General Polyhedral Approximation of Two-Stage Robust Linear Programming

Lukas Grunau, Tim Niemann, and Sebastian Stiller

Technische Universität Braunschweig, Institute for Mathematical Optimization, Universitätsplatz 2, 38106 Braunschweig, Germany
tim.niemann@tu-braunschweig.de

Abstract. We consider two-stage robust linear programs with a budget of uncertainty for the right hand side. In this scenario set, which is frequently used in robust optimization, the uncertain right hand side for each row lies in an interval and the relative increases summed over all rows are less than a budget $Γ$. We develop a General Polyhedral Approximation (GPA) method, in which the uncertainty set $U$ is substituted by a finite set of polytopes derived from the vertex set of an arbitrary polytope that dominates $U$. The method combines the solutions for the individual polytopes and thereby achieves an approximation factor of $β \sum_{i=1}^{N} σ_i$ significantly improving on the literature. Moreover, the method allows for even stronger results on specific problems as we exemplify for the Transportation Location Problem.

Previous methods reach a threshold at $Γ > \sqrt{m}$ after which they are not better than a quasi nominal solution, i.e., the power of second stage decisions cannot be used. The GPA method significantly improves the approximation factor before this threshold. After the threshold, as the GPA allows to use scaled budget polytopes, it provides a scheme to trade-off the number of vertices against the approximation factor. This way the GPA breaks the threshold and finds solutions that are better than the quasi nominal solution even for cases with $Γ > \sqrt{m}$.

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 $x$ for which, second, the malign adversary chooses a scenario from a set of scenarios $U$, and third, the optimizer chooses a second-stage solution $y$. Feasibility and cost of a solution $x$ are determined by the worst scenario specifically chosen for $x$ and with respect to a best possible second-stage decision $y$. When the scenario set $U$ is a polytope, it suffices in many (convex) settings to consider the vertices of the polytope as scenarios.
We consider linear programs with uncertain right-hand side. Our scenario set $U$ is the widely used set introduced in [10]. It is restricted in two ways: For each row the uncertain right-hand 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 $\Gamma$. This so-called budget of uncertainty set $U$, see Definition 1, is a polytope whose number of vertices grows exponentially with $\Gamma$. Considering each vertex of $U$ 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 $U$ 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 yields results that are even more conservative than the solutions obtained when simply computing a nominal solution to the scenario where all right-hand side values have maximal deviation, i.e., a scenario worse than all scenarios in $U$. 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 for the budget of uncertainty when pursued in a different, more general way: Instead of dominating with the simplex, we develop a method to use an arbitrary polytope dominating $U$, 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 $U$, we show a way to combine their solutions to a feasible solution for $U$. This solution has superior approximation factors. Also, the proofs become somewhat simpler by this more general approach. This General Polyhedral Approximation (GPA) method allows to use arbitrary polyhedra. We show that, in particular, using scaled budget of uncertainty polytopes with smaller budget to dominate the original set $U$ gives significantly better approximations. Also, the GPA method allows to calculate instance specific approximation guarantees. Finally, 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.

Related Work. It is known, that two-stage robust optimization problems are NP-hard [15] and not approximable within a factor better than $\Omega\left(\frac{\log m}{\log \log m}\right)$, c.f. [12]. This factor can be matched by affine policies [11] which are furthermore
optimal, if the uncertainty set is a simplex [7]. Further details on affine policies can be found in [1, 4, 9, 16]. The approximation factor of piecewise affine policies by Ben-Tal et al. was recently improved by Grunau [14] and Thomä et al. [19] in the case of budget of uncertainty sets. Bertsimas et al. analyze the quality of static solutions for the robust problem [8] and for the stochastic variant [6].

Other approaches consider the dualization of the inner minimization problem [13], which results in a nonlinear problem or column and constraint generation to potentially reduce the size of the problem. [17, 18, 20]. If the decision process is divided in more than two stages, there are algorithms considering finite adaptability [5] and approximation methods [19] 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 [13], which we will later discuss.

