

第9回 整数計画法I

— 導入と欲張り法 —

9.1 整数計画法とは

整数計画法とは、変数が整数のみをとるという条件で最適化を行う数理計画法である。また、整数に限らず、解ベクトルの各要素が有限個の値に限定されているような数理計画問題を、組合せ最適化問題という。したがって、整数計画問題は組合せ最適化問題の一種である。また、前章で扱った最短経路問題では、 x_i は0か1（その枝を選ぶか選ばないか）に制限されているので、組合せ最適化問題である。

組合せ最適化問題では、実行可能解の数は有限であるから、すべての実行可能解について目的関数の値を計算することによって最適解を得ることができる。しかし、変数の数が多くなると、解の数が爆発的に増える。たとえば、 n 変数が0,1のみを取る場合を考えても、解の個数は _____ あり、 $n = 10, 30, 50$ のときにはそれぞれ _____, _____, _____ となる。したがって、しらみつぶしに目的関数を計算する方法は、 n が大きくなると現実的ではない。

そこで、問題の性質や構造を利用して、効率的なアルゴリズムを作る必要がある。この章では、以下の方法について、問題例をあげた実行例を示しながら説明する。

- 欲張り法 …… その場その場で最良であるものを取り入れていくという、単純な考え方で最適解を求めようとする方法の総称。欲張り法で最適解が求まる問題と、求まらない問題がある。
- 分枝限定法 …… しらみつぶしに実行可能解について調べるために場合分けを行う際に、不必要な場合分けを省略して計算量を削減する方法。
- 動的計画法 …… 全体の問題を求めるために、小さい問題の最適解を求め、次第に拡張していく方法の総称。
- 近似解法 …… 最適解を求めるのはとりあえずあきらめ、最適解に近い解を探索する解法。必要に応じてさらにより近似最適解を探索することもある。

9.2 欲張り法

その場その場で最良であるものを取り入れていくという、単純な考え方で最適解を求める方法の総称。したがって、最適解が求まる問題は限られている。

- 最適解が必ず求まる問題例：最小木問題
- 最適解が求まらない問題例：ナップザック問題

ここで、うまくいく例として、最小木問題を例にして欲張り法を説明する。ただし、ここでは無向グラフ（ネットワーク）をあつかう。

[問題 9.1] 図 9.1 に示すネットワークに対して、最小木を求めよ。

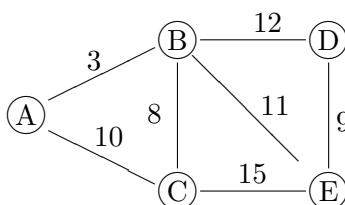


図 9.1: 最小木を求めるネットワーク

木の長さ: 木に含まれる全ての枝の長さの和。

最小木: 長さが最小である木（スパニングツリー）。

最小木問題に対する欲張り法は、_____ と呼ばれている。具体的には以下のとおり。ただし、各枝の長さを a_i とする。

—— クラスカル法 ——

1. まず、全ての枝 $e_i \in E$ を長さの短い順でソートする。したがって、

$$a_1 \leq a_2 \leq \cdots \leq a_m$$

となる。 $T = \{e_1\}, k = 2$ とする。

2. $T \cup \{e_k\}$ が閉路を含まないならば、 $T \leftarrow T \cup \{e_k\}$ とする。閉路を含んでしまうときは T はそのまま。
3. T がスパニングツリーになっていれば終了。そうでなければ、 $k \leftarrow k + 1$ として、2. に戻る。

スパニングツリーは必ず _____ 本の枝からなるので, 3. の判定は集合 T に含まれる枝数を数えればよい.

問題 9.1 をクラスカル法を用いて解いてみよう. まず, 枝は

$$E = \{(A, B), (A, C), (B, C), (B, D), (B, E), (C, E), (D, E)\}$$

の 7 本. 長さの順に並べると,

$$\{e_1, \dots, e_7\} = \underline{\hspace{10cm}}$$

となる. あとは図 9.2 の (a)~(e) のとおり. 長さにアンダーラインのついた枝は e_k として注目されている枝を示し, 太線は T に含まれる枝を示す. 図の (d) では $e_4 = (A, C)$ に注目しているが, これを T に加えると閉路ができてしまうので, 何もしていない. 図の (e) で T に $4 (= n - 1)$ 本の枝が入ったので終了している.

