DEWS2005 5-C-i5

コミュニティ集合族発見手法の比較

伊野　秀彦†　　工藤　峰一†　　中村　篤祥†

† 北海道大学情報科学研究科，〒 060-0814 北海道札幌市北区北 14 条西 9 丁目
E-mail: †{hino,mine,atsu}@main.ist.hokudai.ac.jp

あらまし　近年，Web のリンク構造におけるグラフから，コミュニティと呼ばれる密に結合した部分グラフを抽出する研究が盛んに行なわれている．しかし，抽出された部分グラフが厳密に定義された条件を満たすことを保証する方法はほとんど提案されていない．本論文では，我々が [5] において提案したコミュニティ定義 (Flake らが提案したコミュニティに条件を加えたもの) を厳密に満たす部分グラフを抽出する問題を扱う．この問題において，全てのコミュニティを効率的に見つけることは困難であることが知られている．そこで，条件を満たす多くの部分グラフが効率的に見つかる可能性のある方法として，Gomory-Hu 木の各枝が表す分割から探す方法，及び edge-betweenness による分割から探す方法を考え，比較実験を行う．また，見つかった個々のコミュニティをランク付けするための評価基準の提案を行う．

キーワード　Web マイニング，Web コミュニティ，ネットワーク，グラフ理論

Comparison of Methods Finding a Community Subclass

Hidehiko INO†, Mineichi KUDO†, and Atsuyoshi NAKAMURA†

† Laboratory for Pattern Recognition and Machine Learning
Division of Computer Science
Graduate School of Information Science and Technology
Hokkaido University, Sapporo 060-0814, Hokkaido, Japan
E-mail: †{hino,mine,atsu}@main.ist.hokudai.ac.jp

Abstract　Recently, researches on extraction of densely connected subgraphs, which are called communities, from the Web graph representing link structure in WWW, are very popular. However, few methods guarantee that extracted subgraphs satisfy community conditions which are strictly defined. In this paper, we consider the problem of extracting subgraphs that strictly satisfy the community conditions defined in [5]. It is known that finding all communities is hard with respect to this problem. Here, as methods that possibly find many communities efficiently, we experimentally compare two methods, a method using a Gomory-Hu tree and a method using edge-betweenness. We also propose evaluation criterion for ranking found communities.

Key words　Web mining, Web community, network, graph theory

1. はじめに

近年の WWW(World-Wide Web) の発展に伴い，インターネットを土台とした研究が盛んに行なわれている．中でもインターネットのハイパーリンク構造に着目し，WWW をグラフとみなして解析を試みる方法が様々提案されている．有名なものでは Google の検索結果のランキングに使われている PageRank アルゴリズム [1] が挙げられる．

WWW のグラフにおいて，興味深い構造の一つに Web コミュニティ [2](以下コミュニティと呼ぶ) というものがある．コミュニティとは，「リンクにより密に結合した Web ページ集合」のことである．リンクはページ製作者により人為的に張られたものであることから，コミュニティは興味の似た人々により作られたページの集合であると考えられる．このことから，Web におけるコミュニティ構造の発見は，検索やデータマイニングに役に立つと考えられる．

密に結合した部分グラフとしてのコミュニティ定義の主なものとして，次の 2 つがあげられる．1 つ目は，Kumar ら [3] が提案した「あるサイズ以上の完全 2 部グラフを含む密な 2 部グラフ」がコミュニティであるという定義である．2 つ目は，Flake, Lawrence, Giles [4] が提案した「属するどの頂点も，集合外の頂点との間の辺の数以上の辺を集合内の頂点との間にもつような頂点集合」がコミュニティであるという定義である．kumar らの定義は密な部分，つまりコミュニティの核に着目した定義

であり，Flakeらの定義は疎な部分，つまりコミュニティの境界に着目した定義であるとみることができる．どちらの着眼点も重要であり，2つの定義は相反するものではない．実際，両方の定義を満たすコミュニティ定義も可能である．本論文ではコミュニティ境界に着目した定義について考える．

我々は，[5] においてFlakeらの提案したコミュニティ定義[4]の境界の曖昧性を指摘し，条件を付加したコミュニティ定義を提案した．また，この定義を満たすコミュニティ集合族の一部を発見する効率的なアルゴリズムとして，最大流アルゴリズムを繰り返し適用することにより構築できるGomory-Hu木によるコミュニティ発見手法を提案した．[5] で指摘したように，最大流アルゴリズムで求まるカットにより定義を満たす集合が常に見つかるとは限らないことに注意されたい．

