The First Course in Combinatorial Optimization

岩間研 M1 市場孝之

2008/01/28

概要

8 章ではサブモジュラ関数を扱う．マトロイドについてのいくつかの重要な問題が，あるサブモジュラ関数の最小化・最大化で解ける．したがって一般のサブモジュラ関数の最小化・最大化問題を解くアルゴリズムは，基本的で有用なアルゴリズムである．

8 Optimizing Submodular Functions

Definition. 関数 $f : 2^E \mapsto \mathbb{R}$ が次の不等式を満たすとき，サブモジュラ関数という．

$$^\forall A, B \subset E,\ f(A \cup B) + f(A \cap B) \leq f(A) + f(B)$$

Theorem. マトロイド M 上のランク関数 r_M はサブモジュラ関数である．

証明は 1.5 節.

8.1 Minimizing Submodular Functions

マトロイドにおけるいくつかの最適化問題は，サブモジュラ関数の最小化問題を解くことで解決することができる．いくつかの例を次に示す．

Separation problem for the rank inequalities

マトロイド M について，M のランク不等式はマトロイド多面体 $\mathcal{P}_{I(M)}(I(M)$ を端点とする多面体) を表す．

$\sum_{e \in S} x_e \leq r_M(S),\ \forall S \subset E(M).$

このランク不等式の Separation Problem は，ある $x^* \in \mathbb{R}^{E(M)}$ が $\mathcal{P}_{I(M)}$ に含まれるか否かを判定する問題である．

次のような関数 $f : 2^{E(M)} \mapsto \mathbb{R}$ を考える．

$f(S) := r_M(S) - \sum_{e \in S} x^*_e.$

$f(S)$ の $S \subset E(M)$ 上での最小値が 0 未満である時，またその時に限って $x^* \in \mathcal{P}_{I(M)}$ が成り立つ．

$f(S)$ がサブモジュラ性を持つことは，$r_M(S)$, $\sum_{e \in S} x^*_e$ の両方がサブモジュラ性を持つことから明らかである．

したがって，このランク不等式の Separation Problem はサブモジュラ関数 $f(S)$ の最小化問題に変換することができ，

Minimum-weight cuts
有向グラフ G と G 上の異なる 2 点 $v, w \in V(G)$，枝の容量 $c : E(G) \mapsto \mathbb{R}_+$ が与えられている．
$S \in V(G) \setminus \{v, w\}$ について，次のような関数 f を考える．
$f(S) := \sum\{c(e)\ :\ e \in \delta_G^+(S+v)\}$.
すなわち $f(S)$ は $S + v$ へ入る枝の容量の総和であり，
$f(S)$ の最小化問題は v, w を分ける最小カット問題と等価である．

$f(S)$ はサブモジュラ性を持つ．

Maximum-cardinality matroid intersection
E を共通の台集合とするマトロイド M_1, M_2 が与えられている ($E(M_1) = E(M_2) = E$).
次のような関数 f を考える．
$f(S) := r_{M_1}(S) + r_{M_2}(E \setminus S)$.
$f(S)$ の最小化問題は．M_1, M_2 の最大共通独立集合の発見問題を解決する．

$f(S)$ はサブモジュラ性を持つ．(証明略)

次に，一般のサブモジュラ関数 $f : 2^E \mapsto \mathbb{R}$ の最小化問題を考える．(ここでは $f(\emptyset) = 0$ であるとしているが，そうでなければ $f(S) \leftarrow f(S) - f(\emptyset)$ と再定義すればよい)．f を $[0,1]^E$ 上に拡張した関数 f' を次のように定義する．まず，$x \in \{0,1\}^E$ に対しては，$f' : (x) := f(S(x))$ とする ($S(x) := \{e \in E : x_e \neq 0\}$)，そうでない $x \in [0,1]^E$ については，まず次のように x を"分解"する．

$$x = \sum_{j=1}^{m} \lambda_j x^j, \quad \text{where} \quad m \leq |E|; \tag{1}$$
$$\lambda_j > 0,\ \text{for } j = 1, 2, ..., m; \tag{2}$$
$$x^j \in \{0,1\}^E,\ \text{for } j = 1, 2, ..., m; \tag{3}$$
$$x^1 \geq x^2 \geq \cdots x^m \neq 0. \tag{4}$$

