A new framework to generate Lagrangian cuts in multistage stochastic mixed-integer programming

Christian Füllner[1*], X. Andy Sun[2], and Steffen Rebennack[1]

[1]Institute for Operations Research (IOR), Stochastic Optimization (SOP), Karlsruhe Institute of Technology (KIT), Karlsruhe, Germany
[*]Address correspondence to: christian.fuellner@kit.edu
[2]Sloan School of Management, Massachusetts Institute of Technology, USA

Abstract

Based on recent advances in Benders decomposition and two-stage stochastic integer programming we present a new generalized framework to generate Lagrangian cuts in multistage stochastic mixed-integer linear programming. This framework can be incorporated into existing solution methods, such as the stochastic dual dynamic integer programming (SDDiP) algorithm. We show how different normalization techniques can be applied in order to generate cuts satisfying specific properties with respect to the convex hull of the epigraph of the value functions, e.g. having a maximum depth or being facet-defining. We provide computational results to evaluate the efficacy and performance of different normalizations in our new framework, showing that compared to existing techniques from the literature significantly better lower bounds can be obtained.

1 Introduction

In this paper, we study cut generation strategies that can be applied in decomposition methods for solving multi-stage stochastic mixed-integer linear programs (MS-MILP). More precisely, we present an alternative framework for the generation of Lagrangian cuts, as they are used for instance in stochastic dual dynamic integer programming (SDDiP) proposed by [33].

1.1 Motivation and Prior Work

Multistage stochastic programs are very relevant to model decision-making processes in practice because often sequential decisions have to be made over a finite number of stages and under uncertainty considering the problem data of the following stages. For the solution of these problems, tailored decomposition methods are most widely used, among the most prominent ones nested Benders decomposition (BD) [6] and stochastic dual dynamic programming (SDDP) [25]. These methods decompose the large-scale problem into stage- and scenario-specific subproblems, coupled by state variables and so-called *value functions*, denoted $Q_n(\cdot)$. For *linear* problems. these functions are convex polyhedral and can be exactly represented by finitely many affine functions called *cuts*

[7]. However, in many applications some of the decision variables have to be integer or binary. In this case, we obtain an MS-MILP and the value functions are in general non-convex and discontinuous.

A key challenge is that in this case, linear under-estimators are in general not tight. They may at best yield the closed convex envelope $\overline{\text{co}}(Q_n)(\cdot)$ of $Q_n(\cdot)$, which is the pointwise supremum of all affine functions majorized by $Q_n(\cdot)$ [4]. Even this property is not achieved by classical Benders cuts in general. Therefore, more focus has been put on Lagrangian cuts lately, which are constructed by solving special Lagrangian dual problems [33]. These Lagrangian cuts have useful properties: They are valid under-estimators of $Q_n(\cdot)$ [33] and they can be used to recover $\overline{\text{co}}(Q_n)(\cdot)$ [19]. As shown in [33], if all state variables of the MS-MILP are binary, this even ensures tightness for $Q_n(\cdot)$, which is sufficient to establish almost sure finite convergence of SDDiP.

Nonetheless, applying Lagrangian cuts computationally in practice comes with some considerable challenges: First, Lagrangian dual problems are often degenerate with multiple optimal solutions. Even if all the cuts associated with these solutions are tight, their approximation quality may differ significantly. This issue is especially common for the binary state space required in SDDiP because all cuts are constructed at extreme points of the state space.

Second, even if the Lagrangian dual is a convex optimization problem, it may be very costly to solve repeatedly due to its non-smooth objective function. This is only aggravated if the state space is artificially increased by a binary expansion, as proposed in SDDiP for the case of non-binary state variables [33]. This drawback is already identified in the original SDDiP work [33]. It is concluded that performance-wise the improvement in cut quality is often not worth the significant increase in solution time.

Finally, tight Lagrangian cuts are crucial to ensure theoretical convergence of SDDiP, but this convergence may be quite slow. At worst, a complete enumeration of the binary state space is required. It is plausible that there exist alternative, possibly non-tight, cuts that may significantly speed-up the convergence process.

For the aforementioned reasons, in this paper we address the question of how to improve the generation and usage of Lagrangian cuts in decomposition methods for MS-MILP such as SDDiP.

Some first attempts with this aim have been made recently. In the original SDDiP paper it is proposed to combine tight cuts with strengthened Benders cuts that are not tight in general, but outperform classical Benders cuts and are efficient to compute [33]. Rahmaniani et al. [26] present a heuristic to generate Lagrangian cuts more efficiently using inner approximations or partial relaxations. Chen and Luedtke [11] suggest to restrict the feasible set of dual multipliers to the span of Benders cut coefficients of previous iterations for two-stage problems.

In addition, there has been a lot of research on alternative cut generation techniques for BD, which may be applicable to the stochastic and Lagrangian setting as well. To address degeneracy and dominated cuts in BD, Magnanti and Wong [23] present a two-step approach to generate Pareto-optimal cuts. Their ideas are improved by [24] who shows that it is sufficient to solve a single optimization problem to generate Pareto-optimal cuts. Sherali and Lunday [29] propose to generate certain maximal non-dominated Benders cuts by solving a perturbation of the original subproblem.

A novel framework to generate Benders cuts is introduced in [17]. In contrast to standard BD, it allows to generate optimality and feasibility cuts in a unified way using the same cut generation problem. Even more, several different cut generation techniques can be explored. More precisely, applying the framework initially leads to an unbounded separation problem. For the actual cut generation, unbounded rays have to be identified, which allows for a lot of methodological flexibility.

Fischetti et al. [17] show that using a special normalization of the cut generation problem, the obtained cuts can be proven to correspond to minimal infeasible subsystems. Hosseini and Turner [22] use a different normalization to generate *deep* cuts in BD, which are characterized by their property to maximize the distance between the separating hyperplane and the point to separate. They report considerable performance gains compared to classical BD. The idea of deep cuts is not new, but priorly discussed in context of disjunctive programming [10, 13]. Brandenberg and Stursberg [9] show how facet-defining and Pareto-optimal cuts can be generated in BD using the unified framework and the so-called reverse polar set, see also [30]. It is shown that the performance improvement using this approach is significant. The same cut generation procedure is put forward in [27], but motivated from a different angle, that is geometrically. A different approach to separate facet-defining cuts is presented by [12] for disjunctive programming. To our knowledge, only the work by Fischetti et al. has been applied to the stochastic setting and to Lagrangian cuts so far [11]. The authors use the unified framework and a specific normalization technique to generate Lagrangian cuts for two-stage stochastic MILPs.

In this paper, we provide a more general framework for the multistage case and compare various different normalization techniques from a theoretical and computational perspective. In particular, we analyze the Lagrangian cuts that are obtained if the Lagrangian dual is normalized using norm constraints or linear constraints. We show that under some assumptions, these cuts are deep, facet-defining or Pareto-optimal.

1.2 Contribution

The key contributions of this paper are summarized below.

(1) We show how the alternative cut generation framework proposed for BD from [17] can be applied to the generation of Lagrangian cuts for MS-MILPs. This idea has already been used by [11] in two-stage stochastic MILPs, but to our knowledge has not been extended to the multistage case yet.

(2) As the Lagrangian dual problems in this framework are unbounded, some normalization is required to select cut coefficients in a reasonable way. We draw on recent concepts for cut selection in BD, such as optimizing over the reverse polar set [9] or generating deep cuts [22], and extend them to the stochastic and Lagrangian setting. This way, we obtain a variety of different normalization techniques and by that generalize the cut generation approach from [11]. We show that depending on the chosen normalization, cuts satisfying different quality criteria can be obtained, *e.g.*, deep cuts, facet-defining cuts or Pareto-optimal cuts. Moreover, we investigate in detail the geometrical ideas and relations behind these normalizations.

(3) We show that linear normalizations are closely related to the identification of core points in the epigraphs $\text{epi}(Q_n)$, which can be challenging for multistage stochastic problems. Therefore, we propose four heuristic approaches for the computation of core point candidates.

(4) We perform extensive computational tests for SDDiP incorporating the new cut generation framework on a capacitated lot-sizing problem from the literature. We

show that the obtained lower bounds in SDDiP are majorly improved using cuts from our proposed generation framework compared to classical Lagrangian cuts or Benders cuts. We also observe that this does not necessarily guarantee an improvement of the in-sample performance of the obtained policies, though.

1.3 Structure

This paper is structured as follows. In Sect. 2 we formally introduce MS-MILPs and our notation. In Sect. 3 we introduce the new cut generation framework for Lagrangian cuts. We then present different types of Lagrangian cuts that can be obtained by using special normalizations. In Sect. 4, we present computational experiments for SDDiP incorporating these cuts for two MS-MILPs from the literature. We finish with a conclusion in Sect. 5. For reasons of space, the proofs of our theoretical results are shifted to the appendix.

2 Problem Formulation

We start by introducing MS-MILPs and their decomposition formally, mostly following the notation from [33]. We consider MS-MILPS with a finite number $T \in \mathbb{N}$ of stages, where some of the problem data is uncertain and evolves according to a known stochastic process $\xi := (\xi_1, \dots, \xi_T)$ with deterministic ξ_1. We assume that the random data vectors $\xi_t, t = 1, \dots, T$, are discrete and finite, such that the uncertainty can be modeled by a finite scenario tree. Let $\mathcal{N}$ denote the set of nodes of this tree. For each node $n \in \mathcal{N}$, the unique ancestor node is denoted by $a(n)$ and the set of child nodes is denoted by $\mathcal{C}(n)$. The probability for some node n is $p_n > 0$ and assumed to be known. The transition probabilities between adjacent nodes $n, m \in \mathcal{N}$ can then be determined as $p_{nm} := \frac{p_m}{p_n}$. For the root node r, we assume $a(r) = \emptyset$ and $p_r = 1$. We define $\overline{\mathcal{N}} := \mathcal{N} \setminus \{r\}$ to address the set of nodes without the root node, $\widetilde{\mathcal{N}}$ to address the set of nodes without leaf nodes and denote by $\mathcal{N}(t)$ the nodes at stage t.

For each node $n \in \mathcal{N}$, we distinguish state variables $x_n \in \mathbb{R}^{d_n}$, which also appear in child nodes of n, and local variables $y_n \in \mathbb{R}^{\tilde{d}_n}$. $f_n(\cdot)$ denotes the objective function of node n and $\mathcal{F}_n(x_{a(n)})$ denotes the feasible set of node n, which depends on the state variable $x_{a(n)}$ from the ancestor node. We assume that $f_n(\cdot)$ is a linear function in x_n and y_n, and that $\mathcal{F}_n(\cdot)$ is a mixed-integer polyhedral set for all $x_{a(n)}$. More precisely, we assume it to be defined by

$$\begin{aligned}\mathcal{F}_n(x_{a(n)}) := \Big\{ (x_n, y_n) \in \mathbb{R}^{d_n} \times \mathbb{R}^{\tilde{d}_n} \ : x_n \in X_n,\ y_n \in Y_n, \\ A_n x_{a(n)} + B_n x_n + C_n y_n \geq b_n \Big\}.\end{aligned} \tag{1}$$

Here, A_n, B_n, C_n, b_n denote appropriately defined data matrices and vectors. The sets X_n and Y_n are intersections of polyhedral sets $\bar{X}_n, \bar{Y}_n$ and possible integrality constraints. In the following, we also refer to X_n as the *state space*.

An MS-MILP can then be expressed by its dynamic programming equations. For the root node, we obtain

$$v^* := \min_{x_r, y_r} \Big\{ f_r(x_r, y_r) + \sum_{m \in \mathcal{C}(r)} p_{rm} Q_m(x_r) \ : \ (x_r, y_r) \in \mathcal{F}_r(x_{a(r)}) \Big\} \tag{2}$$

with $x_{a(r)} = 0$, and v^* is the optimal value of the original problem. Let $\overline{\mathbb{R}} := \mathbb{R} \cup \{+\infty\}$. For all $n \in \overline{\mathcal{N}}$, the value function $Q_n : \mathbb{R}^{d_{a(n)}} \to \overline{\mathbb{R}}$ is defined by

$$Q_n(x_{a(n)}) := \min_{x_n, y_n} \Big\{ f_n(x_n, y_n) + \sum_{m \in \mathcal{C}(n)} p_{nm} Q_m(x_n) \ : \ (x_n, y_n) \in \mathcal{F}_n(x_{a(n)}) \Big\}.$$

For the leaf nodes $n \in \mathcal{N} \setminus \widetilde{\mathcal{N}}$, we set $\sum_{m \in \mathcal{C}(n)} p_{nm} Q_m(x_n) \equiv 0$. Moreover, we set $Q_n(x_{a(n)}) = +\infty$ if $\mathcal{F}_n(x_{a(n)}) = \emptyset$, and denote by $\mathrm{dom}(Q_n)$ the *effective domain* of $Q_n(\cdot)$.

Remark 2.1. *Note that regarding $Q_n(\cdot)$ as a function on $\mathbb{R}^{d_{a(n)}}$ is not standard in stochastic programming. Often it is (implicitly) assumed to be defined only on the domain $X_{a(n)}$. However, from our view, allowing $Q_n(\cdot)$ to be defined on $\mathbb{R}^{d_{a(n)}}$ with extended real values appears more suitable for the following steps.*

As this proves beneficial in the cut generation process, we introduce local variables z_n and accompany them with copy constraints $x_{a(n)} = z_n$ and constraints $z_n \in Z_{a(n)}$, with $Z_{a(n)} \supseteq X_{a(n)}$. The most natural choice is $Z_{a(n)} = X_{a(n)}$, but also other choices are possible, *e.g.*, $Z_{a(n)} = \mathrm{conv}(X_{a(n)})$. For more details we refer to [19]. This reformulation yields the equivalent subproblems

$$\begin{aligned} Q_n(x_{a(n)}) = \min_{x_n, y_n, z_n} \Big\{ f_n(x_n, y_n) + \sum_{m \in \mathcal{C}(n)} p_{nm} Q_m(x_n) \ : (z_n, x_n, y_n) \in \mathcal{F}_n, \\ z_n = x_{a(n)}, \ z_n \in Z_{a(n)} \Big\}, \end{aligned} \tag{3}$$

where, $\mathcal{F}_n := \Big\{ (x_n, y_n, z_n) \in \mathbb{R}^{d_{a(n)}} \times \mathbb{R}^{d_n} \times \mathbb{R}^{\tilde{d}_n} \ : \ (x_n, y_n) \in \mathcal{F}_n(z_n) \Big\}$.

For the remainder of this article, we make some basic assumptions.

Assumption 1. *The following conditions are satisfied by* (1)-(3)*:*

(A1) For all $n \in \mathcal{N}$, the sets X_n and Y_n are compact.

(A2) For all $n \in \mathcal{N}$, all coefficients in A_n, B_n, C_n, b_n, f_n, $\bar{X}_n$ and $\bar{Y}_n$ are rational.

(A3) For all $n \in \overline{\mathcal{N}}$, $Z_{a(n)}$ is compact, rational MILP-representable and $Z_{a(n)} = \mathrm{dom}(Q_n)$.

Note that (A1) immediately implies that $F_n(x_{a(n)})$ is bounded for all $x_{a(n)} \in \mathbb{R}^{d_{a(n)}}$ and $n \in \mathcal{N}$. Property (A3) implies *relatively complete recourse*, a standard assumption in stochastic programming.

We have the following standard properties for the value functions.

Lemma 2.2. *For all $n \in \overline{\mathcal{N}}$, the value functions $Q_n(\cdot)$ are proper, l.sc. (lower semi-continuous) and piecewise polyhedral with finitely many pieces.*

By applying the properness reasoning to the root node, we conclude that v^* is finite.

3 A New Cut Generation Framework

In this section, we present a novel framework for the generation of Lagrangian cuts for MS-MILPs, which serves as an alternative to the classical Lagrangian cut generation framework from SDDiP [33], which we state in Appendix A for comparison. The new framework is based on ideas from [17], and in one specific form has been applied to two-stage stochastic programs in [11].

3.1 An Epigraph Perspective on Cut Generation

Recall the definition of the epigraph of the value functions $Q_n(\cdot), n \in \overline{\mathcal{N}}$:

$$\operatorname{epi}(Q_n) = \Big\{(x_{a(n)}, \theta_n) \in \mathbb{R}^{d_{a(n)}} \times \mathbb{R} \ : \ \theta_n \geq Q_n(x_{a(n)}),\ x_{a(n)} \in \operatorname{dom}(Q_n)\Big\}. \tag{4}$$

We can use this definition to reformulate the subproblems (3) to

$$\begin{aligned} Q_n(x_{a(n)}) = \min_{x_n, y_n, z_n, (\theta_m)} \Big\{ f_n(x_n, y_n) + \sum_{m \in \mathcal{C}(n)} p_{nm}\theta_m \ : (z_n, x_n, y_n) \in \mathcal{F}_n, \\ z_n = x_{a(n)},\ z_n \in Z_{a(n)}, \\ (x_n, \theta_m) \in \operatorname{epi}(Q_m),\ m \in \mathcal{C}(n) \Big\}. \end{aligned} \tag{5}$$

Here, and in the following, we use (θ_m) as a shortened notation for $(\theta_m)_{m \in \mathcal{C}(n)}$.

