Approximating value functions via corner Benders' cuts

Matheus Jun Ota[*1], Ricardo Fukasawa[†1], and Aleksandr M. Kazachkov[‡2]

[1]Department of Combinatorics and Optimization, University of Waterloo,
Waterloo, Ontario, Canada

[2]Department of Industrial and Systems Engineering, University of Florida,
Gainesville, Florida, United States of America

September 26, 2025

Abstract

We introduce a novel technique to generate Benders' cuts from a conic relaxation ("corner") derived from a basis of a higher-dimensional polyhedron that we aim to outer approximate in a lower-dimensional space. To generate facet-defining inequalities for the epigraph associated to this corner, we develop a computationally-efficient algorithm based on a compact reverse polar formulation and a row generation scheme that handles the redundant inequalities. Via a known connection between arc-flow and path-flow formulations, we show that our method can recover the linear programming bound of a Dantzig-Wolfe formulation using multiple cuts in the projected space. In computational experiments, our generic technique enhances the performance of a problem-specific state-of-the-art algorithm for the vehicle routing problem with stochastic demands, a well-studied variant of the classic capacitated vehicle routing problem that accounts for customer demand uncertainty.

Keywords: integer programming, Benders' decomposition, Dantzig-Wolfe decomposition, Lagrangian relaxation, network flow, stochastic vehicle routing problem.

1 Introduction

Optimizing a large-scale *linear program* (LP) often relies on *decomposition* methods that distinguish "easier-to-solve" structure in the problem from "complicating" subproblems, repeatedly deriving information from the subproblems as needed until eventually converging to optimality (Vanderbeck and Wolsey, 2010). We focus on improving *Benders' decomposition* (Benders, 1962), a method that partitions the LP variables into *master variables* and *subproblem variables*, which we may also refer to as *first-stage* and *second-stage* variables. In this approach, an initial convex relaxation is formulated in the space of the first-stage variables and is iteratively refined with *Benders' cutting planes (cuts)*, which are inequalities derived from solving the second-stage subproblems.

Benders' cuts relay constraints and costs from the subproblems by projecting them into the space of the master decision variables. This projection step requires the solution

[*] (mjota@uwaterloo.ca)

[†] (rfukasawa@uwaterloo.ca)

[‡] (akazachkov@ufl.edu)

of an LP in the space of the subproblem variables and the subsequent cut is derived based on the LP optimal primal/dual solutions (i.e., the Benders' subproblem). While these cuts suffice for eventual convergence, the procedure may require a high overall computational time (Rahmaniani et al., 2017). Our starting point is the observation that using only an optimal solution of a Benders' subproblem discards other potentially relevant information, such as the corresponding optimal basis. This inspires our main contributions:

- We propose *corner Benders' cuts* to **capture more subproblem structure from basis information** and accelerate the convergence of typical Benders' decomposition methods.
- We argue that, even though the corner Benders' cuts are obtained from a relaxation of the original LP, any regular Benders' cut is in fact a corner Benders' cut.
- We show how one can **efficiently obtain and generate corner Benders' cuts** based on objective function cuts (i.e., inequalities that impose a lower bound on the objective function value), attaining the same LP bound implied by those cuts, but without the drawback of using objective-parallel inequalities.
- We computationally demonstrate that the ideas derived lead to **state-of-the-art branch-and-cut algorithm performance** for the *vehicle routing problem with stochastic demands.*

As far as we are aware, this is the first work that leverages basis information to achieve better computational performance of Benders' cuts without exploiting integrality constraints.

1.1 Problem setup

Concretely, we consider the following optimization problem over variables $x \in X \subseteq \mathbb{R}^n$ and $y \in Y \subseteq \mathbb{R}^m$, where X and Y are nonempty polyhedra defined by rational data, with Y assumed to be pointed. The variables are coupled by p equations, $Tx + Qy = h$, referred to as *linking constraints*:

$$\begin{aligned} z^* := \min_{x,y} \quad & c^\intercal x + d^\intercal y \\ & Tx + Qy = h, \\ & (x, y) \in X \times Y. \end{aligned} \tag{LP}$$

While we do not include any integer restrictions on the variables, problem (LP) arises in the discrete setting as well, when solving a relaxation of a mixed-integer LP instance, which is the context of our computational experiments.

We define the first-stage problem in the x variables and we treat Y as a polyhedron associated with the second-stage subproblem.[1] Benders' decomposition projects out the y variables via cuts that approximate the second-stage *value function* given fixed first-stage decisions $\bar{x}$:

$$f_Y(\bar{x}) := \min_y \{d^\intercal y : Qy = h - T\bar{x},\, y \in Y\}.$$

This leads to the following reformulation of (LP) (with same optimal objective value z^*) in the space of the first-stage variables and an additional *epigraph* variable θ:

$$\min_{x,\theta} \{c^\intercal x + \theta : \theta \geq f_Y(x),\, x \in X\}. \tag{1}$$

[1] The set Y could encompass a block structure with multiple smaller subproblems. For an extension to this setting, see Appendix C.

1.2 Intuition for corner Benders' cuts

The purpose of Benders' cuts is to provide an outer approximation of Problem (1) with linear inequalities. In prior work, a central focus for improving Benders' decomposition has been choosing an appropriate normalization for a cut-generating linear program associated to the dual formulation of $f_Y(\bar{x})$ for a given candidate solution $(\bar{x}, \bar{\theta})$ violating the epigraph constraint in Problem (1) (Fischetti et al., 2010; Bonami et al., 2020; Conforti and Wolsey, 2019; Brandenberg and Stursberg, 2021; Hosseini and Turner, 2024). We instead concentrate on a new *approach* to Benders' cuts. We note that there have been many other works that deal with Benders' cuts that use the integrality of some of the variables, for example (Küçükyavuz and Sen, 2017; Gade et al., 2014; Zou et al., 2019; Rahmaniani et al., 2020; Chen and Luedtke, 2022). We refrain from doing a thorough literature review of those types of works since our improvement is based solely on (LP), which lacks integer restrictions on the variables.

The core motivation for our approach is computational: a single Benders' cut is generated by solving an LP in the Y space and provides an outer approximation of Problem (1), but the same LP also offers basis information from which we will derive a set of Benders' cuts and obtain a tighter approximation of the value function. For efficiency in computing these cuts, we replace Y by a much simpler relaxation C. This idea is illustrated in Figure 1, where f_C denotes the function obtained by replacing Y with C in the definition of f_Y. Specifically, given a basic feasible solution to $f_Y(\bar{x})$, we use for C the *basis cone*, which (perhaps more evocatively) is also known as the *relaxed (Gomory's) corner polyhedron*—an object often used for deriving cuts for integer programs (Conforti et al., 2011)—defined by the subset of constraints that are tight at an optimal solution and associated to the nonbasic variables (Gomory, 1965, 1969). We show that every Benders' cut is valid for the epigraph of f_C, for some appropriately chosen basis cone C. Moreover, in our computational experiments, when C is selected according to an objective function cut $c^{\mathsf{T}}x + \theta \geq z^*$ (as discussed in Section 3.4), generating valid inequalities for the epigraph of f_C exhibits better empirical performance than the standard Benders' decomposition approach.

We are aware of two previous works that use basis information for two-stage stochastic programs (Gade et al., 2014; Romeijnders et al., 2016), both of which focus on the case with integrality constraints on the second-stage variables. In Gade et al. (2014) the authors use basis information to generate parametric Gomory cuts for the second-stage problem. In contrast, Romeijnders et al. (2016) use a convex approximation of the second-stage value function and leverage periodicity properties of value functions associated with corner polyhedra (maintaining integrality on the variables) to bound the approximation error. Despite also using basis information, both approaches are fundamentally different from our proposed method.

1.3 Network flow structures

While our overall framework applies generically, when we further assume that Y is a network flow polytope, we obtain additional theoretical insights and algorithmic enhancements. This is achieved by using the known combinatorial structure of bases of network flow problems (Ahuja et al., 1988), which allows us to accelerate the Benders' decomposition process and derivation of corner Benders' cuts (see Section 4 and Appendix F).

A network flow structure in the second-stage decisions is encountered frequently and captures important classes of problems. For example, it appears in numerous transportation and logistics applications that have been shown to benefit from Benders' decompo-

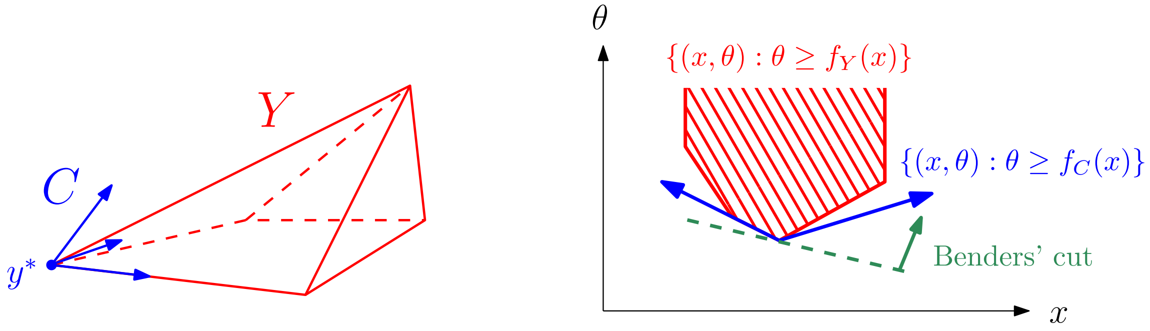

Y
C
y^*
θ
$\{(x,\theta) : \theta \geq f_Y(x)\}$
$\{(x,\theta) : \theta \geq f_C(x)\}$
Benders' cut
x

Figure 1: The basis associated to a vertex y^* of Y yields the translated cone (the relaxed corner polyhedron) $C \supseteq Y$, shown in the left panel. The facets of C can then be projected into the (x, θ)-domain, depicted in the right panel.

sition (Costa, 2005; Rahmaniani et al., 2017).

A network flow polytope can also be used to efficiently encode the convex hull of a finite set of vectors (de Lima et al., 2022). This principle underlies how *decision diagrams* can offer strong relaxations of combinatorial optimization problems (Becker et al., 2005; Behle, 2007; Bergman et al., 2016; Tjandraatmadja and van Hoeve, 2019; Castro et al., 2022; Lozano et al., 2022), which can be used to model second-stage problems for two-stage stochastic programs (Lozano and Smith, 2022).

The convex hull subproblem is also a fundamental aspect of *Dantzig-Wolfe (DW) decomposition*, for which the partitioning scheme is based on a complicating subsystem of constraints, rather than of the variables as in Benders' decomposition. Satisfying the complicating constraints can be rewritten as requiring that solutions can be expressed as a convex combination of extreme points (in the case that the subproblem is a polytope), which gives rise to a network flow structure, as we discuss in Section 4.1. This is reflected in (LP) by treating the x space as "original" problem variables and the y space as flow variables coming from a DW reformulation.

In this context, Chen et al. (2024) recently presented a strategy to replace an objective function cut that recovers the *DW bound* (i.e., the LP bound of a DW reformulation) by a family of cuts in the original variables that are not parallel to the objective function. In that sense, the approach of Chen et al. (2024) shares some of the same ideas as our work. In fact, our computational experiments are framed in the context of adding corner Benders' cuts based on DW decomposition bounds. Our approach of taking advantage of basis information, provides computational advantages compared to the method in Chen et al. (2024), which does not exploit that.

We emphasize that, as highlighted in Uchoa et al. (2024), a motivation for recovering the DW bound through cuts is that many commercial optimization solvers have native support for the addition of cuts, but not for the addition of variables needed in a branch-and-price method. Moreover, existing branch-cut-and-price frameworks (e.g., Jünger and Thienel (2000); Bolusani et al. (2024); Gamrath and Lübbecke (2010); Sadykov and Vanderbeck (2021)), remain less developed than the leading optimization solvers. In this sense, formulations that use only a fixed number of variables are easier to implement and can use the latest technology of commercial optimization solvers.

1.4 Paper organization

We proceed by first establishing central definitions and notation in Section 2. Section 3 contains our main results, introducing a new way of generating Benders' cuts via projected corner relaxations. Section 4 takes advantage of network flow structure and an explicit connection to DW reformulations, then specializes the methodology to the two-stage capacitated vehicle routing problem with stochastic demands (Gendreau et al., 2016), in which routes need to be decided in a first stage before customer demands are revealed, leading to second-stage recourse decisions. The computational results, discussed in Section 5, indicate that for instances with a large number of vehicles, corner Benders' cuts enhance the performance of a problem-specific, state-of-the-art branch-and-cut algorithm and compare favorably to the recently-proposed method by Chen et al. (2024).

2 Preliminaries: key notation and definitions

In this section, we establish the notation and briefly present the main mathematical tools that we use throughout the paper.

2.1 Supports and Fenchel cuts

The main goal of this work is to study valid inequalities for Problem (1), for which a useful point of view is to regard such inequalities as Fenchel cuts (Boyd, 1994). The concept of a Fenchel cut is fundamental: whenever we are given an optimization oracle over a set, we can use this oracle to compute inequalities *valid* (i.e., satisfied by all feasible points) for the set. In this way, Fenchel cuts arise as a natural consequence of fundamental results on the separation of convex sets (Hiriart-Urruty and Lemaréchal, 1996; Boyd and Vandenberghe, 2004), and led to the development of important tools for integer and stochastic programming, such as local cuts (Chvátal et al., 2013), Lagrangian cuts (Zou et al., 2019; Chen et al., 2024), and Fenchel decomposition (Ntaimo, 2013). We start with the following definition.

Definition 1. For $\mathcal{X} \subseteq \mathbb{R}^t$, the *support* of $\mathcal{X}$ is the function $\sigma_{\mathcal{X}} : \mathbb{R}^t \to \mathbb{R} \cup \{\pm\infty\}$ that sends each $\alpha \in \mathbb{R}^t$ to the value $\inf_{x \in \mathcal{X}}\{\alpha^\intercal x\}$.

Notice that by the definition of the infimum, $\sigma_{\mathcal{X}}(\alpha) = +\infty$ if $\mathcal{X}$ is empty, and $\sigma_{\mathcal{X}}(\alpha) = -\infty$ if $\{\alpha^\intercal x\}_{x \in \mathcal{X}}$ is not bounded below by a finite real number.

We make a couple of additional observations based on Definition 1. First, the standard support function used in convex analysis (Hiriart-Urruty and Lemaréchal, 1996; Cornuéjols and Lemaréchal, 2006) replaces the infimum by the supremum in Definition 1; however, we shall see that for the purposes of writing Benders' cuts, it will be more convenient to use the infimum. Second, if each element of $\mathcal{X}$ is identified with a tuple $(x, y) \in \mathbb{R}^t \times \mathbb{R}^{t'}$, then instead of writing $\sigma_{\mathcal{X}}((\alpha, \gamma))$ to refer to the value of $\inf_{(x,y) \in \mathcal{X}}\{\alpha^\intercal x + \gamma^\intercal y\}$, we use the shorthand $\sigma_{\mathcal{X}}(\alpha, \gamma)$.

With Definition 1 in hand, obtaining valid inequalities for $\mathcal{X}$ is immediate.

Definition 2. For any set $\mathcal{X} \subseteq \mathbb{R}^t$ and $\alpha \in \mathbb{R}^t$ such that $\sigma_{\mathcal{X}}(\alpha)$ is finite, an inequality (valid for $\mathcal{X}$) is said to be a *Fenchel cut* if it has the form $\alpha^\intercal x \geq \sigma_{\mathcal{X}}(\alpha)$.

Definition 2 shows that, if we have access to an optimization oracle over $\mathcal{X}$, then we can generate Fenchel cuts. As mentioned earlier, this is an old idea, and in the context of integer programming, it goes back to the seminal work of Boyd (1994), where the author generates cutting planes by calling an oracle that solves knapsack problems. In a similar spirit, we shall see in Section 3 that Benders' cuts can be interpreted as Fenchel cuts that use an optimization oracle over the higher-dimensional polyhedron Y.

2.2 Value functions and epigraphs

As discussed in Section 1, valid inequalities for Problem (1) can be derived using the concepts of value functions and epigraphs, which we formalize next. Recall that p denotes the number of rows in the linking constraints $Tx+Qy = h$. We *redefine* the value function f_Y in terms of a generic set $\mathcal{Y}$ (which may differ from polyhedron Y), a generalization that aids the discussion in Section 3. For convenience, we also use a change of variable from $x \in \mathbb{R}^n$ to $w \in \mathbb{R}^p$, representing (for fixed x values) the quantity $h - Tx$. This enables us to simplify how we define the domain and epigraph of the value function in Definition 4.

Definition 3. The *value function* with respect to a set $\mathcal{Y} \subseteq \mathbb{R}^m$ is $f_{\mathcal{Y}}(w) := \inf_{y\in\mathcal{Y}}\{d^\intercal y : Qy = w\}$, for any $w \in \mathbb{R}^p$.

Definition 4. Let $\mathcal{Y} \subseteq \mathbb{R}^m$. The *domain* and *epigraph* of f_Y are the sets $\textsc{dom}(f_{\mathcal{Y}}) := \{w \in \mathbb{R}^p : f_{\mathcal{Y}}(w) < +\infty\}$ and $\textsc{epi}(f_{\mathcal{Y}}) := \{(w, \theta) \in \mathbb{R}^p \times \mathbb{R} : \theta \geq f_{\mathcal{Y}}(w)\}$.

Let $\mathcal{F}$ denote the feasible region of Problem (1). Since, for $x \in X$, $\theta \geq \min_{y\in Y}\{d^\intercal y : Qy = h - Tx\}$ if and only if $\theta \geq f_Y(h - Tx)$, we may write (applying Definition 4 with $\mathcal{Y} = Y$)

$$\mathcal{F} = \{(x, \theta) : x \in X, (h - Tx, \theta) \in \textsc{epi}(f_Y)\}. \tag{$\mathcal{F}$}$$

Note that if $\alpha^\intercal w + \alpha_0\theta \geq \beta$ is valid for $\textsc{epi}(f_Y)$, then $\alpha^\intercal(h-Tx) + \alpha_0\theta \geq \beta$ is valid for $\mathcal{F}$.

2.3 Benders' cuts

The following is our definition of a Benders' cut.

Definition 5. An inequality $\alpha^\intercal w + \alpha_0\theta \geq \beta$ is a *Benders' cut* if it is valid for $\textsc{epi}(f_Y)$.

Since variable θ can increase arbitrarily in $\mathcal{F}$ (for any $x \in X$, $(x, \theta) \in \mathcal{F}$ for all $\theta \geq f_Y(x)$), we know that $\alpha_0 \geq 0$. The use of a generic coefficient α_0 in Definition 5 allows us to treat optimality and feasibility cuts in a unified way (similarly to the approach taken in Fischetti et al. (2010)): in the case that $\alpha_0 = 0$, we have that the Benders' feasibility cut $\alpha^\intercal w \geq \beta$ is a valid inequality for $\textsc{dom}(f_Y)$; otherwise, we may scale the inequality to get the optimality cut $\theta \geq \frac{1}{\alpha_0}(-\alpha^\intercal w + \beta)$.

Before we close this section, we make a few more notes on Definition 5. First, by linear programming duality we know that the epigraph of the value function f_Y is convex and closed. (In fact, Proposition 1.50 of Mordukhovich and Nam (2014) shows that $\textsc{epi}(f_{\mathcal{Y}})$ is convex and closed even for much more general choices of $\mathcal{Y} \subseteq \mathbb{R}^m$.) Therefore, $\textsc{epi}(f_Y)$ can be written as the intersection of the halfspaces defined by a set of Benders' cuts. Second, to separate Benders' cuts, it suffices to consider the coefficients (α, α_0) (ignoring β), since, if $\alpha^\intercal w + \alpha_0\theta \geq \beta$ is a Benders' cut, then β is at most $\sigma_{\textsc{epi}(f_Y)}(\alpha, \alpha_0)$. In other words, we may restrict our attention only to Fenchel cuts for $\textsc{epi}(f_Y)$.

