Integer Programming

Lecture 20

Variable Decomposition

- Recall the basic principle of decomposition: by relaxing/fixing the linking variables/constraints, we then get a model that is easier to solve.
- Here, we discuss methods of decomposing by fixing *complicating variables*.
- "Classical" decomposition arises from *block structure* in the constraint matrix.

$$\begin{pmatrix} A_{10} & A_{11} & & & \\ A_{20} & & A_{22} & & \\ \vdots & & & \ddots & \\ A_{\gamma 0} & & & & A_{\kappa\kappa} \end{pmatrix}$$

- After fixing variables the problem becomes separable and the separability lends itself nicely to parallel implementation.
- However, there can be other reasons why problems become easier to solve upon fixing certain variables.

(Generalized) Benders' Decomposition

- Most of what we're referring to as *variable decomposition* methods are derivatives of an algorithm proposed by Benders.
- Benders' original method was for the case of LPs, but the algorithm is easy to generalize.
- From a mathematical standpoint, Benders' method amounts to projection of the problem into the space of a subset of the variables.
- The projection effectively amounts to a reformulation of the problem in terms of the value function of a restriction of the problem.

Benders' Principle (Linear Programming)

$$z_{\mathrm{LP}} = \max_{(x,y)\in\mathbb{R}^n} \{cx + dy \mid Ax \le b, Dx + Gy \le d\}$$
$$= \max_{x\in\mathbb{R}^{n'}} \{cx + \phi(d - Dx) \mid Ax \le b\},$$

where

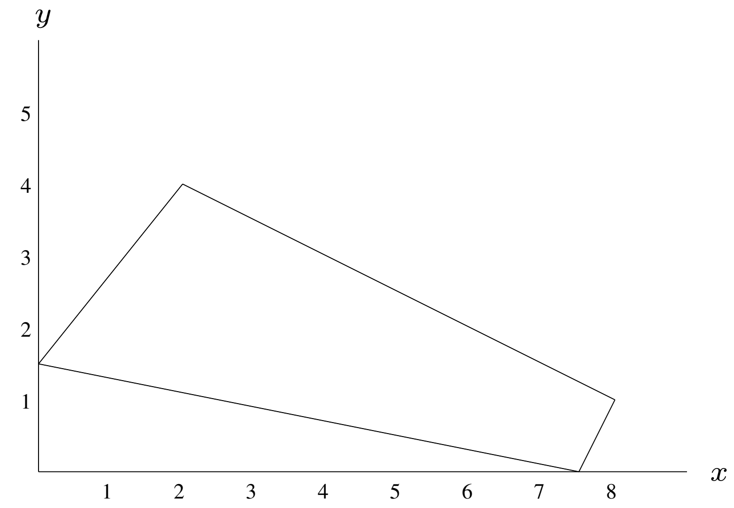
$$\phi(\beta) = \max dy$$
$$\text{s.t. } Gy \le \beta$$
$$y \in \mathbb{R}^{n''}$$

Basic Strategy:

- The function ϕ is the value function of a linear program.
- We iteratively approximate it by generating *dual functions*.

Example

$$
\begin{aligned}
z_{LP} \quad = \quad \max \quad & x - y \\
\text{s.t.} \quad & -25x + 20y \leq 30 \\
& x + 2y \leq 10 \\
& 2x - y \leq 15 \\
& -2x - 10y \leq -15 \\
& x \in [0, 10] \\
& y \in [0, 5]
\end{aligned}
$$