Remark 3.1. *The condition $x_n \in \operatorname{dom}(Q_m)$ from the definition in* (4) *is always satisfied implicitly in problem* (5)*, since* $\operatorname{dom}(Q_m) = Z_n$ *by Assumption 1 and $x_n \in X_n \subseteq Z_n$ by construction.*

In classical Benders-like decomposition methods, iteratively *optimality cuts* are constructed to approximate $Q_n(\cdot)$ and, if required, *feasibility cuts* are constructed to approximate $\operatorname{dom}(Q_n)$. This is done by solving distinct cut generation problems. However, from (4) it is evident that both types of cuts actually approximate $\operatorname{epi}(Q_n)$. Therefore, we may as well consider a unified cut generation problem to obtain polyhedral approximations Ψ_m of the sets $\operatorname{epi}(Q_m)$ [17]. Whereas we solely focus on optimality cuts in this paper, as the need of feasibility cuts is ruled out by Assumption 1 (A3), the resulting cut generation framework still proves itself valuable, as we shall see.

Remark 3.2. *In multistage stochastic programming the cuts are often aggregated to obtain cuts for the expected value functions $\sum_{m \in \mathcal{C}(n)} p_{nm} Q_m(x_n)$ (*single-cut *approach), as this reduces the total number of cuts in the subproblems. In this paper, instead, we consider a separate set of cuts,* i.e.*, separate approximations Ψ_m, for each* $\operatorname{epi}(Q_m)$ *(*multi-cut *approach). This approach is required in the new cut generation framework.*

As the value functions $Q_n(\cdot)$ are not known explicitly, for the cut generation process we replace each occurrence of $\operatorname{epi}(Q_m)$ in (5) with its current approximation Ψ_m^{i+1} (with iteration index i). We refer to the associated value functions $\underline{Q}_m^{i+1}(\cdot)$ as *approximate value functions*. This way, we actually generate cuts approximating $\operatorname{epi}(\underline{Q}_m^{i+1})$. However, by construction these cuts do also yield outer approximations of $\operatorname{epi}(Q_m)$. To avoid unboundedness, each set is initialized with a valid outer approximation $\Psi_m^0, m \in \mathcal{C}(n)$.

For notational simplicity, for the remainder of this paper we define the set

$$\mathcal{W}_n^{i+1} := \Big\{ \big(x_n, y_n, z_n, (\theta_m)\big) \ : \ (x_n, y_n, z_n) \in \mathcal{F}_n,\ z_n \in Z_{a(n)},\ (x_n, \theta_m) \in \Psi_m^{i+1}, m \in \mathcal{C}(n) \Big\},$$

and further define $\lambda_n := \big(x_n, y_n, (\theta_m)\big)$ and $c_n^\top \lambda_n := f_n(x_n, y_n) + \sum_{m \in \mathcal{C}(n)} p_{nm}\theta_m$ (recall that $f(\cdot)$ is linear). Then, the approximate value function associated with problem (5) can be compactly written as

$$\underline{Q}_n^{i+1}(x_{a(n)}^i) = \min_{\lambda_n, z_n} \Big\{ c_n^\top \lambda_n \ : \ (\lambda_n, z_n) \in \mathcal{W}_n^{i+1},\ z_n = x_{a(n)}^i \Big\}. \tag{6}$$

We make another assumption for the remainder of this paper. As for Assumption 1, we assume it to always be satisfied, even if not explicitly stated in our results.

Assumption 2. *For all $n \in \overline{\mathcal{N}}$ and all iterations i, all linear cuts defining the polyhedral set Ψ_m^{i+1} are defined by rational coefficients.*

Furthermore, in the next sections, we often require the convex hull $\operatorname{conv}(\mathcal{W}_n^{i+1})$ of a set $\mathcal{W}_n^{i+1}$. It has the following important property.

Remark 3.3. *The set $\operatorname{conv}(\mathcal{W}_n^{i+1})$ is a closed convex polyhedron. That means that there exist matrices $\tilde{A}_n, \tilde{B}_n$ and a vector $\tilde{d}_n$ such that*

$$\operatorname{conv}(\mathcal{W}_n^{i+1}) = \Big\{(\lambda_n, z_n) \;:\; \tilde{A}_n\lambda_n + \tilde{B}_n z_n \geq \tilde{d}_n\Big\}\,.$$

3.2 A Feasibility Problem for the Epigraph

We can now start to address the actual cut generation process in the proposed unified framework. Given a point $(x_{a(n)}^i, \theta_n^i)$, we consider the feasibility problem

$$v_n^{f,i+1}(x_{a(n)}^i, \theta_n^i) := \min_{\lambda_n, z_n} \Big\{0 \;:\; (\lambda_n, z_n) \in \mathcal{W}_n^{i+1},\; z_n = x_{a(n)}^i,\; \theta_n^i \geq c_n^\top \lambda_n\Big\}, \tag{7}$$

which can be shown to verify if $(x_{a(n)}^i, \theta_n^i) \in \operatorname{epi}(\underline{Q}_n^{i+1})$, see Appendix B.1 for a proof.

Lemma 3.4. *Given a point $(x_{a(n)}^i, \theta_n^i)$, problem (7) is a feasibility problem for $\operatorname{epi}(\underline{Q}_n^{i+1})$, that is,*

$$v_n^{f,i+1}(x_{a(n)}^i, \theta_n^i) = \begin{cases} 0, & \textit{if } (x_{a(n)}^i, \theta_n^i) \in \operatorname{epi}(\underline{Q}_n^{i+1}) \\ +\infty, & \textit{else.} \end{cases}$$

3.3 Lagrangian Cuts in the New Framework

To generate Lagrangian cuts, we apply a Lagrangian relaxation to problem (7). A key difference to the classical Lagrangian relaxation from SDDiP (see Appendix A) is that not only $x_{a(n)}^i$, but $(x_{a(n)}^i, \theta_n^i)$ is regarded as a fixed incumbent for the cut generation process. Therefore, we relax all constraints containing either $x_{a(n)}^i$ or θ_n^i. For given multipliers $(\pi_n, \pi_{n0}) \in \mathbb{R}^{d_{a(n)}} \times \mathbb{R}^+$ the dual function is then given by

$$\mathscr{L}_n^{i+1}(\pi_n, \pi_{n0}) := \min_{\lambda_n, z_n} \Big\{\pi_n^\top z_n + \pi_{n0} c_n^\top \lambda_n \;:\; (\lambda_n, z_n) \in \mathcal{W}_n^{i+1}\Big\}. \tag{8}$$

The corresponding Lagrangian dual problem is

$$\max_{\pi_n, \pi_{n0}} \Big\{\mathscr{L}_n^{i+1}(\pi_n, \pi_{n0}) - \pi_n^\top x_{a(n)}^i - \pi_{n0}\theta_n^i \;:\; \pi_{n0} \geq 0\Big\}. \tag{9}$$

We state some important properties of this dual problem, with a proof provided in Appendix B.2.

Theorem 3.5. *For the Lagrangian dual (9) it holds:*

(i) The dual function $\mathscr{L}_n^{i+1}(\cdot)$ is piecewise linear concave in (π_n, π_{n0}).

(ii) Its optimal value $\widehat{v}_n^{D,i+1}(x_{a(n)}^i, \theta_n^i)$ *satisfies*

$$\widehat{v}_n^{D,i+1}(x_{a(n)}^i, \theta_n^i) = \begin{cases} 0, & \text{if } (x_{a(n)}^i, \theta_n^i) \in \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})) \\ +\infty, & \text{else.} \end{cases}$$

Theorem 3.5 implies that the Lagrangian dual is unbounded whenever $(x_{a(n)}^i, \theta_n^i) \notin \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$. Therefore, there exists an unbounded ray (π_n^i, π_{n0}^i) such that

$$\mathscr{L}_n^{i+1}(\pi_n^i, \pi_{n0}^i) - (\pi_n^i)^\top x_{a(n)}^i - \pi_{n0}^i \theta_n^i > 0,$$

and $(x_{a(n)}^i, \theta_n^i)$ violates the following Lagrangian cut:

Definition 3.6. *For all* $n \in \overline{\mathcal{N}}$ *and some multipliers* (π_n^i, π_{n0}^i)*, a Lagrangian cut is given by*

$$\pi_{n0}^i \theta_n + (\pi_n^i)^\top x_{a(n)} \geq \mathscr{L}_n^{i+1}(\pi_n^i, \pi_{n0}^i). \tag{10}$$

This cut is valid for any feasible (π_n^i, π_{n0}^i) in (9), as proven in Appendix B.3.

Lemma 3.7. *For any* $(\pi_n^i, \pi_{n0}^i) \in \mathbb{R}^{d_{a(n)}} \times \mathbb{R}^+$ *the Lagrangian cut* (10) *is satisfied by all* $(x_{a(n)}, \theta_n) \in \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$*, and thus by all* $(x_{a(n)}, \theta_n) \in \operatorname{epi}(Q_n)$*.*

We analyze the relation between the Lagrangian cuts (10) and the classical ones (19). We restrict to $\pi_{n0} > 0$ because $\pi_{n0} = 0$ leads to feasibility cuts that by assumption are not required in our case.

Remark 3.8. *Let* $\pi_{n0}^i > 0$ *in cut* (10)*. As shown in Proposition 1 in [11], with division by* π_{n0}^i*, it follows*

$$\begin{aligned} \theta_n &\geq \frac{1}{\pi_{n0}^i} \mathscr{L}_n^{i+1}(\pi_n^i, \pi_{n0}^i) - \left(\frac{\pi_n^i}{\pi_{n0}^i}\right)^\top x_{a(n)} \\ &= \mathscr{L}_n^{i+1}(\widehat{\pi}_n, 1) - \widehat{\pi}_n^\top x_{a(n)} \\ &= \mathcal{L}_n^{i+1}(\widehat{\pi}_n) - \widehat{\pi}_n^\top x_{a(n)} \end{aligned}$$

with $\widehat{\pi}_n := \frac{\pi_n^i}{\pi_{n0}^i}$*. This is an equivalent representation of* (10) *in the form of a classical Lagrangian optimality cut* (19)*, see Appendix A.*

We should emphasize that despite the scaling relation shown in Remark 3.8, the new cut generation framework may yield different cuts than the classical one because the choice of dual multipliers is based on a different dual problem.

3.4 Cut Selection Criteria

It is not immediately clear how to select cut coefficients (π_n^i, π_{n0}^i) from the unbounded Lagrangian dual (9) in the most reasonable way. On the one hand, we aspire to determine coefficients (π_n^i, π_{n0}^i) by solving a *bounded* and feasible optimization problem. On the other hand, we want to make sure that the obtained cuts are not only separating the incumbent $(x_{a(n)}^i, \theta_n^i)$ from $\operatorname{epi}(Q_n)$, but also of good approximation quality.

The first aim can be achieved by bounding problem (9) artificially, *e.g.*, by introducing bounds on the dual multipliers or introducing a normalizing constraint. Another

Table 1: Examination of cut quality criteria in the literature on BD and disjunctive programming.

Paper	MIS	Max. depth	Facet-def.	Pareto-opt.
Magnanti and Wong [23]				✓
Cornuéjols and Lemaréchal [13]		✓	✓	
Papadakos [24]				✓
Cadoux [10]		✓	✓	
Fischetti et al. [17]	✓			
Sherali and Lunday [29]				✓
Conforti and Wolsey [12]			✓	
Brandenberg and Stursberg [9]	✓		✓	✓
Hosseini and Turner [22]		✓		
Seo et al. [27]			✓	
This paper		✓	✓	✓

MIS: Cuts that correspond to minimal infeasible subsytems of the feasibility subproblem.

common approach is to fix its unbounded objective to 1. Combined with Remark 3.3 this allows to identify unbounded rays by analyzing a compact polyhedron [17].

With regard to the second aim, various quality criteria for cutting-planes have been put forward in the literature [14]. Many of these criteria have been applied in the context of BD or disjunctive programming before, as shown in Table 1, but our paper is the first one applying them to Lagrangian cuts, and incorporating most of them at once. We focus on three important criteria:

- *Facet-defining cuts.* These cuts reproduce facets of a convex polyhedral set, in our case $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}_n^{i+1}))$, and thus may be helpful in ensuring finite convergence.
- *Pareto-optimal cuts* [23]. For $\pi_{n0} > 0$, these cuts (10) are *non-dominated* in the sense that there exists no other cut $\tilde{\pi}_{n0}\theta_n + (\tilde{\pi}_n)^\top x_{a(n)} \geq \mathscr{L}_n^{i+1}(\tilde{\pi}_n, \tilde{\pi}_{n0})$ such that
$$\frac{1}{\tilde{\pi}_{n0}}\Big(\mathscr{L}_n^{i+1}(\tilde{\pi}_n, \tilde{\pi}_{n0}) - (\tilde{\pi}_n)^\top x_{a(n)}\Big) \geq \frac{1}{\tilde{\pi}_{n0}}\Big(\mathscr{L}_n^{i+1}(\pi_n^i, \pi_{n0}^i) - (\pi_n^i)^\top x_{a(n)}\Big)$$
for all $(x_{a(n)}, \theta_n) \in \mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}_n^{i+1}))$. Importantly, Pareto-optimality with respect to $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}_n^{i+1}))$ does not necessarily imply Pareto-optimality with respect to $\mathrm{epi}(\underline{Q}_n^{i+1})$, but is easier to achieve.
- *Deep cuts* [8]. These cuts are *deep* in the sense that a maximum distance between the incumbent $(x_{a(n)}^i, \theta_n^i)$ and the separating hyperplane is realized, *i.e.*, they cut as deep as possible into the suboptimal region.

As shown in the literature, especially in [9], many of these criteria can be satisfied by optimizing over the so-called *reverse polar set* of $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}_n^{i+1}))$ shifted by the incumbent $(x_{a(n)}^i, \theta_n^i)$. The reverse polar set is an important tool in the theory on cut generation, as it is directly linked to the support function of $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}_n^{i+1}))$, and thus provides a characterization of normal vectors of $(x_{a(n)}^i, \theta_n^i)$-separating hyperplanes [9, 13].

The reverse polar set was first introduced by [3] and can be defined as follows.

Definition 3.9. *The reverse polar set of a set $S \subset \mathbb{R}^n$ is defined as*

$$S^- := \Big\{ d \in \mathbb{R}^n \ : \ d^\top x \leq -1 \quad \forall x \in S \Big\}.$$

To simplify notation, we set $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i) := \big(\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})) - (x_{a(n)}^i, \theta_n^i) \big)^-$ for the reverse polar set of $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ shifted by $(x_{a(n)}^i, \theta_n^i)$. Using Remark 3.3, it can be reformulated.

Lemma 3.10. *The reverse polar set $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i)$ can be expressed as*

$$\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i) = \left\{ (\gamma_n, \gamma_{n0}) \in \mathbb{R}^{d_{a(n)}} \times \mathbb{R} \ : \ \exists \mu_n \geq 0 \ : \ \begin{array}{l} \gamma_{n0} \leq 0 \\ -\tilde{A}_n^\top \mu_n - \gamma_{n0} c_n = 0 \\ -\tilde{B}_n^\top \mu_n - \gamma_n = 0 \\ \tilde{d}_n^\top \mu_n + \gamma_n^\top x_{a(n)}^i + \gamma_{n0} \theta_n^i \geq 1 \end{array} \right\}.$$

We provide a proof in Appendix B.4.

Remark 3.11. *Even with the above reformulation of $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i)$, an explicit formulation is usually not readily available due to the existence quantor and due to $\tilde{A}_n$, $\tilde{B}_n$ and $\tilde{d}_n$ not being known.*

Based on the existing work for BD, in the next sections, we present and investigate different strategies to generate Lagrangian cuts satisfying the above quality criteria. In the light of Remark 3.11, $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i)$ may not be used without further ado to generate such cuts computationally. Still, it proves useful in the derivation of Lagrangian cuts with favorable properties. In particular, as we shall see, optimizing over $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i)$ is closely linked to solving *normalized* Lagrangian dual problems. So in fact, our two perspectives to approach cut selection are intertwined and boil down to considering specific (bounded) normalizations of problem (9).

We first define the normalized Lagrangian dual in a general form and state some important properties.

Definition 3.12. *For some homogeneous normalization function $g_n : \mathbb{R}^{d_n} \times \mathbb{R}^+ \to \mathbb{R}$, the normalized Lagrangian dual is defined as*

$$\widehat{v}_n^{ND,i+1}(x_{a(n)}^i, \theta_n^i) := \max_{\pi_n, \pi_{n0}} \Big\{ \mathscr{L}_n^{i+1}(\pi_n, \pi_{n0}) - \pi_n^\top x_{a(n)}^i - \pi_{n0} \theta_n^i \ : \\ g_n(\pi_n, \pi_{n0}) \leq 1, \pi_{n0} \geq 0 \Big\}. \tag{11}$$

Remark 3.13. *As long as the normalization constraint $g_n(\pi_n, \pi_{n0}) \leq 1$ is satisfied by some neighborhood N of the origin, we do not exclude any potential cuts due to the scaling property of π_{n0}, see Remark 3.8 and [11].*

Lemma 3.14. *If $(x_{a(n)}^i, \theta_n^i) \in \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$, then $\widehat{v}_n^{ND,i+1}(x_{a(n)}^i, \theta_n^i) = 0$, and vice versa.*

We provide a proof in Appendix B.5. Lemma 3.14 allows us to solely focus on the case where $(x_{a(n)}^i, \theta_n^i) \notin \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ for the remainder of this section.

In the next two subsections, we consider two different types of normalization: by norm constraints and by linear constraints. As we show, both types of normalization can be viewed from three different perspectives (a primal perspective, a projection

Table 2: Examination of normalized cut generation problems and different perspectives on it in the literature.

