Online Supplement to "Perspective Benders Decomposition with Applications to Fixed-Charge Nonlinear Resource Allocation"

Kang Yang*, Guopeng Song*, Rui Wang†
College of Systems Engineering, National University of Defense Technology, Changsha 410073, China
youngclover23@outlook.com; guopeng.song@nudt.edu.cn; ruiwangnudt@gmail.com

Haoxiang Yang
School of Data Science, The Chinese University of Hong Kong, Shenzhen (CUHK-Shenzhen), Shenzhen 518172, China
yanghaoxiang@cuhk.edu.cn

Roel Leus
ORSTAT, Faculty of Economics and Business, KU Leuven, 3000 Leuven, Belgium
roel.leus@kuleuven.be

Online Supplement A: Proofs

A.1. Proof of Proposition 1

Proof: Given a convex function $\phi(x)$ and $|\phi(0)| < \infty$, due to the convexity of $\phi(x)$ (Hiriart-Urruty and Lemaréchal 1996), for $y \in [0,1]$, the following inequality holds :

$$y\phi(\frac{x}{y}) + (1-y)\phi(0) \geq \phi(y\frac{x}{y} + (1-y)0) = \phi(x).$$

Given that for all $j \in [n]$, $\phi_j(0) = 0$, the above inequality implies that $\sum_{i\in[m]}\sum_{j\in[n_i]} \overline{y}_i \phi_j(\frac{x_j^*}{\overline{y}_i}) \geq \sum_{j\in[n]} \phi_j(x_j^*)$. As $\boldsymbol{x}^*$ and $\boldsymbol{x}^\star$ are the feasible solutions of the optimality subproblems of (P) with $\boldsymbol{x}^\star$ being the optimal solution, then $\sum_{j\in[n]} \phi_j(x_j^*) \geq \sum_{j\in[n]} \phi_j(x_j^\star)$. Therefore it holds that $\sum_{i\in[m]}\sum_{j\in[n_i]} \overline{y}_i \phi_j(\frac{x_j^*}{\overline{y}_i}) \geq \sum_{j\in[n]} \phi_j(x_j^*) \geq \sum_{j\in[n]} \phi_j(x_j^\star)$. $\square$

A.2. Proof of Proposition 2

Proof: Given a feasible solution of the relaxed master problem $\overline{\boldsymbol{y}}$ and optimal solution $\boldsymbol{x}^*$ for SP, summing the perspective cuts separated at the point $(\boldsymbol{x}^*, \overline{\boldsymbol{y}})$ and the constraints of SP weighted with the optimal

* These authors contributed equally to this work.
† The corresponding author.

multipliers $(\boldsymbol{\lambda}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ will yield the following linear inequality:

$$\begin{aligned}\theta &\geq \sum_{i\in[m]}\underbrace{\sum_{j\in[n_i]}((\phi_j(\frac{x_j^*}{\overline{y}_i}) - \frac{x_j^*}{\overline{y}_i}\phi_j'(\frac{x_j^*}{\overline{y}_i}))y_i + \phi_j'(\frac{x_j^*}{\overline{y}_i})x_j)}_{\text{the right-hand side of perspective cuts}} + \boldsymbol{\lambda}^{*T}(\boldsymbol{B}\boldsymbol{x} - \boldsymbol{r}) + \sum_{i\in[m]}\boldsymbol{\alpha}_i^{*T}(\boldsymbol{D}_i\boldsymbol{x}_i - y_i\boldsymbol{d}_i) - \boldsymbol{\beta}^{*T}\boldsymbol{x} \\ &= \sum_{i\in[m]}\sum_{j\in[n_i]}\overline{y}_i\phi_j(\frac{x_j^*}{\overline{y}_i}) + \sum_{i\in[m]}(\sum_{j\in[n_i]}(\phi_j(\frac{x_j^*}{\overline{y}_i}) - \frac{x_j^*}{\overline{y}_i}\phi_j'(\frac{x_j^*}{\overline{y}_i})))(y_i - \overline{y}_i) + \sum_{i\in[m]}\sum_{j\in[n_i]}\phi_j'(\frac{x_j^*}{\overline{y}_i})(x_j - x_j^*) \\ &+ \boldsymbol{\lambda}^{*T}(\boldsymbol{B}\boldsymbol{x} - \boldsymbol{r}) + \sum_{i\in[m]}\boldsymbol{\alpha}_i^{*T}(\boldsymbol{D}_i\boldsymbol{x}_i - y_i\boldsymbol{d}_i) - \boldsymbol{\beta}^{*T}\boldsymbol{x}.\end{aligned}$$

Reformulate the right-hand sides of the inequality using the first-order approximation of the linear constraints at the point $(\boldsymbol{x}^*, \overline{\boldsymbol{y}})$, resulting in the following inequality:

$$\begin{aligned}\theta &\geq \sum_{i\in[m]}\sum_{j\in[n_i]}\overline{y}_i\phi_j(\frac{x_j^*}{\overline{y}_i}) + \sum_{i\in[m]}(\sum_{j\in[n_i]}(\phi_j(\frac{x_j^*}{\overline{y}_i}) - \frac{x_j^*}{\overline{y}_i}\phi_j'(\frac{x_j^*}{\overline{y}_i})))(y_i - \overline{y}_i) + \sum_{i\in[m]}\sum_{j\in[n_i]}\phi_j'(\frac{x_j^*}{\overline{y}_i})(x_j - x_j^*) + \boldsymbol{\lambda}^{*T}(\boldsymbol{B}\boldsymbol{x}^* - \boldsymbol{r}) \\ &+ \boldsymbol{\lambda}^{*T}\boldsymbol{B}(\boldsymbol{x} - \boldsymbol{x}^*) + \sum_{i\in[m]}\boldsymbol{\alpha}_i^{*T}(\boldsymbol{D}_i\boldsymbol{x}_i^* - \overline{y}_i\boldsymbol{d}_i) + \sum_{i\in[m]}\boldsymbol{\alpha}_i^{*T}\boldsymbol{D}_i(\boldsymbol{x}_i - \boldsymbol{x}_i^*) - \sum_{i\in[m]}\boldsymbol{\alpha}_i^{*T}\boldsymbol{d}_i(y_i - \overline{y}_i) - \boldsymbol{\beta}^{*T}\boldsymbol{x}^* - \boldsymbol{\beta}^{*T}(\boldsymbol{x} - \boldsymbol{x}^*).\end{aligned}$$