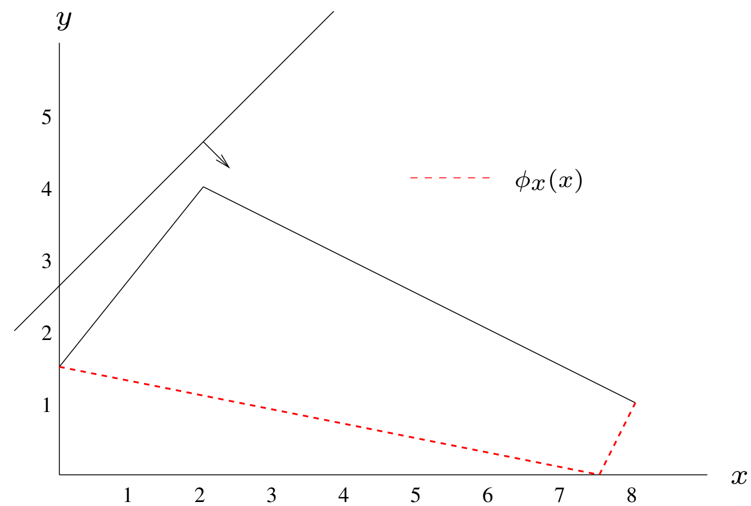


Value Function Reformulation

$$z_{LP} = \max_{0 \leq x \leq 10} x + \phi_x(x),$$

where

$$\begin{aligned} \phi_x(x) = \max \quad & -y \\ \text{s.t.} \quad & 20y \leq 30 + 25x \\ & 2y \leq 10 - x \\ & -y \leq 15 - 2x \\ & -10y \leq -15 + 2x \\ & y \in [0, 5] \end{aligned}$$



- Note that $\phi_x(x) = \phi(d - Dx)$ and is not the value function itself.
- Also, it only coincides with the boundary of the feasible region because of the specific objective funcion in this case.
- The reformulated problem can be interpreted precisely as the projection into the space of the first set of variables.

Generalized Benders

Benders' Master Problem (iteration k)

$$\begin{aligned} \max \quad & cx + z \\ \text{subject to} \quad & Ax \le b \\ & z \le \overline{\phi}_i(d - Dx), 1 \le i \le k \\ & x \in \mathbb{Z}^{n'} \end{aligned}$$

Basic Scheme

- Solve master problem to obtain new candidate solution x^k and lower bound.
- Solve subproblem by evaluating $\phi(d - Dx^k)$ to obtain $\overline{\phi}_k$ (dual function and new upper bound.
- Terminate when upper bound equals lower bound.

Where do we get $\overline{\phi}_k$?

Benders Optimality Cuts

- $\overline{\phi}_k$ is a *dual function* that we construct by evaluating $\phi(d - Dx^k)$.
- The dual functions arising in each iteration are combined into a global dual function through the constraints on z.
- Each evaluation of ϕ yields information that we can use to build up this overall global approximation.
- In the LP case, the dual functions are linear functions that arise as the dual solutions to the subproblems.
- The constraint $z \leq \overline{\phi}_i(d - Dx)$ added in iteration i reduce to $z \leq {u^i}^\top(d - Dx)$, where u^i is the dual solution to the subproblem.
- These are linear inequalities and Benders can hence be seen as a cutting plane method in this case.

Benders Feasibility Cuts

- Note that it can happen that the subproblem is infeasible.
- This is accounted for in the general algorithm by the fact that the value function is defined over the extended reals.
- We define $\phi_x(x) = -\infty$ when there is no y such that $Gy \leq d - Dx$.
- In the master problem we are disallowing x such that $\phi_x(x) = -\infty$.
- In practical computations, we need constraints to enforce this.
- In the LP case, when $\phi_x(x) = -\infty$, then the proof of infeasibility is a ray r of the dual feasible region that proves unboundedness.
- In other words, the proof is a dual ray r such that $r^\top(d - Dx) < 0$.
- Thus, we can disallow this value of x in the master by adding the constraint $r^\top(d - Dx) \geq 0$.
- In the LP case, such constraints are the so-called *Benders' feasibility cuts* (in contrast to *Benders' optimality cuts* of the previous slide).

