



LINMA2491 Operational Research

SIMON DESMIDT
ISSAMBRE L'HERMITE DUMONT

Academic year 2024-2025 - Q2



UCLouvain

Contents

1	Definition and notation	2
1.1	Reminders on subgradients	4
2	Modelling	5
2.1	Introduction	5
2.2	Representations	5
2.3	Multi Stage Stochastic Linear Program	6
3	Performance	9
3.1	Notations	9
3.2	Expected value of perfect information	9
3.3	The value of the stochastic solution	10
3.4	Basic inequalities	10
3.5	Bounds on EVPI and VSS	11
3.6	Estimations of WS and EEV	12
4	Benders Decomposition	13
4.1	Cutting plane methods	13
4.2	Context and description	14
4.3	Benders Decomposition Algorithm	17

Definition and notation

- Given Ω , a sigma-algebra \mathcal{A} is a set of subsets of Ω , with the elements called events, such that:
 - $\Omega \in \mathcal{A}$
 - if $A \in \mathcal{A}$ then also $\Omega - A \in \mathcal{A}$
 - if $A_i \in \mathcal{A}$ for $i = 1, 2, \dots$ then also $\cup_{i=1}^{\infty} A_i \in \mathcal{A}$
 - if $A_i \in \mathcal{A}$ for $i = 1, 2, \dots$ then also $\cap_{i=1}^{\infty} A_i \in \mathcal{A}$
- Consider:



- The state space is the set of all values of the system at each stage.

$$S_0 = \{C\}, \quad S_1 = \{C_u, C_d\}, \quad S_2 = \{C_{uu}, C_{ud}, C_{dd}\} \quad (1.1)$$

- The sample space is the set of all possible combination of the system.

$$\Omega = S_0 \times S_1 \times S_2 = \{(C, C_u, C_{uu}), (C, C_u, C_{ud}), (C, C_u, C_{dd}), \dots\} \quad (1.2)$$

- The power set of Ω is the set of all of the subsets, denoted $\mathcal{B}(\Omega)$.
- The probability space is the triplet (Ω, \mathcal{A}, P) where P is a probability measure.
 - $P(\emptyset) = 0$
 - $P(\Omega) = 1$
 - $P(\cup_{i=1}^{\infty} A_i) = \sum_i P(A_i)$ if A_i are disjoint
- $\forall t, A_t$ is the set of events on which we have information at stage t . For example, $A_0 = \{C\}$, $A_1 = \{C, C_u, C_d\}$. Thus is it evident that $t_1 \leq t_2 \Rightarrow \mathcal{A}_{t_1} \subseteq \mathcal{A}_{t_2}$

- Consider the following problem with $x \in \mathbb{R}^n$ and domain \mathcal{D} :

$$\begin{aligned} \min f_0(x), \quad & \text{s.t.} \\ f_i(x) &\leq 0, i = 1, \dots, m \\ h_j(x) &= 0, j = 1, \dots, p \end{aligned} \quad (1.3)$$

Then the Lagrangian function is defined as $L : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$:

$$L(x, \lambda, v) = f_0(x) + \sum_{i=1}^m \lambda_i f_i(x) + \sum_{j=1}^p v_j h_j(x) \quad (1.4)$$

- The Lagrange dual function is defined as $g : \mathbb{R}^m \times \mathbb{R}^p \rightarrow \mathbb{R}$:

$$g(\lambda, v) = \inf_{x \in \mathcal{D}} L(x, \lambda, v) \quad (1.5)$$

- The Lagrange dual problem is a lower bound on the optimal value of the primal problem
- Lagrange relaxation of Stochastic Programs, consider the two problems:

$$\begin{aligned} \min f_1(x) + \mathbb{E}_\omega[f_2(y(\omega), \omega)] & \quad \min f_1(x) + \mathbb{E}_\omega[f_2(y(\omega), \omega)] \\ \text{s.t. } h_{1i}(x) \leq 0, i = 1, \dots, m_1 & \quad \text{s.t. } h_{1i}(x) \leq 0, i = 1, \dots, m_1 \\ h_{2i}(x, y(\omega), \omega) \leq 0, i = 1, \dots, m_2 & \quad h_{2i}(x(\omega), y(\omega), \omega) \leq 0, i = 1, \dots, m_2 \\ & \quad \textcolor{red}{x(\omega) = x} \end{aligned} \quad (1.6)$$

The red constraint is the non-anticipativity constraint, it transforms the deterministic variable into a stochastic variable. **A VERIFIER**