Paper	Norm normalization			Linear normalization		
	Prim	Proj	RP	Prim	Proj	RP
Cornuéjols and Lemaréchal [13]	✓	✓		✓	✓	✓
Cadoux [10]	✓	✓	✓			
Fischetti et al. [17]				(✓)		
Brandenberg and Stursberg [9]				✓		✓
Hosseini and Turner [22]	✓	✓		✓		
Chen and Luedtke [11]	(✓*)					
Seo et al. [27]					✓	
This paper	✓*	✓*	✓*	✓*	✓*	✓*

Prim: Primal perspective. Proj: Projection perspective. RP: Reverse polar perspective.
(✓): Perspective is applied, but not further explored. ✓*: Examination for Lagrangian cuts.

perspective and a reverse polar perspective). These perspectives have been analyzed in the literature before, as shown in Table 2, but have not been linked all together in a generalized framework and have not been applied to Lagrangian cuts.

3.5 Normalization by Norm and Deep Cuts

We consider the normalized Lagrangian dual (11) and the associated Lagrangian cuts if some norm is used as the normalization function.

Definition 3.15. *Let $\|\cdot\|$ be some arbitrary norm. The Lagrangian cut* (10) *defined by the solution (π_n, π_{n0}) to the normalized Lagrangian dual* (11) *with $g_n(\pi_n, \pi_{n0}) = \|\pi_n, \pi_{n0}\|$ is called* $\|\cdot\|$-deep Lagrangian cut*. For ℓ^p-norms we may also use the term* ℓ^p-deep Lagrangian cuts*.*

If appropriate norms are used, *e.g.*, ℓ^1 or ℓ^∞, then the normalization can be expressed by linear constraints. For the special choice of the ℓ^1-norm, it is used by [11] to generate Lagrangian cuts in two-stage stochastic programs, however without discussing the conceptual idea behind it in detail.

Deep cuts allow for three theoretical and geometrical interpretations (cf. Table 2), which also explain why they are called *deep*. As the existing results from the literature can be applied to the multistage and Lagrangian setting in a straightforward way, for reasons of space we do not provide proofs here.

(1) **Maximizing cut depth.** Deep cuts maximize the *depth* or scaled violation among all potential cuts, *i.e.*, the distance between the incumbent $(x^i_{a(n)}, \theta^i_n)$ and the hyperplane associated with a potential cut in the dual norm $\|\cdot\|_*$ of $\|\cdot\|$. This means that they cut as deep as possible into the suboptimal region.

Lemma 3.16 (based on [22])**.** *Let $\|\cdot\|$ be some norm and $\|\cdot\|_*$ its dual norm. Further, let $d_n\big((x^i_{a(n)}, \theta^i_n); (\pi_n, \pi_{n0})\big)$ denote the distance between the hyperplane defined by (π_n, π_{n0}) and the point $(x^i_{a(n)}, \theta^i_n) \notin \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}^{i+1}_n))$ measured in $\|\cdot\|_*$. Then, the optimal value $\widehat{v}^{ND,i+1}_n(x^i_{a(n)}, \theta^i_n)$ of problem* (11) *with $g_n(\pi_n, \pi_{n0}) = \|\pi_n, \pi_{n0}\|$*

equals

$$\max_{\pi_n,\pi_{n0}} \Big\{ d_n\big((x^i_{a(n)},\theta^i_n);(\pi_n,\pi_{n0})\big) \ : \ \pi_{n0} \geq 0 \Big\}.$$

(2) **Projection onto the epigraph.** From a primal perspective, generating $\|\cdot\|$-deep cuts is in some sense equivalent to minimizing the distance in $\|\cdot\|_*$ between the incumbent $(x^i_{a(n)},\theta^i_n)$ and the epigraph $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$, *i.e.*, related to projecting $(x^i_{a(n)},\theta^i_n)$ onto the epigraph.

Lemma 3.17 (based on Lemma 2.5 in [10])**.** *Let $\|\cdot\|$ be some norm and $\|\cdot\|_*$ its dual norm. Then, the optimal value $\widehat{v}^{ND,i+1}_n(x^i_{a(n)},\theta^i_n)$ of problem (11) with $g_n(\pi_n,\pi_{n0}) = \|\pi_n,\pi_{n0}\|$ equals that of the projection problem*

$$\min_{x_{a(n)},\theta_n} \Big\{ \|x_{a(n)} - x^i_{a(n)}, \theta_n - \theta^i_n\|_* \ : \ (x_{a(n)},\theta_n) \in \mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n)) \Big\}. \tag{12}$$

Lemma 3.17 implies that $\widehat{v}^{ND,i+1}_n(x^i_{a(n)},\theta^i_n) > 0$ if $(x^i_{a(n)},\theta^i_n) \notin \mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$, whereas $\widehat{v}^{ND,i+1}_n(x^i_{a(n)},\theta^i_n) = 0$ if not, so it confirms Lemma 3.14. Therefore, as for the non-normalized case, we have a unique flag for cases where no separating cut has to be constructed. However, in contrast to the non-normalized case, the dual problem is bounded.

We can also conclude from Lemma 3.17 that a deep cut supports $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$.

Corollary 3.18 (based on Proposition 3 in [22])**.** *Suppose $(x^i_{a(n)},\theta^i_n) \notin \mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$ and let $(\widehat{x}_{a(n)},\widehat{\theta}_{a(n)}) \in \mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$ be a solution to (12). Then, any $\|\cdot\|$-deep cut separating $(x^i_{a(n)},\theta^i_n)$ from $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$ supports $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$ at $(\widehat{x}_{a(n)},\widehat{\theta}_{a(n)})$.*

(3) **Minimizing a norm over the reverse polar set.** Interestingly, deep cuts allow for another geometric interpretation that is related to the reverse polar set $\mathcal{R}^{i+1}_n(x^i_{a(n)},\theta^i_n)$. It is based on the observation that $\mathcal{R}^{i+1}_n(x^i_{a(n)},\theta^i_n)$ is directly linked to the normals of separating hyperplanes.

Lemma 3.19 (Lemma 2.9 in [10])**.** *Let (π^i_n,π^i_{n0}) be the coefficients of a $\|\cdot\|$-deep cut constructed at $(x^i_{a(n)},\theta^i_n) \notin \mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}^{i+1}_n))$. Then there exists some $\alpha > 0$ such that $-\alpha(\pi^i_n,\pi^i_{n0})$ minimizes $\|\cdot\|$ over the reverse polar set $\mathcal{R}^{i+1}_n(x^i_{a(n)},\theta^i_n)$.*

The main idea to prove this result is that due to positive homogeneity, the norm $\|\cdot\|$ in the normalization constraint and the dual objective can be swapped.

To illustrate these three perspectives for different norms we provide an illustrative example in Appendix C.1. It highlights that, whereas deep cuts can be unique, also degenerate solutions are possible if the optimization over $\mathcal{R}^{i+1}_n(x^i_{a(n)},\theta^i_n)$ in Lemma 3.19 does not have a unique solution. This degeneracy of the dual (11) may lead to selection of non-dominant cuts, compare Sect. 1.1, and is only excluded for the ℓ^2-norm. Further, even if unique, deep cuts are not guaranteed to be facet-defining. In fact, analyses in Cadoux [10] and Hosseini and Turner [22] show that they tend to be flat, at least when the optimality gap is still large. Finally, also the projection problem (12) is not guaranteed to have a unique solution for all but the ℓ^2-norm (see Fig. 10). However, if the solution is non-unique, the associated cut is unique and facet-defining.

3.6 Linear Normalization

We consider the normalized Lagrangian dual (11) with a linear normalization function $g_n(\pi_n, \pi_{n0}) = u_n^\top \pi_n + u_{n0}\pi_{n0}$ defined by some coefficients $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \in \mathbb{R}$. Recall that the initial Lagrangian dual problem (9) is unbounded and that we introduce the normalization constraint in (11) in order to transform the problem to a bounded one to identify unbounded rays. In contrast to the norm-based normalization from Sect. 3.5, a linear normalization does not guarantee boundedness, though. Hence, the choice of (u_n, u_{n0}) is crucial to ensure that an optimal solution exists. We further analyze this later in this section, but for now take the following assumption.

Assumption 3. *Given some $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$, the normalized Lagrangian dual* (11) *with $g_n(\pi_n, \pi_{n0}) = u_n^\top \pi_n + u_{n0}\pi_{n0}$ has a finite optimal value $\widehat{v}_n^{ND,i+1}(x_{a(n)}^i, \theta_n^i) > 0$.*

3.6.1 Linear Normalization Cuts

We can then define the associated type of Lagrangian cuts.

Definition 3.20. *Let $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$ and let the normalized Lagrangian dual* (11) *satisfy Assumption 3. Then, we refer to the Lagrangian cut* (10) *defined by its solution (π_n, π_{n0}) as a* linear normalization (LN) Lagrangian cut*.*

Again, we can take three different perspectives on LN cuts.

(1) **Pseudonorm perspective.** Hosseini and Turner [22] restrict to choices of (u_n, u_{n0}) such that $g_n(\pi_n, \pi_{n0}) = u_n^\top \pi_n + u_{n0}\pi_{n0} \geq 0$ for all $(\pi_n, \pi_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}_+$. In such a case, $g_n(\cdot)$ is a *linear pseudonorm.* This means that LN cuts can be interpreted as maximizing the distance between the associated hyperplane and $(\widehat{x}_{a(n)}^i, \widehat{\theta}_n^i)$ in a linear pseudonorm.

(2) **Projection on a line segment.** This perspective has been brought up several times in the literature in different variants, most recently by [27]. Even though the geometric idea is the same in the Lagrangian context, the formal description changes a bit.

First, we exploit that for linear $g_n(\cdot)$, the normalized Lagrangian dual (11) can be reformulated as an LP and then be dualized with strong duality.

Lemma 3.21. *Let $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$. The normalized Lagrangian dual* (11) *with $g_n(\pi_n, \pi_{n0}) = u_n^\top \pi_n + u_{n0}\pi_{n0}$ can be formulated as an LP, and its dual is*

$$\min_{\lambda_n, z_n, \eta_n} \Big\{ \eta_n \; : \; (\lambda_n, z_n) \in \operatorname{conv}(\mathcal{W}_n^{i+1}), \quad \eta_n \geq 0, \; u_{n0}\eta_n \geq c_n^\top \lambda_n - \theta_n^i, \; u_n \eta_n = z_n - x_{a(n)}^i \Big\}. \tag{13}$$

We provide a proof in Appendix B.6. Problem (13) can be interpreted as finding the smallest scaling factor $\eta_n \geq 0$ such that starting from $(x_{a(n)}^i, \theta_n^i)$ along direction (u_n, u_{n0}) a point in $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ is reached. Again, for the optimal value it follows $\widehat{v}_n^{ND,i+1}(x_{a(n)}^i, \theta_n^i) = \eta_n^* > 0$ if and only if $(x_{a(n)}^i, \theta_n^i) \notin \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$. Given the optimal value, the projection of $(x_{a(n)}^i, \theta_n^i)$ onto $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ along direction (u_n, u_{n0}) can be determined as

$$(\widehat{x}_{a(n)}, \widehat{\theta}_{a(n)}) = (x_{a(n)}^i, \theta_n^i) + \eta_n^*(u_n, u_{n0}). \tag{14}$$

We then obtain the following result:

Corollary 3.22. *Let* $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$ *and let the normalized Lagrangian dual* (11) *satisfy Assumption 3. Furthermore, let* $(\widehat{x}_{a(n)}, \widehat{\theta}_{a(n)}) \in \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ *satisfy* (14)*. Then, the associated LN cut supports* $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ *at* $(\widehat{x}_{a(n)}, \widehat{\theta}_{a(n)})$.

(3) **Maximizing a linear function over the reverse polar set.** Again, an interpretation with respect to $\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n)$ is possible. More precisely, LN cuts can be obtained by maximizing the linear objective function $u_n^\top \gamma_n + u_{n0}\gamma_{n0}$ over $\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n)$:

$$\max_{\gamma_n, \gamma_{n0}} \Big\{ u_n^\top \gamma_n + u_{n0}\gamma_{n0} \;:\; (\gamma_n, \gamma_{n0}) \in \mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n) \Big\}. \tag{15}$$

This perspective is discussed in detail in [9] for BD and in [13] for general convex sets. For the relation to the normalized Lagrangian dual (11), the following result holds. For a sketch of the proof, see Appendix B.7.

Theorem 3.23 (based on [9])**.** *Let* $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$ *and* $(x^i_{a(n)}, \theta^i_n) \notin \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$. *Then the following results hold:*

(i) If problem (11) *satisfies Assumption 3, then problem* (15) *has a finite optimal value, and vice versa.*

(ii) The induced cuts for $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ *are equivalent for both problems.*

Geometrically, a favorable property of generating cuts based on $\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n)$ is that (given that $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ is a full-dimensional polyhedron) each vertex (γ_n, γ_{n0}) of $\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n)$ corresponds to the normal vector of a facet of $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$, and vice versa [9]. In order to identify such points, we can use the LP (15), given a choice of $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$ such that a finite optimum is attained.

In fact, perspectives (2) and (3) make a sufficient condition for such a finite optimum readily available, and by that also allow us to conclude when Assumption 3 is satisfied. Whereas this result is known from convex analysis [9, 13], in Appendix B.8, we provide an alternative proof based on perspective (2) and the Lagrangian setting.

Lemma 3.24. *Assumption 3 is satisfied if*

$$(u_n, u_{n0}) \in \operatorname{cone}\big(\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})) - (x^i_{a(n)}, \theta^i_n)\big) \setminus \{0\}, \tag{16}$$

where $\operatorname{cone}(S)$ *denotes the conical hull of a set* S.

Lemma 3.24 implies that also choosing (u_n, u_{n0}) from $\operatorname{epi}(\underline{Q}_n^{i+1}) - (x^i_{a(n)}, \theta^i_n)$ or even from $\operatorname{epi}(Q_n) - (x^i_{a(n)}, \theta^i_n)$ is sufficient. In other words, choosing reasonable coefficients $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$ boils down to finding a *core point* within one of these epigraphs. We discuss this in more detail in Sect. 3.6.2.

We now state some beneficial properties of LN cuts with respect to the aforementioned cut quality criteria.

Theorem 3.25. *Let* $(u_n, u_{n0}) \in \mathbb{R}^{d_n} \times \mathbb{R}$. *Consider the normalized Lagrangian dual problem* (11) *with* $g_n(\pi_n, \pi_{n0}) = u_n^\top \pi_n + u_{n0}\pi_{n0}$. *Then,* *(i) for all* $(u_n, u_{n0}) \in \operatorname{cone}\big(\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})) - (x^i_{a(n)}, \theta^i_n)\big) \setminus \{0\}$*, the optimal point* (π_n^*, π_{n0}^*) *defines a supporting cut for* $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$*,*

(ii) for all $(u_n, u_{n0}) \in \operatorname{cone}\big(\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})) - (x^i_{a(n)}, \theta^i_n)\big) \setminus \{0\}$*, there exists an optimal extreme point* (π_n^*, π_{n0}^*)*, such that the obtained cut is facet-defining for* $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$*,*

(iii) for all $(u_n, u_{n0}) \in \operatorname{relint}\big(\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})) - (x^i_{a(n)}, \theta^i_n)\big)$*, any optimal point* (π_n^*, π_{n0}^*) *with* $\pi_{n0}^* > 0$ *defines a Pareto-optimal cut for* $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ *on* $\operatorname{conv}(Z_{a(n)})$*.*

Part (i) directly follows from Lemma 3.24 and Corollary 3.22. Part (ii) follows from Theorem 3.3 in [9], and part (iii) follows from Theorem 3.43 in [30]. Note that the results from the literature require to choose (u_n, u_{n0}) from the relative interior of the epigraph restricted to $\operatorname{conv}(X_{a(n)})$. However, under Assumption 1, if we choose $Z_n = X_{a(n)}$ or $Z_n = \operatorname{conv}(X_{a(n)})$, in our case the considered epigraphs are always restricted to this set.

Considering part (ii), the only case in which no facet-defining cut is obtained occurs if the optimal solution to the normalized Lagrangian dual (11) is not unique. However, this is only the case for a small subset of choices (u_n, u_{n0}). Especially if the choice is adapted in each iteration, the occurrences of such cases should be negligible [9]. With respect to (iii), we should emphasize again that Pareto-optimality for $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ does not necessarily imply Pareto-optimality for $\operatorname{epi}(\underline{Q}_n^{i+1})$.

We finish our theoretical results in this subsection with two remarks.

Remark 3.26. *Based on the perspective taken, LN cuts are also called* pseudo-deep cuts *[22] or* closest cuts *[27] in the literature on BD.*

Remark 3.27. *Choosing* $(u_n, u_{n0}) = (0, 1)$ *in Definition 3.20 yields the classical Lagrangian cuts presented in Appendix A.*

We illustrate the different perspectives on LN cuts again using an example in Appendix C.2.

3.6.2 Identifying Core Points

As described before, a key challenge of generating LN cuts with favorable properties is to choose (u_n, u_{n0}) appropriately, *i.e.*, according to Lemma 3.24 or Theorem 3.25. In the literature, it is often proposed to evaluate feasible points in the objective function to obtain core points $(\widehat{x}^i_{a(n)}, \widehat{\theta}^i_n)$, and by that reasonable coefficients (u_n, u_{n0}). Whereas this approach is straightforward for BD or the two-stage stochastic case [9, 30], in the multistage case, evaluating $Q_n(\cdot)$ exactly is computationally prohibitive in general. Therefore, we propose different heuristics based on using function $\underline{Q}_n^{i+1}(\cdot)$.

We consider the following approaches:

- `Mid.` We set $\widehat{x}^i_{a(n)} = \operatorname{mid}(\operatorname{conv}(X_{a(n)}))$ where mid denotes the midpoint (that is, we assume box constraints for $x_{a(n)}$). The idea is that incentivizing the LN cuts to support $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})$ in the interior of the state space may be useful to avoid the degeneracy issues discussed in Sect. 1.1. We then set $\widehat{\theta}^i_n = \underline{Q}_n^{i+1}(\widehat{x}^i_{a(n)})$, even if this does not satisfy the relative interior requirement in Theorem 3.25.

