CS711008Z Algorithm Design and Analysis

Lecture 9. Algorithm design technique: Linear programming and duality

Dongbo Bu

Institute of Computing Technology
Chinese Academy of Sciences, Beijing, China

Outline

- The first example: the dual of DIET problem;
- Understanding duality: Lagrangian function, Lagrangian dual function, and Lagrangian dual problem;
- Conditions of optimal solution;
- Four properties of duality for linear program;
- Solving LP using duality: Dual simplex algorithm, PRIMAL AND DUAL algorithm, and interior point method;
- Applications of duality: Farkas lemma, von Neumann' s MINIMAX theorem, Yao's MINIMAX theorem, Dual problem in SVM, and SHORTESTPATH problem;
- Appendix: Proof of Slater theorem, and techniques to finding initial solution to dual problem.

Importance of duality

- **When minimizing a function $f(x)$, it is invaluable to know a lower bound of $f(x)$ in advance.** Calculation of lower bound is extremely important to the design of approximation algorithm and branch-and-bound method.
- **Duality** and **relaxation** (say Lagrangian relaxation, integer relaxation, convex relaxation) are powerful techniques to obtain a reasonable lower bound.
- Linear programs come in primal/dual pairs. It turns out that every feasible solution for one of these two problems provides a bound for the objective value of the other problem.
- The **dual problems are always convex** even if the primal problems are not convex.

The first example: the dual of DIET problem.

Revisiting DIET problem

- A housewife wonders how much money she must spend on foods in order to get all the energy (2000 kcal), protein (55 g), and calcium (800 mg) that she needs every day.

Food	Energy	Protein	Calcium	Price	Quantity
Oatmeal	110	4	2	3	x_1
Whole milk	160	8	285	9	x_2
Cherry pie	420	4	22	20	x_3
Pork beans	260	14	80	19	x_4

- Linear program:

$$\begin{array}{lrcrcrcrll}
\min & 3x_1 & + & 9x_2 & + & 20x_3 & + & 19x_4 & & \text{money} \\
s.t. & 110x_1 & + & 160x_2 & + & 420x_3 & + & 260x_4 & \geq 2000 & \text{energy} \\
 & 4x_1 & + & 8x_2 & + & 4x_3 & + & 14x_4 & \geq 55 & \text{protein} \\
 & 2x_1 & + & 285x_2 & + & 22x_3 & + & 80x_4 & \geq 800 & \text{calcium} \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq 0 &
\end{array}$$

Dual of DIET problem: PRICING problem

- Consider a company producing protein powder, energy bar, and calcium tablet as substitution to foods.
- The company wants to design a reasonable pricing strategy to earn money as much as possible.
- However, the price cannot be arbitrarily high due to the following considerations:
  1. If the prices are competitive with foods, one might consider choosing a combination of the ingredients rather than foods;
  2. Otherwise, one will choose to buy foods directly.

LP model of PRICING problem

Food	Energy	Protein	Calcium	Price (cents)
Oatmeal	110	4	2	3
Whole milk	160	8	285	9
Cherry pie	420	4	22	20
Pork with beans	260	14	80	19
Price	y_1	y_2	y_3	

- Linear program:

$$
\begin{array}{rrcrcrcrl}
\max & 2000y_1 & + & 55y_2 & + & 800y_3 & & & \text{money} \\
s.t. & 110y_1 & + & 4y_2 & + & 2y_3 & \leq & 3 & \text{oatmeal} \\
 & 160y_1 & + & 8y_2 & + & 285y_3 & \leq & 9 & \text{milk} \\
 & 420y_1 & + & 4y_2 & + & 22y_3 & \leq & 20 & \text{pie} \\
 & 260y_1 & + & 14y_2 & + & 80y_3 & \leq & 19 & \text{pork\&beans} \\
 & y_1 & , & y_2 & , & y_3 & \geq & 0 &
\end{array}
$$

Primal problem and Dual problem

$$
\begin{array}{cccccc}
c_1 & c_2 & \cdots & c_n & \\
a_{11} & a_{12} & \cdots & a_{1n} & b_1 \\
a_{21} & a_{22} & \cdots & a_{2n} & b_2 \\
 & & \cdots & & \\
a_{m1} & a_{m2} & \cdots & a_{mn} & b_m
\end{array}
$$

- Primal problem and Dual problem are two points of view of the coefficient matrix A:
  - Primal problem: row point of view
  - Dual problem: column point of view

PRIMAL problem

$$
\begin{array}{lcccccccccc}
\min & c_1 x_1 & + & c_2 x_2 & + & \ldots & + & c_n x_n & & & \\
 & a_{11} x_1 & + & a_{12} x_2 & + & \ldots & + & a_{1n} x_n & \geq & b_1 & \\
 & a_{21} x_1 & + & a_{22} x_2 & + & \ldots & + & a_{2n} x_n & \geq & b_2 & \\
 & & & & & \ldots & & & & & \\
 & a_{m1} x_1 & + & a_{m2} x_2 & + & \ldots & + & a_{mn} x_n & \geq & b_m & \\
 & & & & & & & x_i & \geq & 0 & \text{for each } i
\end{array}
$$

- Primal problem: row point of view (in red);

$$
\begin{array}{rrcl}
\min & c^T x & & \\
s.t. & Ax & \geq & b \\
 & x & \geq & 0
\end{array}
$$

Dual problem

$$
\begin{array}{ccccc}
c_1 & c_2 & \cdots & c_n & \\
\geq & \geq & & \geq & \max \\
y_1 a_{11} & y_1 a_{12} & \cdots & y_1 a_{1n} & y_1 b_1 \\
+ & + & & + & + \\
y_2 a_{21} & y_2 a_{22} & \cdots & y_2 a_{2n} & y_2 b_2 \\
+ & + & & + & + \\
\vdots & \vdots & \cdots & \vdots & \vdots \\
+ & + & & + & + \\
y_m a_{m1} & y_m a_{m2} & \cdots & y_m a_{mn} & y_m b_m \\
 & & & y_j \quad \geq & 0 \quad \text{for each } j
\end{array}
$$

- Dual problem: column point of view (in blue).

$$
\begin{array}{rrcl}
\max & b^T y & & \\
s.t. & y & \geq & 0 \\
 & A^T y & \leq & c
\end{array}
$$

How to write DUAL problem? Case 1

- For each **constraint** in the PRIMAL problem, a **variable** is set in the DUAL problem.
- If the PRIMAL problem has **inequality constraints**, the DUAL problem is written as follows.
  - Primal problem:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & \geq & b \\ & x & \geq & 0 \end{array}$$

  - Dual problem:

$$\begin{array}{rrcl} \max & b^T y & & \\ s.t. & y & \geq & 0 \\ & A^T y & \leq & c \end{array}$$

How to write DUAL problem? Case 2

- For each **constraint** in the PRIMAL problem, a **variable** is set in the DUAL problem.
- If the PRIMAL problem has **inequality constraints**, the DUAL problem is written as follows.
  - Primal problem:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & \leq & b \\ & x & \geq & 0 \end{array}$$

  - Dual problem:

$$\begin{array}{rrcl} \max & b^T y & & \\ s.t. & y & \leq & 0 \\ & A^T y & \leq & c \end{array}$$

How to write DUAL problem? Case 3

- For each **constraint** in the PRIMAL problem, a **variable** is set in the DUAL problem.
- If the PRIMAL problem has **equality constraints**, the DUAL problem is as follows.
  - Primal problem:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & = & b \\ & x & \geq & 0 \end{array}$$

  - Dual problem:

$$\begin{array}{rrcl} \max & b^T y & & \\ s.t. & & & \\ & A^T y & \leq & c \end{array}$$

  - Note: there is neither $y \geq 0$ nor $y \leq 0$ constraint in the dual problem.

Why can the DUAL problem be written as above?
— Understanding duality from the Lagrangian dual point of view

Standard form of constrained optimization problems

- Consider the following constrained optimization problem (might be non-convex).

$$\begin{array}{rlcll} \min & f_0(x) & & & \\ s.t. & f_i(x) & \leq & 0 & i = 1, ..., m \\ & h_i(x) & = & 0 & i = 1, ..., p \end{array}$$

- Here the variables $x \in \mathbb{R}^n$ and we use $\mathcal{D} = \bigcap_{i=0}^{m} \textbf{dom } f_i \cap \bigcap_{i=1}^{p} \textbf{dom } h_i$ to represent the domain of definition. We use p^* to represent the optimal value of the problem.

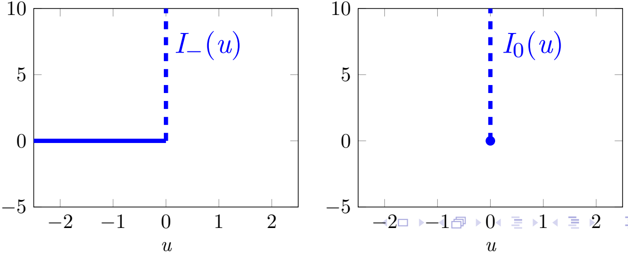
An equivalent unconstrained optimization problem

- We can transform this **constrained optimization** problem into an equivalent **unconstrained optimization** problem:

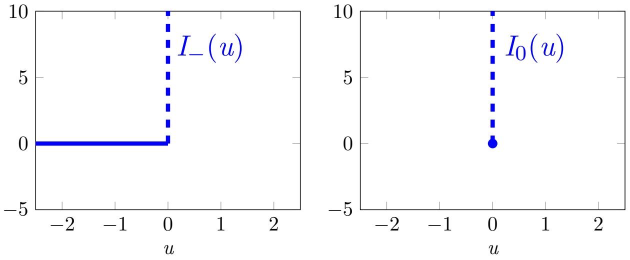
$$\min f_0(x) + \sum_{i=1}^{m} I_-(f_i(x)) + \sum_{i=1}^{p} I_0(h_i(x))$$

where $x \in \mathcal{D}$, $I_-(u)$ and $I_0(u)$ are indicator functions for non-positive reals and the set $\{0\}$, respectively:

$$I_-(u) = \begin{cases} 0 & u \leq 0 \\ \infty & u > 0 \end{cases} \qquad I_0(u) = \begin{cases} 0 & u = 0 \\ \infty & u \neq 0 \end{cases}$$



Difficulty in solving the optimization problem

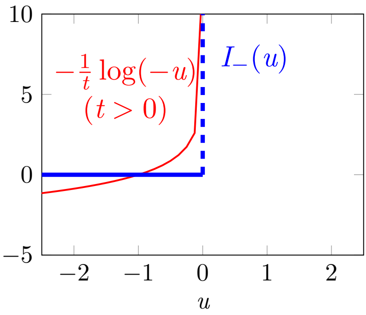


- Intuitively, $I_-(u)$ and $I_0(u)$ represent our "infinite dissatisfaction" with the violence of constraints.
- However both $I_0(u)$ and $I_-(u)$ are non-differentiable, making the optimization problem

$$\min f_0(x) + \sum_{i=1}^{m} I_-(f_i(x)) + \sum_{i=1}^{p} I_0(h_i(x))$$

, although unconstrained, not easy to solve.
- Question: How to efficiently solve this optimization problem?

Approximating $I_-(u)$ using a differentiable function (1)

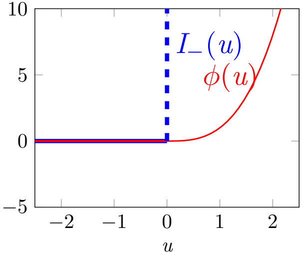


- An approximation to $I_-(u)$ is **logarithm barrier function**:

$$\hat{I}_-(u) = -\frac{1}{t}\log(-u) \qquad (t > 0)$$

- The difference between $\hat{I}_-(u)$ and $I_-(u)$ decreases as t increases. This approximation was used in the interior point method.

Approximating $I_{-}(u)$ using a differentiable function (2)

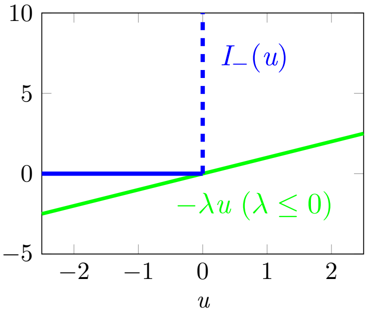


- Another approximation to $I_{-}(u)$ is a **penalty function**:

$$\hat{I}_{-}(u) = \phi(u) = \begin{cases} u^t \quad (t > 1) & u \geq 0 \\ 0 & otherwise \end{cases}$$

The penalty function "penalizes" any u which is greater than zero. It is a "hands-off" method for converting contracted problems into unconstrained problems, to which an initial feasible solution is easy to obtained. However, in some cases it cannot be applied because the objective function is undefined out of the feasible set or the unconstrained problem becomes ill-conditioned as t increases.

Approximating $I_{-}(u)$ using a differentiable function (3)

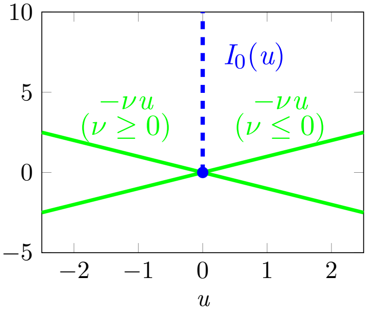


- Another approximation to $I_{-}(u)$ is a simple **linear function**:

$$\hat{I}_{-}(u) = -\lambda u \qquad (\lambda \leq 0)$$

- Despite the considerable difference between $\hat{I}_{-}(u)$ and $I_{-}(u)$, $\hat{I}_{-}(u)$ still provides lower bound information of $I_{-}(u)$.

Approximating $I_0(u)$ using a differentiable function



- $I_0(u)$ can also be approximated using **linear function**:

$$\hat{I}_0(u) = -\nu u$$

- Although $\hat{I}_0(u)$ deviates considerably from $I_0(u)$, $\hat{I}_0(u)$ still provides lower bound information of $I_0(u)$.
- It is worthy pointed out that unlike $\hat{I}_-(u)$, $\hat{I}_0(u)$ has no restriction on ν.

Lagrangian function

- Consider the unconstrained optimization problem

$$\min f_0(x) + \sum_{i=1}^{m} I_-(f_i(x)) + \sum_{i=1}^{p} I_0(h_i(x)).$$

- Now let's replace $I_-(u)$ with $-\lambda u$ ($\lambda \leq 0$) and replace $I_0(u)$ with $-\nu u$. Then the objective function becomes:

$$L(x, \lambda, \nu) = f_0(x) - \sum_{i=1}^{m} \lambda_i f_i(x) - \sum_{i=1}^{p} \nu_i h_i(x)$$

  This function is called **Lagrangian function**, which is **a lower bound of $f_0(x)$** for any feasible solution x when $\lambda \leq 0$.
- Here we call λ_i **Lagrangian multiplier** for the i-th inequality constraint $f_i(x) \leq 0$ and ν_i **Lagrangian multiplier** for the i-th equality constraint $h_i(x) = 0$.

Lagrangian connecting primal and dual: An example

- Primal problem:

$$\begin{array}{lrcl} \min & x^2 - 2x & & \\ s.t. & -x & \leq & 0 \end{array}$$

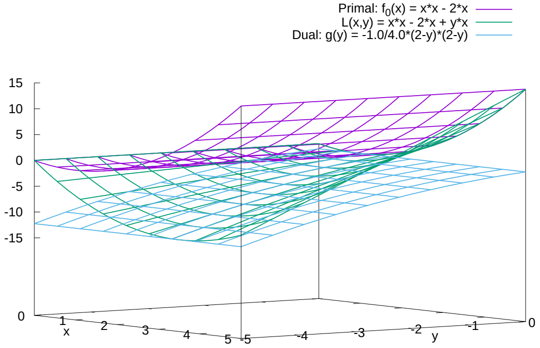
- Lagrangian:

$$L(x, \lambda) = x^2 - 2x + \lambda x$$

- Note that $L(x, \lambda)$ is a lower bound of the primal objective function $x^2 - 2x$ when $\lambda \leq 0$ and $-x \leq 0$.
- Dual problem:

$$\begin{array}{lr} \max & -\frac{1}{4}(2 - \lambda)^2 \\ s.t. & \lambda \leq 0 \end{array}$$

Lagrangian connecting primal and dual



- Observation: PRIMAL objective function $\geq$ Lagrangian $\geq$ DUAL objective function in the feasible region.

Lagrangian dual function and Lagrangian dual problem

Lagrangian dual function

- Consider the following constrained optimization problem.

$$\begin{array}{rrcll} \min & f_0(x) & & & \\ s.t. & f_i(x) & \leq & 0 & i = 1, ..., m \\ & h_i(x) & = & 0 & i = 1, ..., p \end{array}$$

- **Lagrangian function:**

$$L(x, \lambda, \nu) = f_0(x) - \sum_{i=1}^{m} \lambda_i f_i(x) - \sum_{i=1}^{p} \nu_i h_i(x)$$

  which is a lower bound of $f_0(x)$ for any feasible solution x when $\lambda \leq 0$.
- Now let's consider the infimum of Lagrangian function (called **Lagrangian dual function**):

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

  Note: infimum rather than minimum is used here as some sets have no minimum.

Lagrangian dual problem

- Lagrangian dual function provides lower bound of the primal objective function, i.e.

$$f_0(x) \geq L(x, \lambda, \nu) \geq g(\lambda, \nu)$$

  for any feasible solution x when $\lambda \leq 0$.