- The dual of a stochastic program is:

$$\begin{aligned} g(v) &= g_1(v) + \mathbb{E}_\omega(g_2(v, \omega)) \\ \text{where} \\ g_1(v) &= \inf f_1(x) + \left(\sum_{\omega \in \Omega} v(\omega) \right)^T x \\ \text{s.t. } h_{1i}(x) &\leq 0, i = 1, \dots, m_1 \\ \text{and} \\ g_2(v, \omega) &= \inf f_2(y(\omega), \omega) - vx(\omega) \\ \text{s.t. } h_{2i}(x(\omega), y(\omega), \omega) &\leq 0, i = 1, \dots, m_2 \end{aligned} \quad (1.7)$$

- With p^* the solution of the primal problem and d^* the solution of the dual problem, we have:

- Weak duality: $d^* \leq p^*$
- Strong duality: $d^* = p^*$

- The KKT conditions are necessary and sufficient for optimality in convex optimization, there aren't unique. They are:

- Primal constraint: $f_i(x) \leq 0, i = 1, \dots, m, h_j(x) = 0, j = 1, \dots, p$
- Dual constraint: $\lambda \geq 0$
- Complementarity slackness: $\lambda_i f_i(x) = 0, i = 1, \dots, m$
- Gradient of the Lagrangian: $\nabla_x L(x, \lambda, \nu) = 0$
- An extreme point of a polyhedron P is a point $x \in P$ such that it cannot be expressed as a linear combination of two distinct points in P , i.e. an extreme point is a vertex of the polyhedron.
- An extreme ray of a polyhedron P is $\sigma \in \mathbb{R}^n$ such that for all $x \in P$, for all $\lambda \in [0, 1]$,

$$(x + \lambda\sigma) \in P \quad (1.8)$$

i.e. it is a direction in which we can travel infinitely without leaving the polyhedron.

1.1 Reminders on subgradients

π is a subgradient of the function g at u if

$$g(w) \geq g(u) + \pi^T(w - u) \quad \forall w \quad (1.9)$$

If $g = \max\{g_1, g_2\}$ with $g_{1,2}$ convex and differentiable, the subgradient of g at u_0 is

- $\pi = \nabla g_1(u_0)$ if $g_1(u_0) > g_2(u_0)$
- $\pi = \nabla g_2(u_0)$ if $g_2(u_0) > g_1(u_0)$
- The line segment $[\nabla g_1(u_0), \nabla g_2(u_0)]$ if $g_1(u_0) = g_2(u_0)$

The subdifferential of g at u is the set of all subgradients of g at u , denoted $\partial g(u)$. If g is convex, then its subdifferential is nonempty on its domain, and g is differentiable at u if its $\partial g(u) = \{\pi\}$.

1.1.1 Use in duality

Define $c(u)$ as the optimal value of

$$\begin{aligned} c(u) &= \min f_0(x) \\ f_i(x) &\leq u_i \quad i = 1, \dots, m \end{aligned} \quad (1.10)$$

where $x \in \text{dom} f_0$ and f_0, f_i are convex functions.

- $c(u)$ is convex;
- If strong duality holds, denote λ^* as the maximizer of the dual function

$$\inf_{x \in \text{dom} f_0} (f_0(x) - \lambda^T(f(x) - u)) \quad (1.11)$$

for $\lambda \leq 0$. Then, $\lambda^* \in \partial c(u)$. λ_i represents the sensitivity of $c(u)$ to a marginal change in the right-hand side of the i -th constraint.

Modelling

2.1 Introduction

- For a certain sequence of events $x \rightarrow \omega \rightarrow y(\omega)$, where ω is the uncertainty,
 - A first-stage decision is a decision that is made before the uncertainty is revealed (i.e. in x);
 - A second-stage decision is a decision that is made after the uncertainty is revealed (i.e. in $y(\omega)$).
- We can have the following mathematical formulation:

$$\begin{aligned}
 \min \quad & c^T x + \mathbb{E}[\min q(\omega)^T y(\omega)] \\
 \text{s.t.} \quad & Ax = b \\
 & T(\omega)x + W(\omega)y(\omega) = h(\omega) \\
 & x \geq 0, y(\omega) \geq 0
 \end{aligned} \tag{2.1}$$

