General Polyhedral Approximation of Two-Stage Robust Linear Programming

Lukas Grunau[1[0000−0002−5100−7726]], Tim Niemann[1,2,3[0000−0003−4830−1884]], and Sebastian Stiller[1,2[0000−0002−8792−4390]]

[1] Technische Universität Braunschweig, Institute for Mathematical Optimization, Universitätsplatz 2, 38106 Braunschweig, Germany
[2] Cluster of Excellence SE^{2}A - Sustainable and Energy-Efficient Aviation, Hermann-Blenk-Str. 42, 38108 Braunschweig, Germany
[3] tim.niemann@tu-braunschweig.de

Abstract. We consider two-stage robust linear programs with uncertain righthand side. We develop a General Polyhedral Approximation (GPA), in which the uncertainty set $\mathcal{U}$ is substituted by a finite set of polytopes derived from the vertex set of an arbitrary polytope that dominates $\mathcal{U}$. The union of the polytopes need not contain $\mathcal{U}$. We analyse and computationally test the performance of GPA for the frequently used budgeted uncertainty set $\mathcal{U}$ (with m rows). For budgeted uncertainty affine policies are known to be best possible approximations (if coefficients in the constraints are nonnegative for the second-stage decision). In practice calculating affine policies typically requires inhibitive running times. Therefore an approximation of $\mathcal{U}$ by a single simplex has been proposed in the literature. GPA maintains the low practical running times of the simplex based approach while improving the quality of approximation by a constant factor. The generality of our method allows to use any polytope dominating $\mathcal{U}$ (including the simplex). We provide a family of polytopes that allows for a trade-off between running time and approximation factor. The previous simplex based approach reaches a threshold at $\Gamma > \sqrt{m}$ after which it is not better than a quasi nominal solution. Before this threshold, GPA significantly improves the approximation factor. After the threshold, it is the first fast method to outperform the quasi nominal solution. Moreover, GPA allows for even stronger results on specific problems as we exemplify for the Transportation Location Problem.

Keywords: Robust Optimization · Two-Stage Robust Optimization · Linear Programming · Approximation Algorithm · Transportation Location Problem

1 Introduction

In two-stage robust optimization, first, the optimizer chooses a first-stage solution $\boldsymbol{x}$ for which, second, the malign adversary chooses a scenario from a set of scenarios $\mathcal{U}$, and third, the optimizer chooses a second-stage solution $\boldsymbol{y}$. Feasibility and cost of a solution $\boldsymbol{x}$ are determined by the worst scenario specifically chosen for $\boldsymbol{x}$ and with respect to a best possible second-stage decision $\boldsymbol{y}$. When the scenario set $\mathcal{U}$ is a polytope, it suffices in many (convex) settings to consider the vertices of the polytope as scenarios.

We consider linear programs with uncertain righthand side and develop our main result for general uncertainty sets $\mathcal{U}$. We then analyse its performance theoretically and computationally for the widely used uncertainty set introduced in [11,12]. It is restricted in two ways: For each row the uncertain righthand side is restricted to an interval from the nominal value to the nominal value plus the maximal increase. In each scenario the actual increase normalized by the maximal increase and summed over all rows is limited by a budget traditionally denoted by Γ. This so-called budgeted uncertainty set $\mathcal{U}$, see Definition 1, is a polytope whose number of vertices grows exponentially with Γ. Considering each vertex of $\mathcal{U}$ separately is therefore not an efficient approach in most cases. Also, at least naively, the first-stage optimization and the choice of the worst-case scenario need to consider any possible second-stage decision. However, it is known that piecewise-affine-linear policies for the second-stage decision are best possible [3,5]. Still, two-stage robust optimization remains challenging in general and in many special cases. The standard textbook on robust optimization from 2009 considers "two-stage robust optimization rather wishful thinking than an actual tool" [2].

One of the most important approaches to let this wish come true is to approximate an optimal solution by firstly replacing the scenario set $\mathcal{U}$ by a dominating simplex, thus arriving at linearly many vertices, and secondly calculating piecewise-affine-linear second stage policies. This has been proposed in Ben-Tal et al. [3]. While the idea of polyhedral approximation stands to reason, a closer examination shows that for the budget of uncertainty polytope the methods yield results that are even more conservative than the solutions obtained when simply computing a nominal solution to the scenario where all righthand side values have maximal deviation, i.e., a scenario worse than all scenarios in $\mathcal{U}$. This is also the case for the instances with budget of uncertainty considered in the original paper by Ben-Tal et al. [3].

Our Results. We show how the basic idea of polyhedral approximation can work when pursued in a different, more general way: Instead of dominating with the simplex, we develop a method to use an arbitrary polytope dominating any uncertainty set $\mathcal{U}$, given by its vertices, then partition the vertices and calculate separate solutions for the polytopes spanned by the subsets of the dominating polytope. While the union of these polytopes does not in general dominate $\mathcal{U}$, we show a way to combine their solutions to a feasible solution for $\mathcal{U}$. This solution has superior approximation factors. Also, the proofs become somewhat simpler by this more general approach. This General Polyhedral Approximation (GPA) allows to use arbitrary polyhedra. We then analyse the performance of GPA for the budgeted uncertainty set. We show that, in particular, using scaled budgeted uncertainty polytopes with smaller budget to dominate the original set $\mathcal{U}$ gives a constant factor improvement for the approximation factor compared to the literature. Also, GPA allows to calculate instance specific approximation guarantees. Furthermore, we apply the method to the Transportation Location Problem, arriving at an even better approximation factor depending on a parameter that catches how different the values in the input data are. Finally, we will compare our methods numerically against other state-of-the-art approaches and show that we can either still deliver comparatively good solutions with significantly shorter runtimes or achieve better results with the same runtime.

Related Work. It is known, that two-stage robust optimization problems are NP-hard [20] and not approximable within a factor better than $\Omega(\frac{\log m}{\log\log m})$, c.f. [17]. If the constraint matrix of the second-stage decision, also called recourse matrix, is nonnegative, this factor can be matched by affine policies, as shown in [15] and improved more recently in [14]. If there are no such constraints on the first and second stage constraint matrices, El Housni and Goyal [15] show that the gap between the optimal affine policy and the optimal adjustable solution can be arbitrarily large. In the setting of our work, with a nonnegative first stage constraint matrix and a general recourse matrix with possibly negative entries, affine policies have a worst case approximation factor of $3\sqrt{m}$, c.f. Bertsimas and Goyal [7]. Furthermore, the authors show that affine policies are optimal, if the uncertainty set is a simplex. Further details on affine policies can be found in [1, 4, 10, 21].

The approximation factor of $2\sqrt{m}$ with piecewise affine policies by Ben-Tal et al. was recently improved by Grunau [19] and Thomä et al. [24] in the case of budget of uncertainty sets. Bertsimas et al. analyze the quality of static solutions for the robust problem [8, 9] and for the stochastic variant [6].

Other approaches consider the dualization of the inner minimization problem [18], which results in a nonlinear problem or column and constraint generation to potentially reduce the size of the problem. [22, 23, 25]. If the decision process is divided in more than two stages, there are algorithms considering finite adaptability [5] and approximation methods [24] for the multistage problem. Special cases of the two-stage robust optimization problem are also studied in the literature, e.g., the transportation location problem [18], which we will later discuss. El Housni et al. [16] consider a variant of this problem with unit demand, which they call soft-capacitated robust facility location. They prove that static assignment policies for the second stage decision give a $O(\frac{\log \Gamma}{\log\log \Gamma})$-approximation when the demand per client is binary and the distance costs are metric.

