線性規劃與整數規劃

完整教材 (依據 5CCS2FC2 Foundations of Computing II 第八週內容編寫)

線性規劃、單純形法、整數規劃與分枝定界

整理自 Dr. Christopher Hampson(King's College London) 之投影片
並參考補充文獻擴充編寫

2026 年 7 月

本週學習目標

1. 理解**線性規劃 (LP)**: 線性目標函數、線性限制式、可行區域與最佳解, 並會用矩陣形式 $A\vec{x} \leq \vec{b}$ 表達。
2. 會把實際問題 (能源調度、工作指派) **建模**成線性/整數規劃。
3. 掌握**單純形法 (Simplex Method)**: 鬆弛變數、表格 (tableau)、樞軸運算, 並能完整解出一個小型 LP。
4. 知道單純形法最壞情況是**指數時間** (Klee–Minty), 但 LP 本身可在**多項式時間**求解 (Khachiyan)。
5. 理解**整數規劃 (IP)** 是 NP-hard(SAT $\leq_p$ IP), 並掌握**線性鬆弛**與**分枝定界 (Branch-and-Bound)**。

Contents

1 線性規劃 (Linear Programming)

1.1 定義與基本組成

定義 1.1 (線性規劃的組成)**.** 一個**線性規劃**問題由兩部分構成:

- **線性目標函數 (objective function)**: 要最大化/最小化的量;
- **線性限制式 (constraints)**: 一組線性不等式, 限制哪些解是「可行的」。

貫穿本週的例子

$$\begin{aligned} \text{最大化:} \quad & 2x + 3y \\ \text{限制式:} \quad & 3x + 2y \le 15 \\ & 2y - x \le 5 \\ & x + 2y \le 7 \end{aligned}$$

(LP 可能有唯一解、無窮多解, 或無解。)

1.2 矩陣表示法

線性限制式可以簡潔地寫成矩陣–向量不等式 $A\vec{x} \le \vec{b}$:

$$\underbrace{\begin{pmatrix} 3 & 2 \\ -1 & 2 \\ 1 & 2 \end{pmatrix}}_{A} \underbrace{\begin{pmatrix} x \\ y \end{pmatrix}}_{\vec{x}} \le \underbrace{\begin{pmatrix} 15 \\ 5 \\ 7 \end{pmatrix}}_{\vec{b}}$$

(注意 $2y - x \le 5$ 改寫成 $-x + 2y \le 5$ 後, 係數列為 $(-1, 2)$。)

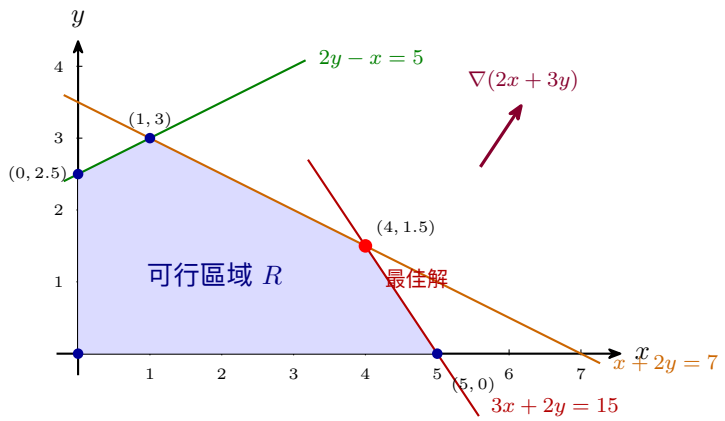
定義 1.2 (線性規劃問題 LP)**. 輸入** 一組線性限制式 $A\vec{x} \le \vec{b}$, 以及線性目標函數 $f : \mathbb{R}^n \to \mathbb{R}$。
輸出 一個解 $\vec{x} \in (\mathbb{R}_{\ge 0})^n$, 使得 (i) $\vec{x}$ 滿足 $A\vec{x} \le \vec{b}$, 且 (ii) $f(\vec{x})$ 最大。