3 Corner Benders' cuts

We are now ready to address how to use basis information to generate Benders' cuts more effectively, as illustrated in Figure 1. Consider the scenario where we solve a relaxation of Problem (1) to obtain a vector $(\bar{x}, \bar{\theta}) \in \mathbb{R}^{n+1}$. Suppose that we solve the separation problem for $\textsc{epi}(f_Y)$ (i.e., the *Benders' subproblem*) and we obtain an inequality (valid for $\textsc{epi}(f_Y)$) of the form

$$\alpha^\intercal w + \alpha_0 \theta \geq \beta \tag{2}$$

such that $\alpha^\intercal(h - T\bar{x}) + \alpha_0\bar{\theta} < \beta$. At this point, the standard Benders' decomposition method would add the cut $\alpha^\intercal(h - Tx) + \alpha_0\theta \geq \beta$ to the master problem and repeat the process by solving another relaxation of Problem (1).

Our main idea is to use the Benders' cut (2) to compute a set of inequalities that yield a tighter approximation of $\textsc{epi}(f_Y)$ than the single cut (2) alone. To accomplish this, we first define in Section 3.1 a conic relaxation C of polyhedron Y that we call a *corner*. Next, we show in Section 3.2 how to find a corner C such that every point in the epigraph of f_C satisfies the given Benders' cut (2) (see Figure 1). Rather than adding only a single optimality cut to the master problem, we propose in Section 3.3 to use an algorithm that can efficiently separate multiple facets of $\textsc{epi}(f_C)$. Finally, in Section 3.4 we establish the existence of a single corner C such that, in replacing f_Y by f_C in Problem (1), we still preserve its optimal value z^*.

3.1 Definitions

Henceforth, $\textsc{cone}(\cdot)$ is the conical-hull operator, and the summation of sets refers to the Minkowski sum. We first define a corner of Y as any conical relaxation whose apex lies at an extreme point of Y.

Definition 6. A set $C \subseteq \mathbb{R}^m$ is a *corner* of Y, if $C = \{y^*\} + \textsc{cone}(R)$ contains Y, where $y^* \in \mathbb{R}^m$ is an extreme point of Y and $R \subseteq \mathbb{R}^m$. If $C \subseteq \mathbb{R}^m$ is a corner of Y and $\sigma_C(\gamma)$ is finite, then C is said to be an *optimal corner* with respect to $\sigma_Y(\gamma)$.

The next result justifies the term "optimal corner".

Lemma 1. *If $C \subseteq \mathbb{R}^m$ is an optimal corner with respect to $\sigma_Y(\gamma)$, then $\sigma_C(\gamma) = \sigma_Y(\gamma)$.*

Proof. By the definition of a corner, $C = \{y^*\} + \textsc{cone}(R)$, where y^* is an extreme point of Y and $R \subseteq \mathbb{R}^m$. For every $\bar{y} \in C \supseteq Y$, we can write $\bar{y} = y^* + \sum_{r \in R} \mu_r\, r$, where $\mu_r \geq 0$, for all $r \in R$. Hence, $\gamma^\intercal \bar{y} = \gamma^\intercal y^* + \sum_{r \in R} \mu_r(\gamma^\intercal r) \geq \gamma^\intercal y^*$, where the last inequality follows because $\sigma_C(\gamma)$ is finite, so $\gamma^\intercal r \geq 0$, for every $r \in R$. Since y^* is a point in $Y \subseteq C$ that attains the lower bound of $\gamma^\intercal y^*$, we conclude that $\sigma_C(\gamma) = \sigma_Y(\gamma) = \gamma^\intercal y^*$. □

In Remark 1, we illustrate how we will construct corners in our experiments.

Remark 1. Suppose $Y = \{y \in \mathbb{R}^m : Ay = b, y \geq 0\}$ is a polyhedron in standard equality form. For each $j \in [m]$, we use A_j to denote the j-th column of A (so $Ay = \sum_{j \in [m]} A_j y_j$) and for a set $J \subseteq [m]$, we use A_J to denote the submatrix of A containing exactly the set of columns $\{A_j\}_{j \in J}$. Suppose that B is an optimal basis for the problem of minimizing $\gamma^\intercal y$ over Y. Let $N = [m] \setminus B$ and rewrite $Ay = b$ as the system $A_B y_B + A_N y_N = b$. Define $\bar{b} := (A_B)^{-1} b$ and $\bar{a}^j := (A_B)^{-1} A_j$, for each $j \in N$. Thus, we rewrite $Ay = b$

as $y_B = \bar{b} - \sum_{j \in N} \bar{a}^j y_j$. By the choice of B, $y^* = (y_B^*, y_N^*) = (\bar{b}, \mathbf{0})$ is an optimal basic feasible solution. Each nonbasic variable $j \in N$ gives a ray $r^j \in \mathbb{R}^m$ with entries

$$r_i^j = \begin{cases} -\bar{a}_i^j, & \text{if } i \in B, \\ 1, & \text{if } i = j, \\ 0, & \text{otherwise.} \end{cases}$$

Since each nonbasic variable has a nonnegative reduced cost, the set $C = \{y^*\} + \textsc{cone}(\{r^j\}_{j \in N})$ is an optimal corner with respect to $\sigma_Y(\gamma)$. □

Now that we have the definition of corners, we formalize what we mean by corner Benders' cuts.

Definition 7. An inequality $\alpha^\intercal w + \alpha_0 \theta \geq \beta$ is a *corner Benders' cut* if it is valid for $\textsc{epi}(f_C)$, where C is a corner of Y.

Note that since C is a relaxation of Y, it follows from the definition of value functions that $\textsc{epi}(f_C)$ contains $\textsc{epi}(f_Y)$; in other words, corner Benders' cuts are also Benders' cuts (i.e., they are valid inequalities for the epigraph of f_Y).

3.2 Finding an optimal corner from a Benders' cut

Our strategy will be to find an optimal corner based on a Benders' cut. To accomplish this, we prove Lemma 2, which shows that depending on the choice of the objective function, optimizing over a set $\mathcal{Y} \subseteq \mathbb{R}^m$ is equivalent to optimizing over the epigraph $\textsc{epi}(f_\mathcal{Y}) \subseteq \mathbb{R}^{p+1}$.

Lemma 2. *Let $\mathcal{Y} \subseteq \mathbb{R}^m$. For every $(\alpha, \alpha_0) \in \mathbb{R}^p \times \mathbb{R}_+$, it holds that*

$$\sigma_\mathcal{Y}(Q^\intercal \alpha + \alpha_0 d) = \sigma_{\textsc{epi}(f_\mathcal{Y})}(\alpha, \alpha_0).$$

Proof. Since $\alpha_0 \geq 0$, we may write

$$\begin{aligned}
\sigma_{\textsc{epi}(f_\mathcal{Y})}(\alpha, \alpha_0) &= \inf_{(w,\theta) \in \textsc{epi}(f_\mathcal{Y})} \{\alpha^\intercal w + \alpha_0 \theta\} \\
&= \inf_{w,\theta} \left\{ \alpha^\intercal w + \alpha_0 \theta : \theta \geq \inf_{y \in \mathcal{Y}} \{d^\intercal y : Qy = w\} \right\} \\
&= \inf_{w} \left\{ \alpha^\intercal w + \alpha_0 \inf_{y \in \mathcal{Y}} \{d^\intercal y : Qy = w\} \right\} && (3) \\
&= \inf_{w,y} \{\alpha^\intercal w + \alpha_0 (d^\intercal y) : Qy = w, y \in \mathcal{Y}\} && (4) \\
&= \inf_{y \in \mathcal{Y}} \{\alpha^\intercal (Qy) + \alpha_0 (d^\intercal y)\} \\
&= \sigma_\mathcal{Y}(Q^\intercal \alpha + \alpha_0 d).
\end{aligned}$$

The fourth equality holds because optimizing first over w and then over y is equivalent to jointly optimizing over (w, y) (see Section 4.1.3, "Optimizing over some variables", in Boyd and Vandenberghe (2004)). More explicitly, note that if we fix w to some arbitrary $\bar{w} \in \textsc{dom}(f_\mathcal{Y})$, both (3) and (4) reduce to $\alpha^\intercal \bar{w} + \alpha_0 \inf_{y \in \mathcal{Y}} \{d^\intercal y : Qy = \bar{w}\} = \alpha^\intercal \bar{w} + \alpha_0 f_\mathcal{Y}(\bar{w})$. □

The condition that $\alpha_0 \geq 0$ in Lemma 2 is required, as otherwise we might have that $\sigma_{\mathcal{Y}}(Q^\mathsf{T}\alpha + \alpha_0 d)$ is finite while $\sigma_{\text{EPI}(f_{\mathcal{Y}})}(\alpha, \alpha_0) = -\infty$.

By Lemma 1, we know that an optimal corner is a translated cone C that is a relaxation of polyhedron Y for which optimizing a given linear objective function attains the same optimal value. The next theorem tells us that the epigraphs of f_Y and f_C also have a similar property. Although this result may seem straightforward, it is crucial to our approach, since it shows that, given a Benders' cut $\alpha^\mathsf{T} w + \alpha_0\theta \geq \beta$, one can simply optimize over Y to find a corner C such that $\alpha^\mathsf{T} w + \alpha_0\theta \geq \beta$ is valid for $\text{EPI}(f_C)$.

Theorem 1. *Let C be a corner of Y. A Benders' cut $\alpha^\mathsf{T} w + \alpha_0\theta \geq \beta$ is valid for $\text{EPI}(f_C)$ if and only if C is an optimal corner with respect to $\sigma_Y(Q^\mathsf{T}\alpha + \alpha_0 d)$. Moreover, there exists a finite set $\mathcal{C}$ of corners of Y such that $\text{EPI}(f_Y) = \bigcap_{C' \in \mathcal{C}} \text{EPI}(f_{C'})$.*

Proof. Fix a Benders' cut $\alpha^\mathsf{T} w+\alpha_0\theta \geq \beta$ and note that, as we observed in Section 2.3, $\alpha_0 \geq 0$ and $\beta \leq \sigma_{\text{EPI}(f_Y)}(\alpha, \alpha_0)$. If $\alpha^\mathsf{T} w + \alpha_0\theta \geq \beta$ is valid for $\text{EPI}(f_C)$, we have that $-\infty < \beta \leq \sigma_{\text{EPI}(f_C)}(\alpha, \alpha_0) < +\infty$, where the last inequality follows from C being nonempty. Hence, by Lemma 2, $\sigma_C(Q^\mathsf{T}\alpha + \alpha_0 d)$ is also finite. So, by Definition 6, the set C is an optimal corner with respect to $\sigma_Y(Q^\mathsf{T}\alpha + \alpha_0 d)$.

Conversely, let $C \subseteq \mathbb{R}^m$ be an optimal corner with respect to $\sigma_Y(Q^\mathsf{T}\alpha + \alpha_0 d)$. Then

$$\sigma_{\text{EPI}(f_Y)}(\alpha, \alpha_0) \overset{\text{Lemma 2}}{=} \sigma_Y(Q^\mathsf{T}\alpha + \alpha_0 d) \overset{\text{Lemma 1}}{=} \sigma_C(Q^\mathsf{T}\alpha + \alpha_0 d) \overset{\text{Lemma 2}}{=} \sigma_{\text{EPI}(f_C)}(\alpha, \alpha_0),$$

which proves that $\beta \leq \sigma_{\text{EPI}(f_C)}(\alpha, \alpha_0)$, so $\alpha^\mathsf{T} w + \alpha_0\theta \geq \beta$ is valid for $\text{EPI}(f_C)$.

To prove the second part of the statement, let $\mathcal{C}$ be the set of corners associated with every feasible basis of Y (see, for example, Remark 1). It is clear that $\mathcal{C}$ is finite. Moreover, for every $C' \in \mathcal{C}$, we have $\text{EPI}(f_Y) \subseteq \text{EPI}(f_{C'})$. Indeed, since $Y \subseteq C'$, for any $(w, \theta) \in \text{EPI}(f_Y)$, Definition 3 implies that $\theta \geq f_Y(w) \geq f_{C'}(w)$, meaning that $(w, \theta) \in \text{EPI}(f_{C'})$. This argument shows that $\text{EPI}(f_Y) \subseteq \bigcap_{C' \in \mathcal{C}} \text{EPI}(f_{C'})$. But we have just proven that any valid inequality $\alpha^\mathsf{T} w+\alpha_0\theta \geq \beta$ for $\text{EPI}(f_Y)$ is also valid for $\text{EPI}(f_{C'})$, where C' is an optimal corner with respect to $\sigma_Y(Q^\mathsf{T}\alpha + \alpha_0 d)$. In particular, by choosing the basis appropriately, we may assume that C' belongs to the set $\mathcal{C}$, proving the reverse inclusion. □

3.3 Separating corner Benders' cuts via polarity

Throughout this subsection, we fix a corner $C = \{y^*\} + \text{CONE}(R)$, and we consider the problem of generating strong valid inequalities for $\text{EPI}(f_C)$. Recall that an *inner description* (also called $\mathcal{V}$-polyhedral, see Ziegler (2012) and Balas and Kazachkov (2025)) of a polyhedral set is given by the Minkowski sum of the convex hull of a set of points with the conical hull of a set of rays. Since we have access to an inner description of C, we immediately obtain an inner description of $\text{EPI}(f_C)$.

Lemma 3. *Let $C = \{y^*\} + \text{CONE}(R)$ be a corner of Y with R finite and let $(w^*, \theta^*) = (Qy^*, d^\mathsf{T} y^*)$. Then*

$$\text{EPI}(f_C) = \{(w^*, \theta^*)\} + \text{CONE}(\{(Qr, d^\mathsf{T} r)\}_{r\in R} \cup \{(\mathbf{0}, 1)\}).$$

Proof. By the inner description of C and the definition of the epigraph of f_C,

$$\begin{aligned}\textsc{epi}(f_C) =&\{(w,\theta) : \theta \geq \inf_{y\in C}\{d^\intercal y : Qy = w\}\}\\ =&\{(w,\theta) : \theta \geq \min_{y\in C}\{d^\intercal y : Qy = w\}, w \in \textsc{dom}(f_C)\}\\ =&\{(Qy, d^\intercal y) + \mu_0(\mathbf{0},1) : y \in C, \mu_0 \geq 0\}\\ =&\left\{\left(Qy^* + \sum_{r\in R}\mu_r(Qr), d^\intercal y^* + \mu_0 + \sum_{r\in R}\mu_r(d^\intercal r)\right) : \mu_i \geq 0,\ \forall i \in R \cup \{0\}\right\},\end{aligned}$$

where the second equality holds since R is finite, so for every $w \in \textsc{dom}(f_C)$, the set $\{y \in C : Qy = w\}$ is polyhedral. □

The finiteness of R in Lemma 3 is required, since otherwise we may consider the following example. Assume that $(w^*,\theta^*) = (0,0) \in \mathbb{R}^2$ and $R = \{(y_1, y_2, y_1/y_2)\}_{y_1,y_2\in[1,+\infty)}$, with $Qr = y_1$ and $d^\intercal r = y_1/y_2$, for each $r = (y_1, y_2, y_1/y_2) \in R$. For any $\bar{y}_1 \in [1,+\infty)$, we have $\inf_{y\in C}\{d^\intercal y : Qy = \bar{y}_1\} = 0$, so $(\bar{y}_1, 0) \in \textsc{epi}(f_C)$. However, since $d^\intercal r > 0$ for every $r \in R$, the point $(\bar{y}_1, 0)$ does not belong to $\{(w^*,\theta^*)\}+\textsc{cone}(\{(Qr, d^\intercal r)\}_{r\in R}\cup\{(\mathbf{0},1)\})$.

Suppose now that we are given a candidate point $(\bar{w}, \bar{\theta})$ that we wish to separate from $\textsc{epi}(f_C)$. With Lemma 3 in hand, we can do this by searching for an infeasibility certificate for the system

$$\min_{\mu\geq 0}\left\{0 : \sum_{r\in R}(Qr)\mu_r = \bar{w},\ \sum_{r\in R}(d^\intercal r)\mu_r \leq \bar{\theta}\right\}. \tag{5}$$

(In Appendix B, we show a connection between this approach and the method of Chen and Luedtke (2022) to generate Lagrangian cuts by optimizing over restricted dual problems.) Notice, however, that whenever (5) is infeasible, its dual is unbounded, and therefore one would typically have to choose among different normalizations (Balas et al., 1993; Louveaux et al., 2015; Serra, 2020). Here we propose to separate $(\bar{w}, \bar{\theta})$ from $\textsc{epi}(f_C)$ by normalizing the cut-generating linear program through known ideas from *polarity* (Balas, 1979; Schrijver, 1998; Cadoux and Lemaréchal, 2013).

To do so, we first translate $\textsc{epi}(f_C)$ so that the origin lies in the relative interior of the set. In other words, we take a point (w', θ') in the relative interior of $\textsc{epi}(f_C)$ (this can be found by projecting a point in the relative interior of Y) and we consider the set

$$E := \textsc{epi}(f_C) - (w',\theta') = \{(w^*,\theta^*) - (w',\theta')\} + \textsc{cone}(\{(Qr, d^\intercal r)\}_{r\in R} \cup \{(\mathbf{0},1)\}),$$

where $w^* = Qy^*$ and $\theta^* = d^\intercal y^*$. We then consider the following particular case of a *reverse polar set of* E (see Balas (1979)):

$$E^{\#} := \left\{ (\alpha,\alpha_0) \in \mathbb{R}^{p+1} : \begin{array}{ll} \alpha^\intercal(w^* - w') + \alpha_0(\theta^* - \theta') \geq -1, & \\ \alpha^\intercal(Qr) + \alpha_0(d^\intercal r) \geq 0, & \forall r \in R \\ \alpha_0 \geq 0, & \end{array}\right\}, \tag{6}$$

and we refer to the inequalities associated with vectors in R as *ray inequalities* of $E^{\#}$.

Observe that, as pointed out by Balas (1979), the standard polar set of E is given by $E^\circ = -E^{\#}$ (see, for instance, Theorem 9.1 of Schrijver (1998)). We use the reverse polar $E^{\#}$ simply to be consistent with our choice of expressing Benders' cuts in the form $\alpha^\intercal w + \alpha_0\theta \geq \beta$ rather than $\alpha^\intercal w + \alpha_0\theta \leq \beta$.

Let $\alpha^\intercal w + \alpha_0\theta \geq \beta$ be a valid inequality for E that is not an implicit equality. Since the origin lies in the relative interior of E, we know that $\beta < 0$ and by scaling, we may assume that $\beta = -1$. It can now be easily verified that (α, α_0) belongs to $E^\#$. In fact, when E is full dimensional, it is known that $E^\#$ is bounded, with its extreme points corresponding to facets of E (see Schrijver (1998) and Cadoux and Lemaréchal (2013)), which in turn translate into facets of $\textsc{epi}(f_C)$. In our case, however, the set E is not necessarily full dimensional, and we need to consider the *recession cone* of $E^\#$, denoted $\textsc{recc}(E^\#)$. Given our use of the reverse polar $E^\#$ and this additional technical detail on its boundedness, we state Theorem 2 below. (This result follows from standard arguments on polarity, and we leave the proof to Appendix A.)

