I

Quy hoạch tuyến tính

1 Cơ sở Lý thuyết

§1.1 Một số định lý nền tảng

Định lý 1.1.1 (Định lý đối ngẫu yếu - `Weak Duality`)

(i) Nếu $x \in \mathbb{R}^n$ là nghiệm khả thi của bài toán gốc (`Primal`), gọi là $\mathcal{P}$; và $y \in \mathbb{R}^m$ là nghiệm khả thi của bài toán đối ngẫu (`Dual`), gọi là $\mathcal{D}$, thì:

$$c^T x \leq y^T A x \leq b^T y.$$

Do đó, nếu $\mathcal{P}$ không bị chặn, thì $\mathcal{D}$ vô nghiệm; và ngược lại nếu $\mathcal{D}$ không bị chặn, thì $\mathcal{P}$ vô nghiệm.

(ii) Hơn nữa, nếu $c^T \bar{x} = b^T \bar{y}$ với $\bar{x}$ là nghiệm khả thi cho $\mathcal{P}$ và $\bar{y}$ là nghiệm khả thi cho $\mathcal{D}$, thì $\bar{x}$ và $\bar{y}$ cũng sẽ là nghiệm tối ưu tương ứng.

Chứng minh. `(i)` Giả sử $x \in \mathbb{R}^n$ là nghiệm khả thi của $\mathcal{P}$ và $y \in \mathbb{R}^m$ là nghiệm khả thi của $\mathcal{D}$. Khi đó:

$$c^T x = \sum_{j=1}^{n} c_j x_j \leq \sum_{j=1}^{n} \left(\sum_{i=1}^{m} a_{ij} y_i \right) x_j \quad (\text{do } 0 \leq x_j \text{ và } c_j \leq \sum_{i=1}^{m} a_{ij} y_i),$$

$$= y^T A x = \sum_{i=1}^{m} \left(\sum_{j=1}^{n} a_{ij} x_j \right) y_i \leq \sum_{i=1}^{m} b_i y_i = b^T y,$$

vì $0 \leq y_i$ và $\sum_{j=1}^{n} a_{ij} x_j \leq b_i \Rightarrow \left(\sum_j a_{ij} x_j \right) y_i \leq b_i y_i$.

(ii) Giả sử ta có $c^T \bar{x} = b^T \bar{y}$ với $\bar{x}$ là nghiệm khả thi cho $\mathcal{P}$ và $\bar{y}$ là nghiệm khả thi cho $\mathcal{D}$. Ta sẽ chứng minh $\bar{x}$ và $\bar{y}$ cũng sẽ là nghiệm tối ưu tương ứng cho $\mathcal{P}$ và $\mathcal{D}$.

Trước tiên để ý rằng với mọi cặp (x, y) là cặp nghiệm khả thi của $\mathcal{P}$ và $\mathcal{D}$ thì theo **(i)**, ta có bất đẳng thức sau:

$$c^T x \leq b^T y.$$

Do $\bar{x}, \bar{y}$ cũng là cặp nghiệm khả thi theo giả thiết, và thỏa $c^T \bar{x} = b^T \bar{y}$, nên ta có:

$$c^T x \leq b^T \bar{y} = c^T \bar{x} \leq b^T y.$$

Điều này cho thấy $c^T \bar{x}$ là giá trị lớn nhất của hàm mục tiêu cho $\mathcal{P}$ và $b^T \bar{y}$ là giá trị nhỏ nhất của hàm mục tiêu cho $\mathcal{D}$, nên $\bar{x}$ và $\bar{y}$ là các nghiệm tối ưu tương ứng.

□

Định lý 1.1.2 (Định lý cơ bản của Quy hoạch tuyến tính (`Fundamental Theorem of Linear Programming`))
Mọi bài toán quy hoạch tuyến tính, giả sử gọi là $\mathcal{P}$, đều thỏa mãn ba tính chất sau:

(i) Nếu $\mathcal{P}$ không có nghiệm tối ưu, thì nó hoặc vô nghiệm, hoặc không bị chặn.

(ii) Nếu $\mathcal{P}$ có nghiệm khả thi (`feasible solution`), thì nó cũng tồn tại một nghiệm cơ sở khả thi (`basic feasible solution`).

(iii) Nếu $\mathcal{P}$ bị chặn, thì tồn tại một nghiệm cơ sở khả thi tối ưu (`optimal basic feasible solution`).