- Now let's try to find **the tightest lower bound** of the primal objective function, which can be obtained by solving the following **Lagrangian dual problem**:

$$\begin{array}{rrcl} \max & g(\lambda, \nu) & & \\ s.t. & \lambda & \leq & 0 \end{array}$$

Dual problem is always convex

- Note that the Lagrangian dual function $g(\lambda, \nu)$ is a point-wise minimum of affine functions over λ, ν.

$$
\begin{aligned}
g(\lambda, \nu) &= \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \\
&= \inf_{x \in \mathcal{D}} \left(f_0(x) - \sum_{i=1}^{m} \lambda_i f_i(x) - \sum_{i=1}^{m} \nu_i h_i(x) \right)
\end{aligned}
$$

- Thus **Lagrangian dual function $g(\lambda, \nu)$ is always concave** and **the dual problem is always a convex programming problem** even if the primal problem is non-convex.

Note: The dual is not intrinsic

- It is worthy pointed out that the dual problem and its optimal objective value are not properties of the primal feasible set and primal objective function alone. They also depend on the specific constraints in the primal problem.
- Thus we can **construct equivalent primal optimization problem with different duals** through the following ways:
  - Replacing primal objective function $f_0(x)$ with $h(f_0(x))$ where $h(u)$ is monotonically increasing.
  - Introducing new variables.
  - Adding redundant constraints.

Lagrangian function and dual function: An example I

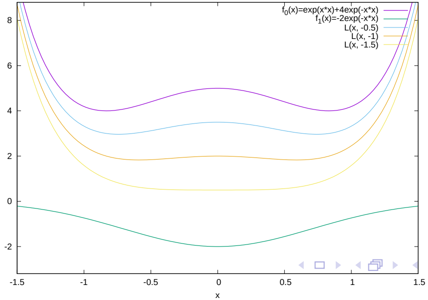
- Consider the following primal problem:

$$\begin{array}{rrcll} \min & e^{x^2} + 4e^{-x^2} & & & \\ s.t. & -2e^{-x^2} + 1 & \leq & 0 & i = 1, ..., m \end{array}$$

- Lagrangian function:

$$L(x, \lambda) = e^{x^2} + 4e^{-x^2} - \lambda(-2e^{-x^2} + 1)$$

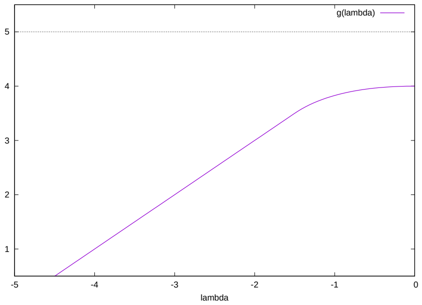
where $\lambda \leq 0$.



Lagrangian function and dual function: An example II

- Lagrangian dual function:

$$\begin{aligned} g(\lambda) &= \inf_{x \in \mathbb{R}} L(x, \lambda) \\ &= \begin{cases} 5 + \lambda & \lambda \leq -1.5 \\ 2\sqrt{4 + 2\lambda} - \lambda & -1.5 \leq \lambda \leq 0 \end{cases} \end{aligned}$$



Deriving dual problem of linear program in standard form

Dual problem of LP problem in standard form

- Consider a LP problem in standard form:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & \leq & b \\ & x & \geq & 0 \end{array}$$

- Lagrangian function:

$$L(x, \lambda, \nu) = c^T x - \sum_{i=1}^{m} \lambda_i (a_{i1} x_1 + \ldots + a_{in} x_n - b_i) - \sum_{i=1}^{n} \nu_i x_i$$

  where $\lambda \leq 0$, $\nu \geq 0$.
- Notice that for any feasible solution x and $\lambda \leq 0$, $\nu \geq 0$, **Lagrangian function is a lower bound of the primal objective function**, i.e. $c^T x \geq L(x, \lambda, \nu)$, and further

$$c^T x \geq L(x, \lambda, \nu) \geq \inf_{x \in \mathcal{D}} L(x, \lambda, \nu)$$

- Let's define **Lagrangian dual function** $g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu)$ and rewrite the above inequality as

$$c^T x \geq L(x, \lambda, \nu) \geq g(\lambda, \nu)$$

Lagrangian dual function

- What is the Lagrangian dual function $g(\lambda, \nu)$?

$$
\begin{aligned}
g(\lambda,\nu) &= \inf_{x\in\mathcal{D}} L(x,\lambda,\nu) \\
&= \inf_{x\in\mathcal{D}} (c^T x - \sum_{i=1}^{m} \lambda_i (a_{i1}x_1 + \ldots + a_{in}x_n - b_i) - \sum_{i=1}^{m} \nu_i x_i) \\
&= \inf_{x\in\mathcal{D}} (\lambda^T b + (c^T - \lambda^T A - \nu^T) x) \\
&= \begin{cases} \lambda^T b & \text{if } c^T = \lambda^T A + \nu^T \\ -\infty & \text{otherwise} \end{cases}
\end{aligned}
$$

- Note that $\mathcal{D} = \mathbb{R}^n$. Thus $g(\lambda,\nu) = \lambda^T b$ if $c^T = \lambda^T A + \nu^T$; otherwise, $g(\lambda,\nu) = -\infty$, which is a trivial lower bound for the primal objective function $c^T x$.
- We usually denote the domain of $g(\lambda,\nu)$ as $\mathbf{dom}\ g = \{(\lambda,\nu) | g(\lambda,\nu) > -\infty\}$.

Lagrangian dual problem

- Now let's try to find **the tightest lower bound** of the primal objective function c^Tx, which can be calculated by solving the following **Lagrangian dual problem**:

$$\begin{array}{rrcl} \max & g(\lambda, \nu) & = & \begin{cases} \lambda^T b & \text{if } c^T = \lambda^T A + \nu^T \\ -\infty & \text{otherwise} \end{cases} \\ s.t. & \lambda & \leq & 0 \\ & \nu & \geq & 0 \end{array}$$

or explicitly representing constraints in $\mathbf{dom}\ g$:

$$\begin{array}{rrcl} \max & \lambda^T b & & \\ s.t. & \lambda^T A & \leq & c^T \\ & \lambda & \leq & 0 \end{array}$$

- Note that this is actually the DUAL form of LP if replacing λ by y; thus, we have another explanation of DUAL variables y — the Lagrangian multiplier.

An example

- Primal problem:

$$\begin{array}{lrcl} \min & x & & \\ s.t. & x & \geq & 2 \\ & x & \geq & 0 \end{array}$$

- Lagrangian function:

$$L(x, \lambda, \nu) = x - \lambda(x-2) - \nu x = 2\lambda + (1 - \lambda - \nu)x$$

- Note that when $\lambda \geq 0$, $\nu \geq 0$ and $x \geq 2$, $L(x, \lambda, \nu)$ is a lower bound of the primal objective function x.
- Lagrangian dual function:

$$g(\lambda, \nu) = \inf_{x \in \mathbb{R}} L(x, \lambda, \nu) = \begin{cases} 2\lambda & \text{if } 1 - \lambda - \nu = 0 \\ -\infty & \text{otherwise} \end{cases}$$

- Dual problem:

$$\begin{array}{lrcl} \max & 2\lambda & & \\ s.t. & \lambda & \leq & 1 \\ & \lambda & \geq & 0 \end{array}$$

Deriving dual problem of linear program in slack form

Dual problem of LP problem in slack form

- Consider a LP problem in slack form:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & = & b \\ & x & \geq & 0 \end{array}$$

- Lagrangian function:

$$L(x, \lambda, s) = c^T x - \sum_{i=1}^{m} \lambda_i (a_{i1} x_1 + \ldots + a_{in} x_n - b_i) - \sum_{i=1}^{n} \nu_i x_i$$

- Notice that for any feasible solution x and $\nu \geq 0$, **Lagrangian function is a lower bound of the primal objective function**, i.e. $c^T x \geq L(x, \lambda, \nu)$, and further

$$c^T x \geq L(x, \lambda, \nu) \geq \inf_{x \in \mathcal{D}} L(x, \lambda, \nu)$$

- Let's define **Lagrangian dual function** $g(\lambda, \nu) = \inf_{x \in \mathcal{D}} L(x, \lambda, \nu)$ and rewrite the above inequality as

$$c^T x \geq L(x, \lambda, \nu) \geq g(\lambda, \nu)$$

Lagrangian dual function

- What is the Lagrangian dual function $g(\lambda, \nu)$?

$$
\begin{aligned}
g(\lambda, \nu) &= \inf_{x \in \mathcal{D}} L(x, \lambda, \nu) \\
&= \inf_{x \in \mathcal{D}} (c^T x - \sum_{i=1}^{m} \lambda_i (a_{i1} x_1 + \ldots + a_{in} x_n - b_i) - \sum_{i=1}^{m} \nu_i x_i) \\
&= \inf_{x \in \mathcal{D}} (\lambda^T b + (c^T - \lambda^T A - \nu^T) x) \\
&= \begin{cases} \lambda^T b & \text{if } c^T = \lambda^T A + \nu^T \\ -\infty & \text{otherwise} \end{cases}
\end{aligned}
$$

- Note that $\mathcal{D} = \mathbb{R}^n$. Thus $g(\lambda, \nu) = \lambda^T b$ if $c^T = \lambda^T A + \nu^T$; otherwise, $g(\lambda, \nu) = -\infty$, which is a trivial lower bound for the primal objective function $c^T x$.

Lagrangian dual problem

- Now let's try to find **the tightest lower bound** of the primal objective function $c^T x$, which can be calculated by solving the following **Lagrangian dual problem**:

$$\begin{aligned} \max \quad g(\lambda, \nu) &= \begin{cases} \lambda^T b & \text{if } c^T = \lambda^T A + \nu^T \\ -\infty & \text{otherwise} \end{cases} \\ s.t. \quad \nu &\geq 0 \end{aligned}$$

, or explicitly representing constraints in $\mathbf{dom}\ g$:

$$\begin{aligned} \max \quad & \lambda^T b \\ s.t. \quad & \lambda^T A \leq c^T \end{aligned}$$

- Note that the dual problem does not have the $\lambda \leq 0$ constraint as for any λ_i, $\lambda_i(a_{i1}x_1 + ... + a_{in}x_n - b_i)$ is a lower bound for $I_0(a_{i1}x_1 + ... + a_{in}x_n - b_i)$.

Explanations of dual variables

Two explanations of dual variables y: L. Kantorovich vs. T. Koopmans

1. **Price interpretation**: Constrained optimization plays an important role in economics. Dual variables are also called as **shadow price** (by T. Koopmans), i.e. the instantaneous change in the optimization objective function when constraints are relaxed, or **marginal cost** when strengthening constraints.
2. **Lagrangian multiplier interpretation**: Dual variables are essentially **Lagrangian multiplier**, which describe the effect of constraints on the objective function (by L. Kantorovich). For example, it can describe how much the optimal objective function will change when b_i increase to $b_i + \Delta b_i$. In fact, we have $\frac{\partial L(x,\lambda)}{\partial b_i} = \lambda_i$.

Explanation of dual variables y: using DIET as an example

- Optimal solution to primal problem with $b_1 = 2000, b_2 = 55, b_3 = 800$:
$$x = (14.24, 2.70, 0, 0), \qquad c^T x = 67.096.$$
- Optimal solution to dual problem:
$$y = (0.0269, 0, 0.0164), \qquad y^T b = 67.096.$$
- Let's make a slight change on b, and examine its effect on $\min c^T x$.

b_1	b_2	b_3	$\min c^T x$
2000	55	800	67.096
2001	55	800	67.123
2000	56	800	67.096
2000	55	801	67.112

- We can observe that:
$$y_1 = 0.0269 = 67.123 - 67.096$$
$$y_2 = 0 = 67.096 - 67.096$$
$$y_3 = 0.0164 = 67.112 - 67.096$$

Property of Lagrangian dual problem: Weak duality

Weak duality

- Let's p^* and d^* denote the optimal objective value of a primal problem and its dual problem, respectively. We always have

$$d^* \leq p^*$$

  regardless of non-convexity of the primal problem. The difference $p^* - d^*$ is called **duality gap**.
- Weak duality holds even if $p^* = -\infty$, which means the infeasibility of the dual problem. Similarly, if $d^* = +\infty$, the primal problem is infeasible.
- As dual problems are always convex, it is relatively easy to calculate d^* efficiently and thus obtain a lower bound for p^*.

An example of non-zero duality gap

- Consider the following non-convex optimization problem.

$$\begin{array}{lrcl} \min & x_1 x_2 & & \\ s.t. & x_1 & \geq & 0 \\ & x_2 & \geq & 0 \\ & x_1^2 + x_2^2 & \leq & 1 \end{array}$$

- Lagrange dual function:

$$g(\lambda) = \inf_{x \in \mathcal{D}} (x_1 x_2 - \lambda_1 x 1 - \lambda_2 x_2 + \lambda_3 (x_1^2 + x_2^2 - 1))$$

- Dual problem:

$$\begin{array}{lrcl} \max & g(\lambda) & & \\ s.t. & \lambda_1 & \geq & 0 \\ & \lambda_2 & \geq & 0 \\ & \lambda_3 & \geq & \frac{1}{2} \end{array}$$

- Duality gap: $p* = 0 > d^* = -\frac{1}{2}$.

Property of Lagrangian dual problem: Strong duality

Strong duality

- Strong duality holds if $p^* = d^*$, i.e., the duality gap is 0.
- Strong duality doesn't necessarily hold for any optimization problem, but it almost always holds for convex ones, i.e.

$$\begin{array}{rrcll} \min & f_0(x) & & & \\ s.t. & f_i(x) & \leq & 0 & i = 1, ..., m \\ & Ax & = & b & \end{array}$$

  where $f_i(x)$, $(i = 0, 1, ..., m)$ are convex functions.
- The conditions that guarantee strong duality are called **regularity conditions**, one of which is the Slater's condition.

Slater's condition

- Slater's condition: Consider a convex optimization problem. The strong duality holds if there exists a vector $x \in \textbf{relint } \mathcal{D}$ such that
$$f_i(x) < 0, i = 1, ..., m, \quad Ax = b$$
- Suppose the first k constraints are affine, then the Slater's conditions turns into: there exists a vector $x \in \textbf{relint}\mathcal{D}$ such that
$$f_i(x) \leq 0, i = 1, ..., k, \quad f_i(x) < 0, i = 1 + 1, ..., m, \quad Ax = b$$
- Specifically, if all constraints are affine, then the original constraints are themselves the Slater's conditions.
- Please refer to Appendix for the proof of the Slater theorem.

Conditions of optimal solution: KKT conditions

Three types of optimization problems

- It is relatively easy to optimize an objective function **without any constraint**, say:

$$\min \ f_0(x)$$

- But how to optimize an objective function **with equality constraints**?

$$\begin{array}{rrcll} \min & f_0(x) & & & \\ s.t. & h_i(x) & = & 0 & i = 1, 2, ..., p \end{array}$$

- And how to optimize an objective function **with inequality constraints**?

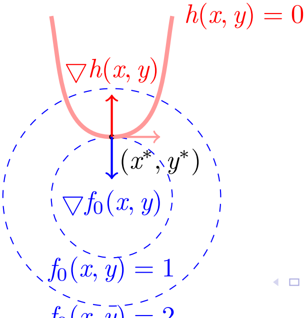
$$\begin{array}{rrcll} \min & f_0(x) & & & \\ s.t. & f_i(x) & \leq & 0 & i = 1, 2, ..., m \\ & h_i(x) & = & 0 & i = 1, 2, ..., p \end{array}$$

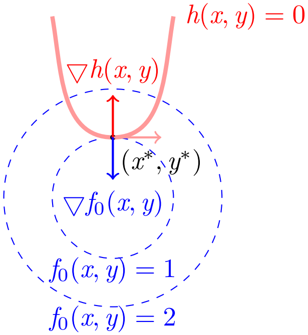
Optimization problem over **equality constraints**

- Consider the following optimization problem:

$$\begin{array}{rlcl} \min & f_0(x,y) & & \\ s.t. & h(x,y) & = & 0 \end{array}$$

- Intuition: suppose (x^*, y^*) is the optimum point. Thus at (x^*, y^*), $f_0(x, y)$ does not change when walking along the curve $h(x,y) = 0$; otherwise, we can follow the curve to make $f_0(x,y)$ smaller, meaning that the starting point (x^*, y^*) is not optimum.





- So at (x^*, y^*), the red line tangentially touches a blue contour, i.e. there exists a real λ such that:

$$\bigtriangledown f_0(x, y) = \lambda \bigtriangledown h(x, y)$$

- Lagrange must have cleverly noticed that the equation above looks like partial derivatives of some larger scalar function:

$$L(x, y, \lambda) = f_0(x, y) - \lambda h(x, y)$$

- Necessary conditions of optimum point: If (x^*, y^*) is local optimum, then there exists a λ such that $\bigtriangledown L(x^*, y^*, \lambda) = 0$.

Understanding Lagrangian function

- Lagrangian function: a combination of the original optimization objective function and constraints:
$$L(x, y, \lambda) = f_0(x, y) - \lambda h(x, y)$$
- The critical point of Lagrangian $L(x, y, \lambda)$ occurs at saddle points rather than local minima (or maxima). Thus, to utilize numerical optimization techniques, we must first transform the problem such that the critical points lie at local minima. This is done by calculating the magnitude of the gradient of Lagrangian.

