1999年度日本オペレーションズ・リサーチ学会
秋季研究発表会

k-NA 枝連結かつ 2-NA 点連結な最小領域配置問題

板津裕一郎 (入会申請中) 伊藤大雄 (01009550) 上原秀幸 横山光雄
豊橋技術科学大学 情報工学専攻

1 はじめに

通信網では同じサービスを提供する複数のサーバが存在する場合がある．ユーザはこのサービスを受ける場合, 通信網上でどれか 1 つのサーバに接続できればよい．これをグラフにモデル化すると, 重要なのは節点間の連結度ではなく, 節点と節点部分集合（領域）間の連結度であるといえる．そこで, 従来の点 (枝) 連結度を一般化した概念であり, 節点と領域間の連結度を表わす NA 点 (枝) 連結度という概念が導入された [1].

与えられた領域グラフに対して, k-NA 枝連結を満たす最小な領域を求める問題は多項式時間で解けることが示されている [1]. また, k-NA 点連結を満たす節点数が最小な領域を求める問題は $k \leq 2$ に対して多項式時間で解け, $k \geq 3$ に対しては NP 困難であることが証明されている [2]. 本稿では, k-NA 枝連結性と 2-NA 点連結性を同時に満たす節点数最小の領域を求める, $k \geq 4$ の場合 $O(m+n^2+n\min\{m,kn\}\min\{k,n\})$ 時間, $k \leq 3$ の場合は $O(m+n)$ 時間のアルゴリズムを提案する.

2 定義

$G=(V,E)$ を無向グラフとし, 並列枝を含んでも良いものとする．V, Eはそれぞれ, 節点集合, 枝集合を表わし,$|V|=n, |E|=m$. グラフ G を明示したいときは, それぞれ $V(G), E(G)$ と表す. また, 2 つの節点部分集合 $Y, Z (Y \cap Z=\phi)$ に対し, $E(Y,Z)$ を Y, Z間にある枝の集合とする．グラフ $G=(V,E)$ と節点集合 Vの部分集合族 $X=\{W_1, W_2, ..., W_p\}, W_i \subseteq V$ の組合せを領域グラフと呼び, (G,X) で表現し, 各 W_iを領域と呼ぶ.

グラフ$G=(V,E)$において, $V' \subseteq V, x \in V-V'$に対し, $|W| < k-|E(\{x\},V')|$ を満たす任意の $W \subseteq V-(V' \cup x)$ に対し, G から Wと Wに接続している枝の集合 E', 及び $E(\{x\},V')$ を取り除いたグラフ $G'=(V-W, E-E'-E(\{x\},V'))$ において, x とある $y \in V'$が連結であるならば, x と V'は k-NA(*node-to-area*) 点連結であると言う．x と V'が k-NA 点連結である最大の k を x と V'の NA 点連結度とし, $\kappa(x,V')$ と表現する.

また, 領域グラフ (G,X) において任意の領域 $V_i \in X$と V_iに含まれない任意の節点 $x \in V-V_i$ に対して, x と V_iが k-NA 点連結ならば領域グラフ (G,X) は k-NA 点連結であると言う．領域グラフ (G,X) が k-NA 点連結である様な最大の k を (G,X) の NA 点連結度と言い, $\kappa(G,X)$ と表現する.

$V' \subseteq V$と $x \in V-V'$に対し x と全ての $y \in V'$ を非連結にするために取り除く枝の数が少なくとも kであるとき, x と V'は k-NA 枝連結と言い, 最大の kを x と V'の NA 枝連結度と呼び, $\lambda(x,V')$ と表現する.

また, 領域グラフ (G,X) において, 任意の領域 $V_i \in X$と V_iに含まれない任意の節点 $x \in V-V_i$に対して x と V_iが k-NA 枝連結であるならば (G,X) は k-NA 枝連結であると言う．(G,X) が k-NA 枝連結であるような最大の kを領域グラフ (G,X) の NA 枝連結度と呼び, $\lambda(G,X)$ と表現する.

本稿では以下の問題を取り扱う.

$(k,2)$-領域節点最小配置問題 ((k,2)-AREALOC)
入力:$G=(V,E)$
出力:$(G, X=\{W\})$
目的関数:$|W| \rightarrow min$
制約:$\lambda(G,X) \geq k, \kappa(G,X) \geq 2$
但し, kは任意の整数をとり得る定数とする.

本問を取り扱う際に 2 点連結成分 [3] が重要な役割となる．1 つの節点の除去で 2 つ以上の部分グラフに分割されない極大な部分グラフを 2 点連結成分と呼び, 異なる 2 点連結成分が 1 つの節点を共有する場合, この節点を境界点と呼び, 境界点を 1 つしか持たない 2 点連結成分を葉成分と呼ぶ.