Chứng minh. **(i)** Giả sử $\mathcal{P}$ không có nghiệm tối ưu. Khi đó hoặc nó vô nghiệm, hoặc nó có nghiệm khả thi nhưng không bị chặn.

Trong trường hợp $\mathcal{P}$ có nghiệm khả thi, thì pha đầu tiên của thuật toán đơn hình hai pha (`two-phase`) sẽ xây dựng được một nghiệm cơ sở khả thi. Sau đó, pha 2 sẽ hoặc tìm ra nghiệm tối ưu, hoặc phát hiện bài toán không bị chặn.

Từ giả thiết ta đã có $\mathcal{P}$ không tồn tại nghiệm tối ưu, nên $\mathcal{P}$ không bị chặn. Vậy $\mathcal{P}$ hoặc vô nghiệm, hoặc không bị chặn.

(ii) Nếu $\mathcal{P}$ có nghiệm khả thi, thì pha đầu tiên của thuật toán đơn hình hai pha sẽ tìm được một nghiệm cơ sở khả thi.

(iii) Giả sử $\mathcal{P}$ bị chặn. Khi đó, bài toán tồn tại miền khả thi (`feasible region`), nên theo (ii), tồn tại một nghiệm cơ sở khả thi.

Khi đó pha 2 của thuật toán đơn hình sẽ hoặc tìm ra nghiệm tối ưu, hoặc phát hiện bài toán không bị chặn. Nhưng do giả thiết, $\mathcal{P}$ bị chặn, nên $\mathcal{P}$ có nghiệm cơ sở khả thi tối ưu. □

§1.2 Những kết quả cần chứng minh

Định lý 1.2.1 (Phân tích độ phức tạp của thuật toán đơn hình)
Xét một bài toán quy hoạch tuyến tính dạng chuẩn với n biến và m ràng buộc. Thuật toán đơn hình là một phương pháp duyệt qua các đỉnh cực biên (nghiệm cơ sở) của miền khả thi để tìm giá trị tối ưu. Độ phức tạp thời gian của thuật toán đơn hình không cố định, mà phụ thuộc mạnh vào đặc điểm hình học của bài toán và chiến lược chọn biến. Tổng quan nhất, có ba trường hợp sau đây:

(i) **Tốt nhất:** Tốt nhất tuyệt đối là $O(1)$ nếu bắt đầu tại nghiệm tối ưu; $O(n)$ hoặc $O(m)$ trong trường hợp thuận lợi.

(ii) **Trung bình:** $\mathbb{E}[T]$ là một đa thức theo n, m

(iii) **Tệ nhất:** $T_{\text{worst}} = \Omega(2^n)$.

Chứng minh. **(i) Trường hợp tốt nhất:**

Trong một số trường hợp lý tưởng nhất, thuật toán đơn hình có thể tìm được nghiệm tối ưu sau rất ít bước lặp. Đặc biệt, nếu cơ sở ban đầu trùng với nghiệm tối ưu của bài toán, thuật toán sẽ dừng ngay lập tức mà không cần thực hiện bất kỳ bước `pivot` nào. Khi đó, độ phức tạp thời gian là hằng số:

$$T_{\text{best}} = O(1).$$

Tuy nhiên, đây là một tình huống hiếm gặp, chỉ xảy ra nếu ta may mắn chọn được một cơ sở ban đầu tối ưu hoặc bài toán có cấu trúc đặc biệt thuận lợi.

Mặt khác, trong các bài toán thực tế có cấu trúc đơn giản, ít suy biến hoặc được khởi tạo tốt (chẳng hạn như nhờ thuật toán hai pha), số bước `pivot` cần làm để đạt nghiệm tối ưu có thể chỉ tăng tuyến tính theo kích thước đầu vào:

$$T = O(n) \quad \text{hoặc} \quad O(m),$$

với n là số biến và m là số ràng buộc.

Tóm lại, trong trường hợp tốt nhất, nếu cơ sở ban đầu là tối ưu, thuật toán dừng ngay với độ phức tạp tốt nhất tuyệt đối (chỉ mang tính lý thuyết) là $O(1)$. Tuy nhiên, trong thực tế, độ phức tạp tốt nhất thường được ghi nhận ở mức $O(n)$ hoặc $O(m)$, khi bài toán có cấu trúc thuận lợi và không suy biến.

