パラメータ化された連結性に基づくWebページのグループ化

正田　備也†　　高須　淳宏††　　安達　　淳††

† 東京大学 情報理工学系研究科　〒 113–8656 東京都文京区本郷 7–3–1
†† 国立情報学研究所　〒 101–8430 東京都千代田区一ツ橋 2–1–2
E-mail: †{masada,takasu,adachi}@nii.ac.jp

あらまし　本論文では，ハイパーリンクから得られる情報のみを用いてWebページをグループ化する手法を提案する．グループ化のねらいは，Web上での効果的な検索やマイニングの単位として利用できる，適度な大きさのWebページの集合を構成することである．提案手法は，強連結成分の細分化としてのグループを与える．その際，グループの大きさは，閾値パラメータと呼ばれるパラメータを調節することで制御できる．また，本論文は1500万のWebページを対象とするグループ化実験の結果を含む．

キーワード　WWW，リンク解析，ハイパーリンク，クラスタリング，強連結成分

Grouping Web Pages Based on Parameterized Connectivity

Tomonari MASADA†, Atsuhiro TAKASU††, and Jun ADACHI††

† Graduate School of Information Science and Technology, The University of Tokyo　Hongo 7–3–1, Bunkyo-ku, Tokyo, 113–8656 Japan
†† The National Institute of Informatics　Hitotsubashi 2–1–2, Chiyoda-ku, Tokyo, 101–8430 Japan
E-mail: †{masada,takasu,adachi}@nii.ac.jp

Abstract　This paper proposes a method for grouping Web pages based only on hyperlink information. The aim of grouping is to construct Web page sets of moderate sizes which can be utilized as units for effective search and mining on Web. Our method provides groups as subdivisions of strongly connected components. Group sizes can be controlled by adjusting a parameter, called threshold parameter. This paper also includes experimental results of grouping 15 million Web pages.

Key words　WWW, link analysis, hyperlink, clustering, strongly connected component

1. はじめに

1.1 研究の目的

本研究では，ハイパーリンクから得られる情報のみを用いてWebページをグループ化するための手法を提案する．グループ化のねらいは，Web上での効果的な検索やマイニングの単位として，個別のWebページよりも大きく，かつ，一つのサーバよりも小さなWebページの集合を，可能な限りヒューリスティクスを排除した手法によって構成することに存する．

Webページの爆発的な増加によって，少なくともWebの世界においては，テキスト情報を利用した従来の慎重な文書クラスタリングの手法が活躍の機会を狭めつつある．なぜなら，形態素解析や構文解析のようなテキスト処理を，Webページの集合のように，その処理においてスケーラビリティの要求される巨大な文書集合へ適用することは困難だからである．そこで，相対的に処理コストの小さいリンク情報をWebページのグループ化に利用する試みが，近年多く見られる．本研究では，テキスト情報だけでなく，URLや，アンカーテキスト，サーバ内のディレクトリ構造などの情報も利用せず，ハイパーリンクによるWebページ相互の参照関係のみを用いてグループ化を実現する手法を提案する．

1.2 従来研究

従来の文書クラスタリングは，専ら，各文書に一定の次元のベクトルを割り当て，そのベクトルの類似性によって元の文書の類似性を評価する，という枠組みを採用している．各文書に対応するベクトルは，ほぼ単語の種類だけの次元をもち，文書 d_j に割り当てられるベクトル w_j の，単語 t_i に対応するエントリ $w_{i,j}$ は，t_i が文書 d_j に出現する回数 $freq_{i,j}$ と，単語 t_i が現われる文書の総数 n_i とから算出される．一例を挙げれば，文書の総数を n として，$w_{i,j} = \frac{freq_{i,j}}{\max_l freq_{l,j}} \cdot \log \frac{n}{n_i}$ という式によってベクトル w_j のエントリが計算されていく．

このように，高次元のベクトルとして表現された文書の集合に適用しうるクラスタリング手法としては，Scatter/Gather [6] やCLARANS [14] を挙げることができる．しかしながら，こ