最大流アルゴリズムは，結合が密な部分を探すのではなく，逆に疎な部分を探すアルゴリズムである．コミュニティ境界では結合が疎になっているため，そのようなアルゴリズムでコミュニティは検出されやすい．同様に結合が疎な部分を探す手法として，edge-betweennessによるコミュニティ発見手法[6]が挙げられる．edge-betweennessとは，グラフにおける中心性尺度であるbetweennessの概念を辺に適用したものである．edge-betweennessの値は，その辺を通る最短経路の本数であるので，コミュニティとその外側を結ぶ辺のedge-betweennessは高い値をとると予想される．そこで，edge-betweennessの値が高い辺から順に削除していけば，コミュニティとなる部分グラフが切り出せるであろうということである．この手法において求まる（頂点数 -1）の各々のカットで分割される集合がコミュニティの定義を満たしている可能性は高いと考えられる．

そこで，本論文では論文[5]で定義したコミュニティ定義を満たすコミュニティ発見手法として，効率的に多くのコミュニティを発見する可能性のあるGomory-Hu木による方法とedge-betweennessによる方法の比較実験を行う．

我々の実験結果によれば，Gomory-Hu木による方法の方が多くのコミュニティを発見できた．しかし，edge-betweennessによる方法ではGomory-Hu木による方法では見つからないコミュニティも多数見つかるので，両方使うことにより，より多くのコミュニティを見つけることができる．

多くのコミュニティが見つかった場合，その中から興味深いコミュニティを探すのには手間がかかる．そこで，個々のコミュニティを，コミュニティ内の辺密度の高さとコミュニティ境界の辺疎度の高さから評価する評価基準を提案する．edge-betweennessの方が辺密度の高いものが見つかるが，Gomory-Hu木の方が境界の辺疎度の高いものが見つかる傾向にあることがわかった．

2. コミュニティ定義

我々はFlakeらのコミュニティ定義に注目し，問題点をある程度改善した新たなコミュニティ定義を提案した[5]．

ページの集合を頂点集合 V，異なるページ間のリンクの集合を辺集合 E とする無向グラフ $G=(V,E)$ を考える．全ての頂点の組 (u,v) に対し，重み $w_{u,v} \geqq 0$ が与えられているものとする．ただし，(u,v) と (v,u) は同一視し，$w_{u,v}=w_{v,u}$ とする．E に属さない組 (u,v) に関しては $w_{u,v}=0$ を満たすとする．E に属する組に関しては，どのように重みが与えられていてもよいが，断りのない限り1の重みが与えられているものとする．

Flakeら[4]は，Webコミュニティを次のように定義した．（本論文では，Flakeらの定義によるコミュニティをFLG-コミュニティと呼ぶ．）

［定義1］ FLG-コミュニティとは頂点の部分集合 $C \subset V$ で条件1を満たすものである．

［条件1］ $\sum_{v\in C} w_{uv} \geqq \sum_{v\in V-C} w_{uv}$ $for\ all\ u \in C.$

この定義には，境界が曖昧であるという問題点がある．例えば，図1のグラフでは頂点集合 C_1，C_2，C_3，C_4 および C_5 は，本質的に1つの密に結合した部分であるにもかかわらず，全てFLG-コミュニティである．したがって，1つの密に結合した部分から1つのコミュニティを抽出したい場合にはどれを代表とすべきかの問題が生ずる．

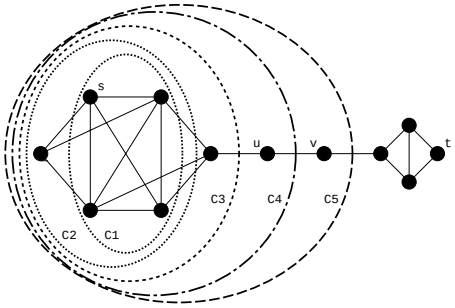

図1 FLG-コミュニティの例 $C_1, C_2, ..., C_5$

そこで，我々は[5]において次のようなより厳しいコミュニティ定義を提案した．

［定義2］ **弱コミュニティ**とは，頂点の部分集合 $C \subset V$ であり，条件1および条件2を満たすものである．

［条件2］ $\sum_{v\in C} w_{uv} \leqq \sum_{v\in V-C} w_{uv}$ $for\ all\ u \in V-C.$

［定義3］ **コミュニティ**とは，頂点の部分集合 $C \subset V$ であり，条件3および条件2を満たすものである．

［条件3］ $\sum_{v\in C} w_{uv} > \sum_{v\in V-C} w_{uv}$ $for\ all\ u \in C.$

V と $\emptyset$ はコミュニティである．以下断りのない限り，コミュニティとは定義3を満たすコミュニティのことを指す．図1のグラフでは，5つのFLG-コミュニティ C_1，C_2，...，C_5 の内，C_3，C_4，C_5 が弱コミュニティであり，C_3 だけがコミュニティである．

