Solution - L-shaped method

24-03-2020

Decomposition for the wind power producer problem

Take the wind power producer example from Lecture 03 (Multi-stage stochastic programming). Assume we only trade in the day-ahead market and balancing market, i.e., we result in a two-stage stochastic program (page 9, lecture 03).

Sets	
$\mathcal{S}$	Set of scenarios $s \in \mathcal{S}$
Parameters	
λ^D	Electricity price on the day-ahead market
λ^+	Selling electricity price on the balancing market
λ^-	Purchasing electricity price on the day-ahead market
$\overline{P}$	Capacity of the wind farm
W_s	Wind power production in scenario $s \in \mathcal{S}$
π_s	Probability scenario $s \in \mathcal{S}$
Variables	
p^D	Power sold on the day-ahead market [MWh]
p^+	Excess production sold on the balancing market [MWh]
p^-	Missing production bought on the balancing market [MWh]

$$\text{Max} \ \ \lambda^D p^D + \sum_{s \in \mathcal{S}} \pi_s (\lambda^+ p_s^+ - \lambda^- p_s^-) \tag{1a}$$

$$\text{s.t.} \ \ p^D \leq \overline{P} \tag{1b}$$

$$W_s - p^D = p_s^+ - p_s^- \qquad \forall s \in \mathcal{S} \tag{1c}$$

$$p^D \in \mathbb{R}^+ \tag{1d}$$

$$p_s^+, p_s^- \in \mathbb{R}^+ \qquad \forall s \in \mathcal{S} \tag{1e}$$

The objective function (1a) maximizes the profit based on the transactions on the day-ahead and balancing market. Constraint (1b) restricts the amount offered in the day-ahead market to the capacity of the wind farm. Constraint (1c) determines the imbalances compared to the day-ahead market offer based on the realization of the scenario. In constraints (1d) and (1e) the non-negativity of the variables is ensured.

Tasks

1. Decompose the problem in master problem and subproblems.
2. Based on the general formulation of the model, derive the formula for the optimality cut for the L-shaped method (single-cut version). This means define how the coefficients E_i and e_i ($i \in \mathcal{I}$ being the set of cuts) are calculated based on the parameters and variables in the master and subproblems and write down the inequality for the cut based on E_i and e_i.
3. Do we need infeasibility cuts for this model? Justify your answer.

Solution

1)

Master problem

$$
\begin{aligned}
\text{Max} \;\; & \lambda^D p^D + \theta && \text{(2a)}\\
\text{s.t.} \;\; & p^D \leq \overline{P} && \text{(2b)}\\
& E_i p^D + \theta \leq e_i \qquad \forall i \in \mathcal{I} && \text{(2c)}\\
& p^D \in \mathbb{R}^+ && \text{(2d)}
\end{aligned}
$$

Set $\mathcal{I}$ is the set of cuts.
ATTENTION! Cut (2c) is now a $\leq$-constraint, because we have a maximization problem.

Subproblem

for each scenario $s \in \mathcal{S}$:

$$
\begin{aligned}
\text{Max} \;\; & (\lambda^+ p_s^+ - \lambda^- p_s^-) && \text{(3a)}\\
\text{s.t.} \;\; & p_s^+ - p_s^- = W_s - p_{fix}^D \qquad : \sigma_s && \text{(3b)}\\
& p_s^+, p_s^- \in \mathbb{R}^+ && \text{(3c)}
\end{aligned}
$$

p_{fix}^D is the fixed first-stage solution.

2)

We define the cut coefficients based on the dual subproblem with σ_s being the dual variable based on constraint (3b).

Objective function of the dual problem:

$$\text{Min } [W_s - p_{fix}^D]\sigma_s$$

Take the expected value to approximate the recourse function over all scenarios:

$$
\begin{aligned}
& \sum_{s \in S} \pi_s [W_s - p_{fix}^D]\sigma_s \\
\Leftrightarrow & \sum_{s \in S} \pi_s W_s \sigma_s - \sum_{s \in S} \pi_s p_{fix}^D \sigma_s
\end{aligned}
$$

New values of the recourse function are bounded from above by this value (maximization). Thus, the cut inequality is given by:

$$
\begin{aligned}
& \theta \leq \sum_{s \in S} \pi_s W_s \sigma_s - \sum_{s \in S} \pi_s p^D \sigma_s \\
\Leftrightarrow & \underbrace{\sum_{s \in S} \pi_s p^D \sigma_s}_{E_i} + \theta \leq \underbrace{\sum_{s \in S} \pi_s W_s \sigma_s}_{e_i}
\end{aligned}
$$

3)

We do not need infeasibility cuts in this case, because we have a complete recourse, i.e., first-stage decisions never result in infeasible solutions in the subproblems. This is due to the fact that any imbalances between the parameter W_s and first-stage variable p^D can be captured by the variables p_s^+ and p_s^-, which are basically slack variables.