Theorem 2. *Let $C = \{y^*\} + \textsc{cone}(R)$ be a corner of Y. Let $(w^*, \theta^*) = (Qy^*, d^\intercal y^*)$, (w', θ') be a point in the relative interior of $\textsc{epi}(f_C)$, and $(\bar{w}, \bar{\theta})$ be an arbitrary point in $\mathbb{R}^p \times \mathbb{R}$. Define $E^\#$ as in (6) and consider the optimization problem*

$$z^\circ := \min_{\alpha,\alpha_0} \left\{ \alpha^\intercal(\bar{w} - w') + \alpha_0(\bar{\theta} - \theta') : (\alpha, \alpha_0) \in E^\# \right\}. \tag{7}$$

It holds that

(a) if $z^\circ \geq -1$, then $(\bar{w}, \bar{\theta})$ belongs to $\textsc{epi}(f_C)$;

(b) if $z^\circ = -\infty$ (i.e., Problem (7) is unbounded), then we have a vector $(\alpha, \alpha_0) \in \textsc{recc}(E^\#)$ such that $\alpha^\intercal w + \alpha_0\theta = \alpha^\intercal w' + \alpha_0\theta'$ defines an implicit equality of $\textsc{epi}(f_C)$ that is violated by $(\bar{w}, \bar{\theta})$;

(c) otherwise, $-\infty < z^\circ < -1$ and we have an extreme point (α, α_0) of $E^\#$ such that the inequality $\alpha^\intercal w + \alpha_0\theta \geq -1 + \alpha^\intercal w' + \alpha_0\theta'$ defines a facet of $\textsc{epi}(f_C)$ that is violated by $(\bar{w}, \bar{\theta})$.

Row generation of ray inequalities. We solve (7) via Algorithm 1 (SolveReversePolar). We adopt a row generation approach for the ray inequalities in $E^\#$: in each iteration, we solve a relaxed variant of (7), which we call (CGLP) in the algorithm, over constraints for only a partial set of rays. This is because, due to the projection $r \to (Qr, d^\intercal r)$, several of the ray inequalities in $E^\#$ may be redundant. For example, if $r^1, r^2 \in R$ satisfy $Qr^1 = Qr^2$ and $0 < d^\intercal r^1 < d^\intercal r^2$, then $(Qr^2, d^\intercal r^2)$ is not an extreme ray of $\textsc{epi}(f_C)$. Consequently, the ray inequality for r^2 in $E^\#$ is redundant.

The (CGLP) that we will solve contains ray inequalities for two sets of rays: "current" rays R' and "warm-start" rays $\tilde{R}$. In each iteration, the set R' will be augmented with rays from a set R'' (line 12 of SolveReversePolar), which contains the rays from R that correspond to violated ray inequalities with respect to the current solution $(\bar{\alpha}, \bar{\alpha}_0)$. There are various strategies for selecting rays R' from R'' based on which ray inequalities we want to impose for Problem (CGLP). For example, one could choose to add a fixed number of inequalities with the maximum violation. We tailor this to our specific formulation of the vehicle routing problem with stochastic demands (see Section 5.2 for further details).

The "warm-start" ray set $\tilde{R}$ is passed as an argument to SolveReversePolar, which is repeatedly called as a subroutine in Algorithm 2 (SeparateCornerBendersCuts). The procedure SeparateCornerBendersCuts begins by using Theorem 1 to build an optimal corner C from a Benders' cut $\alpha^\intercal w + \alpha_0\theta \geq \sigma_{\textsc{epi}(f_Y)}(\alpha, \alpha_0)$. Then, in line 8, SolveReversePolar is called to find violated corner Benders' cuts with respect to $\textsc{epi}(f_C)$. The "warm-start" rays $\tilde{R}$ initialize Problem (CGLP) with ray inequalities that were active for cuts generated in prior iterations, collected in SolveReversePolar at line 20.

Algorithm 1 SolveReversePolar

Input: Candidate solution $(\bar{x}, \bar{\theta}) \in \mathbb{R}^{n+1}$; point (w', θ') in the relative interior of $\textsc{epi}(f_Y)$; a corner $C = \{y^*\} + \textsc{cone}(R)$ given as point $y^* \in \mathbb{R}^p$ and finite set $R \subseteq \mathbb{R}^p$; and set $\tilde{R} \subseteq R$ of "warm-start" rays.

Output: Optimal value z° of Theorem 2; tuple (ρ, ρ_0, β) (representing a corner Benders' cut of the form $\rho^\top w + \rho_0 \theta \geq \beta$); and updated set of "warm-start" rays.

```
procedure SolveReversePolar($(\bar{x}, \bar{\theta}), (w', \theta'), y^*, R, \tilde{R}$)
    Set $(w^*, \theta^*) = (Qy^*, d^\top y^*)$ and $\bar{w} = h - T\bar{x}$.
    Set $R' = \emptyset$ and start Problem (CGLP) with the linear program below.

    $$\begin{array}{llll} \text{(CGLP)} \quad z^\circ = \min\limits_{\alpha, \alpha_0} & \alpha^\top(\bar{w} - w') + \alpha_0(\bar{\theta} - \theta') & & \\ & \alpha^\top(w^* - w') + \alpha_0(\theta^* - \theta') \geq -1, & & (\xi) \\ & \alpha^\top(Qr) + \alpha_0(d^\top r) \geq 0, & \forall r \in R' \cup \tilde{R}, & (\mu_r) \\ & \alpha_0 \geq 0. & & (\mu_0) \end{array}$$

    $(\bar{\alpha}, \bar{\alpha}_0) \leftarrow (\mathbf{0}, 0)$
    repeat
        $R'' \leftarrow \emptyset$
        Solve Problem (CGLP) to get $z^\circ$ and update $(\bar{\alpha}, \bar{\alpha}_0)$ to the obtained extreme point/ray.
        for $r \in R$ do
            if $\bar{\alpha}^\top(Qr) + \bar{\alpha}_0(d^\top r) < 0$ then
                $R'' \leftarrow R'' \cup \{r\}$
        if $R'' \neq \emptyset$ then
            Choose some rays $r \in R''$ and add them to $R'$ (effectively adding the corresponding inequalities $\alpha^\top(Qr) + \alpha_0(d^\top r) \geq 0$ to Problem (CGLP)).
    until $R'' = \emptyset$
    if $z^\circ \geq -1$ then
        return $((\mathbf{0}, 0, 0), z^\circ, \tilde{R})$.
    else if $z^\circ = -\infty$ then
        return $((\bar{\alpha}, \bar{\alpha}_0, \beta = \bar{\alpha}^\top w' + \bar{\alpha}_0 \theta'), z^\circ, \tilde{R})$.
    else
        Let $(\bar{\xi}, \bar{\mu})$ be optimal dual multipliers associated with the current optimal solution $(\bar{\alpha}, \bar{\alpha}_0)$.
        $\tilde{R}' \leftarrow \{r \in R' : \bar{\mu}_r > 0\}$.
        return $((\bar{\alpha}, \bar{\alpha}_0, \beta = -1 + \bar{\alpha}^\top w' + \bar{\alpha}_0 \theta'), z^\circ, \tilde{R} \cup \tilde{R}')$.
```

Algorithm 2 SeparateCornerBendersCuts

Input: Set $\tilde{\mathcal{F}}$ that is a relaxation of the feasible region $\mathcal{F}$; point y' that belongs to the relative interior of Y; and vector $(\alpha, \alpha_0) \in \mathbb{R}^{p+1}$ (representing a Benders' cut of the form $\alpha^\intercal w + \alpha_0 \theta \geq \sigma_{\text{EPI}(f_Y)}(\alpha, \alpha_0)$).

Output: Corner C of Y and finite set of vectors $\Omega \subseteq \mathbb{R}^{p+2}$ (where each element $(\rho, \rho_0, \beta) \in \Omega$, represents a corner Benders' cut of the form $\rho^\intercal w + \rho_0 \theta \geq \beta$).

```
procedure SeparateCornerBendersCuts($\tilde{\mathcal{F}}, y', \alpha, \alpha_0$)
    Set $C = \{y^*\} + \text{cone}(R)$ to be an optimal corner w.r.t. $\sigma_Y(Q^\intercal \alpha + \alpha_0 d)$ (for
    example, as described in Remark 1).
    Set $(w', \theta') = (Qy', d^\intercal y' + \varepsilon)$, where $\varepsilon > 0$ is an arbitrary constant.
    $\Omega \leftarrow \emptyset$
    $\tilde{\mathcal{R}} \leftarrow \emptyset$
    repeat
        Solve problem (M) below to get a candidate solution $(\bar{x}, \bar{\theta})$.

            (M)    $\min_{x,\theta} \quad c^\intercal x + \theta$
                   $(x, \theta) \in \tilde{\mathcal{F}},$
                   $\rho^\intercal (h - Tx) + \rho_0 \theta \geq \beta, \qquad \forall (\rho, \rho_0, \beta) \in \Omega.$

        $((\rho, \rho_0, \beta), z^\circ, \tilde{R}) \leftarrow$ SolveReversePolar($(\bar{x}, \bar{\theta}), (w', \theta'), y^*, R, \tilde{R}$)
        if $z^\circ = -\infty$ then
            $\mathcal{C} \leftarrow \mathcal{C} \cup \{(\rho, \rho_0, \beta), (-\rho, -\rho_0, -\beta)\}$
        else if $z^\circ < -1$ then
            $\Omega \leftarrow \Omega \cup \{(\rho, \rho_0, \beta)\}$
    until $z^\circ \geq -1$
    return $\Omega$
```

While we cannot guarantee that the rays in the "warm-start" set $\tilde{R}$ are extreme, we next show that, in a sense, at least these rays belong to the boundary of $\textsc{epi}(f_C)$.

Claim 1. *Let $((\bar{\alpha}, \bar{\alpha}_0, \beta), z^\circ, \tilde{R} \cup \tilde{R}')$ be the output of* SolveReversePolar *(with $\tilde{R}'$ being set as in line 20 of Algorithm 1). Suppose that $-\infty < z^\circ < -1$ and consider the facet $F = \textsc{epi}(f_C) \cap \{(w, \theta) : \bar{\alpha}^\intercal w + \bar{\alpha}_0 \theta = \beta\}$. Then, for every $r \in \tilde{R}'$, $(w^* + Qr, \theta^* + d^\intercal r)$ belongs to F.*

Proof. Fix $r \in \tilde{R}'$. Since $(Qr, d^\intercal r)$ is a recession direction of $\textsc{epi}(f_C)$, we know that $(w^* + Qr, \theta^* + d^\intercal r)$ belongs to $\textsc{epi}(f_C)$. Since $\bar{\mu}_r > 0$, we have by complementary slackness that $\bar{\alpha}^\intercal (Qr) + \bar{\alpha}_0 (d^\intercal r) = 0$. Combining this with the fact that $(w^*, \theta^*) \in F$, we conclude that $\bar{\alpha}^\intercal (w^* + Qr) + \bar{\alpha}_0 (\theta^* + d^\intercal r) = \beta$. □

3.4 Finding a corner that recovers the optimal bound

The previous subsections contain the main message of this paper: given a Benders' cut, we find an optimal corner C using Theorem 1 and we then separate corner Benders' cuts using Theorem 2 and Algorithms 2 and 1. The goal of this subsection is to show that by generating corner Benders' cuts for a fixed $\textsc{epi}(f_C)$, we can already recover the optimal bound z^* in Problem (1). More precisely, we prove the following result.

Proposition 1. *Let z^* be the optimal value of Problem (1) and let $\hat{\alpha}$ be an optimal solution to*

$$\max_{\alpha \in \mathbb{R}^p} \left\{ -\alpha^\intercal h + \sigma_X(c + T^\intercal \alpha) + \sigma_Y(Q^\intercal \alpha + d) \right\}. \tag{8}$$

If C is an optimal corner with respect to $\sigma_Y(Q^\intercal \hat{\alpha} + d)$, then it holds that

$$\min_{x,\theta} \left\{ c^\intercal x + \theta : x \in X, (h - Tx, \theta) \in \textsc{epi}(f_C) \right\} = z^*. \tag{9}$$

Before we dive into the proof of Proposition 1, let us discuss the relevance of this result and its connections to the existing literature. It is immediate that the *objective function cut* $c^\intercal x + \theta \geq z^*$ recovers the optimal bound z^*. Recall, however, that our overall goal is to accelerate branch-and-bound MIP solvers that repeatedly solve relaxations that take the form of the linear program shown in Problem (1). In this sense, it is often detrimental to a MIP solver to impose the objective function cut at the root node of a branch-and-bound tree (Gamrath and Lübbecke, 2010). Chen et al. (2024) argue that an explanation for this poorer performance is that adding the objective function cut to the LP relaxation usually leads to an optimal face with almost the same dimension as the original polyhedron, leading to performance degradation due to dual degeneracy. In contrast, Proposition 1 and Algorithm 2 allow us to decompose the objective function cut into multiple corner Benders' cuts; and this might lead to lower-dimensional optimal faces. Indeed, as we later show in our computational experiments, when taking Problem (1) to be the root node of a branch-and-bound tree, the approach based on Proposition 1 outperforms both the objective function cut and a single Benders' cut method inspired by the work of Chen et al. (2024) (see Theorem 3 next).

Let us now address the proof of Proposition 1. First, we show that, perhaps not surprisingly, separating a Fenchel cut for $\mathcal{F}$ (the feasible region of Problem (1)) amounts to separating a Fenchel cut for X and another Fenchel cut for $\textsc{epi}(f_Y)$. (In Appendix C, we further show that Theorem 3 can be easily adapted to the case where Problem (LP) has a "block-diagonal" structure.)

Theorem 3. *Let $(\rho, \rho_0) \in \mathbb{R}^n \times \mathbb{R}_+$ and assume $\sigma_{\mathcal{F}}(\rho, \rho_0)$ is finite. Let $\hat{\alpha} \in \mathbb{R}^p$ be optimal for the problem*

$$\max_{\alpha \in \mathbb{R}^p} \{-\alpha^\intercal h + \sigma_X(\rho + T^\intercal \alpha) + \sigma_Y(Q^\intercal \alpha + \rho_0 d)\}. \tag{10}$$

Then the following hold.

(i) $(\rho + T^\intercal \hat{\alpha})^\intercal x \geq \sigma_X(\rho + T^\intercal \hat{\alpha})$ *is valid for* X*;*

(ii) $\hat{\alpha}^\intercal(h - Tx) + \rho_0 \theta \geq \sigma_{\text{EPI}(f_Y)}(\hat{\alpha}, \rho_0)$ *is valid for* $\text{EPI}(f_Y)$*; and*

(iii) $\rho^\intercal x + \rho_0 \theta \geq \sigma_{\mathcal{F}}(\rho, \rho_0)$ *is implied by the inequalities in items (i) and (ii).*

Proof. By summing the inequalities in (i) and (ii), it suffices to show that $\sigma_{\mathcal{F}}(\rho, \rho_0) = -\hat{\alpha}^\intercal h + \sigma_X(\rho + T^\intercal \hat{\alpha}) + \sigma_{\text{EPI}(f_Y)}(\hat{\alpha}, \rho_0)$. Since X and Y are nonempty polyhedra, it follows from Lagrangian duality (Boyd and Vandenberghe, 2004) that

$$\begin{aligned}
\sigma_{\mathcal{F}}(\rho, \rho_0) = \min_{x,y} \quad & \rho^\intercal x + \rho_0(d^\intercal y) \\
& h - Tx = Qy, \\
& (x, y) \in X \times Y, \\
= \max_{\alpha \in \mathbb{R}^p} \min_{x,y} \quad & \rho^\intercal x + \rho_0(d^\intercal y) + \alpha^\intercal(Qy - (h - Tx)) \\
& (x, y) \in X \times Y, \\
= \max_{\alpha \in \mathbb{R}^p} & \{-\alpha^\intercal h + \sigma_X(\rho + T^\intercal \alpha) + \sigma_Y(Q^\intercal \alpha + \rho_0 d)\} \\
= & -\hat{\alpha}^\intercal h + \sigma_X(\rho + T^\intercal \hat{\alpha}) + \sigma_Y(Q^\intercal \hat{\alpha} + \rho_0 d) \qquad \text{(by optimality of } \hat{\alpha}) \\
= & -\hat{\alpha}^\intercal h + \sigma_X(\rho + T^\intercal \hat{\alpha}) + \sigma_{\text{EPI}(f_Y)}(\hat{\alpha}, \rho_0). \qquad \text{(by Lemma 2)}
\end{aligned}$$

□

Based on Theorem 3, we henceforth say that a *Lagrangian cut* is an inequality of the form $\hat{\alpha}^\intercal(h - Tx) + \theta \geq \sigma_{\text{EPI}(f_Y)}(\hat{\alpha}, 1)$ where $\hat{\alpha} \in \mathbb{R}^p$ is optimal for the Lagrangian dual problem (10), with $\rho = c$ and $\rho_0 = 1$. Therefore, a Lagrangian cut implies the objective function cut $c^\intercal x + \theta \geq \sigma_{\mathcal{F}}(c, 1) = z^*$. Proving Proposition 1 is now immediate.

Proof of Proposition 1. Since $\text{EPI}(f_C)$ is a relaxation of $\text{EPI}(f_Y)$, the optimal value of the problem in the left-hand side of (9) is at most z^*. For the converse, apply Theorem 1 with the Lagrangian cut $\hat{\alpha}^\intercal(h - Tx) + \theta \geq \sigma_{\text{EPI}(f_Y)}(\hat{\alpha}, 1)$. □

4 Network flow structure and the Dantzig-Wolfe bound

In this section, we focus on the special case where the inequalities defining Y arise from a network flow formulation. We start with some basic definitions. As usual (Cook et al., 2011; Ahuja et al., 1988), given a finite set $\mathcal{S}$, we index vectors by the elements of the set: for a set $\mathbb{D}$ (e.g., $\mathbb{R}$ or $\mathbb{Z}$), we denote by $\mathbb{D}^{\mathcal{S}}$ a vector indexed by $\mathcal{S}$ where each entry has domain $\mathbb{D}$.

Definition 8. A *network* is a tuple $(\mathcal{N}, b, u)$, where $\mathcal{N}$ is a directed graph identified by a tuple $(\mathcal{V}, \mathcal{A})$ of its nodes $\mathcal{V}$ and arcs $\mathcal{A}$; $b \in \mathbb{Z}^{\mathcal{V}}$ specifies the demand/supply values at each node $v \in \mathcal{V}$; and $u \in \mathbb{Z}^{\mathcal{A}}$ is a vector of arc capacities. Let $\mathbf{N} \in \mathbb{R}^{\mathcal{V} \times \mathcal{A}}$ denote the node-arc incidence matrix of $\mathcal{N}$, that is, for each $v \in \mathcal{V}$ and $a \in \mathcal{A}$, the entry $\mathbf{N}_{v,a}$ has value $+1$ if v is the head of a; -1 if v is the tail of a; and 0 otherwise.

Definition 9. Let $(\mathcal{N} = (\mathcal{V}, \mathcal{A}), b, u)$ be a network. Then $\mathcal{Y} \subseteq \mathbb{R}^m$ is a *network flow polytope associated with* $(\mathcal{N}, b, u)$ if it holds that $\mathcal{Y} = \left\{y \in \mathbb{R}^{\mathcal{A}}_{+} : \mathbf{N}y = b, y \leq u\right\}$.

Without loss of generality, we will assume that every network $\mathcal{N}$ mentioned henceforth is connected.

In the remainder of this section, we consider value functions $f_{\mathcal{Y}}$, where $\mathcal{Y}$ is a network flow polytope associated with a network $(\mathcal{N}, b, u)$. Our goal here is threefold: in Section 4.1, we apply the flow-decomposition theorem to establish a connection between such value functions and DW reformulations; in Section 4.2, we observe that corner Benders' cuts can recover the bound from the DW formulation; and in Sections 4.3 and 4.4, we formulate a DW-based linear programming relaxation for the Vehicle Routing Problem with Stochastic Demands (VRPSD) in the form of Problem (1), and we offer a simple VRPSD example to illustrate the generation of corner Benders' cuts.

