A Practical Case of the Multiobjective Knapsack Problem: Design, Modelling, Tests and Analysis

Brahim Chabane[1,2](✉), Matthieu Basseur[1], and Jin-Kao Hao[1]

[1] LERIA, Université d'Angers, 2 Bd Lavoisier, 49045 Angers Cedex 01, France
{chabane,basseur,hao}@info.univ-angers.fr
[2] GePI Conseil, 14 Place de la Dauversire, 49000 Angers, France

Abstract. In this paper, we present a practical case of the multiobjective knapsack problem which concerns the elaboration of the optimal action plan in the social and medico-social sector. We provide a description and a formal model of the problem as well as some preliminary computational results. We perform an empirical analysis of the behavior of three metaheuristic approaches: a fast and elitist multiobjective genetic algorithm (NSGA-II), a Pareto Local Search (PLS) algorithm and an Indicator-Based Multi-Objective Local Search (IBMOLS).

1 Introduction

During the last decades, combinatorial optimization has received great interest and takes an important and even strategic place in industrial settings. Multiobjective metaheuristics have proven their efficiency for solving many practical problems, which usually consist in handling simultaneously several conflicting objectives [2].

The aim of this paper is to present a practical problem, proposed by the company "GePI" which works in the social and medico-social domain. This study is unique in the sector because even if this sector is increasingly computerized these last years, it remains among the sectors where optimization is not yet used as a tool for decision support.

The problem considered in this paper consists in elaborating action plans in order to improve the overall management of the considered structure. The aim is to choose a subset of actions among many possible actions while optimizing several objectives. Each action has a realization cost and can influence other objectives (positively or negatively). The global cost of the solution should not exceed a predefined budget. Our problem is a multiobjective knapsack problem [5,8], which is well known in the literature. The action plan represents the knapsack and the selected actions represent the items to put in the knapsack respecting the budget constraint.

The considered problem can include more than one thousand possible actions and involve up to eight objectives. Here, we are interested in providing efficient techniques in terms of solutions quality and response time.

L. Jourdan et al. (Eds.): LION 9 2015, LNCS 8994, pp. 1–7, 2015.
DOI: 10.1007/978-3-319-19084-6_23

In the following, a description and a formal model of the problem are first introduced. Then, the ways of generating problem instances is provided. Next, we present the first results using three metaheuristic algorithms: PLS (Pareto Local Search) [7,9], IBMOLS (Indicator-Based Multi-Objective Local Search) [1] and NSGA-II [4]. Finally, we end with a conclusion and the future work.

2 Problem Modeling and Description

This project is a part of "*MSQualité*" software developed by the company GePI which is dedicated specifically to the social and medico-social sector that includes 34000 different structures (rest houses, accommodation and rehabilitation centers, work-based support centers, etc.) [10]. Even if the use of computer resources has made considerable progress in recent years in this sector, they are basically employed for the daily management of the structures. In particular, optimization tools are completely absent. In this context of lack of advanced models and tools, GePI has decided to set up this project to develop a multiobjective decision support system to assist managers in their action plan elaboration.

We can define an action plan p as a subset of actions selected among a set of feasible actions A, in order to maximize or minimize a set F of conflicting objectives. p can be represented by a vector $p = (a_1, a_2, ..., a_n)$ with n equal to the size of A. $a_i = 1$ if the action a_i is selected and $a_i = 0$ otherwise. The set of the possible action plans (solutions) is denoted by P. The origins of the actions are either issued from action plans already made in the structure itself or other similar structures, or are decided by the managers for continuous improvements.

The objectives can be of varied nature, namely qualitative (such as "improve resident's quality of life") or quantitative (such as "increase the resident's autonomy"). In both cases, each objective is represented by an objective function f_j which associates to every action a_i ∈A its impact on the objective j.