1.3 可行區域 (Feasible Region)

定義 1.3 (可行區域與最佳解)**. 可行區域**是滿足所有線性限制式的解所成的集合

$$R = \{\vec{x} \in \mathbb{R}^n : \vec{x} \text{ 滿足所有限制式}\}.$$

給定目標函數 f, **最佳解集合**為 $\{\vec{x} \in R : \forall \vec{y} \in R\,(f(\vec{x}) \ge f(\vec{y}))\} \subseteq R$。



備註 1.4 (幾何直覺). 可行區域是凸多面體 (polytope)。沿著目標函數的梯度方向平移「等值線」, 最後離開可行區域的接觸點必落在**頂點** (或一整條邊) 上——因此**最佳解必可在頂點處取得**。本例中五個頂點的目標值分別為 0, 10, 12.5, 11, 7.5, 最大值 12.5 在頂點 $(4, 1.5)$。這個觀察正是單純形法的理論基礎。

1.4 建模實例: 能源調度 (Balancing Energy Demands)

能源生產資料

	天然氣	煤	太陽能	風力	核能
CO_2 排放 (噸/GWh)	500	1000	0	0	0
成本 (GBP/kWh)	40	80	22	22	120
容量 (GW)	32	8	13.5	24	10

建模步驟:

1. **引入變數**: 對每種發電方式各設一個數值變數 G, C, S, W, N, 代表其發電量。
2. **容量限制**: 每種方式不得超過其容量:

$$G \le 32, \quad C \le 8, \quad S \le 13.5, \quad W \le 24, \quad N \le 10.$$

3. **需求限制**: 總發電量必須滿足能源需求:

$$G + C + S + W + N \ \ge \ \text{總能源需求}.$$

4. **排放限制**: 不得超過 CO_2 排放目標:

$$0.5\,G + 1\,C \ \le \ CO_2 \text{ 排放目標}.$$

5. **目標函數**: 最小化總成本:

$$\text{最小化} \quad 40\,G + 80\,C + 22\,S + 22\,W + 120\,N.$$

備註 1.5. 「$\geq$」限制式乘上 -1 即可化為「$\leq$」;「最小化 f」等價於「最大化 $-f$」。因此上述問題仍是標準形式的 LP。

2 單純形法 (The Simplex Method)

2.1 歷史背景

Dantzig 與單純形法

- 單純形法由 **George Dantzig** 於 1946–47 年提出, 源自他在二戰期間為美國空軍所做的作業研究 (operations research)。
- 名稱來自一種 n 維多邊形「**單純形** (simplex)」, 因為可行區域的形狀。
- 趣聞:Dantzig 常被認為是電影《心靈捕手》(Good Will Hunting) 中 Matt Damon 飾演的 Will 的靈感來源——他當學生時遲到, 把黑板上兩個未解的統計難題誤當作業解掉了。

2.2 鬆弛變數 (Slack Variables)

對每條限制式引入一個**鬆弛變數**, 把不等式變成等式, 稱為**鬆弛形式 (slack form)**:

$$\begin{aligned} 3x+2y &\leq 15 \\ 2y-x &\leq 5 \\ x+2y &\leq 7 \end{aligned} \quad\Longrightarrow\quad \begin{aligned} 3x+2y+s_1 &= 15 \\ 2y-x+s_2 &= 5 \\ x+2y+s_3 &= 7 \end{aligned} \qquad (s_1, s_2, s_3 \geq 0)$$

每給定自變數 x, y, 即可算出因變的鬆弛變數:

$$s_1 = 15-3x-2y, \qquad s_2 = 5+x-2y, \qquad s_3 = 7-x-2y.$$

例如 $(x,y)=(3,2)$ 時 $(s_1,s_2,s_3)=(2,4,0)$——$s_3=0$ 表示第三條限制式「頂到邊界」(緊的),$s_i>0$ 表示還有「鬆弛空間」。