Notation We indicate all vectors $\boldsymbol{y}_{ij}$ and matrices $\boldsymbol{A}$ in bold type. Entries of a vector $\boldsymbol{w}$ are written in normal font w_j. Sets $\mathcal{U}$ are shown in italics. The nonnegative real numbers are written as $\mathbb{R}_+$ and the positive part of a number is written as $(h)^+ = \max\{h, 0\}$. With $\mathbf{0}$ and $\mathbf{1}$, we denote vectors containing only ones and zeros respectively and $\boldsymbol{e}_j$ denotes the j-th unit vector. The convex hull of a set $\mathcal{T}$ is written as $\operatorname{conv}\mathcal{T}$.

2 General Polyhedral Approximation (GPA)

This chapter develops the main result, which is an approximation method for two-stage robust optimization. It allows to use an arbitrary polytope to dominate the uncertainty set. We partition the set of vertices of the dominating polytope into subsets, each of which spans a polytope. The key insight is that calculating solutions for each of these polytopes and combining them appropriately gives an approximation.

We consider the following two-stage robust linear problem:

$$\begin{aligned} z_{AR}(\mathcal{U}) = \min_{\boldsymbol{x}} \ & \boldsymbol{c}^T\boldsymbol{x} + \max_{\boldsymbol{h}\in\mathcal{U}} \min_{\boldsymbol{y}(\boldsymbol{h})} \boldsymbol{d}^T\boldsymbol{y}(\boldsymbol{h}) \\ \text{s.t. } & \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{y}(\boldsymbol{h}) \geq \boldsymbol{h} \qquad \forall \boldsymbol{h}\in\mathcal{U} \\ & \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{y}(\boldsymbol{h}) \geq \boldsymbol{b} \qquad \forall \boldsymbol{h}\in\mathcal{U} \\ & \boldsymbol{x}\in\mathbb{R}_+^{n_1},\ \boldsymbol{y}(\boldsymbol{h})\in\mathbb{R}_+^{n_2} \end{aligned} \tag{1}$$

where $\boldsymbol{A}\in\mathbb{R}_+^{m\times n_1}$, $\boldsymbol{B}\in\mathbb{R}^{m\times n_2}$, $\boldsymbol{C}\in\mathbb{R}_+^{p\times n_1}$, $\boldsymbol{D}\in\mathbb{R}^{p\times n_2}$, $\boldsymbol{c}\in\mathbb{R}_+^{n_1}$, $\boldsymbol{d}\in\mathbb{R}_+^{n_2}$ and $\boldsymbol{b}\in\mathbb{R}_+^{p}$. With these restrictions, it is possible to model, e.g., set covering, capacity planning, facility location, Steiner trees, network design and transportation location problems. Although this model is quite general, the nonnegativity of the right-hand side does not allow packing constraints and upper bounds on variables.

We say, a set $\mathcal{V}\subseteq\mathbb{R}^m$ *dominates* the scenario set $\mathcal{U}\subseteq\mathbb{R}^m$ iff for all $\boldsymbol{h}\in\mathcal{U}$, there exists $\hat{\boldsymbol{h}}\in\mathcal{V}$ such that $\boldsymbol{h}\leq\hat{\boldsymbol{h}}$ component-wise. We assume that the dominating set $\mathcal{V}$ is a polytope given by a finite set of points vertices $\boldsymbol{v}_{ij}\in\mathbb{R}^m$. In this paper the $\boldsymbol{v}_{ij}$ will already be the vertices of $\mathcal{V}$. Double indices indicate a partition in N smaller sets, defining the polytopes $\mathcal{V}_i$.

$$\mathcal{V} = \operatorname{conv}\{\boldsymbol{v}_{ij} \mid i\in[N], j\in[M_i]\}, \qquad \mathcal{V}_i = \operatorname{conv}\{\boldsymbol{v}_{ij} \mid j\in[M_i]\}. \tag{2}$$

Note that the union of all $\mathcal{V}_i$ does not necessary dominates $\mathcal{U}$. We assume for a scenario set $\mathcal{U}$, that it is a convex, full-dimensional subset of the unit cube that contains all unit vectors and is *down-monotone*: $\boldsymbol{h}\in\mathcal{U}$ and $\boldsymbol{0}\leq\boldsymbol{h'}\leq\boldsymbol{h}$ implies $\boldsymbol{h'}\in\mathcal{U}$.

It is easy to extend the proof of the following Lemma by Ben-Tal et al. [3] to our more general formulation (1).

Lemma 1 (Ben-Tal et al. [3]). *Let $\mathcal{U}$ be an uncertainty set and $\mathcal{V}$ a dominating set of $\mathcal{U}$ with $\beta\geq 1$ and $\mathcal{V}\subseteq\beta\,\mathcal{U}:=\{\beta\boldsymbol{h}\mid\boldsymbol{h}\in\mathcal{U}\}$. Let $z_{AR}(\mathcal{U}), z_{AR}(\mathcal{V})$ be the optimal values for* (1) *corresponding to $\mathcal{U}$ and $\mathcal{V}$, respectively. The following inequalities hold:*

$$z_{AR}(\mathcal{U}) \leq z_{AR}(\mathcal{V}) \leq \beta\, z_{AR}(\mathcal{U}) \tag{3}$$

We use an arbitrary polytope $\mathcal{V}$ dominating $\mathcal{U}$. We use a general dominating polytope $\mathcal{V}$ instead of a simplex. We compute N sets of first-stage variables $\boldsymbol{x}_i$, from which we construct the first stage decision $\hat{\boldsymbol{x}}$. To compute the $\boldsymbol{x}_i$, we need to solve N linear programs, that are of the same type as (1) but lesser size:

$$\begin{aligned} z_{AR}(\mathcal{V}_i) = \min\ & \boldsymbol{c}^T\boldsymbol{x}_i + w && \\ \text{s.t. } & w \geq \boldsymbol{d}^T\boldsymbol{y}_{ij}, && j \in [M_i], \\ & \boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{B}\boldsymbol{y}_{ij} \geq \boldsymbol{v}_{ij}, && j \in [M_i], \\ & \boldsymbol{C}\boldsymbol{x}_i + \boldsymbol{D}\boldsymbol{y}_{ij} \geq \boldsymbol{b}, && j \in [M_i], \\ & \boldsymbol{x}_i \in \mathbb{R}^{n_1}_+,\ \boldsymbol{y}_{ij} \in \mathbb{R}^{n_2}_+, && j \in [M_i]. \end{aligned} \tag{4}$$

Next we choose functions $\alpha_{ij}(\cdot) : \mathcal{U} \to [0,1]$ with the following two properties:

$$\sum_{i=1}^{N}\sum_{j=1}^{M_i} \alpha_{ij}(\boldsymbol{h})\boldsymbol{v}_{ij} \geq \boldsymbol{h},\ \forall \boldsymbol{h} \in \mathcal{U}, \qquad \sum_{i=1}^{N}\sum_{j=1}^{M_i} \alpha_{ij}(\boldsymbol{h}) \geq 1,\ \forall \boldsymbol{h} \in \mathcal{U}. \tag{5}$$

We will later discuss possibilities for choosing $\alpha_{ij}(\cdot)$. Let $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, $i \in [N]$ be the N solutions of (4). We will show, that for any choice of $\alpha_{ij}(\cdot)$ respecting the conditions (5) above, $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}(\cdot))$, defined as follows, is a feasible solution to (1).

$$\hat{\boldsymbol{x}} = \sum_{i=1}^{N} \sigma_i \boldsymbol{x}_i, \qquad \hat{\boldsymbol{y}} : \mathcal{U} \to \mathbb{R}^{n_2}_+,\ \hat{\boldsymbol{y}}(\boldsymbol{h}) = \sum_{i=1}^{N}\sum_{j=1}^{M_i} \alpha_{ij}(\boldsymbol{h})\boldsymbol{y}_{ij}, \tag{6}$$

The σ_i are defined as $\sigma_i \coloneqq \max_{\boldsymbol{h}\in\mathcal{U}} \sum_{j=1}^{M_i} \alpha_{ij}(\boldsymbol{h})$. We can limit the unnecessary use of the first stage variables $\boldsymbol{x}_i$ in the final mapping by computing this factor. Note, that in the work of Ben-Tal et al. [3], they use $N = 1$ and $\sigma_1 = 2$, whereas in our case the σ_i are chosen as small as possible to guarantee feasibility for each scenario $\boldsymbol{h} \in \mathcal{U}$. Now we show that this gives a feasible solution for the original problem.