(ii) Trường hợp trung bình:
Với giả thiết rằng dữ liệu đầu vào được chọn ngẫu nhiên đều trên cầu đơn vị (`unit sphere`), `Smale` (1983) [4] và `Borgwardt` (1982) [5] đã chứng minh rằng thuật toán đơn hình có thời gian chạy kỳ vọng bị chặn bởi:

$$\mathbb{E}[T] \leq C(n+m),$$

trong đó C là một hằng số dương phụ thuộc vào mô hình xác suất cụ thể (ví dụ phân phối đều trên đơn vị cầu hoặc khối cầu), nhưng *không phụ thuộc vào số biến hay số ràng* *buộc*. Đây là một kết quả quan trọng cho thấy thời gian chạy trung bình là tuyến tính theo kích thước đầu vào trong điều kiện ngẫu nhiên.

Mặt khác, trong mô hình *làm trơn* (`smoothed analysis`) do `Spielman` và `Teng` (2001) [7] đề xuất, giả sử mỗi hệ số trong dữ liệu đầu vào bị nhiễu `Gaussian` nhỏ độc lập với phương sai σ^2, thì thời gian chạy kỳ vọng của thuật toán sẽ bị chặn bởi một đa thức theo n và $1/\sigma$.

Cụ thể, điều đó đồng nghĩa với việc tồn tại một hằng số d sao cho:

$$\mathbb{E}_\sigma[T] \leq C' \cdot \left(\frac{n}{\sigma}\right)^d,$$

trong đó C' và d là các hằng số phụ thuộc vào mô hình phân tích và không phụ thuộc vào dữ liệu cụ thể.

Kết quả này cho thấy rằng chỉ cần một lượng nhiễu rất nhỏ trong dữ liệu (dù là không chủ đích), thì trường hợp tệ nhất gần như bị loại bỏ, và thuật toán sẽ hoạt động với độ phức tạp gần tuyến tính trong hầu hết các tình huống thực tế.

(iii) Trường hợp tệ nhất:
Đa diện khả thi (miền khả thi) của một bài toán Quy hoạch tuyến tính có thể có đến $\binom{n}{m}$ đỉnh, với n là số biến và m là số ràng buộc. Do đó, trong trường hợp xấu nhất, thuật toán đơn hình có thể phải duyệt qua toàn bộ các đỉnh này để tìm nghiệm tối ưu.

Tổng tất cả các tổ hợp con của n phần tử chính là:

$$\sum_{m=0}^{n} \binom{n}{m} = 2^n,$$

tức là hình lập phương n-chiều có 2^n đỉnh.

Trong đó, giá trị lớn nhất của $\binom{n}{m}$ xảy ra khi $m = \lfloor n/2 \rfloor$, và theo bất đẳng thức `Stirling` ta có:

$$\binom{n}{m} \approx \frac{2^n}{\sqrt{n}}.$$

Điều này có nghĩa là: trong trường hợp tệ nhất về mặt hình học, số đỉnh cực biên có thể tăng theo hàm $\Omega\left(\frac{2^n}{\sqrt{n}}\right)$.

Do đó, trong trường hợp xấu nhất, thuật toán đơn hình có thể phải duyệt qua số lượng đỉnh cực biên với độ phức tạp:

$$T_{\text{worst}} = \Omega\left(\binom{n}{m}\right) \geq \Omega\left(\frac{2^n}{\sqrt{n}}\right).$$

Từ đó, khi phân tích độ phức tạp theo bậc lớn, ta có thể viết ngắn gọn:

$$T_{\text{worst}} = \Omega(2^n)$$

□

Nhận xét 1.2.2 (Bài toán Klee–Minty)**.** Vào năm 1972, `Klee` và `Minty` đã đưa ra một ví dụ kinh điển [1] cho thấy rằng thuật toán đơn hình, khi sử dụng quy tắc chọn biến `Dantzig` (chọn biến có hệ số mục tiêu lớn nhất), có thể rơi vào trường hợp tệ nhất với độ phức tạp **hàm mũ** theo số biến.

Họ xây dựng một bài toán quy hoạch tuyến tính trong không gian $\mathbb{R}^n$ có dạng:

$$\text{Maximize} \quad z = x_1 + 10x_2 + 10^2 x_3 + \cdots + 10^{n-1} x_n$$

$$\text{Subject to} \quad \begin{cases} 0 \le x_1 \le 1 \\ 0 \le x_2 \le 1 - 0.1x_1 \\ 0 \le x_3 \le 1 - 0.1x_2 \\ \quad\vdots \\ 0 \le x_n \le 1 - 0.1x_{n-1} \end{cases}$$

Tập nghiệm của bài toán này tạo thành một hình lập phương n-chiều bị bóp méo (`distorted hypercube`), tức là một đa diện khả thi có đúng 2^n đỉnh, nhưng không còn đối xứng như hình lập phương đơn vị ban đầu.