Optimization problems over **inequality constraints**

- Consider the following optimization problem:

$$\begin{array}{rlll} \min & f_0(x,y) & & \\ s.t. & f_1(x,y) & \leq & 0 \end{array}$$

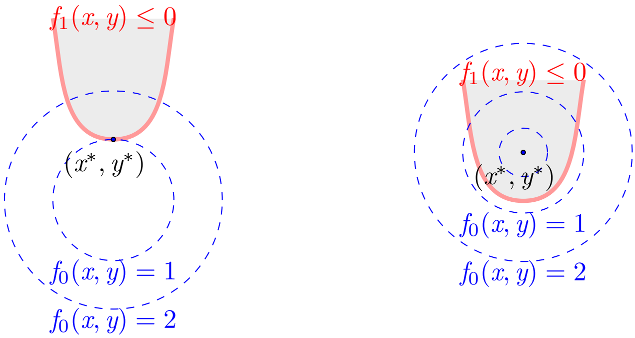


Figure: Case 1: the optimum point (x^*, y^*) lies in the curve $f_1(x,y) = 0$. Thus Lagrangian condition $\bigtriangledown L(x,y,\lambda) = 0$ applies. Case 2: (x^*, y^*) lies within the interior region $f_1(x,y) < 0$; thus we have $\bigtriangledown f_0(x,y) = 0$ at (x^*, y^*)

Complementary slackness

- These two cases can be summarized as the following two conditions:
  - (Stationary point) $\bigtriangledown L(x^*, y^*, \lambda) = 0$
  - (Complementary slackness) $\lambda f_1(x^*, y^*) = 0$
- Reason: In case 2, $f_i(x^*) < 0 \Rightarrow \lambda = 0$ by complementary slackness. We further have $\bigtriangledown f_0(x^*, y^*) = 0$ since $\bigtriangledown L(x^*, y^*, \lambda) = 0$.
- Complementary slackness, also called **orthogonality** by Gomory, essentially equals to the strong duality for convex optimization problems.
- A relaxation of this condition, i.e., $\lambda_i f_i(x^*) = \mu$, where μ is a small positive number, is used in the interior point method.

Proof of complementary slackness

- Assume that strong duality holds, i.e., $p^* = d^*$. Let's x^* and (λ^*, ν^*) $\lambda^* \leq 0$ denote the optimal solution to the primal problem and dual problem, respectively. We have:

$$\begin{aligned}
f_0(x^*) &= g(\lambda^*, \nu^*) \\
&= \inf_{x \in \mathcal{D}} (f_0(x) - \sum_{i=1}^{m} \lambda_i f_i(x) - \sum_{i=1}^{p} \nu_i h_i(x)) \\
&\leq f_0(x^*) - \sum_{i=1}^{m} \lambda_i f_i(x^*) - \sum_{i=1}^{p} \nu_i h_i(x^*) \\
&\leq f_0(x^*)
\end{aligned}$$

- Thus the last two inequalities turns into equalities, which implies that Lagrangian function $L(x, \lambda^*, \nu^*)$ reaches its minimum at x^* and

$$\sum_{i=1}^{m} \lambda_i^* f_i(x^*) = 0.$$

- Note that $\lambda_i^* f_i(x^*) \geq 0$. Hence $\lambda_i^* f_i(x^*) = 0$ for $i = 1, ..., m$.

- Consider the following constrained optimization problem (might be non-convex).

$$\begin{array}{rllll} \min & f_0(x) & & & \\ s.t. & f_i(x) & \leq & 0 & i = 1, ..., m \\ & h_i(x) & = & 0 & i = 1, ..., p \end{array}$$

- Lagrangian function:

$$L(x, \lambda) = f_0(x) - \sum_{i=1}^{m} \lambda_i f_i(x) - \sum_{i=1}^{p} \nu_i h_i(x)$$

- Let x^* and (λ, ν) denote the optimal solutions to the primal and dual problems, respectively. Suppose the strong duality holds.

KKT conditions for general optimization problems II

- Since Lagrangian function reaches its minimum at x^*, its gradient is 0 at x^*, i.e.,

$$\bigtriangledown f_0(x^*) - \sum_{i=1}^{m} \lambda_i \bigtriangledown f_i(x^*) - \sum_{i=1}^{p} \nu_i \bigtriangledown h_i(x^*) = 0$$

- Then x^* and (λ, ν) satisfy the following KKT conditions:
  1. (Stationary point) $\bigtriangledown L(x^*, \lambda, \nu) = 0$
  2. (Primal feasibility) $f_i(x^*) \leq 0,\ i = 1, ..., m;\ h_i(x^*) = 0,$ $i = 1, ..., p$
  3. (Dual feasibility) $\lambda_i \leq 0,\ i = 1, ..., m$
  4. (Complementary slackness) $\lambda_i f_i(x^*) = 0,\ i = 1, ..., m$

KKT conditions for convex problems I

- Consider the following convex optimization problem.

$$\begin{array}{rlll} \min & f_0(x) & & \\ s.t. & f_i(x) & \leq & 0 \quad i = 1, ..., m \\ & Ax & = & b \end{array}$$

- Lagrangian function:

$$L(x, \lambda) = f_0(x) - \sum_{i=1}^{m} \lambda_i f_i(x) - \nu^T(Ax - b)$$

KKT conditions for convex problems II

- x^* and (λ, ν) are optimal solutions to the primal and dual problems, respectively, if they satisfy the following KKT conditions:
  1. (Stationary point) $\bigtriangledown L(x^*, \lambda, \nu) = 0$
  2. (Primal feasibility) $f_i(x^*) \leq 0,\ i = 1, ..., m;\ Ax = b;$
  3. (Dual feasibility) $\lambda_i \leq 0,\ i = 1, ..., m$
  4. (Complementary slackness) $\lambda_i f_i(x^*) = 0,\ i = 1, ..., m$
- KKT conditions, named after William Karush, Harold W. Kuhn, and Albert W. Tucker, are usually not solved directly in optimization; instead, iterative successive approximation is most often used to find the final results that satisfy KKT conditions.

Four properties of duality for linear program

Property 1: Primal is the dual of dual

Theorem

For linear program, primal problem is the dual of dual.

- For a general optimization problem, the dual of dual is not always the primal problem but a convex relaxation of the primal problem.

Property 2: Weak duality

Theorem

(Weak duality) The objective value of **any** *feasible solution to the dual problem is always a lower bound of the objective value of primal problem.*

- It is easy to prove this property as Lagrangian function connects primal objective function and dual objective function.

An example: DIET problem and its dual problem

- Primal problem P:

$$\begin{array}{rrcrcrcrll}
\min & 3x_1 & + & 9x_2 & + & 20x_3 & + & 19x_4 & & \text{money} \\
s.t. & 110x_1 & + & 160x_2 & + & 420x_3 & + & 260x_4 & \geq 2000 & \text{energy} \\
 & 4x_1 & + & 8x_2 & + & 4x_3 & + & 14x_4 & \geq 55 & \text{protein} \\
 & 2x_1 & + & 285x_2 & + & 22x_3 & + & 80x_4 & \geq 800 & \text{calcium} \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & \geq 0 &
\end{array}$$

  Feasible solution $x^T = [0, 8, 2, 0]^T \Rightarrow c^T x = 112$.

- Dual problem D:

$$\begin{array}{rrcrcrll}
\max & 2000y_1 & + & 55y_2 & + & 800y_3 & & \text{money} \\
s.t. & 110y_1 & + & 4y_2 & + & 2y_3 & \leq 3 & \text{oatmeal} \\
 & 160y_1 & + & 8y_2 & + & 285y_3 & \leq 9 & \text{milk} \\
 & 420y_1 & + & 4y_2 & + & 22y_3 & \leq 20 & \text{pie} \\
 & 260y_1 & + & 14y_2 & + & 80y_3 & \leq 19 & \text{pork\&beans} \\
 & y_1 & , & y_2 & , & y_3 & \geq 0 &
\end{array}$$

  Feasible solution $y^T = [0.0269, 0, 0.0164]^T \Rightarrow y^T b = 67.096$

- The theorem states that $c^T x \geq y^T b$ for **any** feasible solutions x and y.

Proof.

- Consider the following PRIMAL problem:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & \geq & b \\ & x & \geq & 0 \end{array}$$

and DUAL problem:

$$\begin{array}{rrcl} \max & b^T y & & \\ s.t. & y & \geq & 0 \\ & A^T y & \leq & c \end{array}$$

- Let x and y denote a feasible solution to primal and dual problems, respectively.
- We have $c^T x \geq y^T A x$ (by the feasibility of dual problem, i.e., $y^T A \leq c^T$, and $x^T \geq 0$)
- Therefore $c^T x \geq y^T A x \geq y^T b$ (by the feasibility of primal problem, i.e., $Ax \geq b$, and $y \geq 0$)

$\square$

Property 3: Strong duality

Theorem

(Strong duality) Consider a linear program. If the primal problem has an optimal solution, then the dual problem also has an optimal solution with the same objective value.

Proof.

- Suppose $x^* = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix}$ be the optimal solution to the primal problem. We have $c^T - c_B^T B^{-1} A \geq 0$.
- Let's set $y^{*T} = c_B{}^T B^{-1}$. We will show that y^{*T} is the optimal solution to the dual problem.
- In fact, we have $y^{*T} b = c_B{}^T B^{-1} b = c^T x^*$.
- That is, $y^{*T} b$ reaches its upper bound. So y^{*T} is an optimal solution to the dual problem.

$\square$

Property 4: Complementary slackness

Theorem

Let x and y denote feasible solutions to the primal and dual problems, respectively. Then x and y are optimal solutions iff $u_i = y_i(a_{i1}x_1 + a_{i2}x_2 + ... + a_{in}x_n - b_i) = 0$ *for any* $1 \leq i \leq m$, *and* $v_j = (c_j - a_{1j}y_1 - a_{2j}y_2 - ... - a_{mj}y_m)x_j = 0$ *for any* $1 \leq j \leq n$.

- Intuition: a constraint of primal problem is loosely restricted $\Rightarrow$ the corresponding dual variable is tight.
- An example: the optimal solutions to DIET and its dual are $x = (14.244, 2.707, 0, 0)$ and $y = (0.0269, 0, 0.0164)$.

$$
\begin{array}{rcrcrcrl}
110x_1 & + & 160x_2 & + & 420x_3 & + & 260x_4 & = 2000 \\
4x_1 & + & 8x_2 & + & 4x_3 & + & 14x_4 & > 55 \quad \Rightarrow y_2 = 0 \\
2x_1 & + & 285x_2 & + & 22x_3 & + & 80x_4 & = 800 \\
x_1 & , & x_2 & , & x_3 & , & x_4 & \geq 0
\end{array}
$$

Proof

Proof.

$u_i = 0$ and $v_j = 0$ for any i and j
$\Leftrightarrow \sum_i u_i = 0$ and $\sum_j v_j = 0$ (since $u_i \geq 0, v_j \geq 0$)
$\Leftrightarrow \sum_i u_i + \sum_j v_j = 0$
$\Leftrightarrow (y^T Ax - y^T b) + (c^T x - y^T Ax) = 0$
$\Leftrightarrow y^T b = c^T x$
$\Leftrightarrow y$ and x are optimal solutions (by strong duality property, i.e., both $y^T b$ and $c^T x$ reach its bound)

$\square$

Summary: 9 cases of primal and dual problems

Primal / Dual	Bounded Optimal Objective Value	Unbounded Optimal Objective Value	Infeasible
Bounded Optimal Objective Value	Possible	Impossible	Impossible
Unbounded Optimal Objective Value	Impossible	Impossible	Possible
Infeasible	Impossible	Possible	Possible

Example 1: PRIMAL has unbounded objective value and DUAL is infeasible

PRIMAL:

$$\begin{array}{rrcrcr} \min & -2x_1 & - & 2x_2 & & \\ s.t. & x_1 & - & x_2 & \leq & 1 \\ & -x_1 & + & x_2 & \leq & 1 \\ & x_1 & & & \geq & 0 \\ & & & x_2 & \geq & 0 \end{array}$$

DUAL:

$$\begin{array}{rrcrcr} \max & y_1 & + & y_2 & & \\ s.t. & y_1 & & & \leq & 0 \\ & & & y_2 & \leq & 0 \\ & y_1 & - & y_2 & \leq & -2 \\ & -y_1 & + & y_2 & \leq & -2 \end{array}$$

Example 2: both PRIMAL and DUAL are infeasible

PRIMAL:

$$\begin{array}{lrcrcr} \min & x_1 & - & 2x_2 & & \\ s.t. & x_1 & - & x_2 & \geq & 2 \\ & -x_1 & + & x_2 & \geq & -1 \\ & x_1 & & & \geq & 0 \\ & & & x_2 & \geq & 0 \end{array}$$

DUAL:

$$\begin{array}{lrcrcr} \max & 2y_1 & - & y_2 & & \\ s.t. & y_1 & & & \geq & 0 \\ & & & y_2 & \geq & 0 \\ & y_1 & - & y_2 & \leq & 1 \\ & -y_1 & + & y_2 & \leq & -2 \end{array}$$

Solving linear program using duality

KKT conditions for linear program

- Consider a linear program in slack form and its dual problem:

$$\begin{array}{llll} \min & c^T x & & \\ s.t. & Ax & = & b \\ & x & \geq & 0 \end{array} \qquad \begin{array}{llll} \max & b^T y & & \\ s.t. & y^T A & \leq & c^T \\ & & . & \end{array}$$

The KKT conditions turns into: x and y are optimal solutions to the primal and dual problems, respectively, if they satisfy the following three conditions:

1. **(Primal feasibility)**

$$Ax = b, x \geq 0$$

2. **(Dual feasibility)**

$$y^T A \leq c^T$$

3. **(Complementary slackness)**

$$c^T x = y^T b$$

- Question: How to obtain x and y that satisfy these three conditions simultaneously?

IMPROVEMENT framework

- We could start with an initial value of x and y that satisfy two constraints, and attempt to improve them to reduce the unsatisfiability of the third constraint. This improvement steps will be repeated until all of the three constraints are satisfied.

```
Initialize (x, y) with values that satisfy two constraints;
while TRUE do
    Improve x and y to reduce the unsatisfiability of the third
    constraint.
    if all the three constraints are satisfied then
        break;
    end if
end while
return (x, y);
```

Strategy 1

$x = x_0$; //Initialize x with a **primal feasible solution**
$y = y_0$; //Calculate initial y according to **complementary slackness**
while TRUE **do**
    $x =$ IMPROVE(x);
    //Improve x to **reduce dual infeasibility of corresponding** y. Throughout this process, primal feasibility is maintained and y is recalculated according to complementary slackness
    if y is **dual feasible** **then**
        break;
    end if
end while
return x;

- Example: Primal simplex

Strategy 2

$y = y_0$; //Initialize y with a **dual feasible solution**

$x = x_0$; //Calculate initial x according to **complementary slackness**

while TRUE **do**

  $y = \textsc{Improve}(y)$;

  //Improve y to **reduce primal infeasibility of corresponding** x. Throughout this process, dual feasibility is maintained and x is recalculated according to complementary slackness

  if x **is primal feasible then**

    break;

  end if

end while

return y;

- Example: Dual simplex, Primal and Dual method

Strategy 3

$x = x_0$; //Initialize x with a **primal feasible solution**
$y = y_0$; //Initialize y with a **dual feasible solution**
while TRUE **do**
  $(x, y) = \textsc{Improve}(x, y)$; //Improve x and y to reduce the unsatisfiability of **complementary slackness**
  if (x, y) satisfies **complementary slackness** **then**
    break;
  end if
end while
return y;

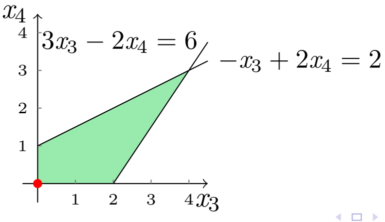
- Example: Interior point method

Dual simplex method

Revisiting primal simplex: An example

$$
\begin{array}{llllllll}
\min & & & -x_3 & +x_4 & & \\
s.t. & x_1 & & -x_3 & +2x_4 & = & 2 \\
& & x_2 & +3x_3 & -2x_4 & = & 6 \\
& x_1, & x_2, & x_3, & x_4 & \geq & 0
\end{array}
$$

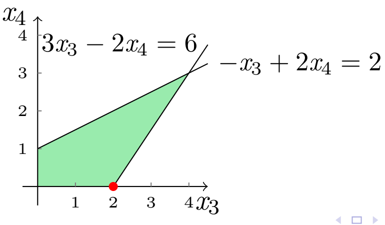
	x_1	x_2	x_3	x_4	RHS
Basis	$\overline{c_1}=0$	$\overline{c_2}=0$	$\overline{c_3}=-1$	$\overline{c_4}=1$	$-z=0$
x_1	1	0	-1	2	2
x_2	0	1	3	-2	6



Step 2

$$\begin{array}{llllll} \min & & & -x_3 & +x_4 & & \\ s.t. & x_1 & & -x_3 & +2x_4 & = & 2 \\ & & x_2 & +3x_3 & -2x_4 & = & 6 \\ & x_1, & x_2, & x_3, & x_4 & \geq & 0 \end{array}$$

	x_1	x_2	x_3	x_4	RHS
Basis	$\overline{c_1} = 0$	$\overline{c_2} = \frac{1}{3}$	$\overline{c_3} = 0$	$\overline{c_4} = \frac{1}{3}$	$-z = 2$
x_1	1	$\frac{1}{3}$	0	$\frac{4}{3}$	4
x_3	0	$\frac{1}{3}$	1	$-\frac{2}{3}$	2



