



Solving multi-stage stochastic mixed integer linear programs by the dual dynamic programming approach

Zhihao Cen

**RESEARCH
REPORT**

N° 7868

January 26, 2012

Project-Teams Commands



Solving multi-stage stochastic mixed integer linear programs by the dual dynamic programming approach

Zhihao Cen * †

Project-Teams Commands

Research Report n° 7868 — January 26, 2012 — 29 pages

Abstract: We consider a model of medium-term commodity contracts management. Randomness takes place only in the prices on which the commodities are exchanged, whilst state variable is multi-dimensional, and decision variable is integer. In [7], we proposed an algorithm based on the quantization of random process and a dual dynamic programming type approach to solve the continuous relaxation problem. In this paper, we study the multi-stage stochastic mixed integer linear program (SMILP) and show the difficulty when using dual programming type algorithm. We propose an approach based on the cutting plane method combined with the algorithm in [7], which gives an upper and a lower bound of the optimal value and a sub-optimal integer solution. Finally, a numerical test on a real problem in energy market is provided.

Key-words: integer programming, stochastic programming, dual dynamic programming, cutting plane method.

* The research of this author was supported by and in collaboration with Total S.A. during his Ph.D. study.

† COMMANDS team, INRIA-Saclay and CMAP, Ecole Polytechnique, Palaiseau, France (zhihao.cen@polytechnique.edu).

**RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE**

Parc Orsay Université
4 rue Jacques Monod
91893 Orsay Cedex

Résolution d'un programmation linéaire en nombre entière stochastique de multi étapes par la programmation dynamique duale

Résumé : Nous considérons un modèle de gestion de contrats de matières premières à moyen terme. Dans ce problème de programmation stochastique, en nombre entier et en plusieurs étapes, les aléas ne portent que sur la dynamique de prix, la variable d'état est multi dimensionnelle, et la variable de contrôle est en nombre entier. Dans [7], nous avons proposé un algorithme pour le problème en relaxation continue, basé sur la méthode de quantification de processus aléatoire, et l'approche de programmation dynamique duale. Dans cet article, nous étudions le problème en nombre entier, et nous montrons la difficulté quand on utilise un algorithme de type programmation dynamique duale. Nous proposons un algorithme sur la méthode de coupe hyperplan combinée avec l'algorithme proposé dans [7], qui donne les bornes supérieures et inférieures et une solution sous-optimale en nombre entier. Enfin, un test numérique du problème réel sur le marché d'énergie est fourni.

Mots-clés : programmation en nombre entier, programmation stochastique, programmation dynamique duale, méthode de coupe hyperplan.

1 Introduction

In this paper, we study a class of problems in the following setting. First of all, we consider a uniform discrete time $t \in \{0, 1, \dots, T\}$, and a discrete time Markov process (ξ_t) in the probability space $(\Omega, (\mathcal{F}_t), \mathbf{P}; \Gamma_t = \{\xi_t^1, \dots, \xi_t^{N_t}\})$ with probability transition

$$p_t^{ij} = \mathbf{P} \left[\xi_t = \xi_{t+1}^j \mid \xi_t = \xi_t^i \right]. \quad (1)$$

The canonic filtration associated with (ξ_t) is denoted by $\mathcal{F}_t := \sigma(\xi_s, 0 \leq s \leq t)$.

If (ξ_t) has a continuous distribution $L^2(\Omega, (\mathcal{F}_t), \mathbf{P}; \mathbb{R}^d)$, we refer to Bally and Pagès [4], Heitsch et al. [14] and the references therein for various discretization methods and the error analysis. The method presented in this paper is still valid using the approximation formulation in our previous paper [7]. Therefore, without lose of generality, we assume here that (ξ_t) is a finite state Markov chain.

The stochastic dynamic decision problem has the following expression:

$$\begin{aligned} \inf \quad & \mathbb{E} \left[\sum_{t=0}^{T-1} c_t(\xi_t) u_t + g(\xi_T, x_T) \right] \\ \text{subject to} \quad & u_t : (\Omega, \mathcal{F}_t) \rightarrow \mathfrak{U}_t \cap \mathbb{Z}^n, \\ & x_{t+1} = x_t + A_t u_t, \\ & x_0 = 0, x_T \in \mathfrak{X}_T, \text{ almost surely;} \end{aligned} \quad (2)$$

where u_t is the decision variable, \mathfrak{U}_t is a compact nonempty convex polyhedral set in \mathbb{R}^n ; $x_t \in \mathbb{R}^m$ represents the state variable, \mathfrak{X}_T is the set of admissible final stage states variable x_T assumed to be a nonempty compact convex polyhedral set in \mathbb{R}^m ; $A_t \in \mathbb{R}^{m \times n}$ is the technology matrix; $c_t(\cdot) : \mathbb{R}^d \rightarrow \mathbb{R}^n$ is the running cost per unit, assumed to be Lipschitz; $g_t(\xi, x)$ is the final cost function, assumed to be convex lower semi-continuous in x , and Lipschitz in (ξ, x) .

Assumption 1.1. *The technology matrix A_t is integer.*

This assumption allows to guarantee that the state variable x_t is integer. It is satisfied in the application that motivated this work – LNG portfolio optimization problem (see Section 7).

In our paper [7], we focused on the way to solve numerically the continuous relaxation problem of (2), i.e.

$$\begin{aligned} \inf \quad & \mathbb{E} \left[\sum_{t=0}^{T-1} c_t(\xi_t) u_t + g(\xi_T, x_T) \right] \\ \text{subject to} \quad & u_t : (\Omega, \mathcal{F}_t) \rightarrow \mathfrak{U}_t, \\ & x_{t+1} = x_t + A_t u_t, \\ & x_0 = 0, x_T \in \mathfrak{X}_T, \text{ almost surely.} \end{aligned} \quad (3)$$

Furthermore, we have studied the sensitivity analysis of the continuous relaxation problem (3) in [8]. In this paper, we focus on the approximation method of the integer problem (2).

The multi-stage stochastic mixed integer linear programs (SMILP) are among the most challenging optimization problems because they combine two generally difficult classes of problems: stochastic programs and discrete optimization problems. Researchers have actively studied the properties and solutions approaches for such problems in the past decades. We refer to the surveys by Sen [23], Ahmed [1] and van der Vlerk [15] for some major results in this area and an annotated bibliography. Most papers of the literature have focused on two stage SMILP. Various decomposition algorithms for SMILP have been studied for different cases. We cite [2, 24, 26, 28, 29, 21, 20] for recently development, that most these combine branch and bound method to deal with integer constraint. However, multi-stage SMILP problems have been much less studied. Exact solution method is in general based on branch and bound type algorithm. Lulli and Sen [17] propose an algorithm based on branch and price method. Carøe and Schultz [10] propose decomposition algorithm with Lagrangian relaxation and branch and bound method for problems having general integer variables. Some heuristic methods have also been developed. Løkketangen and Woodruff [16] solve multi-stage SMILP of binary variables by applying a heuristic combining the progressive hedging algorithm with tabu search. Lulli and Sen [18] also develop a heuristic procedure based on successive approximation of scenario updating method.

In this article, we describe the difficult points in multi-stage SMILP, and an approximation method. Since our methodology relies on stage-wise decomposition and dual dynamic program type algorithm, the main idea here is to approximate the Bellman value function as well as possible. However, it is well known that once working with integer constraints, the Bellman value function does not have convexity anymore. Thus, our objective is to approximate the convex hull of the Bellman value function.

This article is organized as follows: we give the dynamic programming formula in Section 2, and discuss the feasible set in Section 3. We provide a first heuristic method in Section 4. Main discussion of difficulties and some possible improvement points by the cutting plan technique are given in Section 5 and in Section 6. Finally, a numerical test based on a real energy market problem is provided in Section 7.

2 Dynamic programming principle

The integer problem (2) has the desired structure for the dynamic programming principle to hold (see [27, Chapter 3]). In view of the Markov property of (x_t, ξ_t) , we can define the Bellman value at stage t , denoted by $Q^{int}(t, x_t, \xi_t)$:

$$\begin{aligned}
 Q^{int}(t, x_t, \xi_t) &:= \operatorname{ess\,inf} \mathbb{E} \left[\sum_{s=t}^{T-1} c_s(\xi_s) u_s + g(\xi_T, x_T) \middle| \mathcal{F}_t \right] \\
 \text{subject to } & u_s : (\Omega, \mathcal{F}_s) \rightarrow \mathfrak{U}_s \cap \mathbb{Z}^n, \quad s = t, \dots, T-1 \quad (4) \\
 & x_{s+1} = x_s + A_s u_s, \\
 & x_T \in \mathfrak{X}_T \quad \text{almost surely.}
 \end{aligned}$$

The following *dynamic programming principle* is easily established:

$$\begin{aligned} Q^{int}(t, x_t, \xi_t) &= \text{essinf} \quad c_t(\xi_t)u_t + Q^{int}(t+1, x_{t+1}, \xi_t) \\ \text{subject to} \quad &u_t \in \mathfrak{U}_t \cap \mathbb{Z}^n, \\ &x_{t+1} = x_t + A_t u_t; \end{aligned} \quad (5)$$

where $Q^{int}(\cdot, \cdot, \cdot)$ is the conditional expectation of the value at time $t+1$ obtained on state value x_{t+1} :

$$\begin{aligned} Q^{int}(t+1, x_{t+1}, \xi_t) &:= \mathbb{E} [Q^{int}(t+1, x_{t+1}, \xi_{t+1}) | \mathcal{F}_t] \\ &= \mathbb{E} [Q^{int}(t+1, x_{t+1}, \xi_{t+1}) | x_{t+1}, \xi_t]; \end{aligned} \quad (6)$$

where the second equality comes from the fact that x_{t+1} is \mathcal{F}_t measurable.

Finally the Bellman value function at final stage is:

$$Q^{int}(T, x_T, \xi_T) = \begin{cases} g(\xi_T, x_T) & \text{if } x_T \in \mathfrak{X}_T, \\ +\infty & \text{otherwise.} \end{cases} \quad (7)$$

Lemma 2.1. *The Bellman value function $Q^{int}(t, x_t, \xi_t)$ and its conditional expectation $Q^{int}(t, x_t, \xi_{t-1})$ are lower semi-continuous functions in x_t .*

Proof. Blair and Jeroslow [6] have proved this property for the deterministic case. In our framework, random variable appears only in the objective function, and the conditional expectation is a continuous, increasing operator, thus the lower semi-continuous property is kept. The lemma can be deduced by backward induction. \square

Similarly, we can define the Bellman value function $Q^{cont}(t, x_t, \xi_t)$ and its conditional expectation $Q^{cont}(t, x_t, \xi_{t-1})$ for the continuous relaxation problem (3). For the mathematical property of $Q^{cont}(t, x_t, \xi_t)$ and $Q^{cont}(t, x_t, \xi_{t-1})$, readers can refer to Bonnans et al. [7].

Compared to the continuous framework, we lose the continuity and the convexity in the integer case. Indeed, $Q^{int}(t, x_t, \xi_t)$ and $Q^{int}(t, x_t, \xi_t)$ are in general non-convex and discontinuous in x_t (see e.g. Schultz et al. [22]). We can illustrate this fact with the following example.

Example 2.2. We use this example to explain the difficult points by dual approximation and show the possible improvement by the cutting plane method in multi-stage SMILP all through the paper. This example can be viewed as a subproblem of dynamic programming.

$$\begin{aligned} Q^{int}(x_0) &= \text{essinf} \quad \mathbb{E} [\xi u_0 + \vartheta], \\ \text{subject to} \quad &u_0 \in [0, 3] \cap \mathbb{Z}, \\ &x_1 = x_0 + 2u_0, \quad x_1 \leq 6, \\ &\vartheta \geq \frac{x_1 - 3}{2}, \quad \vartheta \geq -\frac{x_1 - 3}{2}; \end{aligned} \quad (8)$$

where $\xi \in \{-2, 0.5\}$ that each takes probability 0.5. We denote by $Q^{int}(x_0, \xi)$ the Bellman value function for each realization of ξ . To simplify the problem, for each realization of ξ , the expectation of future costs approximated by optimality cuts θ is the same. The functions $Q^{int}(x_0, \xi)$ and $Q^{int}(x_0)$ are defined for $x_0 \in [0, 3]$, but we are more interested in the integer points $x_0 \in \{0, 1, 2, 3\}$.

The functions $Q^{int}(x_0, \xi)$ and $Q^{int}(x_0)$ are given in Figure 1, both are non-convex and discontinuous in x_0 .