条件を満たすような m, λ_j, x^j は一意に決まるので，$f'(x)$ は次のように定義する．

$$f'(x) := \sum_{j=1}^{m} \lambda_j f(S(x^j)).$$

Theorem. 関数 f' は凸であり，$[0,1]^E$ 上での最小値を $\{0,1\}^E$ 上に持つ．

■証明．まず f' が凸であることを示す．ある点 $x^* \in \mathbb{R}_+^E$ に対して次の LP を考える．

$$\hat{f}(x^*) := \max \sum_{e \in E} x_e^* z_e$$
$$\text{Subject to:}$$
$$\sum_{e \in T} z_e \leq f(T), \quad {}^\forall T \subset E.$$

LP の目的関数の最大値は，目的関数の係数ベクトル (x^*) についての凸関数である．したがって $f' = \hat{f}$ を示せばよい．

一般性を失うことなく，$E := \{1, 2, ..., n\}$ について $x_1^* \geq x_2^* \geq \cdots x_n^*$ とおける．また，$T_j := \{1, 2, ..., j\}$

とする．このとき

$$\hat{f}(x^*) = \sum_{j=1}^{n} x_j^* \Big[f(T_j) - f(T_{j-1}) \Big]$$
$$= \sum_{j=1}^{n} (x_j^* - x_{j+1}^*) f(T_j)$$

が成り立つ (ただし $x_{n+1}^* := 0$)．したがって $\lambda_j := x_j^* - x_{j+1}^*$ とすれば，$x^* := \sum_{j=1}^{n} \lambda_j x(T_j)$ と"分解"できる ($\lambda_j = 0$ となる j を無視すれば先述の条件も満たす)． □

次に f' が $[0,1]^E$ 上での最小値を $\{0,1\}^E$ 上に持つことを示す．$x^* = \sum_{j=1}^{m} \lambda_j x^j$ を $[0,1]^E$ 上で f' が最小になる点とする．$f'(0) = f(\emptyset) = 0$ であるから，$f'(x^*) \leq 0$．$f'(x^*) = 0$ であれば，f' は $x = 0$ でも最小値をとる．$f'(x^*) < 0$ のとき，f' は $x^1, ..., x^m$ のいずれかで最小になることを背理法で示す．任意の j について $f'(x^j) > f'(x^*)$ であるとすると，$f'(x^*) = \sum_{j=1}^{m} \lambda_j f'(x^j) > \sum_{j=1}^{m} \lambda_j f'(x^*)$．$f'(x^*) < 0$ であるから，$1 < \sum_{j=1}^{m} \lambda_j$．しかし，$x^* \in [0,1]^E$ から $\sum_{j=1}^{m} \lambda_j \leq 1$ であるから，矛盾．したがって f' はある x^j において最小となる． □

凸関数 f' の最小化問題は，楕円体法を用いて多項式時間で解けるため，サブモジュラ関数 f の最小化問題も多項式時間で解ける．

8.2 Minimizing Submodular Functions Over Odd Sets

本節では次のような問題を考える．台集合 E と E の部分集合 T が与えられている．T との共通集合の要素が奇数個であるような $S \subset E$ の中で，サブモジュラ関数 f を最小化したい．

$$\text{Solve } S^* := \text{argmin}\{ f(S) \ : \ S \subset E, \ |S \cap T| \text{ is odd} \}.$$