2.3 成本變數與表格 (Tableau)

把目標函數也當成一條等式, 引入**成本變數** C:

$$C = 2x+3y \quad\Longleftrightarrow\quad -2x-3y+C = 0.$$

把所有等式的係數排成矩陣, 即得**表格 (tableau)**:

$$\begin{aligned} 3x+2y+s_1 &= 15 \\ -x+2y+s_2 &= 5 \\ x+2y+s_3 &= 7 \\ -2x-3y+C &= 0 \end{aligned} \quad\Longrightarrow$$

x	y	s_1	s_2	s_3	C	
3	2	1	0	0	0	15
-1	2	0	1	0	0	5
1	2	0	0	1	0	7
-2	-3	0	0	0	1	0

2.4 單純形法的步驟

單純形法 (表格版)

1. 由鬆弛形式與成本等式建立**初始表格**。
2. 找出最後一列 (目標函數列) 中**最負的係數**, 其所在行為「**樞軸行** (pivot column)」。
3. 計算**列商 (row quotients)**: 最後一欄的每個元素除以樞軸行的對應元素。
4. 取**列商最小**且分子、分母皆為正的那一列為「**樞軸列** (pivot row)」。
5. 「**樞軸值**」= 樞軸行與樞軸列交會處的值。施行列運算:

$$R_{\text{pivot}} \leftarrow \frac{1}{\text{樞軸值}} \times R_{\text{pivot}}, \qquad R_i \leftarrow R_i - \frac{R_i \text{ 在樞軸行的值}}{\text{樞軸值}} \times R_{\text{pivot}} \quad (i \neq \text{pivot}).$$

6. 重複步驟 2–5, 直到最後一列**沒有負係數**為止。
7. 從最終表格**讀出最佳解**。

2.5 完整範例

第 1 輪. 最後一列最負係數為 -3(在 y 行)$\rightarrow$ 樞軸行為 y。列商:$15/2,\ 5/2,\ 7/2$, 最小為 $5/2 \rightarrow$ 樞軸列為第 2 列, 樞軸值 $= 2$。

x	y	s_1	s_2	s_3	C		列商
3	2	1	0	0	0	15	$15/2$
-1	$[\mathbf{2}]$	0	1	0	0	5	$\mathbf{5/2}\ \leftarrow$
1	2	0	0	1	0	7	$7/2$
-2	$\mathbf{-3}$	0	0	0	1	0	

列運算 $R_2 \leftarrow \frac{1}{2}R_2,\ R_1 \leftarrow R_1 - \frac{2}{2}R_2,\ R_3 \leftarrow R_3 - \frac{2}{2}R_2,\ R_4 \leftarrow R_4 + \frac{3}{2}R_2$:

x	y	s_1	s_2	s_3	C	
4	0	1	-1	0	0	10
-0.5	1	0	0.5	0	0	2.5
2	0	0	-1	1	0	2
-3.5	0	0	1.5	0	1	7.5

第 2 輪. 最負係數 -3.5(在 x 行)。列商:$10/4 = 2.5$、$2.5/(-0.5)$(分母為負, 不合格)、$2/2 = 1$, 最小合格者為 $1 \rightarrow$ 樞軸列為第 3 列, 樞軸值 $= 2$。列運算後:

x	y	s_1	s_2	s_3	C	
0	0	1	1	-2	0	6
0	1	0	0.25	0.25	0	3
1	0	0	-0.5	0.5	0	1
0	0	0	-0.25	1.75	1	11

第 3 輪. 最負係數 -0.25(在 s_2 行)。列商:$6/1 = 6$、$3/0.25 = 12$、$1/(-0.5)$(不合格), 最小為 6 $\to$ 樞軸列為第 1 列, 樞軸值 $= 1$。列運算後:

x	y	s_1	s_2	s_3	C	
0	0	1	1	-2	0	6
0	1	-0.25	0	0.75	0	1.5
1	0	0.5	0	-0.5	0	4
0	0	0.25	0	2.25	1	12.5