- `Eps.` For some $\varepsilon > 0$, we use an ε-perturbation of $x^i_{a(n)}$ into the interior of $\operatorname{conv}(X_{a(n)})$, and set $\widehat{\theta}^i_n = \underline{Q}_n^{i+1}(\widehat{x}^i_{a(n)})$. This idea is inspired by the perturbation strategy described in [29] and a similar strategy used in [27]. It is particularly suited to SDDiP, as it avoids the possible degeneracy issues related to generating cuts at extreme points of the state space, see Sect. 1.1. For sufficiently small ε, the LN cut should still be supporting $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}_n^{i+1}))$ at $(x_{a(n)}^i, \overline{\mathrm{co}}(\underline{Q}_n^{i+1})(x_{a(n)}^i))$.

- `Relint.` We solve an auxiliary feasibility problem with slack variables to find a potential core point in relint $\big(\mathrm{epi}(\underline{Q}_n^{i+1})\big)$. This feasibility problem is defined in a similar way to the one described in Sect. 5.1 of [12].

Despite their straightforwardness, these heuristics come with some notable challenges. Crucially, the first two ones yield candidates satisfying $\widehat{x}_{a(n)}^i \in \mathrm{conv}(X_{a(n)})$. However, this choice is not necessarily feasible if integrality constraints are present (especially if $Z_{a(n)} = X_{a(n)}$), leading to $\underline{Q}_n^{i+1}(\widehat{x}_{a(n)}^i) = +\infty$. We thus do not obtain a core point or a reasonable choice of (u_n, u_{n0}). This is clearly unintended and should be detected in practice. Whenever unboundedness is detected, we may take some special counter-measures, such as artificially bounding the normalized Lagrangian dual problem (11) or resorting to strengthened Benders cuts instead of generating LN cuts.

Problem (13) provides a natural way to check for unboundedness, or the validity of direction (u_n, u_{n0}), respectively. However, it cannot be solved immediately, as we do not know $\mathrm{conv}(\mathcal{W}_n^{i+1})$ explicitly. We may instead solve the approximation

$$\min_{\lambda_n, z_n, \eta_n} \Big\{ \eta_n \ : \ (\lambda_n, z_n) \in \mathcal{W}_n^{i+1}, \ \eta_n \geq 0, \ u_{n0}\eta_n \geq c_n^\top \lambda_n - \theta_n^i, \ u_n \eta_n = z_n - x_{a(n)}^i \Big\} \tag{17}$$

using the known set $\mathcal{W}_n^{i+1}$ instead of $\mathrm{conv}(\mathcal{W}_n^{i+1})$. If the dual problem (11) is unbounded, then this problem is infeasible. Therefore, we can use infeasibility of (17) as a proxy for unboundedness. Unfortunately, in the presence of integrality constraints, infeasibility of (17) may occur very often, even given an appropriate direction (u_n, u_{n0}), thus leading to taking counter-measures more often than required. On the other hand, using the LP relaxation of (17) is not sufficient to rule out all cases of unboundedness. For an effective implementation of LN cuts this is a significant challenge.

Finally, let us present an alternative approach to come up with core points.

- `Conv.` We note that any convex combination of two (or more) feasible points $\big(x_1, \underline{Q}_n^{i+1}(x_1)\big)$ and $\big(x_2, \underline{Q}_n^{i+1}(x_2)\big)$ is always contained in $\mathrm{epi}(\overline{\mathrm{co}}(\underline{Q}_n^{i+1})$. One such point is readily available with $\big(x_{a(n)}^i, \underline{Q}_n^{i+1}(x_{a(n)}^i)\big)$. For the case $X_{a(n)} = \{0,1\}^{d_{a(n)}}$, an intuitive strategy to obtain a second one is to consider the diagonal counterpart of $x_{a(n)}^i$ (swapping 0 and 1 for all components) and its function value for $\underline{Q}_n^{i+1}(\cdot)$. If this counterpart is feasible, we obtain a whole family of core points without the requirement of evaluating the approximate value function in non-integer states.

4 Computational Experiments

We report results for a computational study of SDDiP using the proposed cut generation framework. For a detailed description of the SDDiP algorithm, we refer to [33]. For comparison, we also run tests using established cut generation techniques in SDDiP. More precisely, we consider the following approaches to generate cuts:

- `B`: Classical Benders cuts using a single-cut or a multi-cut approach.

- `SB`: Strengthened Benders cuts [33] using a single-cut or a multi-cut approach.

- `L`: Classical Lagrangian cuts [33] (see Appendix A) using a single-cut or a multi-cut approach.
- $\ell^1, \ell^\infty, \ell^{1\infty}$: Deep Lagrangian cuts from Definition 3.15 for the ℓ^1-norm, the ℓ^∞-norm and a linear combination of $0.5\|\cdot\|_1 + 0.5\|\cdot\|_\infty$.
- `LN`: LN Lagrangian cuts from Definition 3.20 using the `Mid`, `Eps`, `Relint` and `Conv` heuristics from Sect. 3.6.2 to identify core points and to determine the normalization coefficients. For `Eps` we use a perturbation of the incumbent by 10^{-2}. For `Conv` we consider convex combination parameters of 0.5, 0.75, 0.9 and 0.99 (with higher values encoding a higher proximity to $\big(x^i_{a(n)}, \underline{Q}^{i+1}_n(x^i_{a(n)})\big)$).

By construction, all deep and LN cuts require using a multi-cut approach.

We test the proposed methods for two classes of MS-MILPs from the literature:

- `CLSP`: a capacitated lot-sizing problem described in [31], with stagewise independent uncertain demand for each product. This problem is also considered in [1] and identified to be challenging for exact decomposition methods like SDDiP. We consider instances with 3 or 10 state variables, 4, 6, 10 or 16 stages and 20 realizations of the uncertain demand at each stage.
- `CFLP`: a capacitated facility location problem with stagewise independent uncertain facility disruptions [28]. We consider an instance with 16 state variables, 100 stages and 20 realization of the uncertainty at each stage.

With respect to the state space, it is important to mention that `CLSP` contains *continuous* state variables. For instances with 3 state variables, in order to ensure exactness of SDDiP, we thus apply a binary approximation of the state space using a discretization precision of 1.0, see [33]. This means that the modified MS-MILP which is tackled by SDDiP contains 30 state variables. We refer to this problem as `CLSP-Bin`. For instances with 10 state variables, using a binary approximation would produce state dimensions which are computationally intractable for SDDiP, as its complexity grows exponentially in the dimension of the state space [32]. Therefore, in these cases, we apply SDDiP to `CLSP` without a binary approximation, and thus without theoretical guarantees to close the optimality gap. In contrast, `CFLP` only contains binary state variables from the outset, so no binary approximation is required.

With respect to the uncertainty, for `CLSP` instances with 3 state variables, we use the exact same scenarios as in [1]. For larger instances that are not covered in [1] we generate new scenarios using the same methodology. For `CFLP`, we use scenarios provided to us by Seranilla and Löhndorf [28].

4.1 Implementation Details

SDDiP and all cut generation approaches are implemented in Julia-1.5.3 [5] based on the existing packages `SDDP.jl` [15] and `JuMP.jl` [16]. The code is available on GitHub as part of a larger project called `DynamicSDDiP.jl` (see `https://github.com/ChrisFuelOR/DynamicSDDiP.jl`).

SDDiP is terminated after a predefined time limit or if the obtained lower bounds start to stall. In each forward pass, one scenario path is randomly sampled. After termination of SDDiP, an in-sample Monte Carlo simulation with 1000 replications is conducted on the finite scenario tree to compute a statistical upper bound for the obtained policy.

The Lagrangian dual problems in SDDiP are solved using a level bundle method with a maximum of 1000 (for `CLSP`) or 100 (for `CFLP`) iterations and an optimality tolerance of 10^{-4} (for `CLSP`) or 10^{-2} (for `CFLP`). The multipliers π_n are initialized with a vector of zeros and π_{n0} is initialized with 1. Sometimes the level bundle method reports infeasibilities in the quadratic auxiliary problem. In that case, we proceed with a standard Kelley step instead. Moreover, in the case of other numerical issues in solving the Lagrangian dual, the solution process is stopped and a valid cut is constructed with the current values of the multipliers.

For the LN cuts, as pointed out before, the choice of normalization coefficients (u_n, u_{n0}) is crucial for the cut quality, but also to achieve a bounded subproblem. If the chosen heuristic yields *only* coefficients (close to or) equal to zero, in our implementation no cut is generated at all. Moreover, non-zero coefficients may not yield a bounded dual problem if they correspond to a direction $(u_n, u_{n0}) \notin \text{cone}\left(\text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1})) - (x_{a(n)}^i, \theta_n^i)\right) \setminus \{0\}$, see Sect. 3.6.2. We use infeasibility of problem (17) as a flag for possible unboundedness. In such a case, we only generate an `SB` cut. Additionally, note that if problem (17) is feasible, we obtain an upper bound for the optimal value of the dual problem (11). We can use this bound to ensure boundedness for the approximating models in the level bundle method.

For `CLSP-Bin`, some preliminary tests indicated a proneness to degeneracy of the dual problem, and thus cuts of bad quality, for approach ℓ^∞. Therefore, for this case we perform an additional step minimizing the norm among all optimal dual solutions, as was previously proposed in `SDDP.jl` [15]. This is not the case for `CLSP` and `CFLP`, though.

All occuring LP, MILP and QP subproblems are solved using Gurobi 9.0.3 with an optimality tolerance of 10^{-4} and a time limit of 300 seconds. All tests are run on a Windows machine with 64 GB RAM and an Intel Core i7-7700K processor (4.2 GHz).

4.2 `CLSP-Bin`: Comparison of Individual Types of Cuts

For our first experiments we apply SDDiP to `CLSP-Bin`, and only use one type of cut for the whole solution process. The results are illustrated in several figures throughout this section. The full results are provided in Appendix D.

First, we consider experiments with $T = 4$ or $T = 6$ stages and a maximum run time of 3 hours and 4 hours, respectively. The obtained lower bounds are depicted in Fig. 1. We observe that `B` and `SB` do not manage to close the optimality gap and that the obtained lower bounds stall very fast. `L`, whereas better in theory, leads to even worse lower bounds. One reason is that solving the dual problems is computationally costly, but additionally, compared to `SB` the tighter cuts seem to lead to worse incumbents for the upcoming stages or iterations. Many variants of deep and LN Lagrangian cuts outperform `SB` and `L` with respect to the lower bounds and gaps, even if the optimality gap is not completely closed in the predefined time horizon. While the quality of the lower bounds is better, the iteration times and the number of iterations in the bundle method are not necessarily reduced despite solving a bounded problem, especially not for `LN` cuts.

We then consider experiments with $T = 10$ and $T = 16$, with run times of 5 and 8 hours, respectively. The obtained lower bounds are depicted in Fig. 2. We observe that deep cuts perform mediocre for 16 stages. `LN` cuts perform best with respect to the lower bounds, but for 16 stages hardly outperform `SB`. This is mainly due to long iteration times, even in comparison to deep cuts (see Appendix D.1). Moreover, as Fig. 3 shows,

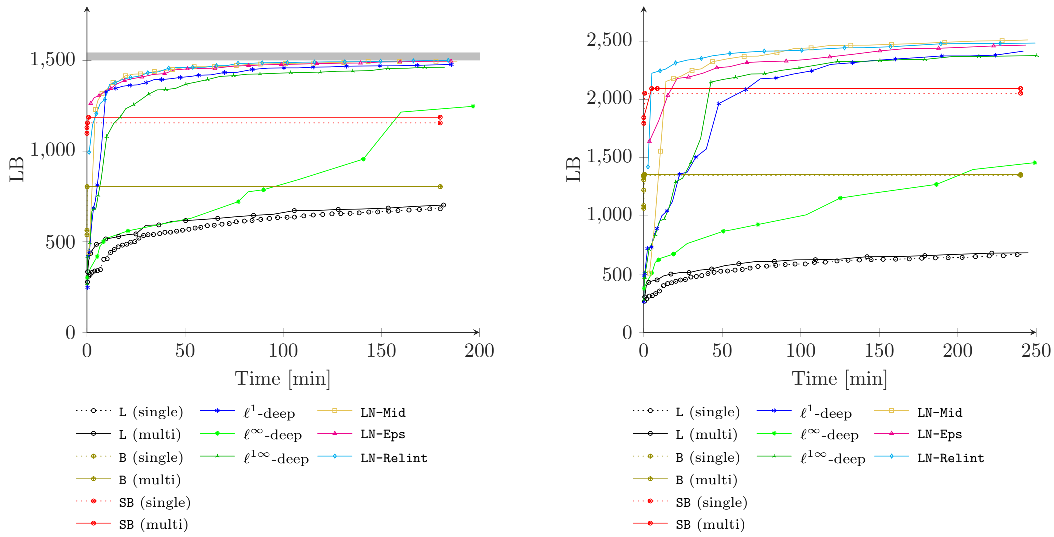


Figure 1: Lower bound development over time for experiments on `CLSP-Bin`.

Note. LEFT: $T = 4$. RIGHT: $T = 6$. For $T = 4$, the shaded gray area is where the optimal value lies according to an approximate solution of the deterministic equivalent. For `B` and `SB`, SDDiP quickly terminates due to stalling lower bounds, so the last lower bound is interpolated over the whole time horizon. Marks at every second iteration, except for `B` and `SB`.

an improvement in lower bounds does not necessarily translate to an improvement of the obtained policies. In fact, `SB` achieves the best simulated upper bounds. It seems that using `SB` it is possible to quickly identify good feasible solutions, but that the lower bounds are too loose to get a certificate for optimality, whereas for Lagrangian cuts it is the opposite.

Finally, we observe no cases of potential unboundedness for `LN` cuts, so the core point identification heuristics seem to work pretty well overall.

4.3 `CLSP-Bin`: Combination with SB Cuts

As shown in the previous section, using SDDiP with *only* Lagrangian cuts becomes extremely slow for large problems. Therefore, in practice, it is reasonable to combine different types of cuts. Already in the original SDDiP work [33] it is proposed to combine Lagrangian cuts, which can provide convergence guarantees, and strengthened Benders cuts, which can be computed efficiently.

To evaluate the performance of deep and LN cuts in this setting, we conduct experiments where we start with only `SB` for the first 20 iterations to get a quick bound improvement, and then generate both SB cuts and Lagrangian cuts in each iteration. The lower bounds are depicted in Fig. 4, while the simulation results and optimality gaps are presented in Fig. 3 – in comparison to those from the previous section.

The lower bound results heavily improve for `L`, but are not affected too much for deep or `LN` cuts. They are still better for these approaches than the ones obtained using only `SB`, though. As the simulated upper bounds are better than in the previous setting for all types of cuts, we can conclude that a combination of SB cuts and Lagrangian cuts combines the advantages of good lower bounds and reasonably good simulation results for the policies.

Worth mentioning, another approach suited to accelerate SDDiP was recently put

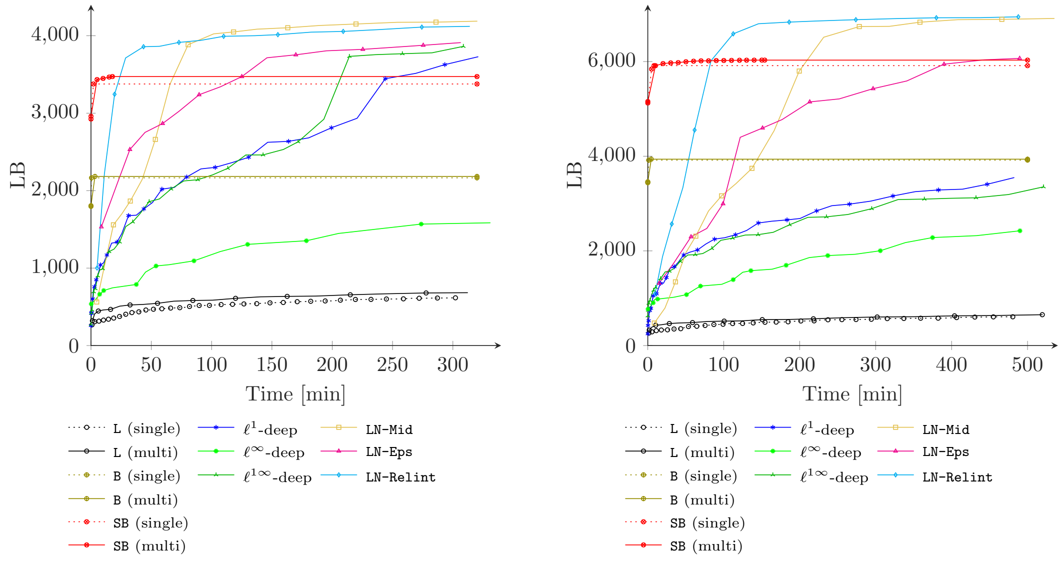


Figure 2: Lower bound development over time for experiments on CLSP-Bin.

Note. LEFT: $T = 10$. RIGHT: $T = 16$.