コミュニティに関しては，以下の命題が成り立つので，1点多いまたは少ない集合はコミュニティにはなり得ない．したがって境界の曖昧性はある程度解消できている．

［命題1］ C_1 と C_2 が異なるコミュニティであれば，2つの集合の対象差は2つ以上の要素を含む．

3. 最大流アルゴリズムによるコミュニティ発見の問題点

Flakeら[4]は，最大流アルゴリズムを用いたFLG-コミュニティ発見手法を提案している．彼らの主張によれば，FLG-コ

ミュニティCは次の条件を満たす2頂点s，tに対するs-t最大流アルゴリズム[7], [13]により発見できるという．

条件A　Cと$V-C$の間の辺の数が，sと$C-\{s\}$間の辺の数およびtと$V-C-\{t\}$間の辺の数より少ない．

しかし，この主張に対し図2のような反例[5]が存在する．図2のグラフでは，C_2は条件Aを満たすが，見つかるのはFLG-コミュニティでもないC_1である．

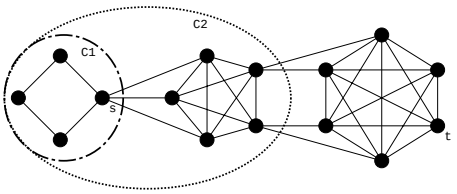

図2　反　　例

しかし，「Cと$V-C$の間の辺の数が，sと$C-\{s\}$間の辺の数およびtと$V-C-\{t\}$間の辺の数より多い場合，$s-t$最大流アルゴリズムでFLG-コミュニティCは同定できない」ということは明らかな事実であり，Flakeらの主張しているように，サイズの大きなFLG-コミュニティを見つけたいのであればソースsとシンクtとして次数の大きい頂点を選ぶべきであるということは正しい．実際，彼らは複数のソースを選びそれらの頂点を疑似ソースと無限の容量の辺で繋いだグラフを使っているが，疑似ソースとつながれた複数のソースに関して，最大流アルゴリズムにより求まった集合がFLG-コミュニティの定義を満たさない可能性があるという問題は依然として存在する．

Flakeらが提案したコミュニティ発見アルゴリズムは，実際にはもっと複雑なことを行っているため，彼らのアルゴリズムで求まったコミュニティがFLG-コミュニティの定義を満たさない可能性が高いと断言はできない．しかし，彼らのアルゴリズムを簡略化したもの(注1)を実データに適用したところ，複数ソースの場合でもFLG-コミュニティ定義を満たさない頂点集合が得られることが確認できた．

使用した実データは，次のように作成した．先ず，検索エンジンGoogle(注2)で，「AIBO」という検索語で検索を行なった結果得られた上位1000件のwebページを頂点とするグラフを作成し，最大連結成分のみ残し，残りを削除した．その後，重複リンクを除き，全ての頂点の次数が2以上になるまで次数1の頂点の削除を繰り返した．その結果，図3のような，頂点数64，辺数102のグラフが得られた．ソースとして日本語のページを4つ指定し，疑似ソースと無限の容量の辺で結んだ．それから，Flakeらが近似コミュニティを求める方法として提案しているように，全ての辺の重みをk倍し，全ての頂点と重さ1の辺で結ばれた疑似シンク加えた．kの値は4とした(注3)．

(注1)：Flakeらはグラフの頂点集合をソースから辿って集めており，ソース及びその隣接頂点は疑似シンクと結んでいない．また，ソースを増やして最大流アルゴリズム繰り返し適用するEMアルゴリズム的な方法を取っている．

(注2)：http://www.google.co.jp/

(注3)：Flakeらの提案法では，ソースの数に設定している為そのように設定した．

このように作成したグラフに最大流アルゴリズムを適用したところ，図3において丸で示された頂点の集合が出力された．図中の二重丸の頂点がソースに選んだ頂点である．この結果では，四角の点線で囲まれた2つの頂点はFLG-コミュニティの定義を満たしていない．図中の破線で示されるカットはソース側の頂点がFLG-コミュニティの定義も満たすカットである．実際の日本語のwebページは色づけされた頂点であり，定義を満たすこのカットの方が最大流アルゴリズムで求まったカットより望ましいことがわかる．