最後一列已無負係數, 演算法終止。

讀出最佳解. 最終表格的每一列都是一條等式。凡是「該行恰有一個 1、其餘為 0」的變數 (基變數) 即可直接讀值; 其餘變數 (非基變數) 設為 0:

$$\underbrace{s_1 = 0, \quad s_3 = 0,}_{\text{非基變數}} \qquad \underbrace{x = 4, \quad y = 1.5, \quad s_2 = 6, \quad C = 12.5}_{\text{基變數 (讀最後一欄)}}.$$

最佳解

$$x = 4, \quad y = 1.5, \quad \text{最大值 } C = 2(4) + 3(1.5) = 12.5.$$

$s_1 = s_3 = 0$ 表示第一、三條限制式是緊的——最佳解正是這兩條邊界線的交點 $(4, 1.5)$, 與可行區域圖中的頂點吻合;$s_2 = 6$ 表示第二條限制式尚有鬆弛空間。

備註 2.1 (幾何解讀). 單純形法每做一次樞軸運算, 就相當於**沿著可行區域的邊走到相鄰的頂點**, 且每一步都使目標值不減。本例的路徑為 $(0,0) \to (0,2.5) \to (1,3) \to (4,1.5)$, 目標值 $0 \to 7.5 \to 11 \to 12.5$ 單調上升。

2.6 單純形法的複雜度

定理 2.2 (Klee–Minty, 1972)**.** 單純形法最壞情況的終止時間是**指數的**。以下的「*Klee–Minty* 立方體」會迫使 *Dantzig* 樞軸規則走遍全部 2^{n+1} 個頂點*:*

$$\begin{aligned}
\text{最大化:}\quad & 2^n x_0 + 2^{n-1} x_1 + \cdots + 2x_{n-1} + x_n \\
\text{限制式:}\quad & x_0 \le 5 \\
& 4x_0 + x_1 \le 25 \\
& 8x_0 + 4x_1 + x_2 \le 125 \\
& \quad \vdots \\
& 2^{n+1} x_0 + 2^n x_1 + \cdots + 4x_{n-1} + x_n \le 5^{n+1} \\
& x_0, x_1, \ldots, x_n \ge 0
\end{aligned}$$

定理 2.3 (Khachiyan, 1979)**.** 線性規劃可在**多項式時間**內求解。

理論 vs. 實務

- Khachiyan 的**橢球法 (ellipsoid method)** 首度證明 $\text{LP} \in \mathbf{P}$, 但實務上很慢。
- Karmarkar(1984) 的**內點法 (interior-point method)** 既是多項式時間、實務上也可與單純形法競爭。
- 儘管最壞情況是指數, 單純形法**平均**表現極佳, 至今仍是最常用的 LP 演算法之一。
- 未解問題: 是否存在使單純形法最壞情況也是多項式時間的樞軸規則?

3 整數規劃 (Integer Programming)

定義 3.1 (整數規劃問題 IP)**. 輸入** 一組線性限制式 $A\vec{x} \le \vec{b}$, 以及線性目標函數 $f : \mathbb{R}^n \to \mathbb{R}$。

輸出 一個解 $\vec{x} \in (\mathbb{Z}_{\ge 0})^n$, 使得 (i) $\vec{x}$ 滿足 $A\vec{x} \le \vec{b}$, 且 (ii) $f(\vec{x})$ 最大。

與 LP 唯一的差別: 解必須是**非負整數**向量。

3.1 建模實例: 最大匹配 (Maximal Matchings)

考慮把一群工人指派給任務 A, B, C, D, 其中每位工人 i 有容量 (可承接的任務數, 如 $\times 2, \times 3$), 每個任務 j 有需求人數。

