A general and practical dichotomy method for applied multiobjective problems

Samuel Buchet[1], Simon de Givry[1]

Univ Toulouse, INRAE, UR 875 MIAT, Castanet-Tolosan, France

{samuel.buchet, simon.de-givry}@inrae.fr

Keywords : *combinatorial optimization, multi-objective optimization, scalarization.*

Practical applications of combinatorial optimization problems often exhibit more than one objective (*e.g.*, environmental costs), which can be addressed by multi-objective optimization. A common query in multiobjective optimization is the enumeration of the Pareto front which is defined as the complete set of non-dominated solutions, *i.e.*, solutions such that there exists none simultaneously better on all the objectives. Unfortunately, multi-objective problems are commonly considered theoretically and practically more challenging than the single-objective ones and exact enumeration is often intractable. As an alternative, the linear scalarization is a simple and popular approach consisting in aggregating the objectives into a single one with a weighted sum. This general approach can be easily implemented with single-objective solvers to enumerate a subset of the Pareto front of the problem by computing the supported extreme solutions: solutions that are located on the convex hull of the Pareto front in the objective space. Although incomplete, this method can be seen as a viable option to easily obtain diverse solutions on a broad range of multi-objective problems and is also employed as part of complete algorithms (*e.g.*, two-phase methods). The main challenge when implementing this approach is to ensure completeness of the enumeration with a reduced set of weights. The dichotomy method has been proposed in [1] to address this challenge in the bi-objective case with a minimalist algorithm that can be easily implemented. Since then, several algorithmic and software contributions have been proposed to provide a generalization to any number of objectives [4, 5, 3, 2]. Despite these efforts, applying the linear scalarization with more than two objectives remains difficult in practice since no all-purpose implementation is currently available to our knowledge, the existing ones being limited to few problems and solvers. We propose an algorithmic and a software contribution to address this current limitation.

Formally, we consider an objective vector function F of dimension d over a discrete or continuous solution space $\mathcal{X}$ and we consider the minimization of a weighted sum of the functions, *i.e.*, $min \quad \langle w, F(x) \rangle \ s.t. \ x \in \mathcal{X}$ where $w \in \mathbb{R}^D_{\geq 0}$ is a positive weight vector and $\langle w, x \rangle$ corresponds to the dot product (weighted sum) between the weight vector w and the objective cost vector $F(x)$. Our method builds upon the algorithm proposed by [5] which operates on a convex set of normalized weights $W^0 = \{w \in \mathbb{R}^d : \sum_{i=1}^d w_i = 1\}$, sufficient for generating all supported extreme solutions. The enumeration rely on a decomposition of W^0 into subpolytopes, each associated to one of the known supported solution y^i : $W^0(y^i) = \{w \in W^0 : \langle w, y^i \rangle = min \langle w, F(x) \rangle\}$ where y^i designates the objective cost vector of a supported solution. Fig 1:(a) illustrates such decomposition for a 3-objective linear assignment problem with a subset of 2 solutions $\{y^0, y^1\}$, each one corresponding to a polytope of weights from which the solution is optimal for the weighted sum (taken from Example 3 in [5]).

Our computation procedure performs an iterative enumeration which is represented in Fig. 1. Starting from a set of two solutions $\{y^0, y^1\}$ in the example, a third solution y^2 is discovered after minimizing the weighted sum with the weight vector w^2 and then a fourth solution is found by optimizing with w^5. Each time a new solution is discovered, the decomposition of the