また, 枝連結成分を次のように定義する．節点部分集合を枝連結度のランクごとに分割する. $x, y, z \in V$について$\lambda(x,y) \geq k$かつ$\lambda(y,z) \geq k$

ならば$\lambda(x,z) \geq k$が成立する．これより $x, y \in \exists C(k)_i \Leftrightarrow \lambda(x,y) \geq k$であるような分割 $C(k)_1, C(k)_2, \cdots, C(k)_{K(k)}$が存在することが言える．(このような分割は各$k$に対して固有のものになる．) このような分割を$k$枝連結成分と呼ぶ．

3 アルゴリズム

k-NA 枝連結, 2-NA 点連結の性質から以下の定理が導かれる．

[定理 1] (G, X) がk-NA 枝連結かつ 2-NA 点連結であるための必要十分条件は以下の 2 つの条件を同時に満たすことである．

- 2 点連結成分の葉成分 D_jは任意の領域の節点を少なくとも 1 つ含む．
- 任意の葉成分 D_jに対して，$D_j \subseteq C(h)_i$かつ $|E(C(h)_i, V - C(h)_i)| < k$である h-枝連結成分 $C(h)_i (h = \{1, 2, ..., k\})$ は任意の領域の節点を少なくとも 1 つ含む．

[証明] 略

この性質を利用してグラフを与えた時, k-NA 枝連結かつ 2-NA 点連結な領域グラフを求めるアルゴリズムが導ける．

Procedure (k,2)-AREALOC
begin $W = \phi$
Step1 2 点連結成分の葉成分 D_jの一覧を作成 (リスト L_1)
Step2 各 $h = 1, 2, ..., k$に対し h 枝連結成分 $C(h)_i$を作成し, $|E(C(h)_i, V - C(h)_i)| < k$を満たす $C(h)_i$で極小なもののリストを作成する．(リスト L_2)
Step3 while $L_2 \neq null$ **do**
$C(h)_i \in L_2$に対し，すべての $D_j \in L_1$に対して，$D_j \not\subseteq C(h)_i$ならば節点 $x \in C(h)_i$を 1 つ取り,$W := W \cup \{x\}$ とする．
$L_2 := L_2 - \{C(h)_i\}$ **end do**
Step4 while $L_1 \neq null$ **do**
$D_j \in L_1'$に対し, $D_j \cap W = \phi$ならば D_jの境界点 x をとり $W := W \cup \{x\}$ とする．
$L_1 := L_1 - \{D_j\}$ **end do**
end

$k = 3$とし, 図 1のグラフを入力した場合, 図 1 の葉成分 D_1, 3 枝連結成分 $C(3)_1, C(3)_2$の 3 つの部分グラフがそれぞれ領域 Wの要素を含めば良いので, 網がけされた 3 節点を Wの要素として選べば良い．

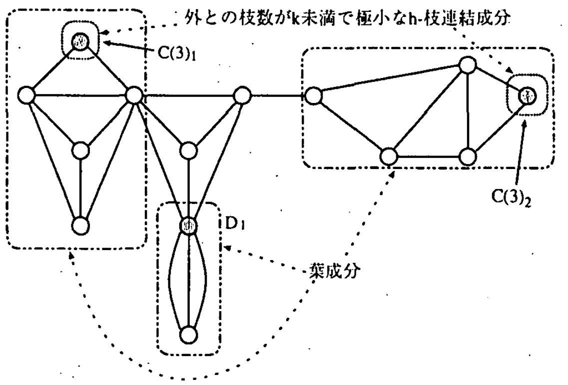

図 1: 3-NA 枝連結かつ 2-NA 点連結な領域グラフ

[定理 2] (k,2)-領域節点最小配置問題は $k \geq 4$ の場合 $O(m + n^2 + n \min\{m, kn\} \min\{k, n\})$ で解くことができ，$k \leq 3$ の場合は $O(m + n)$ で解ける．

[証明の概略] 最適性は定理 1 により保証される． 枝連結成分の作成に $k \geq 4$ の場合 $O(m + n^2 \min\{m, kn\} \min\{k, n\})$ 時間 [4], $k \leq 3$ の場合 $O(m + n)$ 時間 [5] かかる．それ以外の操作は工夫することで線形時間でできる．□

参考文献

[1] H.Ito and M.Yokoyama, Edge Connectivity between Nodes and Node-Subsets, Networks, Vol.31, No.3, pp.157-164, 1998.

[2] 伊藤資泰, 節点と節点部分集合の 2 および 3 点連結化問題, 信学技報, Vol.97, No.628, pp.32-40, 1998.

[3] A.V. エイホ, J.E. ホップクロフト, J.D. ウルマン 著, 野崎昭弘, 野下浩平 訳, アルゴリズムの設計と解析 I, サイエンス社, 1977.

[4] H.Nagamochi and T.Watanabe, Computing k-edge-connected componentes in multigraphs, Trans. Inst. Electron. Inform. Comm. Eng. Jpn, Vol.E76-A, No.4, pp.513-517, 1993.

[5] H.Nagamochi and T.Ibaraki, A linear-time algorithm for computing 3-edge-connected components in a multigraph, Jpn. J. Ind. Appl, Vol.Math.9, No.2, pp.163-180, 1992.