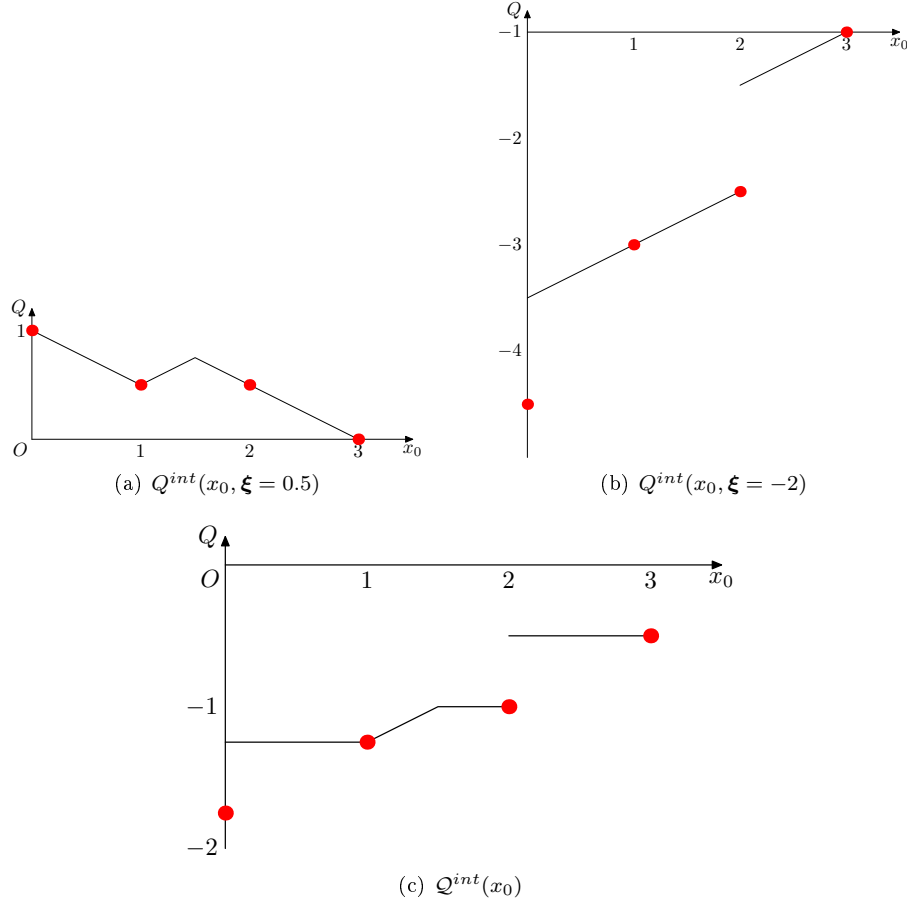


Figure 1: the Bellman value function $Q^{int}(x_0, \xi)$ and its conditional expectation $Q^{int}(x_0)$.

3 Feasible set

The final state constraints of problem (2) $x_T \in \mathfrak{X}_T$ yields to an implicit constraint on decision variable u_t and state variable x_t at each stage. We immediately specify feasible set of the integer problem (2).

We define the local feasible set of decision variable u_t , which depends on the state variable x_t :

$$\mathfrak{U}_t^{int,ad}(x_t) := \left\{ u_t \mid x_T = x_t + \sum_{s=t}^{T-1} A_s u_s \in \mathfrak{X}_T, u_s \in \mathfrak{U}_s \cap \mathbb{Z}^n \right\}. \quad (9)$$

Note that in the definition of $\mathfrak{U}_t^{int,ad}$ we require that all the decision variables (u_t) from time t be integer.

Furthermore, we need to add some constraint on x_t to ensure that $\mathfrak{U}_t^{int,ad}(x_t)$ is non-empty, and that x_t can be reached from time 0. We then define the

feasible set $\mathfrak{X}_t^{int,ad}$:

$$\begin{aligned} \mathfrak{X}_t^{int,ad} &:= \left\{ x_t \mid \mathfrak{U}_t^{int,ad}(x_t) \neq \emptyset, x_t = \sum_{s=0}^{t-1} A_s u_s, u_s \in \mathfrak{U}_s \cap \mathbb{Z}^n \right\} \\ &= \left\{ x_t \mid x_t = \sum_{s=0}^{t-1} A_s u_s, x_t + \sum_{s=t}^{T-1} A_s u_s \in \mathfrak{X}_T, u_s \in \mathfrak{U}_s \cap \mathbb{Z}^n \right\}. \end{aligned} \quad (10)$$

In addition, we introduce a notation of the feasible set in the space (x, u) :

$$\begin{aligned} (\mathfrak{X}, \mathfrak{U})_t^{int,ad} &:= \{x_t, u_t \mid x_0 = 0, x_{s+1} = x_s + A_s u_s, \\ &\quad x_T \in \mathfrak{X}_T, u_s \in \mathfrak{U}_s \cap \mathbb{Z}^n, s = 0, \dots, T-1\}. \end{aligned} \quad (11)$$

Due to Assumption 1.1, any $x_t \in \mathfrak{X}_t^{int,ad}$ and $(x_t, u_t) \in (\mathfrak{X}, \mathfrak{U})_t^{int,ad}$ are integer.

We denote by $\overline{\text{conv}} \mathfrak{U}_t^{int,ad}(x_t)$ (resp. $\overline{\text{conv}} \mathfrak{X}_t^{int,ad}$, $\overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_t^{int,ad}$) the closure of convex hull of $\mathfrak{U}_t^{int,ad}(x_t)$ (resp. $\mathfrak{X}_t^{int,ad}$, $(\mathfrak{X}, \mathfrak{U})_t^{int,ad}$). Note that in our study $\mathfrak{U}_t^{int,ad}$, $\mathfrak{X}_t^{int,ad}$ and $(\mathfrak{X}, \mathfrak{U})_t^{int,ad}$ are bounded, therefore we can replace the closure of convex hull by the convex hull itself since they are equal. However, we will keep on using the notation $\overline{\text{conv}}$.

Finally, we denote by

$$\overline{\mathfrak{U}}_t^{int,ad}(\hat{x}_t) := \{u_t \mid (\hat{x}_t, u_t) \in \overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_t^{int,ad}\}. \quad (12)$$

Remark 3.1. Since problem (2) has no randomness in the constraints, the various feasible sets introduced in this section do not depend on ξ_t . This fact is implied directly by their definitions (9) (10)(11)(12).

Note that we can always restrain ourselves to the case where extreme points of \mathfrak{X}_T are integer due to Assumption 1.1. We have the following lemma.

Lemma 3.2. *If $\hat{x}_t \in \overline{\text{conv}} \mathfrak{X}_t^{int,ad}$, then*

$$\overline{\text{conv}} \mathfrak{U}_t^{int,ad}(\hat{x}_t) \subset \overline{\mathfrak{U}}_t^{int,ad}(\hat{x}_t). \quad (13)$$

Furthermore, if \hat{x}_t is an extreme point of $\overline{\text{conv}} \mathfrak{X}_t^{int,ad}$, the inclusion (13) will become an equality.

Proof. If $u_t \in \overline{\text{conv}} \mathfrak{U}_t^{int,ad}(\hat{x}_t)$, then there exists $\{\lambda_u^i > 0, u_t^i \in \mathfrak{U}_t^{int,ad}(\hat{x}_t), i \in I\}$ such that, $\sum_{i \in I} \lambda_u^i = 1$ and $u_t = \sum_{i \in I} \lambda_u^i u_t^i$. By the definition of $\mathfrak{U}_t^{int,ad}(\hat{x}_t)$, there exists $(u_s^i) \in \mathfrak{U}_s \cap \mathbb{Z}^n, t \leq s \leq T-1$ such that $\hat{x}_t + \sum_{s=t}^{T-1} A_s u_s^i \in \mathfrak{X}_T$. If $\hat{x}_t \in \overline{\text{conv}} \mathfrak{X}_t^{int,ad}$, then there exists $\{\lambda_x^j > 0, x_t^j \in \mathfrak{X}_t^{int,ad}, j \in J\}$ such that, $\sum_{j \in J} \lambda_x^j = 1$, $\hat{x}_t = \sum_{j \in J} \lambda_x^j x_t^j$, and there is no other integer point in $\text{int}(\overline{\text{conv}} \{x_t^j, j \in J\})$, where $\text{int}(X)$ represents interior of X . Following the definition of $\mathfrak{X}_t^{int,ad}$, $\exists(\tilde{u}_s^j) \in \mathfrak{U}_s \cap \mathbb{Z}^n, 0 \leq s \leq t-1$ such that $x_t^j = \sum_{s=0}^{t-1} A_s \tilde{u}_s^j$. Let $x_T^{i,j} = x_t^j + \sum_{s=t}^{T-1} A_s u_s^i$, then

$$\sum_{j \in J} \lambda_x^j x_T^{i,j} = \sum_{j \in J} \lambda_x^j \left(x_t^j + \sum_{s=t}^{T-1} A_s u_s^i \right) = \hat{x}_t + \sum_{s=t}^{T-1} A_s u_s^i \in X_T.$$

Indeed, we have that $\{x_T^{i,j}, \forall j \in J\}$ are integer, and that there is no integer point in $\text{int}(\overline{\text{conv}}\{x_T^{i,j}, j \in J\})$. Owing to the property of \mathfrak{X}_T that \mathfrak{X}_T is convex, compact and its extreme points are integer, we have $x_T^{i,j} \in \mathfrak{X}_T, \forall j \in J$. This result is valid for all $i \in I$. Thus, $(x_t^j, u_t^i) \in \mathfrak{X}_t^{\text{int},ad}, \forall i \in I, j \in J$. Hence, by the convexity, $(\hat{x}_t, u_t) \in \overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_t^{\text{int},ad}$, and $u_t \in \overline{\mathfrak{U}}_t^{\text{int},ad}(\hat{x}_t)$.

Assume that $u_t \in \overline{\mathfrak{U}}_t^{\text{int},ad}(\hat{x}_t)$, or equivalently $(\hat{x}_t, u_t) \in \overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_t^{\text{int},ad}$, then there exist $\{\lambda^i \geq 0, (x_t^i, u_t^i) \in (\mathfrak{X}, \mathfrak{U})_t^{\text{int},ad}, i \in I\}$ such that $\sum_{i \in I} \lambda^i = 1$, $\hat{x}_t = \sum_{i \in I} \lambda^i x_t^i$, and $u_t = \sum_{i \in I} \lambda^i u_t^i$. Since \hat{x}_t is an extreme point of $\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}$, we have that $\forall i, x_t^i = \hat{x}_t$. The definition of $(\mathfrak{X}, \mathfrak{U})_t^{\text{int},ad}$ yields to $u_t^i \in \mathfrak{U}_t^{\text{int},ad}(\hat{x}_t)$. Hence, $u_t \in \overline{\text{conv}} \mathfrak{U}_t^{\text{int},ad}(\hat{x}_t)$. \square

Then, we have immediately the following corollary.

Corollary 3.3. *If $\mathfrak{X}_t^{\text{int},ad} \subset \mathbb{B}^m$, where $\mathbb{B} = \{0, 1\}$ represents the binary set, then $\overline{\mathfrak{U}}_t^{\text{int},ad}(x_t) = \overline{\text{conv}} \mathfrak{U}_t^{\text{int},ad}(x_t)$.*

Proof. Following Assumption 1.1 and forward induction, x_t is integer, and is therefore an extreme point of $\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}$. The conclusion follows. \square

In general, the inclusion of Lemma 3.2 is strict. Let us illustrate it in the following example.

Example 3.4 (example 2.2 continued). We illustrate the feasible set of the example 2.2 in Figure 2. We can observe the strict inclusion in (13) in this example

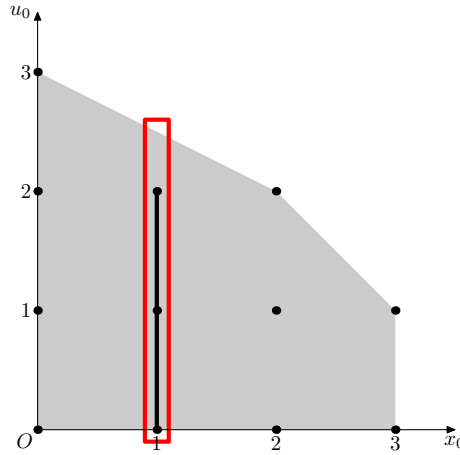


Figure 2: The black dots represent the integer solution $(\mathfrak{X}, \mathfrak{U})_0^{\text{int},ad}$, and the gray zone is the convex hull $\overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_0^{\text{int},ad}$. When $\hat{x}_0 = 1$, the black line is $\overline{\text{conv}} \mathfrak{U}_0^{\text{int},ad}(\hat{x}_0)$, whilst the red block is $\overline{\mathfrak{U}}_0^{\text{int},ad}(\hat{x}_0)$.

that when $\hat{x}_0 = 1$, we have $\overline{\text{conv}} \mathfrak{U}_0^{\text{int},ad}(x_0) = [0, 2] \subset \overline{\mathfrak{U}}_0^{\text{int},ad}(x_0) = [0, 2.5]$.

Finally, we require the following assumption to be satisfied to guarantee the existence of a solution.

Assumption 3.5. $\left(\sum_{t=0}^{T-1} A_t(\mathcal{U}_t)\right) \cap \mathbb{Z}^n \cap \mathfrak{X}_T \neq \emptyset$.

4 First heuristic method

Following the dynamic programming principle (5), if we are able to approximate $Q^{int}(t+1, x_{t+1}, \xi_t)$, then we can compute the optimal integer decision. The first idea is given by the following proposition [5, Chapter 8, Proposition 5].