Lemma 2. *The pair $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}(\cdot))$ defined in* (6) *is a feasible solution for* (1)*, if $\sum_{i=1}^{N}\sum_{j=1}^{M_i} \alpha_{ij}(\boldsymbol{h})\boldsymbol{v}_{ij} \geq \boldsymbol{h}$, $\forall \boldsymbol{h} \in \mathcal{U}$ and $\sum_{i=1}^{N}\sum_{j=1}^{M_i} \alpha_{ij}(\boldsymbol{h}) \geq 1$.*

Proof. For a fixed scenario $\boldsymbol{h} \in \mathcal{U}$, feasibility follows from the following calculations.

$$\begin{aligned} \boldsymbol{A}\hat{\boldsymbol{x}} + \boldsymbol{B}\hat{\boldsymbol{y}}(\boldsymbol{h}) &= \boldsymbol{A}\sum_{i=1}^{N}\sigma_i\boldsymbol{x}_i + \boldsymbol{B}\sum_{i=1}^{N}\sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{y}_{ij} \\ &= \sum_{i=1}^{N}(\sigma_i\boldsymbol{A}\boldsymbol{x}_i + \sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{B}\boldsymbol{y}_{ij}) \\ &\geq \sum_{i=1}^{N}\sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\left(\boldsymbol{A}\boldsymbol{x}_i + \boldsymbol{B}\boldsymbol{y}_{ij}\right) \geq \sum_{i=1}^{N}\sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{v}_{ij} \geq \boldsymbol{h}. \end{aligned}$$

$$\begin{aligned}
\boldsymbol{C}\hat{\boldsymbol{x}} + \boldsymbol{D}\hat{\boldsymbol{y}}(\boldsymbol{h}) &= \boldsymbol{C}\sum_{i=1}^{N}\sigma_i\boldsymbol{x}_i + \boldsymbol{D}\sum_{i=1}^{N}\sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{y}_{ij} \\
&= \sum_{i=1}^{N}(\sigma_i\boldsymbol{C}\boldsymbol{x}_i + \sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{D}\boldsymbol{y}_{ij}) \\
&\geq \sum_{i=1}^{N}\sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\left(\boldsymbol{C}\boldsymbol{x}_i + \boldsymbol{D}\boldsymbol{y}_{ij}\right) \geq \sum_{i=1}^{N}\sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{b} \geq \boldsymbol{b}.
\end{aligned}$$

After proving the feasibility of the solution, we can calculate the mentioned approximation factor.

Theorem 1. *The pair* $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}(\cdot))$ *as mentioned in the previous stated Lemma 2 is a* $\left(\beta\sum_{i=1}^{N}\sigma_i\right)$*-approximation to the original problem* (1).

Proof. The cost of $(\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}(\cdot))$ for any fixed scenario $\boldsymbol{h} \in \mathcal{U}$ is

$$\begin{aligned}
&\boldsymbol{c}^T\hat{\boldsymbol{x}} + \boldsymbol{d}^T\hat{\boldsymbol{y}}(\boldsymbol{h}) = \boldsymbol{c}^T\left(\sum_{i=1}^{N}\sigma_i\boldsymbol{x}_i\right) + \boldsymbol{d}^T\left(\sum_{i=1}^{N}\sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{y}_{ij}\right) \\
&= \sum_{i=1}^{N}\left(\sigma_i\boldsymbol{c}^T\boldsymbol{x}_i + \sum_{j=1}^{M_i}\alpha_{ij}(\boldsymbol{h})\boldsymbol{d}^T\boldsymbol{y}_{ij}\right) \leq \sum_{i=1}^{N}\sigma_i\left(\boldsymbol{c}^T\boldsymbol{x}_i + \max_{j\in[M_i]}\boldsymbol{d}^T\boldsymbol{y}_{ij}\right) \\
&= \sum_{i=1}^{N}\sigma_i z_{AR}(\mathcal{V}_i) \leq \left(\sum_{i=1}^{N}\sigma_i\right)z_{AR}(\mathcal{V}) \leq \left(\beta\sum_{i=1}^{N}\sigma_i\right)z_{AR}(\mathcal{U}).
\end{aligned}$$

In the last line, we used the result of Lemma 1. As this bound holds for any scenario $\boldsymbol{h} \in \mathcal{U}$, the worst-case cost of our solution is at most a factor $\beta\sum_{i=1}^{N}\sigma_i$ of the worst-case cost of the optimal solution $z_{AR}(U)$, i.e., our theorem holds.

In the following subsections, we will give concrete examples of these policies and compare the resulting approximation factors. Note that a compromise between quality and runtime must always be found for all approximation methods. Although the results and proofs in this chapter rely on the specific two-stage problem (1), the general idea of using multiple polytopes is applicable whenever a domination approach can be used to get an approximation for the original problem.

Remark 1 (A Posteriori Approximation Factor). In the proof of Theorem 1, we in particular overestimated $z_{AR}(\mathcal{V}_i)$ with $z_{AR}(\mathcal{V})$. For a given instance, we can thus enhance the analysis using the exact values $z_{AR}(\mathcal{V}_i)$ if $N > 1$. This a posteriori approximation factor is given by

$$\frac{\sum_{i=1}^{N}\sigma_i\; z_{AR}(\mathcal{V}_i)}{\max_{i\in N}\left(z_{AR}(\mathcal{V}_i)\right)}\beta. \tag{7}$$

We will see the affect of this in chapter 7 in Figure 4.

For the examples given in the rest of the paper, we use the following as a scenario set $\mathcal{U}$.

Definition 1. *The budget of uncertainty set [11, 12] for a robustness parameter* $\Gamma > 0$ *is defined as*

$$\mathcal{U} := \mathcal{U}_\Gamma := \left\{ \boldsymbol{h} \in [0,1]^m \,\middle|\, \sum_{j=1}^{m} h_j \leq \Gamma \right\}. \tag{8}$$

3 Scaling the Budget of Uncertainty Polytope

The number of vertices of the budget of uncertainty polytope grows exponentially with Γ. The idea in this section is that dominating $\mathcal{U}_\Gamma$ with $\beta\,\mathcal{U}_{\Gamma/\gamma}$ could be computationally advantageous and less inaccurate than dominating with a simplex. We now show how to choose β and γ.