1. 對每條「工人 i—任務 j」的邊, 設變數 $x_{i,j}$, 並加入限制 $x_{i,j} \le 1$(一條邊最多用一次)。
2. 對每位工人 i 加入**容量限制**:
$$\textstyle\sum_j x_{i,j} \;\le\; \text{工人}i\text{的容量}.$$
3. 對每個任務 j 加入**需求限制**:
$$\textstyle\sum_i x_{i,j} \;\ge\; \text{任務}j\text{的需求}.$$
4. 目標: 最大化總連結數 $\sum_{i,j} x_{i,j}$, 並要求 $x_{i,j} \in \mathbb{Z}$。

完整的整數規劃

$$\begin{aligned}
\text{最大化:}\quad & \sum_{i,j} x_{i,j} \\
\text{限制式:}\quad & x_{i,j} \le 1 \text{ 對所有 } i,j \\
& \sum_j x_{i,j} \le \text{工人}i\text{的容量 對所有 } i \\
& \sum_i x_{i,j} \ge \text{任務}j\text{的需求 對所有 } j \\
& x_{i,j} \in \mathbb{Z} \text{ 對所有 } i,j
\end{aligned}$$

最佳整數解 $x_{i,j} = 1$ 的邊, 恰好構成一個工人對任務的**最大匹配**。

備註 3.2 (為什麼整數限制讓問題變難?). 直覺上「先解 LP 再四捨五入」似乎可行, 但事實上: 捨入後的解**可能不可行** (違反限制式), 即使可行也**可能離最佳解很遠**。整數限制破壞了可行區域的凸性——可行解變成離散的格子點,「沿邊走到頂點」的幾何論證完全失效。

4 整數規劃是 NP-hard

定理 4.1. 整數規劃 *IP* 是 *NP-hard*。

定理 4.2. 布林可滿足性問題可多項式歸約到整數規劃:$\textit{SAT} \le_p \textit{IP}$。

Proof. 給定一個 CNF 公式 F, 我們建構整數規劃 IP_F 使得

$$F \text{ 可滿足} \iff \text{IP}_F \text{ 有整數解.}$$

步驟 1(變數). 對每個命題變數 P, 引入兩個數值變數 x_P 與 $x_{\neg P}$(分別代表「P 為真」與「P 為假」), 並限制 $0 \le x_P, x_{\neg P} \le 1$。

步驟 2(子句 $\to$ 限制式). 每個子句轉成一條線性限制: 子句中各文字對應的變數之和至少為 1(至少一個文字為真):

$$\begin{aligned} (P \vee Q \vee \neg R) &\quad\rightsquigarrow\quad x_P + x_Q + x_{\neg R} \ge 1 \\ (\neg P \vee Q \vee R) &\quad\rightsquigarrow\quad x_{\neg P} + x_Q + x_R \ge 1 \\ (\neg Q \vee S) &\quad\rightsquigarrow\quad x_{\neg Q} + x_S \ge 1 \end{aligned}$$

步驟 3(一致性). 對每個命題變數加入兩條限制, 強制「P 與 $\neg P$ 恰有一個為真」:

$$x_P + x_{\neg P} \le 1 \qquad \text{且} \qquad x_P + x_{\neg P} \ge 1, \qquad \text{即} \quad x_P + x_{\neg P} = 1.$$

步驟 4(正確性). ($\Rightarrow$) 若 F 有滿足賦值 v, 令 $x_P = 1 \iff v(P) = \top$(且 $x_{\neg P} = 1 - x_P$), 則每條子句限制左邊至少為 1, 所有限制皆滿足。($\Leftarrow$) 反之, 任何整數解中 $x_P \in \{0,1\}$ 且 $x_P + x_{\neg P} = 1$, 故「$v(P) = \top \iff x_P = 1$」是良定義的賦值; 子句限制保證每個子句至少一個文字為真, 故 F 可滿足。