Proposition 4.1. *Any optimality cut of $Q^{cont}(t, x_t, \xi_t)$ (resp. $Q^{cont}(t, x_t, \xi_{t-1})$) is a valid lower bound of $Q^{int}(t, x_t, \xi_t)$ and (resp. $Q^{int}(t, x_t, \xi_{t-1})$).*

Following the proposition, our first heuristic method aims to find a sub-optimal integer decision using the optimality cut of $Q^{cont}(t+1, x_{t+1}, \xi_t)$:

$$\begin{aligned} & \text{essinf} && c_t(\xi_t)u_t + \hat{v}(t+1, x_{t+1}, \xi_t) \\ & \text{subject to} && x_{t+1} = x_t + A_t u_t, \\ & \text{(feasibility cut)} && u_t \in \mathcal{U}_t^{int, ad}(x_t), \\ & \text{(optimality cut)} && \hat{v}(t+1, x_{t+1}, \xi_t) \geq x^* x_{t+1} - Q^{*, cont}(t+1, x^*, \xi_t), \forall x^*; \end{aligned} \tag{14}$$

where $Q^{*, cont}(t+1, x^*, \xi_t)$ is the Fenchel transformation of $Q^{cont}(t+1, x_{t+1}, \xi_t)$ in x_{t+1} .

However, the procedure (14) is only available for the forward pass of the SDDP approach since we cannot obtain any dual information from the MILP problem (14). Thus, the continuous relaxation gives us a lower bound of $Q^{int}(t+1, x_{t+1}, \xi_t)$ by some convex function, and then (14) provides a sub-optimal solution. This method has been mentioned in Birge and Louveaux [5, Chapter 8].

Example 4.2 (example 2.2 continued). We plot in Figure 3 $Q^{cont}(x_0, \xi)$ and $Q^{cont}(x_0)$ of example 2.2.

In practice, in the forward pass of the SDDP procedure. We simulate M_f samplings and compute a sub-optimal decision using (14). Then, we can obtain a statistic value \bar{v}^{int} , which is an upper bound of the integer problem (2). We take the lower bound of the continuous relaxation problem (3) as a lower bound of the integer problem (2). In general, the upper and lower bounds do not converge by this method. Then the remainder of the article focuses on reducing the gap between the two bounds.

5 Limitation of dual programming for multi-stage SMILP

Since $Q^{int}(t, x_t, \xi_t)$ (resp. $Q^{int}(t, x_t, \xi_{t-1})$) is in general non-convex and discontinuous in x_t , the best convex and lower semi-continuous function to approximate $Q^{int}(t, x_t, \xi_t)$ (resp. $Q^{int}(t, x_t, \xi_{t-1})$) from below is its convex hull, i.e. the greatest convex function majorized by $Q^{int}(t, x_t, \xi_t)$ (resp. $Q^{int}(t, x_t, \xi_{t-1})$), denoted by $\overline{\text{conv}} Q^{int}(t, x_t, \xi_t)$ (resp. $\overline{\text{conv}} Q^{int}(t, x_t, \xi_{t-1})$). Therefore, $\overline{\text{conv}} Q^{int}(t, x_t, \xi_t)$ and $\overline{\text{conv}} Q^{int}(t, x_t, \xi_{t-1})$ have by the Fenchel-Moreau theorem a dual representation: they can be represented by a supremum of linear functions (optimality cuts).

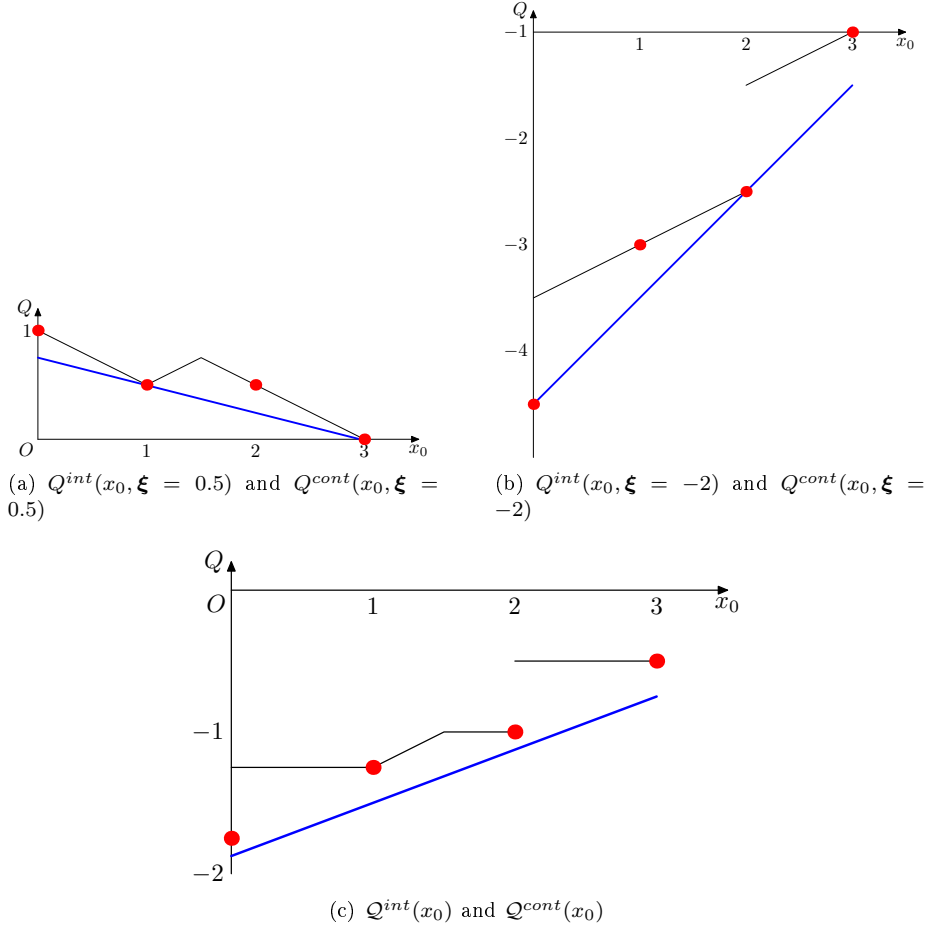


Figure 3: The blue line is $Q^{cont}(x_0, \xi)$ or $Q^{cont}(x_0)$.

Assume that we are able to compute $\overline{\text{conv}} Q^{int}(\tau + 1, x_{\tau+1}, \xi_{\tau+1})$ for some $0 \leq \tau \leq T - 1$, then for all $t \leq \tau$ the following approach gives an alternative of (14)

$$\begin{aligned}
 Q_{\tau}^{int,conv}(t, x_t, \xi_t) &:= \text{essinf} \quad c_t(\xi_t)u_t + \vartheta \\
 &\text{subject to} \quad u_t \in \mathcal{U}_t^{int,ad}(x_t), \\
 &\quad x_{t+1} = x_t + A_t u_t, \\
 &\quad \vartheta \geq \overline{Q}_{\tau}^{int,conv}(t + 1, x_{t+1}, \xi_t);
 \end{aligned} \tag{15}$$

where

$$\overline{Q}_{\tau}^{int,conv}(t, x_t, \xi_{t-1}) = \begin{cases} \mathbb{E} [\overline{\text{conv}} Q^{int}(\tau + 1, x_{\tau+1}, \xi_{\tau+1}) | \mathcal{F}_{\tau}] & \text{if } t = \tau, \\ \mathbb{E} [\overline{\text{conv}} Q_{\tau}^{int,conv}(t + 1, x_{t+1}, \xi_{t+1}) | \mathcal{F}_t] & \text{if } t < \tau. \end{cases} \tag{16}$$

Indeed, $\overline{Q}_{\tau}^{int,conv}(t, x_t, \xi_{t-1})$ is convex in x_t since the conditional expectation operator keeps the convexity of a function. This approach (14) allows us to

continue to use the SDDP approach, In addition, remark that by the definition of a convex hull, we have

$$\overline{\overline{Q}}_{\tau}^{int,conv}(t, x_t, \xi_{t-1}) \leq \overline{\text{conv}} \mathbb{E} [Q_{\tau}^{int,conv}(t, x_t, \xi_t) | \mathcal{F}_{t-1}]. \quad (17)$$

Even if we are able to compute $\overline{\text{conv}} Q^{int}(\tau, x_{\tau}, \xi_{\tau})$, problem (15) at stage $t = \tau$ does not return the exact optimal solution of (5). Moreover, we have for $t \leq \tau$:

$$Q^{cont}(t, x_t, \xi_t) \leq Q_{\tau}^{int,conv}(t, x_t, \xi_t) \leq Q^{int}(t, x_t, \xi_t), \quad (18)$$

and so

$$Q^{cont}(t, x_t, \xi_{t-1}) \leq \overline{\overline{Q}}_{\tau}^{int,conv}(t, x_t, \xi_{t-1}) \leq \overline{\text{conv}} Q^{int}(t, x_t, \xi_{t-1}); \quad (19)$$

and generally they do not coincide with each other. Therefore, (15) always returns a tighter lower bound of $Q^{int}(t, x_t, \xi_t)$ than (14).

One exception is when $\mathfrak{X}_t^{int,ad} \subset \mathbb{B}^m$. We have then the following proposition.

Proposition 5.1. *If for all $t \leq \tau$, $\mathfrak{X}_t^{int,ad} \subset \mathbb{B}^m$, then we have for all $x_t \in \mathfrak{X}_t^{int,ad}$:*

$$\begin{aligned} \overline{\overline{Q}}_{\tau}^{int,conv}(t+1, x_{t+1}, \xi_t) &= Q^{int}(t+1, x_{t+1}, \xi_t), \\ Q_{\tau}^{int,conv}(t, x_t, \xi_t) &= Q^{int}(t, x_t, \xi_t). \end{aligned}$$

Therefore, (15) is equivalent to (5).

Proof. The key point of $x_t \in \mathfrak{X}_t^{int,ad} \subset \mathbb{B}^m$ is that any integer solution x_t is an extreme point of $\mathfrak{X}_t^{int,ad}$.

At stage $t = \tau$, by definition of a convex hull, at extreme point of $\mathfrak{X}_t^{int,ad}$ we have

$$\overline{\text{conv}} Q^{int}(\tau+1, x_{\tau+1}, \xi_{\tau+1}) = Q^{int}(\tau+1, x_{\tau+1}, \xi_{\tau+1}).$$

Taking conditional expectation on both side, we get

$$\overline{\overline{Q}}_{\tau}^{int,conv}(\tau+1, x_{\tau+1}, \xi_{\tau}) Q^{int}(\tau+1, x_{\tau+1}, \xi_{\tau}).$$

Since in (15) we are only interested in $\overline{\overline{Q}}_{\tau}^{int,conv}(\tau+1, x_{\tau+1}, \xi_{\tau})$ where $x_{\tau+1}$ is integer, problems (15) and (5) coincide. Hence, $Q_{\tau}^{int,conv}(\tau, x_{\tau}, \xi_{\tau}) = Q^{int}(\tau, x_{\tau}, \xi_{\tau})$.

At stage $t < \tau$, by backward induction, using the same argument for $t = \tau$, the above result follows. \square

Example 5.2 (example 2.2 continued). We add the convex hulls in example 2.2 in Figure 4. Assume that in (8), the optimality cuts approximate exactly the convex hull $\overline{\text{conv}} Q(x_1, \xi)$. In this example, $\overline{\overline{Q}}_1^{int,conv}(x_0)$ provides some improvement than $Q^{cont}(x_0)$, but it does not coincide with $\overline{\text{conv}} Q^{int}(x_0)$.

Remark 5.3. Nevertheless, the convex hulls $\overline{\text{conv}} Q^{int}(t, x_t, \xi_t)$ and $\overline{\text{conv}} Q^{int}(t, x_t, \xi_{t-1})$ are very difficult to compute. In practice, we are able to compute

$$\overline{\text{conv}} Q^{int}(T, x_T, \xi_T) = g(\xi_T, x_T)$$

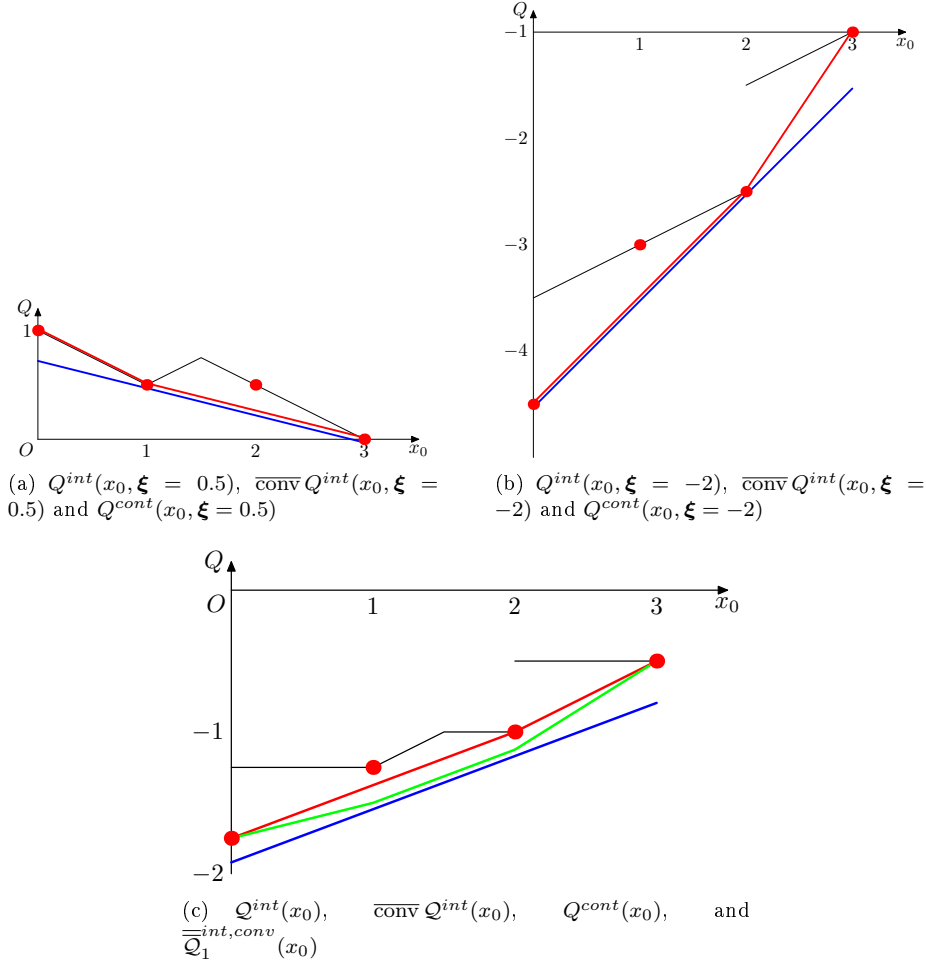


Figure 4: The blue line is Q^{cont} and Q^{cont} ; the red line is $\overline{\text{conv}} Q^{int}$ and $\overline{\text{conv}} Q^{int}$; whilst the green one is $\overline{Q}_1^{int,conv}$.

for the last stage T . Then, by using dual dynamic programming to solve the multi-stage SMILP problem (2), it is almost impossible to obtain the exact optimal integer solution and the exact value function $Q^{int}(t, x_t, \xi_t)$. The best approximated function for $Q^{int}(t, x_t, \xi_t)$ (resp. $Q^{int}(t, x_t, \xi_{t-1})$) is $Q_T^{int,conv}(t, x_t, \xi_t)$ (resp. $\overline{Q}_T^{int,conv}(t, x_t, \xi_{t-1})$). This is also remarked in Sen et al. [25].