Definition 2 (Scaled Budget Policy). *Let* $\mathcal{U}_\Lambda$ *be the uncertainty set for a smaller robustness parameter* $\Lambda \leq \Gamma$. *With* $N = 1, M_1 = \binom{m}{\Lambda}, \sigma_1 = 1, \beta = \frac{\Gamma}{\Lambda}$ *and*

$$\mathcal{V} = \mathcal{V}_1 = \beta\,\mathcal{U}_\Lambda = conv \left\{ \boldsymbol{u} \in \{0, \beta\}^m \,\middle|\, \sum_{i=1}^{m} u_i = \Gamma \right\} = \{\boldsymbol{v}_1, \ldots \boldsymbol{v}_{M_1}\}, \tag{9}$$

we can define the scaled budget policy with the coefficients

$$\alpha_{1j}(\boldsymbol{h}) = \begin{cases} \binom{\Gamma}{\Lambda}^{-1}, & \text{if } \boldsymbol{h} \geq \frac{\boldsymbol{v}_j}{\beta}, \\ 0, & \text{else.} \end{cases} \tag{10}$$

With this definition, Theorem 1 yields the following corollary. We can achieve a constant factor improvement in the approximation bound by increasing the number of vertices of the dominating polytope.

Corollary 1. *The scaled budget policy gives a* $\beta = \frac{\Gamma}{\Lambda}$ *approximation.*

Proof. The feasibility and the approximation factor follow directly after checking the constraints of Lemma 2.

This polytope has at most m^Λ vertices and we will test it in practice for $\Lambda = 1$ with m vertices and $\Lambda = 2$ with $\frac{m(m-1)}{2}$ vertices. For this policy, there is a trade-off between quality of the solution and effort to find it: With more vertices and therefore larger run time to compute the solution of (4), we can get a better approximation factor of the policy. We will later see the effects of this in Figure 1 and Figure 5.

4 The Power of Combining Polytopes

GPA allows to substitute a set $\mathcal{V}$ dominating $\mathcal{U}$ by several sets $\mathcal{V}_i$, of which the union in general does not dominate $\mathcal{U}$. Theorem 1 shows that this gives feasible solutions with an a priori approximation factor. In this section we give a simple example that this split improves the a posteriori approximation factor from Remark 1. We use the original dominating set by Ben-Tal et al. [3], the simplex, but split the vertices into two sets; one of which only contains the vertex of the average case. With this small modification, we achieve an improvement in the approximation bound by up to a factor of 2.

Definition 3 (Combination policy). *Let* $\boldsymbol{w} = \frac{\Gamma}{m}\mathbf{1}$ *be the average scenario with an uniformly distributed budget. Then we define the combination policy with* $N = 2, M_1 = m, M_2 = 1$ *as follows:*

$$\mathcal{V}_1 = \beta\ conv\{\boldsymbol{e}_1, \dots, \boldsymbol{e}_m\}, \qquad \mathcal{V}_2 = \{\beta \boldsymbol{w}\}, \tag{11}$$

$$\alpha_{1j}(\boldsymbol{h}) = \frac{1}{\beta}(h_j - \beta w_j)^+, \qquad \alpha_{21}(h) = 1, \tag{12}$$

$$\sigma_1 = \frac{m\Gamma - \beta\Gamma^2}{\beta m} \leq 1, \qquad \sigma_2 = 1. \tag{13}$$

With this definition, again, Theorem 1 yields the following corollary.

Corollary 2. *The combination policy gives a* $\left(1 - \frac{\Gamma^2}{m}\right)\beta + \Gamma$ *approximation.*

Proof. For a budgeted uncertainty set with $\boldsymbol{w} = \frac{\Gamma}{m}\mathbf{1}$, we know that

$$\sigma_1 = \max_{\boldsymbol{h}\in\mathcal{U}} \sum_{j=1}^{M_1} \alpha_{1j}(\boldsymbol{h}) = \frac{\Gamma}{\beta}\left(1 - \beta\frac{\Gamma}{m}\right) = \frac{m\Gamma - \beta\Gamma^2}{\beta m}. \tag{14}$$

Inserting this into the approximation factor from Theorem 1, we get

$$(\sigma_1 + \sigma_2)\beta = \left(\frac{m\Gamma - \beta\Gamma^2}{\beta m} + 1\right)\beta = \left(1 - \frac{\Gamma^2}{m}\right)\beta + \Gamma. \tag{15}$$

The following Lemma gives an optimal choice for beta.

Lemma 3. *The optimal scaling factor* β_{opt} *dependent on the budget of uncertainty* Γ *for the combination policy is given by:*

$$\beta_{opt} = \begin{cases} \beta^* = \max\left(1, \dfrac{m\Gamma}{\Gamma^2 + m}\right), & \text{if } \Gamma \leq \sqrt{m}, \\ \beta' = \dfrac{m}{\Gamma}, & \text{else.} \end{cases} \tag{16}$$

Proof. The approximation factor of Corollary 2 is an affine linear function in β, so the minimum is dependent on the sign of the gradient. In other words, we need to find the minimal feasible β, if $\left(1 - \frac{\Gamma^2}{m}\right)$ is positive and the maximal feasible β if the term is negative. If the gradient is zero, every feasible β with $\beta \boldsymbol{w} \leq \mathbf{1}$ is optimal. For a positive gradient with $\Gamma < \sqrt{m}$, Ben-Tal et al. [3] already showed that the optimal scaling factor is $\beta^* = \max\left(1, \frac{m\Gamma}{\Gamma^2 + m}\right)$, under the condition that $\sigma_1 \equiv 1$. If the gradient is negative in the range of $\Gamma > \sqrt{m}$, we have to choose β as large as possible but feasible. So $\beta \boldsymbol{w} = \mathbf{1}$ directly gives $\beta' = \frac{m}{\Gamma}$ with $\sigma_1 \equiv 0$.

In the next chapter in Figure 1 the effect of this choice of β will be demonstrated. In a nutshell, for small Γ this factor is strictly better while for larger Γ the effect vanishes as a nominal solution for all righthand values at maximum deviation becomes the best known robust solution anyway. For a similar observation concerning β^* see [19, 24].

5 Choosing The Best Method

The different approximation factors of methods discussed in this paper all depend on Γ, i.e., the number of rows with maximally increased righthand side and m the total number of rows. The order of magnitude of the approximation factors for all presented methods is either $\frac{m}{\Gamma}$, Γ or a combination of both. The differences of the presented methods are due to some constant factors and are not often discussed in the literature. Nonetheless even constant factor improvements give better lower bounds and in our case better solutions, which is indeed relevant for a practical application of these methods. Here we plot the approximation factors for comparison with $m = 100$.

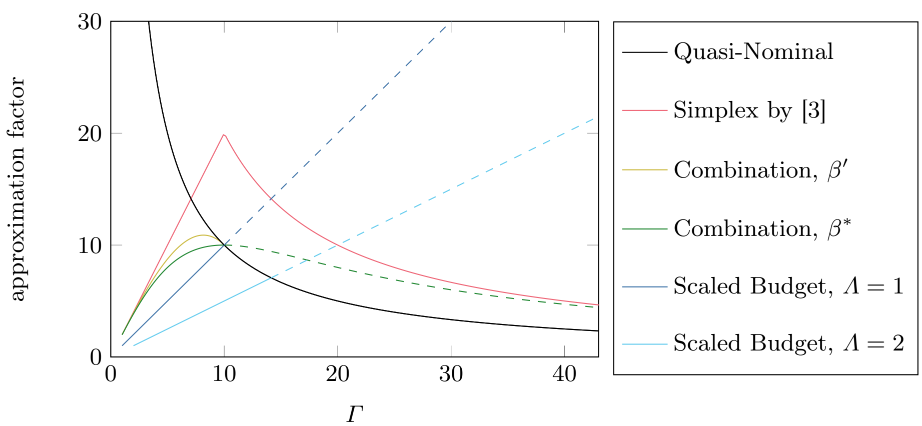

Fig. 1. Approximation factors for $m = 100$

In Figure 1 we consider the following methods:

- Quasi-nominal ($\frac{m}{\Gamma}$): Solving for the single scenario, in which all righthand side values are set to maximum, trivially dominates $\mathcal{U}$ and is computationally equivalent to a nominal problem. The approximation factor of this simplest, quasi-nominal approach is shown as the black line.
- Simplex ($2\min(\Gamma, \frac{m}{\Gamma})$): Dominating with the simplex as proposed in [3] is shown in red.
- Combination policy ($(1 - \frac{\Gamma^2}{m})\beta + \Gamma$): Lemma 3 uses the two approximation factor β' and β^*. It is easy to show that after intersection of these factors β' is equal to the quasi-nominal solution shown in black.
- Scaled budget policy ($\frac{\Gamma}{\Lambda}$): We depict the factor of the scaled budget policies for two budgets Λ. The factors are linear and can be truncated when they are worse than the factor of the quasi-nominal solution, because the intersection can be calculated a priori.