定理 9.1 クラスカル法によって必ず最小木が得られる.

全域森: ある与えられた (必ずしも連結していない) グラフにおいて, どの枝を加えても閉路ができてしまう枝集合を**全域森**という. 1 つのグラフにおいて全域森に属する枝の数は, 一定である. (n' 個の連結グラフからなるグラフでは, 全域森の枝数は _____ 本.)

証明: (定理 4.1) クラスカル法で得られる枝の集合 T は, 閉路を持たず, 枝の本数は $n - 1$ なので, T は _____ である. T 内の枝を T に組み込んだ順に並べたものを $\{e_{l_1}, e_{l_2}, \dots, e_{l_{n-1}}\}$ とすると, それらは _____ 順になっている. つまり,

$$a_{l_1} \leq a_{l_2} \leq \dots \leq a_{l_{n-1}} \quad (9.1)$$

となっている.

さて, 入力されたネットワーク $G = (V, E)$ に対する任意のスパニングツリー T^* を考える. T^* 内の枝を T と同様に長さの短い順に並べたものを $\{e_{q_1}, e_{q_2}, \dots, e_{q_{n-1}}\}$ とする. つまり,

$$a_{q_1} \leq a_{q_2} \leq \dots \leq a_{q_{n-1}} \quad (9.2)$$

となっている. さて, ある r ($1 < r \leq n - 1$) が存在して,

$$\begin{array}{ccc} a_{l_1} & \leq & a_{q_1} \\ & \vdots & \\ a_{l_{r-1}} & \leq & a_{q_{r-1}} \\ a_{l_r} & > & a_{q_r} \end{array}$$