Following (18) (19), the approximation quality decreases as the backward procedure goes on. Indeed, we have that

$$Q_{\tau_1}^{int,conv}(t, x_t, \xi_t) \leq Q_{\tau_2}^{int,conv}(t, x_t, \xi_t), \quad \tau_2 \leq \tau_1. \quad (20)$$

Unfortunately, $Q_T^{int,conv}(t, x_t, \xi_t)$ is the worst one among this class of approximation functions.

In the following section, we introduce the cutting plane method (valid in-

equality) to approximate $\overline{\text{conv}} Q(t, x_t, \xi_{t-1})$. The objective now is just to reduce the gap between $\overline{\text{conv}} Q^{int}(t, x_t, \xi_t)$ and $Q^{cont}(t, x_t, \xi_t)$ (resp. between $\overline{\text{conv}} Q^{int}(t, x_t, \xi_{t-1})$ and $Q^{cont}(t, x_t, \xi_{t-1})$) as much as possible, for instance by computing $\overline{\text{conv}} Q_T^{int,conv}(t, x_t, \xi_t)$ (resp. $\overline{Q}_T^{int,conv}(t, x_t, \xi_t)$).

6 Integer cutting plan technique

In order to obtain a tighter lower bound (better optimality cuts) for $\overline{\text{conv}} Q^{int}(t, x_t, \xi_t)$ and $\overline{\text{conv}} Q^{int}(t, x_t, \xi_{t-1})$, we will use the cutting plane technique. In other words, we are still interested in solving a continuous problem to compute dual value to build optimality cuts, since MILP problem does not give any dual information.

We introduce the Bellman value function by backward induction. Let us first approximate the final cost function $g(\xi_T, x_T)$ by some optimality cuts $O_T(\xi_T)$, where we denote by $Q^{cut}(T, x_T, \xi_T, O_T)$ the approximation function :

$$Q^{cut}(T, x_T, \xi_T, O_T) = \max \{ \theta : \theta \geq \lambda_T^x x_T + \lambda_T^0, (\lambda_T^x, \lambda_T^0) \in O_T(\xi_T) \}. \quad (21)$$

In general, we denote by $Q^{cut}(t, x_t, \xi_t, \mathcal{I}_{[t, T-1]}, O_t)$ ($Q^{cut}(t, x_t, \xi_t)$ for short) the Bellman value function by using the cutting plane method, which depends on the generated cutting planes $\{\mathcal{I}, \mathcal{O}\}$. Here, we separate the cutting planes into two types. One is the feasibility cut, which only depends on time step t :

$$\lambda_t^x x_t + \lambda_t^u u_t \leq \lambda_t^0, \quad (\lambda_t^x, \lambda_t^u, \lambda_t^0) \in \mathcal{I}_t. \quad (22)$$

We denote by $\mathcal{I}_{[t_1, t_2]} = \{\mathcal{I}_{t_1}, \dots, \mathcal{I}_{t_2}\}$, $t_1 \leq t_2$. The other type is the optimality cut which depends on one random variable value ξ_t

$$\theta \geq \lambda_t^x x_t + \lambda_t^0, \quad (\lambda_t^x, \lambda_t^0) \in O_t(\xi_t). \quad (23)$$

or

$$\vartheta \geq \bar{\lambda}_t^x x_{t+1} + \bar{\lambda}_t^u u_t + \bar{\lambda}_t^0, \quad (\bar{\lambda}_t^x, \bar{\lambda}_t^u, \bar{\lambda}_t^0) \in O_t(\xi_t). \quad (24)$$

We then introduce $\overline{Q}^{cut}(t, x_t, \xi_{t-1}, \mathcal{I}_{[t, T-1]}, O_{t-1})$ ($\overline{Q}^{cut}(t, x_t, \xi_{t-1})$ for short) the conditional expectation of $Q^{cut}(t, x_t, \xi_t, \mathcal{I}_{[t, T-1]}, O_t)$:

$$\begin{aligned} \overline{Q}^{cut}(t, x_t, \xi_{t-1}, \mathcal{I}_{[t, T-1]}, O_{t-1}) &:= \mathbb{E} [Q^{cut}(t, x_t, \xi_t, \mathcal{I}_{[t, T-1]}, O_t) | \mathcal{F}_{t-1}]; \\ &= \max \left\{ \vartheta : \vartheta \geq \bar{\lambda}_{t-1}^x x_t + \bar{\lambda}_{t-1}^0, (\bar{\lambda}_{t-1}^x, \bar{\lambda}_{t-1}^0) \in O_{t-1}(\xi_{t-1}) \right\}. \end{aligned} \quad (25)$$

where $\bar{\lambda}_{t-1}^x = \mathbb{E}[\lambda_t^x | \mathcal{F}_{t-1}]$ and $\bar{\lambda}_{t-1}^0 = \mathbb{E}[\lambda_t^0 | \mathcal{F}_{t-1}]$.

In addition, we denote by $Q^{int, cut}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, O_t)$ ($Q^{int, cut}(t, x_t, \xi_t)$ for short) the value function of the following MILP problem:

$$\begin{aligned} Q^{int, cut}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, O_t) &:= \text{essinf} \{ c_t(\xi_t) u_t + \vartheta : \\ &\text{s.t. } (u_t, x_{t+1}, \vartheta) \in \mathcal{A}(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, O_t) \}; \end{aligned} \quad (26)$$

where the feasible set $\mathcal{A}(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, O_t)$ ($\mathcal{A}(x_t, \xi_t)$ for short) is

$$\begin{aligned} \mathcal{A}(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, O_t) &:= \{(u_t, x_{t+1}, \vartheta) | u_t \in \mathcal{U}_t^{int, ad}(x_t), \\ &x_{t+1} = x_t + A_t u_t, \vartheta \geq \overline{Q}^{cut}(t+1, x_{t+1}, \xi_t, \mathcal{I}_{[t+1, T-1]}, O_t)\}. \end{aligned} \quad (27)$$

Since $Q^{int,cut}(t, x_t, \xi_t)$ is non-convex and discontinuous, we are more interesting in building

$$\overline{\text{conv}} Q^{int,cut}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t)$$

and then in approximating by a supremum of linear functions. Before giving the approximation formula to $\overline{\text{conv}} Q^{int,cut}(t, x_t, \xi_t)$, we have the following proposition to characterize $\overline{\text{conv}} Q^{int,cut}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t)$. For technical reason, we need some notations. Let us define

$$\begin{aligned} \overline{\overline{\mathcal{A}}}(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) := & \left\{ (u_t, x_{t+1}, \vartheta) \mid x_{t+1} = x_t + A_t u_t, \right. \\ & \left. (u_t, x_{t+1}, \vartheta) \in \overline{\text{conv}} \left(\bigcup_{\bar{x}_t \in \mathfrak{X}_t^{int, ad}} \mathcal{A}(\bar{x}_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) \right) \right\}; \end{aligned} \quad (28)$$

denote by $Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t)$ the optimal value of the following program

$$\begin{aligned} Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) := & \text{essinf} \{ c_t(\xi_t) u_t + \vartheta : \\ & \text{s.t. } (u_t, x_{t+1}, \vartheta) \in \overline{\overline{\mathcal{A}}}(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) \}; \end{aligned} \quad (29)$$

and introduce

$$\begin{aligned} \overline{\overline{\mathcal{A}}}^c(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) := & \{ (u_t, x_{t+1}, \vartheta) \mid u_t \in \overline{\overline{\mathcal{U}}}_t^{int, ad}(x_t), \\ & x_{t+1} = x_t + A_t u_t, \vartheta \geq \overline{\overline{\mathcal{Q}}}^{cut}(t+1, x_{t+1}, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) \}. \end{aligned} \quad (30)$$

Proposition 6.1. (i) We have that

$$\overline{\text{conv}} Q^{int,cut}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) = Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t). \quad (31)$$

(ii) The projection of $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$ onto the u_t subspace is

$$\text{proj}_{u_t} \overline{\overline{\mathcal{A}}}(x_t, \xi_t) = \overline{\overline{\mathcal{U}}}_t^{int, ad}(x_t). \quad (32)$$

(iii) $\overline{\overline{\mathcal{A}}}(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) \subset \overline{\overline{\mathcal{A}}}^c(x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t)$.

Proof. (i) Since problems (26) and (29) have the same objective function, and since the feasible set $\mathcal{A}(x_t, \xi_t)$ of (26) is included in $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$, we have that $Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t) \leq Q^{int,cut}(t, x_t, \xi_t)$. As x_t is a parameter of the feasible set $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$, the value function $Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t)$ is convex in x_t . By the definition of a convex hull, $Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t) \leq \overline{\text{conv}} Q^{int,cut}(t, x_t, \xi_t)$.

On the other hand, by (29), for all $\epsilon > 0$, there exists $(u_t^\epsilon, x_{t+1}^\epsilon, \vartheta^\epsilon) \in \overline{\overline{\mathcal{A}}}(x_t, \xi_t)$ such that $c_t(\xi_t) u_t^\epsilon + \vartheta^\epsilon \leq Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t) + \epsilon$. By the definition of $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$, there exists $\left\{ \lambda^i \geq 0, x_t^i \in \mathfrak{X}_t^{int, ad}, (u_t^i, x_{t+1}^i, \vartheta^i) \in \overline{\overline{\mathcal{A}}}(x_t^i, \xi_t), i \in I \right\}$ such that $\sum_{i \in I} \lambda^i = 1$, $u_t^\epsilon = \sum_{i \in I} \lambda^i u_t^i$, $x_t = \sum_{i \in I} \lambda^i x_t^i$, and $x_{t+1}^\epsilon = \sum_{i \in I} \lambda^i x_{t+1}^i$. Then,

$$\begin{aligned} \overline{\text{conv}} Q^{int,cut}(t, x_t, \xi_t) & \leq \sum_{i \in I} \lambda^i \overline{\text{conv}} Q^{int,cut}(t, x_t^i, \xi_t) \leq \sum_{i \in I} \lambda^i Q^{int,cut}(t, x_t^i, \xi_t) \\ & \leq \sum_{i \in I} \lambda^i (c_t(\xi_t) u_t^i + \vartheta^i) = c_t(\xi_t) u_t^\epsilon + \vartheta^\epsilon \\ & \leq Q^{\overline{\overline{\mathcal{A}}}}(t, x_t, \xi_t) + \epsilon \end{aligned}$$

Let $\epsilon \rightarrow 0$, we have $\overline{\text{conv}} Q^{\text{int},\text{cut}}(t, x_t, \xi_t) \leq Q^{\overline{\mathcal{A}}}(t, x_t, \xi_t)$.

- (ii) That $\text{proj}_{u_t} \overline{\mathcal{A}}(x_t, \xi_t) \subset \overline{\mathcal{U}}_t^{\text{int},\text{ad}}(x_t)$ follows directly the definitions of $\overline{\mathcal{A}}(x_t, \xi_t)$, $(\mathfrak{X}, \mathfrak{U})_t^{\text{int},\text{ad}}$ and $\overline{\mathcal{U}}_t^{\text{int},\text{ad}}(x_t)$.