4. コミュニティ集合族の効率的発見法

本論文では，与えられたグラフに対し，我々のコミュニティ定義を満たす多くの集合を効率的に求める方法について考える．

与えられた2頂点s,tに対し，sを含みtを含まないようなコミュニティの存在の有無を判定する問題はNP完全であることがわかっている[12]．したがって，すべてのコミュニティを求めるのは現実的に難しいと考えられる．

ここでは，最大流アルゴリズムをベースにしたGomory-Hu木によるコミュニティ発見手法[5]と，edge-betweennessによるコミュニティ発見手法[6]について考える．

4.1 Gomory-Hu木によるコミュニティ発見手法

最大流アルゴリズムを用いれば，与えられたソースとシンクを分離する最小カットを求めることができる．最もソース側の最小カットで分離される，ソースを含む頂点集合はコミュニティの定義を満たしていない可能性があるが，満たしていない可能性があるのはソースとシンクのみである[5]．そこで，ソースとシンクがコミュニティの定義を満たしているか否かをチェックするのみでコミュニティか否かの判定が可能である．Gomory-Hu木によるコミュニティ発見手法は，効率的に求まる$n-1$個の最小カットにより生ずる頂点の部分集合の中から，この事実を利用してコミュニティの定義を満たす集合を効率的に探す方法である．ただし，nはグラフの頂点数とする．グラフGのGomory-Hu木Tは，Gと同じ頂点集合からなり$n-1$本の辺からなる連結グラフであり，Gの任意の2点の最小カットの少なくとも1つは，Tの辺で表現されているような木である[13]．Gomory-Hu木は，最大流アルゴリズムを$n-1$回適用することにより効率的に求めることができる．Gomory-Hu木による表現は一意ではなく，ソースとシンクの選ぶ順番によって同じグラフから異なるGomory-Hu木が得られる可能性がある．このため[5]では，Gomory-Hu木生成の際にコミュニティがより多く求まるように，最もソース側の最小カットと最もシンク側の最小カットの両方対象としてコミュニティが見つかった方のカットを採用するといったヒューリスティックを用いた．最大流アルゴリズムとしてSleatorとTarjan[7]のアルゴリズムを用いれば，$O(n^2 m \log n)$で計算できる．ただし，nは頂点数，mは辺の数とである．

4.2 edge-betweennessによるコミュニティ発見手法

edge-betweennessとは，グラフの中心性指標を表すbetweenness[8][9]の概念を辺に適用させたものである．$g_e^{(st)}$を辺eを通る頂点sからtへの最短経路の本数とし，n_{st}をsからtへ