An LP Example

$$\begin{aligned} \max \quad & x - y \\ \text{s.t.} \quad & -25x + 20y \leq 30 \\ & x + 2y \leq 10 \\ & 2x - y \leq 15 \\ & -2x - 10y \leq -15 \\ & x \in [0, 10] \\ & y \in [0, 5] \end{aligned}$$

Master problem:

$$\begin{aligned} \max \quad & x + z \\ \text{s.t.} \quad & z \leq \overline{\phi}_x^k(x) \\ & x \in [0, 10] \\ & z \text{ free} \end{aligned}$$

Subproblem:

$$\begin{aligned} \phi_x(x^k) = \max \quad & -y \\ \text{s.t.} \quad & 20y \leq 30 + 25x^k & (1) \\ & 2y \leq 10 - x^k & (2) \\ & -y \leq 15 - 2x^k & (3) \\ & -10y \leq -15 + 2x^k & (4) \\ & y \in [0, 5] \end{aligned}$$

An LP Example (cont'd)

- In the first iteration, we have no Benders cuts and hence, we get the solution $x^1 = 10$.
- The subproblem is infeasible because (2) becomes $y \leq 0$ and (3) becomes $y \geq 5$, which are in conflict.
- The vector $r = [0, 1, 2, 0]$ is a ray ($20r_1 + 2r_2 - r_3 - 10r_4 = 0$) and has dual objective value $280r_1 - 5r_3 + 5r_4 = -10$.
- This translates to a feasibility cut $1(10 - x) + 2(15 - 2x) = 8 - x \geq 0$.
- Thus, in the second iteration, we have $x^2 = 8$ and $u^k = [0, 0, 1, 0]$.
- As such, the feasibility cut is $z \leq 15 - 2x$.
- This is equivalent to adding constraint (3) and so in the next iteration, we have $x^3 = 7.5$.
- Solving the subproblem, we determine that the lower bound and upper bound are equal, so we are finished and the optimal solution is $(7.5, 0)$.

Benders' Principle (Integer Programming)

$$
\begin{aligned}
z_{\mathrm{LP}} &= \max_{(x,y)\in\mathbb{Z}^n} \{cx + dy \mid Ax \le b, Dx + Gy \le d\} \\
&= \max_{x\in\mathbb{Z}^{n'}} \{cx + \phi(d - Dx) | Ax \le b\},
\end{aligned}
$$

where

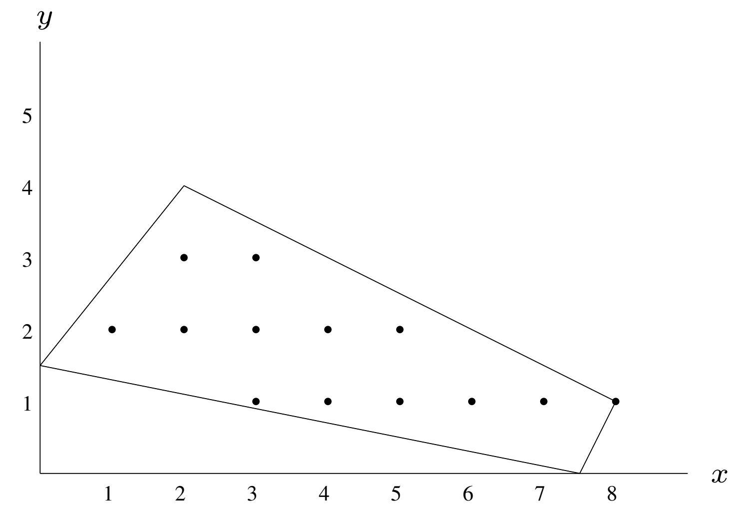
$$
\begin{aligned}
\phi(\beta) = \ & \max \ dy \\
& \text{s.t. } Gy \le \beta \\
& y \in \mathbb{Z}^{n''}
\end{aligned}
$$

Basic Strategy:

- Here, ϕ is the value function of an integer program.
- Here, we also iteratively generate an approximation by constructing a dual functions.