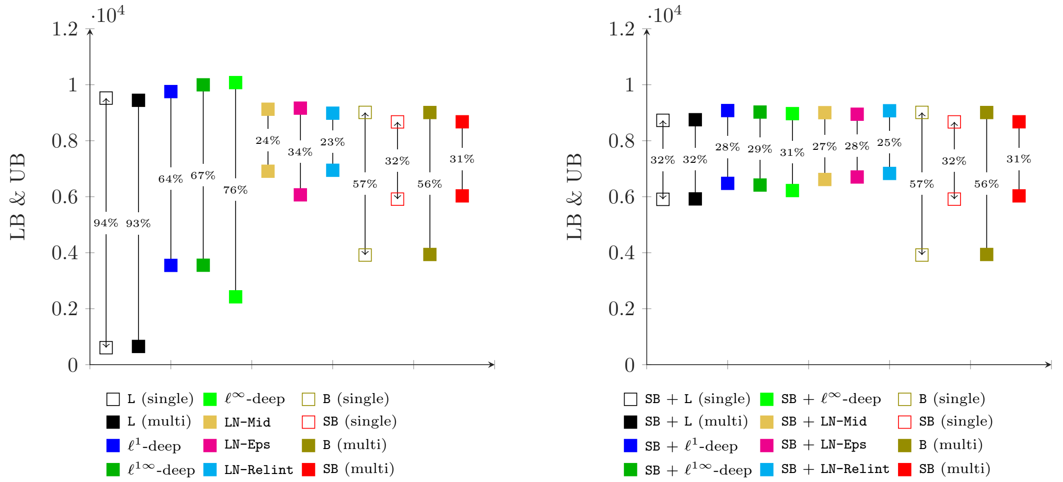


Figure 3: Optimality gaps for experiments on CLSP-Bin with $T = 16$.

Note. LEFT: Run with only one type of cuts. RIGHT: Runs with SB plus Lagrangian cuts starting from iteration 21.

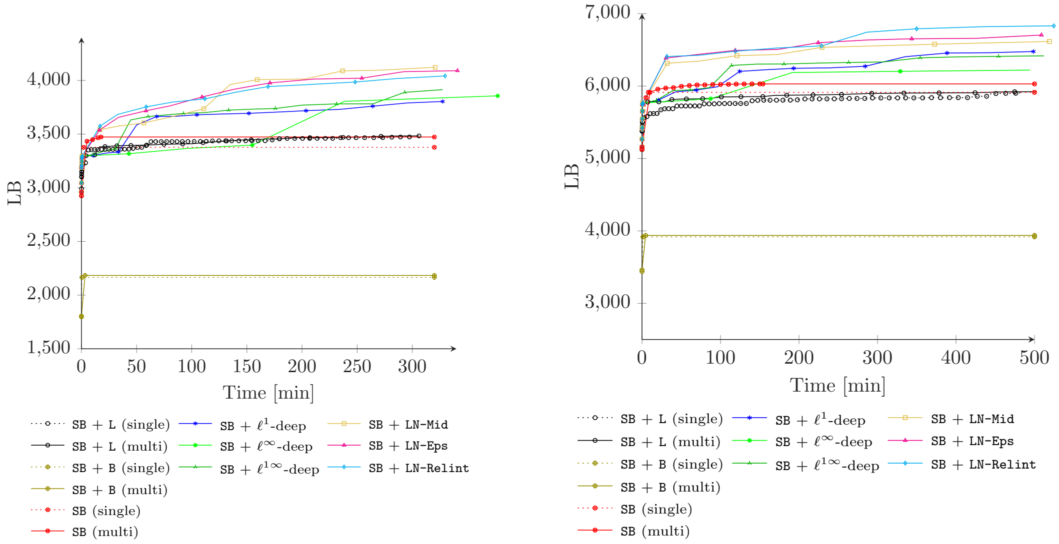


Figure 4: Lower bound development over time for experiments on `CLSP-Bin` using a combination with SB cuts.

Note. LEFT: $T = 10$. RIGHT: $T = 16$. 20 iterations with `SB` and then `SB` and Lagrangian cuts in each iteration.

forward by Chen and Luedtke [11] for the two-stage stochastic case. The idea is to artificially restrict the dual feasible space in (11) in a reasonable way. Whereas this eliminates the convergence guarantees of SDDiP, it may significantly speed-up the solution of the dual problems. We present this approach in more detail in Appendix D.2 and show that in our computational experiments additional lower bound improvements can be achieved using it.

4.4 Results for `CLSP`

As discussed before, for experiments of `CLSP` with 10 state variables, we do not apply a binary approximation. This means that we cannot expect the optimality gap to be closed. However, in return iterations should take considerably less time. Note that this is often the go-to approach to apply SDDP-like methods to mixed-integer programs in practice.

The results show that deep and `LN` Lagrangian cuts manage to achieve better lower bounds and gaps than conventional cuts, see Fig. 5 and 6. This is a considerable improvement compared to Benders cuts, which are most frequently used in practice. It shows that the proposed cut generation techniques may be helpful to improve the convergence behavior of SDDiP even if no binary approximation is applied. However, as for the previous experiments, we cannot conclude that the improvement in lower bounds necessarily leads to an improvement of the in-sample performance of the obtained policy, and thus to better optimality gaps.

We also observe that the average iteration time is reduced considerably (by 60-75%) compared to `CLSP-Bin`. For this reason, and because the considered state space is lower-dimensional, even without theoretical convergence guarantees, better optimality gaps are obtained for `CLSP` in the same time.

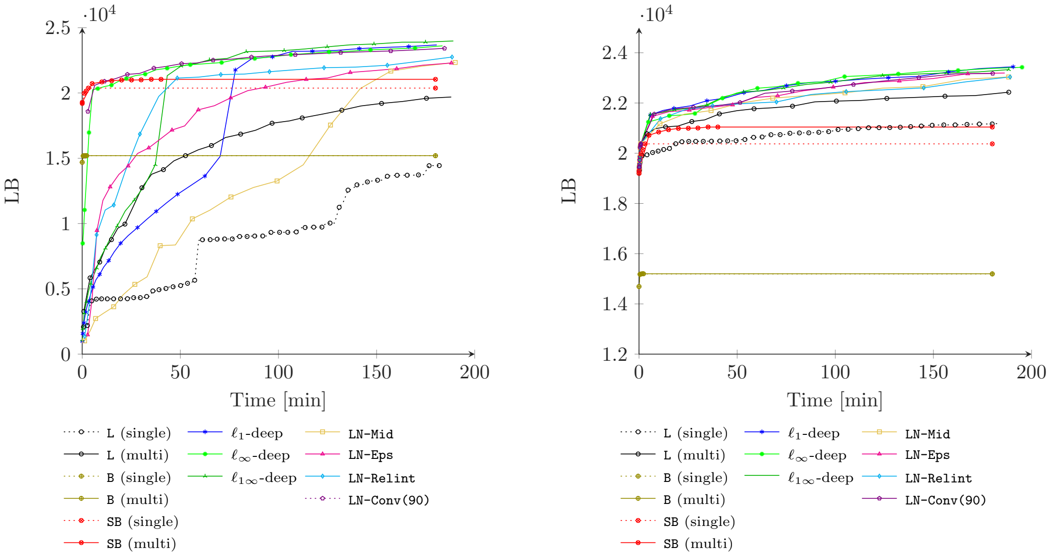


Figure 5: Lower bound development over time for experiments on `CLSP` with $T = 16$.

Note. UP LEFT: Run with only one type of cuts. UP RIGHT: Runs with `SB` plus additional cuts starting from iteration 21.

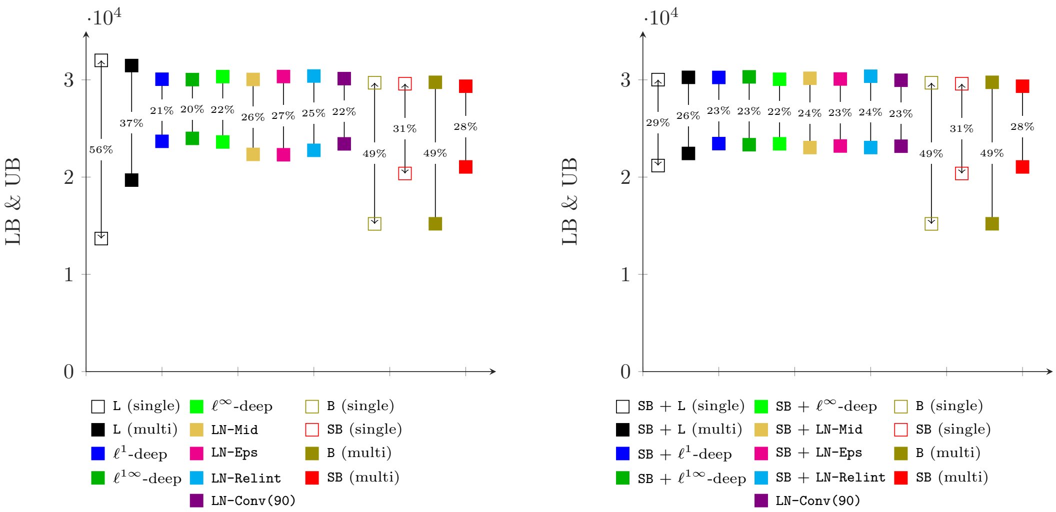


Figure 6: Optimality gaps for experiments on `CLSP` with $T = 16$.

Note. LEFT: Run with only one type of cuts. RIGHT: Runs with `SB` plus additional cuts starting from iteration 21.

4.5 Results for CFLP

For CFLP, again we run experiments with only one type of cut and experiments where we use SB in combination with different Lagrangian cuts. The results are depicted in Fig. 7.

We observe that deep cuts and some types of LN cuts do not yield better lower bounds than B, SB or standard Lagrangian cuts L. The bounds also start to stall fast, with a remaining optimality gap of about 20% (in difference to CLSP, SB also does not lead to better simulated upper bounds here). In contrast, LN cuts that use core points with first components relatively close to $\mathrm{mid}(\mathrm{conv}(X_{a(n)}))$, *i.e.*, Mid, Conv(50) and Conv(75), yield extremely good lower bounds in very few iterations, and almost manage to close the optimality gap. At least for Mid, this is a bit surprising, because for Mid and Relint we detect unboundedness due to integer constraints (cf. Sect. 3.6.2, 4.1) in 85% and 99% of all considered dual problems, respectively, leading to the generation of SB cuts. This means that for Mid only in 15% of the cases LN cuts are constructed, and still this is sufficient to almost close the optimality gap. A combination with SB does not lead to substantial differences in the observed results.

The main drawback of LN cuts, and in particular Conv, where no cases of unboundedness occur, is that the average time spent for one iteration is extremely high. This is mainly due to the large number of 2,000 dual problems to be solved per iteration. Whereas this is also true for L or deep cuts, in these cases only very few bundle method iterations are required to solve the dual problems on average. On the other hand, similarly to B and SB, this leads to extremely flat cuts, and thus to only slowly improving lower bounds.

4.6 Discussion and Potential Improvements

Overall, our results show significant improvements of the obtained lower bounds using the new cut generation framework in almost all cases: CLSP-Bin, CLSP, CFLP, when combining Lagrangian cuts with strengthened Benders cuts or applying them on their own. For CLSP-Bin, especially LN cuts yield strong improvements, whereas for CLSP also deep cuts perform reasonably well. For CFLP in particular those LN cut types seem beneficial that use core points with the first components close to $\mathrm{mid}(\mathrm{conv}(X_{a(n)}))$.

We see that better lower bounds do not in all cases (especially for CLSP and CLSP-Bin) translate to better performances of the obtained policies, though. Additionally, we observe that even using the new framework, SDDiP suffers from well-known computational drawbacks such as high computational cost to solve Lagrangian dual problems (especially for CLSP-Bin and CFLP) and slow convergence of lower bounds due to premature stalling [2], so even after hours of run time the observed optimality gaps often are still considerable. Finally, the proposed cut generation framework requires a multi-cut approach which considerably increases the computational burden of SDDiP, as much more cuts have to be added per iteration than using the often-preferred single-cut approach.

In the future, the performance of SDDiP including our proposed cut generation framework could be improved in several ways. First, the solution of independent Lagrangian duals for nodes $m \in \mathcal{C}(n)$ could be parallelized. Second, potential warm starting or acceleration techniques for the Lagrangian dual (*e.g.* using sub-optimal solutions) could be explored. Third, the dual space restriction suggested by [11] (see Appendix D.2) looks promising to reduce the computational effort for solving Lagrangian dual problems, while not compromising cut quality by too much. We think that future

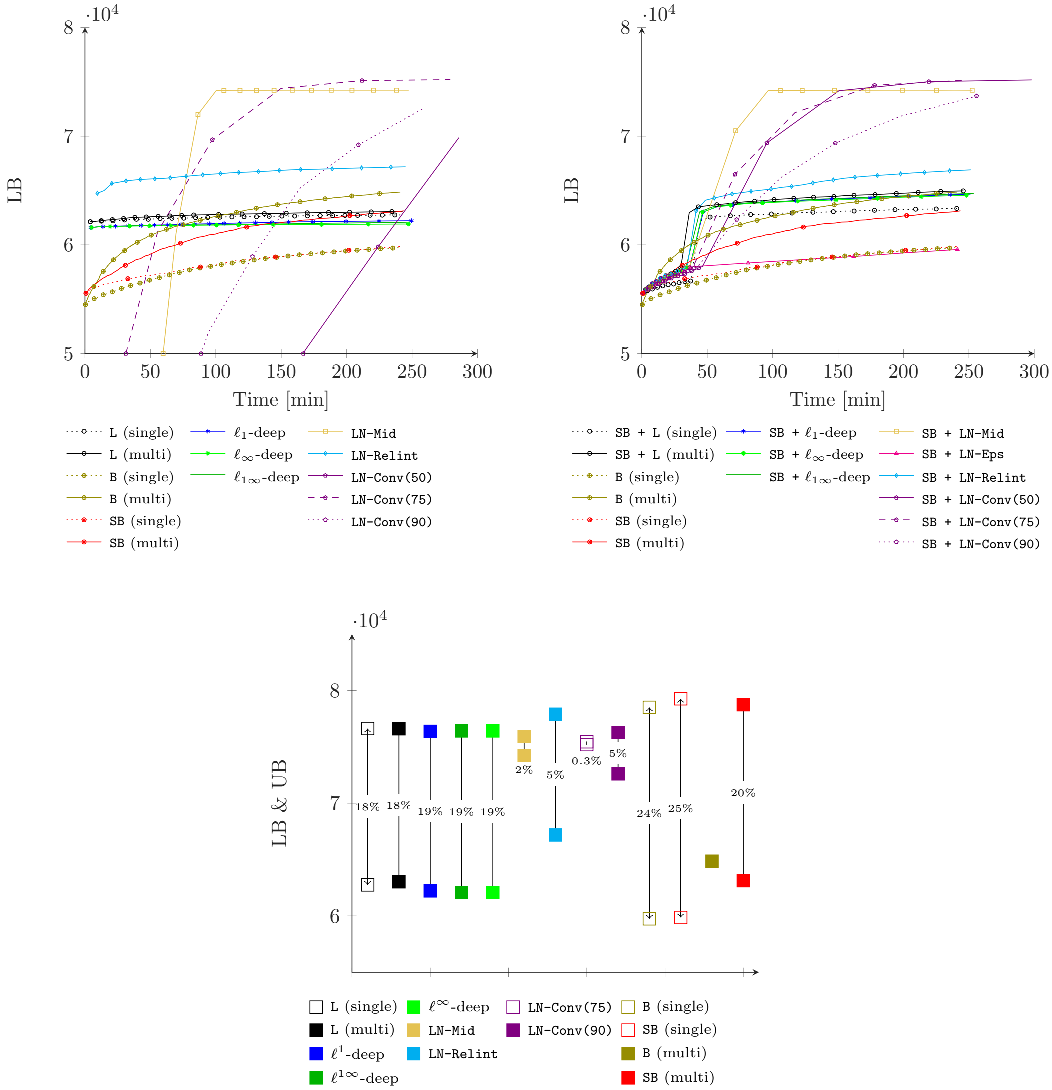


Figure 7: Computational results for experiments on `CFLP`.

Note. UP LEFT: Lower bound development for using only one type of cut. UP RIGHT: Lower bound development for runs with `SB` plus additional cuts from iteration 21. BELOW: Optimality gaps for runs with only one type of cut.

research could focus more on priorly restricting the dual space. Fourth, it might be of interest to investigate potential extensions of the proposed cut generation framework to a single-cut approach.

Our experiments also reveal that, especially for LN cuts, SDDiP may occasionally suffer from numerical issues, and that the obtained results show a high sensitivity with respect to the chosen parameters. It is a common issue of cutting-plane methods that they may lead to ill-conditioned problems if the cut and problem coefficients are not properly scaled. In this context, an appropriate choice of normalization coefficients (u_n, u_{n0}) for LN cuts is crucial. In general, core point identification remains a challenging task, especially when integer requirements are apparent. Addressing these challenges in detail merits further research.

5 Conclusion

In this article, we propose a new framework to generate Lagrangian cuts for value functions occurring in MS-MILPs, which generalizes earlier proposals for 2-stage problems. We prove that using different normalizations of the Lagrangian dual problems, cuts with different favorable properties can be obtained, such as maximal depth, being facet-defining or Pareto-optimal. Our framework allows for a lot of flexibility in cut generation, and thus notably extends the toolbox of SDDiP.

We provide computational results for experiments on a capacitated lot-sizing and a capacitated facility location problem. The results show that the lower bounds in SDDiP can be vastly improved by incorporating our proposed framework, although not eliminating other well-known computational drawbacks, such as excessive computational effort, slow convergence and inability to close the optimality gap. As described in the previous section, therefore more theoretical and computational research is required to efficiently apply our proposed framework, and SDDiP in general, on large-scale problems in practice.

Finally, it should be possible and may be worth exploring to extend the proposed framework to Lipschitz regularized subproblems and value functions for MS-MILPs. This would allow to exploit different types of Lagrangian dual problems in larger frameworks generating non-convex approximations for the value functions, such as the one presented in [18] and [19].