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 budget of 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:

\[
\begin{align*}
    z_{AR}(U) &= \min_x c^T x + \max_{h \in U} \min_{y(h)} d^T y(h) \\
    \text{s.t. } & Ax + By(h) \geq h \quad \forall h \in U \\
    & Cx + D y(h) \geq b \quad \forall h \in U \\
    & x \in \mathbb{R}^{n_1}, y(h) \in \mathbb{R}^{n_2}
\end{align*}
\]

(1)

where \( A \in \mathbb{R}^{m \times n_1}, B \in \mathbb{R}^{m \times n_2}, C \in \mathbb{R}^{p \times n_1}, D \in \mathbb{R}^{p \times n_2}, c \in \mathbb{R}^{n_1}, d \in \mathbb{R}^{n_2} \) and \( b \in \mathbb{R}^p \).

As scenario set \( U \), we use the following.

**Definition 1.** The budget of uncertainty set \([10]\) for a robustness parameter \( \Gamma > 0 \) is defined as

\[
U := U_\Gamma := \left\{ h \in [0,1]^m \mid \sum_{j=1}^m h_j \leq \Gamma \right\}.
\]

(2)

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

\[
V = \text{conv} \{ v_{ij} \mid i \in [N], j \in [M_i] \}, \quad V_i = \text{conv} \{ v_{ij} \mid j \in [M_i] \}.
\]

(3)
Note that the union of all $V_i$ does not necessarily dominates $\mathcal{U}$.

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 $\mathcal{V} \subseteq \beta \mathcal{U} := \{\beta h \mid h \in \mathcal{U}\}$. 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:

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

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 $x_i$, from which we construct the first stage decision $\hat{x}$. To compute the $x_i$, we need to solve $N$ linear programs, that are of the same type as (1) but lesser size:

$$z_{AR}(\mathcal{V}_i) = \min \ c^T x_i + w$$

s.t. $w \geq d^T y_{ij}$, $j \in [M_i], \quad Ax_i + By_{ij} \geq v_{ij}, \quad j \in [M_i], \quad Cx_i + Dy_{ij} \geq b, \quad j \in [M_i], \quad x_i \in \mathbb{R}^{n_1}, \quad y_{ij} \in \mathbb{R}^{n_2}, \quad j \in [M_i].$$

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

$$\sum_{i=1}^{N} \sum_{j=1}^{M_i} \alpha_{ij}(h)v_{ij} \geq h, \quad \forall h \in \mathcal{U}, \quad \sum_{i=1}^{N} \sum_{j=1}^{M_i} \alpha_{ij}(h) \geq 1, \quad \forall h \in \mathcal{U}. \quad (6)$$

We will later discuss possibilities for choosing $\alpha_{ij}(\cdot)$. Let $(x_i, y_i), i \in [N]$ be the $N$ solutions of (5). We will show, that for any choice of $\alpha_{ij}(\cdot)$ respecting the conditions (6) above, $(\hat{x}, \hat{y}(\cdot))$, defined as follows, is a feasible solution to (1).

$$\hat{x} = \sum_{i=1}^{N} \sigma_i x_i, \quad \hat{y}(\cdot) : \mathcal{U} \to \mathbb{R}^{n_2}, \quad \hat{y}(h) = \sum_{i=1}^{N} \sum_{j=1}^{M_i} \alpha_{ij}(h)y_{ij}, \quad (7)$$

The $\sigma_i$ are defined as $\sigma_i := \max_{h \in \mathcal{U}} \sum_{j=1}^{M_i} \alpha_{ij}(h)$. We can limit the unnecessary use of the first stage variables $x_i$ 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 $\sigma_i$ are chosen as small as possible to guarantee feasibility for each scenario $h \in \mathcal{U}$. Now we show that this gives a feasible solution for the original problem.

Lemma 2. The pair $(\hat{x}, \hat{y}(\cdot))$ defined in (7) is a feasible solution for (1), if $\sum_{i=1}^{N} \sum_{j=1}^{M_i} \alpha_{ij}(h)v_{ij} \geq h, \quad \forall h \in \mathcal{U}$ and $\sum_{i=1}^{N} \sum_{j=1}^{M_i} \alpha_{ij}(h) \geq 1$. 