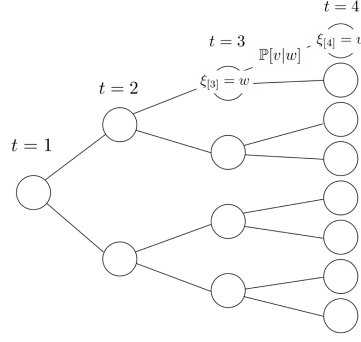
- First-stage decision variable: $x \in \mathbb{R}^{n_1}$
- First-stage parameter: $c \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{m_1}$ and $A \in \mathbb{R}^{m_1 \times n_1}$
- Second-stage decision: $y(\omega) \in \mathbb{R}^{n_2}$
- Second-stage data: $q(\omega) \in \mathbb{R}^{n_2}$, $h(\omega) \in \mathbb{R}^{m_2}$ and $T(\omega) \in \mathbb{R}^{m_2 \times n_1}$, $W(\omega) \in \mathbb{R}^{m_2 \times n_2}$

2.2 Representations

2.2.1 Scenario Trees

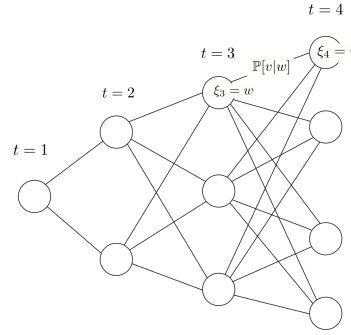
A scenario tree is a graphical representation of a Markov process $\{\xi_t\}_{t \in \mathbb{Z}}$, where the nodes are the history of realizations ($\xi_{[t]} = (\xi_1, \dots, \xi_t)$), and the edges are the transitions from $\xi_{[t]}$ to $\xi_{[t+1]}$.

- We denote the root as $t = 1$;
- An ancestor of a node $\xi_{[t]}$, $A(\xi_{[t]})$ is a unique adjacent node which precedes ξ_t ;
- The children of a node, $C(\xi_{[t]})$ are the nodes that are adjacent to $\xi_{[t]}$ and occur at stage $t + 1$.



2.2.2 Lattice

A lattice is a graphical representation of a Markov process $\{\xi_t\}_{t \in \mathbb{Z}}$, where the nodes are the realizations ξ_t and the edges correspond to the transitions from ξ_t to ξ_{t+1} .



2.2.3 Serial Independence

A process satisfies serial independence if, for every stage t , ξ_t has a probability distribution that does not depend on the history of the process. Thus, the probability measure is

$$\mathbb{P} \left[\xi_t(\omega) = i \mid \xi_{[t-1]}(\omega) \right] = p_t(i) \quad \forall \xi_{[t-1]} \in \Xi_{[t-1]}, i \in \Xi_t \quad (2.2)$$

2.3 Multi Stage Stochastic Linear Program

2.3.1 Notation

- Probability space: $(\Omega, 2^\Omega, \mathbb{P})$ with filtration $\{\mathcal{A}\}_{t \in \{1, \dots, H\}}$
- $c_t(\omega) \in \mathbb{R}^{n_t}$: cost coefficients
- $h_t(\omega) \in \mathbb{R}^{m_t}$: right-hand side parameters
- $W_t(\omega) \in \mathbb{R}^{m_t \times n_t}$: coefficients of $x_t(\omega)$
- $T_{t-1}(\omega) \in \mathbb{R}^{m_t \times n_{t-1}}$: coefficients of $x_{t-1}(\omega)$
- $x_t(\omega)$: set of state and action variables in period t

- We implicitly enforce non-anticipativity by requiring that x_t and ξ_t are adapted to filtration $\{\mathcal{A}\}_{t \in \{1, \dots, H\}}$
- $\forall A \in \mathcal{A}_k \setminus \mathcal{A}_{k-1}, x_t(\omega_1) = x_t(\omega_2) \forall \omega_1, \omega_2 \in A$

2.3.2 General formulation of the MSLP

The extended formulation of the MSLP is:

$$\begin{aligned}
& \min c_1^T x_1 + \mathbb{E}[c_2(\omega)^T x_2(\omega) + \dots + c_H(\omega)^T x_H(\omega)] \\
& \text{s.t. } W_1 x_1 = h_1 \\
& T_1(\omega) x_1 + W_2(\omega) x_2(\omega) = h_2(\omega), \omega \in \Omega \\
& \quad \vdots \\
& T_{t-1}(\omega) x_{t-1}(\omega) + W_t(\omega) x_t(\omega) = h_t(\omega), \omega \in \Omega \\
& \quad \vdots \\
& T_{H-1}(\omega) x_{H-1}(\omega) + W_H(\omega) x_H(\omega) = h_H(\omega), \omega \in \Omega \\
& x_1 \geq 0, x_t(\omega) \geq 0, t = 2, \dots, H
\end{aligned} \tag{2.3}$$