Acknowledgements

Andy Sun's research is partially funded by the National Science Foundation CAREER award 2316675. Christian Füllner's research is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)–445857709. This work was started during Christian Füllner's research visit at Georgia Institute of Technology, which was funded by the Karlsruhe House of Young Scientists (KHYS). The authors thank Filipe Cabral for providing data for the CLSP, and Bonn Kleiford Seranilla and Nils Löhndorf for providing data for the CFLP instances considered in the computational experiments.

References

[1] S. Ahmed, F. G. Cabral, and B. Freitas Paulo da Costa. Stochastic Lipschitz dynamic programming. *Mathematical Programming*, 191:755–793, 2022.

[2] D. Ávila, A. Papavasiliou, and N. Löhndorf. Batch learning SDDP for long-term hydrothermal planning. *IEEE Transactions on Power Systems*, 39(1):614–627, 2024.

[3] E. Balas and P. L. Ivanescu. On the generalized transportation problem. *Management Science*, 11(1):188–202, 1964.

[4] D. P. Bertsekas. *Convex Optimization Theory*. Athena Scientific, 2009.

[5] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah. Julia: a fresh approach to numerical computing. *SIAM Review*, 59(1):65–98, 2017.

[6] J. R. Birge. Solution methods for stochastic dynamic linear programs. Technical report, Stanford University CA Systems Optimization Lab, 1980.

[7] J. R. Birge and F. Louveaux. *Introduction to stochastic programming*. Springer Series in Operations Research and Financial Engineering. Springer Science & Business Media, 2nd edition, 2011.

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

[9] R. Brandenberg and P. Stursberg. Refined cut selection for benders decomposition: applied to network capacity expansion problems. *Mathematical Methods of Operations Research*, 94:383–412, 2021.

[10] F. Cadoux. Computing deep facet-defining disjunctive cuts for mixed-integer programming. *Mathematical Programming, Ser. A*, 122:197–223, 2010.

[11] R. Chen and J. Luedtke. On generating Lagrangian cuts for two-stage stochastic integer programs. *INFORMS Journal on Computing*, 2022.

[12] M. Conforti and L. A. Wolsey. "Facet" separation with one linear program. *Mathematical Programming, Ser. A*, 178:361–380, 2019.

[13] G. Cornuéjols and C. Lemaréchal. A convex-analysis perspective on disjunctive cuts. *Mathematical Programming, Ser. A*, 106:567–586, 2006.

[14] S. S. Dey and M. Molinaro. Theoretical challenges towards cutting-plane selection. *Mathematical Programming*, 170:237–266, 2018.

[15] O. Dowson and L. Kapelevich. SDDP.jl: a Julia package for stochastic dual dynamic programming. *INFORMS Journal on Computing*, 33(1):27–33, 2020.

[16] I. Dunning, J. Huchette, and M. Lubin. JuMP: a modeling language for mathematical optimization. *SIAM Review*, 59(2):295–320, 2017. doi: 10.1137/15m1020575.

[17] M. Fischetti, D. Salvagnin, and A. Zanette. A note on the selection of Benders' cuts. *Mathematical Programming, Ser. B*, 124:175–182, 2010.

[18] C. Füllner and S. Rebennack. Non-convex nested Benders decomposition. *Mathematical Programming*, 196:987–1024, 2022.

[19] C. Füllner, X. A. Sun, and S. Rebennack. On Lipschitz regularization and Lagrangian cuts in multistage stochastic mixed-integer linear programming. Preprint, available online at `https://optimization-online.org/?p=27295.pdf`, 2024.

[20] A. M. Geoffrion. Lagrangean relaxation for integer programming. In M.L. Balinski, editor, *Approaches to integer programming*, pages 82–114. Springer Berlin Heidelberg, Berlin, Heidelberg, 1974. ISBN 978-3-642-00740-8.

[21] M. Guignard. Lagrangean relaxation. *TOP*, 11(2):151–228, 2003.

[22] M. Hosseini and J. G. Turner. Deepest cuts for Benders decomposition. Preprint, available online at `https://arxiv.org/pdf/2110.08448`, 2021.

[23] T.L. Magnanti and R.T. Wong. Accelerating benders decomposition: algorithmic enhancement and model selection criteria. *Operations Research*, 29(3):464–484, 1981.

[24] N. Papadakos. Practical enhancements to the Magnanti–Wong method. *Operations Research Letters*, 36:444–449, 2008.

[25] M. V. F. Pereira and L. M. V. G. Pinto. Multi-stage stochastic optimization applied to energy planning. *Mathematical Programming*, 52(1-3):359–375, 1991.

[26] R. Rahmaniani, S. Ahmed, T. G. Crainic, M. Gendreau, and W. Rei. The Benders dual decomposition method. *Operations Research*, 68(3):878–895, 2020.

[27] K. Seo, S. Joung, C. Lee, and S. Park. A closest Benders cut selection scheme for accelerating the Benders decomposition algorithm. *INFORMS Journal on Computing*, pages 1–24, 2022.

[28] B. K. Seranilla and N. Löhndorf. Multistage stochastic facility location under facility disruption uncertainty. Preprint, available online at `https://optimization-online.org/?p=26395.pdf`, 2024.

[29] H. D. Sherali and B. J. Lunday. On generating maximal nondominated Benders cuts. *Annals of Operations Research*, 210:57–72, 2013.

[30] P. M. Stursberg. *On the mathematics of energy system optimization*. PhD thesis, Technische Unviersität München, 2019.

[31] W. W. Trigeiro, L. J. Thomas, and J. O. McClain. Capacitated lot sizing with setup times. *Management Science*, 35(3):353–366, 1989.

[32] S. Zhang and X. A. Sun. Stochastic dual dynamic programming for multistage stochastic mixed-integer nonlinear optimization. *Mathematical Programming*, 2022.

[33] J. Zou, S. Ahmed, and X. A. Sun. Stochastic dual dynamic integer programming. *Mathematical Programming*, 175:461–502, 2019.

A Classical Lagrangian Cuts

For any $n \in \overline{\mathcal{N}}$, let $\mathfrak{Q}_n^{i+1}(\cdot)$ denote the current cut approximation for value function $Q_n(\cdot)$. Then, a Lagrangian cut can be generated by considering a special Lagrangian relaxation of the nodal subproblem (that is, subproblem (3) with $Q_m(\cdot), m \in \mathcal{C}(n)$, replaced by $\mathfrak{Q}_m^{i+1}(\cdot)$). More precisely, the copy constraints $z_n = x_{a(n)}^i$ are relaxed using a given vector of dual multipliers $\pi_n \in \mathbb{R}^{d_{a(n)}}$. This yields

$$\begin{aligned}\mathcal{L}_n^{i+1}(\pi_n) := \min_{x_n, y_n, z_n, (\theta_m)} \Big\{ f_n(x_n, y_n) + \sum_{m \in \mathcal{C}(n)} p_{nm}\theta_m - \pi_n^\top z_n \ : \ (z_n, x_n, y_n) \in \mathcal{F}_n, \\ z_n \in Z_{a(n)}, \theta_m \geq \mathfrak{Q}_m^{i+1}(\cdot)(x_n), \ m \in \mathcal{C}(n) \Big\}.\end{aligned}$$

For varying π_n, this relaxation defines the *dual function* $\mathcal{L}_n^{i+1}(\cdot)$. The problem of optimizing the dual function over the dual multipliers π_n is the *Lagrangian dual problem*:

$$\max_{\pi_n} \Big\{ \mathcal{L}_n^{i+1}(\pi_n) + \pi_n^\top x_{a(n)}^i \Big\}. \tag{18}$$

By solving problem (18), a Lagrangian cut for $Q_n(\cdot)$ can be derived as

$$\theta_n \geq \mathcal{L}_n^{i+1}(\pi_n^i) + (\pi_n^i)^\top x_{a(n)}, \tag{19}$$

where π_n^i denotes feasible dual multipliers in (18) for node n [33]. If required, feasibility cuts can be derived in a similar fashion [see 11, 26].

B Proofs

In this section, we present the proofs that are not displayed in the main text.

B.1 Proof of Lemma 3.4

Proof. Let $(x_{a(n)}^i, \theta_n^i) \in \operatorname{epi}(\underline{Q}_n^{i+1})$. Then according to (4) we have

$$\theta_n^i \geq \min_{\lambda_n, z_n} \Big\{ c_n^\top \lambda_n \ : \ (\lambda_n, z_n) \in \mathcal{W}_n^{i+1}, \ z_n = x_{a(n)}^i \Big\}. \tag{20}$$

This implies that there exists some (λ_n, z_n) such that for $(x_{a(n)}^i, \theta_n^i)$ all constraints of (7) are satisfied. Hence, $v_n^{f,i+1}(x_{a(n)}^i, \theta_n^i) = 0$.

Let $v_n^{f,i+1}(x_{a(n)}^i, \theta_n^i) = 0$. Then, there exist (λ_n, z_n) such that for $(x_{a(n)}^i, \theta_n^i)$ all constraints of (7) are satisfied. This implies (20), and thus $(x_{a(n)}^i, \theta_n^i) \in \operatorname{epi}(\underline{Q}_n^{i+1})$. □

B.2 Proof of Theorem 3.5

Proof. (i) is a standard result on Lagrangian relaxation [see 21]. Another well-known property of the Lagrangian dual is that it is equivalent to a primal convexification of the original subproblem [20]. In our case, this convexification is given by

$$\min_{\lambda_n, z_n} \Big\{ 0 \ : \ (\lambda_n, z_n) \in \operatorname{conv}(\mathcal{W}_n^{i+1}), \ z_n = x_{a(n)}^i, \ \theta_n^i \geq c_n^\top \lambda_n \Big\}. \tag{21}$$

The closed convex envelope $\overline{\text{co}}(\underline{Q}_n^{i+1})(\cdot)$ can be expressed through the convex problem

$$\overline{\text{co}}(\underline{Q}_n^{i+1})(x^i_{a(n)}) = \min_{\lambda_n, z_n} \Big\{ c_n^\top \lambda_n \ : \ (\lambda_n, z_n) \in \text{conv}(\mathcal{W}_n^{i+1}), \ z_n = x^i_{a(n)} \Big\}, \tag{22}$$

see for instance [19, Theorems 3.8 and 3.9] for a formal proof. Therefore, by the same reasoning as in Lemma 3.4, problem (21) is a feasibility problem for $\text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1})$. □

B.3 Proof of Lemma 3.7

Proof. This is proven in [11] for $\text{epi}(Q_n)$, but we provide a customized proof here. Let $(x_{a(n)}, \theta_n) \in \text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1}))$. Then,

$$\begin{aligned}
(\pi_n^i)^\top x_{a(n)} + \pi^i_{n0}\theta_n &\geq \min_{x_{a(n)}, \theta_n} \Big\{ (\pi_n^i)^\top x_{a(n)} + \pi^i_{n0}\theta_n \ : \ (x_{a(n)}, \theta_n) \in \text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1})) \Big\} \\
&= \min_{\lambda_n, z_n} \Big\{ (\pi_n^i)^\top z_n + \pi^i_{n0} c_n^\top \lambda_n \ : \ (\lambda_n, z_n) \in \text{conv}(\mathcal{W}_n^{i+1}) \Big\} \\
&= \min_{\lambda_n, z_n} \Big\{ (\pi_n^i)^\top z_n + \pi^i_{n0} c_n^\top \lambda_n \ : \ (\lambda_n, z_n) \in \mathcal{W}_n^{i+1} \Big\} \\
&= \mathscr{L}_n^{i+1}(\pi_n^i, \pi_{n0}^i).
\end{aligned}$$

The inequality follows by feasibility. The first equality uses the same relation that is also applied in (21). The second equality exploits that the objective function is linear, and the last one follows from the definition of $\mathscr{L}_n^{i+1}(\cdot)$ in (8). The second part of the assertion follows with $\text{epi}(Q_n) \subseteq \text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1}))$. □

B.4 Proof of Lemma 3.10

Proof. According to [9], $\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n)$ can be rewritten as

$$\begin{aligned}
\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n) = \Big\{ &(\gamma_n, \gamma_{n0}) \in \mathbb{R}^{d_{a(n)}} \times \mathbb{R} \ : \\
&\gamma_n^\top x^i_{a(n)} + \gamma_{n0}\theta^i_n - \text{supp}_{\text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1}))}(\gamma_n, \gamma_{n0}) \geq 1 \Big\},
\end{aligned} \tag{23}$$

where $\text{supp}_{\text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1}))}(\cdot)$ denotes the support function of $\text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1}))$. This function can be expressed as follows:

$$\begin{aligned}
&\text{supp}_{\text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1}))}(\gamma_n, \gamma_{n0}) \\
&= \max_{x_{a(n)}, \theta_n} \{ \gamma_n^\top x_{a(n)} + \gamma_{n0}\theta_n \ : \ (x_{a(n)}, \theta_n) \in \text{epi}(\overline{\text{co}}(\underline{Q}_n^{i+1})) \} \\
&= \max_{x_{a(n)}, \theta_n, \lambda_n, z_n} \Big\{ \gamma_n^\top x_{a(n)} + \gamma_{n0}\theta_n \ : \ \tilde{A}_n \lambda_n + \tilde{B}_n z_n \geq \tilde{d}_n, z_n = x_{a(n)}, \theta_n - c_n^\top \lambda_n \geq 0 \Big\} \\
&= \min_{\mu_n, \pi_n, \pi_{n0}} \Big\{ \tilde{d}_n^\top \mu_n \ : \ \tilde{A}_n^\top \mu_n - c_n \pi_{n0} = 0, \tilde{B}_n^\top \mu_n - \pi_n = 0, \\
&\qquad\qquad \pi_n = \gamma_n, \pi_{n0} = \gamma_{n0}, \pi_{n0} \leq 0, \mu_n \leq 0 \Big\} \\
&= \min_{\mu_n} \Big\{ - \tilde{d}_n^\top \mu_n \ : \ -\tilde{A}_n^\top \mu_n - c_n \gamma_{n0} = 0, -\tilde{B}_n^\top \mu_n - \gamma_n = 0, \gamma_{n0} \leq 0, \mu_n \geq 0 \Big\}.
\end{aligned} \tag{24}$$

The first equation applies the definition of support functions. The second one follows from Remark 3.3 and the third one exploits strong duality for LPs. We insert (24) into (23), and observe that the set remains unchanged if we replace the minimum operator using an existence quantor. □

B.5 Proof of Lemma 3.14

Proof. According to Theorem 3.5, we have $\widehat{v}_n^{D,i+1}(x_{a(n)}^i, \theta_n^i) = 0$ for the non-normalized Lagrangian dual (9). By definition of (9) and its normalization (11), we can thus conclude $\widehat{v}_n^{ND,i+1}(x_{a(n)}^i, \theta_n^i) \leq \widehat{v}_n^{D,i+1}(x_{a(n)}^i, \theta_n^i) = 0$.

Let $(\widehat{\pi}_n, \widehat{\pi}_{n0})$ be an optimal point for problem (9), *i.e.*, $\mathscr{L}_n^{i+1}(\widehat{\pi}_n, \widehat{\pi}_{n0}) - (\widehat{\pi}_n)^\top x_{a(n)}^i - \widehat{\pi}_{n0}\theta_n^i = 0$. If $\|\widehat{\pi}_n, \widehat{\pi}_{n0}\| \leq 1$, then it is also feasible for (11). As the objective of both problems is the same, $\widehat{v}_n^{ND,i+1}(x_{a(n)}^i, \theta_n^i) = 0$.

Otherwise, there exists $\mu > 0$ such that $\frac{1}{\mu}(\widehat{\pi}_n, \widehat{\pi}_{n0})$ is feasible for (11). By feasibility, it follows

$$\begin{aligned}\widehat{v}_n^{ND,i+1}(x_{a(n)}^i, \theta_n^i) &\geq \mathscr{L}_n^{i+1}\left(\frac{1}{\mu}\widehat{\pi}_n, \frac{1}{\mu}\widehat{\pi}_{n0}\right) - \frac{1}{\mu}(\widehat{\pi}_n)^\top x_{a(n)}^i - \frac{1}{\mu}\widehat{\pi}_{n0}\theta_n^i \\ &= \frac{1}{\mu}\left(\mathscr{L}_n^{i+1}(\widehat{\pi}_n, \widehat{\pi}_{n0}) - (\widehat{\pi}_n)^\top x_{a(n)}^i - \widehat{\pi}_{n0}\theta_n^i\right) = 0,\end{aligned} \tag{25}$$

where we exploited that $\mathscr{L}_n^{i+1}(\cdot)$ is positive homogeneous. The reverse direction can be shown in a similar way. □

B.6 Proof of Lemma 3.21

Proof. By definition, the normalized Lagrangian dual problem is equivalent to

$$\begin{aligned}&\max_{(\pi_n,\pi_{n0})\in\Pi_n} \left\{ \min_{(\lambda_n,z_n)\in\mathcal{W}_n^{i+1}} \left\{ \pi_n^\top(z_n - x_{a(n)}^i) + \pi_{n0}(c_n^\top\lambda_n - \theta_n^i) \right\} \right\} \\ &= \max_{(\pi_n,\pi_{n0})\in\Pi_n} \left\{ \min_{(\lambda_n,z_n)\in\operatorname{conv}(\mathcal{W}_n^{i+1})} \left\{ \pi_n^\top(z_n - x_{a(n)}^i) + \pi_{n0}(c_n^\top\lambda_n - \theta_n^i) \right\} \right\}\end{aligned}$$

with $\Pi_n := \left\{(\pi_n, \pi_{n0}) \in \mathbb{R}^{d_{a(n)}} \times \mathbb{R} \;:\; \pi_{n0} \geq 0, u_n^\top \pi_n + u_{n0}\pi_{n0} \leq 1\right\}$. The equation follows from linearity. Using Remark 3.3 and then LP duality for the inner minimization problem, we obtain the equivalent problem

$$\begin{aligned}\max_{\pi_n,\pi_{n0},\mu_n} \Big\{ \tilde{d}_n^\top\mu_n - \pi_n^\top x_{a(n)}^i - \pi_{n0}\theta_n^i \;:\; &\mu_n \geq 0,\ \pi_{n0} \geq 0,\ u_n^\top\pi_n + u_{n0}\pi_{n0} \leq 1, \\ &\tilde{A}_n^\top\mu_n - \pi_{n0}c_n = 0,\ \tilde{B}_n^\top\mu_n - \pi_n = 0 \Big\}.\end{aligned}$$

This is an LP. Using LP duality and Remark 3.3 again, the assertion follows. □

B.7 Proof of Theorem 3.23

Proof. After reformulating the normalized Lagrangian dual problem (11) using Remark 3.3, and linking optimizing over the reverse polar set in (15) with optimizing over the so-called *relaxed alternative polyhedron* (see Theorem 2.1 and Corollary 2.2 in [9]), we can conclude from Theorem 3.20 in [30] that the following result holds:

A point $(\widehat{\gamma}_n, \widehat{\gamma}_{n0})$ is optimal for problem (15) if and only if there exists some certificate $\widehat{\mu}_n \geq 0$ in $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i)$ such that $(\mu_n^*, \pi_n^*, \pi_{n0}^*) = \alpha(-\widehat{\mu}_n, \widehat{\gamma}_n, \widehat{\gamma}_{n0})$ is optimal for problem (11), where $\alpha < 0$ is a negative scaling factor. Moreover, the optimal values of both problems multiply to -1. This result directly implies part (i) of the assertion.