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

\[
A\hat{x} + B\hat{y}(h) = A\sum_{i=1}^{N} \sigma_{i}x_{i} + B\sum_{i=1}^{N} \sum_{j=1}^{M_{i}} \alpha_{ij}(h)y_{ij} = \sum_{i=1}^{N} (\sigma_{i}Ax_{i} + \sum_{j=1}^{M_{i}} \alpha_{ij}(h)By_{ij})
\]

\[
\geq \sum_{i=1}^{N} \sum_{j=1}^{M_{i}} \alpha_{ij}(h) (Ax_{i} + By_{ij}) \geq \sum_{i=1}^{N} \sum_{j=1}^{M_{i}} \alpha_{ij}(h)v_{ij} \geq h.
\]

\[
C\hat{x} + D\hat{y}(h) = C\sum_{i=1}^{N} \sigma_{i}x_{i} + D\sum_{i=1}^{N} \sum_{j=1}^{M_{i}} \alpha_{ij}(h)y_{ij} = \sum_{i=1}^{N} (\sigma_{i}Cx_{i} + \sum_{j=1}^{M_{i}} \alpha_{ij}(h)Dy_{ij})
\]

\[
\geq \sum_{i=1}^{N} \sum_{j=1}^{M_{i}} \alpha_{ij}(h) (Cx_{i} + Dy_{ij}) \geq \sum_{i=1}^{N} \sum_{j=1}^{M_{i}} \alpha_{ij}(h)b \geq b.
\]

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

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

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

\[
c^{T}\hat{x} + d^{T}\hat{y}(h) = c^{T} \left( \sum_{i=1}^{N} \sigma_{i}x_{i} \right) + d^{T} \left( \sum_{i=1}^{N} \sum_{j=1}^{M_{i}} \alpha_{ij}(h)y_{ij} \right)
\]

\[
= \sum_{i=1}^{N} \left( \sigma_{i}c^{T}x_{i} + \sum_{j=1}^{M_{i}} \alpha_{ij}(h)d^{T}y_{ij} \right) \leq \sum_{i=1}^{N} \sigma_{i} \left( c^{T}x_{i} + \max_{j \in [M_{i}]} d^{T}y_{ij} \right)
\]

\[
= \sum_{i=1}^{N} \sigma_{i}z_{AR}(\mathcal{V}_{i}) \leq \left( \sum_{i=1}^{N} \sigma_{i} \right) z_{AR}(\mathcal{V}) \leq \left( \beta \sum_{i=1}^{N} \sigma_{i} \right) z_{AR}(\mathcal{U}).
\]

In the last line, we used the result of Lemma 1. As this bound holds for any scenario \( h \in \mathcal{U} \), 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 \( z_{AR}(\mathcal{U}) \), i.e., our theorem holds.

In the following subsections, we will give concrete examples of these policies and compare the resulting approximation factors.

**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

\[
\frac{\sum_{i=1}^{N} \sigma_{i} z_{AR}(\mathcal{V}_{i})}{\max_{i \in N} \left( z_{AR}(\mathcal{V}_{i}) \right)} \beta.
\]

We will see the affect of this in chapter 7 in Figure 3.
3 Scaling the Budget of Uncertainty Polytope

The number of vertices of the budget of uncertainty polytope grows exponentially with $\Gamma$. 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 $\beta$ and $\gamma$.

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

$$\mathcal{V} = \mathcal{V}_1 = \beta \mathcal{U}_\Lambda = \text{conv} \left\{ u \in \{0, \beta\}^m \left| \sum_{i=1}^m u_i = \Gamma \right. \right\} = \{v_1, \ldots, v_{M_1}\},$$

we can define the scaled budget policy with the coefficients

$$a_{1j}(h) = \begin{cases} \binom{\Gamma}{\Lambda}^{-1}, & \text{if } h \geq \frac{u_j}{\beta}, \\ 0, & \text{else}. \end{cases}$$

With this definition, Theorem 1 yields the following corollary.

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 $m^\Lambda$ 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 (5), we can get a better approximation factor of the policy. We will later see the affects of this in Figure 1 and Figure 4.

4 The Power of Combining Polytopes

The GPA method 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.