整個建構的大小與 F 成線性關係, 顯然多項式時間可完成。故 $\text{SAT} \le_p \text{IP}$。 □

備註 4.3. 由 Cook–Levin 定理,SAT 是 NP-complete, 故 IP 是 NP-hard。事實上其判定版本也屬於 NP(整數解可作為多項式大小的證據), 因此是 NP-complete; 特例 **0-1 整數規劃**更是 Karp(1972) 著名的 21 個 NP-complete 問題之一。對比之下 $\text{LP} \in \mathbf{P}$(Khachiyan)——「把解限制成整數」正是從 **P** 跨入 NP-hard 的關鍵一步。

5 線性鬆弛與分枝定界 (Branch-and-Bound)

既然 IP 是 NP-hard, 實務上怎麼解? 標準做法: 先解**線性鬆弛 (linear relaxation)**——即拿掉「解必須是整數」的要求, 用單純形法解對應的 LP——再對非整數的變數**分枝**。

Algorithm 1 BRANCH-AND-BOUND(LP)

```
求 LP 的線性鬆弛之解 a⃗(用單純形法)
if a⃗ ∈ ℤ^n then return a⃗
else
    選一個非整數分量 x_i = a_i 進行分枝
    a⃗_L ← BRANCH-AND-BOUND(LP ∪ {x_i ≤ ⌊a_i⌋})
    a⃗_R ← BRANCH-AND-BOUND(LP ∪ {x_i ≥ ⌈a_i⌉})
    if f(a⃗_L) ≥ f(a⃗_R) then return a⃗_L
    else return a⃗_R
```

5.1 完整範例

對前面的例子加上整數限制 $x, y \in \mathbb{Z}$:

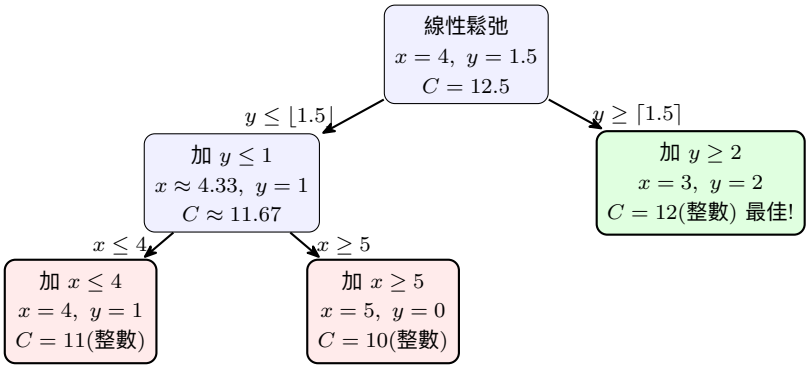
步驟 1. 解線性鬆弛 (即原 LP), 得 $x = 4,\ y = 1.5$,$C = 12.5$。y 非整數。

步驟 2. 對 $y = 1.5$ 分枝: 分別加入 $y \le \lfloor 1.5 \rfloor = 1$ 與 $y \ge \lceil 1.5 \rceil = 2$, 得兩個子問題。

步驟 3. 遞迴求解:

- **右枝** $(y \ge 2)$:LP 解為 $x = 3,\ y = 2$,$C = 12$——已是整數解!
- **左枝** $(y \le 1)$:LP 解為 $x = 13/3 \approx 4.33,\ y = 1$,$C \approx 11.67$。$x$ 非整數, 再分枝:
  - $x \le 4$:LP 解 $x = 4,\ y = 1$,$C = 11$(整數解);
  - $x \ge 5$:LP 解 $x = 5,\ y = 0$,$C = 10$(整數解)。

步驟 4. 比較兩枝的最佳整數解:$\max(12,\ 11,\ 10) = 12$。



最佳整數解

$$x = 3, \quad y = 2, \qquad C = 2(3) + 3(2) = 12.$$

注意: 它不是把 LP 最佳解 $(4, 1.5)$ 四捨五入得到的 $(4, 2)$ 或 $(4, 1)$——$(4, 2)$ 根本不可行 $(x + 2y = 8 > 7)$,$(4, 1)$ 的 $C = 11$ 並非最佳。