Trong trường hợp này, thuật toán `Simplex` bắt đầu từ đỉnh $(0, 0, \ldots, 0)$ và duyệt tuần tự qua từng đỉnh của hình lập phương theo thứ tự "xấu nhất" — nghĩa là phải đi qua toàn bộ 2^n đỉnh trước khi tới nghiệm tối ưu. Do đó, độ phức tạp của thuật toán trong ví dụ này là:

$$T_{\text{Klee–Minty}} = 2^n - 1,$$

tức là:

$$\boxed{T_{\text{worst}} = \Omega(2^n)}.$$

Ví dụ này phủ định giả thuyết khi đó rằng thuật toán đơn hình luôn chạy nhanh, và trở thành phản ví dụ kinh điển cho thấy độ phức tạp tệ nhất của thuật toán này là một siêu đa thức. Để dễ hiểu thì một hàm $f(n)$ được gọi là **siêu đa thức** (*super-polynomial*) nếu:

$$\lim_{n \to \infty} \frac{f(n)}{n^k} = \infty \quad \text{với mọi } k \in \mathbb{N}.$$

Nói cách khác, hàm siêu đa thức là hàm có tốc độ tăng lớn hơn mọi hàm đa thức khi n đủ lớn.

Nói tóm lại, ví dụ của `Klee` và `Minty` làm rõ mối liên hệ giữa số bước lặp và số đỉnh cực biên của hình lập phương n-chiều, khi cho thấy rằng một cấu trúc bài toán phù hợp có thể buộc thuật toán phải duyệt toàn bộ các đỉnh của một `hypercube`.

Định lý 1.2.3 (Định lý đối ngẫu mạnh (`Strong Duality`))

Xét một bài toán quy hoạch tuyến tính gốc $(\mathcal{P})$ dạng chuẩn và bài toán đối ngẫu $(\mathcal{D})$ của nó:

$$(\mathcal{P}): \ \max\{\, c^T x \mid Ax \le b,\ x \ge 0\,\}, \qquad (\mathcal{D}): \ \min\{\, b^T y \mid A^T y \ge c,\ y \ge 0\,\}.$$

Khi đó:

(i) Nếu bài toán $(\mathcal{P})$ có nghiệm tối ưu hữu hạn, thì bài toán $(\mathcal{D})$ cũng có nghiệm tối ưu hữu hạn và hai giá trị mục tiêu tối ưu bằng nhau.

(ii) Ngược lại, nếu một bài toán không bị chặn thì bài toán kia vô nghiệm.

Nói cách khác, nếu cả hai bài toán đều có miền khả thi không rỗng thì `duality gap` bằng 0.

Chứng minh. **Chiều thuận** $(\Rightarrow)$: Giả sử bài toán $(\mathcal{P})$ có nghiệm tối ưu hữu hạn. Theo định lý (1.1.2), tồn tại một nghiệm cơ sở tối ưu. Khi đó, theo công thức tổng quát của bảng đơn hình, tồn tại một ma trận khả nghịch $R \in \mathbb{R}^{n\times n}$ và một vector $y \in \mathbb{R}^m$ sao cho bảng đơn hình tối ưu có dạng như sau:

$$\begin{bmatrix} R & 0 \\ -y^T & 1 \end{bmatrix} \cdot \begin{bmatrix} A & I & b \\ c^T & 0 & 0 \end{bmatrix} = \begin{bmatrix} RA & R & Rb \\ c^T - y^T A & -y^T & -y^T b \end{bmatrix}.$$

Vì đây là bảng đơn hình tối ưu nên

$$c - A^T y \le 0, \quad y \ge 0,$$

với giá trị tối ưu cho hàm mục tiêu của $(\mathcal{P})$ là $c^T x = b^T y$. Điều này cho thấy y là nghiệm khả thi với $(\mathcal{D})$ và đạt được cùng giá trị tối ưu.

