LA-004

ラミナー被覆制約を持つ単調凹関数最小化問題

Minimizing a Monotone Concave Function with Laminar Covering Constraints

坂下　麻里子* Mariko Sakashita　　牧野　和久* Kazuhisa Makino　　藤重　悟† Satoru Fujishige

1　序論

V を $|V| = n$ である有限の集合とする．V の部分集合族 $\mathcal{F} \subseteq 2^V$ は，任意の集合 $X, Y \in \mathcal{F}$ に対して，

$$X \cap Y = \emptyset,\ X \subseteq Y,\ X \supseteq Y$$

のいずれかが成立するとき，ラミナー (laminar) 族と呼ばれる．本研究では，ラミナー族 $\mathcal{F} \subseteq 2^V$，ラミナー族 $\mathcal{F}$ 上の要求関数 $d : \mathcal{F} \to \mathbf{R}_+$，コストを表す単調凹関数 $F : \mathbf{R}_+^V \to \mathbf{R}_+$ が与えられたとき，以下に記述されるラミナー被覆制約を持つ凹関数最小化問題を扱う．

$$\begin{array}{llll} (\mathbf{P}) & \text{Minimize} & F(x) & \\ & \text{subject to} & x(X) \geq d(X) & (X \in \mathcal{F}) \\ & & x(v) \geq 0 & (v \in V) \end{array}$$

ただし，$\mathbf{R}_+$ は非負実数の集合，$x(X) = \sum_{v \in X} x(v)$ とする．

この問題 (**P**) に対しては，会社などの組織の階層構造がラミナー族によって表現できることから分かるように，様々な応用が考えられる．また，一見するとラミナー構造を持たない問題でも問題 (**P**) として定式化可能なものがある．例えば，需要量が一定である無向ネットワークにおけるフローに基づく施設配置問題である (詳しい定義は [1] 参照)．本研究では，コストが従来用いられていた各点での施設建設費用の和として記述できる場合 [1, 3, 4] だけではなく，各点での供給量にも単調凹に依存する一般的な場合も扱っている．

本論文での成果

本研究ではコスト関数 F が一般の単調凹関数である場合に加えて，以下のように記述できる場合について考察する．

$$F_1(x) = \sum_{X \in \mathcal{F}} f_{\Delta X}(x[\Delta X]). \tag{1}$$

$$F_2(x) = \sum_{v \in V} f_v(x(v)). \tag{2}$$

$$F_3(x) = \sum_{v \in V : x(v) > 0} (a_v x(v) + b_v). \tag{3}$$

ただし，$\Delta X = X - \bigcup_{Y \in \mathcal{F}: Y \subset X} Y$，$f_{\Delta X}$ は ΔX 上の非負単調凹関数，$x[\Delta X]$ は変数 x の ΔX 上への制限，f_v は 1 変数の非負単調凹関数，a_v, b_v は非負定数とする．定義から明らかに，F_1 の特殊形が F_2 であり，さらに F_2 の特殊形が F_3 である．

定理 1． コスト関数 F が式 (1) で与えられるとき，問題 (**P**) は $\mathrm{O}(n^2 q)$ 時間で解くことができる．

定理 2． コスト関数 F が式 (2) で記述できる場合でも F がオラクルで与えられるならば，問題 (**P**) を解くために $\Omega(n^2 q)$ 時間必要である．

ここで F のオラクルとは，任意の $x \in \mathbf{R}_+^V$ に対して，その関数値 $F(x)$ を求めるものであり，F がオラクルで与えられるかどうかに関わらず，$x \in \mathbf{R}_+^V$ から $F(x)$ を得るための計算量を q とする．

定理 1 , 2 から式 (1) で記述されるコスト関数 F がオラクルで与えられるときは，定理 1 に示されるアルゴリズムが最適となる．

また，定理 1 より，序論で述べた施設配置問題も効率的に解ける．

系 1． 需要量が一定でかつ，コスト関数が式 (1) で表される無向ネットワーク $\mathcal{N}$ におけるフローに基づく施設配置問題は，$\mathrm{O}(n(m + n\log n + nq))$ 時間で解くことができる．ただし，n, m はそれぞれネットワーク $\mathcal{N}$ の点数と枝数を示す．