We now prove (ii). Let $(\widehat{\gamma}_n, \widehat{\gamma}_{n0})$ be optimal for problem (15) with a valid certificate $\widehat{\mu}_n \geq 0$ in $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i)$. Then, the valid cut

$$\tilde{d}_n^\top \widehat{\mu}_n + (\widehat{\gamma}_n)^\top x_{a(n)} + \widehat{\gamma}_{n0}\theta_n \leq 0$$

is induced, separating $(x_{a(n)}^i, \theta_n^i)$ from $\operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$ [13]. According to the result used in part (i), this is equivalent to

$$-\frac{1}{\alpha}\tilde{d}_n^\top \mu_n^* + \frac{1}{\alpha}(\pi_n^*)^\top x_{a(n)} + \frac{1}{\alpha}\pi_{n0}^*\theta_n \leq 0.$$

However, given $\alpha < 0$, this is equivalent to

$$\tilde{d}_n^\top \mu_n^* - (\pi_n^*)^\top x_{a(n)} - \pi_{n0}^*\theta_n \leq 0$$

and by definition also to

$$\mathscr{L}_n^{i+1}(\pi_n^*, \pi_{n0}^*) - (\pi_n^*)^\top x_{a(n)} - \pi_{n0}^*\theta_n \leq 0,$$

which exactly corresponds to the Lagrangian cut (10). The reverse direction follows in a similar way. □

B.8 Proof of Lemma 3.24

Proof. Suppose that condition (16) is satisfied. Then there exist some $(\tilde{x}_{a(n)}, \tilde{\theta}_n) \in \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1})) - (x_{a(n)}^i, \theta_n^i)$ and $\mu > 0$ such that $(u_n, u_{n0}) = \mu(\tilde{x}_{a(n)}, \tilde{\theta}_n)$. This implies $(\tilde{x}_{a(n)} + x_{a(n)}^i, \tilde{\theta}_n + \theta_n^i) \in \operatorname{epi}(\overline{\operatorname{co}}(\underline{Q}_n^{i+1}))$. Therefore, the system

$$\Big\{(\lambda_n, z_n) \;:\; \tilde{\theta}_n + \theta_n^i \geq c_n^\top \lambda_n, (\lambda_n, z_n) \in \operatorname{conv}(\mathcal{W}_n^{i+1}), z_n = \tilde{x}_{a(n)} + x_{a(n)}^i\Big\}$$

is non-empty. However, this set is equivalent to

$$\Big\{(\lambda_n, z_n) \;:\; \frac{1}{\mu}u_{n0} \geq c_n^\top \lambda_n - \theta_n^i, (\lambda_n, z_n) \in \operatorname{conv}(\mathcal{W}_n^{i+1}), z_n - x_{a(n)}^i = \frac{1}{\mu}u_n\Big\}.$$

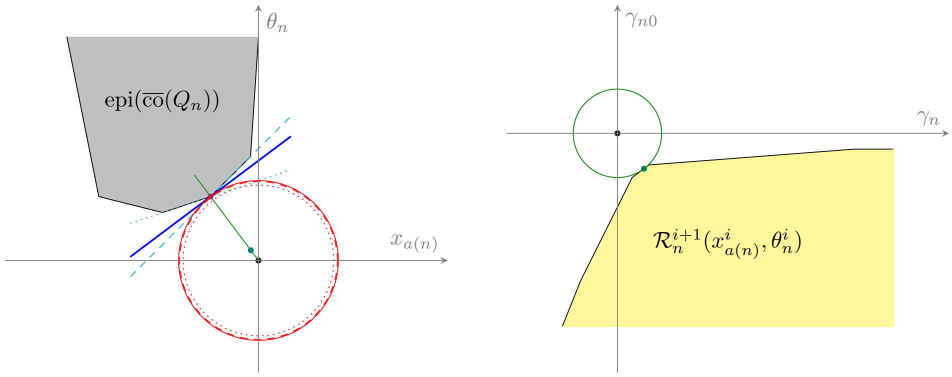
With choosing $\eta_n = \frac{1}{\mu} > 0$ it immediately follows that problem (13) is feasible. By $\eta_n \geq 0$, its optimal value is also bounded from below, hence it is finite. By LP duality this implies that Assumption 3 is satisfied. □

C Illustrative Examples

C.1 Illustrative Example for Deep Cuts

This example is inspired by illustrations from the literature [9, 22].

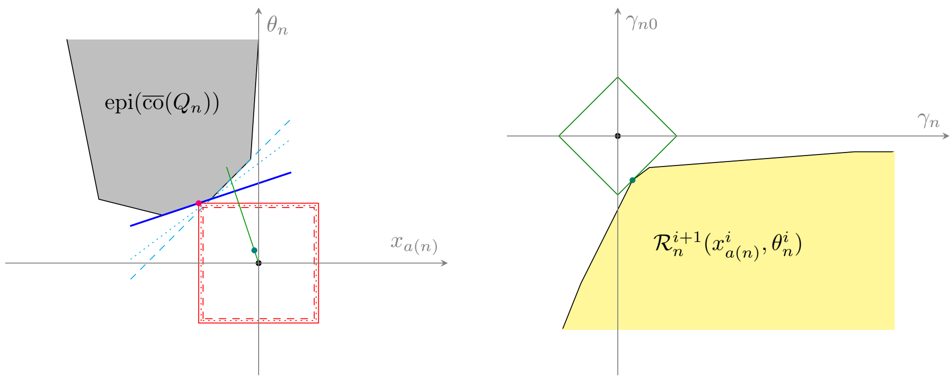
Example C.1. *We consider a given epigraph* $\operatorname{epi}(\overline{\operatorname{co}}(Q_n))$*, an incumbent* $(x_{a(n)}^i, \theta_n^i)$*, the associated reverse polar set* $\mathcal{R}_n^{i+1}(x_{a(n)}^i, \theta_n^i)$ *and the obtained deep Lagrangian cuts for different norms (*ℓ^2*,* ℓ^1*,* ℓ^∞ *and a weighted* ℓ^1*-norm). The sets and cuts are illustrated in Fig. 8-11 for the different norms. In each case, the illustration consists of two parts (a) and (b). In part (a), the incumbent (*black dot*) and the epigraph are* *depicted. Moreover, several norm balls are shown for the respective dual norms (*red lines*). We can see that the obtained deep Lagrangian cuts (*blue lines*) maximize the distance between the incumbent and the hyperplane in the dual norm. This is illustrated by depicting different valid cuts (*dashed/dotted cyan lines*) with smaller distances. On the other hand, it is also shown that the deep cuts minimize the distances between the incumbent and the epigraph in the dual norm and support the epigraphs at the corresponding projection to the epigraph (*violet line or point*). In part (b), the reverse polar set is depicted. Moreover, the optimal solutions (*teal line or point*) for optimizing the given norm (illustrated by a norm ball,* green line*) over the reverse polar set are highlighted. These solutions (apart from sign changes) characterize the normal vectors of the obtained cuts (see Lemma 3.19), as is additionally illustrated in part (a). Note that for none of the considered cases, the deep cuts are tight at $\big(x^i_{a(n)}, \overline{\text{co}}(Q_n)(x^i_{a(n)})\big)$, contrary to classical Lagrangian cuts.*



(a) Epigraph and deep cut.

(b) Reverse polar set.

Figure 8: Illustration of deep cuts for the ℓ^2-norm.



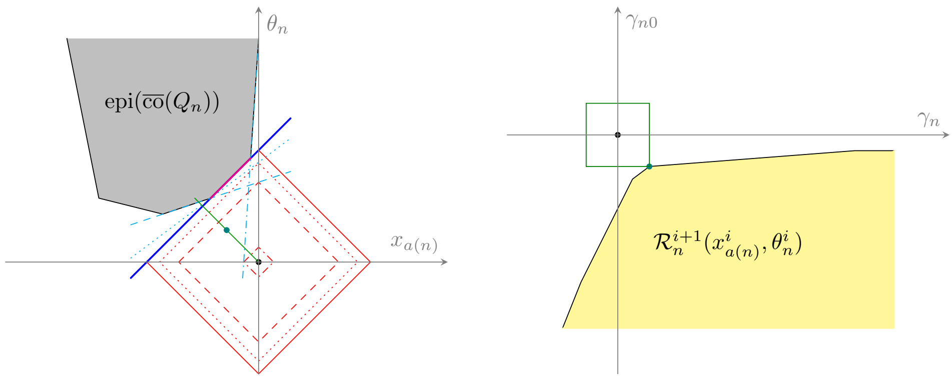
(a) Epigraph and deep cut.

(b) Reverse polar set.

Figure 9: Illustration of deep cuts for the ℓ^1-norm.

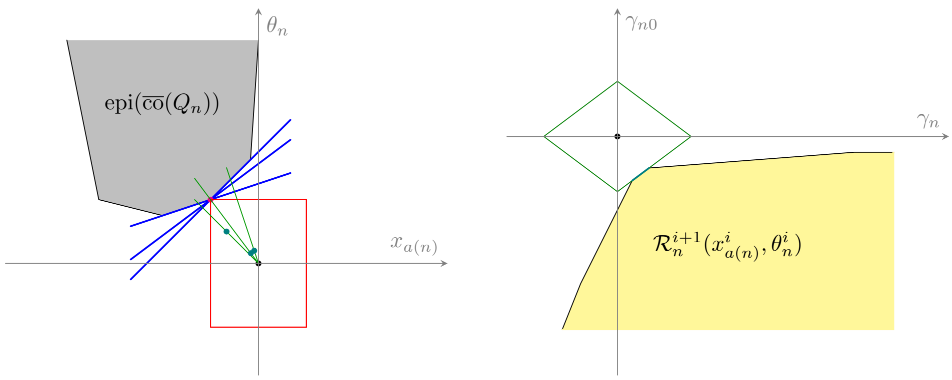
C.2 Illustrative Example for LN Cuts

Example C.2. *Consider epigraph* epi($\overline{\text{co}}(Q_n)$)*, incumbent* $(x^i_{a(n)}, \theta^i_n)$ *and reverse polar set* $\mathcal{R}^{i+1}_n(x^i_{a(n)}, \theta^i_n)$ *from Example C.2. These objects and an exemplary LN cut are*



(a) Epigraph and deep cut. (b) Reverse polar set.

Figure 10: Illustration of deep cuts for the ℓ^∞-norm.



(a) Epigraph and deep cut. (b) Reverse polar set.

Figure 11: Illustration of deep cuts for a weighted ℓ^1-norm.

*illustrated in Fig. 12. In part (a) the geometric idea of projection along a line segment is highlighted. The direction of this line segment is (u_n, u_{n0}) and obtained as the difference between a known core point (*yellow dot*) in* $\mathrm{epi}(\overline{\mathrm{co}}(Q_n))$ *and $(x^i_{a(n)}, \theta^i_n)$. In part (b) we can see that (apart from sign changes) the cut normal to the LN cut can be determined by maximizing the linear function $u_n^\top \gamma_n + u_{n0}\gamma_{n0}$ over $\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n)$. As the solution is an extreme point of $\mathcal{R}_n^{i+1}(x^i_{a(n)}, \theta^i_n)$ (*green dot*), the corresponding LN cut is facet-defining.*

D Additional Computational Results

D.1 Additional Plots

Fig. 13 illustrates the iteration times for SDDiP, as well as the number of level Bundle method iterations required to solve the dual problems for each iteration of SDDiP.

D.2 The Chen-Luedtke Approach: Restricting the Dual Space

To accelerate the solution process, Chen and Luedtke [11] propose to restrict the feasible set of the normalized Lagrangian dual problem (11) to a small subset of valid multipliers (π_n, π_{n0}). More precisely, the idea is to restrict the multipliers π_n to the span of a set

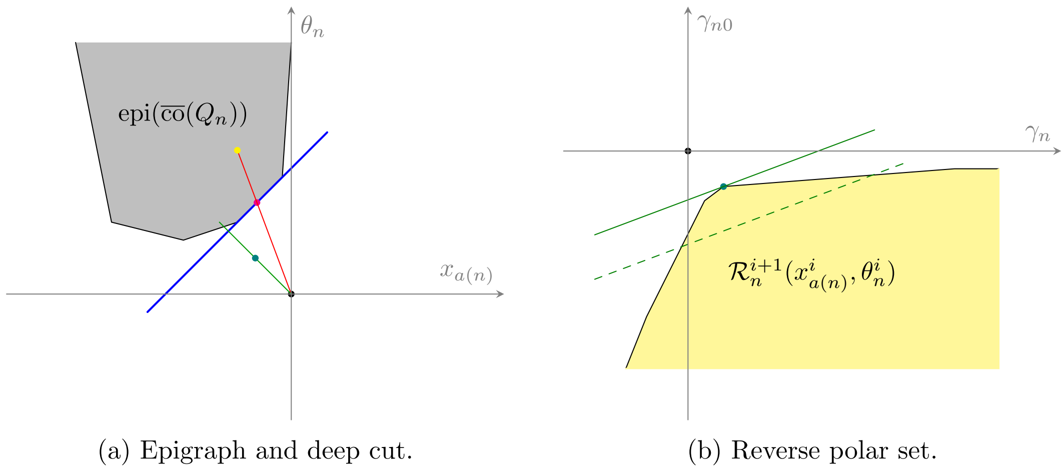


Figure 12: Illustration of LN cuts given some known core point.

of previously generated Benders cut coefficients $\widehat{\pi}_n^k, k = 1, \ldots, K$, for some predefined parameter K. That is, we introduce the constraint

$$\pi_n = \sum_{k=1}^{K} \gamma_{nk} \widehat{\pi}_n^k,$$

which also means that we add variables $\gamma_{nk}, k = 1, \ldots, K$, to the dual problem. While we lose tightness and convergence guarantees using this dual space restriction, the search space for the level Bundle method is significantly reduced, so that cuts can be generated faster. We refer to this as the `CL` approach.

In principle, the `CL` approach can be combined with any of the normalization techniques discussed in this paper. Additionally, it allows for an alternative normalization [11]. Instead of using $g(\pi_n, \pi_{n0}) = \|\pi_n, \pi_{n0}\|_1$, we may as well use some normalization function $g(\pi_n, \pi_{n0}, \gamma_n) = \|\gamma_n, \pi_{n0}\|_1$. This choice should lead to solutions with sparse γ. In the following, we denote this approach by `CL`-γ.

We perform some computational experiments, again considering 20 iterations of only `SB` cuts, and afterwards generating `SB` and Lagrangian cuts in each iteration. However, for the generation of Lagrangian cuts we apply a dual space restriction with parameter $K = 20$. The results are depicted in Fig. 14.

The number of Lagrangian iterations and the time per iteration are reduced significantly. Moreover, compared to only using `SB` or Lagrangian cuts, much better lower bounds are obtained in the same time. Interestingly, the lower bounds are even better *per iteration* than without dual space restriction, similar to what we observed for the combination with `SB` in Sect. 4.3. This illustrates that the quality of cuts does not only depend on tightness or depth, but also on the incumbents which they induce in the upcoming stages and iterations. The chosen normalization approach seems not to be decisive in this setting.