の枠組みを採用すると，まず全文書について対応するベクトルを算出する手間が無視できなくなる．なぜなら，この処理には，少なくとも日本語の文章を単語に区切る操作，つまり形態素解析が含まれるからである．さらに，文書集合をベクトルの集合へと変換した後でもなお，ベクトルの次元数が全文書に現われる単語の種類に匹敵するため，「次元の呪縛 (curse of dimensionality)」という困難に見舞われる．特異値分解法 [7] [1] のような次元削減の手法が，文献検索の分野で活用されるゆえんである．よって，文書のベクトル表現に基づくクラスタリングは，Web のような大規模な文書集合には必ずしも有効でない．

そこで，Web ページがハイパーリンクによって参照しあっているという事実を利用することが考えれられる．リンク構造は，Web ページを頂点，リンクを枝とみなすことによって，一つのグラフ構造とみなされる．よって，グラフにおける頂点のクラスタリング手法を，Web の世界にも適用することができる．田島ら [19] は，重要でないリンクを切断することによって連結な部分グラフを取り出す，という手法を提案している．しかし，リンクの重要性を定量的に表現する際，上述のベクトル表現モデルが採用されている．つまり，ベクトルとして類似性の高い文書どうしを結ぶリンクが重要とみなされる．したがって，リンク構造をグラフとして捉えている点は注目に値するものの，スケーラビリティの観点から Web には適用しがたい．

もちろん，リンク構造のグラフとしての側面から得られる情報だけを利用して，頂点をグループ化することも可能ではあるが，グラフに関わる問題には，解くことの困難なものや，多大な計算量を要するものが多い．だが，近年，グラフの隣接行列やラプラシアンの固有ベクトルを利用することで，カットを小さくするという意味で良いグループ化を求めるための近似アルゴリズムが提案されている [18] [10]．ところが，これらのアルゴリズムは，隣接行列やラプラシアンが対称行列となる無向グラフのみを対象としており，Web のリンク構造に直接適用することはできない．なぜなら，リンクには向きがあり，したがって，リンク構造は有向グラフとみなされるべきだからである．そこで，Kleinberg はリンク構造に対応する有向グラフの隣接行列 A そのものではなく，A^TA および AA^T という対称化された行列に対して同趣旨の手法を適用し，Web ページをグループ化している [12]．具体的には，A^TA と AA^T のそれぞれについて，非主要固有ベクトルを算出し，そのエントリの正負の区別をグループ化に利用している．だが，この手法については，有効性を疑問視する研究がある [2].

その一方，Web グラフの効果的なマイニングや可視化を目的とする研究の中には，個別の Web ページよりも大きく，かつ，一つのサーバよりも小さな Web ページの集合（「サイト」と呼ばれる）を構成するための手法について，興味深い提案が見られる [20] [22]．しかし，いずれの提案においても，ページをどのような場合に同じグループにまとめるかの基準が，URL やサーバ内のディレクトリ構造など，ハイパーリンクとは無関係な情報にも依拠するかたちでヒューリスティックに定められている．そのため，これらの提案に基づいて構築されたシステムにおいては，グループの質の高さを維持することの代償として，定期的にグループ化のルールを人間の手で微調整し，Web ページのそのつどの現状に適合するように正すことが必要となるように思われる．

1.3 研究の特色

本研究は，以上の既存研究とは異なり，有向グラフに特有な概念である強連結成分分解を出発点とし，リンク情報のみに依拠したグループ化を提案する．

確かに，強連結成分分解そのものが，有向グラフの頂点のグループ化を与えてはいる．また，強連結成分分解は，少ない時間および空間計算量で遂行可能である [15]．しかしながら, [3] が実験的に確かめているように，Web のリンク構造上で強連結成分分解を行うと，一つだけ巨大な成分が構成されてしまう．また，相対的に大きなサイズの成分も多数構成される．さらに [5] の結果を利用すれば，Web のリンク構造では，一つだけ巨大な強連結成分が存在することはほぼ確実であること，また，この最大成分のサイズが非常に高い確率で $\Theta(n)$ となってしまうことも言える．n は Web ページの総数である．このように巨大なグループを構成してしまうグループ化の手法は，一つのグループが種々雑多なページを含むと予想されるため，Web グラフ上での検索やマイニングにおける単位としてのグループを構成するには適切でない．