An action $a_i \in A$ can have a positive or a negative impact on an objective $f_j \in F$. This impact is evaluated by the function $f_j(a_i) = v_{ij}$ which assigns to any action a_i an integer value $v_{ij} \in [-100, +100]$ that represents the contribution of the action a_i to the achievement of the objective j ($v_{ij} > 0$) or the degradation of the action a_i for the objective j ($v_{ij} < 0$). v_{ij}=0 when the action a_i has no effect on the objective j. Thus, we can associate to each action a_i an objective vector v=$(f_1(a_i), f_2(a_i), ..., f_m(a_i))$ with m equal to the number of objectives to optimize. We define m in the interval [2,8] because in practice, the projects can have up to eight objectives (otherwise the project management and evaluation[1] will be difficult).

Considering an action plan $p = (a_1, a_2, ..., a_n) \in \{0, 1\}^n$, the impact that p^* has on an objective j is obtained by:

[1] In social and medico-social structures, a project is defined for a period of five years. At the sixth year, the evaluation of the project is carried out and the attainment of each objective is measured. Therefore, the more there are objectives, the more the evaluation is difficult.

$$f_j(p) = \sum_{i=1}^{n} a_i f_j(a_i) \tag{1}$$

Thus, an objective vector $z = (f_1(p), f_2(p), ..., f_m(p))$ is associated to each solution $p \in P$. A constraint c_j is added for every objective j determining the minimal threshold accepted for f_j. In the following, we consider that all the objectives must be improved:

$$f_j(p) \geq c_j \geq 0 \tag{2}$$

An additional constraint concerns the realization cost of the solution which should not exceed some budget β fixed by the decision maker. Indeed, each action a_i has a realization cost ω_i which can take negative values since there may be actions with negative cost when it is about selling of objects or services. Actions with no cost are also to be taken into account. The global cost of a solution p corresponds to the following cost sum of the actions of p:

$$\begin{cases} W(p) = \sum_{i=1}^{n} a_i \omega_i \\ W(p) \leq \beta \end{cases} \tag{3}$$

So, the optimization goal aims to find $p^* \in \arg\max_{p \in P} F(p)$ verifying:

$$\begin{cases} p^* \in \{0,1\}^n \\ \forall_j \in \{1, m\}, f_j(p^*) \geq c_j \\ \sum_{i=1}^{n} a_i \omega_i \leq \beta \end{cases} \tag{4}$$

Since we deals with a multiobjective case, p^* is not unique. Instead, we obtain a set of non-dominated solutions (in Pareto optimality sens). The aim is to approximate the Pareto front effectively.

3 Instance Generation

Based on the above model, we have randomly generated a number of instances with different sizes (actions) {50,100,250,500,750,1000} and different number of objectives $m \in \{2, ..., 8\}$. We have also generated several partially structured instances which are more representative of real cases. To be as close as possible to the real problem, for each objective function, an action has a chance of 50 % to be neutral, 40 % to have a positive impact and 10 % to have a negative impact. Moreover, the cost of 40 % of the actions is set to 0. The non-null action values are uniformly taken from the interval [0,100] (positively or negatively). The non-null action costs are uniformly taken in the interval [-10000,10000].

4 Preliminary Results

We have tested, on random instances, three metaheuristic algorithms: NSGA-II, IBMOLS and PLS. For NSGA-II, we have used a population of size 100, a mutation probability of $1/n$ (where n is the number of the actions). The initial population is generated randomly while verifying that the cost of the individuals do not exceed the budget β. For IBMOLS, we have used the iterative version with a population of size 10 (the initial population is generated in the same way as NSGA-II) and the epsilon indicator as realized in [1] and in [12]. The fitness of each individual in the population is evaluated, with respect to the rest of the population, using the following formula:

$$I_\epsilon(P\backslash\{x\}, x) = \sum_{z \in P\backslash\{x\}} -\exp^{-I_\epsilon(z,x)/k} \tag{5}$$

where $k > 0$ represents the scaling factor [1] (k is set to 0.01 in our experiments).

NSGA-II is compared with IBMOLS and shows to be inferior to IBMOLS on the large size problems. Indeed, both algorithms use a bounded population and the same selection strategy: one random neighbor of each individual of the current population is selected to be a member of the child population in NSGA-II or to integrate the current population in IBMOLS.