Due to complementary slackness, it holds that $\boldsymbol{\lambda}^{*T}(\boldsymbol{B}\boldsymbol{x}^* - \boldsymbol{r}) = \boldsymbol{\alpha}_i^{*T}(\boldsymbol{D}_i\boldsymbol{x}_i^* - \overline{y}_i\boldsymbol{d}_i) = \boldsymbol{\beta}^{*T}\boldsymbol{x}^* = 0$. Therefore, the inequality can be simplified as follows:

$$\begin{aligned}\theta &\geq \sum_{i\in[m]}\sum_{j\in[n_i]}\overline{y}_i\phi_j(\frac{x_j^*}{\overline{y}_i}) + \sum_{i\in[m]}(\sum_{j\in[n_i]}(\phi_j(\frac{x_j^*}{\overline{y}_i}) - \frac{x_j^*}{\overline{y}_i}\phi_j'(\frac{x_j^*}{\overline{y}_i})) - \boldsymbol{d}_i^T\boldsymbol{\alpha}_i^*)(y_i - \overline{y}_i) \\ &+ \sum_{i\in[m]}\sum_{j\in[n_i]}\phi_j'(\frac{x_j^*}{\overline{y}_i})(x_j - x_j^*) + \boldsymbol{\lambda}^{*T}\boldsymbol{B}(\boldsymbol{x} - \boldsymbol{x}^*) + \sum_{i\in[m]}\boldsymbol{\alpha}_i^{*T}\boldsymbol{D}_i(\boldsymbol{x}_i - \boldsymbol{x}_i^*) - \boldsymbol{\beta}^{*T}(\boldsymbol{x} - \boldsymbol{x}^*).\end{aligned} \tag{A.1}$$

According to the first-order condition, it holds that $\frac{\partial\mathcal{L}}{\partial x_i}\big|_{x_i = x_i^*} = 0$ for each $i \in [n]$, which leads to the following equality:

$$\sum_{i\in[m]}\sum_{j\in[n_i]}\phi_j'(\frac{x_j^*}{\overline{y}_i})(x_j - x_j^*) + \boldsymbol{\lambda}^{*T}\boldsymbol{B}(\boldsymbol{x} - \boldsymbol{x}^*) + \sum_{i\in[m]}\boldsymbol{\alpha}_i^{*T}\boldsymbol{D}_i(\boldsymbol{x}_i - \boldsymbol{x}_i^*) - \boldsymbol{\beta}^{*T}(\boldsymbol{x} - \boldsymbol{x}^*) = 0,$$

then the inequality (A.1) can be further simplified into the perspective Benders cuts as inequality (20).

□

A.3. Proof of Theorem 2

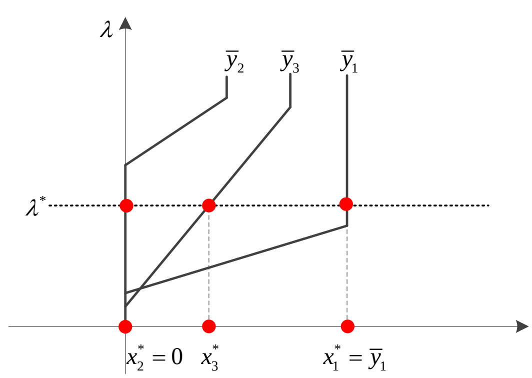


Figure 1 Illustrative example of the relation between λ^* and x^* for the proof of Theorem 2.

Proof: Assuming Slater's condition holds, the convexity of (QP1) ensures that a solution satisfying the KKT conditions is optimal. Specifically, we aim to find a solution $(\boldsymbol{x}^*, \lambda^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)$ that satisfies the following KKT conditions:

$$\frac{\partial \mathcal{L}(\boldsymbol{x}^*, \lambda^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)}{\partial x_i^*} = \frac{x_i^*}{a_i} + b_i - \lambda^* + \alpha_i^* - \beta_i^* = 0, \qquad \forall i \in [m], \tag{A.2}$$

$$(\boldsymbol{x}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \geq \mathbf{0}, \tag{A.3}$$

$$\sum_{i\in[m]} x_i^* = 1, \tag{A.4}$$

$$x_i^* \leq \overline{y}_i, \qquad \forall i \in [m], \tag{A.5}$$

$$\alpha_i^*(x_i^* - \overline{y}_i) = 0, \qquad \forall i \in [m], \tag{A.6}$$

$$\beta_i^* x_i^* = 0, \qquad \forall i \in [m]. \tag{A.7}$$

According to the complementary slackness condition, when $0 < x_i^* < \overline{y}_i$, it follows that $\alpha_i^* = \beta_i^* = 0$. Consequently, from the first-order conditions, we obtain $x_i^* = a_i(\lambda^* - b_i)$. Furthermore, since other x_i can only take boundary values of either 0 or $\overline{y}_i$, and given that x_i^* is a monotonically increasing function of λ^*, we conclude that x_i^* is the projection of $a_i(\lambda^* - b_i)$ onto the interval $[0, \overline{y}_i]$. In other words, for any given λ^*, the corresponding $x_i^*(\lambda^*)$ is computed as $x_i^*(\lambda^*) = \min\{\max\{0, a_i(\lambda^* - b_i)\}, \overline{y}_i\}$. This can be visually illustrated by Figure 1. The value set for λ^* determines associated values for x_i^*. Let $r(\lambda^*) := \sum_{i\in[m]} \min\{\max\{0, a_i(\lambda^* - b_i)\}, \overline{y}_i\}$. Consequently, finding the optimal solution to (QP1) is equivalent to identifying a value λ^* such that $r(\lambda^*) = 1$. Before introducing the algorithm for finding the optimal multiplier λ^*, let us first explore the following residual problem, (QP2), which disregards the box constraints (34)–(35), with $S \subseteq [m]$ and $\tilde{r} \in [0, 1]$:

$$\text{(QP2)} \quad \min \sum_{i\in S} \frac{\tilde{x}_i^2}{2a_i} + b_i \tilde{x}_i \tag{A.8}$$

$$s.t. \sum_{i\in S} \tilde{x}_i = \tilde{r}. \tag{A.9}$$

Let $\tilde{\lambda}^* \geq 0$ be the optimal dual multiplier for constraint (A.9). It is well-known (Günlük et al. 2007) – and easily derived from the KKT conditions – that the closed-form solution of (QP2) is as follows:

$$\begin{cases} \tilde{\lambda}^* = \dfrac{\tilde{r} + \sum_{i\in S} a_i b_i}{\sum_{i\in S} a_i}; \\ \tilde{x}_i^*(\tilde{\lambda}^*) = a_i(\tilde{\lambda}^* - b_i), \ \forall i \in S. \end{cases} \tag{A.10}$$

An efficient strategy for finding the optimal multiplier λ^* in (QP1) involves initializing it with the optimal multiplier $\tilde{\lambda}^*$ obtained from the corresponding (QP2). Under the initial conditions, let S be the set $[m]$, and $\tilde{r}$ equals 1. The relationship between $r(\tilde{\lambda}^*)$ and $\tilde{r}$ is as follows:

$$\begin{aligned} r(\tilde{\lambda}^*) - \tilde{r} &= \sum_{i\in S} \min\{\max\{0, a_i(\tilde{\lambda}^* - b_i)\}, \overline{y}_i\} - \tilde{r} = \sum_{i\in S} \min\{\max\{0, a_i(\tilde{\lambda}^* - b_i)\}, \overline{y}_i\} - \sum_{i\in S} \tilde{x}_i^* \\ &= \sum_{i\in S^-} (0 - \tilde{x}_i^*) + \sum_{i\in S^+} (\overline{y}_i - \tilde{x}_i^*) + \sum_{i\in S'} (a_i(\tilde{\lambda}^* - b_i) - \tilde{x}_i^*) = \sum_{i\in S^-} (0 - \tilde{x}_i^*) + \sum_{i\in S^+} (\overline{y}_i - \tilde{x}_i^*) := \delta^- - \delta^+, \end{aligned}$$

where S^+ denotes the set $\{i \in S : \tilde{x}_i^* \geq \overline{y}_i\}$, S^- denotes the set $\{i \in S : \tilde{x}_i^* \leq 0\}$ and S' denotes the set $S \backslash (S^+ \cup S^-)$. Notice that $r(\lambda)$ is a monotonically increasing, piecewise linear function of λ. Therefore, we

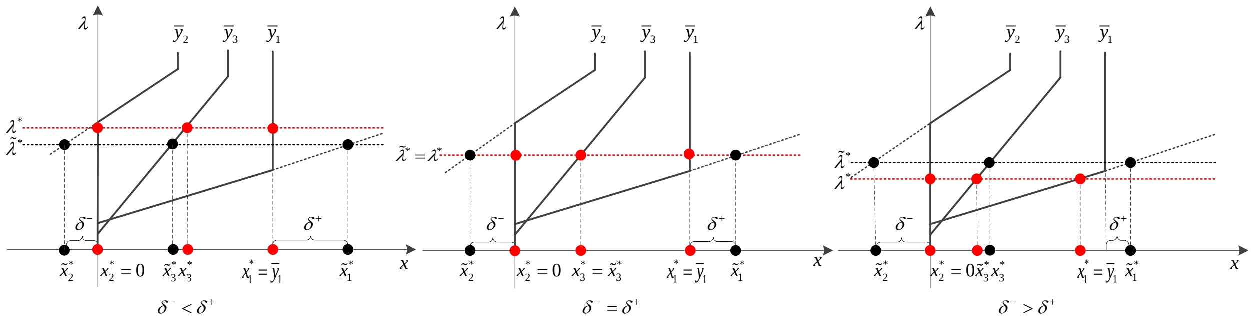


Figure 2 **Examples to illustrate the correlation between the value of the optimal multipliers, λ^* for (QP1) and $\tilde{\lambda}^*$ for (QP2), in relation to the value of δ^- and δ^+. The graphical representation depicts the details of solutions for (QP1) in red and those for (QP2) in black.**

can determine the relationship between $\tilde{\lambda}^*$ and the optimal multiplier λ^* based on the comparison of $\delta^- - \delta^+$ with 0. Since $r(\lambda)$ is monotonic and $\tilde{r}$ is initiated to 1, there is $\tilde{\lambda}^* = \lambda^*$ when $r(\tilde{\lambda}^*) = \tilde{r} = 1$.