そこで，本研究では，**強連結成分の細分化**としてのグループ化手法を提供することを目指す．まず，リンク情報のみに基づき，Web ページ間の距離の概念を導入する．具体的には，一つのページから別の一つのページへの，向きのついた移行のしやすさの指標として**ドリフト**という概念を定義する．そして，このドリフトに基づいて，二つのページ間の距離をあらわす概念である**相互リンク距離**を定義する．Web ページのグループ化は，相互リンク距離の意味で近くにあるページをまとめることで実現される．グループ化のアルゴリズムは以下のとおりである．まず，任意に一つのページを選び出し，これを今から構成しようとしているグループの中心をなすページとする．そして，当のページからの相互リンク距離が所与のパラメータ τ 以下であるページだけを，一つのグループとしてまとめ上げる．この操作は，すべてのページがいずれかのグループに属するまで繰り返される．値 τ は，グループ化のアルゴリズム全体に対するパラメータとされているので，この値を調節することで，得られるグループの粒度を制御できるようになっている．パラメータ τ を小さく設定すれば，ごく近くのページしかまとめられないため，グループの粒度は小さくなる．τ を大きく設定すれば逆の結果を得る．このパラメータ τ を，**閾値パラメータ**と呼ぶ．本研究の提案するグループ化手法において，ヒューリスティックに決定されるべき事項は，閾値パラメータの設定だけである．さらに，閾値パラメータ τ をある決まった値以上に設定すると，アルゴリズムは強連結成分分解を与える．つまり，本研究の提案するグループ化手法は，この意味で強連結成分分解の一般化とみなされうる．そこで，このアルゴリズムによって構成される連結成分を，**パラメータ化された連結成分 (parameterized connected component)** と呼び，PCC と略記する．本研究の特色をまとめると以下のようになる．

- ハイパーリンクから得られる情報のみに基づいて Web ページをグループ化する．
- グループ化の理論的枠組みとして，ドリフトおよび相互リンク距離という新しい概念を導入している．
- グループの粒度を，閾値パラメータと呼ばれるパラメータの値を調整することで制御できる．
- PCC は，強連結成分の一般化とみなすことができる．

本論文の構成は以下のとおりである．2. 節では，グループ化手法の理論的枠組みについて述べる．3. 節では，アルゴリズムの詳細とその実装方法および改良の可能性について議論する．4. 節では，1500 万の Web ページを対象に行なったグループ化の実験の結果と，その評価の試みを示す．最後に 5. 節でまとめと今後の展望を述べる．

2. 新しい概念

2.1 ドリフト

Web ページは，ハイパーリンクを介した参照関係によって一つの巨大な有向グラフを形成している．この有向グラフ $G = (V, E)$ の頂点集合 V は Web ページの集合であり，有向枝の集合 E はハイパーリンクの集合である．$|V| = n, |E| = m$ とする．なお，頂点の集合 V を，自然数の集合 $\{1, \ldots, n\}$ として表し，各頂点を便宜上そのインデックスと同一視する．

有向枝（以下「枝」と略記する）は，順序つきの頂点対として表される．頂点 u から v へ張られた枝は (u, v) と表される．枝には**重み** $w_{u,v}$ が与えられており，すべての $(u, v) \in E$ について $0 < w_{u,v} < 1$ が満たされるとする．直感的には，重みは，頂点 u から v への関連の度合いが高いほど大きくなるように定める．また，$0 < c < 1$ なる定数 c について $\overline{w_{u,v}} \equiv \log_c w_{u,v}$ と定義される値を枝 (u, v) の**対数重み (logarithmic weight)** と呼ぶ．$0 < c < 1$ より，重みが大きいほど対数重みは小さくなる．$(u, v) \notin E$ のときは $w_{u,v} = 0$ および $\overline{w_{u,v}} = \infty$ とする．重み $w_{u,v}$ を第 u 行，第 v 列のエントリとする $n \times n$ の行列 W を**重み行列**と呼ぶ．