■適用例 最大重みマッチング

グラフ H と，枝の重み関数 c が与えられている．H における最大重みマッチング S^* を求めたい．

まず，S^* には負の重みの枝は含まれないので，c は非負であるとみなせる．次に H のコピー H' を作り，H と H' の対応する節点間に枝を張り，新しくグラフ G とする．ただし，H' 内の枝，H と H' 間に張った枝の重みは 0 であるとする，

このとき次のようにして，任意の H のマッチング S は，G の完全マッチング M に拡張できる．S に含まれる任意の枝 $e \in S \subset E(H)$ と，e に対応する枝 $e' \in E(H')$ を M に入れる．加えて，S に含まれない節点 $i \in V(H)$ と，対応する節点 $i \in V(H')$ 間に張られた枝を M に入れる．こうして作られた M は G の完全マッチングであり，その重みは S の重みと等しい．

逆に，任意の G の完全マッチング M は，H のマッチング $S := M \cap E(H')$ に変換でき，S の重みは M の重みに等しい．したがって，H の最大重みマッチング S を見つけることは，G の最大重み完全マッチング M を見つけることと同値である．

ここで任意のグラフ G と枝の重み c(非負) が与えられているとする．$\bar{\mathcal{M}}(G)$ を G の完全マッチングの集合とする．完全マッチング多面体 $\mathcal{P}_{\bar{\mathcal{M}}(G)}$ は，以下の不等式の解集合である．

$$-x_e \leq 0, \quad {}^{\forall} e \in E(G); \tag{1}$$

$$\sum_{e \in \delta_G(v)} x_e = 1, \quad {}^{\forall} \nu \in V(G); \tag{2}$$

$$\sum_{e \in E(G[W])} x_e \leq \frac{|W|-1}{2}, \quad {}^{\forall} W \subset V(G) \text{ with } |W| \geq 3 \text{ odd.} \tag{3}$$

式 (3) は,(2) を使えば次の式に置き換えられる．

$$(3') \sum_{e \in \delta_G(W)} x_e \geq 1, \quad {}^{\forall} W \subset V(G) \text{ with } |W| \text{ odd;}$$

何故ならば,

$$\begin{array}{c} 2\Big(\displaystyle\sum_{E \in E(G[W])} x_e \leq \frac{|W|-1}{2} \Big) \\ -\displaystyle\sum_{\nu \in W} \Big(\sum_{e \in \delta_G(\nu)} x_e = 1 \Big) \\ \hline -\displaystyle\sum_{e \in \delta_G(W)} x_e \leq -1. \end{array}$$

(3') の separation problem は次の関数 $f(W)$ の最小化によって解ける．

$$f(W) := -1 + \sum_{e \in \delta_G(W)} x_e^*$$

ただし最小化は要素数が奇数である任意の $W \subset V(G)$ について行う．1.5 節で既に示したように，f はサブモジュラ関数である．したがってこの問題は，サブモジュラ関数を要素数が奇数である部分集合族上で最小化することで解ける．

■アルゴリズム　一般のサブモジュラ関数 f について,

$$\text{Solve } X^* := \operatorname{argmin}\{f(X) \ : \ X \subset E, \ |X \cap T| \text{ is odd}\}.$$

を解くアルゴリズムを考える．

Step 1:　$|T|$ が奇数である場合はこの STEP で $|T|$ が偶数の場合に変換する．

X^* は T の要素を全て含むか，もしくは T の要素のいずれかを含まない．これを踏まえて，次の計算をする．

$$X_T := \operatorname{argmin}\{f(X) \ : \ X \subset E, \ T \subset X\}, \tag{4}$$

$$= \operatorname{argmin}\{f(Y \cup T) \ : \ Y \subset E \setminus T\} \tag{5}$$

また，全ての $e \in T$ について次の計算をする．

$$X_e := \operatorname{argmin}\{f(X) \ : \ X \subset E - e, \ |X \cap (T - e)| \text{ is odd}\}. \tag{6}$$