On the other hand, if $u_t \in \overline{\mathcal{U}}_t^{\text{int},\text{ad}}(x_t)$, or equivalently $(x_t, u_t) \in \overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_t^{\text{int},\text{ad}}$, then there exists $\left\{ \lambda^i \geq 0, x_t^i \in \mathfrak{X}_t^{\text{int},\text{ad}}, u_t^i \in \mathfrak{U}_t^{\text{int},\text{ad}}(x_t^i), i \in I \right\}$ such that $\sum_{i \in I} \lambda^i = 1$, $x_t = \sum_{i \in I} \lambda^i x_t^i$ and $u_t = \sum_{i \in I} \lambda^i u_t^i$. Let us take $x_{t+1}^i = x_t^i + A_t u_t^i$, and $\vartheta^i \geq \overline{\mathcal{Q}}^{\text{cut}}(t+1, x_{t+1}^i, \xi_t)$ for each $i \in I$. Then, $(u_t^i, x_{t+1}^i, \vartheta^i) \in \mathcal{A}(x_t^i, \xi_t)$. Taking $x_{t+1} = \sum_{i \in I} \lambda^i x_{t+1}^i$ and $\vartheta_{t+1} = \sum_{i \in I} \lambda^i \vartheta^i$, then $(u_t, x_{t+1}, \vartheta) \in \overline{\mathcal{A}}_t(x_t, \xi_t)$, or equivalently $u_t \in \text{proj}_{u_t} \overline{\mathcal{A}}_t(x_t, \xi_t)$.

- (iii) Using the same argument in the proof of the second result, and the convexity of $\overline{\mathcal{Q}}(t+1, x_{t+1}, \xi_t)$, we obtain immediately that $\overline{\mathcal{A}}(x_t, \xi_t) \subset \overline{\mathcal{A}}^c(x_t, \xi_t)$. \square

We have the following corollary for the extreme points of $\mathfrak{X}_t^{\text{int},\text{ad}}$.

Corollary 6.2. *If x_t is an extreme point of $\mathfrak{X}_t^{\text{int},\text{ad}}$, then*

$$\overline{\mathcal{A}}(x_t, \xi_t) = \overline{\text{conv}} \mathcal{A}(x_t, \xi_t). \quad (33)$$

Remark 6.3. Following the third property of Proposition 6.1, the optimality cuts $\vartheta \geq \overline{\lambda}_t^x x_{t+1} + \overline{\lambda}_t^0$, $(\overline{\lambda}_t^x, \overline{\lambda}_t^0) \in \mathcal{O}_t(\xi_t)$, which can only approximate $\overline{\mathcal{Q}}^{\text{cut}}(t+1, x_{t+1}, \xi_t)$, are not enough. That is why we extend the optimality cut to form

$$\vartheta \geq \overline{\lambda}_t^x x_{t+1} + \overline{\lambda}_u^x u_t + \overline{\lambda}_t^0.$$

At this stage, we are ready to introduce the cutting plane method to approximate $\overline{\mathcal{A}}(x_t, \xi_t)$ and then $\overline{\text{conv}} Q^{\text{int},\text{cut}}(t, x_t, \xi_t)$. In practice, we write (29) in the following formulation:

$$\begin{aligned} & \text{essinf} && c_t(\xi_t)u_t + \hat{\vartheta}(t+1, x_{t+1}, \xi_t, \mathcal{O}_t) \\ & \text{subject to} && u_s \in \mathfrak{U}_s, \\ & && x_{s+1} = x_s + A_s u_s, \quad s = t, \dots, T-1, \\ & && x_T \in \mathfrak{X}_T, \\ & \text{(feasibility cut)} && \lambda_s^x x_s + \lambda_s^u u_s \leq \lambda_s^0, \quad \forall (\lambda_s^x, \lambda_s^u, \lambda_s^0) \in \mathcal{I}_s, s = t, \dots, T-1, \\ & \text{(optimality cut)} && \hat{\vartheta}(t+1, x_{t+1}, \xi_t, \mathcal{O}_t) \geq \overline{\lambda}^x x_{t+1} + \overline{\lambda}^u u_t + \overline{\lambda}^0, \forall (\overline{\lambda}^x, \overline{\lambda}^u, \overline{\lambda}^0) \in \mathcal{O}_t(\xi_t); \end{aligned} \quad (34)$$

Note that $\mathcal{O}_t(\xi_t)$ in (34) is different from the one in (14), even though we use the same notations. Remark that in (34) at time t , we add all feasibility cuts $\mathcal{I}_{[t, T-1]}$ of future time steps $t \leq s \leq T-1$.

We denote the value function of (34) by $Q^{\text{cut}}(t, x_t, \xi_t, \mathcal{I}_{[t, T-1]}, \mathcal{O}_t)$ ($Q^{\text{cut}}(t, x_t, \xi_t)$ for short). Using the Fenchel-Moreau theorem, we here note it by its dual approximation:

$$Q^{\text{cut}}(t, x_t, \xi_t, \mathcal{I}_{[t+1, T-1]}, \mathcal{O}_t) = \max \left\{ \theta : \theta \geq \lambda_t^x x_t + \lambda_t^0, (\lambda_t^x, \lambda_t^0) \in \mathcal{O}_t(\xi_t) \right\}. \quad (35)$$

These feasibility cuts $\mathcal{I}_{[t,t-1]}$ and optimality cuts \mathcal{O}_t are here to approximate $\overline{\overline{\mathcal{A}}}(x_t, \xi_t, \mathcal{I}_{[t+1,t-1]}, \mathcal{O}_t)$. Since we build the cutting planes in the SDDP algorithm, we can only generate a finite number of cutting planes to approximate $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$. Then, the value obtained in (34) $Q^{cut}(t, x_t, \xi_t, \mathcal{I}_{[t,T-1]}, \mathcal{O}_t)$ is itself a lower bound of $\overline{\text{conv}} Q^{int,cut}(x_t, \xi_t, \mathcal{I}_{[t+1,T-1]}, \mathcal{O}_{t+1})$.

Remark 6.4. Since our algorithm is still in the SDDP framework, we cannot have a tighter lower bound than $Q_T^{int,conv}(t, x_t, \xi_t)$ as mentioned in remark 5.3:

$$Q^{int,cut}(t, x_t, \xi_t, \mathcal{I}_{[t+1,T-1]}, \mathcal{O}_{t+1}) \leq Q_T^{int,conv}(t, x_t, \xi_t); \quad (36)$$

$$Q^{cut}(t, x_t, \xi_t, \mathcal{I}_{[t+1,T-1]}, \mathcal{O}_{t+1}) \leq \overline{\text{conv}} Q_T^{int,conv}(t, x_t, \xi_t). \quad (37)$$

Our algorithm, combining the SDDP algorithm with the cutting plane method, consists in iterations of a forward and a backward pass. In the forward pass, we intend to compute a sub-optimal decision solution for each simulated trajectory (ξ_t) and to obtain an upper bound of the optimal value of (2). In the backward pass, we aim at generating a tighter lower bound to better approximate $Q^{int}(t, x_t, \xi_t)$ and to get a lower bound of the optimal value of (2). The generation of the cutting plane occurs in the forward pass procedure. We first solve (34) where \hat{x}_t is integer due to Assumption 1.1, and denote by $(\hat{u}_t^{cont}, \hat{x}_{t+1}, \hat{v})$ the optimal continuous solution.

- If \hat{u}_t^{cont} is integer, we set $\hat{u}_t^{int} = \hat{u}_t^{cont}$, and go on directly to the next time step $t + 1$.
- Otherwise, in some cases, we are able to use the cutting plane technique to eliminate the non-integer solution $(\hat{u}_t^{cont}, \hat{x}_{t+1}, \hat{v})$, and thus refine the approximation of the polyhedron $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$.

Remark 6.5. We use (34) to build a convex function to approximate $\overline{\text{conv}} Q^{int,cut}(t, x_t, \xi_t)$. Then, the objective of (34) is different from using cutting plane method to solve MILP by the well known theorem [19, Theorem 6.3, Section I.4.6]. If we want to apply the theorem to transfer the MILP (26) to linear programming (LP), the feasible set should be $\overline{\text{conv}} \mathcal{A}(x_t, \xi_t)$ instead of $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$. The key difference is that the cutting planes for $\overline{\overline{\mathcal{A}}}(x_t, \xi_t)$ are valid for all state values $\mathfrak{X}_t^{int,ad}$, whilst that the ones for $\overline{\text{conv}} \mathcal{A}(x_t, \xi_t)$ may be only valid for some specific values $\hat{x}_t \in \mathfrak{X}_t^{int,ad}$.

During the forward pass of our algorithm, we are interested in computing an integer solution. Then, the cutting plane for $\overline{\text{conv}} \mathcal{A}(x_t, \xi_t)$ is useful. Nevertheless, they are totally useless in the backward pass of our algorithm.

6.1 Generating cutting planes

We discuss below how to generate cutting planes. We distinguish 3 cases depending on where \hat{u}_t^{cont} to build a cutting plane is located.

6.1.1 Case when $\hat{u}_t^{cont} \notin \overline{\overline{\mathcal{U}}}_t^{int,ad}(\hat{x}_t)$

This is the simplest case where we are sure to eliminate the optimal non-integer solution \hat{u}_t^{cont} from the feasible set $\overline{\overline{\mathcal{U}}}_t^{int,ad}(\hat{x}_t) = \text{proj}_u \overline{\overline{\mathcal{A}}}(\hat{x}_t, \xi_t)$. Using various

cutting plane techniques (see e.g. [11, 19]), we can compute a cutting plane of the feasibility cut form: $\lambda_t^x x_t + \lambda_t^u u_t \leq \lambda_t^0$. Here, we present the Fenchel cut method [9]. The MIP problem associated to build Fenchel cut is

$$\begin{aligned} \max_{\lambda_t^x, \lambda_t^u} \min_{x_t, u_t} & \lambda_t^x (\hat{x}_t - x_t) + \lambda_t^u (\hat{u}_t^{cont} - u_t) \\ \text{subject to} & (\lambda_t^x, \lambda_t^u) \in \Lambda, \\ & (x_t, u_t) \in (\mathfrak{X}, \mathfrak{U})_t^{int, ad}, \end{aligned} \quad (38)$$

where Λ is the unit sphere in a proper norm. The cutting plane is $\lambda_t^x x_t + \lambda_t^u u_t \leq \lambda_t^0$, where $\lambda_t^0 = \lambda_t^x \hat{x}_t + \lambda_t^u \hat{u}_t^{cont}$, and $(\lambda_t^x, \lambda_t^u, \hat{x}_t, \hat{u}_t^{cont})$ is the optimal solution of (38). We then add the cutting plane into the feasibility cut collection \mathcal{I}_t .

The MIP problem (38) can also help to characterize the set $\overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_t^{int, ad}$. If $(\hat{x}_t, \hat{u}_t^{cont}) \in \overline{\text{conv}}(\mathfrak{X}, \mathfrak{U})_t^{int, ad}$, the optimal value of (38) is 0. Otherwise, the optimal value is strict positive.

Remark 6.6. It may be surprising that we use a MIP problem (38) to generate a cutting plane. However, in this paper, this is meaningful since we intend to solve a stochastic problem. The cutting plane generated in (38) is universal valid for any realization of random variable ξ_t . We have such nice property since the feasible sets introduced in Section 3 are deterministic as mentioned in remark 3.1.

Example 6.7 (example 2.2 continued). Let us consider the subproblem where $\xi = -2$ and $\hat{x}_0 = 3$. The continuous relaxation problem returns the optimal solution $(\hat{u}_0^{cont}, \hat{x}_1, \hat{\vartheta}) = (1.5, 6, 1.5)$, where $\hat{u}_0^{cont} \notin \overline{\mathfrak{U}}_0^{int, ad}(\hat{x}_0) = [0, 1]$.

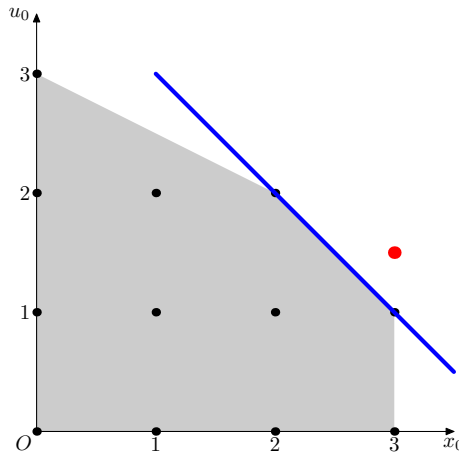


Figure 5: the feasible set on (x_0, u_0) subspace; the red point is the optimal continuous solution; the blue line is the cutting plane.

A cutting plane $x_0 + u_0 \leq 4$ can be generated to eliminate the non-integer point (see Figure 5). Then, after adding this cutting plane, the subproblem (34) returns the optimal solution $(\hat{u}_0^{cont}, \hat{x}_1, \hat{\vartheta}) = (1, 5, 1)$, where \hat{u}_0^{cont} is integer. Finally, the dual value of (34) helps build an optimality cut $1.5x_0 - 5.5$ of $Q^{cut}(x_0, \xi = -2)$ which is exactly the convex hull $\overline{\text{conv}}Q^{int, cut}(x_0, \xi = -2)$ at point $x_0 = 3$ (see Figure 4).

Nevertheless, if the non-integer solution $\hat{u}_t^{cont} \in \overline{\mathfrak{U}}_t^{int,ad}(\hat{x}_t)$, the problem is much more complex, and that is the reason of the non-convexity of the Bellman value function of the integer problem. In the following analysis, we separate 2 cases either $\hat{u}_t^{cont} \in \overline{\mathfrak{U}}_t^{int,ad}(\hat{x}_t) \setminus \overline{\text{conv}} \mathfrak{U}_t^{int,ad}(\hat{x}_t)$ or $\hat{u}_t^{cont} \in \overline{\text{conv}} \mathfrak{U}_t^{int,ad}(\hat{x}_t)$. Furthermore, if $\hat{u}_t^{cont} \in \overline{\mathfrak{U}}_t^{int,ad}(\hat{x}_t)$, any cutting plane of the feasibility cut form $\lambda^x x_t + \lambda^u u_t \leq \lambda^0$ is not useful anymore by Proposition 6.1. Then, we will focus on generating cutting planes of the optimality cut form: $\vartheta \geq \bar{\lambda}^x x_{t+1} + \bar{\lambda}^u u_t + \bar{\lambda}^0$.