略証: 最大フロー最小カットの定理より，ネットワークの極集合族 $\mathcal{F} \subseteq 2^V$ を用いて施設配置問題が問題 (**P**) として定式化可能である (ここで $\mathcal{F}$ がラミナー族になることに注意)．極集合族は [2] より $\mathrm{O}(n(m + n\log n))$ 時間で求めることができる．従って定理 1 より，系 1 が示された．

F が式 (3) で与えられるときには，以下のように高速に解ける．

定理 3． F が式 (3) で表されるとき，問題 (**P**) は $\mathrm{O}(n \log^2 n)$ 時間で解くことができる．また F がオラクルとして与えられる場合でも，$\mathrm{O}(n(\log^2 n + q))$ 時間で解くことができる．

さらに，目的関数 F が一般の単調凹関数のときは，次の否定的な結果を得る．

定理 4． 目的関数 F が一般の単調凹関数のとき，

(I) F がオラクルとして与えられるとき，問題 (**P**) を解くためには $\Omega(2^{\frac{n}{2}} q)$ 時間必要である．

(II) F が明示的に与えられるときでも，問題 (**P**) は NP 困難である．

以下，まとめた結果を表 1 に示す．

表 1: 本論文で得られた成果

	F_1	F_2	F_3	一般
明示的	$\mathrm{O}(n^2 q)$	$\mathrm{O}(n^2 q)$	$\mathrm{O}(n \log^2 n)$	NP 困難
オラクル	$\Theta(n^2 q)$	$\Theta(n^2 q)$	$\mathrm{O}(n(\log^2 n + q))$	$\Omega(2^{\frac{n}{2}} q)$

本論文では，頁数制限のため，定理 1 と定理 2 の略証，および，定理 3 のアイデアのみを記す．

*大阪大学大学院 基礎工学研究科　Graduate School of Engineering Science, Osaka University.
†京都大学 数理解析研究所　Research Institute for Mathematical Sciences, Kyoto University.

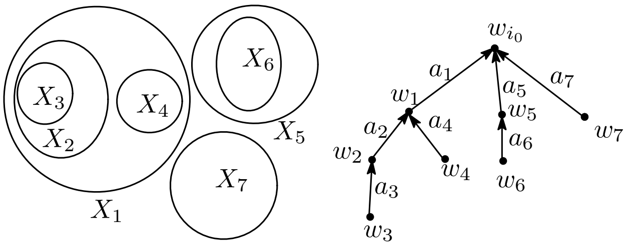
2　定理１の略証

ラミナー族 $\mathcal{F}$ は，その構造から木表現 $T=(W,A)$ を持つ．ただし，点集合 W は $X_i \in \mathcal{F}$ に対応する点 w_i と根となる点 w_{i_0} から成り，枝集合 A は以下の (i)(ii) の枝から成る．

(i) $(w_i, w_j) \in A : X_i \subset X_j$, かつ, $X_i \subset Y \subset X_j$ である $Y \in \mathcal{F}$ が存在しないとき

(ii) $(w_i, w_{i_0}) \in A : X_i$ が $\mathcal{F}$ において極大集合のとき

図 1 にラミナー族 $\mathcal{F}$ に対する木表現 T の例を示す．



$\mathcal{F}=\{X_1, X_2, \cdots, X_7\}$　　$\mathcal{F}$ の木表現 $T=(W,A)$

図 1: ラミナー構造の木表現

この木表現 T に基づき、集合 $X_i \in \mathcal{F}$ に対して $\mathcal{S}(X_i) = \{X_j \mid (w_j, w_i) \in A\}$ を X_iの子集合族, $\Delta X_i = X_i - \bigcup_{(w_j,w_i)\in A} X_j$を差分 と呼ぶ．また，関数 $d : \mathcal{F} \to \mathbf{R}$ に対して，差分 Δd を $\Delta d(X) = d(X) - \sum_{Y\in\mathcal{S}(X)} d(Y)$ により定義する．

以下では一般性を失うことなしに，我々の問題 $(\mathbf{P})$ において $F(0)=0$, $\Delta d(X) > 0\,(X \in \mathcal{F})$, $V = \bigcup_{X\in\mathcal{F}} X$ と仮定する．

本節では，ラミナー被覆制約を持つ凹関数最小化問題のうち，目的関数 F が式 (1) で表現できる場合，すなわち

$$F(x) = \sum_{X\in\mathcal{F}} f_{\Delta X}(x[\Delta X])$$

で与えられる場合について考察する．

このとき，次のような構造的性質を持つ最適解が存在することが分かる．

補題１． F が式 (1) のように分離可能なとき，次の条件 (I), (II) を満たす $W - \{w_{i_0}\}$ の分割 $\mathcal{P} = \{P_1, \cdots, P_k\}$ とその分割に基づく最適解 x^* が存在する．

(I) $P_j \in \mathcal{P}$ は $\mathcal{F}$ の木表現 $T=(W,A)$ において有向パス $w_{j_0} \to w_{j_1} \to \cdots \to w_{j_{r_j}}$ となる．また，$\Delta X_{j_0} \neq \emptyset$.