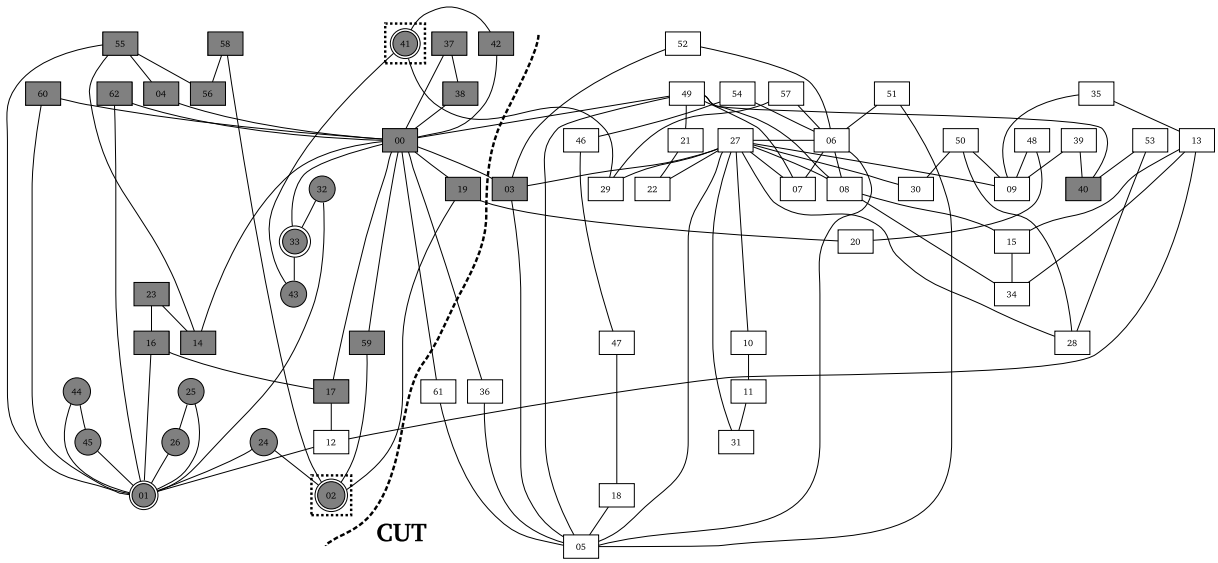

図 3　実際の web グラフにおける例

の最短経路の総数とすると，辺数 m のグラフにおける辺 e の edge-betweenness は以下のように定義される．

$$EB_e = \frac{\sum_{s<t} g_e^{(st)}/n_{st}}{\frac{1}{2}m(m-1)}.$$

edge-betweenness によるコミュニティ発見手法は，edge-betweenness の値が最も高い辺を削除するという操作を繰り返すことによりグラフを分割する方法である．

辺の削除を m 回繰り返す内に $n-1$ 回の分割が生ずる．ただし，m は辺数，n は頂点数とする．この各々の分割で生ずる頂点集合に対して，コミュニティの定義を満たしているか否かをチェックする．最大流アルゴリズムの場合と異なり，どの頂点も条件を満たしていることが保証されないため，全ての頂点について条件を満たしているか否かをチェックする必要が生ずる(注4)．そのため，Newman と Girvan のアルゴリズムの計算時間 $O(nm^2)$ に加えてこのチェックのために $O(n(n+m))$ の計算時間が必要である．したがって計算時間の上界としては,m が $O(n)$ の場合には Edge-betweenness の方がよく，m が $O(n^2)$ のときは Gomory-Hu 木の方がよいことになる．

Newman と Girvan の行った人工データによる実験の結果 [6] は，edge-betweenness によるコミュニティ発見手法により，我々のコミュニティ定義を満たす頂点集合が多く見つかるのではないかという期待を膨らませるものである．彼らは，128 個の頂点を 32 個ずつの 4 つのグループに分け，各頂点から 16 本の辺を k 本は所属グループ外との頂点との間に，$16-k$ 本は所属グループ内の頂点との間にランダムに生成させたグラフから，edge-betweenness によるコミュニティ発見手法でグループをうまく分離できるかという実験を行なった．$k \leqq 7$ の場合，4 つのグループは我々のコミュニティ定義を満たすことに注意して欲しい．実験の結果，$k \leqq 6$ の場合には，完全にグループを抽出できたと報告されている．

(注4)：境界の頂点のみチェックすればよいので，平均的にはそれほど計算量を必要としないと考えられる．

5. コミュニティの評価基準

前節では，効率的に多くのコミュニティ集合が求まる可能性のある方法について考えた．しかし，実際に多くのコミュニティが求まってしまうと，その中から興味深いコミュニティを探すのは手間のかかる作業である．そこで，見つかったコミュニティをランク付けるための評価基準について考える．

コミュニティは「密に結合した web の部分グラフ」であるという考えに基づけば，コミュニティらしさの指標は

- コミュニティ内の辺密度の高さ
- コミュニティ境界における辺疎度の高さ

の 2 点であると考えられる．これら 2 つの指標の数値化を以下のように行う方法を提案する．

グラフ $G=(V,E)$ において，頂点集合 C がコミュニティの定義を満たしているものとする．このとき，コミュニティ外の頂点集合を $\bar{C} = V - C$ とする．G，C，$\bar{C}$ における頂点数をそれぞれ n，n_C，$n_{\bar{C}}$ とする．また，G，C に属する頂点間の辺の数をそれぞれ m，m_C とし，C に属する頂点と $\bar{C}$ に属する頂点間の辺数を $m_{C,\bar{C}}$ とおく．

コミュニティC 内の辺密度の高さ d_C を以下のように定義する．

$$d_C = \frac{m_C}{\binom{n_C}{2}}.$$

ただし，$\binom{x}{2} = \frac{x(x-1)}{2}$ である．これは，実際の辺数と C が完全グラフの場合の辺数との比である．グラフ全体の辺密度 $d_G = m/\binom{n}{2}$ を基準とした C の対数相対辺密度を

$$\log(d_C/d_G)$$

と定義し，これをコミュニティC 内の辺密度の高さの指標とする．コミュニティC の境界における辺密度 $d_{C,\bar{C}}$ を以下のように定義する．

$$d_{C,\bar{C}} = \frac{m_{C,\bar{C}}}{n_C \cdot n_{\bar{C}}}.$$

これは，C と $\bar{C}$ の間の実際の辺数と，それが完全 2 部グラフである場合の辺数との比である．d_G を基準とした C の境界における対数相対辺疎度