4.1 Equivalence between path-flow and arc-flow formulations

Let Π be a finite set of vectors, which we interpret as the incidence vectors of feasible "patterns" for a combinatorial optimization problem, e.g., a route for a vehicle routing problem. For simplicity, we assume that Π is a subset of $\mathbb{R}^n$, but one can check that the following reasoning also holds for the general case of $\Pi \subseteq \mathbb{R}^p$ (as long as we later choose matrices Q and T appropriately). Next, we follow the presentation in de Lima et al. (2022) to show that network flows can be used to optimize over the set $X \cap (k \cdot \textsc{conv}(\Pi))$, where $k \in \mathbb{Z}_{++}$ denotes the number of patterns to be selected and $k \cdot \textsc{conv}(\Pi) := \{k\pi : \pi \in \textsc{conv}(\Pi)\}$. For each vector $\pi \in \Pi$, let $\hat{d}_\pi$ be a rational number representing the "cost" of using the pattern associated with π. We can minimize a linear function over $x \in X \cap (k \cdot \textsc{conv}(\Pi))$ as follows.

$$
\begin{aligned}
\hat{z} := \min_{x,\lambda} \quad & c^{\intercal}x + \sum_{\pi \in \Pi} \hat{d}_\pi \cdot \lambda_\pi \\
& x = \sum_{\pi \in \Pi} \pi \cdot \lambda_\pi, \\
& \sum_{\pi \in \Pi} \lambda_\pi = k, \\
& x \in X, \\
& \lambda_\pi \geq 0, \qquad\qquad \forall \pi \in \Pi.
\end{aligned}
\tag{11}
$$

Typically, Problem (11) is solved in a *column generation* framework, where the variables λ_π are added iteratively via a *pricing subproblem.*

One can always construct an acyclic directed graph $\mathcal{N} = (\mathcal{V}, \mathcal{A})$, with source vertex $s \in \mathcal{V}$ and sink vertex $t \in \mathcal{V}\backslash\{s\}$, such that there exists a one-to-one mapping between the $s-t$ paths in $\mathcal{N}$ and the elements of Π. Furthermore, we can set $d \in \mathbb{R}^{\mathcal{A}}$ and $Q \in \mathbb{R}^{n \times \mathcal{A}}$ so that, for each $\pi \in \Pi$, with corresponding $s-t$ path P, we have that $\sum_{a \in P} d_a = \hat{d}_\pi$ and $\sum_{a \in P} Q_a = \pi$ (the summation is over the arcs in P and $Q_a \in \mathbb{R}^n$ is the column of Q indexed by $a \in \mathcal{A}$). This can be accomplished trivially by setting $\mathcal{A}$ to be made of $|\Pi|$ parallel arcs between s and t. More interestingly, whenever optimizing a linear function over Π can be done with a *dynamic programming* (DP) algorithm $\mathbb{A}$, we can set the vertices (respectively, arcs) of $\mathcal{N}$ to correspond to the DP states (respectively, DP state transitions) of algorithm $\mathbb{A}$. In this way, the $s-t$ paths in $\mathcal{N}$ correspond to feasible outputs of $\mathbb{A}$, and therefore, to elements of Π (see de Lima et al. (2022) for more details and examples).

The previous argument shows that we can interpret Problem (11) as a *path-flow* formulation, where each variable λ_π corresponds to a flow variable over an $s-t$ path in $\mathcal{N}$. Since $\mathcal{N}$ is acyclic, we can apply the flow decomposition theorem (see Section 3.5 of Ahuja et al. (1988)) to map each path-flow solution $\lambda \in \mathbb{R}^{\Pi}$ to a corresponding *arc-flow* solution $y \in \mathbb{R}^{\mathcal{A}}$ and vice versa. For any $v \in \mathcal{V}$, let $\mathbb{1}_v \in \mathbb{R}^{\mathcal{V}}$ be the characteristic vector of $\{v\}$. It follows that, with Y representing the network flow polytope associated with $\mathcal{N}$, the optimal value $\hat{z}$ of Problem (11) can be equivalently obtained by

$$\min_{x,y} \left\{ c^\intercal x + d^\intercal y : x = Qy,\, x \in X,\, y \in Y = \{ y \in \mathbb{R}^{\mathcal{A}}_{+} : \mathbf{N}y = k \cdot \mathbb{1}_t - k \cdot \mathbb{1}_s \} \right\}. \tag{12}$$

Hence, with $h = 0$ and T as the negative identity matrix, via Problem (1), we can attain the same bound $\hat{z}$, even if we are only allowed to use network flow polytopes for Y. As a result, in what follows, we return to the x variable space because $x = w$ in our setting.

We also mention in passing that this path-flow/arc-flow equivalence also relates our work with important studies on cut generation via decision diagrams (Becker et al., 2005; Behle, 2007; Tjandraatmadja and van Hoeve, 2019; Castro et al., 2022; Lozano and Smith, 2022), but in Section 4.2 we position our contributions in the context of DW formulations.

4.2 Recovering the DW bound

Consider the setting where the set Π is given by the set of extreme points of a polytope. In this case, Problem (11) encompasses different modeling paradigms for DW formulations, and the value $\hat{z}$ corresponds to the so-called *DW bound.* For example, when $\hat{d} \equiv 0$, then the problem falls in a format similar to the *explicit master problem* in de Aragao and Uchoa (2003) and also the general DW decomposition formulation (such as in Chen et al. (2024)). In addition, one may use $\hat{d}_\pi$ to model a "second-stage" cost associated with the λ variables; this situation occurs, for instance, in column generation formulations for two-stage stochastic programming problems (Christiansen and Lysgaard, 2007; Gauvin et al., 2014; Silva and Wood, 2006).

The case of DW formulations is relevant because it is well known that, for many combinatorial optimization problems, DW bounds can be significantly stronger than the bounds arising from LP relaxations of natural formulations in lower-dimensional spaces (Pessoa et al., 2020; de Lima et al., 2022). However, algorithms based on DW formulations typically use a branch-and-price scheme, which is not natively supported by many commercial MIP solvers. Furthermore, as highlighted in Uchoa et al. (2024), most MIP solvers have a native support for the addition of cuts, but not for the addition of the variables needed in a branch-and-price method. In this context, Chen et al. (2024) recently proposed an approach that recovers the DW bound with cuts in the original space of the variables. We argue next that our approach extends the results of Chen et al. (2024) by providing multiple corner Benders' cuts that recover the DW bound. Our computational experiments show that this is indeed advantageous, allowing us to obtain state-of-the-art results for a well-studied combinatorial optimization problem.

As discussed earlier, let $\mathbb{A}$ be a DP algorithm that solves the pricing subproblem with respect to Problem (11) and $\mathcal{N}$ be a corresponding network. Setting Y as in Problem (12), and recalling that $x = w$ in our setting, yields $\hat{z} = \min_{x,\theta} \{ c^\intercal x + \theta : x \in X, (x, \theta) \in \text{EPI}(f_Y) \}$. Solving Problem (8), we obtain a Lagrangian cut $\alpha^\intercal x + \theta \geq \sigma_{\text{EPI}(f_Y)}(\alpha, 1)$, and by Theorem 3 with $(\rho, \rho_0) = (c, 1)$, this inequality implies the objective function cut $c^\intercal x + \theta \geq \sigma_{\mathcal{F}}(c, 1) = \hat{z}$. This is essentially the same reasoning as the one devised by Chen et al. (2024) to recover the DW bound via cuts (see also Appendix C for the case of multiple "blocks"). Proposition 1 then shows that instead of recovering the DW bound with a single Lagrangian cut, we can recover this bound with multiple corner Benders' cuts. As we argued earlier in Section 3.4, this can be advantageous because it might reduce the dimension of an optimal face.

4.3 Application: the vehicle routing problem with stochastic demands (VRPSD)

In this section, we introduce the combinatorial optimization problem that will be used in our computational experiments, namely the Vehicle Routing Problem with Stochastic Demands (VRPSD). The reason we introduce the problem at this point is so that we can give concrete examples of some of the concepts that have been discussed, which will hopefully help the reader to better understand them.

The VRPSD is a variant of the classic capacitated vehicle routing problem where, rather than deterministic, customer demands are stochastic and follow a given probability distribution. This problem has a very rich literature, with many studies in the last few years (Gendreau et al., 2016; Florio et al., 2020, 2022; Hoogendoorn and Spliet, 2023; Parada et al., 2024; Ota and Fukasawa, 2024). As usual, we model the VRPSD as a *two-stage stochastic program*, where a recourse policy describes the recourse actions that should be taken in the case of *route failures*, i.e., when a violation of the vehicle capacity constraint is observed. The objective function then combines the initial planned routes cost (*first-stage cost*) and the expected recourse cost (*second-stage cost*). Next, we present a formal definition.

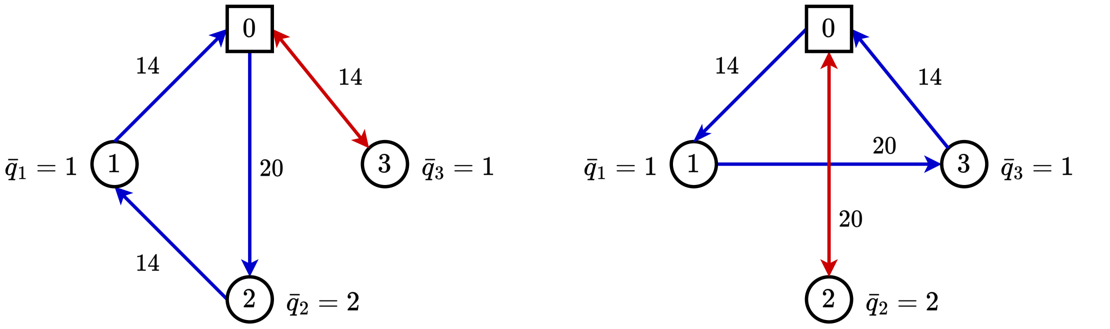
4.3.1 Definition of the VRPSD

Let $G = (V = \{0\} \cup V_+, E)$ be a complete undirected graph. The vertex 0 represents the *depot* and the set V_+ indicates the *customers.* The vector $c \in \mathbb{Q}^E_{++}$ denotes the edge costs, $k \in \mathbb{Z}_{++}$ refers to the desired number of vehicles, and $C \in \mathbb{Q}_{++}$ is the vehicle capacity. Let q be a random vector over V_+, where each entry q_v is a random variable corresponding to the demand of customer $v \in V_+$. The vector q is governed by a probability distribution $\mathbb{P}$, and we define $\bar{q} := \mathbb{E}[q]$. A *route* R is a tuple of customers $(v_1, \ldots, v_\ell)$ and R is *elementary* if the customers in R are all distinct. We use $q(R)$ to refer to the sum of random variables $\sum_{i=1}^{\ell} q_{v_i}$, and we say that route R is a *q-route* if the expected total demand of the route is below capacity, i.e., if $\bar{q}(R) := \mathbb{E}[q(R)] \leq C$. The notation $c(R)$ denotes the *first-stage cost of route R*, i.e., $c(R) := c_{0v_1} + c_{0v_\ell} + \sum_{j=1}^{\ell-1} c_{v_j v_{j+1}}$ ($c(R) = 2c_{0v_1}$ if $\ell = 1$). Moreover, with each route R we associate a random variable $\mathcal{Q}(R)$ and the expectation $\mathbb{E}[\mathcal{Q}(R)]$ denotes the (expected) *second-stage cost of R.*

The definition of the random variable $\mathcal{Q}(R)$ depends on the choice of the recourse policy. In this work, we focus on the *classical recourse policy*, that is, whenever a failure is observed, the recourse policy determines that the vehicle should execute a back-and-forth trip between the depot and the failed customer. In this context, Dror et al. (1989) show that, for any q-route $R = (v_1, \ldots, v_\ell)$, the second-stage cost can be calculated with the following formula, which computes the probability that t returns to the depot are needed to satisfy customer j's demand. We use the shorthand $[i]$ for an integer $i \geq 0$ to denote the set $\{1, \ldots, i\}$ and define $[0] := \emptyset$.

$$\mathbb{E}[\mathcal{Q}(R)] = 2 \sum_{j \in [\ell]} \sum_{t=1}^{\infty} \mathbb{P}\left[\sum_{i \in [j-1]} q_{v_i} \leq tC < \sum_{i \in [j]} q_{v_i}\right] c_{0v_j}. \tag{13}$$

The goal of the VRPSD is to find a set of k distinct elementary q-routes $\{R_1, \ldots, R_k\}$ that forms a partition of the customers V_+ and minimizes the sum $\sum_{i\in[k]}(c(R_i) + \mathbb{E}[\mathcal{Q}(R_i)])$.



(a) A set of q-routes $\{R_1, R_2\}$ that minimizes the first-stage cost. Here we have that $\sum_{i\in[2]} c(R_i) = 76$.

(b) A solution $\{R'_1, R'_2\}$ that is optimal for the VRPSD. The sum of the first and second-stage costs is $\sum_{i\in[2]}(c(R'_i) + \mathbb{E}[\mathcal{Q}(R'_i)]) = 88$.

Figure 2: Example of a VRPSD instance with $V_+ = \{1, 2, 3\}$, $k = 2$, and $C = 3$. The edge costs are $c_{01} = c_{03} = c_{12} = c_{23} = 14$ and $c_{13} = c_{02} = 20$. The expected demand values $\bar{q}_v$ are given beside each customer $v \in V_+$. In this example, each q_i follows a normal probability distribution with variance 0.1, so if a q-route R has $\bar{q}(R) \le 2$, then $\mathbb{E}[\mathcal{Q}(R)] \approx 0$. The solution in (a) has total cost $\sum_{i\in[2]}(c(R_i) + \mathbb{E}[\mathcal{Q}(R_i)]) = 76 + \mathbb{E}[\mathcal{Q}((2, 1))] = 76 + (1/2) \cdot 28 = 90$, where the 1/2 appears because the normal distribution is symmetric and results in 50% chance of exceeding capacity at customer 1.

Throughout this paper, we only consider instances of the VRPSD that satisfy Assumption ($\star$), which says that, at any point along a route, the second-stage cost is just a function of the last vertex visited and the accumulated expected demand.

Assumption ($\star$). There exists a function $\Psi : \mathbb{R} \times V_+ \to \mathbb{R}$ such that, for any elementary q-route $R = (v_1, \ldots, v_\ell)$,

$$\Psi(v_\ell, \bar{q}(R)) = 2c_{0v_\ell} \sum_{t=1}^{\infty} \mathbb{P}\left[\sum_{i\in[\ell-1]} q_{v_i} \le tC < \sum_{i\in[\ell]} q_{v_i} \right]. \tag{$\star$}$$

In particular, this implies that for any elementary q-route $R = (v_1, \ldots, v_\ell)$,

$$\mathbb{E}[\mathcal{Q}(R)] = \sum_{j\in[\ell]} \Psi(v_j, \bar{q}((v_1, \ldots, v_j))).$$

Such an assumption is common in the VRPSD literature (Christiansen and Lysgaard, 2007; Gauvin et al., 2014; Florio et al., 2020), as it facilitates a column generation formulation of the VRPSD where the pricing subproblem is solved via a “knapsack-like” DP algorithm that defines one state for every possible choice of $(v_\ell, \bar{q}(R))$ (observe that multiple routes might collapse to the same state in this construction). Moreover, branch-and-cut algorithms (Jabali et al., 2014; Hoogendoorn and Spliet, 2023; Parada et al., 2024) consider instances that either satisfy Assumption ($\star$), or a similar assumption where function Ψ also depends on other parameters such as the total variance of route R.

4.3.2 Formulation via network flows

We now present a mixed-integer programming formulation for the VRPSD whose LP relaxation has the form of Problem (1). We begin by building on the branch-and-price algorithm of Christiansen and Lysgaard (2007) and the connection mentioned in Section 4.1.

We construct a set partitioning formulation of the VRPSD via Problem (11). Let $\mathcal{R}$ be the set of all q-routes with respect to the input. Additionally, for each route $R \in \mathcal{R}$ and edge $e \in E$, let $\textsc{count}(R, e)$ denote how many times edge e appears in route R. For each route $R \in \mathcal{R}$, let $\pi(R) \in \mathbb{Z}^E$ be a vector where each entry has value $\textsc{count}(R, e)$, and let $\Pi = \{\pi(R) : R \in \mathcal{R}\}$. We define $\hat{d}_{\pi(R)} = \mathbb{E}[\mathcal{Q}(R)]$ for each $R \in \mathcal{R}$, k as the number of vehicles in the VRPSD instance, and

$$X = \left\{ x \in [0,2]^E : \begin{array}{ll} x(\delta(0)) = 2k, & \\ x(\delta(v)) = 2, & \forall v \in V_+ \\ x(\delta(S)) \geq 2 \left\lceil \dfrac{\bar{q}(S)}{C} \right\rceil, & \forall S \subseteq V_+, S \neq \emptyset \\ x_e \leq 1, & \forall e \in E \setminus \delta(0) \end{array} \right\}.$$

Here, $\delta(v)$ denotes the set of edges incident to v, and we allow edges to be traversed twice (for a route that visits a single customer).

Note that the linking constraints (which take the form $x_e = \sum_{R \in \mathcal{R}} \textsc{count}(R, e) \cdot \lambda_R$, for all $e \in E$) and the definition of X imply that $\lambda_R \leq 1$, for all $R \in \mathcal{R}$. We refrain from further explaining this set partitioning formulation, as it follows the format of relatively standard formulations used in various branch-cut-and-price algorithms for different vehicle routing variants, including the VRPSD (Fukasawa et al., 2006; Pessoa et al., 2020; Christiansen and Lysgaard, 2007; Gauvin et al., 2014; Florio et al., 2020).

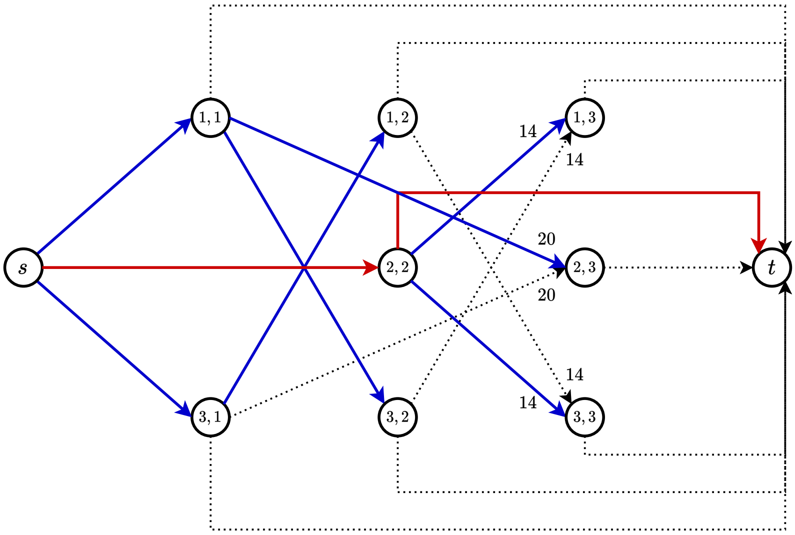


Figure 3: Network $\mathcal{N} = (\mathcal{V}, \mathcal{A})$ corresponding to the example shown in Figure 2. The numbers next to the arcs indicate the arc cost d_a. We omit these numbers when the arc cost is zero. From left to right, each layer ℓ (indexed by the second number of each node label, corresponding to accumulated demand) contains a node for every customer with $\bar{q}_v \leq \ell$.

Given Assumption ($\star$), iteratively finding routes (pricing in a column generation framework) for the problem can be modeled as a shortest $s-t$ path problem in a network $\mathcal{N} = (\mathcal{V}, \mathcal{A})$ as follows (see Figure 3). The vertex set $\mathcal{V}$ is associated to "states" and comprises a source vertex s (state $[0,0]$), a sink vertex t, and there is a vertex/state $\mathcal{S} = [v, \mu]$ for each $v \in V_+$ and $\mu \in \{\bar{q}_v, \ldots, C\}$. Intuitively, state $[v, \mu]$ corresponds to a DP state of the pricing algorithm where we are currently at customer v with an accumulated average demand of μ. With this interpretation in mind, we define $\textsc{vertex} : \mathcal{V} \to V$ as a mapping such that $\textsc{vertex}(\mathcal{S}) = v$, if $\mathcal{S} = [v, \mu]$, and $\textsc{vertex}(\mathcal{S}) = 0$, if $\mathcal{S} \in \{s, t\}$.