(II) 各 $X_{j_0} (P_j \in \mathcal{P})$ に対して, ある $v_j \in \Delta X_{j_0}$ が存在して,

$$x^*(t) = \begin{cases} \displaystyle\sum_{w_i\in P_j} \Delta d(X_i) & (t = v_j,\ P_j \in \mathcal{P} \text{ のとき}) \\ 0 & (t \in V - \bigcup_{P_j\in\mathcal{P}}\{v_j\} \text{ のとき}) \end{cases}$$

以下では，補題１に示す構造的性質を持つ最適解を求めるアルゴリズムを構成する．

任意の $X \in \mathcal{F}$ に対して，対応する $w \in W$ から根 w_{i_0} までの有向パスを $w_{j_0} (= w)$, $w_{j_1}, \cdots, w_{j_{h(X)-1}}$, $w_{j_{h(X)}} (= w_{i_0})$ とする．この $X \in \mathcal{F}$ と整数 $k\,(0 \le k \le h(X)-1)$ に対して，以下のように定義される問題を考える．

$$\begin{array}{lll} (\mathbf{P}') & \text{Minimize} & \displaystyle\sum_{Y\in\mathcal{F}:Y\subseteq X} f_{\Delta Y}(x[\Delta Y]) \\ & \text{subject to} & x(X) \ge d(X) + \displaystyle\sum_{i=1}^{k} \Delta d(X_{j_i}) \\ & & x(Y) \ge d(Y) \qquad (Y \in \mathcal{F},\ Y \subseteq X) \\ & & x(v) \ge 0 \qquad (v \in V) \end{array}$$

問題 $(\mathbf{P}')$ に補題１を用いることで，この問題が $\{w_i \mid X_i \in \mathcal{F},\ X_i \subseteq X\}$ の分割 $\mathcal{P}$ とそれに基づく最適解を持つことが分かる．今，X に対応する W の点 w を含む集合を $P_j \in \mathcal{P}$ とする．ただし，$h(X)$ は木表現 T における w の深さを示す．$X \in \mathcal{F}$ に対するテーブル g_X は $\{0, 1, \cdots, h(X)-1\}$ から $\mathbf{R}_+ \times V$ への関数であり，$g_X(k)_1$ は $(\mathbf{P}')$ の最適値, $g_X(k)_2$ は上記の集合 $P_j \subseteq W$ に対する（補題１ (II) 中の）$v_j \in \Delta X_{j_0}$ を与えるものとする．以下ではこのテーブルの作成法を考える．

T の葉に対応する（すなわち極小な）$X \in \mathcal{F}$ と $k = 0, 1, \cdots, h(X)-1$ に対する問題 $(\mathbf{P}')$ においては，$v \in X$ に対する

$$z_v^k(t) = \begin{cases} \displaystyle\sum_{i=0}^{k} \Delta d(X_{j_i}) & (t = v) \\ 0 & (t \in \Delta X - \{v\}) \end{cases}$$

のいずれかが最適解になることから ($d(X_{j_0}) = \Delta d(X_{j_0})$ に注意),

$$g_X(k) = \left(\min_{v\in X} f_X(z_v^k),\ \arg\min_{v\in X} f_X(z_v^k)\right) \tag{4}$$

となる．ただし，$\arg\min_{v\in X} f_X(z_v^k)$ とは $f_X(z_{v^*}^k) = \min_{v\in X} f_X(z_v^k)$ となるような $v^* \in X$ のことである．

極小でない $X \in \mathcal{F}$ については，補題１より

$$g_X(k)_1 = \min\left\{ \min_{v\in\Delta X}\left\{ f_{\Delta X}(z_v^k) + \sum_{Y\in\mathcal{S}(X)} g_Y(0)_1 \right\},\ \min_{Y\in\mathcal{S}(X)}\left\{ g_Y(k+1)_1 + \sum_{\substack{Z\in\mathcal{S}(X):\\ Z\neq Y}} g_Z(0)_1 \right\} \right\} \tag{5}$$

$$g_X(k)_2 = \begin{cases} v & \left(g_Y(k)_1 = f_{\Delta X}(z_v^k) + \displaystyle\sum_{Y\in\mathcal{S}(X)} g_Y(0)_1 \text{のとき}\right) \\ g_Y(k+1)_2 & (\text{そうでないとき}) \end{cases}$$