$$\log(d_G/d_{C,\bar{C}})$$

で定義し，これをコミュニティC の境界における辺密度の低さの指標とする．ただし，$d_{C,\bar{C}} = 0$ の場合，つまり C が連結成分全体である場合は，この指標における評価値を ∞ とする．これら両方の指標を用いた総合的な評価値として，以下を定義する．

$$w\log(d_C/d_G) + (1-w)\log(d_G/d_{C,\bar{C}}).$$

ただし，$0 \leqq w \leqq 1$ とし，以下断りのない限り $w = 0.5$ とする．

6. 実　　験

Gomory-Hu 木および edge-betweenness による方法を，実際の Web グラフに適用し，発見されるコミュニティの数，及び提案した評価基準による値により評価し比較を行った．

6.1 実験データ

キーワードを与え，Kleinberg [11] により提案された手続き *Subgraph* により Web グラフの部分グラフを作成した．手続き Subgraph では，検索エンジンにより集められた t ページをベースに，それらのページからリンクしているページとそれらのページをリンクしているページ（ただし 1 ページあたり d ページに限る）を加えたページ集合を頂点集合としてグラフを作成する．実験では，検索エンジンとして Google（www.google.co.jp）を採用し，t を 200，d を 50 に設定した．Kleinberg が行ったようにドメインが等しいページへのリンクは全て除いてある．様々な検索語に対して手続き Subgraph を適用した結果，表 1 に示されるような頂点，辺，連結成分をもつグラフが作成された．

表 1　各グラフデータ詳細 : 頂点数は孤立点を含んでいない．

検索語	頂点数	辺数	連結成分数
active learning	1156	2182	39
information filtering	1863	4138	26
TSP	1672	2946	47
boosting	2448	12140	66
BDD	926	1243	63
DFA	1854	6208	55
Michael Jordan	1957	7663	38
DNF	1209	3940	77
data mining	4059	20705	7
baseball matsui	3703	41658	39
learning theory	2679	5934	22
jaguar	2457	16173	52

6.2 発見されたコミュニティ数

表 2 は，2 つの方法を各検索語のグラフに対し適用した結果，得られたコミュニティ数(注5)を示している．表中の ‘GH’，‘EB’ にはそれぞれ Gomory-Hu 木及び edge-betweenness による方法により発見されたコミュニティ数が示されており，‘重複数’ には，共通に見つかったコミュニティの数が示されている．全て

表 2　発見されたコミュニティ数

検索語	GH	EB	重複数
active learning	91	36	7
information filtering	92	14	5
TSP	116	36	8
boosting	127	63	13
BDD	69	55	10
DFA	123	41	8
Michael Jordan	128	63	17
DNF	55	39	4
data mining	118	27	11
baseball matsui	81	45	10
learning theory	117	23	15
jaguar	147	43	23

のグラフにおいて，Gomory-Hu 木による方法の方がより多くのコミュニティを発見していることがわかる．しかし，重複するものは ‘learning theory’ と ‘jaguar’ を除いて edge-betweenness による方法により見つかったコミュニティの半分以下であり，edge-betweenness による方法では，Gomory-Hu 木による方法で見つからないようなコミュニティが発見されていることがわかる．我々は [5] において，コミュニティトポロジーによるグラフ分割を提案したが，分割に用いるコミュニティ集合族を求めるのに，両方の手法を用いることにより，より良い分割が得られる可能性がある．

6.3 提案評価基準による評価

表 3，4 は，2 つの方法により見つかったコミュニティの各々を，提案した基準で評価した値 $\log(d_C/d_G)$，$\log(d_G/d_{C,\bar{C}})$，総合評価値それぞれの平均値と最大値を示している．

表 3　EB での評価基準の実験結果

	$\log(d_C/d_G)$		$\log(d_G/d_{C,\bar{C}})$		総合	
検索語	mean	max	mean	max	mean	max
active learning	**4.548**	6.359	1.709	4.632	**3.04**	**4.629**
information filtering	**5.875**	6.654	0.505	1.385	2.609	4.019
TSP	**5.364**	6.728	1.486	2.88	**3.425**	3.946
boosting	**5.555**	7.109	1.881	4.938	3.622	5.034
BDD	**4.305**	5.954	1.652	4.74	**2.979**	3.759
DFA	**4.963**	6.667	1.392	3.817	2.909	4.573
Michael Jordan	**4.838**	6.549	1.719	4.793	3.234	4.421
DNF	**4.834**	6.367	2.305	5.435	3.5	4.607
data mining	**5.793**	7.21	1.174	2.925	**3.27**	**4.807**
baseball matsui	**5.308**	7.212	2.346	5.587	**3.614**	5.577
learning theory	**6.121**	**7.1**	0.841	2.319	**2.962**	4.292
jaguar	**5.74**	6.825	1.533	3.985	**3.498**	**4.61**

