

# Lecture 7: Lagrangian Relaxation and Duality Theory

(3 units)

## Outline

- ▶ Lagrangian dual for linear IP
- ▶ Lagrangian dual for general IP
- ▶ Dual Search
- ▶ Lagrangian decomposition

# Joseph Louis Lagrange



Joseph Louis Lagrange (1736-1813)

Lagrange was one of the creators of the calculus of variations, deriving the Euler-Lagrange equations for extrema of functionals. He also extended the method to take into account possible constraints, arriving at the method of Lagrange multipliers.

# Lagrangian Dual for Linear Integer Programming

- ▶ IP with “nice” and complicated constraints:

$$\begin{aligned} \max & c^T x \\ \text{s.t. } & Ax \leq b \text{ (nice constraints)} \\ & Dx \leq d \text{ (complicated constraints)} \\ & x \in \mathbb{Z}_+^n. \end{aligned}$$

Observation: Dropping  $Dx \leq d$  results in an easy problem.

- More general form:

$$\begin{aligned} (IP) \quad & \max c^T x \\ & \text{s.t. } Dx \leq d \\ & x \in X, \end{aligned}$$

where  $Dx \leq d$  is  $m$  hard constraints.

- Lagrangian relaxation:

$$\begin{aligned} (LP(u)) \quad & z(u) = \max c^T x + u^T (d - Dx) \\ & \text{s.t. } x \in X. \end{aligned}$$

**Property:** Problem  $(LP(u))$  is a relaxation of  $(IP)$  for all  $u \geq 0$ .

- Lagrangian dual of  $(IP)$ :

$$(LD) \quad \min_{u \geq 0} z(u).$$

For equality constraints:  $Dx = d$ . The Lagrangian dual is

$$(LD) \quad \min_{u \in \mathbb{R}^m} z(u).$$

## Example 1

- **Uncapacitated facility location problem:** The uncapacitated facility location problem (UFLP) involves locating an undetermined number of facilities to minimize the sum of the (annualized) fixed setup costs and the variable costs of serving the market demand from these facilities.
- **Formulation:**

$$\begin{aligned} (IP) \quad & \max \sum_{i \in M} \sum_{j \in N} c_{ij} x_{ij} - \sum_{j \in N} f_j y_j \\ & s.t. \sum_{j \in N} x_{ij} = 1, \quad i \in M \\ & \quad x_{ij} - y_j \leq 0, \quad i \in M, j \in N \\ & \quad x \in \mathbb{R}^{|M| \times |N|}, y \in \{0, 1\}^{|N|}. \end{aligned}$$

**Locations:**  $N = \{1, \dots, n\}$ ; **Clients:**  $M = \{1, \dots, m\}$ ,  $c_{ij}$ : profit

- Lagrangian relaxation ( $IP(u)$ ):

$$\begin{aligned}
 z(u) = & \max \sum_{i \in M} \sum_{j \in N} (c_{ij} - u_i) x_{ij} - \sum_{j \in N} f_j y_j + \sum_{i \in M} u_i \\
 \text{s.t. } & x_{ij} - y_j \leq 0, \quad i \in M, j \in N \\
 & x \in \mathbb{R}^{|M| \times |N|}, y \in \{0, 1\}^{|N|}.
 \end{aligned}$$

- **Decomposition:**  $z(u) = z_j(u) + \sum_{i \in M} u_i$ , where

$$\begin{aligned}
 z_j(u) = & \max \sum_{i \in M} (c_{ij} - u_i) x_{ij} - f_j y_j \\
 \text{s.t. } & x_{ij} - y_j \leq 0, \quad i \in M \\
 & x_{ij} \geq 0, \quad i \in M, y_j \in \{0, 1\}.
 \end{aligned}$$

It is a subproblem for location  $j$ . We have

$$z_j(u) = \max\{0, \sum_{i \in M} \max(c_{ij} - u_i, 0) - f_j\}.$$

## Example 2

- **Generalized assignment problem (GAP).** The generalized assignment problem is an excellent example for illustrating Lagrangian relaxation because it is rich with readily apparent structure.
- Formulation:

$$Z = \min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (1)$$

$$\text{s.t.} \quad \sum_{i=1}^m x_{ij} = 1, \quad j = 1, \dots, n \quad (2)$$

$$\sum_{j=1}^n a_{ij} x_{ij} \leq b_i, \quad i = 1, \dots, m \quad (3)$$

$$x_{ij} \in \{0, 1\}, \text{ for all } i, j. \quad (4)$$

► Relaxations 1:

$$\begin{aligned} Z_{D1}(u) &= \min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} + \sum_{j=1}^n u_j \left( \sum_{i=1}^m x_{ij} - 1 \right) \\ &\quad \text{s.t. (3), (4)} \\ &= \min \sum_{i=1}^m \sum_{j=1}^n (c_{ij} + u_j) x_{ij} - \sum_{j=1}^n u_j \\ &\quad \text{s.t. (3), (4)}. \end{aligned}$$

This problem reduces to  $m$  0-1 knapsack problems and can thus be solved in time proportional to  $n \sum_{i=1}^m b_i$ .