D.3 Detailed Results for `CLSP-Bin`

The full computational results for our experiments of CLSP with state binarization are depicted in Tables 3-5. The table columns contain the number of stages, the used cut

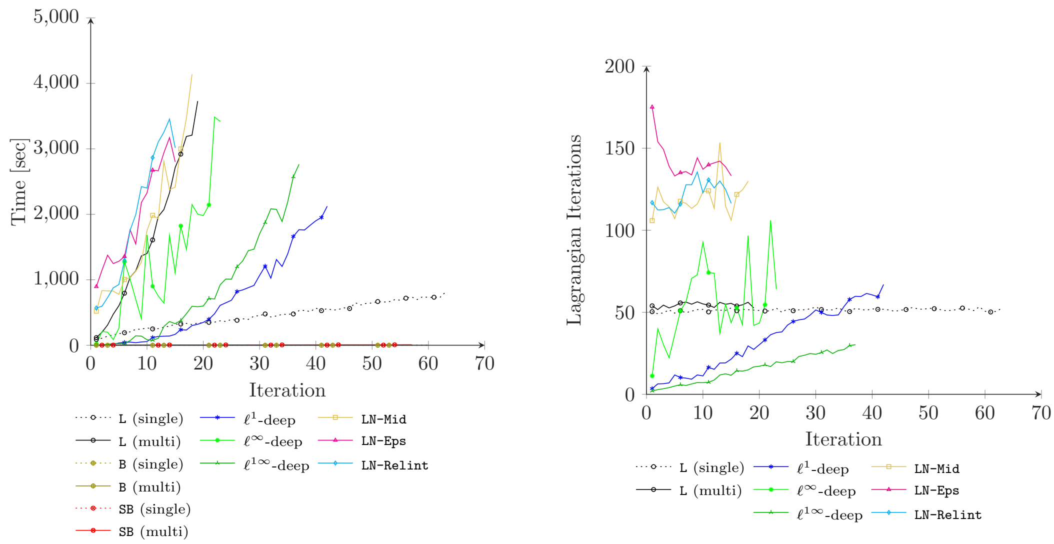


Figure 13: Different analyses for experiments on CLSP-Bin and $T = 16$.

Note. LEFT: Time per iteration of SDDiP. RIGHT: Iterations required to solve Lagrangian dual over iterations of SDDiP.

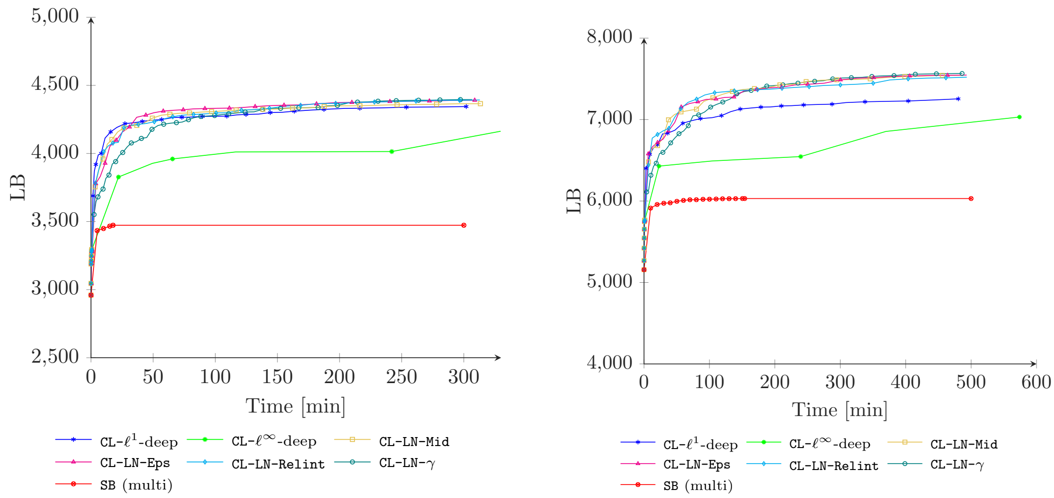


Figure 14: Lower bound development over time for experiments on CLSP-Bin using the CL approach.

Note. LEFT: $T = 10$. RIGHT: $T = 16$. 20 iterations with SB and then SB and Lagrangian cut in each iteration with dual space restriction for $K = 20$.

generation approach, the best lower bound obtained by SDDiP, a simulated statistical upper bound computed after termination of SDDiP (we report the upper limit of the computed confidence interval), the number of iterations, the time in seconds, the average time per iteration and the average number of iterations required in the level bundle method to solve the Lagrangian dual per iteration. We should note that in some cases, the simulation did not yield an upper bound estimate due to numerical issues.

D.4 Detailed Results for `CLSP`

For our tests of `CLSP` without state binarization, the full results are stated in Table 6.

D.5 Detailed Results for `CFLP`

For our tests of `CFLP`, the full results are stated in Table 7.

Table 3: SDDiP results for `CLSP-Bin` for $T = 4$, $T = 6$ and $T = 10$.

Method	Best LB	Stat. UB	Gap [%]	# Iter.	Time [s]	Time/It [s]	Lag-It/It	Unb [%]
				$\boldsymbol{T = 4}$ (3 hours)				
Det. Equiv.	1503.0	1542.3	3					
`B` (single)	803.2	1,595.5	50	85	6	0	-	-
`B` (multi)	804.5	1,602.6	50	59	5	0	-	-
`SB` (single)	1,155.8	**1,572.9**	27	32	16	1	-	-
`SB` (multi)	1,187.1	1,608.7	26	73	49	1	-	-
`L` (single)	681.8	1,766.4	61	109	10,800	99	49	-
`L` (multi)	702.6	1,736.6	60	27	10,908	404	51	-
ℓ^1-deep	1,478.3	1,582.5	7	39	11,196	272	67	-
$\ell^{1\infty}$-deep	1,462.9	1,649.4	11	38	10,944	288	63	-
ℓ^∞-deep	1,247.4	1,644.7	24	15	11,808	787	96	-
`LN-Mid`	1,496.6	1,579.0	**5**	36	11,304	314	82	0
`LN-Eps`	1,497.0	1,615.0	7	27	11,124	412	117	0
`LN-Relint`	**1,500.9**	1,623.3	8	33	11,052	335	85	0
				$\boldsymbol{T = 6}$ (4 hours)				
`B` (single)	1,354.5	2,854.2	53	479	69	0	-	-
`B` (multi)	1,355.0	2,889.4	53	165	37	0	-	-
`SB` (single)	2,077.9	**2,825.1**	26	46	41	1	-	-
`SB` (multi)	2,093.4	2,838.7	26	251	517	2	-	-
`L` (single)	669.7	3,095.0	78	90	14,580	162	50	-
`L` (multi)	682.5	3,111.7	78	24	14,688	612	130	-
ℓ^1-deep	2,414.1	2,975.4	19	34	14,508	427	62	-
$\ell^{1\infty}$-deep	2,375.8	2,984.5	20	35	15,012	429	59	-
ℓ^∞-deep	1,456.9	3,342.2	56	17	14,940	879	72	-
`LN-Mid`	**2,510.3**	2,962.3	15	24	14,688	612	101	0
`LN-Eps`	2,466.0	2,880.0	**14**	20	14,976	731	138	0
`LN-Relint`	2,483.7	2,915.6	15	22	14,976	681	121	0
				$\boldsymbol{T = 10}$ (5 hours)				
`B` (single)	2,165.3	5,120.8	58	136	26	0	-	-
`B` (multi)	2,183.6	5,168.8	58	334	197	0	-	-
`SB` (single)	3,377.1	4,949.8	32	89	122	1	-	-
`SB` (multi)	3,472.9	**4,942.9**	30	206	1,066	5	-	-
`L` (single)	616.4	5,466.6	89	69	18,144	263	51	-
`L` (multi)	682.4	5,443.1	88	20	18,720	936	54	-
ℓ^1-deep	3,782.5	-	-	38	19,260	507	47	-
$\ell^{1\infty}$-deep	3,862.7	-	-	37	18,504	500	47	-
ℓ^∞-deep	1,584.7	5,894.1	73	18	19,872	1,104	69	-
`LN-Mid`	**4,188.0**	5,163.4	**19**	18	19,188	1,066	113	0
`LN-Eps`	3,910.9	5,329.2	27	16	18,396	1,150	140	0
`LN-Relint`	4,121.1	5,185.6	21	16	18,828	1,177	120	0
				$\boldsymbol{T = 16}$ (8 hours)				
`B` (single)	3,917.5	9,012.6	57	224	74	0	-	-
`B` (multi)	3,937.2	9,008.2	56	254	274	1	-	-
`SB` (single)	5,913.7	**8,673.6**	32	194	493	3	-	-
`SB` (multi)	6,030.2	8,676.8	31	585	9,252	16	-	-
`L` (single)	607.4	9,524.5	94	63	28,836	458	51	-
`L` (multi)	651.6	9,444.7	93	19	31,176	1,641	55	-
ℓ^1-deep	3,547.5	9,756.3	64	42	28,944	689	34	-
$\ell^{1\infty}$-deep	3,353.0	9,997.3	67	37	31,248	845	16	-
ℓ^∞-deep	2,424.5	10,077.4	76	23	29,412	1,279	55	-
`LN-Mid`	6,910.7	9,125.9	24	18	32,112	1,784	119	0
`LN-Eps`	6,067.6	9,164.6	34	15	29,376	1,958	143	0
`LN-Relint`	**6,945.3**	8,984.5	**23**	15	29,268	1,951	122	0

Table 4: SDDiP results for `CLSP-Bin` using Lagrangian cuts combined with SB cuts.

Method	Best LB	Stat. UB	Gap [%]	# Iter.	Time [s]	Time/Iter [s]	Lag-Iter/Iter	Unb [%]
				$\boldsymbol{T=10}$ (5 hours)				
`SB` + `L` (single)	3,475.5	**5,003.9**	31	132	18,072	136	50	-
`SB` + `L` (multi)	3,483.0	5,014.4	31	51	18,360	360	51	-
`SB` + ℓ^1-deep	3,804.1	5,161.6	26	35	19,656	562	88	-
`SB` + $\ell^{1\infty}$-deep	3,914.2	5,233.7	25	36	19,656	546	83	-
`SB` + ℓ^∞-deep	3,856.5	5,290.3	27	25	22,644	906	150	-
`SB` + `LN-Mid`	**4,121.9**	5,121.8	**20**	31	19,296	623	115	0
`SB` + `LN-Eps`	4,091.1	5,092.3	**20**	31	20,448	660	140	0
`SB` + `LN-Relint`	4,041.7	5,157.5	22	31	19,800	639	119	0
				$\boldsymbol{T=16}$ (8 hours)				
`SB` + `L` (single)	5,907.1	**8,728.4**	32	136	28,944	213	47	-
`SB` + `L` (multi)	5,922.0	8,750.8	32	52	30,096	579	48	-
`SB` + ℓ^1-deep	6,477.5	9,078.4	28	33	29,916	907	90	-
`SB` + $\ell^{1\infty}$-deep	6,417.4	9,024.6	29	34	30,708	903	87	-
`SB` + ℓ^∞-deep	6,220.9	8,968.8	31	24	29,628	1235	164	-
`SB` + `LN-Mid`	6,615.9	9,003.1	27	29	31,140	1,074	126	0
`SB` + `LN-Eps`	6,705.0	8,947.8	**25**	29	30,528	1,053	146	0
`SB` + `LN-Relint`	**6,832.1**	9,071.4	**25**	29	31,464	1,085	135	0

Table 5: SDDiP results for `CLSP-Bin` with $T = 10$ and $T = 16$ using the `CL` approach.

Method	Best LB	Stat. UB	Gap [%]	# Iter.	Time [s]	Time/Iter [s]	Lag-Iter/Iter	Unb [%]
				$\boldsymbol{T=10}$ (5 hours)				
`CL`-γ	**4,395.5**	5,224.4	16	74	18,396	249	17	-
`CL`-ℓ^1-deep	4,344.5	-	-	53	18,108	341	41	-
`CL`-ℓ^∞-deep	4,162.8	5,270.9	21	26	19,764	760	278	-
`CL-LN-Mid`	4,365.7	5.259.2	17	53	18,792	355	36	0
`CL-LN-Eps`	4,390.9	**5,185.3**	**15**	51	18,540	364	39	0
`CL-LN-Relint`	4,392.7	5,203.3	16	52	18,756	361	34	0
				$\boldsymbol{T=16}$ (8 hours)				
`CL`-γ	**7,565.5**	9,067.7	**17**	65	29,160	449	18	-
`CL`-ℓ^1-deep	7,254.2	9,095.4	20	49	29,160	589	42	-
`CL`-ℓ^∞-deep	7,031.0	9,117.4	23	25	34,452	1378	320	-
`CL-LN-Mid`	7,543.1	**9,048.1**	**17**	48	29,268	610	38	0
`CL-LN-Eps`	7,545.7	9,050.4	**17**	48	29,556	616	39	0
`CL-LN-Relint`	7,520.3	9,050.4	**17**	48	29,592	617	38	0

Table 6: SDDiP results for `CLSP` with 10 state variables and no binary approximation.

Method	Best LB	Stat. UB	Gap [%]	# Iter.	Time [s]	Time/Iter [s]	Lag-Iter/Iter	Unb [%]
$T = 16$ (3 hours), **One type of cut**								
`B` (single)	15,190	29,700	49	193	42	0	-	-
`B` (multi)	15,199	29,747	49	207	140	1	-	-
`SB` (single)	20,373	29,600	31	111	189	2	-	-
`SB` (multi)	21,045	**29,343**	28	281	2,459	9	-	-
`L` (single)	14,440	32,429	56	93	10,908	117	43	-
`L` (multi)	19,689	31,468	37	30	11,268	376	45	-
ℓ^1-deep	23,678	30,064	21	36	10,836	301	32	-
$\ell^{1\infty}$-deep	**23,982**	30,026	**20**	34	11,340	333	34	-
ℓ^∞-deep	23,610	30,333	22	30	11,016	367	40	-
`LN-Mid`	22,336	30,041	26	21	11,412	543	157	0
`LN-Eps`	22,296	30,340	27	25	11,304	452	64	0
`LN-Relint`	22,753	30,393	25	19	11,304	595	99	3
`LN-Conv`(50)	22,994	30,148	24	19	11,808	622	101	0
`LN-Conv`(75)	23,285	30,183	23	20	11,556	578	90	0
`LN-Conv`(90)	23,416	30,127	22	21	11,088	528	79	0
`LN-Conv`(99)	22,967	30,054	24	25	11,664	467	63	0
$T = 16$ (3 hours), **Combination with `SB`**								
`SB` + `L` (single)	21,184	30,022	29	100	10,980	110	42	-
`SB` + `L` (multi)	22,431	30,257	26	39	11,304	290	44	-
`SB` + ℓ^1-deep	**23,443**	30,248	23	37	11,448	309	51	-
`SB` + $\ell^{1\infty}$-deep	23,331	30,307	23	39	11,304	306	49	-
`SB` + ℓ^∞-deep	23,424	30,062	**22**	36	11,700	300	45	-
`SB` + `LN-Mid`	23,036	30,161	24	31	11,304	365	110	0
`SB` + `LN-Eps`	23,195	30,086	23	34	11,160	328	67	0
`SB` + `LN-Relint`	23,038	30,371	24	31	11,340	366	113	0
`SB` + `LN-Conv`(50)	22,854	30,020	24	32	12,132	379	111	0
`SB` + `LN-Conv`(75)	22,867	30,131	24	32	10,836	339	96	0
`SB` + `LN-Conv`(90)	23,176	**29,957**	23	33	10,800	327	80	0
`SB` + `LN-Conv`(99)	23,237	30,399	24	35	11,952	342	68	0

Table 7: SDDiP results for `CFLP` (4 hours). Objective values scaled by 10^{-3}.

Method	Best LB	Stat. UB	Gap [%]	# Iter.	Time [s]	Time/Iter [s]	Lag-Iter/Iter	Unb [%]
$\boldsymbol{T = 16}$ (3 hours), **One type of cut**								
`B` (single)	59,744.4	78,498.9	24	592	14,436	24	-	-
`B` (multi)	64,853.1	-	-	355	14,436	41	-	
`SB` (single)	59,859.6	79,264.9	24	95	14,544	153	-	-
`SB` (multi)	63,127.0	78,737.6	20	90	14,616	162	-	-
`L` (single)	62,752.3	76,623.2	18	51	15,012	294	51	-
`L` (multi)	63,029.1	76,604.9	18	35	14,868	425	35	-
ℓ^1-deep	62,221.1	76,370.7	19	37	14,976	405	37	-
$\ell^{1\infty}$-deep	61,919.9	76,267.9	19	23	14,832	645	23	-
ℓ^∞-deep	62,080.0	76,410.9	19	27	14,760	547	27	-
`LN-Mid`	74,228.0	75,914.5	2	25	14,832	593	25	85
`LN-Eps`	6,337.6	-	-	17	14,760	868	17	2
`LN-Relint`	67,180.2	77,891.9	14	32	14,688	459	32	99
`LN-Conv`(50)	69,851.1	-	-	7	17,136	2,448	7	0
`LN-Conv`(75)	**75,199.1**	**75,451.3**	**0**	7	16,740	2,391	7	0
`LN-Conv`(90)	72,604.5	76,269.8	5	9	15,552	1,728	9	0
`LN-Conv`(99)	5,579.4	-	-	15	15,012	1,001	15	0
$\boldsymbol{T = 16}$ (3 hours), **Combination with `SB`**								
`SB` + `L` (single)	63,444.8	77,722.2	18	49	14,580	298	49	-
`SB` + `L` (multi)	64,977.7	-		47	14,904	317	47	-
`SB` + ℓ^1-deep	64,664.6	-		36	15,048	418	36	-
`SB` + $\ell^{1\infty}$-deep	64,557.9	-		35	14,904	426	35	-
`SB` + ℓ^∞-deep	64,743.7	-		37	15,228	412	37	-
`SB` + `LN-Mid`	74,224.6	-		35	15,156	433	35	88
`SB` + `LN-Eps`	59,572.8	77,320.9	23	25	14,472	579	25	2
`SB` + `LN-Relint`	66,898.8	77,849.9	14	46	15,084	328	46	99
`SB` + `LN-Conv`(50)	**75,172.8**	**75,443.6**	**0**	24	17,856	744	24	0
`SB` + `LN-Conv`(75)	75,163.2	-		24	14,940	623	24	0
`SB` + `LN-Conv`(90)	73,684.8	-		25	15,336	613	25	0
`SB` + `LN-Conv`(99)	59,536.0	77,384.8	23	25	17,208	688	25	0