As illustrated in Figure 2, if $\delta^- - \delta^+ = 0$, then we can conclude that $\tilde{\lambda}^*$ is the optimal multiplier λ^*, and the associated optimal solution $x_i^*(\lambda^*)$ is calculated as $\min\{\max\{0, a_i(\lambda^* - b_i)\}, \overline{y}_i\}$. If $\delta^- - \delta^+ < 0$, then we can infer that $\tilde{\lambda}^* < \lambda^*$, and $\overline{y}_i < \tilde{x}_i^*(\tilde{\lambda}^*) < \tilde{x}_i^*(\lambda^*)$ for each $i \in S^+$. Therefore, we can set x_i^* as $\overline{y}_i$ for each $i \in S^+$ at Step 26 in Algorithm 1 without compromising the optimality of the solution. If $\delta^- - \delta^+ > 0$, then we can conclude that $\tilde{\lambda}^* > \lambda^*$, and $0 > \tilde{x}_i^*(\tilde{\lambda}^*) > \tilde{x}_i^*(\lambda^*)$ for each $i \in S^-$. Hence, we can set x_i^* as 0 at Step 21 in Algorithm 1 for each $i \in S^-$ without losing the optimality of the solution.

The aforementioned procedure, initiated by setting S as $[m]$ and $\tilde{r}$ as 1, involves solving the corresponding residual problem in a single iteration (a complete inner loop from Step 7 to Step 28), thereby it can fix at least one variable. Subsequently, we utilize S' to further initialize S while continuing the next iteration by employing 1 minus the relevant fixed variable values to update $\tilde{r}$. This iterative process is repeated to effectively fix variables. As the decision variables $\boldsymbol{x}$ have a finite dimension m, when all dimensions of the variables are fixed, the algorithm converges to the optimal solution of the problem. In the worst-case scenario, the algorithm fixes one variable in each iteration, requiring at most m iterations. Since the dimension of variables $\boldsymbol{x}$ is initially m and we fix at least one variable after each iteration, the total work is linear in $(m + (m-1) + (m-2) + \ldots + 1) = m/2(m+1)$, so the time complexity of Algorithm 1 is $\mathcal{O}(m^2)$.

□

A.4. Compute the Multiplier λ^* Using an Incremental Updating Approach

The procedure in A.3 for updating $\tilde{\lambda}^*$ using Equation (A.10) requires computing $\sum_{i\in S} a_i b_i$ and $\sum_{i\in S} a_i$ from scratch. Here, we present a more efficient incremental updating approach that avoids this recalculation. Let λ and λ' be the value of $\tilde{\lambda}^*$ at Step 28 in Algorithm 1 before and after the removal of i from S arising at Step 26, respectively, and let r and r' be the corresponding values of $\tilde{r}$, then the following relationships can be established:

$$\begin{aligned}
\lambda' \sum_{i\in S'} a_i = r' + \sum_{i\in S'} a_i b_i &= r - \sum_{i\in S^+} \overline{y}_i + \sum_{i\in S} a_i b_i - \sum_{i\in S^+} a_i b_i = \lambda \sum_{i\in S} a_i - \sum_{i\in S^+} \overline{y}_i - \sum_{i\in S^+} a_i b_i \\
&= \lambda \sum_{i\in S'} a_i + \sum_{i\in S^+} ((a_i\lambda - a_i b_i) - \overline{y}_i) = \lambda \sum_{i\in S'} a_i + \sum_{i\in S^+} (\tilde{x}_i^* - \overline{y}_i).
\end{aligned}$$

Let $\tau := \sum_{i\in S'} a_i = \sum_{i\in S} a_i - \sum_{i\in S^+} a_i$ and $\delta^+ := \sum_{i\in S^+} (\tilde{x}_i^* - \overline{y}_i)$, then we have following update equation at Step 28 in Algorithm 1:

$$\lambda' = \lambda + \delta^+/\tau.$$

Similarly, the following equality can be established:

$$\lambda' \sum_{i\in S'} a_i = r' + \sum_{i\in S'} a_i b_i = r + \sum_{i\in S} a_i b_i - \sum_{i\in S^-} a_i b_i = \lambda \sum_{i\in S} a_i - \sum_{i\in S^-} a_i b_i$$
$$= \lambda \sum_{i\in S'} a_i + \sum_{i\in S^-} (a_i\lambda - a_i b_i) = \lambda \sum_{i\in S'} a_i + \sum_{i\in S^-} \tilde{x}_i^*.$$

Let $\tau := \sum_{i\in S'} a_i = \sum_{i\in S} a_i - \sum_{i\in S^-} a_i$ and $\delta^- := \sum_{i\in S^-} (0 - \tilde{x}_i^*)$, then we have following update equation at Step 23 in Algorithm 1:

$$\lambda' = \lambda - \delta^-/\tau.$$

A.5. Proof of Proposition 3

Proof: The KKT conditions for the (SP1) problem are as follows:

$$\frac{\partial \mathcal{L}(\boldsymbol{x}^*, \lambda^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*)}{\partial x_i^*} = \phi_i'(\frac{x_i^*}{\overline{y}_i}) - \lambda^* + \alpha_i^* - \beta_i^* = 0, \qquad \forall i \in [m], \tag{A.11}$$
$$(\boldsymbol{x}^*, \boldsymbol{\alpha}^*, \boldsymbol{\beta}^*) \geq \mathbf{0}, \tag{A.12}$$
$$\sum_{i\in[m]} x_i^* = 1, \tag{A.13}$$
$$x_i^* \leq \overline{y}_i, \qquad \forall i \in [m], \tag{A.14}$$
$$\alpha_i^*(x_i^* - \overline{y}_i) = 0, \qquad \forall i \in [m], \tag{A.15}$$
$$\beta_i^* x_i^* = 0, \qquad \forall i \in [m]. \tag{A.16}$$