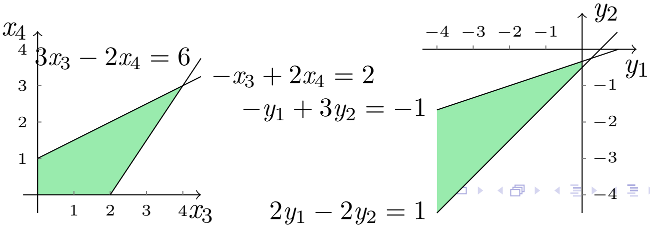
The viewpoint of dual problem

- Primal problem P:

$$\begin{array}{rccccl} \min & & & -x_3 & +x_4 & \\ s.t. & x_1 & & -x_3 & +2x_4 & = 2 \\ & & x_2 & +3x_3 & -2x_4 & = 6 \\ & x_1, & x_2, & x_3, & x_4 & \geq 0 \end{array}$$

- Dual problem D:

$$\begin{array}{rcccl} \max & 2y_1 & +6y_2 & & \\ s.t. & y_1 & & \leq & 0 \\ & & y_2 & \leq & 0 \\ & -y_1 & +3y_2 & \leq & -1 \\ & 2y_1 & -2y_2 & \leq & 1 \end{array}$$



Set primal and dual solutions according to a basis

- Let's consider a linear program in slack form and its dual problem, i.e.
  - Primal problem:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & = & b \\ & x & \geq & 0 \end{array}$$

  - Dual problem:

$$\begin{array}{rrcl} \max & b^T y & & \\ s.t. & A^T y & \leq & c \end{array}$$

- From any basis B of A, we can set a primal solution x and a dual solution y simultaneously, i.e.,
  - **Primal solution**: $x = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix}$. x is feasible if $B^{-1}b \geq 0$.
  - **Dual solution**: $y^T = c_B^T B^{-1}$. y is feasible if $y^T A \leq c^T$, i.e., $\overline{c^T} = c^T - c_B^T B^{-1} A = c^T - y^T A \geq 0$.
- Note that by this setting, **complementary slackness** follows as

$$c^T x = c_B^T B^{-1} b = y^T b$$

Primal feasible basis

- Consider the PRIMAL problem:

$$
\begin{array}{lrrrrcr}
\min & & & -x_3 & +x_4 & & \\
s.t. & x_1 & & -x_3 & +2x_4 & = & 2 \\
 & & x_2 & +3x_3 & -2x_4 & = & 6 \\
 & x_1, & x_2, & x_3, & x_4 & \geq & 0
\end{array}
$$

	x_1	x_2	x_3	x_4	RHS
Basis	$\overline{c_1}=0$	$\overline{c_2}=0$	$\overline{c_3}=-1$	$\overline{c_4}=1$	$-z=0$
x_1	1	0	-1	2	2
x_2	0	1	3	-2	6

- Primal variables: x; Feasible: $B^{-1}b \geq 0$.
- Basis B is called **primal feasible** if all elements in $B^{-1}b$ (**the first column except for $-z$**) are non-negative.

Dual feasible basis

- Now let's consider the DUAL problem:

$$\begin{array}{rrrcr} \max & 2y_1 & +6y_2 & & \\ s.t. & y_1, & y_2 & \leq & 0 \\ & -y_1 & +3y_2 & \leq & -1 \\ & 2y_1 & -2y_2 & \leq & 1 \end{array}$$

- Consider PRIMAL simplex tabular again:

	x_1	x_2	x_3	x_4	RHS
Basis	$\overline{c_1}=0$	$\overline{c_2}=0$	$\overline{c_3}=-1$	$\overline{c_4}=1$	$-z=0$
x_1	1	0	-1	2	2
x_2	0	1	3	-2	6

- Dual variables: $y^T = c_B^T B^{-1}$; Feasible: $y^T A \leq c^T$.
- Basis B is called **dual feasible** if all elements in $\overline{c^T} = c^T - c_B^T B^{-1} A = c^T - y^T A$ (**the first row except for** $-z$) are non-negative.

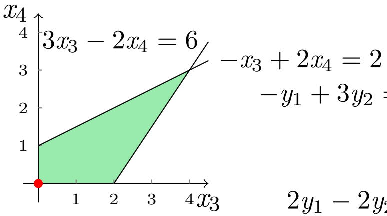
- Thus another view point of the PRIMAL SIMPLEX algorithm can be described as:
  1. **Starting point:** The PRIMAL SIMPLEX algorithm starts with a primal basic feasible solution (the first column in simplex table $x_B = B^{-1}b \geq 0$). By setting dual variable $y^T = c_B^T B^{-1}$, the complementary slackness holds, i.e., $c^T x = c^T B^{-1} b = y^T b$.
  2. **Improvement:** By pivoting basis, we move towards dual feasibility, i.e. the first row in simplex table $\overline{c^T} = c^T - c_B^T B^{-1} A = c^T - y^T A \geq 0$. Here, we **minimize dual infeasibility** by selecting a negative element in the first row in the pivoting process. Throughout the process we maintain the primal feasibility and complementary slackness, i.e., $c^T x = c^T B^{-1} b = y^T b$.
  3. **Stopping criterion:** $\overline{c}^T = c^T - c_B^T B^{-1} A \geq 0$, i.e., $y^T A \leq c^T$. In other words, the iteration process ends when the basis is both primal feasible and dual feasible.

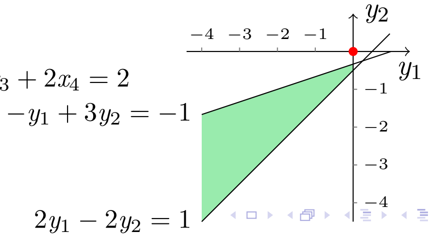
Another viewpoint of primal simplex: Step 1

$$
\begin{array}{lccccccc}
\min & & & -x_3 & +x_4 & & \\
s.t. & x_1 & & -x_3 & +2x_4 & = & 2 \\
 & & x_2 & +3x_3 & -2x_4 & = & 6 \\
 & x_1, & x_2, & x_3, & x_4 & \geq & 0
\end{array}
$$

	x_1	x_2	x_3	x_4	RHS
Basis	$\overline{c_1}=0$	$\overline{c_2}$=0	$\overline{c_3}$=-1	$\overline{c_4}$=1	$-z=0$
x_1	1	0	-1	2	2
x_2	0	1	3	-2	6

- Dual solution: $y^T = c_B^T B^{-1} = [0\ 0]$, which is infeasible.



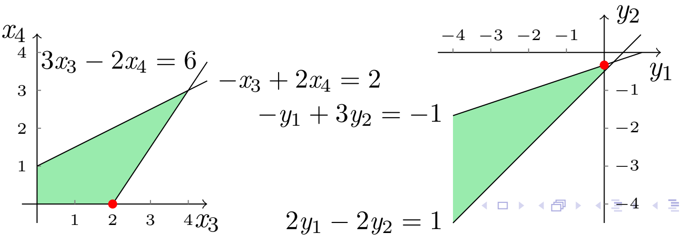


Another viewpoint of primal simplex: Step 2

$$
\begin{array}{llllllll}
\min & & & -x_3 & +x_4 & & \\
s.t. & x_1 & & -x_3 & +2x_4 & = & 2 \\
 & & x_2 & +3x_3 & -2x_4 & = & 6 \\
 & x_1, & x_2, & x_3, & x_4 & \geq & 0
\end{array}
$$

	x_1	x_2	x_3	x_4	RHS
Basis	$\overline{c_1}=0$	$\overline{c_2}=\frac{1}{3}$	$\overline{c_3}=0$	$\overline{c_4}=\frac{1}{3}$	$-z=2$
x_1	1	$\frac{1}{3}$	0	$\frac{4}{3}$	4
x_3	0	$\frac{1}{3}$	1	$-\frac{2}{3}$	2

- Dual solution: $y^T = c_B^T B^{-1} = [0 \;\; -\frac{1}{3}]$, which is feasible.



Dual simplex works in just an opposite fashion

- Dual simplex:
  1. **Starting point:** The Dual Simplex algorithm starts with a dual basic feasible solution $y^T = c_B^T B^{-1}$ such that $y^T A \leq c^T$, i.e., the first row in simplex table $\overline{c^T} = c^T - c_B^T B^{-1} A = c^T - y^T A \geq 0$. By setting primal variables $x_B = B^{-1} b$ and $x_N = 0$, the complementary slackness holds, i.e., $c^T x = c^T B^{-1} b = y^T b$.
  2. **Improvement:** By pivoting basis, we move towards primal feasibility, i.e. the first column in simplex table $B^{-1} b \geq 0$. Here, we **minimize primal infeasibility** by selecting a negative element from the first column in the pivoting process. Throughout the process we maintain the dual feasibility and complementary slackness, i.e., $c^T x = c^T B^{-1} b = y^T b$.
  3. **Stopping criterion:** $x_B = B^{-1} b \geq 0$. In other words, the iteration process ends when the basis is both primal feasible and dual feasible.

PRIMAL SIMPLEX vs. DUAL SIMPLEX

- Both PRIMAL SIMPLEX and DUAL SIMPLEX terminate at the same condition, i.e. the basis is primal feasible and dual feasible simultaneously.
- However, the final objective is achieved in totally opposite fashions— the PRIMAL SIMPLEX method keeps the primal feasibility while the DUAL SIMPLEX method keeps the dual feasibility during the pivoting process.
- The PRIMAL SIMPLEX algorithm *first selects an entering variable and then determines the leaving variable*.
- In contrast, the DUAL SIMPLEX method does the opposite; it *first selects a leaving variable and then determines an entering variable*.

DUAL SIMPLEX(B_I, z, A, b, c)

//DUAL SIMPLEX starts with a dual feasible basis. Here, B_I contains the indices of the basic variables.

while TRUE **do**
    if there is no index l $(1 \leq l \leq m)$ has $b_l < 0$ **then**
        $x =$CALCULATEX(B_I, A, b, c);
        return (x, z);
    end if;
    choose an index l having $b_l < 0$ according to a certain rule;
    for each index j $(1 \leq i \leq n)$ **do**
        if $a_{lj} < 0$ **then**
            $\Delta_j = -\frac{c_j}{a_{lj}}$;
        else
            $\Delta_j = \infty$;
        end if
    end for
    choose an index e that minimizes Δ_j;
    if $\Delta_e = \infty$ **then**
        return ``no feasible solution'';
    end if
    $(B_I, A, b, c, z) =$ PIVOT(B_I, A, b, c, z, e, l);
end while

An example

- Standard form:

$$\begin{array}{rrcrcrcr}
\min & 5x_1 & + & 35x_2 & + & 20x_3 & & \\
s.t. & x_1 & - & x_2 & - & x_3 & \leq & -2 \\
 & -x_1 & - & 3x_2 & & & \leq & -3 \\
 & x_1 & , & x_2 & , & x_3 & \geq & 0
\end{array}$$

- Slack form:

$$\begin{array}{rrcrcrcrcrcr}
\min & 5x_1 & + & 35x_2 & + & 20x_3 & & & & & & \\
s.t. & x_1 & - & x_2 & - & x_3 & + & x_4 & & & = & -2 \\
 & -x_1 & - & 3x_2 & & & & & + & x_5 & = & -3 \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & , & x_5 & \geq & 0
\end{array}$$

Step 1

	x_1	x_2	x_3	x_4	x_5	RHS
Basis	$\overline{c_1}=5$	$\overline{c_2}=35$	$\overline{c_3}=20$	$\overline{c_4}=0$	$\overline{c_5}=0$	$-z=0$
x_4	1	-1	-1	1	0	-2
x_5	-1	-3	0	0	1	-3

- Basis (in blue): $B = \{a_4, a_5\}$
- Solution: $x = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix} = (0, 0, 0, -2, -3)$.
- Pivoting: choose a_5 to leave basis since $b'_2 = -3 < 0$; choose a_1 to enter basis since $\min_{j, a_{2j}<0} \frac{\overline{c}_j}{-a_{2j}} = \frac{\overline{c}_1}{-a_{21}}$.

Step 2

	x_1	x_2	x_3	x_4	x_5	RHS
Basis	$\overline{c_1}$= 0	$\overline{c_2}$=20	$\overline{c_3}$=20	$\overline{c_4}$=0	$\overline{c_5}$=5	$-z = -15$
x_4	0	-4	-1	1	1	-5
x_1	1	3	0	0	-1	3

- Basis (in blue): $B = \{a_1, a_4\}$
- Solution: $x = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix} = (3, 0, 0, -5, 0)$.
- Pivoting: choose a_4 to leave basis since $b'_1 = -5 < 0$; choose a_2 to enter basis since $\min_{j, a_{1j}<0} \frac{\bar{c}_j}{-a_{1j}} = \frac{\bar{c}_2}{-a_{12}}$.

Step 3

	x_1	x_2	x_3	x_4	x_5	RHS
Basis	$\overline{c_1}=0$	$\overline{c_2}=0$	$\overline{c_3}=15$	$\overline{c_4}=5$	$\overline{c_5}=10$	$-z=-40$
x_2	0	1	$\frac{1}{4}$	$-\frac{1}{4}$	$-\frac{1}{4}$	$\frac{5}{4}$
x_1	1	0	$-\frac{3}{4}$	$\frac{3}{4}$	$-\frac{1}{4}$	$-\frac{3}{4}$

- Basis (in blue): $B = \{a_1, a_2\}$
- Solution: $x = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix} = (\frac{5}{4}, -\frac{3}{4}, 0, 0, 0)$.
- Pivoting: choose a_1 to leave basis since $b'_2 = -\frac{3}{4} < 0$; choose a_3 to enter basis since $\min_{j, a_{2j}<0} \frac{\bar{c}_j}{-a_{2j}} = \frac{\bar{c}_3}{-a_{23}}$.

Step 4

	x_1	x_2	x_3	x_4	x_5	RHS
Basis	$\overline{c_1}= 20$	$\overline{c_2}=0$	$\overline{c_3}=0$	$\overline{c_4}=20$	$\overline{c_5}=5$	$-z = -55$
x_2	$\frac{1}{3}$	1	0	0	$-\frac{1}{3}$	1
x_3	$-\frac{4}{3}$	0	1	-1	$\frac{1}{3}$	1

- Basis (in blue): $B = \{a_2, a_3\}$
- Solution: $x = \begin{bmatrix} B^{-1}b \\ 0 \end{bmatrix} = (0, 1, 1, 0, 0)$.
- Done!

When dual simplex method is useful?

- The dual simplex algorithm is most suited for problems for which an **initial dual feasible solution** is easily available. It is particularly useful for **reoptimizing** a problem after a constraint has been added or some parameters have been changed so that primal feasibility was destroyed.
- An example is mixed integer programming: branching at a fractional variable creates two sub-problems, each of which has **a newly added constraint on the variable**. The addition of new constraints or valid inequality usually breaks the primal feasibility. However, dual feasibility usually holds as adding a row in primal corresponds to adding a column in dual.
- Trying dual simplex is particularly useful if your LP appears to be **highly degenerate**, i.e. there are many vertices of the feasible region for which the associated basis is degenerate. We may find that a large number of iterations (moves between adjacent vertices) occur with little or no improvement.[1]

[1] References: Operations Research Models and Methods, Paul A. Jensen and Jonathan F. Bard; OR-Notes, J. E. Beasley

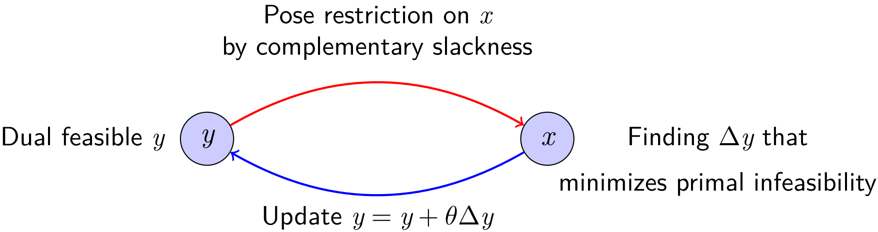
PRIMAL AND DUAL method: another strategy to find a pair of primal and dual variables satisfying KKT conditions

PRIMAL AND DUAL method: a brief history

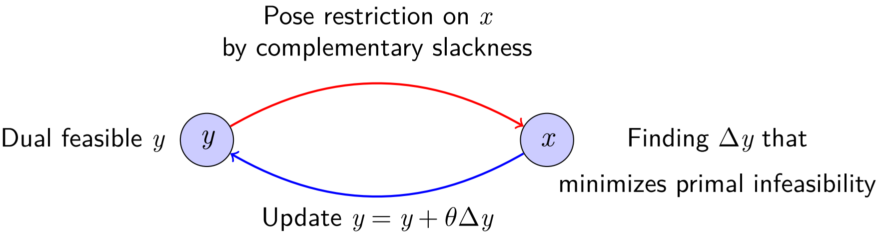
- In 1955, H. Kuhn proposed the Hungarian method for the MAXWEIGHTEDMATCHING problem. This method effectively explores the duality property of linear programming.
- In 1956, G. Dantzig, R. Ford, and D. Fulkerson extended this idea to solve linear programming problems.
- In 1957, R. Ford, and D. Fulkerson applied this idea to solve network-flow problem and Hitchcock problem.
- In 1957, J. Munkres applied this idea to solve the transportation problem.

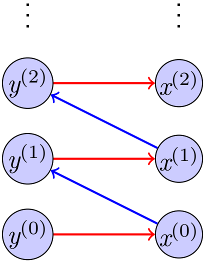
Primal and Dual method

- Basic idea: It is not easy to find primal variables x and dual variables y to satisfy the KKT conditions simultaneously. A reasonable strategy is starting from a dual feasible y, and pose restrictions on x according to complementary slackness. Next a step-by-step improvement procedure was performed towards primal feasibility of x. To achieve this goal, we **minimize primal infeasibility** while maintaining both dual feasibility and complementary slackness.