► Relaxation 2

$$\begin{aligned} Z_{D2}(v) &= \min \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} + \sum_{i=1}^m v_i \left( \sum_{j=1}^n a_{ij} x_{ij} - b_i \right) \\ &\quad \text{s.t. (2), (4)} \\ &= \min \sum_{j=1}^n \sum_{i=1}^m (c_{ij} + v_i a_{ij}) x_{ij} - \sum_{i=1}^m v_i b_i \\ &\quad \text{s.t. (2), (4).} \end{aligned}$$

Since constraints (2) are **generalized upper bound (GUB) constraints**, it is 0-1 GUB problem. Such a problem is easily solved in time proportional to  $nm$  by determining  $\min_i (c_{ij} + v_i a_{ij})$  for each  $j$  and setting the associated  $x_{ij} = 1$ . Remaining  $x_{ij}$  are set to zero.

- **Classical paper:** Marshall L. Fisher, The Lagrangian Relaxation Method for Solving Integer Programming Problems, Manage. Sci., Vol. 50, No. 12 Supplement, December 2004, pp. 1861 – 1871. (Original, MS, Vol. 27(1), 1-18, 1981).

# How Good is Lagrangian Dual?

- Consider linear program:

$$\min\{c^T x \mid Ax \leq b, \quad Cx \leq d, \quad x \in X\},$$

where  $X$  is a finite integer set, e.g.,  $X = \{0, 1\}^n$ .

- Let  $\{x^1, \dots, x^N\}$  be the finite integer points in  $\{x \in X \mid Cx \leq d\}$ .
- Dual problem:

$$\begin{aligned}v(D) &= \max_{\lambda \geq 0} d(\lambda) \\&= \max_{\lambda \geq 0} \min_{x \in X} [c^T x + \lambda^T (Ax - d)] \\&= \max_{\lambda \geq 0} \min_{i=1, \dots, N} [c^T x^i + \lambda^T (Ax^i - b)] \\&= \max \eta \\&\quad \text{s.t. } c^T x^i + \lambda^T (Ax^i - b) \leq \eta, \quad i = 1, \dots, N, \\&\quad \lambda \geq 0, \quad \eta \in \mathbb{R}.\end{aligned}$$

- The latter problem is a linear problem. Taking its dual gives

$$\begin{aligned} v(D) &= \min \sum_{i=1}^N \mu_i (c^T x^i) \\ \text{s.t. } &\sum_{i=1}^N \mu_i (Ax^i - d) \leq 0, \\ &\sum_{i=1}^N \mu_i = 1, \mu_i \geq 0. \end{aligned}$$

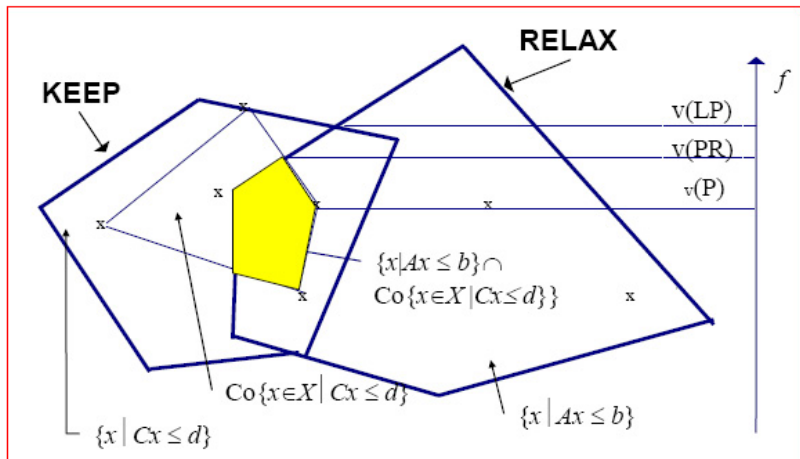
Setting  $x = \sum_{i=1}^N \mu_i x^i$ , we imply that  $(D)$  is equivalent to the following LP:

$$\begin{aligned} (PR) \quad &\min c^T x \\ \text{s.t. } &Ax \leq b, \\ &x \in \text{Co}(\{x \in X \mid Cx \leq d\}). \end{aligned}$$

► Theorem:

$$v(D) = v(PR) = \min\{c^T x \mid Ax \leq b, x \in \text{Co}(\{x \in X \mid Cx \leq d\})\}.$$

- $v(D) \geq v(LP) = \min\{c^T x \mid Ax \leq b, Cx \leq d\}.$
- If  $\text{Co}(\{x \in X \mid Cx \leq b\}) = \{x \in X \mid Cx \leq b\}$ , then  $v(D) = v(LP)$ . Lagrangian bound is equal to the LP relaxation. Example:  $\{x \mid \mathbb{Z}_+^n \mid l \leq x \leq u\}.$



**Figure:** Lagrangian bound for linear integer programming.

# Key issues in Lagrangian dual

- ▶ How to solve the Lagrangian relaxation?
- ▶ How to solve the dual problem?
- ▶ When Lagrangian relaxation useful?