We can now consider two specific instantiations of the MSLP: the scenario tree (MSLP-ST) and the lattice (MSLP-L). Using these notations:

- $\omega_t \in S_t$: index in the support Ξ_t of random input ξ_t
- $\omega_{[t]} \in S_1 \times \dots \times S_t$ (interpretation: index in $\Xi_{[t]} = \Xi_1 \times \dots \times \Xi_t$, which is the history of realizations, up to period t)

2.3.3 Scenario Tree formulation

$$\begin{aligned}
& \min c_1^T x_1 + \mathbb{E} \left[c_2(\omega_{[2]})^T x_2(\omega_{[2]}) + \dots + c_H(\omega_{[H]})^T x_H(\omega_{[H]}) \right] \\
& \text{s.t. } W_1 x_1 = h_1 \\
& T_1(\omega_{[2]}) x_1 + W_2(\omega_{[2]}) x_2(\omega_{[2]}) = h_2(\omega_{[2]}), \omega_{[2]} \in S_1 \times S_2 \\
& \quad \vdots \\
& T_{t-1}(\omega_{[t]}) x_{t-1}(\omega_{[t-1]}) + W_t(\omega_{[t]}) x_t(\omega_{[t]}) = h_t(\omega_{[t]}), \omega_{[t]} \in S_1 \times \dots \times S_t \\
& \quad \vdots \\
& T_{H-1}(\omega_{[H]}) x_{H-1}(\omega_{[H-1]}) + W_H(\omega_{[H]}) x_H(\omega_{[H]}) = h_H(\omega_{[H]}), \omega_{[H]} \in S_1 \times \dots \times S_H \\
& x_1 \geq 0, x_t(\omega_{[t]}) \geq 0, t = 2, \dots, H
\end{aligned} \tag{2.4}$$

2.3.4 Lattice formulation

$$\begin{aligned}
& \min c_1^T x_1 + \mathbb{E} \left[c_2(\omega_2)^T x_2(\omega_{[2]}) + \cdots + c_H(\omega_H)^T x_H(\omega_{[H]}) \right] \\
& s.t. \quad W_1 x_1 = h_1 \\
& \quad T_1(\omega_2) x_1 + W_2(\omega_2) x_2(\omega_{[2]}) = h_2(\omega_2), \omega_{[2]} \in S_1 \times S_2 \\
& \quad \quad \quad \vdots \\
& \quad T_{t-1}(\omega_t) x_{t-1}(\omega_{[t-1]}) + W_t(\omega_t) x_t(\omega_{[t]}) = h_t(\omega_t), \omega_{[t]} \in S_1 \times \cdots \times S_t \\
& \quad \quad \quad \vdots \\
& \quad T_{H-1}(\omega_H) x_{H-1}(\omega_{[H-1]}) + W_H(\omega_H) x_H(\omega_{[H]}) = h_H(\omega_H), \omega_{[H]} \in S_1 \times \cdots \times S_H \\
& \quad x_1 \geq 0, x_t(\omega_{[t]}) \geq 0, t = 2, \dots, H
\end{aligned} \tag{2.5}$$

→ Note: There exists some relations to other decision making problems such as statistical decision theory, dynamic programming, online optimization and stochastic control.

Performance

3.1 Notations

Using (2.1), let's define the following:

- $z(x, \xi) = c^T x + Q(x, \xi) + \delta(x|K_1)$
- $Q(x, \xi) = \min_y \{q(\omega)^T y \mid W(\omega)y = h(\omega) - T(\omega)x\}$
- $K_1 = \{x \mid Ax = b, x \geq 0\}$ is the set of feasible first-stage decisions
- $K_2(\omega) = \{x \mid \exists y \geq 0 : W(\omega)y = h(\omega) - T(\omega)x\}$ is the set of first-stage decisions that have a feasible reaction in the second stage for $\omega \in \Omega$
- It is possible that $z(x, \xi) = +\infty$ (if $x \notin K_1 \cap K_2(\omega)$)
- It is possible that $z(x, \xi) = -\infty$ (unbounded below)

3.2 Expected value of perfect information

There are 2 tactics:

- **wait-and-see** value is the expected value of reacting with perfect foresight (we know everything that will happen) $x^*(\xi)$ to ξ :

$$WS = \mathbb{E}[\min_x z(x, \xi)] = \mathbb{E}[z(x^*(\xi), \xi)] \quad (3.1)$$

- **here-and-now** value is the expected value of the recourse problem (remove non-anticipativity constraint):

$$SP = \min_x \mathbb{E}[z(x, \xi)] \quad (3.2)$$

The **expected value of perfect information** is like the value we give to getting a perfect forecast for the future and is thus defined like this:

$$EVPI = SP - WS \quad (3.3)$$

3.3 The value of the stochastic solution

Here too there are 2 tactics:

- **expected value problem**

$$EV = \min_x z(x, \bar{\zeta}) = \mathbb{E}[\bar{\zeta}] \quad (3.4)$$

and its **expected value solution** is noted $x^*(\bar{\zeta})$.