Basic idea of PRIMAL AND DUAL method





Difference between PRIMAL AND DUAL and dual simplex

- Both PRIMAL AND DUAL and dual simplex algorithms start with x and y that satisfy complementary slackness; however, they differ in how to obtain such x and y.
- In dual simplex, x and y are generated from the same basis B as follows:

$$\begin{tikzcd} & B \arrow[dl] \arrow[dr] & \\ y & & x \end{tikzcd}$$

  We set $x = B^{-1}b$ and $y = c_B B^{-1}$. The complementary slackness follows since $c^T x = b^T y$.
- In PRIMAL AND DUAL approach, we derive some x_i from y directly:

$$y \longrightarrow x$$

$$a_{1i}y_1 + a_{2i}y_2 + \ldots + a_{mi}y_m < c_i \Rightarrow x_i = 0$$

Three steps of PRIMAL AND DUAL method

- Primal P:

$$\begin{array}{rccccccccccc}
\min & c_1 x_1 & + & c_2 x_2 & + & \ldots & + & c_n x_n & & & \\
s.t. & a_{11} x_1 & + & a_{12} x_2 & + & \ldots & + & a_{1n} x_n & = & b_1 & (y_1) \\
 & a_{21} x_1 & + & a_{22} x_2 & + & \ldots & + & a_{2n} x_n & = & b_2 & (y_2) \\
 & & & & & \ldots & & & & & \\
 & a_{m1} x_1 & + & a_{m2} x_2 & + & \ldots & + & a_{mn} x_n & = & b_m & (y_m) \\
 & x_1 & , & x_2 & , & \ldots & , & x_n & \geq & 0 &
\end{array}$$

- Dual D:

$$\begin{array}{rccccccccccc}
\max & b_1 y_1 & + & b_2 y_2 & + & \ldots & + & b_m y_m & & & \\
s.t. & a_{11} y_1 & + & a_{21} y_2 & + & \ldots & + & a_{m1} y_m & \leq & c_1 & (x_1) \\
 & a_{12} y_1 & + & a_{22} y_2 & + & \ldots & + & a_{m2} y_m & \leq & c_2 & (x_2) \\
 & & & & & \ldots & & & & & \\
 & a_{1n} y_1 & + & a_{2n} y_2 & + & \ldots & + & a_{mn} y_m & \leq & c_n & (x_n)
\end{array}$$

- Let's start with a **dual feasible solution** y and construct a primal solution x that satisfies the **complementary slackness** first.

Step 1: $y \Rightarrow x$ I

- Dual problem D:

$$\begin{array}{lccccccccccl} \max & b_1 y_1 & + & b_2 y_2 & + & \dots & + & b_m y_m & & & \\ s.t. & a_{11} y_1 & + & a_{21} y_2 & + & \dots & + & a_{m1} y_m & \leq & c_1 & ('=' \Rightarrow x_1 \geq 0) \\ & & & & & \dots & & & & & \\ & a_{1n} y_1 & + & a_{2n} y_2 & + & \dots & + & a_{mn} y_m & \leq & c_n & ('<' \Rightarrow x_n = 0) \end{array}$$

- How to set x that satisfies the complementary slackness?
  1. Let's use J to record the index of **tight constraints** where "$=$" holds. We set $x_i = 0$ if the ith constraint is not tight: $a_{1i}y_1 + a_{2i}y_2 + \dots + a_{mi}y_m < c_i \Rightarrow x_i = 0$
  (Reason: Complement slackness requires that $(a_{1i}y_1 + a_{2i}y_2 + \dots + a_{mi}y_m - c_i) \times x_i = 0$.)

Step 1: $y \Rightarrow x$ II

$$
\begin{array}{ccccc}
 & & & & x_n = 0 \\
 & & & & \Uparrow \\
 & J & & & \\
c_1 & c_2 & \cdots & & c_n \\
\| & \| & & & \vee \\
y_1 a_{11} & y_1 a_{12} & \cdots & & y_1 a_{1n} \\
+ & + & & & + \\
y_2 a_{21} & y_2 a_{22} & \cdots & & y_2 a_{2n} \\
+ & + & & & + \\
\vdots & \vdots & \cdots & & \vdots \\
+ & + & & & + \\
y_m a_{m1} & y_m a_{m2} & \cdots & & y_m a_{mn}
\end{array}
$$

Step 1: $y \Rightarrow x$ III

2. Now y is dual feasible, and we represent the complementary slackness as
$$x_i = 0, \quad i \notin J$$
If we can find a x such that x is primal feasible and satisfies the complementary slackness, then all the three KKT conditions hold and thus both x and y are optimal solution.
3. We can find such x through solving the following **restricted primal (RP)**, which has complementary slackness as an appended constraint:

$$\begin{array}{rcrcccccrcll}
a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & = & b_1 & \\
a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & = & b_2 & \\
 & & & & \dots & & & & & \\
a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & = & b_m & \\
 & & & & & & x_i & = & 0 & i \notin J \\
 & & & & & & x_i & \geq & 0 & i \in J
\end{array}$$

But how to solve RP? I

- RP:

$$\begin{array}{ccccccccccc}
a_{11}x_1 & + & a_{12}x_2 & + & \ldots & + & a_{1n}x_n & = & b_1 & \\
a_{21}x_1 & + & a_{22}x_2 & + & \ldots & + & a_{2n}x_n & = & b_2 & \\
 & & & & \ldots & & & & & \\
a_{m1}x_1 & + & a_{m2}x_2 & + & \ldots & + & a_{mn}x_n & = & b_m & \\
 & & & & & & x_i & = & 0 & i \notin J \\
 & & & & & & x_i & \geq & 0 & i \in J
\end{array}$$

- How to solve RP? Recall that $Ax = b, x \geq 0$ can be solved via solving an extended LP.

But how to solve RP? II

- RP (extended through introducing slack variables):

$$
\begin{array}{llllllllll}
\min & w = s_1 & +s_2 & \dots & +s_m & & & & & \\
s.t. & s_1 & & & & +a_{11}x_1 & \dots & +a_{1n}x_n & = b_1 & \\
 & & s_2 & & & +a_{21}x_1 & \dots & +a_{2n}x_n & = b_2 & \\
 & & & \dots & & & \dots & & & \\
 & & & & s_m & +a_{m1}x_1 & \dots & +a_{mn}x_n & = b_m & \\
 & & & & & & & {\color{blue}x_i} & {\color{blue}= 0} & {\color{blue}i \notin J} \\
 & & & & & & & {\color{red}x_i} & {\color{red}\geq 0} & {\color{red}i \in J} \\
 & & & & & & & s_i & \geq 0 & \forall i
\end{array}
$$

- Intuitively, RP aims to **minimize infeasibility** of $Ax = b, x \geq 0$.
  1. If $w_{OPT} = 0$, then we find a feasible solution to RP, implying that y is an optimal solution;
  2. If $w_{OPT} > 0$, y is not an optimal solution.

- Alternatively, we can solve the dual of RP, called DRP:

$$\begin{array}{rrcrcccrcc}
\max & w = b_1 y_1 & + & b_2 y_2 & + & \dots & + & b_m y_m & & \\
s.t. & a_{11} y_1 & + & a_{21} y_2 & + & \dots & + & a_{m1} y_m & \leq & 0 \\
 & a_{12} y_1 & + & a_{22} y_2 & + & \dots & + & a_{m2} y_m & \leq & 0 \\
 & & & & & \dots & & & & \\
 & a_{1|J|} y_1 & + & a_{2|J|} y_2 & + & \dots & + & a_{m|J|} y_m & \leq & 0 \\
 & y_1, & & y_2, & & \dots & & y_m & \leq & 1
\end{array}$$

1. If $w_{OPT} = 0$, y is an optimal solution.
2. If $w_{OPT} > 0$, y is not an optimal solution. However, the optimal solution to DRP, denoted as Δy, can be used to improve y as RP **aims to minimize primal infeasibility**.

The difference between DRP and D

- Dual problem D:

$$\begin{array}{rccccccccc}
\max & b_1 y_1 & + & b_2 y_2 & + & \dots & + & b_m y_m & & \\
s.t. & a_{11} y_1 & + & a_{21} y_2 & + & \dots & + & a_{m1} y_m & \le & c_1 \\
 & a_{12} y_1 & + & a_{22} y_2 & + & \dots & + & a_{m2} y_m & \le & c_2 \\
 & & & & & \dots & & & & \\
 & a_{1n} y_1 & + & a_{2n} y_2 & + & \dots & + & a_{mn} y_m & \le & c_n
\end{array}$$

- DRP:

$$\begin{array}{rccccccccc}
\max & w = b_1 y_1 & + & b_2 y_2 & + & \dots & + & b_m y_m & & \\
s.t. & a_{11} y_1 & + & a_{21} y_2 & + & \dots & + & a_{m1} y_m & \le & 0 \\
 & a_{12} y_1 & + & a_{22} y_2 & + & \dots & + & a_{m2} y_m & \le & 0 \\
 & & & & & \dots & & & & \\
 & a_{1|J|} y_1 & + & a_{2|J|} y_2 & + & \dots & + & a_{m|J|} y_m & \le & 0 \\
 & y_1, & & y_2, & & \dots & & y_m & \le & 1
\end{array}$$

- How to write DRP from D?
  - Replacing c_i with 0;
  - Keeping only $|J|$ restrictions in DRP;
  - Additional constraints: $y_1, y_2, ..., y_m \le 1$;

Step 3: $\Delta y \Rightarrow y$ I

- Why Δy can be used to improve y?
- From the point of view of RP, the corresponding primal variables x minimizes primal infeasibility.
- Now we explain this from the point of view of objective value. Consider an improved dual solution $y' = y + \theta\Delta y, \theta > 0$. We have:
- **Objective function:** Since $\Delta y^T b = w_{OPT} > 0$, $y'^T b = y^T b + \theta w_{OPT} > y^T b$. In other words, $(y + \theta\Delta y)$ is better than y.
- **Constraints:** The dual feasibility requires that:
    - For any $j \in J$, $a_{1j}\Delta y_1 + a_{2j}\Delta y_2 + ... + a_{mj}\Delta y_m \leq 0$. Thus we have $y'^T a_j = y^T a_j + \theta\Delta y^T a_j \leq c_j$ for any $\theta > 0$.

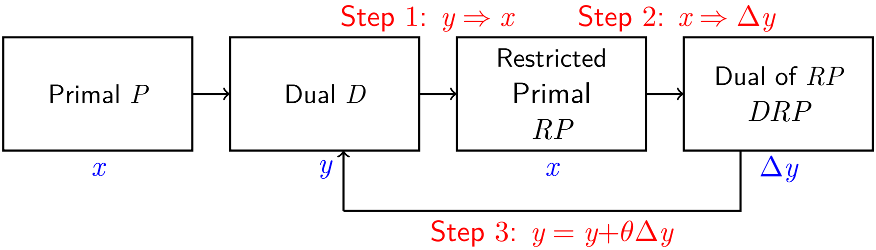
Step 3: $\Delta y \Rightarrow y$ II

- For any $j \notin J$, there are two cases:
  1. $\forall j \notin J, a_{1j}\Delta y_1 + a_{2j}\Delta y_2 + ... + a_{mj}\Delta y_m \leq 0$:
     Thus y' is feasible for any $\theta > 0$ since for $\forall 1 \leq j \leq n$,

$$\begin{aligned} & a_{1j}y'_1 + a_{2j}y'_2 + ... + a_{mj}y'_m && (1)\\ = \quad & a_{1j}y_1 + a_{2j}y_2 + ... + a_{mj}y_m && (2)\\ + \quad & \theta(a_{1j}\Delta y_1 + a_{2j}\Delta y_2 + ... + a_{mj}\Delta y_m) && (3)\\ \leq \quad & c_j && (4)\end{aligned}$$

     Hence dual problem D is unbounded and the primal problem P is infeasible.
  2. $\exists j \notin J, a_{1j}\Delta y_1 + a_{2j}\Delta y_2 + ... + a_{mj}\Delta y_m > 0$:
     We can safely set $\theta \leq \frac{c_j - (a_{1j}y_1 + a_{2j}y_2 + ... + a_{mj}y_m)}{a_{1j}\Delta y_1 + a_{2j}\Delta y_2 + ... + a_{mj}\Delta y_m} = \frac{c_j - y^T a_j}{\Delta y^T a_j}$ to guarantee that $y'^T a_j = y^T a_j + \theta \Delta y^T a_j \leq c_j$.

Primal and dual algorithm



Primal and dual algorithm

Infeasible = "No"
Optimal = "No"
$y = y_0$; $//y_0$ is a feasible solution to the dual problem D
while TRUE **do**
  Finding tight constraints index J, and set corresponding $x_j = 0$ for $j \notin J$.
  Thus we have a smaller RP.
  Solve DRP. Denote the solution as Δy.
  if DRP objective function $w_{OPT} = 0$ **then**
    Optimal="Yes"
    return y;
  end if
  if $\Delta y^T a_j \leq 0$ (for all $j \notin J$) **then**
    Infeasible = "Yes";
    return ;
  end if
  Set $\theta = \min \frac{c_j - y^T a_j}{\Delta y^T a_j}$ for $\Delta y^T a_j > 0$, $j \notin J$.
  Update y as $y = y + \theta \Delta y$;
end while

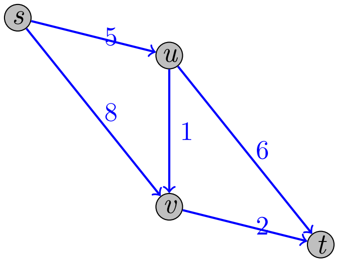
Advantages of PRIMAL AND DUAL algorithm

- Primal and dual algorithm ends if using anti-cycling rule. (Reason: the objective value $y^T b$ increases if there is no degeneracy.)
- Both RP and DRP do not explicitly rely on c. In fact, **the information of c is represented in J**.
- This leads to another advantage of primal and dual technique, i.e., **RP is usually a purely combinatorial problem**. Take SHORTESTPATH as an example. RP corresponds to a "connection" problem. An optimal solution to DRP usually has combinatorial explanation, especially for graph-theory problems.
- More and more constraints become tight in the primal and dual process.
- Unlike dual simplex starting from a **dual basic feasible solution**, PRIMAL AND DUAL method requires only a **dual feasible solution**. (See Lecture 10 for a primal and dual algorithm for MAXIMUMFLOW problem.)

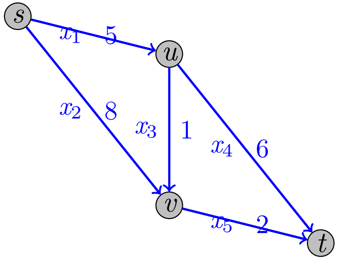
SHORTESTPATH: Dijkstra's algorithm is essentially Primal_Dual algorithm

ShortestPath problem

INPUT: n cities, and a collection of roads. A road from city i to j has a distance $d(i,j)$. Two specific cities: s and t.
OUTPUT: the shortest path from city s to t.