備註 5.1 (「定界」在哪裡?). 線性鬆弛的值是該子樹所有整數解的**上界** (bound)。在找到整數解 $C = 12$ 之後, 左枝的鬆弛上界 $\approx 11.67 < 12$, 因此整個左子樹其實可以直接剪掉 (pruning) 不必探索——這就是「Branch-and-**Bound**」中定界的威力。最壞情況分枝數仍是指數 (IP 是 NP-hard, 無可避免), 但良好的定界與剪枝使它在實務中 (CPLEX、Gurobi 等求解器) 非常有效。

6 線性規劃的其他變體

混合整數線性規劃 (MILP) 整數規劃與古典線性規劃的混合體: 部分變數要求是整數, 其餘變數可取非整數值。

0-1 整數規劃 (Zero-one IP) 整數規劃的限制版: 所有變數只能取 0 或 1。常用於「選或不選」的決策 (背包、指派、集合覆蓋); 它是 Karp 的 21 個 NP-complete 問題之一。

	變數值域	複雜度	代表演算法
LP	$\mathbb{R}_{\geq 0}$	$\in \mathbf{P}$(Khachiyan)	單純形法、內點法
MILP	部分 $\mathbb{Z}$、部分 $\mathbb{R}$	NP-hard	分枝定界 + 割平面
IP	$\mathbb{Z}_{\geq 0}$	NP-hard	分枝定界
0-1 IP	$\{0,1\}$	NP-complete(Karp)	分枝定界、SAT 求解器

7 本週重點整理

主題	重點
線性規劃 LP	線性目標 + 線性限制 $A\vec{x} \leq \vec{b}$; 可行區域是凸多面體; 最佳解在頂點
單純形法	鬆弛變數 $\rightarrow$ 表格 $\rightarrow$ 反覆樞軸 (最負係數行、最小正列商列) 直到無負係數; 幾何上沿邊走頂點
讀解	基變數讀最後一欄; 非基變數為 0; 鬆弛變數 $= 0 \iff$ 限制式是緊的
LP 複雜度	單純形法最壞指數 (Klee–Minty); 但 LP $\in \mathbf{P}$(Khachiyan 橢球法、Karmarkar 內點法)
整數規劃 IP	解限制在 $\mathbb{Z}_{\geq 0}$;NP-hard(SAT $\leq_p$ IP: 子句 $\rightarrow \sum \geq 1$,$x_P + x_{\neg P} = 1$)
分枝定界	解線性鬆弛; 非整數分量 $x_i = a_i$ 分枝成 $x_i \leq \lfloor a_i \rfloor$、$x_i \geq \lceil a_i \rceil$; 鬆弛值作上界剪枝
變體	MILP(部分整數)、0-1 IP(布林變數,Karp 21 問題之一)

8 練習題 (附解答)

練習 1. 把下列問題寫成標準形式 $A\vec{x} \leq \vec{b}$(最大化): 最小化 $3x - y$, 限制 $x + y \geq 2$、$x - 2y \leq 4$、$x, y \geq 0$。

解答

最小化 $3x - y$ 等價於最大化 $-3x + y$;$x + y \geq 2$ 乘 -1 得 $-x - y \leq -2$。故: 最大化 $-3x + y$, 限制

$$\begin{pmatrix} -1 & -1 \\ 1 & -2 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \leq \begin{pmatrix} -2 \\ 4 \end{pmatrix}, \qquad x, y \geq 0.$$

練習 2. 在單純形法中, 為什麼選樞軸列時要取「列商最小且分子分母皆正」的那一列?

解答

樞軸運算會把進入變數提高到「列商」的值。取**最小正列商**可保證所有其他限制式仍被滿足(最後一欄保持非負)——即新頂點仍在可行區域內。若分母 ≤ 0, 提高進入變數不會讓該限制式變緊, 不構成瓶頸, 故不考慮。

練習 3. 用最終表格說明: 為什麼 $s_1 = s_3 = 0$ 告訴我們最佳解位於哪兩條邊界的交點?