Each arc $\mathcal{S}_1\mathcal{S}_2 \in \mathcal{A}$ is mapped to an edge $\textsc{edge}(\mathcal{S}_1\mathcal{S}_2) := \{\textsc{vertex}(\mathcal{S}_1), \textsc{vertex}(\mathcal{S}_2)\} \in E$. The intuition is that taking arc $\mathcal{S}_1\mathcal{S}_2$ in an $s-t$ path in $\mathcal{N}$ corresponds to adding $\textsc{edge}(\mathcal{S}_1\mathcal{S}_2)$ to a route in the VRPSD instance graph G. The arc set $\mathcal{A}$ of the network is given by the union of the following sets

$$\begin{aligned}
&\texttt{SRC} = \{(s, [v, \bar{q}_v]) : v \in V_+\},\\
&\texttt{FWD} = \{([u, \mu], [v, \mu + \bar{q}_v]) : u, v \in V_+, \mu \in \{\bar{q}_u, \ldots, C - \bar{q}_v\}\}, \text{ and}\\
&\texttt{SNK} = \{([v, \mu], t) : v \in V_+, \mu \in \{\bar{q}_v, \ldots, C\}\}.
\end{aligned}$$

In practice, any state $\mathcal{S}$ that is not reachable from the source s will be removed in a preprocessing step.

We set the arc costs $d \in \mathbb{R}^{\mathcal{A}}$ as

$$d_a = \begin{cases} \Psi(v, \mu + \bar{q}_v), & \text{if } a = ([u, \mu], [v, \mu + \bar{q}_v]) \in \texttt{SRC} \cup \texttt{FWD} \\ 0, & \text{otherwise.} \end{cases}$$

Let $\mathbb{I}(\,\cdot\,)$ denote the indicator function. Applying the derivation of Problem (12) in Section 4.1, we obtain the following flow-based formulation that is equivalent to the earlier set partitioning formulation based on Problem (11), where

$$Y = \left\{y \in \mathbb{R}_+^{\mathcal{A}} : y(\delta^-(\mathcal{S})) - y(\delta^+(\mathcal{S})) = k \cdot \mathbb{I}(\mathcal{S} = t) - k \cdot \mathbb{I}(\mathcal{S} = s),\ \forall \mathcal{S} \in \mathcal{V}\right\}$$

$$z_{SP} = \min_{x,y} \left\{ \sum_{e \in E} c_e x_e + \sum_{a \in \mathcal{A}} d_a y_a : \sum_{a \in \mathcal{A} : \textsc{edge}(a) = e} y_a = x_e \text{ for all } e \in E,\, x \in X,\, y \in Y \right\}. \tag{14}$$

Let $Q \in \mathbb{R}^{E \times \mathcal{A}}$ be such that each column Q_a, with $a \in \mathcal{A}$, is the incidence vector of $\textsc{edge}(a) \in E$. Then

$$z_{SP} = \min_{x,y}\{c^\intercal x + d^\intercal y : x = Qy, x \in X, y \in Y\} = \min_{x,\theta}\{c^\intercal x + \theta : x \in X, (x, \theta) \in \textsc{epi}(f_Y)\},$$

and we can solve the VRPSD to integrality by enforcing that $x \in \mathbb{Z}^E$ above.

4.4 Corners of network flow polytopes and numerical example of corner Benders' cuts

In this subsection, we shall see that Figure 3 can be interpreted as an illustration of a corner C of the network flow polytope Y defined in Section 4.3.2. To do this, we first recall some basic facts from network flow theory (see Section 11.11 of Ahuja et al. (1988)) to establish a connection between the geometric notion of corners and the combinatorial concepts of spanning trees and cycles.

4.4.1 Network simplex background

One can show that exactly one of the equations in the system $\mathbf{N}y = b$ is redundant. Thus, from now on, we use $\mathbf{N}'y = b'$ to refer to the system of equations obtained by removing the equation associated with vertex $s \in \mathcal{V}$. For any subset of arcs $\mathcal{B}$ of $\mathcal{A}$ we write $\mathbf{N}_{\mathcal{B}}$ (respectively $\mathbf{N}'_{\mathcal{B}}$) to denote the submatrix of $\mathbf{N}$ (respectively $\mathbf{N}'$) obtained by dropping the columns associated with arcs in $\mathcal{A}\backslash\mathcal{B}$. The following results are well known in network flow and network simplex theory.

Theorem 4. *Let $\mathcal{T} \subseteq \mathcal{A}$ be such that $|\mathcal{T}| = |\mathcal{V}| - 1$. The columns of $\mathbf{N}'_{\mathcal{T}}$ are linearly independent if and only if the arcs in $\mathcal{T}$ forms a spanning tree of $\mathcal{N}$.*

Definition 10. Let $C^a \subseteq \mathcal{N}$ be a cycle containing arc $a \in \mathcal{A}$. The notation $\mathbb{1}(C^a) \in \mathbb{R}^{\mathcal{A}}$ denotes the incidence vector of C^a, i.e., for each $a' \in \mathcal{A}$,

$$(\mathbb{1}(C^a))_{a'} := \begin{cases} 1, & \text{if } a' \text{ is a forward arc in } C^a, \\ -1, & \text{if } a' \text{ is a backward arc in } C^a, \\ 0, & \text{otherwise.} \end{cases}$$

Fact 1. *Let $\mathcal{T}$ be a basis (spanning tree) of Y. For each $a \in \mathcal{A}\setminus\mathcal{T}$, define a vector $r^{a,\mathcal{T}} \in \mathbb{R}^{\mathcal{A}}$ as follows*

$$r^{a,\mathcal{T}} := \begin{pmatrix} (\mathbf{N}'_{\mathcal{T}})^{-1}\mathbf{N}'_{\{a\}} \\ 1 \\ 0 \end{pmatrix} \begin{array}{l} \} \ \mathcal{T} \\ \} \ \{a\} \\ \} \ (\mathcal{A} \setminus \{a\}) \setminus \mathcal{T}, \end{array}$$

where the sets in the right indicate the corresponding entries of the vector. Then $r^{a,\mathcal{T}} = \mathbb{1}(C^a)$, where C^a is the unique cycle in the graph induced by $\mathcal{T} \cup \{a\}$.

Henceforth, whenever $\mathcal{T}$ is clear from the context, we omit $\mathcal{T}$ from the superscript of r.

4.4.2 Numerical example

We now present a numerical example that illustrates corner Benders' cuts. Consider the instance and network illustrated in Figures 2 and 3. Suppose that our current solution $(\bar{x}, \bar{\theta})$ has $\bar{x}_{01} = \bar{x}_{02} = \bar{x}_{12} = 1$, $\bar{x}_{03} = 2$ and $\bar{\theta} = 0$ (same as Figure 2a, but with $\bar{\theta}$ less than $f_Y(\bar{x}) = \mathbb{E}[\mathcal{Q}((2,1))] = (1/2) \cdot 28 = 14$). We first solve Problem (8) to obtain the Lagrangian cut

$$(\alpha^1)^\intercal x + \theta = x_{01} + x_{03} - 15x_{12} - 2x_{13} - 15x_{23} + \theta \geq 0. \tag{15}$$

Note that $(\bar{x}, \bar{\theta})$ violates this inequality: $(\alpha^1)^\intercal\bar{x} + \bar{\theta} = -12 < 0$. Then, via Theorem 3 with $(\rho, \rho_0) = (c, 1)$, (15) implies the objective function cut $c^\intercal x + \theta \geq \sigma_{\mathcal{F}}(c, 1) = z_{SP}$, and in this case, it turns out that $z_{SP} = 88$. We then solve $\min_y\{(\alpha^1)^\intercal Qy + d^\intercal y : y \in Y\}$ to obtain an optimal solution y^* and a corresponding optimal basis (spanning tree) $\mathcal{T}$. These are illustrated in Figure 3: the solid arcs indicate the arcs in the tree and y^* is given by the red arcs (i.e., path $(s, [2,2], t)$), so $\sigma_Y(Q^\intercal\alpha^1 + d) = 0$.

Let $C = \{y^*\} + \text{CONE}(R)$ be a corner of Y derived from the basis $\mathcal{T}$ according to Remark 1. By Fact 1, we have $R = \{r^a\}_{a\in\mathcal{A}\setminus\mathcal{T}}$, that is, the rays in Remark 1 correspond to the cycles that we obtain by adding arcs in $\mathcal{A}\setminus\mathcal{T}$ to the spanning tree $\mathcal{T}$. Moreover, using Lemma 3, we have that $\text{EPI}(f_C) = \{(Qy^*, d^\intercal y^*)\} + \text{CONE}(\{(Qr^a, d^\intercal r^a\}_{a\in\mathcal{A}\setminus\mathcal{T}} \cup \{(\mathbf{0}, 1)\})$.

In this way, we apply Theorem 2 to separate the corner Benders' cut

$$(\alpha^2)^\intercal x + \theta = -14x_{12} - 14x_{23} + \theta \geq 0. \tag{16}$$

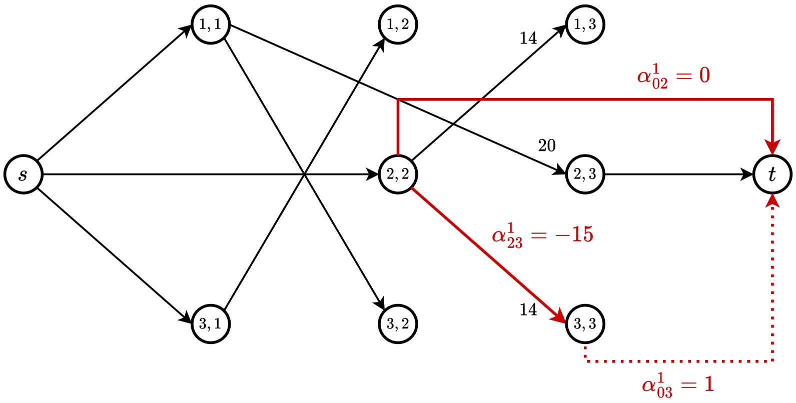


Figure 4: Illustration of the ray (cycle) $r^{([3,3],t)}$. Notice that $(\alpha^1)^\intercal(Qr^{([3,3],t)})+d^\intercal r^{([3,3],t)}) = 0$.

Observe that (16) dominates (15) since summing some of the degree constraints defining X to (15) yields

$$\begin{aligned}
&(\alpha^1)^\intercal x + \theta - 2x(\delta(0)) + x(\delta(1)) + x(\delta(3)) \geq -2(2k) + 2(2)\\
\iff &(\alpha^1)^\intercal x + \theta + (-2x_{01} - 2x_{02} - 2x_{03}) + (x_{01} + x_{12} + x_{13}) + (x_{03} + x_{13} + x_{23}) \geq -4\\
\iff &-2x_{02} - 14x_{12} - 14x_{23} + \theta \geq -4,
\end{aligned}$$

which is equivalent to summing (16) with inequality $-2x_{02} \geq -4$. In addition, we can manually verify that, as predicted by Theorem 2, inequality (16) defines a facet of $\textsc{epi}(f_C)$. To see validity, we can just check that $(\alpha^2)^\intercal(Qy^*)+d^\intercal y^* = 0$ and $(\alpha^2)^\intercal(Qr^a)+d^\intercal r^a \geq 0$, for all $a \in \mathcal{A}\setminus\mathcal{T}$. To check the dimension of the face induced by (16) we observe that $\alpha = \alpha^2$ is the unique solution to the system of equations below.

$$\begin{aligned}
&\alpha^\intercal(Qy^*) + d^\intercal y^* = 2\alpha_{02} + 2\alpha_{02} = 0\\
&\alpha^\intercal Qr^{([1,1],t)} + d^\intercal r^{([1,1],t)} = 2\alpha_{01} - 2\alpha_{02} = 0,\\
&\alpha^\intercal Qr^{([1,2],t)} + d^\intercal r^{([1,2],t)} = \alpha_{01} - 2\alpha_{02} + \alpha_{03} + \alpha_{13} = 0,\\
&\alpha^\intercal Qr^{([1,3],t)} + d^\intercal r^{([1,3],t)} = \alpha_{01} - \alpha_{02} + \alpha_{12} + 14 = 0,\\
&\alpha^\intercal Qr^{([3,1],t)} + d^\intercal r^{([3,1],t)} = 2\alpha_{03} - 2\alpha_{02} = 0,\\
&\alpha^\intercal Qr^{([3,3],t)} + d^\intercal r^{([3,3],t)} = \alpha_{03} - \alpha_{02} + \alpha_{23} + 14 = 0.
\end{aligned}$$

5 Computational experiments

To evaluate the performance of the proposed corner Benders' cuts, we conducted computational experiments on VRPSD instances. Therefore, in the remainder of this section, we use the notation and concepts introduced in Section 4.3. In particular, we use the sets $X \subseteq \mathbb{R}^E$ and $Y \subseteq \mathbb{R}^\mathcal{A}$ as defined in Section 4.3.2.

As a baseline, we implemented our own version of the state-of-the-art integer L-shaped algorithm by Parada et al. (2024) (denoted PARADA). We then evaluated the algorithm's performance when the root node relaxation was strengthened by the following methods for separating Benders' cuts:

- BENDERS: Benders decomposition with the normalization proposed by Fischetti et al. (2010);

- LAGRANGE: single Lagrangian cut derived from Theorem 3 (inspired by Chen et al. (2024)); and
- CORNER: corner Benders' cuts for EPI(f_C), where C is obtained according to Proposition 1.

We do not show the results with the objective function cut, since it is always worse than LAGRANGE.

5.1 Benchmark instances

We tested the efficiency of our method on the instances proposed by Parada et al. (2024) with $|V| \in \{20, 30, 40, 50\}$ vertices, giving a total of 720 benchmark instances. Three instances with 20 customers and $k = 7$ are infeasible (`20_7_0.95_0`, `20_7_0.95_3` and `20_7_0.95_8`). Accordingly, we adjusted the number of vehicles to 8 for these instances.[2] In every instance, for each customer $v \in V_+$, customer demand q_v follows a normal distribution with variance and mean equal to $\bar{q}_v$, meaning that these instances satisfy Assumption ($\star$). To more accurately assess the performance of our method, we divided the instances into "few vehicles" or "small k" instances ($k \in \{2, 3, 4\}$) and "many vehicles" or "large k" instances ($k \in \{5, 6, 7\}$). This categorization was motivated by the observation that the algorithm PARADA already performs very well for small k instances, but it struggles for instances with large k. For each instance, we generate the network $\mathcal{N}$ according to the construction described in Section 4.3.2. In Table 1 we report the average number of nodes and arcs in the resulting networks.

	$k=2$		$k=3$		$k=4$		$k=5$		$k=6$		$k=7$	
$\lvert V\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{A}\rvert$
20	1023	17494	647	10367	445	6606	350	4739	280	3364	228	2426
30	2477	67252	1563	40851	1131	28373	923	22074	715	16232	589	12595
40	4567	169719	2879	104160	2153	75738	1740	59308	1378	45448	1141	36239
50	6988	329636	4609	212992	3488	157730	2749	121479	2240	96592	1850	77707

Table 1: Average size of the network $\mathcal{N}$ for different values of k.

5.2 Implementation details

All algorithms were executed in single-thread mode on a machine equipped with an Intel(R) Xeon(R) Gold 6142 CPU @ 2.60GHz processor. The code was implemented in C++, with Gurobi 12 as the LP/MIP solver and the Lemon library (Dezső et al., 2011) for handling basic graph operations. The time limit for every run of the algorithms was set at 1 hour. Our implementation of the state-of-the-art algorithm of Parada et al. (2024) is an exact branch-and-cut algorithm for the VRPSD that separates the rounded capacity inequalities (RCIs) of Laporte and Nobert (1983) and the integer L-shaped (ILS) cuts of Parada et al. (2024).[3] The formulation proposed by Parada et al. (2024) uses "disaggregated" second-stage variables, that is, for each customer $v \in V_+$, the formulation has

[2] The complete set of instances and their computational results are available on the website of one of the authors: `https://sites.google.com/view/jfcote/`. The reported solutions for the three mentioned instances indeed use 8 vehicles.

[3] Appendix H provides further details on the separation routines of Parada et al. (2024), identifies a minor mistake in their algorithm description, and compares in more detail the results of our implementation with those in the authors' online table.

a variable θ'_v that represents the expected second-stage cost paid at customer v. Parada et al. (2024) propose different ILS cuts that guarantee that, at integer solutions, the sum $\sum_{v \in V_+} \theta'_v$ correctly matches the expected second-stage cost of the first-stage solution. We refer the reader to the paper by Parada et al. (2024) for more details. Note that any Benders' cut $\alpha^\intercal x + \alpha_0 \theta \geq \beta$ can be translated to the space of variables $(x, \theta') \in \mathbb{R}^E \times \mathbb{R}^{V_+}$ as the inequality $\alpha^\intercal x + \alpha_0 \left(\sum_{v \in V_+} \theta'_v\right) \geq \beta$.

In our experiments, we observed that our implementation of the algorithm proposed by Parada et al. (2024) is competitive with the results reported in their online table: we can solve 5 more instances in the time limit and our implementation achieves root gaps that are, on average, approximately 1.5 times smaller. To achieve such stronger LP relaxation bounds, we start our solver by relaxing the integrality constraints and using a custom cutting-plane loop for separating RCIs and ILS cuts. This loop terminates under one of two conditions: (i) no additional cuts can be separated, or (ii) the bound fails to improve by at least 10^{-3} over the last 10 iterations. Once the loop stops, we enforce integrality on the $\{x_e\}_{e \in E}$ variables and solve the resulting MIP model with Gurobi, using callbacks to separate RCIs and ILS cuts during the solution process. Appendix D contains a pseudocode of our cutting-plane loop.

To evaluate the performance of BENDERS, LAGRANGE, and CORNER, we adapted our custom cutting-plane procedure to include the separation of the corresponding cuts. Appendix E provides further details on the separation of each type of Benders' cuts.