- **expected value of using the EV solution** measures the performance of $x^*(\bar{\zeta})$:

$$EEV = \mathbb{E}[z(x^*(\bar{\zeta}), \zeta)] \quad (3.5)$$

The **value of the stochastic solution** is noted like this:

$$VSS = EEV - SP \quad (3.6)$$

3.4 Basic inequalities

3.4.1 Crystal Ball

For every ζ , we have $z(x^*(\bar{\zeta}), \zeta) \leq z(x^*, \zeta)$ where x^* is the optimal solution to the stochastic program. And if we take the expectation of this inequality, we have $WS \leq SP$, because WS is a relaxation. It explains that we can do better with a crystal ball.

3.4.2 Lazy solution

Knowing that x^* is the optimal solution of $\min_x \mathbb{E}[z(x, \zeta)]$ and $x^*(\bar{\zeta})$ is a solution but not necessarily optimal then we have $SP \leq EEV$, because:

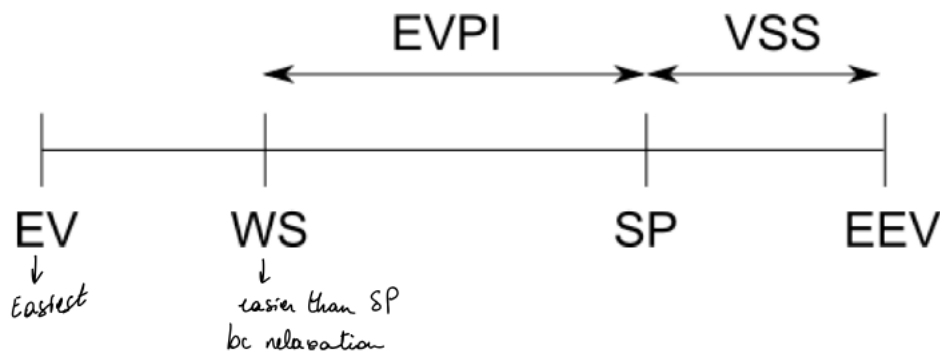
$$\min_x \mathbb{E}[z(x, \zeta)] = SP \leq EEV = \mathbb{E}[z(x^*(\bar{\zeta}), \zeta)] \quad (3.7)$$

3.4.3 Link between all the values

We know that:

- $VSS \geq 0$
- $EVPI \leq EEV - EV$
- $EVPI \geq 0$
- If $EEV - EV = 0$ then $VSS = EVPI = 0$
- $VSS \leq EEV - EV$

and the inequalities can be summarized in the following diagram:



3.5 Bounds on EVPI and VSS

First let's introduce the pairs subproblem of ξ^r and ξ^k :

$$\begin{aligned} \min z^P(x, \xi^r, \xi^k) &= c^T x + p^r q^T y(\xi^r) + (1 - p^r) q^T y(\xi^k) \\ \text{s.t. } Ax &= b \\ Wy(\xi^r) &= \xi^r - Tx \\ Wy(\xi^k) &= \xi^k - Tx \\ x, y &\geq 0 \end{aligned} \tag{3.8}$$

- $(\bar{x}^k, \bar{y}^k, y(\xi^k))$ denotes an optimal solution to the problem and z^k is the optimal objective function value $z^P(\bar{x}^k, \bar{y}^k, y(\xi^k))$
- $z^P(x, \xi^r, \xi^r)$ corresponds to the deterministic optimization against the reference scenario
- if $\xi^r \notin \Xi$, $p^r = 0$ and $z^P(x, \xi^r, \xi^k) = z(x, \xi^k)$

The **sum of pairs expected value (SPEV)**:

$$SPEV = \frac{1}{1 - p^r} \sum_{k=1, k \neq r}^K p^k \min z^P(x, \xi^r, \xi^k) \tag{3.9}$$

When $\xi^r \notin \Xi$ then $SPEV = WS$: When $p^r = 0$, $z^P(x, \xi^r, \xi^k)$ coincides with $z(x, \xi^k)$. Therefore $SPEV = \sum_{k=1}^K p^k \min_x z(x, \xi^k) = WS$. We then know $WS \leq SPEV \leq SP$.