PLS [7] is used with an archive of unbounded size and an initial population of one individual. The neighborhood generation is the same as for PLS and IBMOLS. The i^{th} neighbor of the solution $p = (a_1, a_2, ..., a_n)$ is obtained by flipping the value of a_i and only the neighbors verifying the constraint β are accepted. The budget constraint β is fixed to one million e for the three methods.

For the quality assessment, we have performed 30 runs of each method to solve each instance. For IBMOLS and NSGA-II, a run time of $n^2 * m$ milliseconds is used for each run (where n is the number of actions and m is the number of objectives). But for PLS, the run time is limited to one hour because the size of the archive and the response time increase exponentially with the instance size, making PLS inefficient for large size problems. Our experiments are realized on an Intel core i5-2400 CPU machine with 2 x 3.10 Ghz and 16 Gb of RAM. Then, we have evaluated our outputs using the R and ϵ indicators and computed their average values over the 30 runs for each algorithm and each tested instance. For the statistical analysis, we have used the Mann-Whitney test. In our experiments, we say that algorithm A outperforms algorithm B if the Mann-Whitney test provides a confidence level greater than 95 %. To calculate the indicator values and the Mann-Whitney test, we have used the performance assessment package (PISA) [6] which is available at: http://www.tik.ee.ethz.ch/sop/pisa/?page=assessment.php.

Table 1 shows a comparison of NSGA-II and IBMOLS in terms of the mean values obtained for R and ϵ indicators over 30 runs, using 30 instances with different sizes. The first column presents the instance size "m_n" where m and n are the number of objectives and actions respectively. The values in bold mean that the corresponding algorithm is at least 95 % statistically better than the other one for the considered instance and indicator.

Table 1. Comparison of mean values of I_ϵ and I_R of IBMOLS and NSGA-II

Instance	I_ϵ		I_R	
	NSGA-II	IBMOLS	NSGA-II	IBMOLS
2_50	**0.520**	0.135	**0.160**	0.030
2_100	**0.520**	0.135	**0.160**	0.030
2_150	**0.491**	0.200	**0.159**	0.055
2_250	**0.521**	0.283	**0.174**	0.074
2_500	**0.558**	0.358	**0.191**	0.097
2_1000	**0.567**	0.306	**0.191**	0.072
3_50	**0.415**	0.229	**0.112**	0.044
3_100	**0.412**	0.368	**0.119**	0.096
3_150	**0.461**	0.386	0.122	0.108
3_250	0.442	0.411	0.108	0.109
3_500	0.451	**0.482**	0.121	**0.143**
3_1000	0.480	**0.654**	0.119	**0.086**
4_50	0.401	0.398	0.094	0.086
4_100	0.351	**0.438**	0.083	**0.108**
4_150	0.451	0.450	0.099	**0.133**
4_250	0.405	**0.537**	0.100	**0.189**
2_500	0.347	**0.594**	0.089	**0.230**
4_1000	0.396	**0.709**	0.085	**0.278**
5_50	**0.364**	0.292	**0.073**	0.047
5_100	0.375	**0.459**	0.081	**0.111**
5_150	0.396	**0.556**	0.092	**0.165**
5_250	0.316	**0.650**	0.077	**0.246**
5_500	0.374	**0.700**	0.086	**0.285**
5_1000	0.307	**0.78**8	0.084	**0.353**
6_50	**0.381**	0.350	0.091	0.085
6_100	0.267	**0.385**	0.046	**0.084**
5_150	0.291	**0.564**	0.066	**0.204**
6_250	0.219	**0.664**	0.042	**0.278**
6_500	0.336	**0.748**	0.104	**0.368**
6_1000	0.199	**0.846**	0.058	**0.459**

From Table 1 we can conclude that on the whole NSGA-II is more efficient on the small instances (instances with 50 actions or no more than 3 objectives) but IBMOLS performs better than NSGA-II as soon as we exceed 4 objectives. It still remains that the diversity of the compromise solutions is reduced with IBMOLS and should be improved.