After the execution of Algorithm 1, if the set S is non-empty, implying the existence of $0 < x_i^* < \overline{y}_i$ for $i \in [m]$, according to complementary slackness, we have $\alpha_i^* = \beta_i^* = 0$. Furthermore, from the first-order condition (A.11), it can be inferred that $\lambda^* = \phi_i'(x_i^*/\overline{y}_i)$. If the set S is empty, then the solution already satisfies constraint (33). Therefore, constraint (33) is redundant, thus λ^* equals 0. When $x_i^* = 0 < \overline{y}_i$, we have $\alpha_i^* = 0$. When $x_i^* = \overline{y}_i$, we have $\beta_i^* = 0$, furthermore, it can be derived that $\alpha_i^* = \lambda^* - \phi_i'(x_i^*/\overline{y}_i) = \lambda^* - \phi_i'(\overline{y}_i/\overline{y}_i)$. Considering that $\phi_i'(x_i^*/\overline{y}_i)$ is a monotonically increasing function with respect to x_i^*, when $0 < x_i^* < \overline{y}_i$, we have $\lambda^* - \phi_i'(\overline{y}_i/\overline{y}_i) < \lambda^* - \phi_i'(x_i^*/\overline{y}_i) = 0 = \alpha_i^*$, thus all α_i^* can be calculated uniformly by $\max\{\lambda^* - \phi_i'(\overline{y}_i/\overline{y}_i), 0\}$. It is worth noting that not all $\overline{y}_i/\overline{y}_i$ are equal to 1, as $\overline{y}_i$ may be equal to 0, for which 0/0 is set to 0. □

A.6. Proof of Theorem 3

Proof: Let points $(0, 0, z_0) \in C_0 = \{(x, y, z) \in C : y = 0\}$ and points $(x_1, 1, z_1) \in C_1 = \{(x, y, z) \in C : y = 1\}$, and $conv(C)$ is the convex hull of C. We prove the two directions of the theorem sequentially:

- We first prove $conv(C) \subseteq C_{PR}$. Consider that an arbitrary point (x, y, z) of $conv(C)$ is a convex combination of $(0, 0, z_0)$ and $(x_1, 1, z_1)$, as follows:

$$(x, y, z) = \lambda(x_1, 1, z_1) + (1 - \lambda)(0, 0, z_0) = (\lambda x_1, \lambda, \lambda z_1 + (1 - \lambda)z_0), \quad \forall \lambda \in [0, 1].$$

Since $l \le x_1 \le u$, then $l\lambda \le \lambda x_1 \le u\lambda$, and observe that $f(x_1) \le z_1$ and $f(0) \le z_0$, then:

$$\lambda f(\frac{\lambda x_1}{\lambda}) + (1-\lambda)f(0) = \lambda f(x_1) + (1-\lambda)f(0) \le \lambda z_1 + (1-\lambda)z_0,$$

we therefore have $conv(C) \subseteq C_{PR}$.

- Then we prove $C_{PR} \subseteq conv(C)$. Apparently, the function $yf(x/y)$ is ill-defined when $y=0$. However, assuming the continuity of $f(x)$ and $|f(0)| < \infty$, we can define $0f(0/0)=0$. This implies that when $y=0$ or $y=1$, then $(x,y,z) \in C_{PR} \subseteq conv(C)$. When $0<y<1$, then an arbitrary point $(x,y,z) \in C_{PR}$ is a convex combination of $(0,0,f(0)) \in C_0$ and $(x/y, 1, \frac{z-(1-y)f(0)}{y})$, as follows:

$$(x,y,z) = (1-y)(0,0,f(0)) + y(x/y,1,\frac{z-(1-y)f(0)}{y}), \quad \forall y \in (0,1).$$

In order to prove $C_{PR} \subseteq conv(C)$, it is sufficient to prove that an arbitrary point $(x/y,1,\frac{z-(1-y)f(0)}{y}) \in C_1 \subseteq conv(C)$. Since $ly \le x \le uy$, then $l \le x/y \le u$. and observe that $yf(x/y) + (1-y)f(0) \le z$, then:

$$f(\frac{x}{y}) \le \frac{z-(1-y)f(0)}{y},$$

therefore $(x/y,1,\frac{z-(1-y)f(0)}{y}) \in C_1 \subseteq conv(C)$, we have that $C_{PR} \subseteq conv(C)$.

□

Online Supplement B: The Details of the Rewriting Process

In this section, we provide a detailed rewriting process from $\overline{y}_i\phi_i(x_i/\overline{y}_i)$ to $a_i\psi(x_i/a_i+b_i)$ in Table 1.

- For $\phi_i(x) := c_i x^p$, the $\overline{y}_i\phi_i(x_i/\overline{y}_i)$ can be rewritten as $a_i\psi(x_i/a_i+b_i)$ by choosing $a_i = \frac{\overline{y}_i}{c_i^{\frac{1}{p-1}}}$ and $b_i = 0$ for each $i \in [m]$ and $\psi(x) := x^p$ with $p > 1$:

$$c_i\overline{y}_i(\frac{x_i}{\overline{y}_i})^p = c_i x_i^p/\overline{y}_i^{p-1} = \frac{\overline{y}_i}{c_i^{\frac{1}{p-1}}}(\frac{x_i c_i^{\frac{1}{p-1}}}{\overline{y}_i})^p = a_i\psi(\frac{x_i}{a_i}).$$

- For $\phi_i(x) := \frac{c_i x}{1-x/k_i}$, the $\overline{y}_i\phi_i(x_i/\overline{y}_i)$ can be rewritten as $a_i\psi(x_i/a_i+b_i)$ by choosing $a_i = \sqrt{c_i}\overline{y}_i k_i$ and $b_i = -\frac{1}{\sqrt{c_i}}$ for each $i \in [m]$ and $\psi(x) := -\frac{1}{x}$ with $x < 0$:

$$\frac{c_i x_i}{1 - x_i/(k_i\overline{y}_i)} = -c_i\overline{y}_i k_i + \frac{c_i\overline{y}_i^2 k_i^2}{\overline{y}_i k_i - x_i} = -c_i\overline{y}_i k_i - \frac{\sqrt{c_i}\overline{y}_i k_i}{x_i/(\sqrt{c_i}\overline{y}_i k_i) - 1/\sqrt{c_i}} = -c_i\overline{y}_i k_i + a_i\psi(\frac{x_i}{a_i} + b_i).$$