となる．

木表現 T の葉から根に向かい各 $w \in W - \{w_{i_0}\}$ において対応する $X \in \mathcal{F}$ のテーブル g_X を順次計算することにより，最適解を求めるアルゴリズムを以下に示す．

アルゴリズム テーブル

ステップ０: $\tilde{T}(= (\tilde{W}, \tilde{A})) := T$.

ステップ１: $\tilde{T}$ の葉 w を一つ選んで,

(1-I) 対応する $X \in \mathcal{F}$ のテーブル g_X を式 (4), 式 (5) に基づき作成する．

(1-II) $\tilde{W} := \tilde{W} - \{w\}$.
もし，$\tilde{W} = \{w_{i_0}\}$ ならばステップ 2へ．
そうでなければステップ１へ戻る．

ステップ２: $\tilde{T} := T$,　$x^*(v) := 0\,(v \in V)$.

ステップ3: $\tilde{T}$ における $Y \in \mathcal{S}(X_{i_0})$ を一つ選んで，$(g_Y(0)_2 \in \Delta X_{j_0}, w_{j_0} \rightarrow w_{j_1} \rightarrow \cdots \rightarrow w_{j_{l+1}} (= w_{i_0})$ は $\tilde{T}$ 上の有向パスとする.)

(3-I) $x^*(g_Y(0)_2) := \sum_{i=0}^{l} \Delta d(X_{j_i})$.

(3-II) $\tilde{T}$ からこの有向パスを取り除き，更新. もし，$\tilde{W} = \{w_{i_0}\}$ ならばステップ 4 へ. そうでなければステップ 3 へ戻る.

ステップ4: x^* を出力し，終了する.

詳しい証明は省略するが，補題１より，このアルゴリズムは，コスト関数 F が式 (1) で与えられるときに問題 ($\mathbf{P}$) の最適解を求めることが分かる．また計算量は，ステップ０，２，３，４が線形時間で計算可能であるため，ステップ１，すなわち，テーブル g_X の構成に要する時間のみを考察すればよい．テーブル g_X は式 (4),(5) によって作成されるため，

$$\mathrm{O}\left(\sum_{X \in \mathcal{F}} h(X)(|\mathcal{S}(X)| + |\Delta X| q)\right) = \mathrm{O}(n^2 q)$$

となることが分かる．よって定理１が示された.

3　定理２の略証

本節では，式 (2) で記述されるコスト関数 F のオラクルが与えられたとき，問題を解くために必要なオラクル呼び出し回数の下界について考察する.

次の問題例について考える.

問題例 I

ラミナー族 : $\mathcal{F} = \{X_1, X_2, \cdots, X_{\frac{n}{2}+1}\}$,
ただし, $X_i = \{v_1, \cdots, v_{\frac{n}{2}-1+i}\} (i = 1, \cdots, \frac{n}{2}+1)$
要求関数　: $d(X_i) = i \quad (i = 1, \cdots, \frac{n}{2}+1)$
コスト関数 : $F(x) = \sum_{v \in V} f_v(x(v))$

$$f_{v_i}(\alpha) = \begin{cases} g_0(\alpha) & (v_i \in X_1) \\ g_i(\alpha) & (v_i \in V - X_1) \end{cases}$$

ただし，$g_0 : \mathbf{R}_+ \rightarrow \mathbf{R}_+$ は厳密に凹である単調増加関数である．例えば，$g_0(x) = \log(x+1)$ $(x \geq 0)$ とする．g_i は

$$g_i(\alpha) = \begin{cases} g_0(\frac{n}{2}+1) - g_0(i - \frac{n}{2}) & (\alpha > 0) \\ 0 & (\alpha = 0) \end{cases}$$

により与える．また，n は偶数とする.

補題２. 式 (2) で記述されるコスト関数 F がオラクルとして与えられる問題 ($\mathbf{P}$) では，少なくとも $\frac{n}{2}(\frac{n}{2}+1)$ 回以上のオラクル呼び出しを必要とする場合がある.