3.5.1 Upper bound on SP: EVRS and EPEV

- The **expected value of the reference scenario** is $EVRS = \mathbb{E}_{\xi}(\bar{x}^r, \xi)$, where \bar{x}^r is the optimal solution to $z(x, \xi^r)$.
- The **expectation of pairs of expected value** is defined as

$$EPEV = \min_{k=1, \dots, K \cup \{r\}} \mathbb{E}_{\xi}(\bar{x}^r, \xi)$$

where $(\bar{x}^k, \bar{y}^k, y(\xi^k))$ is the optimal solution to the pairs subproblem of ξ^r and ξ^k .

As $SP, EPEV, EVRS$ are the optimal values of $\min_x \mathbb{E}_{\xi} z(x, \xi)$ over smaller feasible sets:

$$SP \leq EPEV \leq EVRS \tag{3.10}$$

Because

- SP : $x \in K_1 \cap K_2$
- $EPEV$: $x \in K_1 \cap K_2 \cap \{\bar{x}^k, k = 1, \dots, K \cup \{r\}\}$
- $EVRS$: $x \in \bar{x}^r \cap K_1 \cap K_2$

3.6 Estimations of WS and EEV

An estimation of WS and EEV can be done through a sample mean approximation: from samples $\tilde{\xi}_i = \tilde{\xi}(\omega_i)$ for $i = 1, \dots, K$,

1. Compute $x^*(\tilde{\xi})$;
2. Compute $WS_i = z(x^*(\tilde{\xi}_i), \tilde{\xi}_i)$ and $EEV_i = c^T x^*(\tilde{\xi}_i) + Q(x^*(\tilde{\xi}), \tilde{\xi}_i)$;
3. Estimate $\bar{WS} = \frac{1}{K} \sum_{i=1}^K WS_i$ and $\bar{EEV} = \frac{1}{K} \sum_{i=1}^K EEV_i$.

3.6.1 Central Limit Theorem

Suppose $\{X_1, \dots, X_K\}$ is a sequence of iid rv with $\mathbb{E}[X_i] = \mu$ and $\text{Var}[X_i] = \sigma^2 < \infty$. Then, as n approaches infinity, $\sqrt{n}(S_n - \mu)$ converge in distribution to a normal $\mathcal{N}(0, \sigma^2)$:

$$\sqrt{n} \left(\left(\frac{1}{n} \sum_{i=1}^n X_i \right) - \mu \right) \xrightarrow{d} \mathcal{N}(0, \sigma^2) \quad (3.11)$$

The central limit theorem is useful to decrease the importance of rare but extreme events.

3.6.2 Importance sampling

Suppose we wish to estimate $\mathbb{E}[C(\omega)]$, where ω is distributed according to $f(\omega)$ and estimates $\mathbb{E}[C(\omega)]$ with $\sum_{i=1}^N \frac{1}{N} C(\omega_i)$. A sample average pulls samples ω_i according to the distribution function $f(\omega)$, while the importance sampling pulls the samples ω_i according to the distribution $g(\omega) = \frac{f(\omega)C(\omega)}{\mathbb{E}[C(\omega)]}$, where the $\mathbb{E}[C(\omega)]$ is an approximation of the real expectation. It then estimates $\mathbb{E}[C(\omega)]$ with $\sum_{i=1}^N \frac{1}{N} \frac{f(\omega_i)C(\omega_i)}{g(\omega_i)}$.

Benders Decomposition

4.1 Cutting plane methods

A cutting plane method is an optimisation method based on the idea of iteratively refining the objective function, or a set of feasible constraints of a problem through linear inequalities (see LINMA2450).

4.1.1 Nomenclature

- The benders decomposition is a specific method for obtaining the cutting planes when $F(x)$ is the value function of a second-stage linear program.
- The L-shaped method is a specific instance of Benders decomposition when the second-stage linear program is decomposable into a set of scenarios.
- The multi-cut L-shaped method is an alternative to the L-shaped method which generates multiple cutting planes at step 1 of Kelley's method (see 4.1.2).

4.1.2 Kelley's Cutting Plane Algorithm

This algorithm is designed to solve convex but non-differentiable optimization problems of the form

$$\begin{aligned} z^* &= \min c^T x + F(x) \\ \text{s.t. } x &\in X \end{aligned} \tag{4.1}$$

where $X \subseteq \mathbb{R}^n$ is convex and compact, $F : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex and $c \in \mathbb{R}^n$ is a parameter vector.