- For $\phi_i(x) := c_i(exp(-k_i x) - 1)$, the $\overline{y}_i\phi_i(x_i/\overline{y}_i)$ can be rewritten as $a_i\psi(x_i/a_i+b_i)$ by choosing $a_i = \frac{\overline{y}_i}{k_i}$ and $b_i = -\log(c_i k_i)$ for each $i \in [m]$ and $\psi(x) := exp(-x)$ with $x \in \mathbb{R}$:

$$c_i\overline{y}_i exp(-k_i x_i/\overline{y}_i) - c_i\overline{y}_i = -c_i\overline{y}_i + \frac{\overline{y}_i}{k_i}exp(\frac{-x_i}{\overline{y}_i/k_i} + \log(c_i k_i)) = -c_i\overline{y}_i + a_i\psi(\frac{x_i}{a_i} + b_i).$$

- For $\phi_i(x) := -c_i\log(1+k_i)$, the $\overline{y}_i\phi_i(x_i/\overline{y}_i)$ can be rewritten as $a_i\psi(x_i/a_i+b_i)$ by choosing $a_i = c_i\overline{y}_i$ and $b_i = \frac{1}{k_i c_i}$ for each $i \in [m]$ and $\psi(x) := -\log(x)$ with $x > 0$:

$$-c_i\overline{y}_i\log(1+k_i x_i/\overline{y}_i) = -c_i\overline{y}_i\log(k_i c_i\frac{\overline{y}_i/k_i + x_i}{c_i\overline{y}_i}) = -c_i\overline{y}_i\log(\frac{x_i}{c_i\overline{y}_i} + \frac{1}{k_i c_i}) - c_i\overline{y}_i\log(k_i c_i) = a_i\psi(\frac{x_i}{a_i} + b_i) - c_i\overline{y}_i\log(k_i c_i).$$

- For $\phi_i(x) := \dfrac{c_i}{k_i x+1}$, the $\overline{y}_i\phi_i(x_i/\overline{y}_i)$ can be rewritten as $a_i\psi(x_i/a_i+b_i)$ by choosing $a_i=\overline{y}_i\sqrt{\dfrac{c_i}{k_i}}$ and $b_i=\dfrac{1}{\sqrt{c_ik_i}}$ for each $i\in[m]$ and $\psi(x):=\dfrac{1}{x}$ with $x>0$:

$$\frac{c_i\overline{y}_i}{k_ix_i/\overline{y}_i+1}=\frac{c_i\overline{y}_i^2}{k_ix_i+\overline{y}_i}=\sqrt{\frac{c_i\overline{y}_i^2}{k_i}}\frac{1}{x_i\sqrt{\dfrac{k_i}{c_i\overline{y}_i^2}}+\dfrac{1}{\sqrt{c_ik_i}}}=a_i\psi(\frac{x_i}{a_i}+b_i).$$

- For $\phi_i(x):=xf(c_ix)$, the $\overline{y}_i\phi_i(x_i/\overline{y}_i)$ can be rewritten as $a_i\psi(x_i/a_i+b_i)$ by choosing $a_i=\dfrac{\overline{y}_i}{c_i}$ and $b_i=0$ for each $i\in[m]$ and convex function $\psi(x):=xf(x)$:

$$x_if(c_i\frac{x_i}{\overline{y}_i})=\frac{\overline{y}_i}{c_i}\frac{x_i}{\overline{y}_i/c_i}f(\frac{x_i}{\overline{y}_i/c_i})=a_i\psi(\frac{x_i}{a_i}).$$

Online Supplement C: The Details of Constructing the SOCP Representations

In this section, we provide a detailed introduction to the construction of SOCP representations for different perspective functions, including rational power and delay functions.

C.1. The Details of Constructing the SOCP Representations of Rational Power Functions

Lemma 2. *(Alizadeh and Goldfarb 2003) For a positive integer l, an inequality of the form $x^{2^l}\le s_1s_2,...,s_{2^l}$, for $x\in\mathbb{R}$ and $s_1,s_2,...,s_{2^l}\ge 0$, can be expressed equivalently by 2^{l-1} inequalities of the form $w^2\le uv$, where the variables w,u,v are all nonnegative.*

In this section, we present construction details of the SOCP representations for rational power functions x^p/y^{p-1} when $p=1.5$ and 2.5. The construction details for $p=2$ and 3 can be found in Aktürk et al. (2009). First let the rational exponent $p=a/b>1$, where the integers $a\ge b>0$. Then constraint $x^p\le y^{(p-1)}z$ can be transformed into constraint $x^a\le y^{(a-b)}z^b$, and constraint $x^a\le y^{(a-b)}z^b$ can be rewritten as constraint $x^{2^l}\le y^{(a-b)}z^bx^{2^l-a}$. Based on Lemma 2 and the construction method using the binary tree in Alizadeh and Goldfarb (2003), we can reformulate constraint $x^p\le y^{(p-1)}z$ to a set of SOC constraints.