全体的な傾向として，コミュニティ内の辺密度に関しては edge-betweenness による手法の方がよいものが見つかるが，コミュニティ境界の辺疎度に関しては Gomory-Hu 木による手法の方がよいものが見つかるといえる．したがって総合的には 2

(注5) : 連結成分全体はコミュニティの定義を満たすが，コミュニティ数に含めていない．

表 4　GH での評価基準の実験結果

	$\log(d_C/d_G)$		$\log(d_G/d_{C,\bar{C}})$		総合	
検索語	mean	max	mean	max	mean	max
active learning	1.744	6.359	**3.44**	**5.293**	2.592	3.722
information filtering	2.24	**6.836**	**3.0**	**4.939**	**2.62**	**4.109**
TSP	2.734	6.728	**3.565**	**5.799**	3.15	**4.031**
boosting	3.336	7.109	**4.011**	**6.192**	**3.673**	5.034
BDD	2.309	**6.137**	**3.394**	**5.395**	2.852	3.759
DFA	2.575	6.667	**3.9**	**6.055**	**3.237**	4.573
Michael Jordan	3.065	**6.886**	**3.41**	**5.797**	**3.238**	4.421
DNF	4.057	6.367	**3.03**	**5.665**	**3.543**	4.607
data mining	2.308	**7.615**	**2.034**	**3.489**	2.171	4.298
baseball matsui	3.684	7.212	**3.494**	**5.993**	3.589	5.577
learning theory	1.978	6.977	**2.992**	**5.02**	2.485	4.292
jaguar	3.024	**7.113**	**3.916**	**6.097**	3.47	4.31

つの指標の重み w によりどちらがよいかは変わることになるが，$w = 1/2$ の場合は edge-betweenness による手法の方が若干良さそうにみえる．

表 5 は検索語 'jagaur' において，評価値が高かったもの上位 5 コミュニティの評価値（総合評価値とその時の $\log(d_C/d_G)$，$\log(d_G/d_{C,\bar{C}})$）である．

表 5　jaguar における両手法の評価値における上位 5 コミュニティ

	$\log(d_C/d_G)$		$\log(d_G/d_{C,\bar{C}})$		総合	
rank	EB	GH	EB	GH	EB	GH
1	5.235	4.898	**3.985**	**3.721**	4.61	4.31
2	4.898	5.428	3.721	2.82	4.31	4.124
3	5.643	**6.825**	2.696	1.383	4.17	4.104
4	**6.825**	5.59	1.383	2.56	4.104	4.075
5	6.708	6.485	1.385	1.606	4.046	4.045

edge-betweenness による方法の 1，3，5 位のコミュニティは Gomory-Hu 木による方法では見つからなかったものであり，Gomory-Hu 木による方法の 2，4 位のコミュニティは edge-betweenness による方法では見つからなかったものである．また，$\log(d_C/d_G)$，$\log(d_G/d_{C,\bar{C}})$ の値から，総合的な評価がコミュニティ内の辺密度と境界の辺疎度のどちらかに左右されるのではなく，2 つの指標がうまく反映されていることがわかる．

表 6 は，jaguar のグラフにおいて一番評価値の高かったコミュニティ（A）と 2 番目に評価値の高かったコミュニティ（B）に属するページの URL を示している．表中の URL はコミュニティ内にグラフを制限した場合に次数が高かったページで，ドメインの違うページを次数順に 5 ページ表示している．URL の前の数字は次数であり，後の [] の中の文字列はタイトルである．表中の A コミュニティは，www.anort.com，chulkov.com というサイトのコミュニティであり，車の jaguar に関するページが複数含まれていた．また，B コミュニティは（Belize 国にある）Jaguar Reef Lodge というリゾート地に関するコミュニティであった．

これら 2 つのコミュニティの内，A コミュニティは同じサイト内の微妙に異なるドメインのページ群であり，サイト内の参照が多いローカルなページがほとんどであるため評価値が高くなったものと考えられる．このようなほとんどサイトに閉じたローカルなコミュニティは，あまり面白いとはいえない．同一サイトへのリンクを除いてそのようなコミュニティが出来ないようにするか，または評価基準にサイトのバラツキの度合を反映させる等，何らかの工夫が必要であると考えられる．一方，B のコミュニティは様々なドメインのページからなるコミュニティであり，'jaguar reef lodge' という共通のトピックを持った興味深いコミュニティとなっている．