Let us define

- $L_k : \mathbb{R}^n \rightarrow \mathbb{R}$ a lower bound function of $F(x)$ at iteration k ;
- A lower bound L_k of z^* at iteration k ;
- An upper bound U_k of z^* at iteration k .

Algorithm 1 Kelley's Cutting plane algorithm

- 1: **Step 0:** Set $k = 0$ and assume $x_1 \in X$ is given. Set $L_0(x) = -\infty$ for all $x \in X$, $U_0 = c^T x_1 + F(x_1)$, and $L_0 = -\infty$.
2: **Step 1:** Set $k = k + 1$. Find $a_k \in \mathbb{R}$ and $b_k \in \mathbb{R}^n$ such that

$$F(x_k) = a_k + b_k^T x_k$$

$$F(x) \geq a_k + b_k^T x \quad x \in X$$

- 3: **Step 2:** Set

$$U_k = \min(U_{k-1}, c^T x_k + F(x_k))$$

and

$$L_k(x) = \max(L_{k-1}(x), a_k + b_k^T x) \quad x \in X$$

- 4: **Step 3:** Compute

$$L_k = \min_{x \in X} L_k(x) + c^T x$$

and denote x_{k+1} as the optimal solution of this problem.

- 5: **Step 4:** If $U_k - L_k = 0$, stop. Otherwise, go to step 1.
-

4.2 Context and description

Consider the following optimization problem:

$$\begin{aligned} z^* &= \min c^T x + q^T y \\ Ax &= b \\ Tx + Wy &= h \\ x, y &\geq 0 \end{aligned} \tag{4.2}$$

with $x \in \mathbb{R}^{n_1}$, $y \in \mathbb{R}^{n_2}$, $c \in \mathbb{R}^{n_1}$, $q \in \mathbb{R}^{n_2}$, $A \in \mathbb{R}^{m_1 \times n_1}$, $b \in \mathbb{R}^{m_1}$, $T \in \mathbb{R}^{m_2 \times n_1}$, $W \in \mathbb{R}^{m_2 \times n_2}$, $h \in \mathbb{R}^{m_2}$ ¹.

We use Benders decomposition when the entire problem is difficult to solve, and if the constraint $Tx + Wy = h$ is ignored, the problem becomes easy to solve, or if fixing x simplifies the computation of the solution.

4.2.1 Idea of Benders decomposition

Define the value function $V : \mathbb{R}^{n_1} \rightarrow \mathbb{R}$:

$$\begin{aligned} (S) : V(x) &= \min_y q^T y \\ Wy &= h - Tx \\ y &\geq 0 \end{aligned} \tag{4.3}$$

¹It is not necessarily a stochastic problem

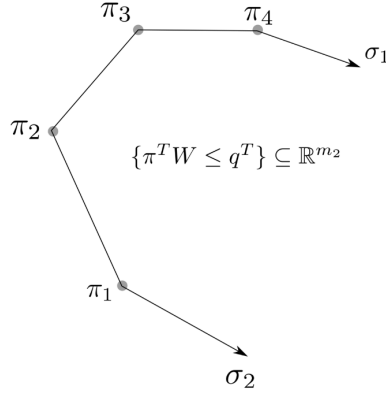
Or equivalently,

$$\begin{aligned}
\min \quad & c^T x + V(x) \\
\text{s.t.} \quad & Ax = b \\
& x \in \text{dom}(V) \\
& x \geq 0
\end{aligned} \tag{4.4}$$

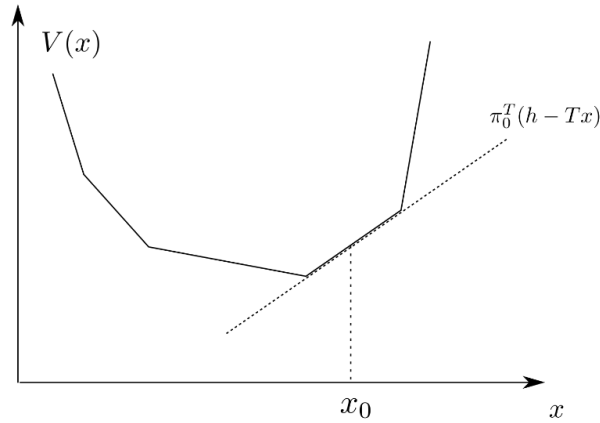
where $\text{dom}(V) = \{x \in \mathbb{R}^{n_1} \mid \exists y \geq 0 : Wy = h - Tx\}$.
The dual of (4.3) is

$$\begin{aligned}
\max_{\pi} \quad & \pi^T (h - Tx) \\
\text{s.t.} \quad & \pi^T W \leq q^T
\end{aligned} \tag{4.5}$$

Let us call E the set of extreme points of $\pi^T W \leq q^T$ and R the set of extreme rays of $\pi^T W \leq q^T$ (see (1.8) for definitions).