# Solving Relaxation Problem

- ▶ Linear integer program;
- ▶ Many combinatorial optimization problems
- ▶ separable integer programming (knapsack problem)
- ▶ 0-1 unconstrained quadratic problem
- ▶ 0-1 quadratic knapsack program

# Dual Search Methods

- ▶ Problem: How to solve the dual problem (D)?

$$(D) \quad \max_{\lambda \geq 0} d(\lambda).$$

- ▶ Three Basic Methods:
  - ▶ Subgradient method
  - ▶ Outer approximation method
  - ▶ Bundle method



# Subgradient method

- ▶ Subgradient method

$$\lambda^{k+1} = P^+(\lambda^k + s_k \xi^k / \|\xi^k\|),$$

where  $s_k > 0$  is the stepsize.

- ▶ Stepsize rules for  $s_k$ :
  - ▶ Rule 1 for stepsize (constant):

$$s_k = \epsilon,$$

where  $\epsilon > 0$  is a constant.

- ▶ Rule 2 for stepsize:

$$\sum_{k=1}^{+\infty} s_k^2 < +\infty \text{ and } \sum_{k=1}^{+\infty} s_k = +\infty.$$

- ▶ Rule 3 for stepsize:

$$s_k \rightarrow 0, \quad k \rightarrow +\infty, \quad \text{and} \quad \sum_{k=1}^{+\infty} s_k = +\infty.$$

- A sophisticated stepsize rule:

$$s_k = \rho \frac{w_k - d(\lambda^k)}{\|\xi^k\|}, \quad 0 < \rho < 2,$$

where  $w_k$  is an approximation of the optimal value  $v(D)$ ,  $w_k \geq d(\lambda^k)$  and  $\xi^k \neq 0$ .

# Outer approximation method

- ▶ Rewrite dual problem ( $D$ ) as

$$\begin{aligned} \max \quad & \mu \\ \text{s.t.} \quad & \mu \leq f(x) + \lambda^T(g(x) - b), \forall x \in X, \\ & \lambda \geq 0. \end{aligned}$$

- ▶ Approximate the LP by:

$$\begin{aligned} \max \quad & \mu \\ \text{s.t.} \quad & \mu \leq f(x^j) + \lambda^T(g(x^j) - b), \forall x^j \in T^k, \\ & \lambda \geq 0. \end{aligned}$$

- ▶ Finite convergence
- ▶ Singly constrained case of ( $P$ ):

$$\begin{aligned} (P_s) \quad & \min f(x) \\ \text{s.t.} \quad & g(x) \leq b \\ & x \in X. \end{aligned}$$

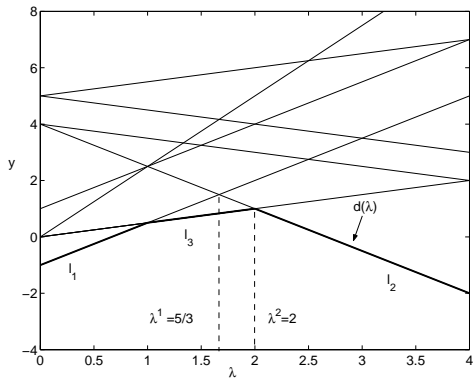


Figure: Dual function.

# Lagrangian decomposition

## ► Purpose:

- Induce decomposition of the problem into independent subproblems;
- Capture different structural characteristics of the problem;
- Obtain stronger bounds than by standard Lagrangean relaxation schemes

## ► How?

- Identify the parts of the problem that should be split;
- Replace variables in each part by copies or substitute new expressions;
- Dualize the copy or substitution expression.

- Convex program with nice and hard constraints:

$$\begin{array}{ll} (CP) & \min f(x) + g(x) \\ & \text{s.t. } x \in X \text{ (nice constraints)} \\ & \quad x \in Y \text{ (hard constraints)} \end{array}$$

- Lagrangian decomposition via copying constraints

$$\begin{array}{ll} (CP) & \min f(x) + g(z) \\ & \text{s.t. } x \in X \text{ (nice constraints)} \\ & \quad x = z, \text{ (link constraints)} \\ & \quad z \in Y \text{ (hard constraints)} \end{array}$$

- Associating a multiplier to each link constraint:

$$\begin{aligned}
 (CP) \quad & \min f(x) + g(z) \\
 & \text{s.t. } x \in X \\
 & \lambda \Rightarrow \quad x = z, \\
 & \quad \quad z \in Y
 \end{aligned}$$

- Decomposition:

$$\begin{aligned}
 d(\lambda) &= \min_{\substack{x \in X \\ z \in Y}} f(x) + g(z) + \lambda^T(x - z) \\
 &= \min_{x \in X} f(x) + \lambda^T x + \min_{z \in Y} g(z) - \lambda^T z \\
 &= d_1(\lambda) + d_2(\lambda) \Rightarrow \text{two easier subproblems!}
 \end{aligned}$$