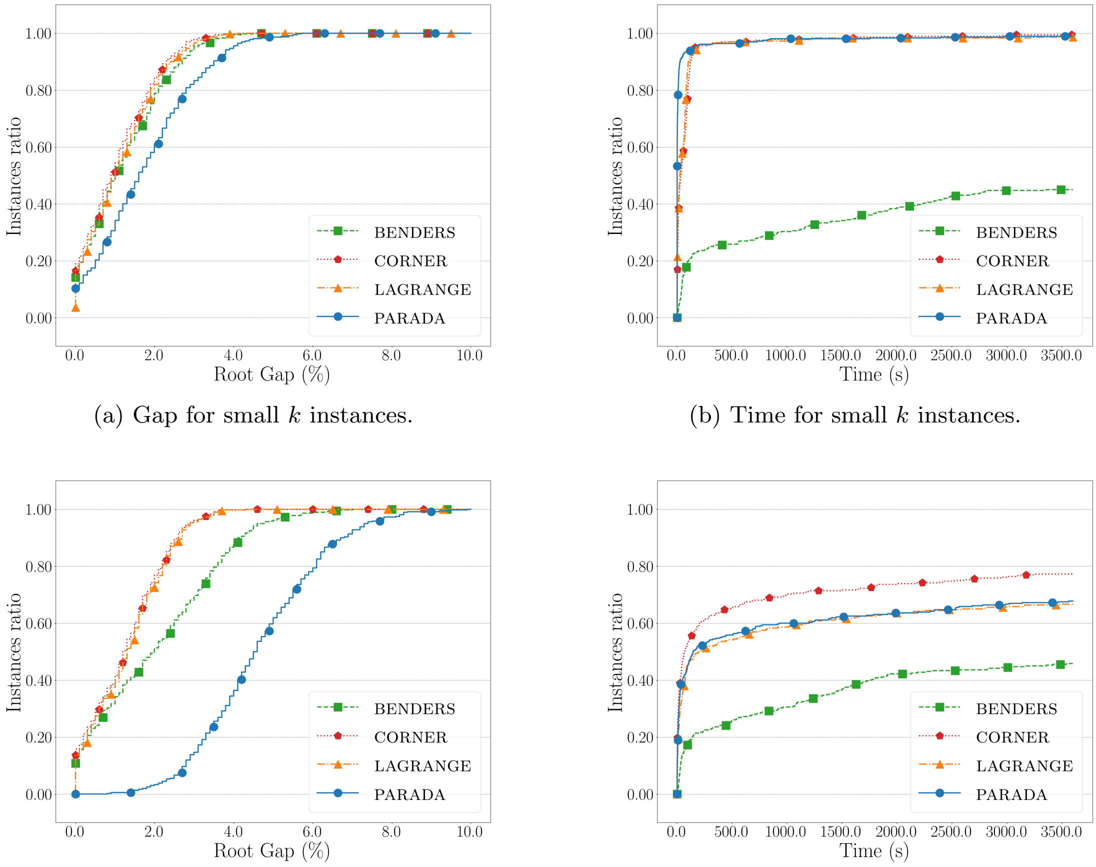
5.3 Computational results

The results of our experiments are summarized in Figure 5. Loosely speaking, these figures show the cumulative distribution of the instances with respect to one of two metrics: *gap* or *time*.[4] Take Figure 5a as an example and let $p = (p_1, p_2) \in \mathbb{R}^2$ be a point in the line of Figure 5a that corresponds to algorithm CORNER. Let $\mathcal{I}$ be the set of all small k instances and let $\hat{\mathcal{I}}$ be the number of these instances whose LP relaxation bound computed by CORNER has a gap of at most p_1. The number p_2 corresponds to the ratio $|\hat{\mathcal{I}}|/|\mathcal{I}|$. Similarly, for Figure 5b and a point $p' = (p'_1, p'_2) \in \mathbb{R}^2$ in the line of algorithm CORNER, we have that $p'_2 = |\hat{\mathcal{I}}'|/|\mathcal{I}|$, where $\hat{\mathcal{I}}' \subseteq \mathcal{I}$ is the set of instances that CORNER solved the problem in a runtime of at most p'_1.

On one hand, we see in Figure 5b that, for small k instances, all the approaches (except BENDERS) are fairly efficient and can solve almost all of the instances within the time limit. Moreover, Figure 5a shows that both CORNER and LAGRANGE achieved slightly stronger root gaps than BENDERS, and the execution time of both is competitive with PARADA. On the other hand, in Figures 5c and 5d, we see that, for large k instances, none of the approaches can solve all of the instances; and only CORNER can solve more instances than PARADA. Looking at Figure 5c, we see again that both CORNER and LAGRANGE achieved fairly strong root gaps, and Figure 5d shows that algorithm CORNER has a significantly better overall execution time. Table 2 shows the total number of instances solved by each algorithm within the time limit.

Number of branch-and-bound nodes. The fact that CORNER is faster than LAGRANGE might be because algorithm CORNER gives information of several facet-defining inequalities for EPI(f_C), while the only information that LAGRANGE has on the value function f_Y is a single Lagrangian cut. To further investigate this matter, we show in

[4] The *(remaining) gap* is computed as $100 \times (z_{\text{opt}} - z_{\text{cuts}})/z_{\text{opt}}$, where z_{opt} is the optimal value of the problem (known for our instances) and z_{cuts} is the bound after adding cuts to the LP relaxation.



(a) Gap for small k instances.

(b) Time for small k instances.

(c) Gap for large k instances.

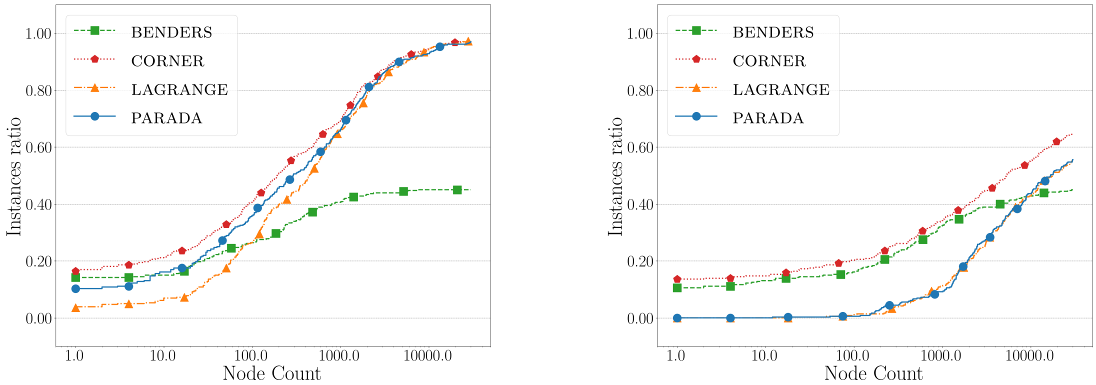
(d) Time for large k instances.

Figure 5: Computational results for instances of Parada et al.

PARADA	BENDERS	LAGRANGE	CORNER
600	327	595	636

Table 2: Number of instances (out of 720) solved by each algorithm in the time limit of 1 hour.

Figure 6 the number of branch-and-bound nodes explored by each method. In these figures, a point $p = (p_1, p_2) \in \mathbb{R}^2$ in the line of an algorithm is so that $p_2 = |\hat{\mathcal{I}}|/|\mathcal{I}|$, where $\mathcal{I}$ is the set of all considered instances and $\hat{\mathcal{I}} \subseteq \mathcal{I}$ is the set of instances to which the algorithm solved the problem to optimality by exploring at most p_1 nodes of the branch-and-bound tree.



(a) Nodes explored for small k instances.

(b) Nodes explored for large k instances.

Figure 6: Results for the number of branch-and-bound nodes explored by the different algorithms.

Figure 6 clearly shows that CORNER explores fewer branch-and-bound nodes than LAGRANGE. Furthermore, CORNER solves approximately 16% (respectively, 14%) of all small k instances (respectively, large k instances) in the root node. This result suggests that corner Benders' cuts might be an efficient way of replacing the objective function cut, while the Lagrangian cut approach may still lead to high-dimensional optimal faces, which can be detrimental to the performance of the MIP solver.

Root bound convergence. Next, we analyze the convergence speed of CORNER to the optimal bound z^* in Problem 1, comparing it with the other two Benders' cuts-based methods. Figure 7 illustrates the improvement in the LP relaxation bound over time (we use a representative instance to showcase our findings, but similar behavior is observed across other instances). The horizontal line for LAGRANGE indicates that the optimal bound for this instance is $z^* = 1038.07$, which, according to Theorem 3, can be achieved with a single Lagrangian cut. Observe that in 7 seconds, algorithm CORNER reaches the optimal bound, while BENDERS fails to even reach a bound of 1015. For this same instance, Appendix I reports the time to separate each Benders' cut and the number of ray inequalities generated by Algorithm SolvePolar.

Table 3 presents the average number of Benders' cuts separated (recall that corner Benders' cuts are Benders' cuts) and the average root node solving time (which is capped

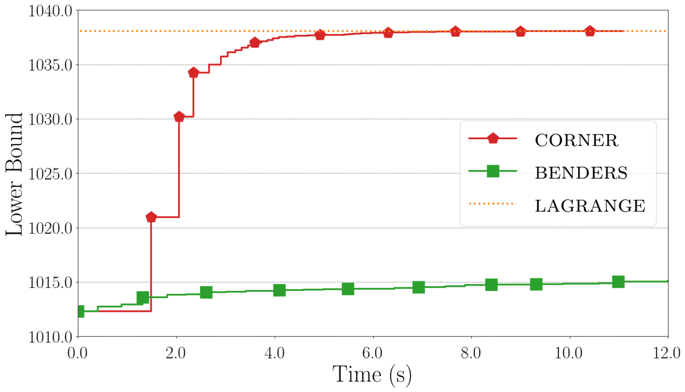


Figure 7: Bound improvement for instance "30_7_0.95_1.00_8" of Parada et al. (2024) ($|V| = 30$ and $k = 7$).

at 1 hour if the algorithm does not solve the root node within the time limit). Observe that BENDERS separates significantly more cuts than CORNER and takes much more time to converge, strengthening our point that BENDERS has worse convergence to the optimal bound. Furthermore, as previously shown, while LAGRANGE reaches the optimal bound with a single cut, it is generally computationally less efficient than CORNER for solving the problem to integrality.

	Benders' cuts		Root Time (s)	
	BENDERS	CORNER	BENDERS	CORNER
small k	143.88	20.31	2315.67	44.31
large k	636.98	75.43	2217.13	42.65

Table 3: Average number of Benders' cuts separated and root node solving time for algorithms BENDERS and CORNER.

6 Conclusion

In this work, we have developed a new way to generate Benders' cuts by exploiting basis information on the LP solved by the Benders' cut generating subproblem. When applied to instances of the VRPSD, our approach improved the performance of a state-of-the-art problem-specific algorithm found in the literature.

We believe the idea of exploring basis information to generate Benders' cuts shows much promise and future work includes testing this approach in other important problems, as well as considering how the consideration of integrality on the Benders' subproblem variables may affect the cut-generating process using such basis information.

References

Ravindra K Ahuja, Thomas L Magnanti, and James B Orlin. *Network flows*. Cambridge, Mass.: Alfred P. Sloan School of Management, Massachusetts . . . , 1988.

Egon Balas. Disjunctive programming. *Ann. Discrete Math.*, 5:3–51, 1979. URL `https://doi.org/10.1016/S0167-5060(08)70342-X`.

Egon Balas and Aleksandr M. Kazachkov. $\mathcal{V}$-polyhedral disjunctive cuts. *Mathematical Programming*, May 2025. ISSN 1436-4646.

Egon Balas, Sebastián Ceria, and Gérard Cornuéjols. A lift-and-project cutting plane algorithm for mixed 0–1 programs. *Mathematical programming*, 58(1):295–324, 1993.

Bernd Becker, Markus Behle, Friedrich Eisenbrand, and Ralf Wimmer. BDDs in a branch and cut framework. In *Experimental and Efficient Algorithms*, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

Markus Behle. *Binary decision diagrams and integer programming.* Ph.d. thesis, Max Planck Institute for Computer Science, Germany, 2007.

J. F. Benders. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1):238–252, Dec 1962. ISSN 0945-3245.

David Bergman, Andre A. Cire, Willem-Jan van Hoeve, and John Hooker. *Decision Diagrams for Optimization.* Artificial Intelligence: Foundations, Theory, and Algorithms (AIFTA). Springer, Cham, 2016.

Suresh Bolusani, Mathieu Besançon, Ksenia Bestuzheva, Antonia Chmiela, João Dionísio, Tim Donkiewicz, Jasper van Doornmalen, Leon Eifler, Mohammed Ghannam, Ambros Gleixner, Christoph Graczyk, Katrin Halbig, Ivo Hedtke, Alexander Hoen, Christopher Hojny, Rolf van der Hulst, Dominik Kamp, Thorsten Koch, Kevin Kofler, Jurgen Lentz, Julian Manns, Gioni Mexi, Erik Mühmer, Marc E. Pfetsch, Franziska Schlösser, Felipe Serrano, Yuji Shinano, Mark Turner, Stefan Vigerske, Dieter Weninger, and Lixing Xu. The SCIP Optimization Suite 9.0. Technical report, Optimization Online, February 2024.

Pierre Bonami, Domenico Salvagnin, and Andrea Tramontani. Implementing automatic Benders decomposition in a modern MIP solver. In *Integer Programming and Combinatorial Optimization: 21st International Conference, IPCO 2020, London, UK, June 8–10, 2020, Proceedings*, pages 78–90. Springer, 2020.

E Andrew Boyd. Fenchel cutting planes for integer programs. *Operations Research*, 42 (1):53–64, 1994.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization.* Cambridge university press, 2004.

René Brandenberg and Paul Stursberg. Refined cut selection for Benders decomposition: applied to network capacity expansion problems. *Mathematical Methods of Operations Research*, 94(3):383–412, 2021.

Florent Cadoux and Claude Lemaréchal. Reflections on generating (disjunctive) cuts. *EURO Journal on Computational Optimization*, 1:51–69, 2013.

Margarita P Castro, Andre A Cire, and J Christopher Beck. A combinatorial cut-and-lift procedure with an application to 0–1 second-order conic programming. *Mathematical Programming*, 196(1):115–171, 2022.

Rui Chen and James Luedtke. On generating Lagrangian cuts for two-stage stochastic integer programs. *INFORMS Journal on Computing*, 34(4):2332–2349, 2022.

Rui Chen, Oktay Günlük, and Andrea Lodi. Recovering Dantzig–Wolfe bounds by cutting planes. *Operations Research*, 2024.

Christian H. Christiansen and Jens Lysgaard. A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands. *Operations Research Letters*, 35(6):773–781, 2007. ISSN 0167-6377.

Vašek Chvátal, William Cook, and Daniel Espinoza. Local cuts for mixed-integer programming. *Mathematical Programming Computation*, 5(2):171–200, 2013.

Michele Conforti and Laurence A Wolsey. "Facet" separation with one linear program. *Mathematical Programming*, 178(1):361–380, 2019.

Michele Conforti, Gérard Cornuéjols, and Giacomo Zambelli. Corner polyhedron and intersection cuts. *Surveys in Operations Research and Management Science*, 16(2): 105–120, 2011. URL `http://dx.doi.org/10.1016/j.sorms.2011.03.001`.

Michele. Conforti, Gerard Cornuejóls, and Giacomo. Zambelli. *Integer programming.* Graduate texts in mathematics, . 271. Springer, Cham, 2014. ISBN 9783319110073.

W.J. Cook, W.H. Cunningham, W.R. Pulleyblank, and A. Schrijver. *Combinatorial Optimization.* Wiley Series in Discrete Mathematics and Optimization. Wiley, 2011. ISBN 9781118031391. URL `https://books.google.ca/books?id=tarLTNwM3gEC`.

Gérard Cornuéjols and Claude Lemaréchal. A convex-analysis perspective on disjunctive cuts. *Mathematical Programming*, 106(3):567–586, 2006.

Alysson M Costa. A survey on Benders decomposition applied to fixed-charge network design problems. *Computers & operations research*, 32(6):1429–1450, 2005.

M Poggi de Aragao and Eduardo Uchoa. Integer program reformulation for robust branch-and-cut-and-price algorithms. In *Mathematical program in rio: a conference in honour of nelson maculan*, pages 56–61, 2003.

Vinícius L de Lima, Cláudio Alves, François Clautiaux, Manuel Iori, and José M Valério de Carvalho. Arc flow formulations based on dynamic programming: Theoretical foundations and applications. *European Journal of Operational Research*, 296(1):3–21, 2022.

Balázs Dezső, Alpár Jüttner, and Péter Kovács. Lemon–an open source C++ graph template library. *Electronic Notes in Theoretical Computer Science*, 264(5):23–45, 2011.

Moshe Dror, Gilbert Laporte, and Pierre Trudeau. Vehicle routing with stochastic demands: Properties and solution frameworks. *Transportation Science*, 23(3):166–176, 1989.

Matteo Fischetti, Domenico Salvagnin, and Arrigo Zanette. A note on the selection of Benders' cuts. *Mathematical Programming*, 124:175–182, 2010.

Alexandre M. Florio, Richard F. Hartl, and Stefan Minner. New exact algorithm for the vehicle routing problem with stochastic demands. *Transportation Science*, 54(4): 1073–1090, 2020.

Alexandre M Florio, Michel Gendreau, Richard F Hartl, Stefan Minner, and Thibaut Vidal. Recent advances in vehicle routing with stochastic demands: Bayesian learning for correlated demands and elementary branch-price-and-cut. *European Journal of Operational Research*, 2022.

Antonio Frangioni. About lagrangian methods in integer optimization. *Annals of Operations Research*, 139:163–193, 2005.

Ricardo Fukasawa, Humberto Longo, Jens Lysgaard, Marcus Poggi de Aragão, Marcelo Reis, Eduardo Uchoa, and Renato F. Werneck. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming*, 106(3):491–511, 2006. ISSN 1436-4646. doi: 10.1007/s10107-005-0644-x. URL `http://dx.doi.org/10.1007/s10107-005-0644-x`.

Dinakar Gade, Simge Küçükyavuz, and Suvrajeet Sen. Decomposition algorithms with parametric Gomory cuts for two-stage stochastic integer programs. *Mathematical Programming*, 144(1):39–64, 2014.

Gerald Gamrath and Marco E Lübbecke. Experiments with a generic Dantzig-Wolfe decomposition for integer programs. In *International Symposium on Experimental Algorithms*, pages 239–252. Springer, 2010.

Charles Gauvin, Guy Desaulniers, and Michel Gendreau. A branch-cut-and-price algorithm for the vehicle routing problem with stochastic demands. *Computers & Operations Research*, 50:141–153, 2014. ISSN 0305-0548.

Michel Gendreau, Ola Jabali, and Walter Rei. 50th anniversary invited article—future research directions in stochastic vehicle routing. *Transportation Science*, 50(4):1163–1173, 2016.

R. E. Gomory. On the relation between integer and noninteger solutions to linear programs. *Proc. Nat. Acad. Sci. U.S.A.*, 53:260–265, 1965. ISSN 0027-8424. URL `https://doi.org/10.1073/pnas.53.2.260`.

Ralph E Gomory. Some polyhedra related to combinatorial problems. *Linear algebra and its applications*, 2(4):451–558, 1969. URL `https://doi.org/10.1016/0024-3795(69)90017-2`.

Jean-Baptiste Hiriart-Urruty and Claude Lemaréchal. *Convex analysis and minimization algorithms I: Fundamentals*, volume 305. Springer science & business media, 1996.

YN Hoogendoorn and R Spliet. An improved integer L-shaped method for the vehicle routing problem with stochastic demands. *INFORMS Journal on Computing*, 35(2):423–439, 2023.

Mojtaba Hosseini and John Turner. Deepest cuts for benders decomposition. *Operations Research*, 2024. doi: 10.1287/opre.2021.0503. URL `https://doi.org/10.1287/opre.2021.0503`.

Ola Jabali, Walter Rei, Michel Gendreau, and Gilbert Laporte. Partial-route inequalities for the multi-vehicle routing problem with stochastic demands. *Discrete Applied Mathematics*, 177:121–136, 2014. ISSN 0166-218X.

Michael Jünger and Stefan Thienel. The ABACUS system for branch-and-cut-and-price algorithms in integer programming and combinatorial optimization. *Software: Practice and Experience*, 30(11):1325–1352, 2000.

Simge Küçükyavuz and Suvrajeet Sen. An introduction to two-stage stochastic mixed-integer programming. In *Leading Developments from INFORMS Communities*, pages 1–27. INFORMS, 2017.

Gilbert Laporte and Yves Nobert. A branch and bound algorithm for the capacitated vehicle routing problem. *Operations-Research-Spektrum*, 5(2):77–85, 1983.

Quentin Louveaux, Laurent Poirrier, and Domenico Salvagnin. The strength of multi-row models. *Mathematical Programming Computation*, 7(2):113–148, 2015.

Leonardo Lozano and J Cole Smith. A binary decision diagram based algorithm for solving a class of binary two-stage stochastic programs. *Mathematical Programming*, pages 1–24, 2022.

Leonardo Lozano, David Bergman, and Andre A Cire. Constrained shortest-path reformulations for discrete bilevel and robust optimization. *arXiv preprint arXiv:2206.12962*, 2022.

Jens Lysgaard, Adam N. Letchford, and Richard W. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100(2):423–445, 2004. ISSN 1436-4646.

Boris Sholimovich Mordukhovich and Nguyen Mau Nam. *An easy path to convex analysis and applications*. Springer, 2014.

Lewis Ntaimo. Fenchel decomposition for stochastic mixed-integer programming. *Journal of Global Optimization*, 55:141–163, 2013.