Definition 3 (Combination policy). Let $\mathbf{v} = \frac{\Gamma}{m} \mathbf{1}$ 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:

\[ V_1 = \beta \text{conv}\{e_1, \ldots, e_m\}, \quad V_2 = \{\beta v\}, \]

\[ \alpha_{1j}(h) = \frac{1}{\beta} (h_j - \beta v_j)^+, \quad \alpha_{21}(h) = 1, \]

\[ \sigma_1 = \frac{m\Gamma - \beta^2 \Gamma^2}{\beta m} \leq 1, \quad \sigma_2 = 1. \]

(11) (12) (13)

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

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

**Proof.** For a budgeted uncertainty set with \( v = \frac{r}{m} \mathbb{1} \), we know that

\[ \sum_{j=1}^{M_1} \alpha_{1j}(h) = \Gamma \frac{1}{\beta} (1 - \beta \gamma) = \frac{\Gamma}{\beta} \left(1 - \beta \frac{\Gamma}{m}\right) = \frac{m\Gamma - \beta^2 \Gamma^2}{\beta m} = \sigma_1. \]

(14)

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

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

(15)

The following lemma gives an optimal choice for beta.

**Lemma 3.** The optimal scaling factor \( \beta_{opt} \) dependent on the budget of uncertainty \( \Gamma \) for the combination policy is given by:

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

(16)

**Proof.** The approximation factor of Corollary 2 is an affine linear function in \( \beta \), so the minimum is dependent on the sign of the gradient. In other words, we need to find the minimal feasible \( \beta \), if \( \left(1 - \frac{r^2}{m}\right) \) is positive and the maximal feasible \( \beta \) if the term is negative. If the gradient is zero, every feasible \( \beta \) with \( \beta v \leq 1 \) 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 \( \beta \) as large as possible but feasible. So \( \beta v = 1 \) directly gives \( \beta' = \frac{m}{\Gamma} \) with \( \sigma_1 \equiv 0 \).

In the next chapter in Figure 1 the effect of this choice of \( \beta \) will be demonstrated. In a nutshell, for small \( \Gamma \) this factor is strictly better while for larger \( \Gamma \) 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 \( \beta^* \) see [14, 19].
5 Choosing The Best Method

The different approximation factors of methods discussed in this paper all depend on \( \Gamma \), i.e., the number of rows with maximally increased righthand side and \( m \) the total number of rows. Here we plot these factors for comparison with \( m = 100 \).

![Figure 1. Approximation factors for \( m = 100 \) dependent on \( \Gamma \).](image)

In Figure 1 we consider the following methods:

- Quasi-nominal (\( \frac{\Gamma}{m} \)): 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 \( \min(\Gamma, \frac{\Gamma}{m}) \)): 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 \( \beta' \) and \( \beta^* \). It is easy to show that after intersection of these factors \( \beta' \) 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 \( \Lambda \). The factors are linear and can be truncated when they are worse than the that of the quasi-nominal solution, because the intersection can be calculated a priori.

For large \( \Gamma \) the quasi-nominal solution performs best. For small \( \Gamma \) the scaled budget policy is best. The threshold between small and large \( \Gamma \) is at \( \sqrt{m} \) for all policies except for more complex scaled budget polytopes, i.e., \( \Lambda \geq 2 \). Increasing \( \Lambda \) in turn exponentially increases the size of the linear programs to be solved. Therefore, the choice of the best method depends on \( m, \Gamma \), 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 the GPA method 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 the 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. We assume that each client $k \in [m]$ has a maximum demand $H_k$ and that not more than $\Gamma$ clients will request their demand or - leading to equivalent solutions - the normalized demand $h_k \in [0, 1]^m$ lies inside the budget of uncertainty set $U$. Formally, this can be expressed by the following optimization problem:

$$
\begin{align*}
\min_{x} & \quad c^T x + \max_{h \in U} \min_y d^T y \\
\text{s.t.} & \quad \sum_{f=1}^{n} y^{fk} \geq h_k H_k, \quad k \in [m], \quad x^f - \sum_{k=1}^{m} y^{fk} \geq 0, \quad f \in [n], \quad x, y \geq 0. 
\end{align*}
$$

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 $k$. To improve readability, we use double indices for the vector $(y_{ij})_{n,m}$. The storage costs are contained in the vector $c$, $d_{fk}$ 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 $h$ is a vector from the budget of uncertainty set $U$ from (2). Both variables, $x$ and $y$, are continuous. To meet the notation of the previous chapters, one can reformulate problem (17) to:

$$
\begin{align*}
\min_x & \quad c^T x + \max_{h \in U} \min_{y(h)} d^T y(h) \\
\text{s.t.} & \quad B y(h) \geq h, \forall h \in U, \quad C x + D y(h) \geq 0, \forall h \in U, \quad x, y \geq 0.
\end{align*}
$$

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. $C$ is the $n \times n$ identity matrix and the matrices
from these bounds: overestimate the total demand. The approximation factor follows almost directly.