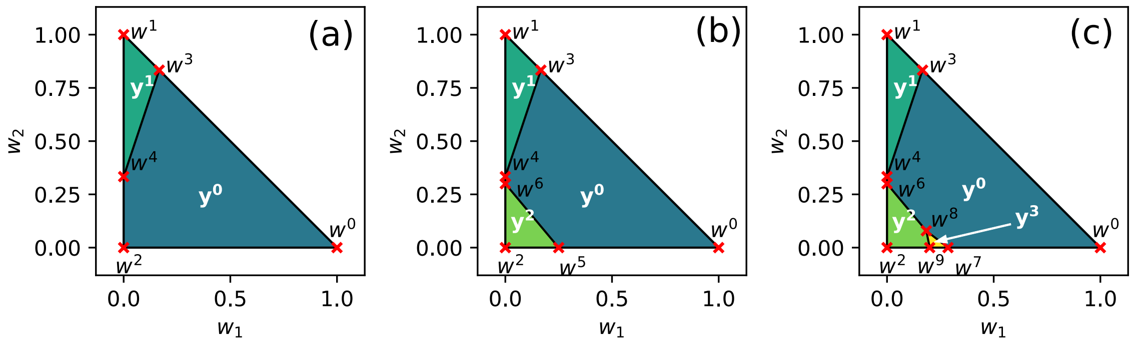

FIG. 1: Update procedure of our algorithm. (a) Decomposition of the weight set with 5 weights vectors w^0, ... , w^4, obtained from two solutions $\{y^0, y^1\}$. (b) A new solution y^2 is discovered by optimizing with w^2 and two new extreme weights w^5 and w^6 are obtained via polytope enumeration. (c) A fourth solution y^3 is discovered from w^5. The update then removes w^5 and adds w^7, w^8 and w^9.

weight set is updated accordingly via polytope enumeration. This procedure similar to [4, 3, 2] provide completeness with a limited set of weight vectors. We also propose an efficient update procedure of the weight set decomposition by maintaining a precise description of weights-to-solutions adjacency. For example in Fig. 1:(c), the solution y^1 is not necessary in the definition of $W^0(y^3)$ because w^5 is not an extreme point of $W^0(y^1)$. We also argue that a lexicographic solver is not required for solving the weighted sum as opposed to what is stated in [5] which broaden the set of single-objective solvers compatible with our algorithm.

Practically, our software contribution is a library that is solver and problem independent. Concretely, the user defines the call to their single-objective solver in a callback function and they are able to run the method to enumerate the supported extreme solutions. Our proof of concept is implemented in C++ and Python and is available at `https://forge.inrae.fr/opteam/generaldichotomy`. It relies on polytope computation libraries for the decomposition of the weights[1]. The multi-language interface facilitates its use with modeling languages (*e.g.*, CPMPY[2], etc.) and solvers (*e.g.*, toulbar2[3], etc.).

References

[1] Y. P. Aneja and K. P. K. Nair. Bicriteria transportation problem. *Management Science*, 25(1):73–78, 1979.

[2] Fritz Bökler, Levin Nemesch, and Mirko H. Wagner. Pamilo: A solver for multi-objective mixed integer linear optimization and beyond. In Oliver Grothe, Stefan Nickel, Steffen Rebennack, and Oliver Stein, editors, *Operations Research Proceedings 2022*, pages 163–170, Cham, 2023. Springer International Publishing.

[3] Ralf Borndörfer, Sebastian Schenker, Martin Skutella, and Timo Strunk. Polyscip. In Gert-Martin Greuel, Thorsten Koch, Peter Paule, and Andrew Sommese, editors, *Mathematical Software – ICMS 2016*, pages 259–264, Cham, 2016. Springer International Publishing.

[4] Özgür Özpeynirci and Murat Köksalan. An exact algorithm for finding extreme supported nondominated points of multiobjective mixed integer programs. *Management Science*, 56(12):2302–2315, 2010.

[5] Anthony Przybylski, Xavier Gandibleux, and Matthias Ehrgott. A recursive algorithm for finding all nondominated extreme points in the outcome set of a multiobjective integer programme. *INFORMS Journal on Computing*, 22(3):371–386, 2010.

[1]CDD `https://github.com/cddlib/cddlib` and PPL `https://github.com/BUGSENG/PPL`
[2]CPMpy: `https://github.com/CPMpy/cpmpy`
[3]ToulBar2: `https://miat.inrae.fr/toulbar2/`