(5) は 8.1 節で構築した最小化アルゴリズムで解ける．(6) は X^* の計算と変わりないが，$|T - e|$ が偶数であるので，次の STEP 以降で紹介するアルゴリズムで解ける．

STEP 2: T が偶数であるとする．このとき次の最小化問題が解けたとする．

$$U := \mathrm{argmin}\{f(X)\ :\ X \subset E,\ X \text{ splits } T\} \tag{7}$$

ただし，"X splits T(X が T を分割する)" とは，$X \cap T$ と $T \setminus X$ のどちらもが空集合でないことを指す．$|T|$ は偶数であるから，$|X \cap T|$ が奇数であれば X は T を分割する．したがって，$|U \cap T|$ が奇数であれば，そもそもの問題は $X^* = U$ という解を持つことになる．

次に U の計算方法について述べる．$\forall e, f \in T(e \neq f)$ について，次の最小化問題を解く．

$$\begin{aligned} U_{e,f} :&= \mathrm{argmin}\{f(X)\ :\ X \subset E - f,\ e \in X\} && (8) \\ &= \mathrm{argmin}\{f(Y + e)\ :\ Y \subset E - f\} && (9) \end{aligned}$$

(テキストでは (8) に X odd という記述があるが，誤りだと思われる) (9) は 8.1 節で構築した最小化アルゴリズムで解けるので，全ての e, f の組合せについてこれを解けば次の式で U が求められる．

$$U := \mathrm{argmin}\{f(U_{e,f})\ :\ e, f \in T\}. \tag{10}$$

STEP 3: $|U \cap T|$ が偶数であるとする．この場合は以下のようにして X^* を解くことができる (正しさの証明は後述)．

$$\begin{aligned} X^* :&= \mathrm{argmin}\{f(U_1),\ f(U_2)\}, && (11) \\ U_1 :&= \mathrm{argmin}\{f(X)\ :\ X \subset E,\ |X \cap (T \cap U)| \text{ odd},\ X \text{ does not split } T \setminus U\}, && (12) \\ U_2 :&= \mathrm{argmin}\{f(X)\ :\ X \subset E,\ |X \cap (T \setminus U)| \text{ odd},\ X \text{ does not split } T \cap U\}. && (13) \end{aligned}$$

ただし，U_1 は $T \setminus U$ を一つの要素とみなすことで，再帰的手続きで解くことができる (U_2 も同様)．また，$|T \cap U|$ が偶数であるから，STEP2 からの繰り返しとなる．更に，T を二つに分割して再帰的に解くので，再帰回数は高々 $|T|$ 回である．

Theorem.

$|U \cap T|$ が偶数であれば，$X^* = \mathit{argmin}\{f(U_1),\ f(U_2)\}$ が $\mathit{min}\{f(X)\ :\ X \subset E,\ |X \cap T| \mathit{odd}\}$ の解である．

証明 (背理法)． $X^* := \mathrm{argmin}\{f(X)\ :\ X \subset E,\ |X \cap T| \mathrm{odd}\}$ の解 X^* について，$f(X^*) < f(U_1)$ かつ $f(X^*) < f(U_2)$ であると仮定する．U_1 の定義から，X^*は $T \setminus U$ を分割する．したがって $X^* \cup U$ は T を分割する．同様に，U_2 の定義から，X^*は $T \cap U$ を分割する．したがって，$X^* \cap U$ は T を分割する．

ここで，$|T \cap X^*|$ は奇数，$|T \cap U|$ は偶数であるから，$|T \cap (X^* \cup U)|$ か $|T \cap (X^* \cap U)|$ のいずれか一つだけが奇数である．$|T \cap (X^* \cup U)|$ が奇数であるとすると，X^*および U の定義より，

$$f(X^* \cap U) \geq f(X^*).$$
$$f(X^* \cup U) \geq f(U).$$

しかし f はサブモジュラ関数であるから，

$$f(X^* \cap U) = f(X^*).$$
$$f(X^* \cup U) = f(U).$$

である．