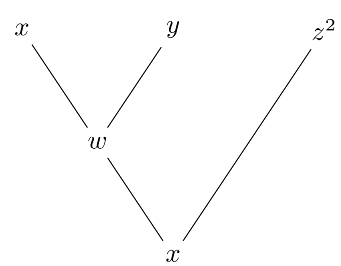
For example, consider the two cases:

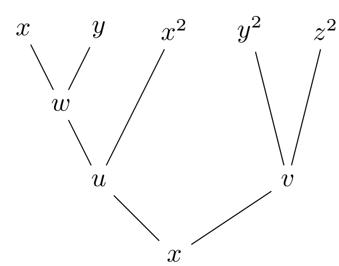
$$\textit{Constraint } 1: \quad x^{1.5}\le y^{0.5}z, \qquad\qquad \textit{Constraint } 2: \quad x^{2.5}\le y^{1.5}z.$$

We can rewrite the above constraints to:

$$\textit{Constraint } 1: \quad x^{4}\le xyz^2, \qquad\qquad \textit{Constraint } 2: \quad x^{8}\le x^3y^3z^2.$$

Introducing the auxiliary variables, the corresponding binary trees of these formulations can be illustrated below:





Finally, these constraints can be represented respectively by inequality systems as follows:

$$\mathit{Constraint}\ 1: \begin{cases} w^2 \le xy \\ x^2 \le wz \end{cases}, \qquad \mathit{Constraint}\ 2: \begin{cases} w^2 \le xy \\ u^2 \le xw \\ v^2 \le yz \\ x^2 \le uv \end{cases}.$$

From the binary tree construction method presented in Alizadeh and Goldfarb (2003), it is apparent that the SOCP representations for Constraint 2 are not unique.

C.2. Construction of Delay Function's SOCP Representations

In this section, we provide the details of constructing the SOCP representations of delay function $\frac{x}{1-x/(ky)}$, where $0 < k \le 1$, as discussed in Günlük and Linderoth (2010).

Consider the epigraph constraint including delay function $\frac{x}{1-x/(ky)}$ as follows :

$$\frac{x}{1-x/(ky)} \le z.$$

Rewrite the above constraint to:

$$kxy + (x^2 - x^2) \le (ky - x)z,$$

the above constraint can be simplified as follows:

$$x^2 \le (ky - x)(z - x). \tag{C.1}$$

Given that the variables x are continuous and normalized within the interval $[0,1]$, with ky being the upper bounds, both $ky - x$ and $z - x$ are non-negative terms. Therefore, we can introduce non-negative continuous variables u and v to replace $z - x$ and $ky - x$, respectively. This allows us to transform the constraint (C.1) into an equivalent inequality system, as follows:

$$\begin{cases} x^2 \le uv \\ u = z - x \\ v = ky - x \end{cases}.$$

Online Supplement D: Construction of AP^{2}R Formulations

In this section, we present the detailed procedure of constructing AP^{2}R formulations for specific GSPP variants, including rational power and delay functions, as discussed in Frangioni et al. (2011) and Frangioni et al. (2016). In this section, we only deal with the AP^{2}R formulation for *one block* at a time. For ease of notation, we omit the subscript i in this section. Furthermore, based on the formulation of GSPP, it is given that the univariate y falls within the interval $[x, 1]$ with $x \ge 0$.

D.1. The Details of Constructing AP^{2}R Formulations of Rational Power Functions

The perspective function of an objective function of the form $fy + cx^p$ is $fy + \frac{cx^p}{y^{p-1}}$. The optimal solution of the unconstrained optimization problem with respect to variable y can be obtained by solving the first-order optimality condition $f + \frac{c(1-p)x^p}{y^p} = 0$, which yields $\hat{y} = x\left(\frac{c(p-1)}{f}\right)^{\frac{1}{p}}$. Below, we will discuss the different cases based on whether $\hat{y}$ falls within the interval $[x, 1]$:

- If $\hat{y} = x\left(\frac{c(p-1)}{f}\right)^{\frac{1}{p}} \leq x$, then $\left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}} \geq 1$. In this case, the optimal solution is the projection of $\hat{y}$ over the feasible interval $[x,1]$, i.e., $y^* = x$, and the objective function value is $(f+c)x$.
- If $\hat{y} = x\left(\frac{c(p-1)}{f}\right)^{\frac{1}{p}} > x$, then $\left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}} < 1$. In this case, we need to further discuss the relationship between $\hat{y}$ and 1 in order to determine the optimal solution y^*. If $\hat{y} = x\left(\frac{c(p-1)}{f}\right)^{\frac{1}{p}} \geq 1$, then we have $x \geq \left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}}$. In this case, the optimal solution is the projection of $\hat{y}$ over the feasible interval $[x,1]$, i.e., $y^* = 1$. If $\hat{y} = x\left(\frac{c(p-1)}{f}\right)^{\frac{1}{p}} < 1$, then we have $x < \left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}}$. In this case, the optimal solution is $y^* = x\left(\frac{c(p-1)}{f}\right)^{\frac{1}{p}}$.

The optimal objective function value is a piecewise function with respect to variable x, as follows:

$$\begin{cases} \left(\frac{p}{p-1}\left(c(p-1)f^{p-1}\right)^{\frac{1}{p}}\right)x, & 0 \leq x < \left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}}, \\ f + cx^p, & \left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}} \leq x \leq 1. \end{cases}$$

Based on the above analysis, it is evident that we can eliminate the binary variable y through projection operations, thereby avoiding the intractability caused by the fractional term $y_i\phi_j(x_j/y_i)$. This allows us to construct a relaxed model with respect to the continuous variable x. Considering that when $\left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}} < 1$, the objective function is a two-piecewise function with respect to the continuous variable x, we can employ the scheme proposed by Frangioni and Gentile (2011) to transform the two-piecewise objective function into an equivalent non-piecewise function with respect to two continuous variables x_1 and x_2, as follows:

$$\begin{aligned} \min \ & \left(\frac{p}{p-1}\left(c(p-1)f^{p-1}\right)^{\frac{1}{p}}\right)x_1 + (f + c(x_2 + \widehat{x})^p) - (f + c\widehat{x}^p) \\ s.t. \ & 0 \leq x_1 \leq \widehat{x}, \\ & 0 \leq x_2 \leq 1 - \widehat{x}, \\ & x = x_1 + x_2, \end{aligned}$$

where $\widehat{x} := (\frac{f}{c(p-1)})^{\frac{1}{p}}$.

