [31]. Deb and Jain's convergence metric evaluates convergence towards a reference set and is akin to monitoring hyper-volume measure; this may not be used effectively for unknown problems. Kumar and Rockett [23] proposed use of rank-histograms which assess the movement of solution-front towards convergence without needing a true Pareto-front.

2.3. Avoiding local convergence

For solving unknown problems there is a common concern, whether the obtained solution front is close to the true Pareto-front or not. For some problems (e.g., the bimodal problem [7,23]), apparently, it seemed that the EA had converged to the Pareto-front but conceivably it had got *stuck* at some sub-optimal point. Such a local minima cannot be detected for unknown problems by most of the known metrics because an obtained local front gives excellent numerical values for both diversity and convergence.

For such cases, there is little point in continuing the optimization and it should be terminated. We argue that there is always a certain *inheritance* of genetic material or content belonging to one population and there may not be much appreciable evolutionary gain beyond a certain number of generations. This implies that the genetic precursors available within a finite population may be inherently incapable of evolving to the true Pareto-front. Instead, we suggest that alternative genetic material should be acquired in the form of another population. Each population sample is run to its *own* convergence, the obtained solutions are then merged and tested across populations. Such a multi-island approach is essentially a test on convergence rather than parallelizing the computational efforts as done by others, e.g., Cantu-Paz [3]. Therefore, we suggest this strategy of EA optimization through multiple independently initialized populations – as a *test* on convergence – to be particularly suited to *harder* problems of unknown nature, see Kumar and Rockett [23]. Additionally, this strategy can be used for parallelism.