7. おわりに

提案したコミュニティ定義を厳密に満たす多くの集合を効率的に求める方法として，Gomory-Hu 木による方法と edge-betweenness による方法の比較実験を行った．Gomory-Hu 木による方法の方が多くのコミュニティを発見することができたが，edge-betweenness による方法は Gomory-Hu 木で見つけることができないコミュニティを多数発見しており，2 つの方法を両方用いてコミュニティを見つけることにより，より多くのコミュニティを発見できることがわかった．また，コミュニティ内の辺密度と境界の辺疎度を反映したコミュニティ評価基準を提案し，Gomory-Hu 木による方法と edge-betweenness による方法で見つかったコミュニティに対して評価を行った．提案した評価基準において，辺密度に関してはに edge-betweenness の方が高いものがみつかり，境界の辺疎度に関しては Gomory-Hu 木の方が高いものが見つかる傾向にあった．

本論文で比較した 2 つの方式では，見つからないコミュニティがまだ多数存在していると考えられる．その中にはもっと興味深いコミュニティが隠れている可能性がある．そのような，既存法では見つからない興味深いコミュニティを見つける方法を今後検討したい．

表 6　jaguar コミュニティの例

Community A "**www.anort.com**" and "**chulkov.com**"

52	http://chulkov.com/jaguar/jaguar.htm	[JAGUAR X-Type]
20	http://member.anort.com/signup.htm?ref=a	[Anort Member]
20	http://guestbook.anort.com/gb/index.htm?idgb=141	[Home Page]
20	http://www.anort.com?ref=a	[notitle]
16	http://www.anort.net/?ref=a	[Anort]
–	other 54 pages.	

Community B "**Jaguar Reef Lodge**"

35	http://www.jaguarreef.com	[Belize Resorts and hotels]
13	http://www.caribbean-shores.com	[Belize Caribbean Bed and Breakfast Inn, B & B]
12	http://www.cocoplumcay.com	[Belize Resorts, Vacations, Accommodations]
12	http://sitteerivermarina.com	[Welcome to Sittee River Marina]
12	http://www.dibbern.com/dd2000carib.htm	[Caribbean web sites and travel marketing]
–	other 53 pages.	

文　　献

[1] S. Brin, L. Page, The Anatomy of a Large-scale Hypertextual Web Search Engine. Computer Networks **30**(1–7), 1998, 107–117.

[2] 村田剛志, Web コミュニティ. 情報処理学会学会誌 **44**(7), 2003, 702–706.

[3] R. Kumar, P. Raghavan, S. Rajagopalan and A. Tomkins, Trawling the Web for Emerging Cyber-Communities. *Computer Networks*, **31**(11–16), 1999, 1481–1493.

[4] G. W. Flake, S. Lawrence and C. L. Giles, Efficient Identification of Web Communities. *6th ACM SIGKDD Conference on Knowledge Discovery and DataMining*, 2000, 150–160.

[5] 中村篤祥・工藤峰一, コミュニティトポロジーによる web グラフ分割, 第 4 回データマイニングワークショップ 日本ソフトウェア科学会 データマイニング研究会, 2004, 57–64.

[6] M. E. J. Newman and M. Girvan, Finding and evaluating community structure in networks, *Phys. Rev. E*, **69**(2004), 026113.

[7] R. Tarjan, Data Structure and Network Algorithm. *Society for Industrial and Applied Mathematics*, (1983).

[8] L. C. Freeman, A set of measures of centrality based upon betweenness. *Sociometry*, **40**(1977), 35–41.

[9] L. C. Freeman, Centrality in social networks: Conceptual clarification. *Social Networks*, **1**(1979), 215–239.

[10] M. E. J. Newman, Scientific collaboration networks: Ⅱ. Shortest paths, weighted networks, and centrality. *Phys. Rev. E*, **64**(2001), 016132.

[11] J. Kleinberg, Authoritative Sources in a Hyperlinked Environment. *Joutnal of the ACM*, **46**(5), 1999, 604–632.

[12] A. Nakamura, T. Shigezumi and M. Yamamoto, On NK-Community Problem, 2005 年冬の LA シンポジウム予稿集, 2005, 1201–1208.

[13] V. Vazirani, Approximation Algorithms. Springer-Verlag (2001).