Matheus J Ota and Ricardo Fukasawa. Hardness of pricing routes for two-stage stochastic vehicle routing problems with scenarios. *Operations Research*, 2024.

Lucas Parada, Robin Legault, Jean-François Côté, and Michel Gendreau. A disaggregated integer L-shaped method for stochastic vehicle routing problems with monotonic recourse. *European Journal of Operational Research*, 2024.

Artur Pessoa, Ruslan Sadykov, Eduardo Uchoa, and François Vanderbeck. A generic exact solver for vehicle routing and related problems. *Mathematical Programming*, 183: 483–523, 2020.

Ragheb Rahmaniani, Teodor Gabriel Crainic, Michel Gendreau, and Walter Rei. The Benders decomposition algorithm: A literature review. *European Journal of Operational Research*, 259(3):801–817, 2017.

Ragheb Rahmaniani, Shabbir Ahmed, Teodor Gabriel Crainic, Michel Gendreau, and Walter Rei. The Benders dual decomposition method. *Operations Research*, 68(3): 878–895, 2020.

Ward Romeijnders, Rüdiger Schultz, Maarten H van der Vlerk, and Willem K Klein Haneveld. A convex approximation for two-stage mixed-integer recourse models with a uniform error bound. *SIAM Journal on Optimization*, 26(1):426–447, 2016.

Ruslan Sadykov and François Vanderbeck. BaPCod-a generic branch-and-price code. Technical report, Inria Bordeaux Sud-Ouest, 2021.

Alexander Schrijver. *Theory of linear and integer programming.* John Wiley & Sons, 1998.

Kiho Seo, Seulgi Joung, Chungmok Lee, and Sungsoo Park. A closest Benders cut selection scheme for accelerating the Benders decomposition algorithm. *INFORMS Journal on Computing*, 34(5):2804–2827, 2022.

Thiago Serra. Reformulating the disjunctive cut generating linear program. *Annals of Operations Research*, 295(1):363–384, 2020.

Eduardo F Silva and R Kevin Wood. Solving a class of stochastic mixed-integer programs with branch and price. *Mathematical programming*, 108:395–418, 2006.

Christian Tjandraatmadja and Willem-Jan van Hoeve. Target cuts from relaxed decision diagrams. *INFORMS Journal on Computing*, 31(2):285–301, 2019.

Eduardo Uchoa, Artur Pessoa, and Lorenza Moreno. Optimizing with Column Generation: Advanced branch-cut-and-price algorithms (Part I). Technical Report L-2024-3, Cadernos do LOGIS-UFF, Universidade Federal Fluminense, Engenharia de Produção, August 2024.

François Vanderbeck and Laurence A Wolsey. *Reformulation and decomposition of integer programs.* Springer, 2010.

Günter M Ziegler. *Lectures on polytopes*, volume 152. Springer Science & Business Media, 2012.

Jikai Zou, Shabbir Ahmed, and Xu Andy Sun. Stochastic dual dynamic integer programming. *Mathematical Programming*, 175:461–502, 2019.

A Proof of Theorem 2

To facilitate the reading, we repeat the theorem statement.

Theorem 2. *Let $C = \{y^*\} + \textsc{cone}(R)$ be a corner of Y. Let $(w^*, \theta^*) = (Qy^*, d^\intercal y^*)$, (w', θ') be a point in the relative interior of $\textsc{epi}(f_C)$, and $(\bar{w}, \bar{\theta})$ be an arbitrary point in $\mathbb{R}^p \times \mathbb{R}$. Define $E^\#$ as in (6) and consider the optimization problem*

$$z^\circ := \min_{\alpha,\alpha_0} \left\{ \alpha^\intercal(\bar{w} - w') + \alpha_0(\bar{\theta} - \theta') : (\alpha, \alpha_0) \in E^\# \right\}. \tag{7}$$

It holds that

(a) if $z^\circ \geq -1$, then $(\bar{w}, \bar{\theta})$ belongs to $\textsc{epi}(f_C)$;

(b) if $z^\circ = -\infty$ (i.e., Problem (7) is unbounded), then we have a vector $(\alpha, \alpha_0) \in \textsc{recc}(E^\#)$ such that $\alpha^\intercal w + \alpha_0\theta = \alpha^\intercal w' + \alpha_0\theta'$ defines an implicit equality of $\textsc{epi}(f_C)$ that is violated by $(\bar{w}, \bar{\theta})$;

(c) otherwise, $-\infty < z^\circ < -1$ and we have an extreme point (α, α_0) of $E^\#$ such that the inequality $\alpha^\intercal w + \alpha_0\theta \geq -1 + \alpha^\intercal w' + \alpha_0\theta'$ defines a facet of $\textsc{epi}(f_C)$ that is violated by $(\bar{w}, \bar{\theta})$.

Proof. We start with item (b). Suppose that Problem (7) is unbounded, so we have a certificate $(\alpha, \alpha_0) \in \textsc{recc}(E^{\#})$. Since $0 \in E^{\#}$, for all $\mu \in \mathbb{R}_+$, we have that $\mu(\alpha, \alpha_0) \in E^{\#}$. This implies that

$$\alpha^\intercal(w^* - w') + \alpha_0(\theta^* - \theta') \geq 0 \iff \alpha^\intercal w^* + \alpha_0\theta^* \geq \alpha^\intercal w' + \alpha_0\theta'. \tag{17}$$

The inequality above is valid for the unique extreme point of $\textsc{epi}(f_C)$, and since (α, α_0) satisfy all ray inequalities of $E^{\#}$ (and $\alpha_0 \geq 0$), we know that (17) is a valid inequality for $\textsc{epi}(f_C)$. Moreover, (17) is tight at (w', θ'), which is a point in the relative interior of $\textsc{epi}(f_C)$. This shows that (17) holds with equality for all points in $\textsc{epi}(f_C)$.

To prove (a), we assume by contradiction that $z^\circ \geq -1$ and $(\bar{w}, \bar{\theta}) \notin \textsc{epi}(f_C)$. Using the separating hyperplane theorem, we have a Fenchel cut $\alpha^\intercal w + \alpha_0\theta \geq \sigma_{\textsc{epi}(f_C)}(\alpha, \alpha_0)$ (with $\alpha_0 \geq 0$) that is valid for $\textsc{epi}(f_C)$ but is violated by $(\bar{w}, \bar{\theta})$. Suppose first that $\alpha^\intercal w' + \alpha_0\theta' = \sigma_{\textsc{epi}(f_C)}(\alpha, \alpha_0)$, meaning that the Fenchel cut associated with (α, α_0) defines an implicit equality of $\textsc{epi}(f_C)$. We can then follow the reasoning for item (b) to learn that $(\alpha, \alpha_0) \in \textsc{recc}(E^{\#})$; and as $\alpha^\intercal \bar{w} + \alpha_0\bar{\theta} < \sigma_{\textsc{epi}(f_C)}(\alpha, \alpha_0) = \alpha^\intercal w' + \alpha_0\theta'$, we have that $z^\circ = -\infty$, a contradiction. So we can safely assume that $\alpha^\intercal w' + \alpha_0\theta' > \sigma_{\textsc{epi}(f_C)}(\alpha, \alpha_0)$. Now subtract $\alpha^\intercal w' + \alpha_0\theta'$ from both sides of $\alpha^\intercal w + \alpha_0\theta \geq \sigma_{\textsc{epi}(f_C)}(\alpha, \alpha_0)$ to get

$$\alpha^\intercal(w - w') + \alpha_0(\theta - \theta') \geq \beta \coloneqq \sigma_{\textsc{epi}(f_C)}(\alpha, \alpha_0) - \alpha^\intercal w' - \alpha_0\theta', \tag{18}$$

where $\beta < 0$. Dividing both sides of (18) by $|\beta|$ and setting $\alpha' = (1/|\beta|)\alpha$ and $\alpha_0' = (\alpha_0/|\beta|)$ yields the inequality $\alpha'(w - w') + \alpha_0'(\theta - \theta') \geq -1$, which implies that $z^\circ < -1$, a contradiction. To see this, note that since the Fenchel cut defined by (α, α_0) is violated by $(\bar{w}, \bar{\theta})$, we know that $\alpha'(\bar{w} - w') + \alpha_0'(\bar{\theta} - \theta') < -1$. Moreover, as $(\alpha')^\intercal w + \alpha_0'\theta \geq -1$ is a valid inequality for $E = \textsc{epi}(f_C) - (w', \theta')$, we have that (α', α_0') belongs to $E^{\#}$.

Lastly, suppose that we have (α, α_0) such that $\alpha^\intercal(\bar{w} - w') + \alpha_0(\bar{\theta} - \theta') = z^\circ < -1$ and (α, α_0) is a vertex of $E^{\#}$. By translation, it suffices to show that $\alpha^\intercal w + \alpha_0\theta \geq -1$ is a facet of E. Let a minimal representation for E be

$$E = \left\{ (\alpha, \alpha_0) \in \mathbb{R}^{p+1} : \begin{array}{ll} (\rho^j)^\intercal x + \rho_0^j\theta \geq -1, & \forall j \in J \\ (\rho^j)^\intercal x + \rho_0^j\theta = 0, & \forall j \in J' \end{array} \right\}, \tag{19}$$

where J and J' are index sets. For all $j \in J \cup J'$, we know that $(\rho^j, \rho_0^j) \in E^{\#}$. Additionally, as $\alpha^\intercal w + \alpha_0\theta \geq -1$ is a valid inequality for E, there exists $\mu \geq 0$ and $\mu' \geq 0$ such that $(\alpha, \alpha_0) = \sum_{j \in J} \mu_j(\rho^j, \rho_0^j) + \sum_{j \in J'} \mu_j'(\rho^j, \rho_0^j)$ and $-1 = \sum_{j \in J} \mu_j(-1) + \sum_{j \in J'} \mu_j'(0)$ (this follows from the fact that (19) is a minimal representation, see Section 3.9 of Conforti et al. (2014)). Since (α, α_0) is an extreme point of $E^{\#}$, it follows that $|J| = 1$ and $\alpha^\intercal w + \alpha_0\theta \geq -1$ defines a facet of E. □

B Corner Benders' cuts as restricted dual subproblems

Consider the setup in Remark 1 and assume that C is a corner associated with a set of basic variables B (and $N = [m] \setminus B$). In this case, it is known that we can write $C = \{y \in \mathbb{R}^m : Ay = b, y_B \in \mathbb{R}^B, y_N \in \mathbb{R}^N_+\}$ (see Chapter 6.1 of Conforti et al. (2014)), and therefore we can check if $(\bar{w}, \bar{\theta})$ lies in $\textsc{epi}(f_C)$ by solving the problem

$$\begin{array}{lll} \min\limits_{y} & 0 & \\ & -d_B^\intercal y_B - d_N^\intercal y_N \geq -\bar{\theta}, & (\alpha_0) \\ & Q_B y_B + Q_N y_N = \bar{w}, & (\alpha) \\ & A_B y_B + A_N y_N = b, & (\nu) \\ & y_B \in \mathbb{R}^B, y_N \in \mathbb{R}_+^N. & \end{array} \tag{20}$$

By LP duality, we have that $(\bar{w}, \bar{\theta}) \in \textsc{epi}(f_C)$ if and only if the optimal value of the optimization problem below is at most zero.

$$\max_{\nu,\alpha,\alpha_0} \quad \nu^\intercal b + \alpha^\intercal \bar{w} - \alpha_0 \bar{\theta}$$

$$-\alpha_0 d_B + Q_B^\intercal \alpha + A_B^\intercal \nu = 0, \tag{21a}$$

$$-\alpha_0 d_N + Q_N^\intercal \alpha + A_N^\intercal \nu \leq 0, \tag{21b}$$

$$\alpha_0 \geq 0. \tag{21c}$$

Observe that if variables y_B were nonnegative in Problem (20), then (21a) would be inequalities instead of equalities. In this sense, relaxing $\textsc{epi}(f_Y)$ to $\textsc{epi}(f_C)$ is equivalent to constraining the Benders' dual subproblem according to (21a). This idea of restricting a cut-generating program to separate inequalities more efficiently was explored by Chen and Luedtke (2022) in the context of stochastic programs with integer subproblems. Our discussion here shows that corner Benders' cuts can also be interpreted from a similar lens, except that here we constrain the cut-generating program according to the given basis B.

C Lagrangian dual problem for block-diagonal matrices

Suppose Problem (LP) can be written as

$$\begin{array}{llll} \min & c^\intercal x + \sum_{i\in[t]} (d^i)^\intercal y^i & & \\ & T^i x + Q^i y^i = h^i, & \forall i \in [t], & (\alpha^i) \\ & x \in X, & & \\ & y^i \in Y^i, & \forall i \in [t]. & \end{array} \tag{22}$$

Reformulating Problem (22) in an epigraphical form yields

$$\min_{x,\theta} \left\{ c^\intercal x + \sum_{i\in[t]} \theta_i : x \in X,\ (h^i - T^i x, \theta^i) \in \textsc{epi}(f^i),\ \forall i \in [t] \right\}, \tag{23}$$

where $f^i(w) = \min\{(d^i)^\intercal y^i : Q^i y^i = w, y^i \in Y^i\}$ and $\textsc{epi}(f^i) = \{(w, \theta) : \theta \geq f^i(w)\}$, for all $i \in [t]$.

Let $\mathcal{F}'$ be the feasible region of Problem (23). We can easily adapt Theorem 3 to Problem (23). Indeed, given $\rho \in \mathbb{R}^n$ and $\rho_0^1, \ldots, \rho_0^t \in \mathbb{R}_+$, one can check that if $\hat{\alpha}^1, \ldots, \hat{\alpha}^t$ is optimal for

$$\max_{\alpha^1,\ldots,\alpha^t} \left\{ -\sum_{i\in[t]} (\alpha^i)^\intercal h^i + \sigma_X \left(\rho + \sum_{i\in[t]} (T^i)^\intercal \alpha^i \right) + \sum_{i\in[t]} \sigma_{Y^i} \left((Q^i)^\intercal \alpha^i + \rho_0^i d^i \right) \right\}, \tag{24}$$

then the Benders' cuts $(\hat{\alpha}^i)^\intercal(h^i - T^i x) + \rho_0^i \theta^i \geq \sigma_{\text{EPI}(f^i)}(\hat{\alpha}^i, \rho_0^i)$, for $i \in [t]$, imply the inequality $\rho^\intercal x + \sum_{i\in[t]} \rho_0^i \theta \geq \sigma_{\mathcal{F}'}(\rho, \rho_0^1, \dots, \rho_0^t)$. Proposition 1 can also be adapted to this block-diagonal case, yielding one corner C^i for each block of constraints $i \in [t]$.

D Full cutting-plane loop for the VRPSD

In the following algorithm, we denote by (VRPSD-BASE) the linear program below.

$$\begin{aligned}
\min_{x,\theta} \quad & c^\intercal x + \sum_{v \in V_+} \theta'_v \\
& x(\delta(0)) = 2k, \\
& x(\delta(v)) = 2, && \forall v \in V_+, \\
& x_e \leq 1, && \forall e \in E \setminus \delta(0), \\
& x \in [0,2]^E, \theta' \in \mathbb{R}_+^{V_+}.
\end{aligned} \tag{VRPSD-BASE}$$

Algorithm 3 Cutting-plane loop for the VRPSD

```
procedure VRPSD-CuttingPlane(algorithm)
    if algorithm = lagrange or algorithm = corner then
        Get $\hat{\alpha}$ optimal for Problem (8) by solving Problem (26).
        Get a shortest-path tree $\mathcal{T}$ with respect to weights $\{d'_a = d_a + \hat{\alpha}_{\textsc{edge}(a)}\}_{a \in \mathcal{N}}$.
        Set $C = \{y^*\} + \textsc{cone}(R)$ to be the corner associated with $\mathcal{T}$.
        Set $y'$ to be a point in the relative interior of $Y$.
        $(w', \theta') \leftarrow (Qy', d^\intercal y' + 1)$
    $\tilde{\mathcal{R}} \leftarrow \emptyset$
    Initialize (VRPSD-LP) as the linear program (VRPSD-BASE).
    if algorithm = lagrange then
        Add Lagrangian Cut $\hat{\alpha}^\intercal x + \theta \geq \sigma_Y(Q^\intercal \hat{\alpha} + d)$ to (VRPSD-LP).
    separated ← true
    prevBound ← $-\infty$
    count ← 1
    while separated and count $\leq 10$ do
        separated ← false
        Solve (VRPSD-LP) to get a candidate solution $(\bar{x}, \bar{\theta}')$.
        $\bar{z} \leftarrow c^\intercal \bar{x} + \sum_{v \in V_+} \bar{\theta}'_v$
        if $\bar{z}$ − prevBound $\leq 10^{-3}$ then
            count ← count + 1
        else
            count ← 1
        prevBound ← $\bar{z}$
        if Heuristics found violated RCI's or ILS cuts then
            Add violated cuts to (VRPSD-LP).
            separated ← true
        else if algorithm = benders then
            if Problem (25) is feasible then
                Let $\bar{\alpha}$ and $\bar{\alpha}_0$ be optimal for Problem (25).
                Add inequality $\bar{\alpha}_0 \left(\sum_{v \in V_+} \theta'_v\right) \geq \bar{\alpha}^\intercal x$ to (VRPSD-LP).
                separated ← true
        else if algorithm = corner then
            $((\rho, \rho_0, \beta), z^\circ, \tilde{R}) \leftarrow$ SolvePolar($(\bar{x}, \bar{\theta}), (Qy', d^\intercal), y^*, R, \tilde{R}$)
            if $z^\circ < -1$ then
                separated ← true
                Add inequality $\rho^\intercal x + \rho_0 \left(\sum_{v \in V_+} \theta'_v\right) \geq \beta$ to (VRPSD-LP).
                if $z^\circ = -\infty$ then
                    Add inequality $\rho^\intercal x + \rho_0 \left(\sum_{v \in V_+} \theta'_v\right) \leq \beta$ to (VRPSD-LP).
    return (VRPSD-LP)
```

E Separation of different Benders' cuts

We now describe how the different Benders' cuts evaluated in our computational experiments are separated.

Benders' decomposition. To separate optimality/feasibility cuts in the standard Benders decomposition approach, we used the well-known normalization proposed by Fischetti et al. (2010). This is one of the normalizations used in CPLEX (Bonami et al., 2020) (another option is normalizing the righthand side as in Conforti and Wolsey (2019)), and it is used as a baseline approach for recently proposed Benders' cuts selection methods (Seo et al., 2022; Brandenberg and Stursberg, 2021; Hosseini and Turner, 2024).

Let $(\bar{x}, \bar{\theta}') \in \mathbb{R}^E \times \mathbb{R}^{V_+}$ be a candidate solution. Set $\bar{\theta} = \sum_{v \in V_+} \bar{\theta}'_v$ and consider the flow-based formulation described in Section 4.3.2. Our Benders' decomposition implementation separates $(\bar{x}, \bar{\theta})$ from $\textsc{epi}(f_Y)$ by solving the following optimization problem:

$$
\begin{aligned}
\max_{\alpha, \alpha_0, \nu} \quad & \alpha^\intercal \bar{x} - k\nu_s + k\nu_t - \alpha_0 \bar{\theta} && \\
& \nu_{\mathcal{S}_1} - \nu_{\mathcal{S}_2} + \alpha_{\textsc{edge}(\mathcal{S}_1\mathcal{S}_2)} \leq \alpha_0\, d_{\mathcal{S}_1\mathcal{S}_2}, && \forall \mathcal{S}_1\mathcal{S}_2 \in \mathcal{A}, && \text{(25a)} \\
& \|\alpha\|_1 + \alpha_0 \leq 1, && && \text{(25b)} \\
& \alpha_0 \geq 0. && && \text{(25c)}
\end{aligned}
$$

Lagrangian cut. To solve the Lagrangian dual problem in Proposition 1, we substitute the x_e variables according to the linking constraints $\sum_{a \in \mathcal{A}:\textsc{edge}(a)=e} y_a = x_e$ in Problem (14), and we solve the following flow-based formulation.

$$
\begin{aligned}
\min_{y} \quad & \sum_{a \in \mathcal{A}} (c_{\textsc{edge}(a)} + d_a)\, y_a && && \\
& y(\delta^-(\mathcal{S})) - y(\delta^+(\mathcal{S})) = k \cdot \mathbb{1}_t - k \cdot \mathbb{1}_s, && \forall \mathcal{S} \in \mathcal{V}, && (\nu_{\mathcal{S}}) \quad \text{(26a)} \\
& \sum_{e \in \delta(v)} \left(\sum_{a \in \mathcal{A}:\textsc{edge}(a)=e} y_a \right) = 2, && \forall v \in V_+, && (\gamma_{\{v\}}) \quad \text{(26b)} \\
& \sum_{e \in \delta(S)} \left(\sum_{a \in \mathcal{A}:\textsc{edge}(a)=e} y_a \right) \geq 2k(S), && \forall S \subseteq V_+, |S| \geq 2, && (\gamma_S) \quad \text{(26c)} \\
& y_a \geq 0, && \forall a \in \mathcal{A}, && \text{(26d)}
\end{aligned}
$$

where inequalities (26c) are separated heuristically with the CVRPSEP package (Lysgaard et al., 2004), and we stop the cut-generating loop earlier if it runs for more than 60 seconds.

