Extended formulations for the min-max-min problem with few recourse solutions

Marco Silva, Ayse Nur Arslan and Michael Poss

January 29, 2020

Extended formulations for the min-max-min problem with few recourse solutions

Marco Silva[1], Ayşe N. Arslan[2], Michael Poss[3]

[1] CEGI, INESCTEC, Porto, Portugal
marco.c.silva@inesctec.pt
[2] IRMAR, INSA de Rennes, Rennes, France
ayse-nur.arslan@insa-rennes.fr
[3] LIRMM, University of Montpellier, CNRS, France
michael.poss@lirmm.fr

Mots-clés : *mathematical programming, robust optimization, min-max-min, decomposition.*

1 Introduction

In this work, we consider the min-max-min problem formalized as

$$\min_{y^k \in Y, k\in[K]} \max_{\xi\in\Xi} \min_{k\in[K]} g(y^k, \xi) \tag{1}$$

where $[K] = \{1, \ldots, K\}$, $\Xi \subseteq \mathbb{R}^n$ is a polyhedral set, $g : Y \times \Xi \to \mathbb{R}$ is a function concave in $\xi \in \Xi$, and $Y \subseteq \mathbb{Z}^n$ is a finite set. Problem (1) models the situation where the decision maker can prepare the ground for K recourse solutions and choose the best of them upon full knowledge of the uncertain parameters. For instance, if Y contains paths from s to t in a given graph, (1) seeks to prepare K different routes that can be used to evacuate citizens or transport relief supplies in case of a hazardous event [5]. In this work, we propose an extended formulation for the min-max-min problem described in (1) and propose a solution method based on decomposition.

2 Literature review

While several studies (e.g., [5, 6]) have illustrated the practical relevance of problem (1), exact solution algorithms have stayed behind. Two general algorithms have been proposed : [5] reformulates the problem through a Mixed-Integer Linear Programming (MILP) formulation involving big-M, and [6] introduces an ad-hoc branch-and-bound algorithm based on generating a relevant subset of scenarios $\Xi' \subseteq \Xi$ and enumerating over their assignment to the K solutions. Unfortunately, these two approaches can hardly solve the shortest path instances proposed by [5] with more than 25 nodes. The approach proposed in [3] had more success with these instances, solving all of them to optimality (up to 50 nodes) in the special case $K = 2$. Yet this latter approach requires g to be linear, Ξ to have a special structure and does not scale up with K. The purpose of this work is to propose a more general algorithm for solving problem (1) to near optimality. To this end, we model problem (1) as a variant of the p-center problem, assigning a *relevant* subset of scenarios to at most K different solutions from Y. We solve the resulting problem by combining a row-and-column generation algorithm, binary search, preprocessing and efficient dominance rules.

3 Methodological development and algorithms

We first propose an extended formulation for a relaxation of problem (1). To this end, let $Y = \{y_1, \ldots, y_r\}$ and $\Xi' = \{\xi_1, \ldots, \xi_t\} \subset \Xi$. We use the notation $[r] = \{1, \ldots, r\}$ and $[t] = \{1, \ldots, t\}$. We introduce binary variables u_s and v_{sj} for $s \in [r]$ and $j \in [t]$, the former being equal 1 if and only if solution s is used, while the latter takes value 1 if and only if solution s is assigned to scenario j. We then write,

$$\begin{aligned}
\min \quad & \omega && \text{(2a)}\\
\text{s.t.} \quad & \omega \geq \sum_{s\in[r]} g(y_s, \xi_j) v_{sj}, \quad \forall j \in [t] && \text{(2b)}\\
& \sum_{s\in[r]} v_{sj} = 1, \quad \forall j \in [t] && \text{(2c)}\\
& \sum_{s\in[r]} u_s \leq k, && \text{(2d)}\\
& v_{sj} \leq u_s, \quad \forall j \in [t],\ s \in [r] && \text{(2e)}\\
& u, v \geq 0 \text{ integer.} && \text{(2f)}
\end{aligned}$$

This formulation is equivalent to the vertex $p-$center problem, that can be efficiently solved to optimality using binary search, coupled with a covering formulation and dominance rules [4].

We next present a row-and-column generation approach based on (2), where at each iteration *relevant* scenarios are added to this relaxation. To do so, let the optimal solution of (2) be given by $(\omega^*, \bar{u}, \bar{v})$. Then a separation problem can be written as

$$\begin{aligned}
z^* = \max \quad & -\sum_{s\in[r]} \bar{u}_s \pi_s + \gamma\\
\text{s.t.} \quad & \xi \in \Xi\\
& -\pi_s + \gamma \leq g(y_s, \xi), \quad s \in [r]\\
& \pi \geq 0.
\end{aligned}$$

Let the optimal value of this separation problem be denoted by z^*. If $z^* \geq \omega^*$ a scenario will be added to the formulation (2) by generating a new variable v_{sj} and a new constraint (2b). Otherwise, the optimal solution to (1) is found.

Numerical results showing the promise of this approach compared to the MILP approach of [5] will be presented.

Références

[1] Dimitris Bertsimas and Constantine Caramanis. Finite adaptability in multistage linear optimization. *IEEE Transactions on Automatic Control*, 55(12) :2751–2766, 2010.

[2] Christoph Buchheim and Jannis Kurtz. Min–max–min robust combinatorial optimization. *Mathematical Programming*, 163(1-2) :1–23, 2017.

[3] André Chassein, Marc Goerigk, Jannis Kurtz, and Michael Poss. Faster Algorithms for Min-max-min Robustness for Combinatorial Problems with Budgeted Uncertainty. *European Journal of Operational Research*, 2019.

[4] Claudio Contardo, Manuel Iori, and Raphael Kramer. A scalable exact algorithm for the vertex p-center problem. *Computers & Operations Research*, 103 :211–220, 2019.

[5] Grani A Hanasusanto, Daniel Kuhn, and Wolfram Wiesemann. K-adaptability in two-stage robust binary programming. *Operations Research*, 63(4) :877–891, 2015.

[6] Anirudh Subramanyam, Chrysanthos E Gounaris, and Wolfram Wiesemann. K-Adaptability in Two-Stage Mixed-Integer Robust Optimization. arXiv preprint arXiv :1706.07097, 2017.