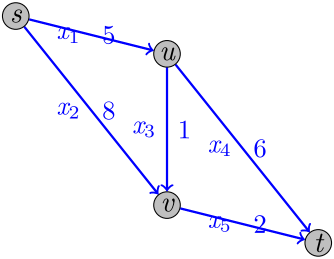
SHORESTPATH problem: PRIMAL problem



- PRIMAL problem: set variables for **roads** (Intuition: $x_i = 0/1$ means whether edge i appears in the shortest path), and a constraint means that *"we enter a node through an edge and leaves it through another edge"*.

$$
\begin{array}{rrrrrrrrrrrll}
\min & 5x_1 & + & 8x_2 & + & 1x_3 & + & 6x_4 & + & 2x_5 & & & \\
s.t. & x_1 & + & x_2 & & & & & & & = 1 & & \text{vertex } s \\
 & & & & & & - & x_4 & - & x_5 & = -1 & & \text{vertex } t \\
 & -x_1 & & & + & x_3 & + & x_4 & & & = 0 & & \text{vertex } u \\
 & & - & x_2 & - & x_3 & & & + & x_5 & = 0 & & \text{vertex } v \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & , & x_5 & = 0/1 & &
\end{array}
$$

SHORESTPATH problem



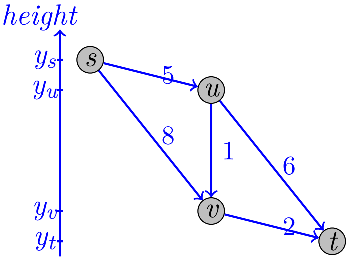
- PRIMAL problem: relax the 0/1 integer linear program into linear program by the **totally uni-modular** property.

$$
\begin{array}{lrcrcrcrcrll}
\min & 5x_1 & + & 8x_2 & + & 1x_3 & + & 6x_4 & + & 2x_5 & & \\
s.t. & x_1 & + & x_2 & & & & & & & = 1 & \text{vertex } s \\
 & & & & & & - & x_4 & - & x_5 & = -1 & \text{vertex } t \\
 & -x_1 & & & + & x_3 & + & x_4 & & & = 0 & \text{vertex } u \\
 & & - & x_2 & - & x_3 & & & + & x_5 & = 0 & \text{vertex } v \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & , & x_5 & \leq 1 & \\
 & x_1 & , & x_2 & , & x_3 & , & x_4 & , & x_5 & \geq 0 &
\end{array}
$$

- DUAL PROBLEM: set variables y_i for **cities** and variables z_i for the constraints $x_i \leq 1$.

$$
\begin{array}{rcccccccccc}
\max & y_s & -y_t & & & +z_1 & +z_2 & +z_3 & +z_4 & +z_5 & \\
s.t. & y_s & & -y_u & & +z_1 & & & & & \leq 5 \\
 & y_s & & & -y_v & & +z_2 & & & & \leq 8 \\
 & & & y_u & -y_v & & & +z_3 & & & \leq 1 \\
 & & -y_t & +y_u & & & & & +z_4 & & \leq 6 \\
 & & -y_t & & +y_v & & & & & +z_5 & \leq 2 \\
 & & & & & z_1, & z_2, & z_3, & z_4, & z_5 & \leq 0
\end{array}
$$

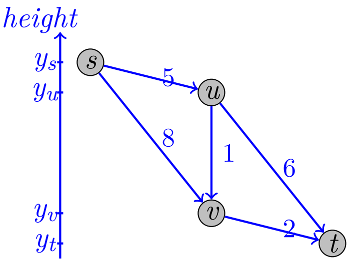
Dual of ShortestPath problem



- Dual problem: set variables for **cities**. (Intuition: y_i means the height of city i; thus, $y_s - y_t$ denotes the height difference between s and t, providing a lower bound of the shortest path length.)

$$
\begin{array}{rrrrrrrrrll}
\max & y_s & - & y_t & & & & & & & \\
s.t. & y_s & & & - & y_u & & & \leq 5 & & x_1 : \text{edge } (s, u) \\
 & y_s & & & & & - & y_v & \leq 8 & & x_2 : \text{edge } (s, v) \\
 & & & & & y_u & - & y_v & \leq 1 & & x_3 : \text{edge } (u, v) \\
 & & - & y_t & + & y_u & & & \leq 6 & & x_4 : \text{edge } (u, t) \\
 & & - & y_t & & & + & y_v & \leq 2 & & x_5 : \text{edge } (v, t)
\end{array}
$$

A simplified version



- Dual problem: simplify by setting $y_t = 0$ (and remove the 2nd constraint in the primal problem P, accordingly)

$$
\begin{array}{rllllll}
\max & y_s & & & & & \\
s.t. & y_s & - & y_u & & \leq 5 & x_1 : \text{edge } (s, u) \\
 & y_s & & & - \; y_v & \leq 8 & x_2 : \text{edge } (s, v) \\
 & & & y_u & - \; y_v & \leq 1 & x_3 : \text{edge } (u, v) \\
 & & & y_u & & \leq 6 & x_4 : \text{edge } (u, t) \\
 & & & & y_v & \leq 2 & x_5 : \text{edge } (v, t)
\end{array}
$$

- Dual feasible solution: $y^T = (0,0,0)$. Let's check the constraints in D:

$$\begin{array}{ccccccccl} y_s & & - & y_u & & & < & 5 & \Rightarrow x_1 = 0 \\ y_s & & & & - & y_v & < & 8 & \Rightarrow x_2 = 0 \\ & & & y_u & - & y_v & < & 1 & \Rightarrow x_3 = 0 \\ & & & y_u & & & < & 6 & \Rightarrow x_4 = 0 \\ & & & & & y_v & < & 2 & \Rightarrow x_5 = 0 \end{array}$$

- Identifying tight constraints in D: $J = \Phi$, implying that $x_1, x_2, x_3, x_4, x_5 = 0$.

- RP:

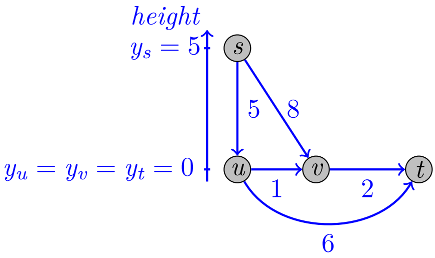
$$\begin{array}{rcccccccccl} \min & s_1 & +s_2 & +s_3 & & & & & & & \\ s.t. & s_1 & & & +x_1 & +x_2 & & & & = 1 & \text{node } s \\ & & s_2 & & -x_1 & & +x_3 & +x_4 & & = 0 & \text{node } u \\ & & & s_3 & & -x_2 & -x_3 & & +x_5 & = 0 & \text{node } v \\ & s_1, & s_2, & s_3, & & & & & & \geq 0 & \\ & & & & x_1, & x_2, & x_3, & x_4, & x_5 & = 0 & \end{array}$$

- DRP:

$$\begin{array}{lllll} \max & y_s & & & \\ s.t. & y_s & & & \leq 1 \\ & & y_u & & \leq 1 \\ & & & y_v & \leq 1 \end{array}$$

- Solve DRP using combinatorial technique: optimal solution $\Delta y^T = (1, 0, 0)$. *Note: the optimal solution is not unique*
- Step length θ: $\theta = \min\{\frac{c_1 - y^T a_1}{\Delta y^T a_1}, \frac{c_2 - y^T a_2}{\Delta y^T a_2}\} = \min\{5, 8\} = 5$
- Update y: $y^T = y^T + \theta \Delta y^T = (5, 0, 0)$.

Iteration 1 III



- From the point of view of Dijkstra's algorithm:
  - Optimal solution to DRP is $\Delta y^T = (1, 0, 0)$: the explored vertex set $S = \{s\}$ in Dijkstra's algorithm. In fact, DRP is solved via identifying the nodes reachable from s.
  - Step length $\theta = \min\{\frac{c_1 - y^T a_1}{\Delta y^T a_1}, \frac{c_2 - y^T a_2}{\Delta y^T a_2}\} = \min\{5, 8\} = 5$: finding the closest vertex to the nodes in S via comparing all edges going out from S.

Iteration 2 I

- Dual feasible solution: $y^T = (5, 0, 0)$. Let's check the constraints in D:

$$
\begin{array}{ccccccl}
y_s & - & y_u & & & = & 5 \\
y_s & & & - & y_v & < & 8 \quad \Rightarrow x_2 = 0 \\
& & y_u & - & y_v & < & 1 \quad \Rightarrow x_3 = 0 \\
& & y_u & & & < & 6 \quad \Rightarrow x_4 = 0 \\
& & & & y_v & < & 2 \quad \Rightarrow x_5 = 0
\end{array}
$$

- Identifying tight constraints in D: $J = \{1\}$, implying that $x_2, x_3, x_4, x_5 = 0$.

- RP:

$$
\begin{array}{lcccccccccl}
\min & s_1 & +s_2 & +s_3 & & & & & & & \\
s.t. & s_1 & & & +x_1 & +x_2 & & & & = 1 & \text{node } s \\
& & s_2 & & -x_1 & & +x_3 & +x_4 & & = 0 & \text{node } u \\
& & & s_3 & & -x_2 & -x_3 & & +x_5 & = 0 & \text{node } v \\
& s_1, & s_2, & s_3, & & & & & & \geq 0 & \\
& & & & & x_2, & x_3, & x_4, & x_5 & = 0 &
\end{array}
$$

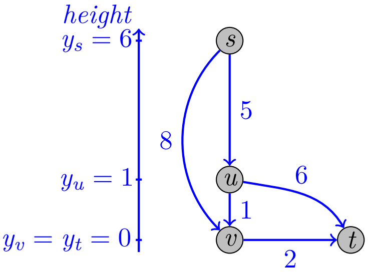
Iteration 2 II

- DRP:

$$\begin{array}{rccccccl} \max & y_s & & & & & & \\ s.t. & y_s & - & y_u & & & & \leq 0 \\ & y_s & , & y_u & , & y_v & & \leq 1 \end{array}$$

- Solve DRP using combinatorial technique: optimal solution $\Delta y^T = (1, 1, 0)$. *Note: the optimal solution is not unique*
- Step length θ: $\theta = \min\{\frac{c_2 - y^T a_2}{\Delta y^T a_2}, \frac{c_3 - y^T a_3}{\Delta y^T a_3}, \frac{c_4 - y^T a_4}{\Delta y^T a_4}\} = \min\{3, 1, 6\} = 1$
- Update y: $y^T = y^T + \theta \Delta y^T = (6, 1, 0)$.

Iteration 2 III



- From the point of view of Dijkstra's algorithm:
  - Optimal solution to DRP is $\Delta y^T = (1, 1, 0)$: the explored vertex set $S = \{s, u\}$ in Dijkstra's algorithm. In fact, DRP is solved via identifying the nodes reachable from s.
  - Step length $\theta = \min\{\frac{c_2 - y^T a_2}{\Delta y^T a_2}, \frac{c_3 - y^T a_3}{\Delta y^T a_3}, \frac{c_4 - y^T a_4}{\Delta y^T a_4}\} = \min\{3, 1, 6\} = 1$: finding the closest vertex to the nodes in S via comparing all edges going out from S.

Iteration 3 I

- Dual feasible solution: $y^T = (6, 1, 0)$. Let's check the constraints in D:

$$
\begin{array}{ccccccccc}
y_s & & - & y_u & & & = & 5 & \\
y_s & & & & - & y_v & < & 8 & \Rightarrow x_2 = 0 \\
& & & y_u & - & y_v & = & 1 & \\
& & & y_u & & & < & 6 & \Rightarrow x_4 = 0 \\
& & & & & y_v & < & 2 & \Rightarrow x_5 = 0
\end{array}
$$

- Identifying tight constraints in D: $J = \{1, 3\}$, implying that $x_2, x_4, x_5 = 0$.
- RP:

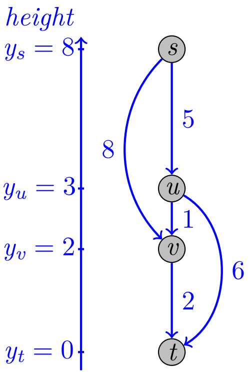
$$
\begin{array}{lccccccccll}
\min & s_1 & +s_2 & +s_3 & & & & & & & \\
s.t. & s_1 & & & +x_1 & +x_2 & & & & = 1 & \text{node } s \\
& & s_2 & & -x_1 & & +x_3 & +x_4 & & = 0 & \text{node } u \\
& & & s_3 & & -x_2 & -x_3 & & +x_5 & = 0 & \text{node } v \\
& s_1, & s_2, & s_3, & & & & & & \geq 0 & \\
& & & & & x_2, & & x_4, & x_5 & = 0 &
\end{array}
$$

- *DRP*:

$$\begin{array}{rccccccc} \max & y_s & & & & & & \\ s.t. & y_s & - & y_u & & & \leq 0 \\ & & & y_u & - & y_v & \leq 0 \\ & y_s & , & y_u & , & y_v & \leq 1 \end{array}$$

- Solve DRP using combinatorial technique: optimal solution $\Delta y^T = (1, 1, 1)$.
- Step length θ: $\theta = \min\{\frac{c_4 - y^T a_4}{\Delta y^T a_4}, \frac{c_5 - y^T a_5}{\Delta y^T a_5}\} = \min\{5, 2\} = 2$
- Update y: $y^T = y^T + \theta \Delta y^T = (8, 3, 2)$.

Iteration 3 III



- From the point of view of Dijkstra's algorithm:
  - Optimal solution to DRP is $\Delta y^T = (1, 1, 1)$: the explored vertex set $S = \{s, u, v\}$ in Dijkstra's algorithm. In fact, DRP is solved via identifying the nodes reachable from s.

- Step length $\theta = \min\{\frac{c_4 - y^T a_4}{\Delta y^T a_4}, \frac{c_5 - y^T a_5}{\Delta y^T a_5}\} = \min\{5, 2\} = 2$: finding the closest vertex to the nodes in S via comparing all edges going out from S.

Iteration 4 I

- Dual feasible solution: $y^T = (8, 3, 2)$. Let's check the constraints in D:

$$\begin{array}{ccccccl}
y_s & - & y_u & & & = & 5 \\
y_s & & & - & y_v & < & 8 \quad \Rightarrow x_2 = 0 \\
& & y_u & - & y_v & = & 1 \\
& & y_u & & & < & 6 \quad \Rightarrow x_4 = 0 \\
& & & & y_v & = & 2
\end{array}$$

- Identifying tight constraints in D: $J = \{1, 3, 5\}$, implying that $x_2, x_4 = 0$.
- RP:

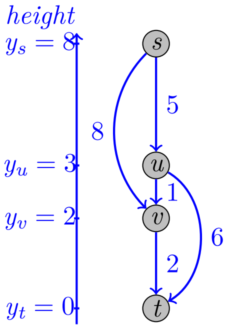
$$\begin{array}{lccccccccl}
\min & s_1 & +s_2 & +s_3 & & & & & & \\
s.t. & s_1 & & & +x_1 & +x_2 & & & & = 1 \quad \text{node } s \\
& & s_2 & & -x_1 & & +x_3 & +x_4 & & = 0 \quad \text{node } u \\
& & & s_3 & & -x_2 & -x_3 & & +x_5 & = 0 \quad \text{node } v \\
& s_1, & s_2, & s_3, & & & & & & \geq 0 \\
& & & & & x_2, & & x_4 & & = 0
\end{array}$$

Iteration 4 II

- *DRP*:

$$\begin{array}{rccccccl}
\max & y_s & & & & & & \\
s.t. & y_s & - & y_u & & & & \leq 0 \\
 & & & y_u & - & y_v & & \leq 0 \\
 & & & & & y_v & & \leq 0 \\
 & y_s & , & y_u & , & y_v & & \leq 1
\end{array}$$

- Solve *DRP* using combinatorial technique: optimal solution $\Delta y^T = (0, 0, 0)$. Done!



- From the point of view of Dijkstra's algorithm:
  - Optimal solution to DRP is $\Delta y^T = (0, 0, 0)$: there is a path from s to t, forcing $y_s = 0$ (note y_t is fixed to be 0). This corresponds to the explored node set $S = \{s, u, v, t\}$ in Dijkstra's algorithm.
- Another intuitive explanation: the **tightest** rope when picking up s.

Application 1: A succinct proof of Farkas lemma [1894]

Theorem (Farkas lemma)

Given vectors $a_1, a_2, ..., a_m, c \in \mathbb{R}^n$. Then either

1. $c \in C(a_1, a_2, ..., a_m)$; *or*
2. *there is a vector $y \in \mathbb{R}^n$ such that for all i, $y^T a_i \geq 0$ but $y^T c < 0$.*

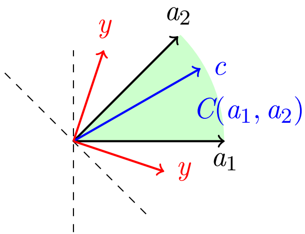


Figure: Case 1: $c \in C(a_1, a_2)$

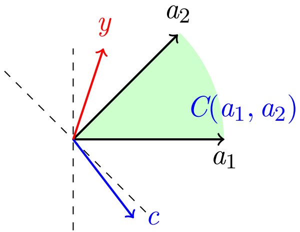


Figure: Case 2: $c \notin C(a_1, a_2)$

- Here, $C(a_1, ..., a_m)$ denotes the cone spanned by $a_1, ..., a_m$, i.e. $C(a_1, ..., a_m) = \{x | x = \sum_{i=1}^{m} \lambda_i a_i, \lambda_i \geq 0\}$.

Proof.

- Suppose for any vector $y \in \mathbb{R}^n$, $y^T a_i \geq 0$ $(i = 1, 2, ..., m)$, we always have $y^T c \geq 0$. We will show that c should lie within the cone $C(a_1, a_2, ..., a_m)$.
- Consider the following PRIMAL problem:

$$\begin{array}{rrcll} \min & c^T y & & & \\ s.t. & a_i^T y & \geq & 0 & i = 1, 2, ..., m \end{array}$$

- It is obvious that the PRIMAL problem has a feasible solution $y = 0$, and is bounded since $c^T y \geq 0$.
- Thus the DUAL problem also has a bounded optimal solution:

$$\begin{array}{rrcl} \max & 0 & & \\ s.t. & x^T A^T & = & c^T \\ & x & \geq & 0 \end{array}$$

- In other words, there exists a vector x such that $c = \sum_{i=1}^{m} x_i a_i$ and $x_i \geq 0$.

Variants of Farkas' lemma

Farkas' lemma lies at the core of linear optimization. Using Farkas' lemma, we can prove SEPARATION theorem, and MINIMAX theorem in the game theory.

Theorem

Let A be an $m \times n$ matrix, and $b \in \mathbb{R}^m$. Then either

1. $Ax = b$, $x \geq 0$ *has a feasible solution; or*
2. *there is a vector* $y \in \mathbb{R}^m$ *such that* $y^T A \geq 0$ *but* $y^T b < 0$.

Theorem

Let A *be an* $m \times n$ *matrix, and* $b \in \mathbb{R}^m$. *Then either*

1. $Ax \le b$ *has a feasible solution; or*
2. *there is a vector* $y \in \mathbb{R}^m$ *such that* $y \ge 0$, $y^T A \ge 0$ *but* $y^T b < 0$.

Caratheodory's theorem

Theorem

Given vectors $a_1, a_2, ..., a_m \in \mathbb{R}^n$. *If* $x \in C(a_1, a_2, ..., a_m)$, *then there is a linearly independent vector set of* $a_1, a_2, ..., a_m$, *say* $a_1, a_2, ..., a_r$, *such that* $x \in C(a_1, a_2, ..., a_r)$.

Theorem

Let $C \subseteq \mathbb{R}^n$ be a closed, convex set, and let $x \in \mathbb{R}^n$. If $x \notin C$, then there exists a hyperplane separating x from C.