6.1.2 Case when $\hat{u}_t^{cont} \in \overline{\mathfrak{U}}_t^{int,ad}(\hat{x}_t) \setminus \overline{\text{conv}} \mathfrak{U}_t^{int,ad}(\hat{x}_t)$

Let us start by the following example to illustrate this case.

Example 6.8 (example 2.2 continued). Let us consider the subproblem where $\xi = -2$ and $\hat{x}_0 = 1$. The continuous solution is $(\hat{u}_0^{cont}, \hat{x}_1, \hat{\vartheta}) = (2.5, 6, 1.5)$, where $\hat{u}_0^{cont} \in \overline{\mathfrak{U}}_0^{int,ad}(\hat{x}_0) \setminus \overline{\text{conv}} \mathfrak{U}_0^{int,ad}(\hat{x}_0) = [2, 2.5]$. Figure 6 illustrates the feasible set and the optimal continuous solution. This example shows that this

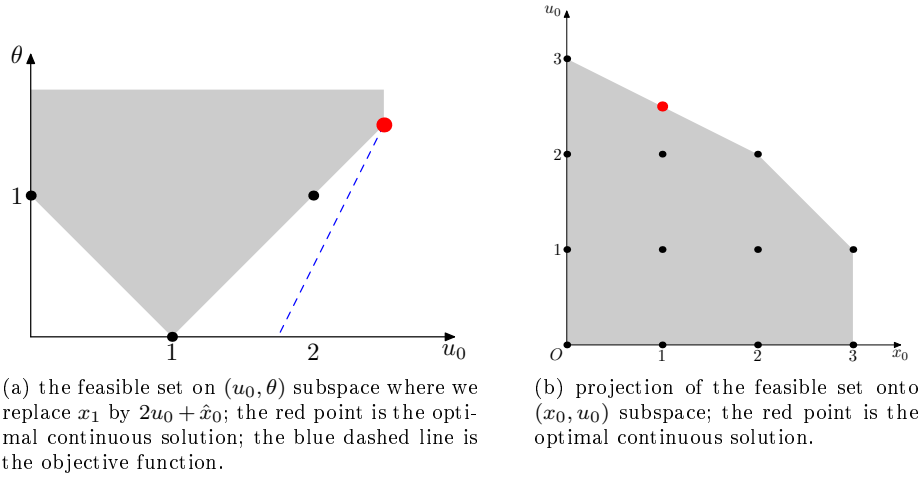


Figure 6: Illustration of example 6.8.

case is sensitive to the non-convexity and the discontinuity of $Q^{int,cut}(t, x_t, \xi_t)$ (see Figure 4 at point $\xi = -2, x_0 = 1$). Therefore, the cutting plane method may fail to compute the optimal integer solution.

If the cutting plane method fails to eliminate the non-integer solution, we have to solve the MILP problem (26) in order to compute an integer solution and to continue the forward pass of the SDDP algorithm. In example 6.8, the MILP (26) returns an integer solution $(\hat{u}_0, \hat{x}_1) = (2, 5)$.

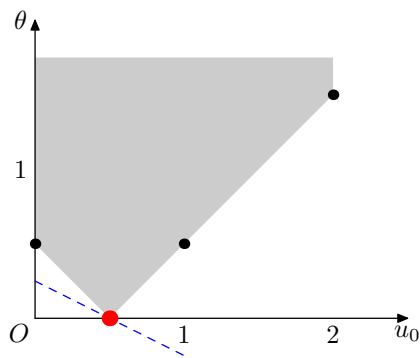
Remark 6.9. Since (26) is a MILP problem, some metaheuristic methods [12] in integer programming can be applied here, for instance the integer local search method (see Walser [30]). Thus, we can avoid solving a MILP problem (26). The solution by metaheuristic method is of course itself sub-optimal.

Following Lemma 3.2, if $\mathfrak{X}_t^{int,ad}$ is binary, $\overline{\mathfrak{U}}_t^{int,ad}(\hat{x}_t) = \overline{\text{conv}} \mathfrak{U}_t^{int,ad}(\hat{x}_t)$ and the case described in the current subsection does not happen.

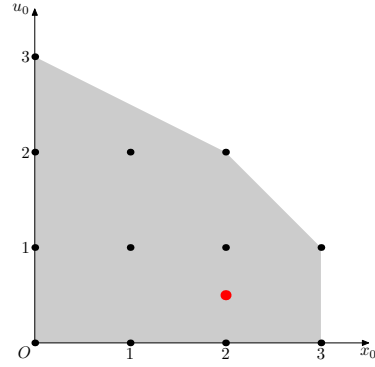
6.1.3 Case when $\hat{u}_t^{cont} \in \overline{\text{conv}} \mathfrak{U}_t^{int,ad}(\hat{x}_t)$

We first illustrate the case with the following example.

Example 6.10 (example 2.2 continued). Let us consider the subproblem where $\hat{x}_0 = 2$, $\xi = 0.5$. Figure 7 illustrates the feasible set and the optimal continuous solution. The optimal continuous solution is $(\hat{u}_0^{cont}, \hat{x}_1, \hat{\vartheta}) = (0.5, 3, 0)$, where



(a) the feasible set on (u_0, θ) subspace where we replace x_1 by $2u_0 + \hat{x}_0$; the red point is the optimal continuous solution; the blue dashed line is the objective function.



(b) projection of the feasible set onto (x_0, u_0) subspace; the red point is the optimal continuous solution.

Figure 7: Illustration of example 6.10.

\hat{u}_0^{cont} is not integer, but it is in $\overline{\text{conv}} \mathfrak{U}_0^{int,ad}(\hat{x}_0) = [0, 3]$. This is another reason to cause the non-convexity of $Q^{int,cut}(x_0, \xi)$ (see Figure 4 at point $x_0 = 2$, $\xi = 0.5$).

Therefore, the cutting plane method again fails to eliminate an optimal non-integer solution. In order to compute an integer solution \hat{u}_t^{int} , we have to solve the MILP problem (26).

Case when \hat{x}_t is an extreme point of $\overline{\text{conv}} \mathfrak{X}_t^{int,ad}$. However, if \hat{x}_t is an extreme point of $\overline{\text{conv}} \mathfrak{X}_t^{int,ad}$, $\overline{\mathcal{A}}(x_t, \xi_t)$ is characterized by Corollary 6.2. Furthermore, if problem (26) has a unique optimal solution, by [19, Theorem 6.3, Section I.4.6], the optimal solution to problem (29) is unique and integer, and both optimal solutions coincide. Therefore, if (34) returns an optimal non-integer solution \hat{u}_t^{cont} , we can generate a cutting plane of optimality cut form to eliminate \hat{u}_t^{cont} and add it into $\mathcal{O}_t(\xi_t)$.

The cutting plane generating procedure are:

- We can first build a cutting plane for $\overline{\text{conv}} \mathcal{A}(\hat{x}_t, \xi_t)$ by various methods (see e.g. [11, 19]), for instance the Gomory cut technique [13, 3]. In view of Corollary 6.2 and the fact that $\hat{u}_t^{cont} \in \mathfrak{U}_t^{int,ad}(\hat{x}_t)$, the cutting plane is

of form:

$$\vartheta \geq \hat{\lambda}_t^x x_{t+1} + \hat{\lambda}_t^0. \quad (39)$$

Note that the cutting plane generated until now may be only valid for $\overline{\text{conv}} \mathcal{A}(\hat{x}_t, \xi_t)$, i.e. not valid for $\overline{\mathcal{A}}(x_t, \xi_t)$, $x_t \neq \hat{x}_t$.

- Since $x_t = x_{t+1} - A_t u_t$, then

$$\langle x^*, x_{t+1} - A_t u_t - \hat{x}_t \rangle \leq 0, \quad \forall x^* \in N_{\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}}(\hat{x}_t); \quad (40)$$

where $N_K(x)$ is the normal cone of K at point x . Since \hat{x}_t is an extreme point of $\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}$, then $\text{int} \left(N_{\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}}(\hat{x}_t) \right)$ is not empty. Finally, adding (40) onto (39), we have

$$\vartheta \geq (\hat{\lambda}_t^x + x^*) x_{t+1} - x^* A_t u_t + (\hat{\lambda}_t^0 - x^* \hat{x}_t), \quad \forall x^* \in N_{\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}}(\hat{x}_t) \quad (41)$$

is still valid for $\overline{\text{conv}} \mathcal{A}(\hat{x}_t, \xi_t)$.

- The final step is to choose $x^* \in N_{\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}}(\hat{x}_t)$ such that this cutting plane (41) is valid for all $\overline{\mathcal{A}}(x_t, \xi_t)$ such that $x_t \in \mathfrak{X}_t^{\text{int},ad}$. Then, (41) is the cutting plane we are looking for. We first solve two linear programs that

$$\vartheta^{lb} := \min \left\{ \vartheta, \quad \text{s.t.} \quad x_{t+1} \in \overline{\text{conv}} \mathfrak{X}_{t+1}^{\text{int},ad}, \right. \\ \left. \vartheta \geq \bar{\lambda}_t^x x_{t+1} + \bar{\lambda}_t^0, \quad \forall (\bar{\lambda}_t^x, \bar{\lambda}_t^0) \in \mathcal{O}_t(\xi) \right\}; \quad (42)$$

and

$$\vartheta^{ub} := \max \left\{ \hat{\lambda}_t^x x_{t+1} + \hat{\lambda}_t^0, \quad \text{s.t.} \quad x_{t+1} \in \overline{\text{conv}} \mathfrak{X}_{t+1}^{\text{int},ad} \right\}; \quad (43)$$

and denote by ϑ^{lb} and ϑ^{ub} the optimal value to (42) and to (43). Remark that in (42), we just add the optimality cuts which approximate $\overline{\mathcal{Q}}^{\text{cut}}(t+1, x_{t+1}, \xi_t)$. Then, we intend to find a x^* such that

$$x^*(x_t - \hat{x}_t) \leq \vartheta^{lb} - \vartheta^{ub}, \quad (44)$$

for all $x_t \in \mathfrak{X}^{\text{ck}}(\hat{x}_t, d, \hat{\alpha})$, $d \in \text{int} \left(N_{\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}}(\hat{x}_t) \right)$, where $\mathfrak{X}^{\text{ck}}(\hat{x}_t, d, \alpha)$, $\alpha \leq 0$ is defined as

$$\mathfrak{X}^{\text{ck}}(\hat{x}_t, d, \alpha) := \left\{ x_t : \langle d, x_t - \hat{x}_t \rangle = \alpha, x_t \in \overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad} \right\}. \quad (45)$$

We define the function $\delta(\alpha) = \max_{x \in \mathfrak{X}^{\text{ck}}(\hat{x}_t, d, \alpha)} |x - \hat{x}_t|_2$. Obviously, $\delta(\cdot)$ is continuous and strict decreasing, and then $\delta^{-1}(\cdot)$ is well defined and strict decreasing. Finally, we take

$$\hat{\alpha} := \delta^{-1}(1) = \left\{ \alpha : \max_{x \in \mathfrak{X}^{\text{ck}}(\hat{x}_t, d, \alpha)} |x - \hat{x}_t|_2 = 1 \right\}. \quad (46)$$

The Figure 8 illustrates the set $\mathfrak{X}^{\text{ck}}(\hat{x}_t, d, \hat{\alpha})$.

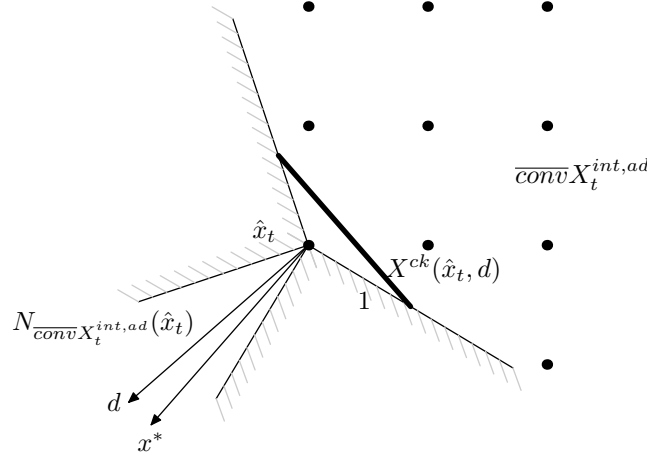


Figure 8: illustration of $\mathfrak{X}_t^{ck}(\hat{x}_t, d)$ and x^* .