We have to use the average demand in the second line, otherwise we would demand of each client must be satisfied by the corresponding approximation factor.

Using this equation in the last line in the proof of Theorem 2 yields the desired approximation factor.

\[ B \in \mathbb{R}^{m \times (n-m)} \text{ and } D \in \mathbb{R}^{n \times (n-m)} \] are defined by

\[
B = \begin{pmatrix}
\frac{1}{H_1} & \cdots & \frac{1}{H_1} & 0 & \cdots & \cdots & \cdots & 0 \\
0 & \cdots & 0 & \frac{1}{H_2} & \cdots & \cdots & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \\
0 & \cdots & \cdots & 0 & \frac{1}{H_m} & \cdots & \cdots & \frac{1}{H_m} \\
\end{pmatrix}, \quad D = (-I - I \cdots - I). \quad (18)
\]

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

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

\[
\frac{\max_{f \in [n], k \in [m]} c_f + d_{fk}}{\min_{f \in [n], k \in [m]} c_f + d_{fk}} \leq \frac{\sigma_1 H_{\max} + \Gamma \bar{H}}{\max(H_{\max}, \Gamma \bar{H})} \beta
\]

for the transportation location problem (TLP). Here \( H_{\max} := \max_{k \in [m]} H_k \) is the maximum and \( \bar{H} := \frac{1}{m} \sum_{k=1}^m H_k \) 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 (5). 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.

\[
\beta H_{\max} \left( \min_{f \in [n], k \in [m]} c_f + d_{fk} \right) \leq z_{TLP}(V_1) \leq \beta H_{\max} \left( \max_{f \in [n], k \in [m]} c_f + d_{fk} \right)
\]

\[
\Gamma \beta \bar{H} \left( \min_{f \in [n], k \in [m]} c_f + d_{fk} \right) \leq z_{TLP}(V_2) \leq \Gamma \beta \bar{H} \left( \max_{f \in [n], k \in [m]} c_f + d_{fk} \right)
\]

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 \( y_{1j} = y(\beta e_j) \). 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:

\[
\frac{\sigma_1 z_{TLP}(V_1) + z_{TLP}(V_2)}{z_{TLP}(V)} \leq \frac{\sigma_1 z_{TLP}(V_1) + z_{TLP}(V_2)}{\max(z_{TLP}(V_1), z_{TLP}(V_2))}
\]

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

\[
= \frac{\max_{f \in [n], k \in [m]} c_f + d_{fk} \sigma_1 H_{\max} + \Gamma \bar{H}}{\min_{f \in [n], k \in [m]} c_f + d_{fk} \max(H_{\max}, \Gamma \bar{H})}
\]

Using this equation in the last line in the proof of Theorem 2 yields the desired approximation factor.
7 Numerical Study

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

Figure 2 of the objective functions is 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 3. 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 4 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 can be calculated in about the same time, since they have almost the same number of vertices.

8 Conclusion

To summarize the picture of two-stage righthand side robust linear programs with a budget of uncertainty: So far, there was a threshold at $\Gamma > \sqrt{m}$ after which the quasi-nominal solution is good enough, i.e., the possibility of second stage decisions need not be considered. The GPA method significantly improves the approximation factor before this threshold. Because, GPA works for any dominating polytope, it is a versatile tool that can produce even stronger results for specific classes of problems. Moreover, as the GPA allows to use scaled budget polytopes we arrive at a scheme to trade-off number of vertices against approximation quality. This way we can break the threshold, find better solutions than the quasi-nominal, and exploit the possibility of second stage decisions even for $\Gamma > \sqrt{m}$.

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