For large Γ the quasi-nominal solution performs best. For small Γ the scaled budget policy is best. The threshold between small and large Γ is at $\sqrt{m}$ for all policies except for more complex scaled budget polytopes, i.e., $\Lambda \geq 2$. Increasing Λ in turn exponentially increases the size of the linear programs to be solved. Therefore, the choice of the best method depends on m, Γ, and the computational resources.

6 Transportation Location Problem and an Input Dependent A Priori Approximation Factor

The original approximation method by Ben-Tal et al. [3] uses a single first-stage solution for all vertices, while GPA calculates and combines different solutions for sets of vertices. As first-stage solutions suitable e.g. for the average case vertex of the simplex in many cases differ strongly from those for the other simplex vertices there is room for improvement of the a priori approximation factor under mild conditions, which are fulfilled naturally in many problem classes.

We show in this chapter for the Transportation Location Problem (TLP) a stronger approximation factor that depends on a term capturing the difference in magnitude for the cost and demand coefficients. The analysis draws in particular from the sparsity of the matrices of the TLP.

In the TLP we are given a complete bipartite graph and have to store resources on the left vertices $f \in [n]$ to meet an uncertain demand on the right at minimal storage and worst-case transportation cost. In contrast to El Housni et al. [16], we will not require that the demand per client is binary and that the transportation costs are metric. We assume that each client $k \in [m]$ has a maximum demand H_k and that not more than Γ clients will request their demand or - leading to equivalent solutions - the normalized demand $\boldsymbol{h} \in [0,1]^m$ lies inside the budget of uncertainty set $\mathcal{U}$. Formally, this can be expressed by the following optimization problem:

$$\begin{aligned} z_{TLP}(\mathcal{U}) = &\min_{\boldsymbol{x}} \ \boldsymbol{c}^T\boldsymbol{x} + \max_{\boldsymbol{h}\in\mathcal{U}} \ \min_{\boldsymbol{y}} \ \boldsymbol{d}^T\boldsymbol{y} \\ \text{s.t.} &\sum_{f=1}^{n} y^{fk} \geq h_k H_k, \ k\in[m], \quad x^f - \sum_{k=1}^{m} y^{fk} \geq 0, \ f\in[n], \quad \boldsymbol{x},\boldsymbol{y}\geq \boldsymbol{0}. \end{aligned} \tag{17}$$

Here x^f corresponds to the amount of commodities stored in facility f and y^{fk} is the quantity that gets transported from facility f to client k. The storage costs are contained in the vector $\boldsymbol{c}$, d_{fk} is the cost of transporting one unit of the commodity from facility f to client k, h_k is the demand of client k and $\boldsymbol{h}$ is a vector from the budget of uncertainty set $\mathcal{U}$ from (8). Both variables, $\boldsymbol{x}$ and $\boldsymbol{y}$, are continuous. To meet the notation of the previous chapters, one can reformulate problem (17) to:

$$\begin{aligned} z_{TLP}(\mathcal{U}) = &\min_{\boldsymbol{x}} \ \boldsymbol{c}^T\boldsymbol{x} + \max_{\boldsymbol{h}\in\mathcal{U}} \min_{\boldsymbol{y}(\boldsymbol{h})} \boldsymbol{d}^T\boldsymbol{y}(\boldsymbol{h}) \\ \text{s.t. } &\boldsymbol{B}\boldsymbol{y}(\boldsymbol{h}) \geq \boldsymbol{h}, \forall \boldsymbol{h}\in\mathcal{U}, \quad \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{y}(\boldsymbol{h}) \geq \boldsymbol{0}, \ \forall \boldsymbol{h}\in\mathcal{U}, \quad \boldsymbol{x},\boldsymbol{y}\geq\boldsymbol{0}. \end{aligned} \tag{TLP}$$

The first constraint assures demands getting satisfied and the second one prevents us from transporting a larger amount of the commodity from each facility than we have stored there. $\boldsymbol{C}$ is the $n\times n$ identity matrix and the matrices $\boldsymbol{B}\in\mathbb{R}_+^{m\times(n\cdot m)}$ and $\boldsymbol{D}\in\mathbb{R}^{n\times(n\cdot m)}$ are defined by

$$\boldsymbol{B} = \begin{pmatrix} \frac{1}{H_1} & \cdots & \frac{1}{H_1} & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & 0 & \frac{1}{H_2} & \cdots & \frac{1}{H_2} & 0 & \cdots & 0 \\ \vdots & & & & & \ddots & & & \vdots \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & \frac{1}{H_m} & \cdots & \frac{1}{H_m} \end{pmatrix}, \quad \boldsymbol{D} = \begin{pmatrix} -\boldsymbol{I} & -\boldsymbol{I} & \ldots & -\boldsymbol{I} \end{pmatrix}. \tag{18}$$

$\boldsymbol{D}$ consists of the negative of m $n\times n$ identity matrices. $\boldsymbol{B}$ has n and $\boldsymbol{D}$ has m non-zero entries per row and both have only one non-zero entry per column.

Theorem 2. *The combination policy from Definition 3 gives an approximation factor of*

$$\frac{\max_{f\in[n],k\in[m]} c_f + d_{fk}}{\min_{f\in[n],k\in[m]} c_f + d_{fk}} \frac{\sigma_1 H_{max} + \Gamma\overline{H}}{\max\left(H_{max}, \ \Gamma\overline{H}\right)}\beta \tag{19}$$

for the transportation location problem (TLP). *Here* $H_{max} := \max_{k\in[m]} H_k$ *is the maximum and* $\overline{H} := \frac{1}{m}\sum_{k=1}^{m} H_k$ *the mean of the maximum demands* H_k*.*

Proof. To prove this theorem we first give lower and upper bounds for the costs of the variables in (4). To get these bounds we consider the maximum and minimum demand possible together with the highest as well as the lowest transportation and storage cost per unit of the commodity.

$$\begin{aligned} \beta H_{\max}\left(\min_{f\in[n],k\in[m]} c_f + d_{fk}\right) &\leq z_{TLP}(\mathcal{V}_1) \leq \beta H_{\max}\left(\max_{f\in[n],k\in[m]} c_f + d_{fk}\right) \\ \Gamma\beta\overline{H}\left(\min_{f\in[n],k\in[m]} c_f + d_{fk}\right) &\leq z_{TLP}(\mathcal{V}_2) \leq \Gamma\beta\overline{H}\left(\max_{f\in[n],k\in[m]} c_f + d_{fk}\right) \end{aligned}$$