Application 2: von Neumann's MiniMax theorem on game theory

Game theory

- Game theory studies competing and cooperative behaviours among intelligent and rational decision-makers.
- In 1928, John von Neumann proved the existence of mixed-strategy equilibria in two-person zero-sum games.
- In 1950, John Forbes Nash Jr. developed a criterion of mutual consistency of players' strategies, which applies to a wider range of games than that proposed by J. von Neumann. He proved the existence of Nash equilibrium in every n-player, non-zero-sum, non-cooperative game (not just 2-player, zero-sum games).
- Game theory was widely applied in mathematical economics, in biology (e.g., analysis of evolution and stability) and computer science (e.g., analysis of interactive computations and lower bound on the complexity of randomized algorithms, the equivalence between linear program and two-person zero-sum game).

Paper-rock-scissors: an example of two-player zero-sum game

- Paper-rock-scissors is a hand game usually played by two players, denoted as row player and column player: each player selects one of the three hand shapes, including "paper", "rock", and "scissors"; then the players show their selections simultaneously.
- It has two possible outcomes other than tie: one player wins and the other player loses, which can be formally described using the following payoff matrix.

	Paper	Rock	Scissors
Paper	0, 0	1, -1	-1, 1
Rock	-1, 1	0, 0	1, -1
Scissors	1, -1	-1, 1	0, 0

- Each player attempts to select appropriate action to maximize his gain.

Matching penny: another example of two-person zero-sum game

- Matching pennies is a game played by two players, namely, row player and column player. Each player has a penny and secretly turns it to head or tail. The players then reveal their selections simultaneously.
- If the pennies match, then row player keeps both pennies; otherwise, column player keeps both. The payoff matrix is as follows.

	Head	Tail
Head	1, -1	-1, 1
Tail	-1, 1	1, -1

- Each player tries to maximize his gain via making an appropriate selection.

Simultaneous games vs. sequential games

- **Simultaneous games** are games in which all players move **simultaneously**. Thus, no player have information of the others' selections in advance.
- **Sequential games** are games in which the later player has some information, although maybe imperfect, of previous actions by the other players. A complete plan of action for every stage of the game, regardless of whether the action actually arises in play, is denoted as **a (pure) strategy**.
- **Normal form** is used to describe simultaneous games while **extensive form** is used to describe sequential games.
- J. von Neumann proposed an approach to **transform strategies in sequential games into actions in simultaneous games**.
- Note that the transformation is one-way, i.e., multiple sequential games might correspond to the same simultaneous game, and it may result in an exponential blowup in the size of the representation.

Normal form

- A game Γ in normal form among m players contains the following items:
    - Each player k has a finite number of **pure strategies** $S_k = \{1, ..., n_k\}$.
    - Each player k is associated with a **payoff function** $H_k : S_1 \times S_2 \times ... \times S_m \to \mathbb{R}$.
- To play the game, each player selects a strategy without information of others, and then reveals the selection **simultaneously**. The players' gain are calculated using corresponding payoff functions.
- Each player attempts to maximize his gain via selecting an appropriate strategy.

Two-person zero-sum game in normal form

- In a two-person zero-sum game game Γ, a player's gain or less is exactly balanced by the other player's loss or gain, i.e.,

$$H_1(s_1, s_2) + H_2(s_1, s_2) = 0.$$

- Thus we can define another function

$$H(s_1, s_2) = H_1(s_1, s_2) = -H_2(s_1, s_2)$$

and represent it using a **payoff matrix**.

	Head	Tail
Head	1	-1
Tail	-1	1

- Row player aims to maximize $H(s_1, s_2)$ by selecting an appropriate strategy s_1 while column player aims to minimize $H(s_1, s_2)$ by selecting an appropriate strategy s_2.

von Neumann's MiniMax theorem: motivation

- When analyzing a two-person zero-sum game Γ, von Neumann noticed that the difficulty comes from the difference between games and ordinary optimization problems: **row player tries to maximize $H(s_1, s_2)$; however, he can control s_1 only as he has no information of the other player's selection s_2**, and so does column player.
- Thus von Neumann suggested to **investigate two auxiliary games without this difficulty**, denoted as Γ_1 and Γ_2, before attacking the challenging game Γ.
  1. Γ_1: Row player selects a strategy s_1 first, and exposes his selection to column player before column player selects a strategy s_2.
  2. Γ_2: Column player selects a strategy s_2 first, and exposes his selection to row player before row player selects a strategy s_1.
- The two auxiliary games are much easier than the original game Γ, and more importantly, they provide **upper and lower bounds for Γ**.

Auxiliary game Γ_1

- Let's consider column player first. As he knows row player's selection s_1, the objective function $H(s_1, s_2)$ becomes an ordinary optimization function over a single variable s_2, and column player can simply select a strategy s_2 with the minimum objective function value $\min_{s_2} H(s_1, s_2)$.

	Head	Tail	Row minimum
Head	-2	1	-2
Tail	-1	2	$v_1 = -1$

- Now consider row player. When he selects a strategy s_1, he can definitely predict the selection of column player. Since $\min_{s_2} H(s_1, s_2)$ is an ordinary function over a single s_1, it is easy for row player to select a strategy s_1 with the maximum objective function value

$$v_1 = \max_{s_1} \min_{s_2} H(s_1, s_2).$$

Auxiliary game Γ_2

- Let's consider row player first. As he knows column player's selection s_2, the objective function $H(s_1, s_2)$ becomes an ordinary optimization function over a single variable s_1, and row player can simply select a strategy s_1 with the maximum objective function value $\max_{s_1} H(s_1, s_2)$.

	Head	Tail
Head	-2	1
Tail	-1	2
Column maximum	$v_2 = -1$	2

- Now consider column player. When he selects a strategy s_2, he can definitely predict the selection of row player. Since $\max_{s_1} H(s_1, s_2)$ is an ordinary function over a single variable s_2, it is easy for column player to select a strategy s_2 with the minimum objective function value

$$v_2 = \min_{s_2} \max_{s_1} H(s_1, s_2).$$

Γ_1 and Γ_2 bound Γ

- For row player, it is clearly Γ_1 is disadvantageous to him as he should expose his selection s_1 to column player.
- On the contrary, Γ_2 is beneficial to row player as he knows column player's selection s_2 before making decision.

	Head	Tail	Row minimum
Head	-2	1	-2
Tail	-1	2	$v_1 = -1$
Column maximum	$v_2 = -1$	2	

- Thus these two auxiliary games provides lower and upper bounds:

$$v_1 \leq v \leq v_2$$

where v denote row player's gain in the original game Γ.

Case 1: $v_1 = v_2$

- For a game with the following payoff matrix, we have $v_1 = v = v_2$ and call this game **strictly determined**.

	Head	Tail	Row minimum
Head	-2	1	-2
Tail	-1	2	$v_1 = -1$
Column maximum	$v_2 = -1$	2	

- The saddle point of the payoff matrix $H(s_1, s_2)$ represents a **pure strategy equilibrium**. In this equilibrium, each player has nothing to gain by changing only his own strategy. In addition, **knowing the opponent's selection will bring no gain**.
- von Neumann proved the existence of the optimal strategy in a perfect information two-person zero-sum game, e.g., chess. L. S. Shapley further showed that a finite two-person zero-sum game has a pure strategy equilibrium if every 2×2 submatrix of the game has a pure strategy equilibrium [?].

Case 2: $v_1 < v_2$

- In contrast, matching penny does not have a pure strategy equilibrium as there is no saddle point in the payoff matrix. So does the paper-rock-scissors game.

	Head	Tail	Row minimum
Head	1	-1	-1
Tail	-1	1	$v_1 = -1$
Column maximum	$v_2 = 1$	1	

- This fact implies that knowing the opponent's selection might bring gain; however, it is impossible to know the opponent's selection as the players reveal their selections simultaneously. In this case, let's play a mixed strategy rather than a pure strategy.

From pure strategy to mixed strategy

- A **mixed strategy** is an assignment of probability to pure strategies, allowing a player to **randomly select a pure strategy**.
- Consider the payoff matrix as below. If the row player select strategy A with probability 1, he is said to play a pure strategy. If he tosses a coin and select strategy A if the coin lands head and B otherwise, then he is said to play a mixed strategy.

	A	B
A	1	-1
B	-1	1

Two types of interpretation of mixed strategy

- From a player's viewpoint: J. von Neumann described the motivation underlying the introduction of mixed strategy as follows: since it is impossible to exactly know opponent's selection, a player could switch to **protect himself by "randomly selecting his own strategy"**, making it difficult for the opponent to know the player's selection. However, this interpretation came under heavy fire for lacking of behaviour supports: Seldom do people make choices following a lottery.
- From opponent's viewpoint: Robert Aumann and Adam Brandenburger interpreted mixed strategy of a player as **opponent's "belief" of the player's selection**. Thus, Nash equilibrium is an equilibrium of "belief" rather than actions.

Existence of mixed strategy equilibrium

- Consider a mixed strategy game: row player has m strategies available and he selects a strategy s_1 according to a distribution u, while column player has n strategies available and he selects a strategy s_2 according to a distribution v, i.e.,

$$\Pr(s_1 = i) = u_i, i = 1, ..., n \quad \Pr(s_2 = j) = v_j, j = 1, ..., m$$

  Here, u and v are independent.
- Thus the expected gain of row player is:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} u_i H_{ij} v_j = u^T H v$$

- row player attempts to minimize $u^T H v$ via selecting an appropriate u, while column player attempts to maximize it via selecting an appropriate v.
- Now let's consider the two auxiliary games Γ_1 and Γ_2 again and answer the following questions: what happens if row player exposes his mixed strategy to column player? And if we reverse the order of the players?

- This question has been answered by the von Neumann's MiniMax theorem.

Theorem

$$\max_u \min_v u^T H v = \min_v \max_u u^T H v$$

- The theorem states that **knowing the other player's strategy will bring no gain in a mixed-strategy zero-sum game**, and the order doesn't change the value.
- A **mixed-strategy Nash equilibrium** exists for **any two-person zero-sum game** with a finite set of actions. A Nash equilibrium in a two-player game is a pair of strategies, each of which is a best response to the other.

von Neumann's MiniMax theorem: proof

- Let's consider the auxiliary game Γ_1 first, in which the strategy of row player, i.e., u, was exposed to column player. This is of course beneficial to column player since he can select the optimal strategy v to minimize $u^T Hv$, which is

$$\inf\{u^T Hv | v \geq 0, \mathbf{1}^T v = 1\} = \min_{j=1,\ldots,n} (u^T H)_j$$

- Thus row player should select u to maximize the above value, which can be formulated as a linear program:

$$\begin{array}{llcl} \max & \min_{j=1,\ldots,n} (u^T H)_j & & \\ s.t. & \mathbf{1}^T u & = & 1 \\ & u & \geq & 0 \end{array}$$

von Neumann's MINIMAX theorem: proof

- The linear program can be rewritten as below.

$$\begin{array}{rlcl} \max & s & & \\ s.t. & u^T H & \geq & s1^T \\ & 1^T u & = & 1 \\ & u & \geq & 0 \end{array}$$

- Similarly we consider the auxiliary game Γ_2 and calculate the optimal strategy v by solving the following linear program.

$$\begin{array}{rlcl} \min & t & & \\ s.t. & Hv & \leq & t1 \\ & 1^T v & = & 1 \\ & v & \geq & 0 \end{array}$$

- These two linear programs are both feasible and form Lagrangian dual. Thus they have the same optimal objective value according to the strong duality property.

An example: paper-rock-scissors game

- For the paper-rock-scissors game, we have the following two linear programs.
  - Linear program for Γ_1:

$$
\begin{array}{rrcrcrl}
\max & s & & & & & \\
s.t. & 0u_1 & - & u_2 & + & u_3 & \geq s \\
 & u_1 & + & 0u_2 & - & u_3 & \geq s \\
 & -u_1 & + & u_2 & + & 0u_3 & \geq s \\
 & u_1 & + & u_2 & + & u_3 & = 1 \\
 & u_1, & & u_2, & & u_3 & \geq 0
\end{array}
$$

  - Linear program for Γ_2:

$$
\begin{array}{rrcrcrl}
\min & t & & & & & \\
s.t. & 0v_1 & + & v_2 & - & v_3 & \leq t \\
 & -v_1 & + & 0v_2 & + & v_3 & \leq t \\
 & v_1 & - & v_2 & + & 0v_3 & \leq t \\
 & v_1 & + & v_2 & + & v_3 & = 1 \\
 & v_1, & & v_2, & & v_3 & \geq 0
\end{array}
$$

- The mixed strategy equilibrium is $u^T = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$ and $u^T = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$ with the game value 0.

Comments on the mixed strategy equilibrium by von Neumann

- Note that a mixed strategy equilibrium always exists no matter whether the payoff matrix H has a saddle point or not.
- Regardless of column player's selection, row player can select an appropriate strategy to guarantee his gain $v_1 \geq 0$.
- Regardless of row player's selection, column player can select an appropriate strategy to guarantee row player's gain $v_1 \leq 0$.
- Using the strategy $u^T = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$, row player can guarantee that he "won't lose", i.e., the probability of losing is less than the probability of winning.
- The strategy $u^T = [\frac{1}{3}, \frac{1}{3}, \frac{1}{3}]$ is designed for "protecting himself" rather than "attacking his opponent", i.e., it cannot be used to benefit from opponent's fault.

Application 3: Yao's MiniMax principle [1977]

Yao's MINIMAX principle

- Consider a problem Π. Let $\mathcal{A} = \{A_1, A_2, ..., A_n\}$ be algorithms to Π, and $\mathcal{I} = \{I_1, I_2, ..., I_m\}$ be the inputs with a given size. Let $T(A_i, I_j)$ be the running time of algorithm A_i on the input I_j.

		Algorithms	
		A_1	A_2
Inputs	I_1	T_{11}	T_{12}
	I_2	T_{21}	T_{22}

- Thus $\max\limits_{I_j \in \mathcal{I}} T(A_i, I_j)$ represents the worst-case time for the deterministic algorithm A_i.
- For a randomized algorithms, however, it is usually difficult to bound its expected running time on worst-case inputs.
- Yao's MINIMAX principle provides a technique to build lower bound for **the expected running time of any randomized algorithm on its worst-case input**.

Expected running time of a randomized algorithm A_q

- A "Las Vegas" randomized algorithm can be viewed as a distribution over all deterministic algorithms $\mathcal{A} = \{A_1, A_2, ..., A_n\}$.
- Specifically, let q be a distribution over $\mathcal{A}$, and A_q be a randomized algorithm chosen according to q, i.e., A_q refers to a deterministic algorithm A_i with probability q_i.
- Given a input I_j, the expected running time of A_q can be written as

$$E[T(A_q, I_j)] = \sum_{i=1}^{n} q_i T(A_i, I_j)$$

- Thus $\max_{I_j \in \mathcal{I}} E[T(A_q, I_j)]$ represents the expected running time of A_q on its worst-case input.

Expected running time of a deterministic algorithm A_i on random input

- Now consider a deterministic algorithm A_i running on random input.
- Let p be a distribution over $\mathcal{I}$, and I_p be a random input chosen from $\mathcal{I}$, i.e., I_p refers to I_j with probability p_j.
- Given a deterministic algorithm A_i, its expected running time on random input I_p can be written as

$$E[T(A_i, I_p)] = \sum_{j=1}^{m} p_j T(A_i, I_j)$$

- Thus $\min_{A_i \in \mathcal{A}} E[T(A_i, I_p)]$ represents the expected running time of the best deterministic algorithm on the random input I_p.

Yao's MiniMax principle

Theorem

For any random input I_p and randomized algorithm A_q,

$$\min_{A_i \in \mathcal{A}} E[T(A_i, I_p)] \leq \max_{I_j \in \mathcal{I}} E[T(A_q, I_j)]$$

- To establish a lower bound for the expected running time of a randomized algorithm on its worst-case input, it suffices to find an appropriate distribution over inputs and prove that on this random input, no deterministic algorithm can do better than the randomized one.
- The power of this technique lies at the fact that one can choose any distribution over inputs and the lower bound is constructed based on deterministic algorithms.

Yao's MiniMax principle: proof

Proof.

$$
\begin{aligned}
\min_{A_i \in \mathcal{A}} E[T(A_i, I_p)] &\leq \max_{u \in \Delta_m} \min_{A_i \in \mathcal{A}} E[T(A_i, I_u)] && (5)\\
&= \max_{u \in \Delta_m} \min_{v \in \Delta_n} E[T(A_v, I_u)] && (6)\\
&= \min_{v \in \Delta_n} \max_{u \in \Delta_m} E[T(A_v, I_u)] && (7)\\
&= \min_{v \in \Delta_n} \max_{I_j \in \mathcal{I}} E[T(A_v, I_j)] && (8)\\
&\leq \max_{I_j \in \mathcal{I}} E[T(A_q, I_j)] && (9)
\end{aligned}
$$

$\square$

- Here, Δ_n denotes the set of n-dimensional probability vectors.
- Equation (3) follows by the von Neumann's MiniMax theorem.

Appendix: Slater theorem

Strong duality and Slater's condition [Morton Slater, 1950]

- Slater's condition is a sufficient condition for strong duality to hold for a convex optimization problem.
- Consider a convex optimization problem.

$$\begin{array}{rlcll} \min & f_0(x) & & & \\ s.t. & f_i(x) & \leq & 0 & i = 1, ..., m \\ & Ax & = & b & \end{array}$$

  where $f_i(x)$ $(i = 0, 1, ..., m)$ are convex.
- We use $\mathcal{D} = \bigcap_{i=0}^{m} \mathbf{dom}\ f_i$ to represent the domain of definition. We use p^* to represent the optimal value of the problem and d^* the optimal value of the dual problem.
- For the sake simplicity, we assume that p^* is finite and the interior of $\mathcal{D}$ is non-empty, i.e. $\mathbf{relint}\ \mathcal{D} = \mathbf{int}\ \mathcal{D}$.