この補題により，定理２を得る．以下では，この補題の略証を与える.

この問題例に補題１を用いると，次のように記述できる最適解が存在することが分かる.

ある $v_i \in X_1, k = 1, \cdots, \frac{n}{2}+1$ に対して,

$$x_{(v_i,k)}(v) = \begin{cases} k & (v = v_i) \\ \frac{n}{2} - k + 1 & (v = v_{\frac{n}{2}+k}) \\ 0 & (v \in V - \{v_i, v_{\frac{n}{2}+k}\}) \end{cases} \tag{7}$$

この問題例の最適値は $g_0(\frac{n}{2}+1)$ となり，(7) 以外に最適解を持たない.

詳しい証明は省略するが，この問題例 I と $|S| \leq \frac{n}{2}(\frac{n}{2}+1) - 1$ である $S \subseteq \mathbf{R}_+^V$ に対して，I と同じラミナー族 $\mathcal{F}$ と要求関数 d を持つが，それぞれのコスト関数 F^- と F^+ が I のコスト関数 F とは異なるような問題例 I$^-$ と I$^+$ を次式を満たすように作成できる.

$$F^{\pm}(x) = \begin{cases} F(x) & (x \in S \text{ のとき}) \\ F(x) \pm \varepsilon & (x = x^* \text{のとき}) \end{cases}$$

ただし，ε は十分小さい正数，$x^* \in \mathbf{R}_+^V$ は式 (7) に示されるベクトルで $x^* \notin S$ である（$|S| \leq \frac{n}{2}(\frac{n}{2}+1) - 1$ より，そのような x^* は存在する）．このとき I$^-$ の最適解は x^* のみであり，I$^+$ の最適解は x^* 以外でかつ，式 (7) で記述できるものである.

ここで，あるアルゴリズム A が問題例 I に対して，上記の S に属するベクトルに対してのみオラクル呼び出しを行い，最適解として y を出力したとすると，$F(x) = F^-(x) = F^+(x)$ $(x \in S)$ より y は問題例 I, I$^-$, I$^+$ すべてにおいて最適解となるはずであるが，これは上記の議論より矛盾している．従って，補題２が示された.

4　定理３のアイデア

本節では，ラミナー被覆制約を持つ凹関数最小化問題のうち，コスト関数 F が式 (3)，すなわち $F = \sum_{v \in V} f_v$

$$f_v(x) = \begin{cases} a_v x + b_v & (x > 0 \text{ のとき}) \\ 0 & (x = 0 \text{ のとき}) \end{cases} \tag{8}$$

と記述できる場合について考察する.

このコスト関数は式 (1) の特殊形であることから，2 節のアルゴリズムを用いて $\mathrm{O}(n^2)$ 時間で解くことができる．2 節のアルゴリズムは，式 (4),(5) に基づき，テーブル $g_X (X \in \mathcal{F})$ を葉から根に向かい順に作成している．今，コスト関数が式 (8) のように分離可能であるため，式 (4),(5) よりテーブル g_X は，各 $v \in X$ の f_v をある一定値平行移動させたもの $\hat{f}_v$ の下側エンベロープ $l_X(x) = \min_{v \in X} \hat{f}_v(x)$ に対応することが分かる．我々は，コスト関数 F が式 (3) で与えられるとき，下側エンベロープ l_X を用いて g_X を間接的に表現することにより高速化 ($\mathrm{O}(n \log^2 n)$ 時間) に成功している.

F がオラクルで与えられる場合は，各 v に対して，2 回のオラクル呼び出しにより，a_v, b_v を求められるので，$\mathrm{O}(n(\log^2 n + q))$ 時間となる.

参考文献

[1] K. Arata, S. Iwata, K. Makino, and S. Fujishige: Locating sources to meet flow demands in undirected networks, *Journal of Algorithms*, **42** (2002), 54-68.

[2] H. Nagamochi: Computing extreme sets in graphs and its applications, *Proc. of the 3rd Hungarian-Japanese Symposium on Discrete Mathematics and Its Applications* (January 21-24, 2003, Tokyo, Japan) 349-357.

[3] H. Tamura, M. Sengoku, S. Shinoda, and T. Abe: Some covering problems in location theory on flow networks, *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, **E75-A** (1992), 678-683.

[4] 田村, 菅原, 仙石, 篠田: 無向フローネットワークにおける総合被覆問題について，電子情報通信学会論文誌, **J81-A** (1998), 863-869.