Mặt khác, theo định lý đối ngẫu yếu (1.1.1), với mọi nghiệm khả thi $\hat{y}$ của bài toán $(\mathcal{D})$, ta luôn có:

$$c^T x \le b^T \hat{y}.$$

Vì $c^T x = b^T y$, nên:

$$b^T y \le b^T \hat{y}, \quad \forall\, \hat{y} \text{ khả thi của } (\mathcal{D}),$$

tức là y đạt giá trị nhỏ nhất của hàm mục tiêu trong $(\mathcal{D})$, do đó y là nghiệm tối ưu.

Chiều đảo $(\Leftarrow)$: Giả sử bài toán $(\mathcal{P})$ không bị chặn, tức là tồn tại dãy nghiệm khả thi $\{x^{(k)}\}$ sao cho $c^T x^{(k)} \to +\infty$. Nếu tồn tại một nghiệm khả thi y của $(\mathcal{D})$, thì theo định lý đối ngẫu yếu (1.1.1):

$$c^T x^{(k)} \le b^T y \quad \forall k.$$

Dễ thấy vế trái tiến tới $+\infty$ còn vế phải là hằng số, dẫn đến mâu thuẫn. Do đó, $(\mathcal{D})$ không thể có nghiệm khả thi.

Ta đã chứng minh cả hai ý **(i)** và **(ii)** của Định lý này. Với trường hợp $(\mathcal{D})$ không bị chặn hoặc $(\mathcal{P})$ vô nghiệm cũng được xử lý tương tự như trên do tính đối ngẫu. □

Định lý 1.2.4 (Định lý độ lệch bù (`Complementary Slackness`))
Cho bài toán gốc $(\mathcal{P})$ và đối ngẫu $(\mathcal{D})$ dạng chuẩn như sau:

$$(\mathcal{P}):\ \max\{\, c^T x \mid Ax \le b,\ x \ge 0\,\}, \qquad (\mathcal{D}):\ \min\{\, b^T y \mid A^T y \ge c,\ y \ge 0\,\}.$$

Khi đó, cặp nghiệm (x, y) là nghiệm tối ưu của $(\mathcal{P})$ và $(\mathcal{D})$ nếu và chỉ nếu:

$$y_i \cdot (b_i - \sum_{j=1}^{n} a_{ij} x_j) = x_{n+i} y_i = 0 \quad \forall i = 1, \ldots, m$$

$$x_j \cdot (\sum_{i=1}^{m} a_{ij} y_i - c_j) = y_{m+j} y_j = 0 \quad \forall j = 1, \ldots, n.$$

Chứng minh. **Chiều thuận $(\Rightarrow)$** : Giả sử x và y là nghiệm khả thi của $(\mathcal{P})$ và $(\mathcal{D})$, tức là:

$$Ax \le b, \quad x \ge 0; \qquad A^T y \ge c, \quad y \ge 0.$$

Khi đó:

$$x^T A^T y \ge x^T c, \qquad y^T Ax \le y^T b \tag{1.1}$$

Để ý rằng $\mathbf{x}^T A^T \mathbf{y} = \langle \mathbf{x}, A^T \mathbf{y} \rangle = \langle A\mathbf{x}, \mathbf{y} \rangle = \mathbf{y}^T A\mathbf{x}$.

Ta lại có $\mathbf{x}^T \in \mathbb{R}^{1\times n}$, $A^T \in \mathbb{R}^{n\times m}$ nên $\mathbf{x}^T A^T \in \mathbb{R}^{1\times m}$; mà $\mathbf{y} \in \mathbb{R}^{m\times 1}$ nên $\mathbf{x}^T A^T \mathbf{y} \in \mathbb{R}^{1\times 1}$, tức là một ma trận 1×1

Vậy từ (1.1) suy ra:

$$x^T c \le x^T A^T y = y^T Ax \le y^T b. \tag{1.2}$$

Theo định lý đối ngẫu mạnh (1.2.3), x và y là các nghiệm tối ưu của $(\mathcal{P})$ và $(\mathcal{D})$ khi và chỉ khi:

$$x^T c = y^T b \text{ (tổng quát là } \langle c, x\rangle = \langle b, y\rangle).$$

Do đó, dấu bằng xảy ra tại hai đầu của (1.2), suy ra:

$$x^T (A^T y - c) = 0, \qquad y^T (b - Ax) = 0.$$

Xét $x^T (A^T y - c) = 0$, ta có:

$$0 = x^T (A^T y - c) = \sum_{j=1}^{n} x_j \left(\sum_{i=1}^{m} a_{ji} y_i - c_j \right).$$