Obviously, we have the existence of a solution $x^* \in \text{int} \left(N_{\text{conv}} \mathfrak{X}_t^{\text{int},ad}(\hat{x}_t) \right)$, since

$$\lim_{\kappa \rightarrow \infty} \langle \kappa x^*, x_t - \hat{x}_t \rangle = -\infty, \quad \forall x^* \in \text{int} \left(N_{\text{conv}} \mathfrak{X}_t^{\text{int},ad}(\hat{x}_t) \right). \quad (47)$$

Furthermore, since for all $x_t \in \mathfrak{X}_t^{\text{int},ad} \setminus \{\hat{x}_t\}$, there exists $\kappa \geq 1, x^{ck} \in X^{ck}(\hat{x}_t, d, \hat{\alpha})$, such that $x_t - \hat{x}_t = \kappa(x^{ck} - \hat{x}_t)$, then for all $x_t \in \mathfrak{X}_t^{\text{int},ad} \setminus \{\hat{x}_t\}$, $u_t \in \overline{\text{conv}} \mathfrak{U}_t^{\text{int},ad}(x_t)$, such that $x_{t+1} = x_t + A_t u_t$:

$$\begin{aligned} & (\hat{\lambda}_t^x + x^*)x_{t+1} - x^* A_t u_t + (\hat{\lambda}_t^0 - x^* \hat{x}_t) \\ &= \hat{\lambda}_t^x x_{t+1} + \hat{\lambda}_t^0 + x^*(x_t - \hat{x}_t) \\ &\leq \vartheta^{ub} + \kappa x^*(x^{ck} - \hat{x}_t) \leq \vartheta^{lb}. \end{aligned} \quad (48)$$

Then, the cutting plane (41) does not contribute anything on other points in $\mathfrak{X}_t^{\text{int},ad}$ except \hat{x}_t . Hence, the cutting plane (41) is valid for all $\overline{\mathcal{A}}(x_t, \xi_t)$ such that $x_t \in \mathfrak{X}_t^{\text{int},ad}$.

The following example illustrates the case when \hat{x}_t is an extreme point of $\overline{\text{conv}} \mathfrak{X}_t^{\text{int},ad}$.

Example 6.11 (example 2.2 continued). Let us consider the subproblem when $\hat{x}_0 = 0, \xi = 0.5$. The optimal continuous solution is $(\hat{u}_0^{\text{cont}}, \hat{x}_1, \hat{\vartheta}) = (1.5, 3, 0)$, where \hat{u}_0^{cont} is not integer, but it is in $\overline{\text{conv}} \mathfrak{U}_0^{\text{int},ad}(0) = [0, 3]$. Figure 9 illustrates the feasible set.

Since $\hat{x}_0 = 0$ is an extreme point of $\overline{\text{conv}} \mathfrak{X}_0^{\text{int},ad} = [0, 3]$, we use the previous procedure to build a cutting plane. Applying the Gomory cut technique onto $\overline{\text{conv}} \mathcal{A}(\hat{x}_0 = 0, \xi = 0.5)$ returns an inequality

$$\vartheta \geq 0.5,$$

which eliminates the non-integer point $\hat{u}_0^{\text{cont}} = 1.5$. The normal cone is $N_{\text{conv}} \mathfrak{X}_t^{\text{int},ad}(x_0 = 0) = \{x^* \leq 0\}$. Obviously, we have $\vartheta^{lb} = 0$ and $\vartheta^{ub} = 0.5$. Then, following (44),

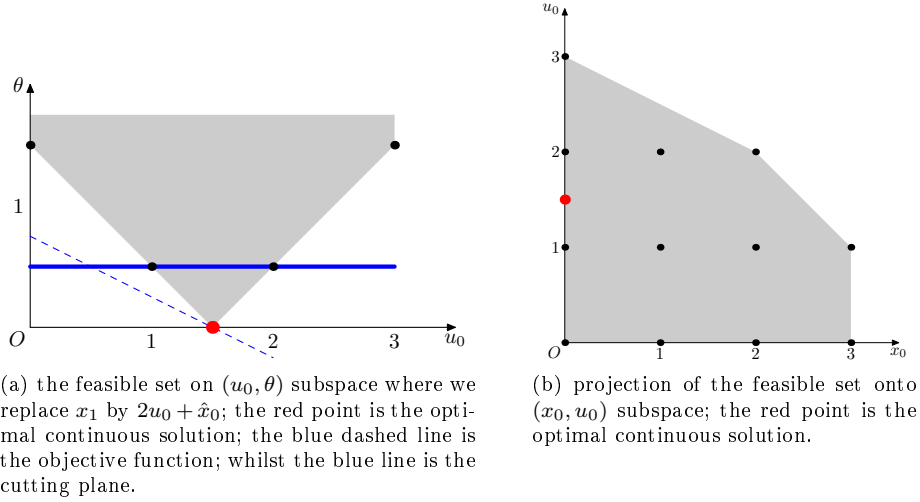


Figure 9: Illustration of example 6.11.

we need to find $x^* \leq 0$ such that $x^* \leq -0.5$. Taking $x^* = -0.5$ gives us that a cutting plane

$$\vartheta \geq u_0 - 0.5x_1 + 0.5.$$

After adding this inequality, problem (34) returns an integer solution $\hat{u}_0 = 1$. Finally, problem (34) can build an optimality cut $\vartheta \geq -0.5x_0 + 1$ of $Q^{cut}(x_0, \xi = 0.5)$ which is exactly the convex hull $\overline{\text{conv}} Q^{int, cut}(x_0, \xi = 0.5)$ around point $x_0 = 0$. Of most importance, this cutting plane is valid for all state values $x_0 \in \{0, 1, 2, 3\}$.

Finally, we have the following theorem.

Theorem 6.12. *If $\mathfrak{X}_t^{int, ad} \subset \mathbb{B}^m$, then we can solve the multi-stage SMILP (2) exactly by our algorithm.*

Proof. We have first that $Q_T^{int, conv}(t, x_t, \xi_t) = Q^{int}(t, x_t, \xi_t)$ by Proposition 5.1, and $Q^{int, cut}(t, x_t, \xi_t) \leq Q_T^{int, conv}(t, x_t, \xi_t)$ by remark 6.4. It now suffices to show that $Q^{int}(t, x_t, \xi_t) \leq Q^{cut}(t, x_t, \xi_t)$, or equivalently to say that (34) always returns an integer solution by cutting plane method.

By forward induction, \hat{x}_t is integer. Following the Corollary 3.3, \hat{u}_t^{cont} is either in $\overline{\text{conv}} \mathfrak{U}_t^{int, cut}(\hat{x}_t)$ or not in $\overline{\mathfrak{U}}_t^{int, cut}(\hat{x}_t)$. In both cases, we can build a cutting planes to eliminate any optimal non-integer solution \hat{u}_t^{cont} . Then, the result follows. \square

6.2 Algorithm

We summarize in Algorithm 1 the algorithm of generating cutting plane and of computing integer solution. Remark that in general, it is possible to build infinite cutting planes, or the cutting plane method fails to generated an integer solution. Therefore, we need to fix a maximum iteration number N_{cut}^{max} to avoid infinite loop.

Algorithm 1 generate cutting plane and compute integer solution

```

 $n_{cut} = 0;$ 
repeat
  solve problem (34) and set optimal solution  $\hat{u}_t^{cont}$ ;
  if  $\hat{u}_t^{cont} \notin \mathbb{Z}^n$ 
     $n_{cut} = n_{cut} + 1;$ 
    if  $\hat{u}_t^{cont} \notin \overline{\mathfrak{U}}_t^{int,ad}(x_t^m)$ 
      compute feasibility cut by Fenchel cut (38) and add into  $\mathcal{I}_t$ ;
    else
      compute optimality cut by the procedure in section 6.1.3 and add into  $\mathcal{O}_t(\xi_t)$ ;
    end if
    if  $n_{cut} \geq N_{cut}^{max}$ 
      solve MILP problem (26) and set optimal solution  $\hat{u}_t^{cont}$ ;
    end if
  end if
until  $\hat{u}_t^{cont} \in \mathbb{Z}^n$ 

```

Finally, we summarize the whole algorithm in Algorithm 2. Since in general, the upper bound \bar{v} and the lower bound \underline{v} do not converge. Hence, we do not apply the usual the SDDP stopping condition $|\underline{v} - \bar{v}| \leq 1.96\sigma(\bar{v})$. We here fix a maximum iteration number N_{loop}^{max} to avoid infinite loop.

Example 6.13 (example 2.2 continued). Finally, having computed cutting planes following the 3 cases, we finally obtain some improvement to obtain tighter lower bounds of $Q^{int}(x_0, \xi)$ and $\overline{Q}^{int}(x_0)$ than $Q^{cont}(x_0, \xi)$ and $\overline{Q}^{cont}(x_0)$ in the example 2.2 (see Figure 10). In this example, we are able to obtain exactly $\overline{\text{conv}} Q_1^{int,conv}(x_0, \xi)$ and $\overline{\overline{Q}}_1^{int,conv}(x_0)$.

7 Numerical test on LNG portfolio

We now consider a gas trading portfolio. A trading company purchases natural gas from a set of producing countries indexed at a price formula and sells it to consuming countries at another other price formula (see Figure 11 for the main market). Annual quantity and price formulae have been agreed contractually, the latter are functions of the future prices of major energy markets ξ_t , such as crude oil price (OIL), north American natural gas price (NA NG), and Europe natural gas price (EU NG). In the numerical test, we consider there are two cargo sizes (2.9 and 3.4) and we assume that the cargo are always fully charged. Thus, the decision variable is the number of different cargos on each route, instead of the LNG quantity. The detail of constraints and price formulae is shown the Table 1 below.

We assume that the random process follows the dynamic

$$\xi_t^i = (F_0^t)^i \exp \left(\sum_{s=0}^{t-1} \sigma^i W_s^i - \frac{1}{2} (\sigma^i)^2 t \right);$$

Algorithm 2 whole algorithm

Step 0 **Initialization:**
 build quantization tree (Γ_t) ;
 solve the continuous relaxation problem (3);
 $n_{loop} = 0$.

Step 1 **Forward pass:**
for $m = 1, \dots, M_f$ *do*
 simulate (ξ_t^m) following dynamic (1);
 for $t = 1, \dots, T - 1$ *do*
 use Algorithm 1 to compute $\hat{u}_t^m \in \mathbb{Z}^n$;
 end for
 compute $v^m = \sum_{t=0}^{T-1} c_t(\xi_t^m) \hat{u}_t^m + g(\xi_T^m, \hat{x}_T^m)$;
end for
 compute $\bar{v} = \frac{1}{M_f} \sum_{m=1}^{M_f} v^m$ and $\sigma(\bar{v}) = \frac{1}{M_f} \sqrt{\sum_{m=1}^{M_f} (v^m - \bar{v})^2}$.

Step 2 **Backward pass:**
for $t = T - 1, \dots, 1$ *do*
 choose M_b samples;
 for $m = 1, \dots, M_b$ *do*
 for $\xi_t^j \in \Gamma_t$ *do*
 solve subproblem (34);
 compute new optimality cut $((\lambda^x)_t^j, (\lambda^0)_t^j)$;
 end for
 for $\xi_{t-1}^i \in \Gamma_{t-1}$ *do*
 compute $\sum_{\xi_t^j \in \Gamma_t} p_{t-1}^{ij} ((\lambda^x)_t^j, (\lambda^0)_t^j)$ and add into \mathcal{O}_{t-1}^i ;
 end for
 end for
end for
 $t = 0$
 solve subproblem (34) and compute backward value \underline{v} .

Step 3 **stopping condition:**
if $n_{loop} \leq N_{loop}^{max}$
 $n_{loop} = n_{loop} + 1$;
 go to Step 1;
else
 stop;
end if

or equivalently

$$\xi_{t+1}^i = \xi_t^i \frac{(F_0^{t+1})^i}{(F_0^t)^i} \exp\left(\sigma^i W_t^i - \frac{1}{2}(\sigma^i)^2\right); \quad (49)$$

where F_0^t is the forward price price with maturity t observed at time $t = 0$, σ is the volatility, and W^i follows the normal distribution $\mathcal{N}(0, 1)$ such that the correlation $\text{corr}(W_t^i, W_t^j) = \rho_{ij}$.