歩道 (walk) は，順序つきの枝集合 $\pi = ((u_1, v_1), ..., (u_l, v_l))$ で，$v_1 = u_2, v_2 = u_3, \ldots, v_{l-1} = u_l$ を満たすものをいう．頂点の重複がない歩道を特に**パス (path)** と呼ぶ．**閉路 (cycle)** とは，始点と終点が一致する歩道のことである．

本研究では，歩道 π の重みを，歩道に含まれる枝の重みの積として定義し，w_π と記す．つまり，歩道 $\pi = ((u_1, v_1), ..., (u_l, v_l))$ の重みは，$w_\pi = w_{u_1,v_1} w_{u_2,v_2} \cdots w_{u_l,v_l}$ となる．また，歩道の対数重みとは，歩道の重みについて底が c の対数をとった値とする．このとき，明らかに，歩道の対数重みは，歩道を構成する枝の対数重みの総和となる．なお，単に歩道の長さと言うときは，その歩道を構成する枝の数を意味する．つまり，歩道 $\pi = ((u_1, v_1), ..., (u_l, v_l))$ の長さは l である．

ここで**ドリフト (drift)** という新しい概念を導入する．頂点 u から v へのドリフト $Dr(u, v)$ とは，あらかじめ定められた演算子 $\bigoplus$ にしたがって，u から v へのすべての歩道の重みを集計したものをいう．つまり，頂点 u から v へのすべての歩道の集合を $\Pi(u, v)$ とすると，u から v へのドリフトは

$$Dr(u, v) \equiv \bigoplus_{\pi \in \Pi(u,v)} w_\pi$$

と定義される．また，頂点 u から v への**対数ドリフト (logarithmic drift)** $\overline{Dr(u, v)}$ を，$\overline{Dr(u, v)} \equiv \log_c Dr(u, v)$ と定義する．直感的に言えば，頂点 u から v へのドリフトが大きいほど（対数ドリフトが小さいほど），u から v への関連の度合いが高い，ということになる．

2.2 相互リンク距離

頂点 u と v の間の**相互リンク距離 (mutual-link distance)** $ML(u, v)$ は，c を $0 < c < 1$ を満たす定数として

$$\begin{aligned} ML(u, v) &\equiv \log_c Dr(u, v) + \log_c Dr(v, u) \\ &= \overline{Dr(u, v)} + \overline{Dr(v, u)} \end{aligned}$$

と定義される．頂点集合 $S \subset V$ の**直径**を，S に属する二つの頂点間の有限な相互リンク距離の最大値と定める．ここで，具体的なドリフトの決め方，つまりは相互リンク距離の決め方を，いくつか挙げる．

(i) 演算子 $\bigoplus$ を，総和をとる操作と定める．このとき，$Dr(u, v)$ は，$n \times n$ 行列 $\sum_{k=1}^{\infty} W^k$ の第 u 行，第 v 列のエントリに一致する．よって，任意の頂点間の相互リンク距離を求めるには，$\sum_{k=1}^{\infty} W^k$ さえ計算しておけばよい．ただし，枝の重みは，この和が収束するように正規化しておく．

このように，重み行列の冪の和によって，ある頂点から別の頂点への関連性の度合いを表現する試みは，すでに Katz によって行われている [11]．だが，今回の実装においては，スケーラビリティの確保という観点から，Katz による方法で相互リンク距離を定めることはしなかった．なぜなら，$\sum_{k=1}^{\infty} W^k = (I - W)^{-1} - I$ より，重み行列の冪の和に基づいて相互リンク距離を求めるには，逆行列の算出というコストの大きな処理が必要となるからである．しかしながら, [21] のように，リンク解析に伴う莫大な数値計算を，潤沢な計算機資源と効果的な実装によって遂行する試みもあるため，この相互リンク距離の決め方が常に非現実的であるわけではない．