We use the maximum demand in the first line on the left hand side, since the demand of each client must be satisfied by the corresponding $\boldsymbol{y}_{1j} = \boldsymbol{y}(\beta \boldsymbol{e}_j)$. We have to use the average demand in the second line, otherwise we would overestimate the total demand. The approximation factor follows almost directly from these bounds:

$$\begin{aligned}
&\frac{\sigma_1 z_{TLP}(\mathcal{V}_1) + z_{TLP}(\mathcal{V}_2)}{z_{TLP}(\mathcal{V})} \leq \frac{\sigma_1 z_{TLP}(\mathcal{V}_1) + z_{TLP}(\mathcal{V}_2)}{\max\left(z_{TLP}(\mathcal{V}_1),\ z_{TLP}(\mathcal{V}_2)\right)} \\
\leq &\frac{\left(\max_{f\in[n],k\in[m]} c_f + d_{fk}\right)\left(\sigma_1 H_{\max} + \Gamma\overline{H}\right)\beta}{\left(\min_{f\in[n],k\in[m]} c_f + d_{fk}\right)\max\left(H_{\max},\ \Gamma\overline{H}\right)\beta} \\
= &\frac{\max_{f\in[n],k\in[m]} c_f + d_{fk}}{\min_{f\in[n],k\in[m]} c_f + d_{fk}} \frac{\sigma_1 H_{\max} + \Gamma\overline{H}}{\max\left(H_{\max},\ \Gamma\overline{H}\right)}
\end{aligned}$$

Using this equation in the last line in the proof of Theorem 1 yields the desired approximation factor.

Remark 2. In the extreme uniform case, where $c \equiv c_f$, $d \equiv d_{fk}$, $H \equiv H_k$, $\forall f \in [n]$, $\forall k \in [m]$, the instance specific approximation factor (19) is $\left(1 - \frac{\Gamma}{m}\right)\beta + 1$. For the choice of β' this again results in the general approximation factor of $\frac{\Gamma}{m}$, but for β^* this yields an instance specific approximation factor of $\frac{(\Gamma+1)m}{\Gamma^2+m}$, which is both almost a factor 2 improvement compared to the general factor of $\frac{2\Gamma m}{\Gamma^2+m}$ and strictly better than the approximation factor of the quasi-nominal solution.

7 Numerical Study

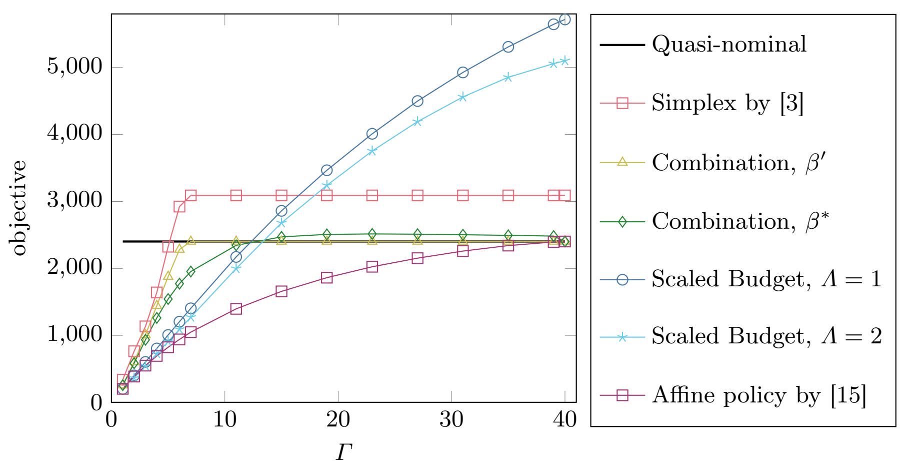

Fig. 2. Objective for TLP instances with $n = 20, m = 40$

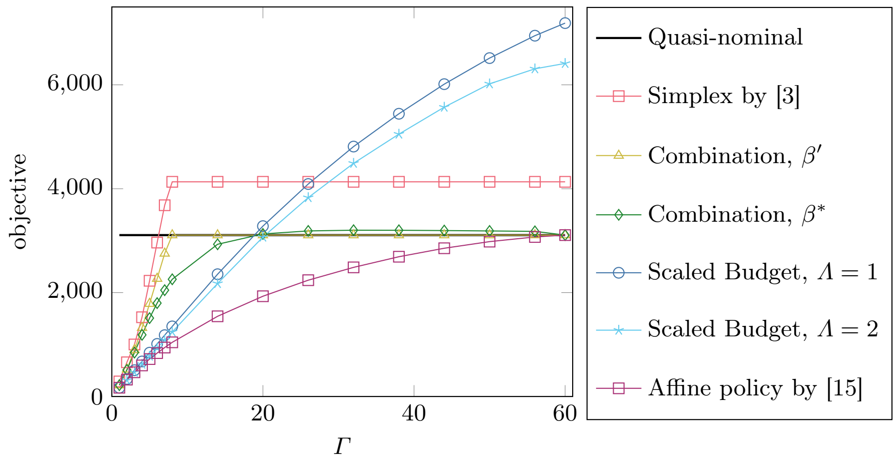

Fig. 3. Objective for TLP instances with $n = 30, m = 60$

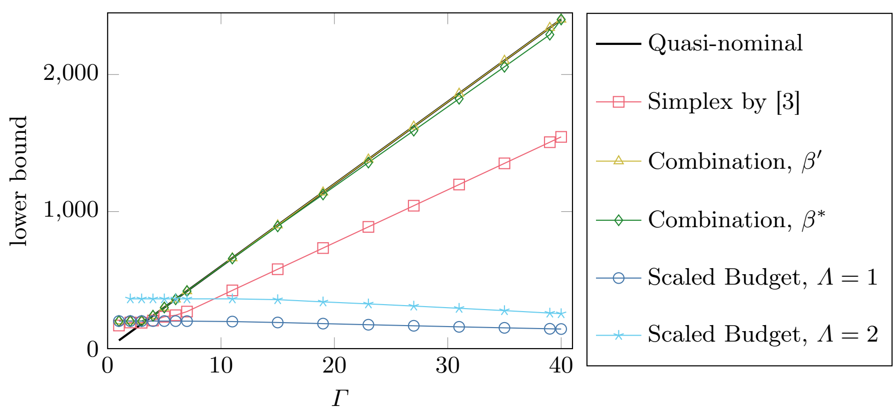

Fig. 4. Posterior bounds for TLP instances with $n = 20, m = 40$

This chapter reports on a numerical study for the TLP. The numerical behavior here with random generated data is very close the theoretical approximation factors shown before. We compare the methods presented in this paper against the simplex domination approach by Ben-Tal et al. [3] and the affine policies by El Housni and Goyal [15]. However, the second method does not provide a valid approximation factor in the case of the TLP, as the matrices B and D in (18) contain negative entries and therefore do not fulfill the necessary requirements. Nevertheless, we can use the method as a heuristic to calculate a feasible solution. All experiments were conducted on an AMD EPYC 7742 processor with a single core on 2480.438 MHz. We implemented all methods using SCIP 8.0.3 with the SoPlex 6.0.3 LP-solver [13].