We can see that $V(x)$ is a piecewise linear convex function of x and, defining x_0 as the dual optimal multiplier of (4.3) given x_0 , then $\pi_0^T (h - Tx_0)$ is a supporting hyperplane of $V(x)$ at x_0 , because it belongs to the subdifferential of $V(x)$ at $h - Tx_0$.



From this, we can also express the domain of V as follows:

$$\text{dom}(V) = \{x \mid \sigma^T (h - Tx) \leq 0, \sigma \in R\} \tag{4.6}$$

where $\sigma \in R$ is the set of extreme rays of $\pi^T W \leq q^T$.

→ Note: when a domain is unbounded in a direction that does not improve the objective value, it is not a problem to its resolution.

4.2.2 Reformulation

The objective of the reformulation is to find a general form for the algorithm. That way, each iteration simply adds constraints of the same form, involving the minimum number of changes to the problem.

$$\begin{aligned}
 & \min c^T x + \theta \\
 & Ax = b \\
 & \sigma_r^T (h - Tx) \leq 0 \quad \sigma_r \in R \\
 & \theta \geq \pi_e^T (h - Tx) \quad \pi_e \in E \\
 & x \geq 0
 \end{aligned} \tag{4.7}$$

where θ is a free variable.

The idea is to relax some inequalities that define $V(x)$ and $\text{dom } V$:

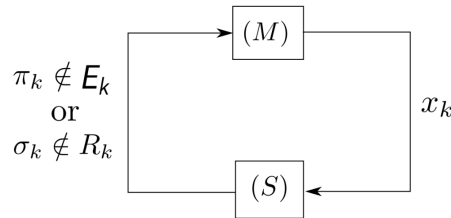
$$\begin{aligned}
 (M) : & \quad z_k = \min c^T x + \theta \\
 & \quad Ax = b \\
 & \quad \sigma_r^T (h - Tx) \leq 0 \quad \sigma_r \in R_k \subseteq R \\
 & \quad \theta \geq \pi_e^T (h - Tx) \quad \pi_e \in E_k \subseteq E \\
 & \quad x \geq 0 \\
 (S) : & \quad V(\bar{x}) = \min_{x,y} q^T y \\
 & \quad Wy = h - Tx \\
 & \quad x = \bar{x} \\
 & \quad y \geq 0
 \end{aligned} \tag{4.8}$$

The solution of the main problem (M) above provides:

- A lower bound $z_k \leq z^*$;
- A candidate solution x_k ;
- An under-estimator of $V(x_k)$, $\theta_k \leq V(x_k)$.

The solution of the subproblem (S) with input x_k provides:

- An upper bound $c^T x_k + q^T y_{k+1} \geq z^*$;
- A new vertex π_{k+1} or a new extreme ray σ_{k+1} .



In addition to the two problems defined above, we have the dual of (S):

$$\begin{aligned}
 (D) : & \quad \max_{\pi, \lambda} \lambda^T x + \pi^T h \quad \pi^T W \leq g^T \\
 & \quad \pi^T T + \lambda = 0
 \end{aligned} \tag{4.10}$$

With this, the formulation of the optimality cut becomes

$$\theta \geq \lambda^T (x - \bar{x}) + V(\bar{x}) \tag{4.11}$$

4.3 Benders Decomposition Algorithm

Algorithm 2 Benders Decomposition Algorithm

```
1: Step 0: Set  $k = 0$ ,  $E_0 = R_0 = \emptyset$ ;  
2: Step 1: Solve (M);  
3: if (M) is feasible then  
4:   Store  $x_k$ ;  
5: else  
6:   break;  
7: end if  
8: Step 2: Solve (S) (or (D)) with  $x_k$  as input;  
9: if (S) is infeasible then  
10:   Let  $R_{k+1} = R_k \cup \{\sigma_{k+1}\}$ ;  
11:    $k \leftarrow k + 1$ ;  
12: else  
13:   Let  $E_{k+1} = E_k \cup \{\pi_{k+1}\}$ ;  
14:   if  $E_{k+1} = E_k$  then  
15:     terminate with  $(x_k, y_{k+1})$  as optimal solution;  
16:   else  
17:     Let  $k \leftarrow k + 1$  and back to step 1.  
18:   end if  
19: end if
```

→ Note: The algorithm takes finite time, as E and R are finite.