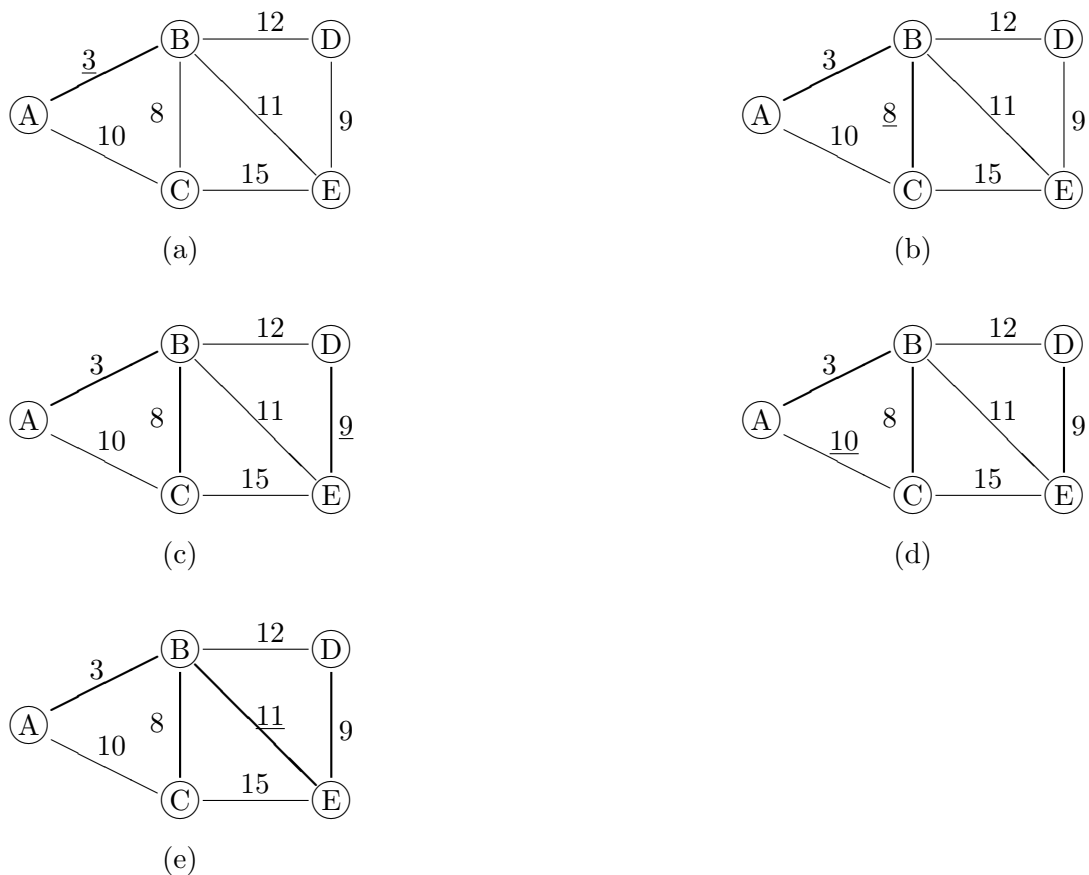


図 9.2: クラスカル法

と書けるとしよう. (e_{l_1} は最も短い枝であるから, $r = 1$ となることはない.) ここで, 枝の集合 $T' = \{e_{l_1}, \dots, e_{l_{r-1}}, e_{q_1}, \dots, e_{q_r}\}$ からなるグラフ G' を考える. (これは, G の部分グラフになっていて, 連結しているとは限らない. また, T' には, 重複した枝が存在するかもしれない.) 式 (9.1)(9.2) より, T' 内の枝は全て a_{l_r} より _____. したがって, クラスカル法において, 枝 e_{l_r} が T として選ばれる前に, T' 内の枝はすでに注目されているので, $r-1$ 本の枝 $\{e_{l_1}, \dots, e_{l_{r-1}}\} \subset T$ は, グラフ G' における _____ になっている. (そうでなければ, T に加えても閉路ができないような枝が T^* に存在するはずで, クラスカル法において e_{l_r} が選ばれる前にその枝が選ばれているはずである.) グラフ G' における全域森の枝数は _____ である. したがって, r 本の枝 $\{e_{q_1}, \dots, e_{q_r}\} \subset T^*$ は閉路を含んでいるはずであり, これは T^* が木であることに矛盾する.

以上より, 全ての i ($1 \leq i \leq n-1$) に対して

$$a_{l_i} \leq a_{q_i}$$

が成り立っている。すなわち、 T の長さは任意の T^* の長さより短かく、 T は最小木である。 Q.E.D.

最小木問題では、欲張り法、つまり、最小化をするために短い枝から順に取り入れていく方法によって最適解を求めることができたが、一般には最適解を求めることはできない。しかし、欲張り法は考え方が簡単で計算量も少ないため、他の手法とあわせて用いられることが多い。次では、欲張り法では上手くいかない例をあげよう。

[問題 9.2] [ナップザック問題] n 個のアイテムは、それぞれ重さが a_i であり、利用価値が c_i である。ナップザックには重さ b までしか入れられないとき、利用価値の総和が最も大きくなるようにするには、どのアイテムを入れればよいか。

この問題は、 i 番目のアイテムをナップザックに入れるときは $x_i = 1$ 、そうでなければ $x_i = 0$ として、次のように定式化できる。

$$\begin{aligned} \text{目的関数} &: \sum_{i=1}^n \text{ } \rightarrow \text{最大化} \\ \text{制約条件} &: \sum_{i=1}^n \text{ } \leq b \\ & x_i \in \{0, 1\} \end{aligned}$$

ここで、アイテムは「単位重さあたりの利用価値」の大きいもの順に並べであるとしよう。つまり、

$$\frac{c_1}{a_1} \geq \frac{c_2}{a_2} \geq \dots \geq \frac{c_n}{a_n}$$

とする。この問題を欲張り法で解くには、
から順に、重さの制限が許す限り入れていけば良い。たとえば、下記のように定式化された問題：

$$\begin{aligned} \text{目的関数} &: 7x_1 + 8x_2 + x_3 + 2x_4 \rightarrow \text{最大化} \\ \text{制約条件} &: 4x_1 + 5x_2 + x_3 + 3x_4 \leq 6 \\ & x_i \in \{0, 1\} \end{aligned}$$

の場合、 $x_1 = 1, x_2 = 0$, $x_3 = 1, x_4 = 0$ と決定していき、最終的に実行可能解 $\mathbf{x} = (1, 0, 1, 0)$ を得る。このとき、利用価値の総和は である。しかし、しらみつぶしに解を調べることにより、 $(0, 1, 1, 0)$ が最適解（目的関数の値は ）であることがわかる。