Using a perturbation strategy for a variant of the knapsack problem

Isma DAHMANI[1], Meriem FERROUM[1], Mhand HIFI[2], Shohre SADEGHSA[2]

[1] AMCD-RO, USTHB, BP 32, El Alia, 16111 Bab Ezzouar, Alger, Algérie
{dahmani.isma,meriem1.mf}@gmail.com

[2] EPROAD, UPJV, 7 rue du Moulin Neuf, 80000 Amiens, France
{mhand.hifi,shohre.sadeghsa}@u-picardie.fr

Mots-clés : *Knapsack, greedy, optimization, perturbation.*

1 Introduction

In this paper, a variant of the well-known binary Knapsack Problem (namely KP) is tackled ; that is, the binary Set Union Knapsack Problem (namely SUKP) −Goldschmidt et al. [1]−.

An instance of SUKP is composed of two sets : a first set containing elements $U = \{1, ..., n\}$ and a set of items $S = \{1, ..., m\}$. Each item $i \in S$ corresponds to a subset of elements, namely S_i, with a nonnegative profit $p : S \to R^+$ and each element is characterized by a nonnegative weight $w : U \to R^+$. The goal of the problem is to determine a subset S^* of items such that $S^* \subseteq S$, where the overall profit $P(S^*)$ is to maximize without exceeding the knapsack capacity ($W(S^*) \leq B$), with B a given budget.

2 Solution procedure

The SUKP is tackled with an iterative search procedure, where a descent procedure is combined with an intensification search and reinforced by using a tabu list. The proposed approach may be summarized by the following steps :

1. *A starting solution.* A initial feasible solution is reached by using a special greedy procedure (Arulselvan [5]).
2. *A reactive search.* A current solution is enhanced by applying several neighborhood procedures-based strategies. As used in Hifi and Michrafy [2], a reactive search is called, where it combines two complementary strategies :
   (i). Degrading strategy. It tries to build a partial solution by randomly dropping some items from the current solution.
   (ii). Augmenting strategy. It tries to select better unassigned items in order to repair the partial solution by augmenting the quality of the solution at hand.
3. *Intensification search.* Herein, both 2-opt and 3-opt are used in order to intensify the search procedure around the solution provided from the last step.
4. *Tabu list.* Such a list is used in order to avoid some cycling and so, accelerating the search process and improving, in some cases, the quality of the final solutions.

Such a process (from step 2 to step 4) is iterated until a maximum number of iterations is reached ; then, the final solution achieved by the method is retuned as the best solution for the problem.

3 Preliminary experimental part

The proposed method was analyzed on a set of benchmark instances extracted from Ozsoydan and Baykasoglu [3]. The purpose of this part is to evaluate the effectiveness of the method when its

#Inst	Results from [3, 4] Av.	Best	This work Av.	Best
`sukp 100_100_0,10_0,75`	13854,71	14044	14044	14044
`sukp 100_100_0,15_0,85`	13347,58	13508	13454,5	13508
`sukp 200_200_0,10_0,75`	11898,73	12522	12522	12522
`sukp 200_200_0,15_0,85`	11584,64	12317	12090,27	12317
Av	12671,41	13097,75	13027,69	13097,75
`sukp 100_85_0,10_0,75`	11486,95	13283	13186,95	13283
`sukp 100_85_0,15_0,85`	11994,36	12274	12352,26	12479
`sukp 200_185_0,10_0,75`	13204,26	13405	13521	13521
`sukp 200_185_0,15_0,85`	10801,41	14044	14102,24	14215
Av	11871,74	13251,50	13290,61	13374,50

TAB. 1 – Behavior of the iterative method versus two meta-heuristics of the literature

achieved results are compared to those published in both Ozsoydan and Baykasoglu [3], and He et al. [4].

Table 1 reports the results, on some instances, achieved by the proposed methods ans those reached by both Ozsoydan and Baykasoglu's [3] and He et al's. [4] algorithms. Column 1 of the table displays the instance label tested, columns 2 and 3 show both best and average bounds (throughout ten trials for each method) related to the best results between the compared meta-heuristics , and columns 4 and 5 tally those achieved by the proposed method. From Table 1, one can observe that the proposed method remains competitive, since it is able to provide three new bounds and matches the other bounds. Note that the proposed method achieves all bounds using an equivalent average runtime when compared to those needed by the available methods in the literature.

4 Conclusion

In this paper, a simple reactive search-based algorithm was proposed for approximately solving a variant of the binary knapsack problem : the set union knapsack problem. The aforementioned problem was tackled by using a combined search procedure, where dropping and repairing strategies were combined with a standard local search and enhanced with a tabu strategy. The preliminary results showed that the proposed simple approach remains competitive by achieving interesting results.

Références

[1] O. Goldschmidt, D. Nehme, and G. Yu. On the set-union knapsack problem. *Naval Research Logistics*, 41(6) :833-842, 1994.

[2] Hifi, M., Michrafy, M. (2006). A reactive local search-based algorithm for the disjunctively constrained knapsack problem. *Journal of the Operational Research Society*, 57, 718–726.

[3] F-B. Ozsoydan and A. Baykasoglu. A swarm intelligence-based algorithm for the set-union knapsack problem. *Future Generation Computer Systems*, 93 :560-569, 2019.

[4] Y He, H Xie, T-L Wong and X Wang. A novel binary artificial bee colony algorithm for the set-union knapsack problem. *Future Generation Computer Systems*, 78(1) :77-86, 2018.

[5] A Arulselvan. A note on the set union knapsack problem, *Discrete Applied Mathematics*, 169 :214-218, 2014.