Do $x_j \ge 0$ và $\sum_{i=1}^{m} a_{ji} y_i - c_j \ge 0$, nên mỗi hạng tử đều không âm; tổng bằng 0 thì từng hạng tử phải bằng 0, tức là:

$$x_j \left(\sum_{i=1}^{m} a_{ji} y_i - c_j \right) = 0, \quad \forall\, j = 1, \ldots, n.$$

Tương tự, từ $y^T (b - Ax) = 0$ ta cũng suy ra:

$$y_i \left(b_i - \sum_{j=1}^{n} a_{ij} x_j \right) = 0, \quad \forall\, i = 1, \ldots, m.$$

Đây chính là hai điều kiện ở *Chiều thuận* ta cần chứng minh. □

Chiều ngược lại ($\Leftarrow$): Giả sử $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ là cặp nghiệm khả thi của $(\mathcal{P})$ và $(\mathcal{D})$, tức là:

$$Ax \leq b, \quad x \geq 0; \qquad A^T y \geq c, \quad y \geq 0.$$

Đặt $w_i = b_i - \sum_{j=1}^n a_{ij}x_j$, $z_j = \sum_{i=1}^m a_{ij}y_i - c_j$ và giả sử hai điều kiện độ lệch bù được thỏa mãn:

$$x_j z_j = 0, \quad y_i w_i = 0 \quad \forall\, j = 1, \ldots, n;\ i = 1, \ldots, m.$$

Ta cần chứng minh (x, y) là nghiệm tối ưu của $(\mathcal{P})$ và $(\mathcal{D})$.

Xét lại chuỗi bất đẳng thức đối ngẫu yếu:

$$c^T x \leq y^T A x \leq b^T y.$$

Ta sẽ chứng minh dấu bằng xảy ra ở cả 2 bất đẳng thức trên. Thật vậy:

$$x^T A^T y = x^T c + \sum_{j=1}^n x_j z_j = c^T x \quad (\text{do } x_j z_j = 0),$$

$$y^T A x = b^T y - \sum_{i=1}^m y_i w_i = b^T y \quad (\text{do } y_i w_i = 0).$$

Vậy:

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

Theo định lý đối ngẫu yếu (1.1.1), điều này chứng tỏ x và y là nghiệm tối ưu tương ứng của $(\mathcal{P})$ và $(\mathcal{D})$. *Chiều đảo* của định lý được chứng minh.

□

Tài liệu tham khảo

[1] Klee, V., & Minty, G. J. (1972). How good is the simplex algorithm? In *Inequalities III*, Academic Press, pp. 159–175.

[2] Optimization Wiki. Simplex Algorithm. https://optimization.cbe.cornell.edu/index.php?title=Simplex_algorithm

[3] University of Waterloo. Worst-case behavior of the Simplex Algorithm (Klee–Minty cube). https://cs.uwaterloo.ca/~vganesh/courses/CS860-F15/lectures/lecture6.pdf

[4] Smale, S. (1983). On the average number of steps of the simplex method of linear programming. *Mathematical Programming*, 27, 241–262. https://doi.org/10.1007/BF02591902

[5] Borgwardt, K.-H. (1982). The average number of steps required by the simplex method is polynomial. *Zeitschrift für Operations Research*, 26(3), 157–177. https://link.springer.com/article/10.1007/BF01580645

[6] Megiddo, N. (1986). Improved asymptotic analysis of the average number of steps performed by the self-dual simplex algorithm. *Mathematical Programming*, 35, 140–172. https://doi.org/10.1007/BF01580645

[7] Spielman, D. A., & Teng, S.-H. (2001). Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. In *Proc. of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pp. 296–305. https://arxiv.org/abs/cs/0111050

[8] Spielman, D. A., & Teng, S.-H. (2004). Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time. *Journal of the ACM*, 51(3), 385–463. https://en.wikipedia.org/wiki/Smoothed_analysis

[9] Boyd, S., & Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press. Chapter 5. https://web.stanford.edu/~boyd/cvxbook/

[10] MIT OpenCourseWare. Lecture notes on LP duality and complementary slackness. https://ocw.mit.edu/courses/15-053-optimization-methods-in-management-science-spring-2013/resources/lecture-notes/

[11] Wikipedia. Duality (optimization). https://en.wikipedia.org/wiki/Duality_(optimization)

[12] Bazarra, M. S., Jarvis, J. J., & Sherali, H. D. (2020). *Linear Programming: Foundations and Extensions* (5th Ed.). Springer. https://link.springer.com/book/10.1007/978-3-030-39415-8