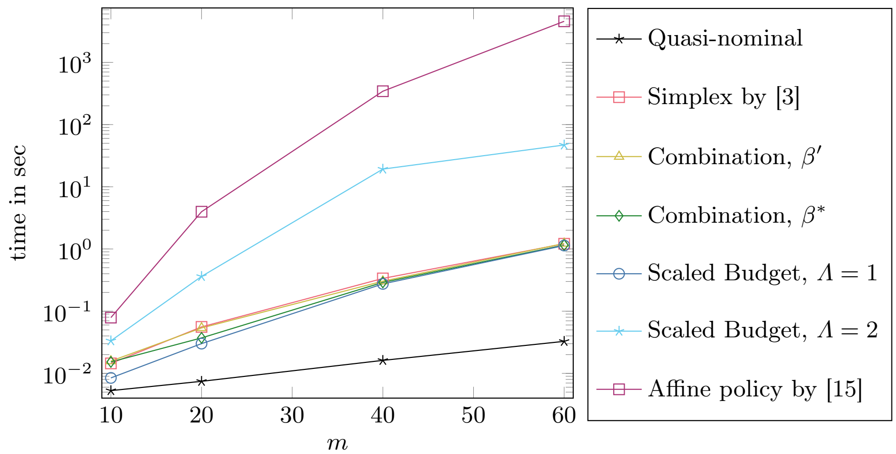

Fig. 5. Running time for TLP instances with $\Gamma = m$

Table 1. Running time for TLP instances with $n = 20$, $m = 40$ in seconds

Γ	Quasi-nominal	Simplex	Combination β'	Combination β^*	Scaled Budget $\Lambda = 1$	Scaled Budget $\Lambda = 2$	Affine policy
1	0.02	0.35	0.31	0.30	0.28	–	23.25
2	0.02	0.34	0.31	0.30	0.28	12.77	62.63
3	0.02	0.35	0.31	0.29	0.28	19.01	75.15
4	0.02	0.35	0.30	0.30	0.27	19.11	74.99
5	0.02	0.35	0.31	0.30	0.28	19.13	85.91
6	0.02	0.35	0.31	0.29	0.28	19.14	83.54
7	0.02	0.34	0.30	0.29	0.27	19.10	88.71
11	0.02	0.34	0.30	0.29	0.28	20.65	91.24
15	0.02	0.34	0.30	0.29	0.28	12.79	106.51
19	0.02	0.34	0.30	0.29	0.28	12.77	118.99
23	0.02	0.34	0.30	0.29	0.28	12.71	143.20
27	0.02	0.34	0.30	0.30	0.28	12.97	181.12
31	0.02	0.34	0.30	0.30	0.28	13.11	220.85
35	0.02	0.33	0.30	0.29	0.28	12.79	307.73
39	0.02	0.34	0.31	0.30	0.28	28.23	452.62
40	0.02	0.34	0.30	0.29	0.27	19.24	343.60

We generate instances with different amounts of customers n and facilities m uniformly over a 5×5 square. The transportation costs $\boldsymbol{d}$ between every pair of customer and facility equals their euclidean distances. The demands $\boldsymbol{H}$ of the customers are uniformly chosen from the interval $[20, 100]$. For the storage costs $\boldsymbol{c}$, we uniformly generated values in $[0.05, 1]$. The average results over five randomly generated instances for the different scenarios are presented for $n = 20$ and $m = 40$ in Figure 2 and for $n = 30$ and $m = 60$ in Figure 3.

Table 2. Running time for TLP instances with $n = 30$, $m = 60$ in seconds

Γ	Quasi-nominal	Simplex	Combination β'	Combination β^*	Scaled Budget $\Lambda = 1$	Scaled Budget $\Lambda = 2$	Affine policy
1	0.04	1.44	1.40	1.33	1.31	–	396.58
2	0.04	1.43	1.41	1.31	1.32	52.54	1551.04
3	0.04	1.40	1.38	1.28	1.29	52.73	1591.21
4	0.04	1.47	1.38	1.33	1.32	52.73	1628.88
5	0.04	1.45	1.38	1.31	1.31	52.85	1720.89
6	0.04	1.42	1.39	1.29	1.30	51.90	1597.51
7	0.04	1.45	1.40	1.32	1.32	53.41	1595.79
8	0.04	1.43	1.39	1.29	1.30	51.59	1641.66
14	0.04	1.41	1.37	1.31	1.31	51.71	1582.05
20	0.04	1.40	1.38	1.29	1.29	51.56	1781.78
26	0.04	1.41	1.39	1.30	1.30	51.99	1963.27
32	0.04	1.38	1.37	1.30	1.28	51.49	2232.49
38	0.04	1.31	1.31	1.22	1.22	48.82	2525.22
44	0.04	1.37	1.37	1.28	1.28	51.89	2969.12
50	0.04	1.37	1.37	1.27	1.28	51.72	4843.60
56	0.03	1.21	1.23	1.15	1.14	46.71	5067.21
60	0.03	1.21	1.23	1.15	1.14	46.94	4571.53

The Figures 2 and 3 of the objective function show that the performance of the methods is largely independent of the dimension of the problem and instead depend on the budget of uncertainty Γ. Besides the extreme cases of a very small or very large budget, the affine policy gives the best objective function values. Apart from this method, the scaled budget policies provide the next best results for small to moderate budgets before they are beaten by the quasi-nominal solution for all remaining larger budgets. The simplex approach is outperformed by our combination policies in every single instance.

Overall, Figures 2 and 3 are obviously similar to Figure 1 of the approximation factors, even though they describe different relations. In fact, the calculation of the quotients of the respective values shows a linear correlation in Figure 4. This gives us a lower bound on the optimal objective function value. The a posterior considerations from Remark 1 have already been used for the combination policies to get a better estimate. This gives us on average the best lower bounds with the combination policy.

In Figure 5 we can see that in each case the runtime of the calculations increases exponentially with the number of customers. The quasi-nominal solution is calculated fastest and the better approximation factors of the scaled budget policy with more vertices cost significantly more time, as predicted. All other policies based on domination can be calculated in about the same time, since they have almost the same number of vertices.

As expected, the affine policy has by far the worst runtimes. Interestingly, however, these times depend not only on the scale of the problem, but also on the budget of uncertainty Γ, as can be seen in Tables 1 and 2. For higher budgets, a multiple of the runtime of the smaller budgets is required. This phenomenon does not occur with any of the other methods.

8 Conclusion

Two-stage robust linear programs with righthand side uncertainty are of broad importance for real-world application, e.g. in logistics and transportation. We aim to find a practically superior and theoretically sound general approach for large-instances. We allow for arbitrary constraint matrices for the second stage and non-negative constraint coefficients for the first stage.

We introduce, analyze and computationally test a General Polyhedral Approximation (GPA). GPA is a general technique to approximate such problems by substituting the scenario set with the set of vertices of a set of polyhedra that jointly dominate (though not necessarily contain) the original scenario set. We extend and significantly strengthen techniques from the literature to derive improved approximation factors. Moreover, Theorem 1 applies in full generality to any such polyhedral approximation.

In particular, we consider budgeted scenario sets, where the m righthand side coefficients lie in bounded intervals and the sum of their relative deviations from the lower boundary of each interval is below a budget Γ. The theoretical analysis of the resulting approximation factor and the objective values in the computational experiments are almost perfectly aligned to each other and show the theoretical strength and practically superiority of GPA.

We compare affine linear approximations, approximation by a single simplex, quasi-nominal solutions and different realizations of GPA for two-stage robust linear programs with righthand side uncertainty. The resulting pictures for the theoretical approximation factor and the computational results are structured by the relation between m and Γ.

For *large budgets* Γ, the quasi-nominal solutions is quickly approaching the quality of affine policies. It is clearly better than approximating with a single simplex and slightly better than or in fact identical to our solutions.

For *small budgets*, i.e. $\Gamma < \sqrt{m}$, GPA is almost as good as the affine linear policies and significantly better than approximating with a single simplex. The quasi-nominal solutions are drastically worse than all adaptive solutions and unreasonable for many real-world applications.