Although the intractability caused by the fractional term $y_i\phi_j(x_j/y_i)$ can be avoided through projection operations, it also eliminates the binary variable y in the model, which means that only a relaxed model of the original model can be solved. Using this projection, one cannot apply an off-the-shelf solver such as CPLEX or GUROBI to solve the original problem completely. To address this issue, we can utilize the scheme proposed by Frangioni et al. (2016) to “lift” the relaxed model back to the original variable space to construct an AP^2R formulation that includes the binary variables y. We distinguish between two cases:

- When $\left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}} \geq 1$, the AP^2R formulation of one block is as follows:

$$\begin{aligned} \min \ & (f+c)y, \\ s.t. \ & y \in \{0,1\}, \\ & x = y. \end{aligned}$$

- When $\left(\frac{f}{c(p-1)}\right)^{\frac{1}{p}} < 1$, according to Theorem 1 of Frangioni et al. (2016), the AP^{2}R formulation of one block is as follows:

$$
\begin{aligned}
\min \ & \frac{pf}{p-1}(y-1)+f+c(z+\widehat{x})^p \\
s.t. \ & -\widehat{x}y \le z \le (1-\widehat{x})y, \\
& y \in \{0,1\}, \\
& x = \widehat{x}y + z.
\end{aligned}
$$

By reintroducing other neglected constraints that are solely related to x, the complete AP^{2}R formulation in Section 5.3.2 can be derived.

D.2. The Details of Constructing AP^{2}R Formulations of Delay Function

For the objective function of the form $fy+\frac{cx}{1-x/k}$, the perspective function is $fy+\frac{ckxy}{ky-x}$. The optimal solution of the unconstrained optimization problem with respect to variable y can be obtained by solving the first-order optimality condition $f-\frac{ckx^2}{(ky-x)^2}=0$ $(0\le x\le ky,\ 0<k\le 1)$, which yields $\hat{y}=\frac{x}{k}+\sqrt{\frac{c}{fk}}x$. Below, we will discuss the different cases based on whether $\hat{y}$ falls within the interval $[x,1]$:

Since $\hat{y} > x$ clearly holds, we need to further discuss the relationship between $\hat{y}$ and 1 in order to determine the optimal solution y^*. If $\hat{y}=\frac{x}{k}+\sqrt{\frac{c}{fk}}x\ge 1$, then we have $x\ge \frac{1}{\frac{1}{k}+\sqrt{\frac{c}{fk}}}$. In this case, the optimal solution is the projection of $\hat{y}$ over the feasible interval $[x,1]$, i.e., $y^*=1$. If $\hat{y}=\frac{x}{k}+\sqrt{\frac{c}{fk}}x<1$, then we have $x<\frac{1}{\frac{1}{k}+\sqrt{\frac{c}{fk}}}$. In this case, the optimal solution is $y^*=\frac{x}{k}+\sqrt{\frac{c}{fk}}x$. The optimal objective function value is a two-piecewise function with respect to variable x, as follows:

$$
\begin{cases}
(c+2\sqrt{\frac{fc}{k}}+\frac{f}{k})x, 0\le x\le \frac{1}{\frac{1}{k}+\sqrt{\frac{c}{fk}}}, \\
f+\frac{ckx}{k-x}, \frac{1}{\frac{1}{k}+\sqrt{\frac{c}{fk}}}\le x\le 1.
\end{cases}
$$

Similarly, we can also transform the two-piecewise objective function into an equivalent non-piecewise function with respect to two continuous variables x_1 and x_2, as follows:

$$
\begin{aligned}
\min \ & \left(c+2\sqrt{\frac{fc}{k}}+\frac{f}{k}\right)x_1+\left(f+\frac{ck(x_2+\widehat{x})}{k-(x_2+\widehat{x})}\right)-\left(f+\frac{ck\widehat{x}}{k-\widehat{x}}\right) \\
s.t. \ & 0\le x_1\le \widehat{x}, \\
& 0\le x_2\le 1-\widehat{x}, \\
& x = x_1+x_2,
\end{aligned}
$$

where $\widehat{x}:=\frac{1}{\frac{1}{k}+\sqrt{\frac{c}{fk}}}$. And, similarly, we can also "lift" the above relaxed model back to the original variable space to construct an AP^{2}R formulation of one block as follows:

$$
\begin{aligned}
\min \ & (f+\sqrt{kcf})(y-1)+f+\frac{ck(z+\widehat{x})}{k-(z+\widehat{x})} \\
s.t. \ & -\widehat{x}y\le z\le (1-\widehat{x})y, \\
& y\in\{0,1\}, \\
& x=\widehat{x}y+z.
\end{aligned}
$$

By reintroducing other neglected constraints that are solely related to x, the complete AP^{2}R formulation in Section 5.3.2 can be derived.

References

Aktürk MS, Atamtürk A, Gürel S (2009) A strong conic quadratic reformulation for machine-job assignment with controllable processing times. *Operations Research Letters* 37(3):187–191.

Alizadeh F, Goldfarb D (2003) Second-order cone programming. *Mathematical Programming* 95(1):3–51.

Frangioni A, Furini F, Gentile C (2016) Approximated perspective relaxations: a project and lift approach. *Computational Optimization and Applications* 63(3):705–735.

Frangioni A, Gentile C, Grande E, Pacifici A (2011) Projected perspective reformulations with applications in design problems. *Operations Research* 59(5):1225–1232.

Günlük O, Lee J, Weismantel R (2007) MINLP strengthening for separable convex quadratic transportation-cost UFL. Technical Report RC24213 (W703-042), Research Division, IBM, Yorktown Heights, NY.

Günlük O, Linderoth J (2010) Perspective reformulations of mixed integer nonlinear programs with indicator variables. *Mathematical Programming* 124(1-2):183–205.

Hiriart-Urruty JB, Lemaréchal C (1996) *Convex analysis and minimization algorithms I: Fundamentals*, volume 305 of *Grundlehren der mathematischen Wissenschaften* (Heidelberg: Springer Berlin).