Example

$$
\begin{aligned}
z_{IP} \quad = \quad \max \quad & -x - y \\
\text{s.t.} \quad & -25x + 20y \leq 30 \\
& x + 2y \leq 10 \\
& 2x - y \leq 15 \\
& -2x - 10y \leq -15 \\
& x, y \in \mathbb{Z}
\end{aligned}
$$

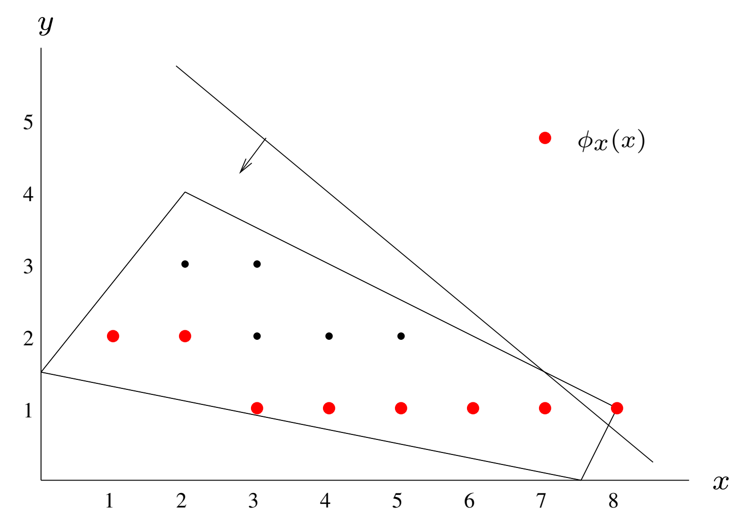


Value Function Reformulation

$$z_{IP} \quad = \quad \max_{x \in \mathbb{Z}} -x + \phi(x),$$

where

$$\begin{aligned} \phi_x(x) \quad = \quad \max \quad & -y \\ \text{s.t.} \quad & 20y \leq 30 + 25x \\ & 2y \leq 10 - x \\ & -y \leq 15 - 2x \\ & -10y \leq -15 + 2x \\ & y \in \mathbb{Z} \end{aligned}$$



- Note again that $\phi_x(x) = \phi(d - Dx)$ and so is not the value function itself.

An MILP Example

$$
\begin{aligned}
\min\ & -x_1 + y_1 + y_2 + y_3 \\
\text{s.t. } & -x_1 + 2y_1 - y_2 + y_3 = 0 \\
& x_1 \in [0, 3] \\
& x_1, y_1 \in \mathbb{Z}_+ \\
& y_2, y_3 \in \mathbb{R}_+
\end{aligned}
$$

Master problem:

$$
\begin{aligned}
\min & -x_1 + \theta \\
\text{s.t.} \quad & \theta \geq \underline{\phi}(x_1) \\
& x_1 \in [0, 3] \\
& x_1 \in \mathbb{Z}_+ \\
& \theta \text{ free}
\end{aligned}
$$

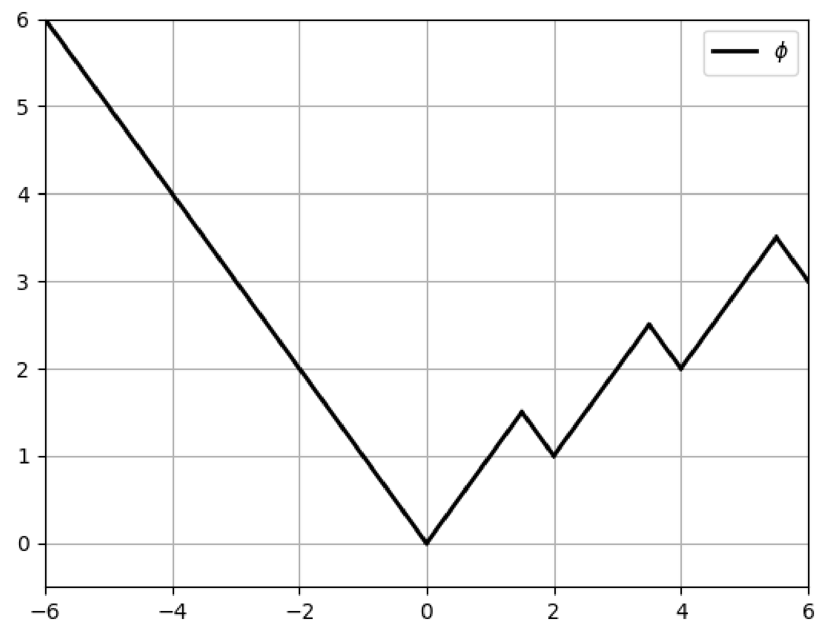
Subproblem ($\beta = x_1$):