解答

$s_1 = 15 - 3x - 2y = 0$ 表示 $3x + 2y = 15$(第一條限制式取等號);$s_3 = 7 - x - 2y = 0$ 表示 $x + 2y = 7$。兩條邊界線的交點: 相減得 $2x = 8$,$x = 4$, 代回得 $y = 1.5$——正是讀出的最佳解 $(4, 1.5)$。

練習 4. 在 $\textsc{Sat} \leq_p \textsc{Ip}$ 的歸約中, 若只加 $x_P + x_{\neg P} \leq 1$ 而不加 ≥ 1, 歸約會失效嗎?

解答

會。少了 ≥ 1 時, 允許 $x_P = x_{\neg P} = 0$, 即「P 既不真也不假」。此時所有變數全設 0…會被子句限制 $\sum \geq 1$ 擋住, 但問題出在反方向: 整數解仍需對應到一個真正的賦值; 若某變數 $x_P = x_{\neg P} = 0$, 我們無法從解中讀出 P 的真值, 且子句可能靠其他文字滿足, 使 F 的可滿足性與 IP 可行性不再等價 (嚴格說, 可行解仍可導出可滿足賦值——任選 P 的值, 子句已由其他文字滿足——但習慣上加上等式使對應一一對應、驗證最簡潔)。標準作法是 $x_P + x_{\neg P} = 1$, 確保每個變數恰有一個真值。

練習 5. 對整數規劃: 最大化 $x + y$, 限制 $2x + 2y \leq 5$、$x, y \in \mathbb{Z}_{\geq 0}$, 執行分枝定界。

解答

線性鬆弛: 最佳在 $x + y = 2.5$(例如 $x = 2.5, y = 0$)。對 x 分枝: **左枝** $x \leq 2$: 鬆弛解 $x = 2$, $y = 0.5$, 仍非整數; 對 y 分枝得 $y \leq 0$:$(2, 0)$ 值 2;$y \geq 1$:$x \leq 1.5$…最終整數解如 $(1, 1)$ 值 2。**右枝** $x \geq 3$:$2(3) + 2y \leq 5$ 無非負解, 不可行 (剪枝)。最佳整數解值為 2(如 $(2, 0)$ 或 $(1, 1)$), 而鬆弛上界 2.5——樓上取整見「整數間隙」。

練習 6 (思考題)**.** 為什麼 Klee–Minty 的例子不與「$\textsc{Lp} \in \mathbf{P}$」矛盾?

解答

Klee–Minty 只證明特定演算法 (Dantzig 樞軸規則的單純形法) 最壞情況是指數, 並非證明問題本身難。Khachiyan 的橢球法是完全不同的演算法, 能在多項式時間解 LP。「問題的複雜度」是所有演算法中最好的那一個, 單一演算法的最壞情況只給出上界的失敗例。

參考資料

1. C. Hampson, *5CCS2FC2 Foundations of Computing II, Week 8* 投影片 (linearprog / simplex / integerprog),King's College London.
2. G. B. Dantzig, *Linear Programming and Extensions*, Princeton University Press, 1963.
3. V. Klee, G. J. Minty, "How good is the simplex algorithm?" in *Inequalities III*, Academic Press, 1972.

4. L. G. Khachiyan, “A polynomial algorithm in linear programming,” *Soviet Math. Dokl.*, 20:191–194, 1979.

5. N. Karmarkar, “A new polynomial-time algorithm for linear programming,” *Combinatorica*, 4:373–395, 1984.

6. R. M. Karp, “Reducibility among combinatorial problems,” in *Complexity of Computer Computations*, 1972.(0-1 IP 是 21 個 NP-complete 問題之一)

7. T. H. Cormen, C. E. Leiserson, R. L. Rivest, C. Stein, *Introduction to Algorithms*, MIT Press.(線性規劃章節)

8. Wikipedia: *Linear programming*; *Simplex algorithm*; *Integer programming*; *Branch and bound*; *Klee–Minty cube*.