For *medium budgets*, i.e. Γ close to $\sqrt{m}$, the *theoretical* picture of approximation factors deviates slightly from the computational results. Theoretically the approximation by most GPA variants and the single simplex approximation fall down the quality of a quasi-nominal solution. Beyond the $\sqrt{m}$-threshold they give worse approximation factors. But, using our family of scaled budget polytopes in GPA we can push this threshold, achieving strictly better approximation factors even for $\Gamma > \sqrt{m}$. In the *computational* experiments most of our GPA methods outperform the quasi-nominal solution even for budgets larger than their theoretical threshold.

The single simplex approach is always outperformed by GPA with the same computational cost. Affine linear policies are known to be best possible approximation - under slightly different assumptions, namely, that the second stage coefficients need to be non-negative but the first stage matrix is arbitrary. Still, they require prohibitive computational time for large instances.

To summarize, for large instances where affine policies cannot be calculated in practice, our findings suggest to use GPA unless Γ is so large that quasi-nominal solutions are the quick and good way to go. This threshold effect for non-adaptive solutions aligns with the finding in [8]. If non-adaptive solutions are not sufficient in practice, GPA with scaled budget polytopes allows for a trade-off between running time and quality until affine policies can be calculated.

We also show in this paper for the fundamental Transportation Location Problem how GPA can be used to find improved instance specific approximation factors. These factors are mirrored in the quality of solution in the computational experiments. This underlines the practical value of GPA.

Acknowledgements We would like to acknowledge the funding by the German Federal Ministry for Education and Research, grant number 05M20MBA-LeoPlan and by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy – EXC 2163/1 - Sustainable and Energy Efficient Aviation – Project-ID 390881007.

References

1. Ardestani-Jaafari, A., Delage, E.: The value of flexibility in robust location–transportation problems. Transportation Science **52**(1), 189–209 (2018). https://doi.org/https://doi.org/10.1287/trsc.2016.0728
2. Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: Robust optimization, vol. 28. Princeton university press (2009). https://doi.org/https://doi.org/10.1515/9781400831050
3. Ben-Tal, A., El Housni, O., Goyal, V.: A tractable approach for designing piecewise affine policies in two-stage adjustable robust optimization. Mathematical Programming **182**(1), 57–102 (2020). https://doi.org/https://doi.org/10.1007/s10107-019-01385-0
4. Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A.: Adjustable robust solutions of uncertain linear programs. Mathematical programming **99**(2), 351–376 (2004). https://doi.org/https://doi.org/10.1007/s10107-003-0454-y
5. Bertsimas, D., Caramanis, C.: Finite adaptability in multistage linear optimization. IEEE Transactions on Automatic Control **55**(12), 2751–2766 (2010). https://doi.org/https://doi.org/10.1109/TAC.2010.2049764
6. Bertsimas, D., Goyal, V.: On the power of robust solutions in two-stage stochastic and adaptive optimization problems. Mathematics of Operations Research **35**(2), 284–305 (2010). https://doi.org/https://doi.org/10.1287/moor.1090.0440
7. Bertsimas, D., Goyal, V.: On the power and limitations of affine policies in two-stage adaptive optimization. Mathematical programming **134**(2), 491–531 (2012). https://doi.org/https://doi.org/10.1007/s10107-011-0444-4

8. Bertsimas, D., Goyal, V.: On the approximability of adjustable robust convex optimization under uncertainty. Mathematical Methods of Operations Research **77**, 323–343 (2013). https://doi.org/https://doi.org/10.1007/s00186-012-0405-6
9. Bertsimas, D., Goyal, V., Lu, B.Y.: A tight characterization of the performance of static solutions in two-stage adjustable robust linear optimization. Mathematical Programming **150**(2), 281–319 (2015). https://doi.org/https://doi.org/10.1007/s10107-014-0768-y
10. Bertsimas, D., de Ruiter, F.J.: Duality in two-stage adaptive linear optimization: Faster computation and stronger bounds. INFORMS Journal on Computing **28**(3), 500–511 (2016). https://doi.org/https://doi.org/10.1287/ijoc.2016.0689
11. Bertsimas, D., Sim, M.: Robust discrete optimization and network flows. Mathematical programming **98**(1), 49–71 (2003). https://doi.org/https://doi.org/10.1007/s10107-003-0396-4
12. Bertsimas, D., Sim, M.: The price of robustness. Operations research **52**(1), 35–53 (2004). https://doi.org/https://doi.org/10.1287/opre.1030.0065
13. Bestuzheva, K., Besançon, M., Chen, W.K., Chmiela, A., Donkiewicz, T., van Doornmalen, J., Eifler, L., Gaul, O., Gamrath, G., Gleixner, A., Gottwald, L., Graczyk, C., Halbig, K., Hoen, A., Hojny, C., van der Hulst, R., Koch, T., Lübbecke, M., Maher, S.J., Matter, F., Mühmer, E., Müller, B., Pfetsch, M.E., Rehfeldt, D., Schlein, S., Schlösser, F., Serrano, F., Shinano, Y., Sofranac, B., Turner, M., Vigerske, S., Wegscheider, F., Wellner, P., Weninger, D., Witzig, J.: The SCIP Optimization Suite 8.0. Technical report, Optimization Online (December 2021)
14. El Housni, O., Foussoul, A., Goyal, V.: Lp-based approximations for disjoint bilinear and two-stage adjustable robust optimization. Mathematical Programming pp. 1–43 (2023). https://doi.org/https://doi.org/10.1007/s10107-023-02004-9
15. El Housni, O., Goyal, V.: On the optimality of affine policies for budgeted uncertainty sets. Mathematics of Operations Research (2021). https://doi.org/https://doi.org/10.1287/moor.2020.1082
16. El Housni, O., Goyal, V., Shmoys, D.: On the power of static assignment policies for robust facility location problems. In: Integer Programming and Combinatorial Optimization: 22nd International Conference, IPCO 2021, Atlanta, GA, USA, May 19–21, 2021, Proceedings 22. pp. 252–267. Springer (2021). https://doi.org/https://doi.org/10.1007/978-3-030-73879-2_18
17. Feige, U., Jain, K., Mahdian, M., Mirrokni, V.: Robust combinatorial optimization with exponential scenarios. In: International Conference on Integer Programming and Combinatorial Optimization. pp. 439–453. Springer (2007). https://doi.org/https://doi.org/10.1007/978-3-540-72792-7_33
18. Gabrel, V., Lacroix, M., Murat, C., Remli, N.: Robust location transportation problems under uncertain demands. Discrete Applied Mathematics **164**, 100–111 (2014). https://doi.org/https://doi.org/10.1016/j.dam.2011.09.015
19. Grunau, L.: Robust Transportation Location Problem. Master's thesis, Technische Universität Braunschweig, Germany (2021)
20. Guslitser, E.: Uncertainty-immunized solutions in linear programming. Ph.D. thesis, Citeseer (2002)
21. Marandi, A., van Houtum, G.J.: Robust location-transportation problems with integer-valued demand. Optimization Online (2020)
22. Simchi-Levi, D., Wang, H., Wei, Y.: Constraint generation for two-stage robust network flow problems. INFORMS Journal on Optimization **1**(1), 49–70 (2019). https://doi.org/https://doi.org/10.1287/ijoo.2018.0003

23. Thiele, A., Terry, T., Epelman, M.: Robust linear optimization with recourse. Rapport technique pp. 4–37 (2009)
24. Thomä, S., Walther, G., Schiffer, M.: Designing tractable piecewise affine policies for multi-stage adjustable robust optimization. Mathematical Programming pp. 1–56 (2024). https://doi.org/https://doi.org/10.1007/s10107-023-02053-0
25. Zeng, B., Zhao, L.: Solving two-stage robust optimization problems using a column-and-constraint generation method. Operations Research Letters **41**(5), 457–461 (2013). https://doi.org/https://doi.org/10.1016/j.orl.2013.05.003