$$
\begin{aligned}
\phi(\beta) = \min\ & y_1 + y_2 + y_3 \\
\text{s.t. } & 2y_1 - y_2 + y_3 = \beta \\
& y_1 \in \mathbb{Z}_+ \\
& y_2, y_3 \in \mathbb{R}_+
\end{aligned}
$$

An MILP Example

Subprobem:

$$
\begin{aligned}
\phi(\beta) = \min \ & y_1 + y_2 + y_3 \\
\text{s.t.} \ & 2y_1 - y_2 + y_3 = \beta \\
& y_1 \in \mathbb{Z}_+ \\
& y_2, y_3 \in \mathbb{R}_+
\end{aligned}
$$



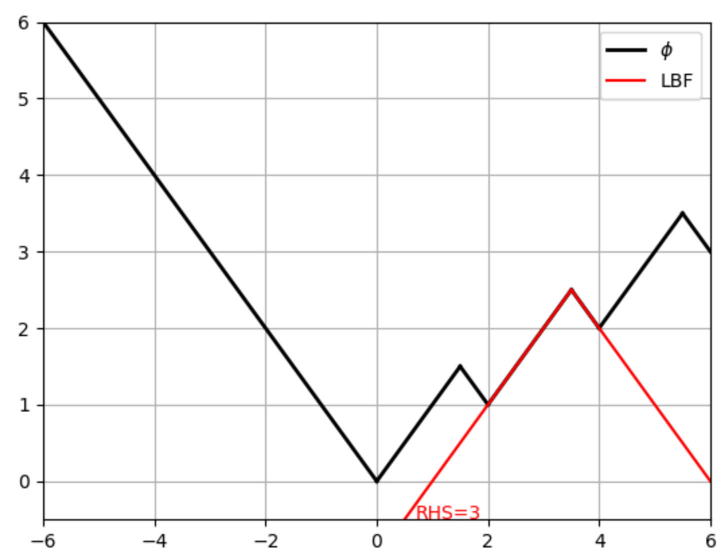
Example

Iteration 1:

$$\begin{aligned} \min\ & -x_1 \\ \text{s.t.}\quad & x_1 \in [0,3] \\ & x_1 \in \mathbb{Z}_+ \end{aligned}$$

$$\begin{aligned} \phi(\beta = x_1^1) = \min\ & y_1 + y_2 + y_3 \\ \text{s.t.}\ & 2y_1 - y_2 + y_3 = 3 \\ & y_1 \in \mathbb{Z}_+ \\ & y_2, y_3 \in \mathbb{R}_+ \end{aligned}$$

$x_1^1 = 3, \theta^1 = -\infty$



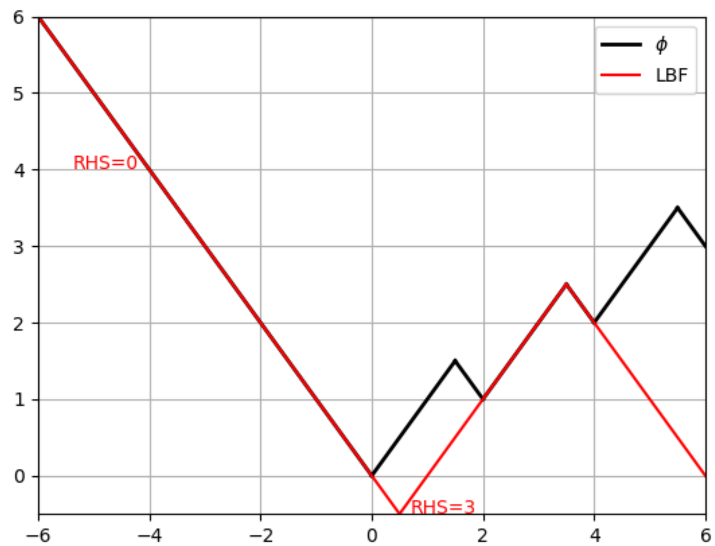
Example

Iteration 2:

$$\begin{aligned} \min\ & -x_1 + \theta \\ \text{s.t.}\quad & \theta \geq \min\{x_1 - 1, -x_1 + 6\} \\ & x_1 \in [0, 3] \\ & x_1 \in \mathbb{Z}_+ \end{aligned}$$

$$\begin{aligned} \phi(\beta = x_1^2) = \min\ & y_1 + y_2 + y_3 \\ \text{s.t.}\ & 2y_1 - y_2 + y_3 = 0 \\ & y_1 \in \mathbb{Z}_+ \\ & y_2, y_3 \in \mathbb{R}_+ \end{aligned}$$

$x_1^2 = 0, \theta^2 = -1$



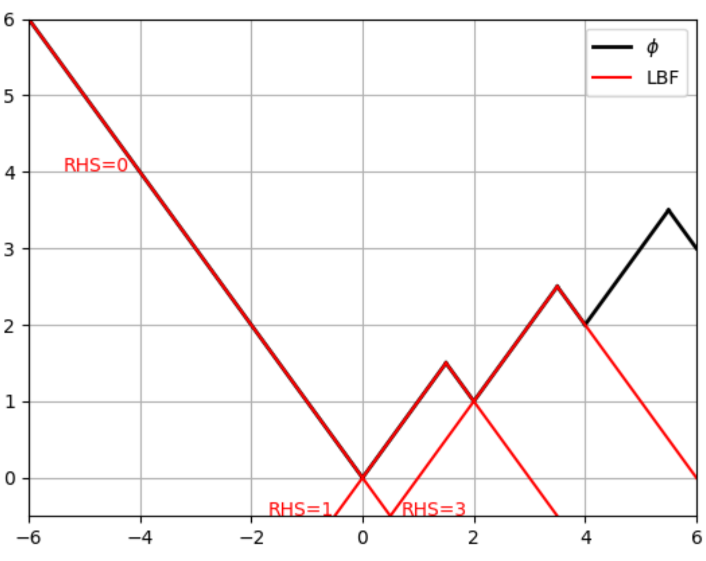
Example

Iteration 3:

$$\min \; -x_1 + \theta$$
$$\text{s.t.} \quad \theta \geq \min\{x_1 - 1, -x_1 + 6\}$$
$$\theta \geq -x_1$$
$$x_1 \in [0, 3]$$
$$x_1 \in \mathbb{Z}_+$$

$$\phi(\beta = x_1^3) = \min \; y_1 + y_2 + y_3$$
$$\text{s.t.} \quad 2y_1 - y_2 + y_3 = 1$$
$$y_1 \in \mathbb{Z}_+$$
$$y_2, y_3 \in \mathbb{R}_+$$

$x_1^3 = 1, \theta^3 = 0$



Example

Iteration 4:

$$\min\ -x_1 + \theta$$
$$\text{s.t.}\quad \theta \geq \min\{x_1 - 1, -x_1 + 6\}$$
$$\theta \geq -x_1$$
$$\theta \geq \min\{x_1, -x_1 + 3\}$$
$$x_1 \in [0, 3]$$
$$x_1 \in \mathbb{Z}_+$$

$$\phi(\beta = x_1^4) = \min\ y_1 + y_2 + y_3$$
$$\text{s.t.}\ \ 2y_1 - y_2 + y_3 = 3$$
$$y_1 \in \mathbb{Z}_+$$
$$y_2, y_3 \in \mathbb{R}_+$$

$x_1^4 = 3, \theta^4 = 2$