The horizon is $T = 6$. The parameters driving the stochastic processes in the test are set as follows: volatility $\sigma_1 = \sigma_2 = \sigma_3 = 40\%$, and correlations $\rho_{12} = 0.7$, $\rho_{13} = 0.2$ and $\rho_{23} = 0.4$. The forward price F_0^t is read from the

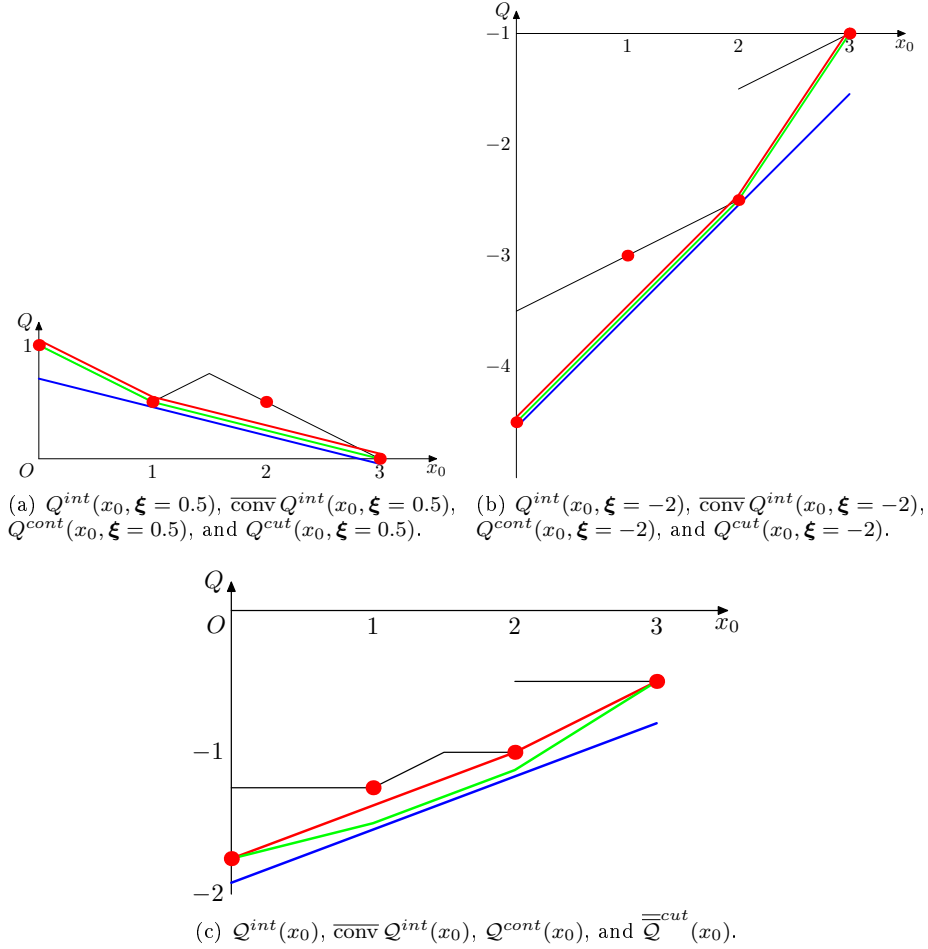


Figure 10: Improvement of example 2.2. The red line is $\overline{\text{conv}} Q^{int}(x_0, \xi)$ or $\overline{\text{conv}} Q^{int}(x_0)$. The blue line is $Q^{cont}(x_0, \xi)$ or $Q^{cont}(x_0)$. The green line is $Q^{cut}(x_0, \xi)$ or $\overline{Q}^{cut}(x_0)$.

energy market. We take the following values in the numerical test (see in Table 2). Finally, We build a quantization tree of 15000 points to discretize the random process ξ_t .

In the test, we apply the heuristic method presented in Section 4 and the improvement by the cutting plane technique in Section 6. In addition, we compare them to the deterministic counterpart MILP problem:

$$\begin{aligned}
 \inf \quad & \sum_{t=0}^{T-1} c_t(\mathbb{E}[\xi_t])u_t + g(\mathbb{E}[\xi_T], x_T) \\
 \text{subject to} \quad & u_t \in \mathbb{Z}^n \cap \mathcal{U}_t, \\
 & x_{t+1} = x_t + A_t u_t, \\
 & x_0 = 0, x_T \in \mathcal{X}_T.
 \end{aligned} \tag{50}$$

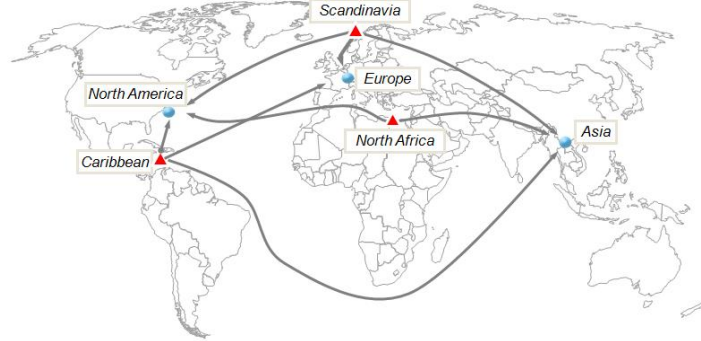


Figure 11: A fictive supply and demand portfolio, as well as the possible routes.
 ▲: producing country; ●: consuming country.

Port	Cargo Size	Annual QC.*	Monthly QC.*	Price formula**
Caribbean	2.9, 3.4	[21.0, 27.0]	[0.0, 6.9]	NA NG - 0.1
Scandinavia	2.9	[12.0, 18.0]	[0.0, 3.0]	0.07OIL + 1.0
North Africa	2.9, 3.4	[47.0, 53.0]	[0.0, 12.8]	0.8NA NG + 0.9
North American	2.9, 3.4	[40.0, 46.0]	[0.0, 9.2]	NA NG
Europe	2.9, 3.4	[33.0, 39.0]	[0.0, 9.2]	EU NG
Asia	2.9, 3.4	[7.0, 13.0]	[0.0, 3.4]	0.08OIL - 0.8

Table 1: Constraints and price formulae

* The quantity unit is in TBtu. MMBtu stands for a million British thermal unit, a TBtu is a trillion British thermal unit thus equivalent to a Million MMBtu.

** The price unit is \$/MMBtu.

Time	OIL*	EU NG*	NA NG*
0	73.0	5.2	5.15
1	73.5	5.3	5.30
2	74.0	5.2	5.22
3	74.5	4.5	4.51
4	75.0	4.4	4.35
5	75.5	4.3	4.33

Table 2: Forward price

* $\xi^1 = \text{OIL}$, $\xi^2 = \text{EU NG}$, and $\xi^3 = \text{NA NG}$.

Furthermore, we use (50) to generate a control variate.

We obtain the following results (see Table 3).

The continuous relaxation returns a lower bound -32.3395 and an upper bound -31.9767 . Based on the optimality cuts computed in resolution of the continuous relaxation problem, the heuristic method in Section 4 leads to an upper bound of -13.8319 for the integer problem, with standard deviation of 0.417892 , with 5000 forward simulations. Finally, applying the cutting plane technique, we can improve the forward value from -13.8319 to -17.0725 , a gain

	u.b.	$\sigma(\text{u.b.})^*$	l.b.	det. counterpart value
Continuous	-31.9767	0.6387	-32.3395	-2.2049
Integer (heuristic)	-13.8319	0.4179		3.0384
Integer (cut)	-17.0725	1.4411	-31.8612	3.0384

* standard deviation of upper bound.

Table 3: comparison of optimal value of the continuous problem and the integer problem

of circa 20%. However, we only obtain little improvement in lower bound value. The main argument is that we still use continuous variables in the backward pass in order to compute the dual value.

The deterministic counterpart value (50) gives a good insight on the gain obtained by using stochastic programming. In the continuous relaxation case, the optimal value obtained is much lower than the deterministic counterpart value, with a gain of circa -30 . Concerning the heuristic method in the stochastic integer problem, although only an upper bound could be generated, it leads to a gain of circa -16.8 with respect to the deterministic counterpart value. This is to be compared to a gain of circa -20.1 by using the cutting plane method. Therefore we conclude that the optimal value of the stochastic integer program lies somewhere between -31.8612 (the continuous relaxation optimal value) and -17.0725 (the integer sub-optimal value).

Nevertheless, building cutting plane is quite expensive: 5 hours per iteration when using $M_f = 1000$ scenarios in forward pass and $M_b = 10$ samples in backward pass.

8 Conclusion

In the article, we describe the critical point of using stage-wise decomposition and dual dynamic programming approach in multi-stage SMILP. The main difficulty comes from the non-convexity and the discontinuity of the Bellman value function $Q^{int}(t, x_t, \xi_t)$ and its conditional expectation $Q^{int}(t, x_t, \xi_{t-1})$ with respect to state variable x_t . We rely on the approximation of the Bellman value function by its convex hull, and try to continue the SDDP approach. However, the approximation discussed here is far from solving exactly multi-stage SMILP problem. The numerical test shows that in the example under consideration, there is still a large gap between the lower and the upper bounds. The cutting plane technique we investigate does yield some improvement in the sub-optimal solution and reduces the gap.

References

- [1] S. Ahmed. Two-stage stochastic integer programming: A brief introduction. In James J. Cochran, Louis A. Cox, Pinar Keskinocak, Jeffrey P. Kharoufeh, and J. Cole Smith, editors, *Wiley Encyclopedia of Operations Research and Management Science*. John Wiley & Sons, Inc., 2010.

- [2] S. Ahmed, M. Tawarmalani, and N.V. Sahinidis. A finite branch-and-bound algorithm for two-stage stochastic integer programs. *Math. Program.*, 100(2, Ser. A):355–377, 2004.
- [3] E. Balas, S. Ceria, G. Cornuéjols, and N. Natraj. Gomory cuts revisited. *Oper. Res. Lett.*, 19(1):1–9, 1996.
- [4] V. Bally and G. Pagès. A quantization algorithm for solving discrete time multi-dimensional optimal stopping problems. *Bernoulli*, 9(6):1003–1049, 2003.
- [5] J.R. Birge and F. Louveaux. *Introduction to Stochastic Programming*. Springer, New York, 2000.
- [6] C.E. Blair and R.G. Jeroslow. The value function of an integer program. *Mathematical Programming*, 23:237–273, 1982. 10.1007/BF01583794.
- [7] J. Bonnans, Zhihao Cen, and Thibault Christel. Energy contracts management by stochastic programming techniques. *Annals of Operations Research*, pages 1–24. online first: 10.1007/s10479-011-0973-5.
- [8] J.F. Bonnans, Z. Cen, and T. Christel. Sensitivity analysis of energy contracts management problem by stochastic programming techniques. In P. Del Moral N. Oudjane R. Carmona, editor, *Numerical Methods in Finance*, Springer Proceeding in Mathematics, to appear.
- [9] E.A. Boyd. Fenchel cutting planes for integer programs. *Oper. Res.*, 42(1):53–64, 1994.
- [10] C.C. Carøe and R. Schultz. Dual decomposition in stochastic integer programming. *Oper. Res. Lett.*, 24(1-2):37–45, 1999.
- [11] G. Cornuéjols. Valid inequalities for mixed integer linear programs. *Math. Program.*, 112(1, Ser. B):3–44, 2008.
- [12] F. Glover and G.A. Kochenberger, editors. *Handbook of metaheuristics*. International Series in Operations Research & Management Science, 57. Kluwer Academic Publishers, Boston, MA, 2003.
- [13] Ralph E. Gomory. An algorithm for integer solutions to linear programs. In *Recent advances in mathematical programming*, pages 269–302. McGraw-Hill, New York, 1963.
- [14] H. Heitsch, W. Römisch, and C. Strugarek. Stability of multistage stochastic programs. *SIAM J. on Optimization*, 17:511–525, 2006.
- [15] H. Klein, K. Willem, and M.H. van der Vlerk. Stochastic integer programming: general models and algorithms. *Annals of Operations Research*, 85:39–57, 1999. 10.1023/A:1018930113099.
- [16] A. Løkketangen and D.L. Woodruff. Progressive hedging and tabu search applied to mixed integer (0,1) multistage stochastic programming. *Journal of Heuristics*, 2:111–128, 1996. 10.1007/BF00247208.

- [17] G. Lulli and S. Sen. A branch and price algorithm for multistage stochastic integer programming with application to stochastic batch sizing problems. *Management Science*, 50:786–796, 2004.
- [18] G. Lulli and S. Sen. A heuristic procedure for stochastic integer programs with complete recourse. *European J. Oper. Res.*, 171(3):879–890, 2006.
- [19] G. Nemhauser and L. Wolsey. *Integer and combinatorial optimization*. Wiley-Interscience Series in Discrete Mathematics and Optimization. John Wiley & Sons Inc., New York, 1999. Reprint of the 1988 original, A Wiley-Interscience Publication.
- [20] L. Ntaimo. Fenchel decomposition for stochastic mixed-integer programming, 2010.
- [21] L. Ntaimo and S. Sen. A comparative study of decomposition algorithms for stochastic combinatorial optimization. *Comput. Optim. Appl.*, 40(3):299–319, 2008.
- [22] R. Schultz, L. Stougie, and M.H. van der Vlerk. Two-stage stochastic integer programming: a survey. *Statist. Neerlandica*, 50(3):404–416, 1996.
- [23] S. Sen. Algorithms for stochastic mixed-integer programming models. In K. Aardal, G.L. Nemhauser, and R. Weismantel, editors, *Handbook of Discrete Optimization*, pages 515–558. North-Holland Publishing Co., 2005.
- [24] S. Sen and J.L. Hige. The C^3 theorem and a D^2 algorithm for large scale stochastic mixed-integer programming: set convexification. *Math. Program.*, 104(1, Ser. A):1–20, 2005.
- [25] S. Sen., J.L. Hige, and J.R. Birge. Duality gaps in stochastic integer programming. *J. of Global Optimization*, 18:189–194, October 2000.
- [26] S. Sen and H.D. Sherali. Decomposition with branch-and-cut approaches for two-stage stochastic mixed-integer programming. *Math. Program.*, 106(2, Ser. A):203–223, 2006.
- [27] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming : Modelling and Theory*. SIAM, Philadelphia, 2009.
- [28] H.D. Sherali and X. Zhu. On solving discrete two-stage stochastic programs having mixed-integer first- and second-stage variables. *Math. Program.*, 108(2-3, Ser. B):597–616, 2006.
- [29] M.H. van der Vlerk. Convex approximations for a class of mixed-integer recourse models. *Ann. Oper. Res.*, 177:139–150, 2010.
- [30] J.P. Walser. *Integer optimization by local search*, volume 1637 of *Lecture notes in artificial intelligence*. Springer-Verlag, Berlin, 1999. A domain independent approach.



**RESEARCH CENTRE
SACLAY – ÎLE-DE-FRANCE**

Parc Orsay Université
4 rue Jacques Monod
91893 Orsay Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399