Let $\bar{\nu}$ and $\bar{\gamma}$ be optimal dual variables associated with Formulation 26. For each edge $e \in E$, we set $\hat{\alpha}_e = c_e - \sum_{\emptyset \subset S \subseteq V_+} \bar{\gamma}_S$, and we prove in Appendix G that $\hat{\alpha}$ is optimal for Problem 8.

Corner Benders' cuts. The first step in generating corner Benders' cuts is to obtain $\hat{\alpha}$ optimal for Problem 8 by solving Problem 26, as described earlier. We then apply Proposition 1 to find a corner that is optimal with respect to $\sigma_Y(Q^\intercal \hat{\alpha} + d)$. As illustrated in Section 4.4, we obtain this corner by setting weights $d'_a = d_a + \hat{\alpha}_{\textsc{edge}(a)}$, for each $a \in \mathcal{A}$, and then using these weights to find a shortest path tree $\mathcal{T}$ rooted at s. To obtain the interior point y' in Algorithm 2, we optimize a zero-objective function over the set Y using Gurobi's barrier algorithm.

In our implementation of Algorithm 1, given a candidate solution $(\bar{\alpha}, \bar{\alpha}_0)$ to (CGLP), we first loop through all arcs in $\mathcal{A} \setminus \mathcal{T}$ to construct mappings (implemented as arrays) $M_1 : E \to \mathcal{A}$ and $M_2 : E \to \mathbb{Q}$ such that, for each edge $e \in E$, $M_1(e)$ is an arc that belongs to $\arg\min \{\bar{\alpha}^\intercal(Qr^a) + \bar{\alpha}_0(d^\intercal r^a) : \textsc{edge}(a) = e,\ a \in \mathcal{A} \setminus \mathcal{T}\}$ (or $\emptyset$ if no such edge exists) and $M_2(e)$ is the value of $\bar{\alpha}^\intercal(Qr^{M_1(e)}) + \bar{\alpha}_0(d^\intercal r^{M_1(e)})$ (or $+\infty$ if $M_1(e) = \emptyset$). (Recall that we defined in Section 4.4 that $r^a = \mathbb{1}(C^a)$, for each $a \in \mathcal{A} \setminus \mathcal{T}$.) Next, we create an array L of edges e that is in nondecreasing order of $M_2(e)$. We then traverse over the first $\min\{|L|, 100\}$ edges e in L, and if $M_2(e) < 0$, we add the inequality $\bar{\alpha}^\intercal(Qr^{M_1(e)}) + \bar{\alpha}_0(d^\intercal r^{M_1(e)}) \geq 0$ to (CGLP).

In Appendix F, we also show how we further explore the special structure of corners arising from network flow polytopes to separate the ray inequalities faster. Although this acceleration is not essential for the effectiveness of corner Benders' cuts, we use it because it can lead to performance improvements. Our preliminary experiments in the instances with large k and $|V| = 40$ indicate that using this acceleration can reduce the execution time of the root cutting-plane loop by an average factor of two, translating to a reduction of 25 seconds in the average runtime.

F Using network flow structure to accelerate the separation of corner Benders' cuts

Consider the separation of ray inequalities in Algorithm 1, specifically the loop in lines 8-10. Since matrix Q has m columns and p rows, each iteration of this loop has a time complexity of $\mathcal{O}(|R|mp)$. In the specific case where Y is a network flow polytope associated with a network $\mathcal{N} = (\mathcal{V}, \mathcal{A})$ and R represents the cycles described in Section 4.4, this time complexity simplifies to $\mathcal{O}(|\mathcal{A}|^2 p)$. In contrast, Algorithm 4 below detects violated ray inequalities in $\mathcal{O}(|\mathcal{A}|p)$ time. Furthermore, in the particular case of the VRPSD formulation shown in Section 4.3.2, the product $\bar{\alpha}^\intercal Q_a$ simplifies to $\bar{\alpha}_{\text{EDGE}(a)}$, which can be computed in $\mathcal{O}(1)$. Consequently, for the VRPSD, Algorithm 4 has a time complexity of $\mathcal{O}(|\mathcal{A}|)$.

Algorithm 4 Separation of Corner Benders' Cuts for Network Flow Subproblems

Input: $(\bar{\alpha}, \bar{\alpha}_0) \in \mathbb{R}^p \times \mathbb{R}_+$, network $(\mathcal{N} = (\mathcal{V}, \mathcal{A}), b, u)$, spanning tree $\mathcal{T}$ and basic feasible solution y^*.

Output: Set of rays R'' such that $\bar{\alpha}^\intercal(Qr) + \bar{\alpha}_0(d^\intercal r) < 0$, for all $r \in R''$.

```
procedure FindViolatedRay($\bar{\alpha}, \bar{\alpha}_0, \mathcal{N} = (\mathcal{V}, \mathcal{A}), b, u, \mathcal{T}, y^*$)
    $R'' \leftarrow \emptyset$
    Let $\vec{\mathcal{T}}$ be the arc set of an arborescence obtained by rooting $\mathcal{T}$ at an arbitrary vertex $s$.
    Let $v_1 = s, v_2, \ldots, v_{|\mathcal{V}|}$ be an ordering obtained with a preorder traversal of $\vec{\mathcal{T}}$.
    $\nu_s \leftarrow 0$
    for $j = 2, \ldots, |\mathcal{V}|$ do
        Let $i < j$ be such that $(v_i, v_j) \in \vec{\mathcal{T}}$.
        if $(v_i, v_j) \in \mathcal{T}$ then
            $\nu_{v_j} \leftarrow \nu_{v_i} + \bar{\alpha}^\intercal Q_a + \bar{\alpha}_0\, d_a$
        else
            $\nu_{v_j} \leftarrow \nu_{v_i} - \bar{\alpha}^\intercal Q_a - \bar{\alpha}_0\, d_a$
    for $uv \in \mathcal{A} \setminus \mathcal{T}$ do
        if $\nu_u - \nu_v + \bar{\alpha}^\intercal Q_{uv} + \bar{\alpha}_0\, d_{uv} < 0$ then
            $R'' \leftarrow R'' \cup \{r^{uv}\}$
    return $R''$
```

The intuition behind Algorithm 4 is as follows: since the set of rays R corresponds to the cycles formed by adding an arc from $\mathcal{A} \setminus \mathcal{T}$ to the spanning tree $\mathcal{T}$, we can precompute the *potentials* $\nu \in \mathbb{R}^{\mathcal{V}}$ along $\mathcal{T}$. In other words, we set ν_v to be the cost (with respect to weights $d' = Q^\intercal \bar{\alpha} + \bar{\alpha}_0\, d$) of the unique path in $\mathcal{T}$ connecting s and v. For a given arc $uv \in \mathcal{A}$, it is straightforward to verify that $\nu_u - \nu_v + \bar{\alpha}^\intercal Q_{uv} + \bar{\alpha}_0\, d_{uv} = \alpha^\intercal(Qr^{uv}) + \alpha_0(d^\intercal r^{uv})$.

Thus, Algorithm 4 efficiently solves the separation problem for the ray inequalities in the polar set (6).

G Solving the Lagrangian cut generating problem with a linear program containing only y variables

Suppose that $X = \{x \in \mathbb{R}^n_+ : (A_x)x \geq b_x\}$ and $Y = \{y \in \mathbb{R}^m_+ : (A_y)y \geq b_y\}$. In addition, assume that $T = -I$, $h = \mathbf{0}$ and $Q \in \mathbb{R}^{n\times m}_+$ (all entries of Q are nonnegative). By Theorem 3, to obtain a Lagrangian cut we need to solve

$$\max_{\alpha\in\mathbb{R}^n} \left\{\sigma_X(c-\alpha) + \sigma_Y(Q^\intercal \alpha + d)\right\}. \tag{27}$$

One way to solve Problem (27) is to first rewrite Problem (LP) to get the following primal-dual pair.

$$\begin{aligned} \min_{x,y}\quad & c^\intercal x + d^\intercal y \\ & x = Qy, && (\alpha) \\ & (A_x)x \geq b_x, && (\gamma) \\ & (A_y)y \geq b_y, && (\nu) \\ & x, y \geq 0. \end{aligned} \tag{28}$$

$$\begin{aligned} \max_{\gamma,\nu}\quad & \gamma^\intercal b_x + \nu^\intercal b_y \\ & \alpha + (A_x)^\intercal \gamma \leq c, \\ & -Q^\intercal \alpha + (A_y)^\intercal \nu \leq d, \\ & \gamma, \nu \geq 0. \end{aligned} \tag{29}$$

It follows from LP duality that if $(\bar{\alpha}, \bar{\gamma}, \bar{\nu})$ is optimal for Problem (29), then $\bar{\alpha}$ is optimal for Problem (27) (see Frangioni (2005) for a nice proof via "partial dualization").

Now substitute $x = Qy$ in Problem (28) to get

$$\begin{aligned} \min_{y}\quad & (Q^\intercal c + d)^\intercal y \\ & (A_x)Qy \geq b_x, && (\gamma) \\ & (A_y)y \geq b_y, && (\nu) \\ & y \geq 0. \end{aligned} \tag{30}$$

$$\begin{aligned} \max_{\gamma,\nu}\quad & \gamma^\intercal b_x + \nu^\intercal b_y \\ & Q^\intercal (A_x)^\intercal \gamma + (A_y)^\intercal \nu \leq Q^\intercal c + d, \\ & \gamma, \nu \geq 0. \end{aligned} \tag{31}$$

Let $(\hat{\gamma}, \hat{\nu})$ be optimal for Problem (31). Set $\hat{\alpha} = c - (A_x)^\intercal \hat{\gamma}$ and notice that $(\hat{\alpha}, \hat{\gamma}, \hat{\nu})$ is optimal for Problem (29), as desired.

H On our implementation of Parada et al. (2024)

We now describe a minor modification that we made to the algorithm proposed by Parada et al. (2024) to address a small issue in their original description. For that, we briefly review the inequalities they use and their separation routines for fractional solutions.

The DL-shaped method proposed by Parada et al. (2024) separates two types of ILS inequalities: *P-cuts* and *S-cuts*. Given an elementary route $R = (v_1, \ldots, v_\ell)$, a P-cut has the form

$$\sum_{i=1}^{\ell} \theta'_{v_i} \geq \mathbb{E}[\mathcal{Q}(R)] \cdot (1 + (x_{v_1 v_2} - 1) + \cdots + (x_{v_{\ell-1} v_\ell} - 1)). \tag{32}$$

Given $\emptyset \subsetneq S \subseteq V_+$, an S-cut has the form

$$\sum_{v\in S} \theta'_v \geq L(S) \cdot \left(1 + \left(\sum_{uv\in E: u,v\in S} x_{uv}\right) - |S| + \left\lceil \frac{\bar{q}(S)}{C} \right\rceil\right), \tag{33}$$

where $L(S)$ is a *recourse lower bound* as defined in Parada et al. (2024) (we use the lower bounds L_1 and L_2 from their work).

Let $(\bar{x}, \bar{\theta}')$ be a fractional solution for (VRPSD-LP) and consider Line 24 of Algorithm 3. The separation algorithm of Parada et al. (2024) first attempts to find violated S-cuts associated with the customer sets generated by CVRPSEP during the separation of RCIs. It then searches for additional violated S-cuts and P-cuts by examining the graph $G(\bar{x})$, which is associated with the edges in the support of $\bar{x}$ that are not incident to the depot, that is, $V(G(\bar{x})) = V_+$ and $E(G(\bar{x})) = \{e \in E \setminus \delta(0) : \bar{x}_e > 0\}$. For each connected component $\mathcal{H}$ of $G(\bar{x})$, the algorithm verifies if the S-cut for $S = V(\mathcal{H})$ is violated by $(\bar{x}, \bar{\theta}')$, and if so, it adds the inequality to the model. Parada et al. (2024) then adds a P-cut whenever $|V(\mathcal{H})| = |E(\mathcal{H})| + 1$, since they claim that in this case $\mathcal{H}$ is a path.

However, this last claim is not always correct, so we instead explicitly verify whether $\mathcal{H}$ is a path by traversing its nodes. As a counterexample, consider the fractional solution $\bar{x}$ in Figure 8. Here, $G(\bar{x})$ is made of a single connected component $\mathcal{H}$ that contains a vertex with degree 3, so it is not a path despite satisfying $|V(\mathcal{H})| = |E(\mathcal{H})| + 1$.

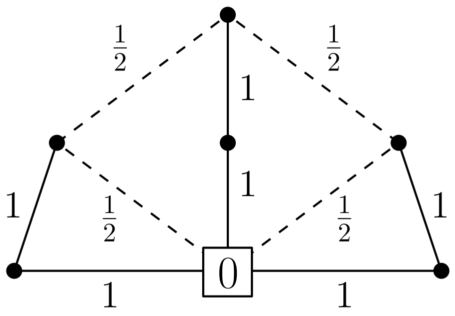


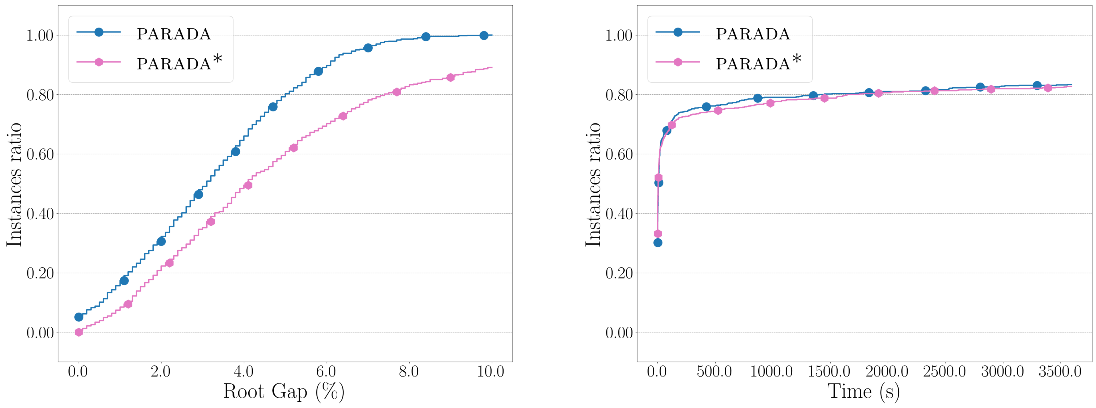
Figure 8: Example of a fractional solution $\bar{x}$ where the associated connected component $\mathcal{H}$ satisfies $|V(\mathcal{H})| = |E(\mathcal{H})| + 1$ but is not a path.

We remark, however, that the issue illustrated in Figure 8 does not seem to occur often, and our preliminary experiments suggest that our fix does not lead to any significant performance improvement.

Figures 9a and 9b compare our implementation of the algorithm from Parada et al. (2024) with the original results reported by the authors. These figures are generated in the same way as Figure 5, except that here we consider all instances together. The curve labeled PARADA* corresponds to the results from the authors' online table, while PARADA denotes our implementation. We note that our implementation achieves comparable execution times but consistently obtains smaller root gaps. This improvement is likely due to our custom cutting-plane loop at the root node of the branch-and-bound tree, as described in Section 5.2 and detailed in Algorithm 3.

I Experiments on the separation of corner Benders' cuts

Here we examine the efficiency of Algorithm 1 (i.e. SOLVEREVERSEPOLAR) to separate corner Benders' cuts. For the same instance as in Figure 7, we show in Figure 10 how much time we take to separate each Benders' cut. For example, a point $p = (p_1, p_2) \in \mathbb{Z}_+ \times \mathbb{R}_+$ in the line of CORNER (time) indicates that CORNER took p_2 seconds to separate the p_1-th corner Benders' cut. Similarly, a point $p = (p_1, p_2) \in \mathbb{Z}_+^2$ in the line of CORNER (rays) indicates that to separate the p_1-th corner Benders' cut, SOLVEPOLAR separated p_2 ray



(a) Gap for all instances.

(b) Time for all instances.

Figure 9: Comparison between our implementation of the algorithm from Parada et al. (2024) and the results reported in their online table.

inequalities. To compare, we also show in Figure 10 the time that BENDERS took to separate each Benders' cut.

Algorithm CORNER only took longer than BENDERS when separating the first two corner Benders' cuts. Additionally, the dotted line illustrates that the number of ray inequalities separated by SOLVEREVERSEPOLAR rapidly decreases, highlighting the effectiveness of our "warm-start" strategy, which is encoded in the choice of the set $\tilde{R}$.

It is also worth noting that the network $\mathcal{N} = (\mathcal{V}, \mathcal{A})$ for this instance contains 608 nodes and 12647 arcs. Consequently, the optimal corner $C = \{y^*\} + \text{CONE}(R)$ used in our approach has $|R| = |\mathcal{A}| - |\mathcal{V}| + 1 = 12040$ rays. Summing the number of rays separated in each iteration of CORNER, we find that a total of 4405 ray inequalities were separated. This collaborates with our observation in Section 3.3 that the projection $r \to (Qr, d^\intercal r)$ makes many ray inequalities redundant. By generating these ray inequalities as cutting planes, we can efficiently generate facet-defining inequalities for $\text{EPI}(f_C)$ without having to repeatedly solve linear programs that have $|R| + 1$ rows.

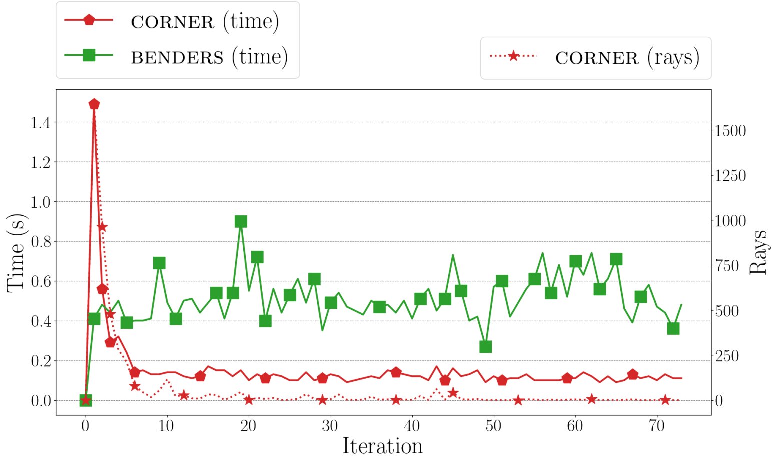


Figure 10: The solid lines indicate the time to separate each (corner) Benders' cuts. The dotted line shows how many ray inequalities were separated by SolveReversePolar for each corner Benders' cut. Experiments were executed on instance "30_7_0.95_1.00_8" of Parada et al. (2024) ($|V| = 30$ and $k = 7$).