5 Conclusion

In this paper, we presented an application of the multiobjective knapsack problem encountered in the structures of the social and medico-social sector. A formal model of the problem has been provided. The efficiency of IBMOLS and its superiority to NSGA-II on a large size problems has been shown. However, the epsilon indicator of IBMOLS does not always maintain naturally the diversity of the population in the objective space. It should be interesting to consider a modified version of IBMOLS or to evaluate the effectiveness of other quality indicators. In [3], the hypervolume contribution indicator has shown a high performance level and outperforms the I_ϵ indicator on different multiobjective combinatorial problems. However, it cannot be applied to the present problem since when the number of objective function is greater than three, the high computational cost of the hypervolume contribution calculation tends to drastically reduce the convergence speed of the algorithm. An interesting idea should be to consider the R2 indicator [11], which can be a good trade-off between a reduced computation cost and an efficient indicator.

Acknowledgments. We are grateful to the reviewers for their useful comments. This work was partially supported by the French Ministry for Research and Education through a CIFRE grant (number 0450/2013). We thank M. Rachid Naitali, the Director of GePI Conseil, for his support.

References

1. Basseur, M., Liefooghe, A., Khoi, L., Burke, E.K.: The efficiency of indicator-based local search for multi-objective combinatorial optimisation problems. J. Heuristics **18**, 263–296 (2012)
2. Basseur, M., Talbi, E., Nebro, A., Alba, E.: Metaheuristics for multiobjective combinatorial optimization problems: review and recent issues. INRIA Research Report 5978 (2006)
3. Basseur, M., Zeng, R.-Q., Hao, J.-K.: Hypervolume-based Multi-objective Local Search. Neural Computing and Applications, pp. 1917–1929. Springer, London (2012)
4. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Trans. Evol. Comput. **6**, 182–197 (2002)
5. Kellerer, H., Pferschy, U., Pisinger, D.: Knapsack Problems. Springer Science & Business Media, Heidelberg (2004)
6. Knowles, J.D., Thiele, L., Zitzler, E.: A Tutorial on the performance assessment of stochastic multiobjective optimizers. Technical report, Computer Engineering and Networks Laboratory (TIK) 214, ETH Zurich (2006)
7. Liefooghe, A., Humeau, J., Mesmoudi, S., Jourdan, L., Talbi, E.-G.: On dominance-based multiobjective local search: design, implementation and experimental analysis on scheduling and traveling salesman problems. J. Heuristics **18**, 317–352 (2012)
8. Lust, T., Teghem, J.: The multiobjective multidimensional knapsack problem: a survey and a new approach. Int. Trans. Oper. Res. **19**, 495–520 (2012)

9. Paquete, L., Chiarandini, M., Sttzle, T.: Pareto local optimum sets in the biobjective traveling salesman problem: an experimental study. In: Gandibleux, X., Sevaux, M., Sörensen, K., T'kindt, V. (eds.) Metaheuristics for Multiobjective Optimisation, pp. 177–199. Springer, Heidelberg (2004)
10. The ministry of social affairs and health: Le champ social et médico-social: une activité en forte croissance, des métiers qui se développent et se diversifient. Repères Anal. 44 (2012)
11. Trautmann, H., Wagner, T., Brockhoff, D.: R2-EMOA: focused multiobjective search using R2-indicator-based selection. In: Nicosia, G., Pardalos, P. (eds.) LION 7. LNCS, vol. 7997, pp. 70–74. Springer, Heidelberg (2013)
12. Zitzler, E., Künzli, S.: Indicator-based selection in multiobjective search. In: Yao, X., Burke, E.K., Lozano, J.A., Smith, J., Merelo-Guervós, J.J., Bullinaria, J.A., Rowe, J.E., Tiňo, P., Kabán, A., Schwefel, H.-P. (eds.) PPSN 2004. LNCS, vol. 3242, pp. 832–842. Springer, Heidelberg (2004)