Slater's conditions for general convex problem

- Slater's condition: There exists a vector (called Slater vector) $\tilde{x} \in \textbf{relint } \mathcal{D}$ such that the **inequality constraints strictly hold**:

$$f_i(\tilde{x}) < 0, i = 1, ..., m, \quad A\tilde{x} = b$$

Theorem

If Slater's condition holds, then there is no duality gap, i.e. $p^* = d^*$, and the set of dual optimal solutions is non-empty and bounded.

Slater's condition for convex program over linear constraints

- **Affine constraints are not required to hold strictly**, i.e., suppose the first k constraints are affine, the Slater's condition becomes: There exists a Slater vector $\tilde{x} \in \textbf{relint}\mathcal{D}$ such that

$$f_i(\tilde{x}) \leq 0, i = 1, ..., k, \quad f_i(\tilde{x}) < 0, i = k+1, ..., m, \quad A\tilde{x} = b$$

- A special case is convex program over linear constraints.

$$\begin{array}{rl} \min & f_0(x) \\ s.t. & Ax \leq b \end{array}$$

where $f_0(x)$ is convex. For such problems, Slater's condition simply reduces to: there exists a Slater vector $\tilde{x} \in \textbf{relint dom} f_0$ such that $A\tilde{x} \leq b$.

Example 1: Least square problem

- Primal problem:

$$\begin{array}{rrcl} \min & x^T x & & \\ s.t. & Ax & = & b \end{array}$$

- Dual problem:

$$\max -\frac{1}{4} y^T A A^T y - b^T y$$

- Strong duality always holds if the primal problem is feasible, i.e., there exists a $\tilde{x}$ such that $A\tilde{x} = b$.

Example 2: Linear program

- Primal problem:

$$\begin{array}{rrcl} \min & c^T x & & \\ s.t. & Ax & \geq & b \\ & x & \geq & 0 \end{array}$$

- Dual problem:

$$\begin{array}{rrcl} \max & b^T y & & \\ s.t. & y & \geq & 0 \\ & A^T y & \leq & c \end{array}$$

- For linear programs, strong duality always holds if the primal problem is feasible. Similarly, strong duality always holds if the dual problem is feasible.
- The only case where strong duality fails is that neither primal nor dual problems is feasible.

Example 3: Quadratic objective over quadratic constraints I

- Primal problem:

$$\begin{array}{lll} \min & \frac{1}{2}x^T P_0 x + q_0^T x + r_0 & \\ s.t. & \frac{1}{2}x^T P_i x + q_i^T x + r_i & \leq \quad 0 \quad i = 1, ..., m \end{array}$$

where P_0 is positive definite, and P_i $(i = 1, .., m)$ are semi-positive definite.

- Lagrangian function:

$$L(x, \lambda) = \frac{1}{2}x^T P(\lambda)x + q(\lambda)^T x + r(\lambda)$$

where

$$P(\lambda) = P_0 - \sum_{i=1}^{m} \lambda_i P_i, \; q(\lambda) = q_0 - \sum_{i=1}^{m} \lambda_i q_i, \; r(\lambda) = r_0 - \sum_{i=1}^{m} \lambda_i r_i$$

.

Example 3: Quadratic objective over quadratic constraints II

- Lagrangian dual function:

$$g(\lambda) = \inf_x L(x, \lambda) = -\frac{1}{2} q(\lambda)^T P(\lambda)^{-1} q(\lambda) + r(\lambda)$$

as $P(\lambda)$ is positive definite when $\lambda \leq 0$.

- Dual problem:

$$\begin{array}{rl} \max & -\frac{1}{2} q(\lambda)^T P(\lambda)^{-1} q(\lambda) + r(\lambda) \\ s.t. & \lambda \leq 0 \end{array}$$

- According to Slater's condition, strong duality holds when quadratic constraints strictly hold, i.e., there exists a Slater vector x such that $\frac{1}{2} x^T P_i x + q_i^T x + r_i < 0$ $(i = 1, ..., m)$.

An example not satisfying the Slater's condition

- Primal problem:

$$\begin{array}{lrcl} \min & e^{-\sqrt{x_1}\sqrt{x_2}} & & \\ s.t. & x_1 & \leq & 0 \end{array}$$

- Here $\mathcal{D} = \{x \mid x_1 \geq 0, x_2 \geq 0\}$ and $\mathbf{relint}\ \mathcal{D} = \{x \mid x_1 > 0, x_2 > 0\}$.
- For this problem, Slater's condition fails as there is no vector $\tilde{x} \in \mathbf{relint}\ \mathcal{D}$ such that $x_1 \leq 0$. In fact, there is a duality gap.

Basic idea of the proof

- We will use a simple problem with a single constraint to explain the basic idea of the proof.

$$\begin{array}{rll} \min & f_0(x) & \\ s.t. & f_1(x) & \leq & 0 \end{array}$$

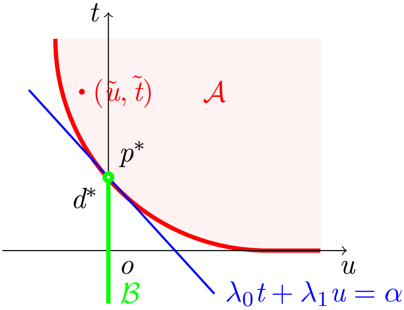
- Lagrangian dual function:

$$g(\lambda) = \inf_{x \in \mathcal{D}} (f_0(x) + \lambda f_1(x))$$

- To prove the theorem, it suffices to prove that when Slater's condition holds, $d^* = \max_{\lambda \geq 0} g(\lambda) \geq p^*$, i.e. there exists a λ ($\lambda \geq 0$) such that

$$g(\lambda) = \inf_{x \in \mathcal{D}} (f_0(x) + \lambda f_1(x)) \geq p^*.$$

- Basic idea: The Slater's condition states the existence of a $\tilde{x} \in \mathcal{D}$ such that $f_1(\tilde{x}) < 0$. Let's denote $\tilde{u} = f_1(\tilde{x})$ and $\tilde{t} = f_0(\tilde{x})$. The two points, namely, $(0, p^*)$ and $(\tilde{u}, \tilde{t})$, such that $\tilde{u} < 0$ but $\tilde{t} \geq p^*$ guarantees the existence of a $\lambda \geq 0$ such that for any $x \in \mathcal{D}$, $f_0(x) + \lambda f_1(x) \geq p^*$.



- To find a $\lambda \geq 0$ such that for any $x \in \mathcal{D}$, $f_0(x) + \lambda f_1(x) \geq p^*$, we first collect all possible values of $f_0(x)$ and $f_1(x)$

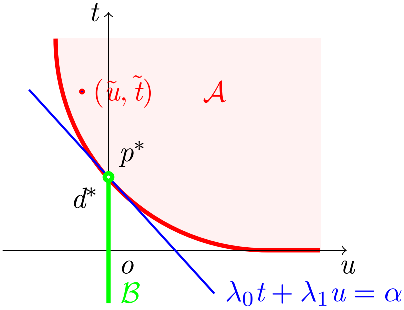
$$\mathcal{G} = \{(f_0(x), f_1(x)) \mid x \in \mathcal{D}\}$$

and then construct its epigraph set

$$\mathcal{A} = \{(u, t) \mid \exists\ x \in \mathcal{D}, f_1(x) \leq u, f_0(x) \leq t\}.$$

- Next we define another set

$$\mathcal{B} = \{(0, t) \mid t < q^*\}.$$



- We claim that both $\mathcal{A}$ and $\mathcal{B}$ are convex and they are disjoint. By *Separating Hyperplane Theorem*, there exists a hyperplane $\lambda_0 t + \lambda_1 u = \alpha$ such that for any $(u, t) \in \mathcal{A}$

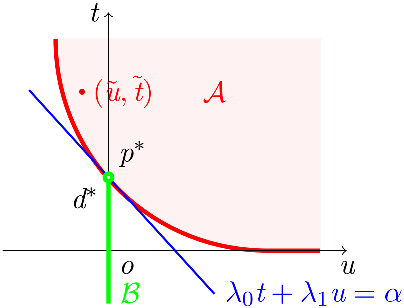
$$\lambda_0 t + \lambda_1 u \geq \lambda_0 p^*$$

- We can further claim that $\lambda_0 > 0$ and $\lambda_1 \geq 0$. By setting $\lambda = \frac{\lambda_1}{\lambda_0} \geq 0$, we finally prove that for any $(u, t) \in \mathcal{A}$

$$t + \lambda u \geq p^*, \text{ and thus for any } x \in \mathcal{D}$$

$$f_0(x) + \lambda f_1(x) \geq p^*.$$

Proof of claim 1: $\mathcal{A}$ and $\mathcal{B}$ are convex and disjoint

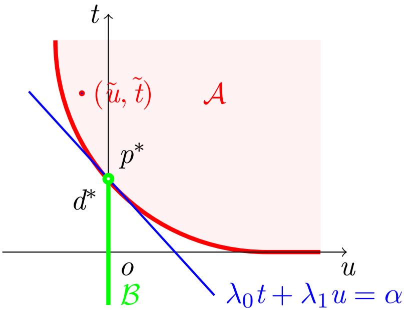


- By the convexity of $f_0(x)$ and $f_1(x)$, it is clear $\mathcal{A}$ is convex.
- Suppose there exists a point $(u, t) \in \mathcal{A} \cap \mathcal{B}$.
- $(u, t) \in \mathcal{B}$ implies $u = 0$, $t < p^*$ while $(u, t) \in \mathcal{A}$ implies the existence of a $x \in \mathcal{D}$ such that

$$f_0(x) \leq t < p^*$$

which contradicts the optimality of p^*.

Proof of claim 2: $\lambda_0 t + \lambda_1 u \geq \lambda_0 p^*$ for any $(u, t) \in \mathcal{A}$



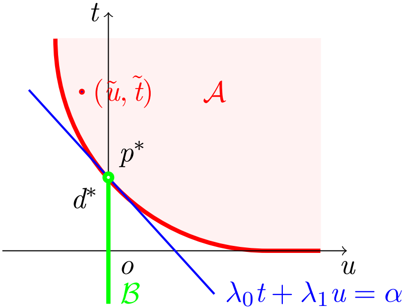
- By *Separating Hyperplane Theorem*, we have

$$\lambda_0 t + \lambda_1 u \geq \alpha \qquad \text{for any } (u, t) \in \mathcal{A}$$

$$\lambda_0 t + \lambda_1 u \leq \alpha \qquad \text{for any } (u, t) \in \mathcal{B}$$

- Note that $\lambda_1 \geq 0$ and $\lambda_0 \geq 0$ (Otherwise, $\lambda_1 t + \lambda_0 u$ will not have a lower bound over $\mathcal{A}$.)
- For any $(0, t) \in \mathcal{B}$, $\lambda_0 t \leq \alpha$ holds, implying $\lambda_0 p^* \leq \alpha$. Therefore, we have $\lambda_0 t + \lambda_1 u \geq \alpha \geq \lambda_0 p^*$ for any $(u, t) \in \mathcal{A}$.

Proof of claim 3: $\lambda_0 > 0$



- Assuming $\lambda_0 = 0$, the inequality $\lambda_0 t + \lambda_1 u \geq \lambda_0 p^*$ reduces into $\lambda_1 u \geq 0$ for any $(u, t) \in \mathcal{A}$, impling that for any $x \in \mathcal{D}$

$$\lambda f_1(x) \geq 0.$$

- This contradicts with the existence of a Slater's vector $\tilde{x} \in \textbf{relint } \mathcal{D}$ such that $f_1(x) < 0$.

Appendix: Finding initial solution to dual problem

Finding initial solution to dual problem

- Consider a primal problem P:

$$\begin{array}{rccccccccc} \min & c_1x_1 & + & c_2x_2 & + & \ldots & + & c_nx_n & & \\ & a_{11}x_1 & + & a_{12}x_2 & + & \ldots & + & a_{1n}x_n & = & b_1 \\ & a_{21}x_1 & + & a_{22}x_2 & + & \ldots & + & a_{2n}x_n & = & b_2 \\ & & & & & \ldots & & & & \\ & a_{m1}x_1 & + & a_{m2}x_2 & + & \ldots & + & a_{mn}x_n & = & b_m \\ & & & & & & & x_i & \geq & 0 \quad \text{for each } i \end{array}$$

  and its dual problem D:

$$\begin{array}{rccccccccc} \max & b_1y_1 & + & b_2y_2 & + & \ldots & + & b_my_m & & \\ & a_{11}y_1 & + & a_{21}y_2 & + & \ldots & + & a_{m1}y_m & \leq & c_1 \\ & a_{12}y_1 & + & a_{22}y_2 & + & \ldots & + & a_{m2}y_m & \leq & c_2 \\ & & & & & \ldots & & & & \\ & a_{1n}y_1 & + & a_{2n}y_2 & + & \ldots & + & a_{mn}y_m & \leq & c_n \end{array}$$

- If $c_i \geq 0$, it is easy to set initial dual solution $y_j = 0$ $j = 1, ..., m$. In general, however, it is not easy to obtain an initial dual solution directly.

Solving modified primal [Beale, 1954; Dantzig, 1956]

- Let's modify P by adding an extra constraint:

$$
\begin{array}{rcccccccccccc}
\min & & & c_1x_1 & + & c_2x_2 & + & \dots & + & c_nx_n & & \\
s.t. & x_0 & + & x_1 & + & x_2 & + & \dots & + & x_n & \leq & b_0 \\
 & & & a_{11}x_1 & + & a_{12}x_2 & + & \dots & + & a_{1n}x_n & = & b_1 \\
 & & & a_{21}x_1 & + & a_{22}x_2 & + & \dots & + & a_{2n}x_n & = & b_2 \\
 & & & & & & & \dots & & & & \\
 & & & a_{m1}x_1 & + & a_{m2}x_2 & + & \dots & + & a_{mn}x_n & = & b_m \\
 & x_0, & & x_1, & & x_2, & & \dots & & x_n & \geq & 0
\end{array}
$$

and consider the corresponding dual problem:

$$
\begin{array}{rcccccccccccc}
\max & b_0y_0 & + & b_1y_1 & + & b_2y_2 & + & \dots & + & b_my_m & & \\
s.t. & y_0+ & & a_{11}y_1 & + & a_{21}y_2 & + & \dots & + & a_{m1}y_m & \leq & c_1 \\
 & y_0+ & & a_{12}y_1 & + & a_{22}y_2 & + & \dots & + & a_{m2}y_m & \leq & c_2 \\
 & & & & & & & \dots & & & & \\
 & y_0+ & & a_{1n}y_1 & + & a_{2n}y_2 & + & \dots & + & a_{mn}y_m & \leq & c_n \\
 & & & & & & & & & y_0 & \leq & 0
\end{array}
$$

- Here x_0 is a slack variable, b_0 is unspecified but is thought of as being arbitrarily large. Notice that a feasible dual solution is readily available: $y_0 = \min\{0, c_1, \dots, c_n\}$ and $y_i = 0$ for $i > 0$.

Finding initial dual basic feasible solution

- The initial dual basic feasible solution can be constructed by pivoting using x_0 as leaving variable and using x_i with the minimum c_i as entering variable. Note that the choice of entering variable ensure that all entries in the first row is nonnegative and thus we have a dual basic feasible solution.
- For example, consider the following primal problem:

$$\begin{array}{rrrrrr} \min & -x_1 & -5x_2 & +x_3 & & \\ s.t. & 2x_1 & -x_2 & +x_3 & \leq & 1 \\ & 3x_1 & +4x_2 & -x_3 & \leq & 1 \\ & x_1, & x_2, & x_3 & \geq & 0 \end{array}$$

and the modified primal:

$$\begin{array}{rrrrrrrrr} \min & & -x_1 & -5x_2 & +x_3 & & & & \\ s.t. & x_0 & +x_1 & +x_2 & +x_3 & & & = & M \\ & & 2x_1 & -x_2 & +x_3 & +x_4 & & = & 1 \\ & & 3x_1 & +4x_2 & -x_3 & & +x_5 & = & 1 \\ & & x_1, & x_2, & x_3, & x_4, & x_5 & \geq & 0 \end{array}$$

Finding initial dual basic feasible solution: An example

- The corresponding simplex tabular is:

	x_0	x_1	x_2	x_3	x_4	x_5	RHS
Basis	$\overline{c_0}$=0	$\overline{c_1}$=-1	$\overline{c_2}$=-5	$\overline{c_3}$= 1	$\overline{c_4}$=0	$\overline{c_5}$=0	$-z = 0$
x_0	1	1	1	1	0	0	M
x_4	0	2	-1	1	1	0	1
x_5	0	3	4	-1	0	1	1

- Now let x_0 leave and x_2 enter the basis. The new tableau is dual feasible (although not primal feasible).

	x_0	x_1	x_2	x_3	x_4	x_5	RHS
Basis	$\overline{c_0}$=5	$\overline{c_1}$=4	$\overline{c_2}$=0	$\overline{c_3}$= 6	$\overline{c_4}$=0	$\overline{c_5}$=0	$-z = -5M$
x_2	1	1	1	1	0	0	M
x_4	1	3	0	0	1	0	$1 + M$
x_5	-4	-1	0	-5	0	1	$1 - 4M$

- More practical approaches please refer to Koberstein2007.