(ii) 枝の重みをすべて c とし，演算子 $\bigoplus$ を最大値をとる操作とする．このとき，すべての枝の対数重みが 1 となり，$\overline{Dr(u, v)}$ は u から v への歩道のうち最も短いものの長さに一致する．したがって，$ML(u, v)$ は，頂点 u から v への最短パス長と，v から u への最短パス長との和となる．つまり，任意の頂点間の相互リンク距離を求めるには，任意の頂点間で，一方から他方へのパスの最短の長さを求めればよいことになる．しかし，実際の Web グラフ上で，様々な頂点対についてここで定められた相互リンク距離を求めると，非常に近い距離にたくさんのページが密集している．このため，単に双方向の最短パス長の和によって相互リンク距離を定めると，きめの細かいグループ化を実現し難い．したがって，今回はこの設定も採用しなかった．

(iii) 枝 (u, v) の重みを $w_{u,v} = c^{d_u^+}$ とする．d_u^+ は頂点 u の出次数である．つまり，枝の対数重みが，その枝の始点の出次数に一致するように重みを決める．そして，演算子 $\bigoplus$ は，(ii) と同様，最大値をとる操作とする．このとき，$\overline{Dr(u, v)}$ は，u か

ら v への歩道のうち，それに沿って存在する頂点の出次数の和が最小のものの，その最小値に一致する．つまり，

$$\overline{Dr(u,v)} = \min_{\pi \in \Pi(u,v)} \left(\sum_{w \in V \text{ s.t. } (w,w') \in \pi} d_w^+ \right)$$

が成り立つ．したがって，$ML(u,v)$ は，頂点 u と v を含む閉路のうち，それに沿って存在する頂点の出次数の和が最小のものの，その最小値に一致する．今回はこの設定を採用した．これによって，(ii) の場合のように，単なる最短パスの長さという観点からごく近くにあるたくさんの頂点の中にも，パスに沿った出次数の和の最小値という観点からは微細な遠近のグラデーションがつき，よりきめの細かいグループ化が可能となった．

なお，この設定の下では，出次数の大きい Web ページが同じグループに属しにくくなる．そのため，以下のような効果も期待できる．つまり，インデックス的な役割を果たすページや，リンク集的な役割を果たすページなどが，相互に排除し合い，結果として，局所的に見れば小さなディレクトリ構造をしているような Web ページ群が，一つのグループとしてまとまりやすくなることが期待される．

ところで，(i) から (iii) のいずれの設定を用いても，相互リンク距離は三角不等式を満たすことが証明できる．実際，任意の三つの頂点 u, v, w について，$\Pi(u,w)$ に含まれる歩道と $\Pi(w,v)$ に含まれる歩道とを連結した歩道の集合は，$\Pi(u,v)$ の部分集合になっており，したがって，

$$Dr(u,v) \geqq Dr(u,w) \cdot Dr(w,v)$$

がすべての $w \in V$ について成立する．よって，

$$\begin{aligned} ML(u,v) &= \log_c Dr(u,v) + \log_c Dr(v,u) \\ &\leqq \log_c (Dr(u,w) \cdot Dr(w,u)) \\ &\quad + \log_c (Dr(v,w) \cdot Dr(w,v)) \\ &= \log_c Dr(u,w) + \log_c Dr(w,u) \\ &\quad + \log_c Dr(v,w) + \log_c Dr(w,v) \\ &= ML(u,w) + ML(v,w) \end{aligned}$$

が任意の $w \in V$ について言える．なお，演算子 $\bigoplus$ が和をとる操作の場合は，どの枝の対数重みも負でないという事実を利用する必要がある．

2.3 パラメータ化された連結成分

パラメータ化された連結成分 (PCC) とは，**閾値パラメータ (threshold parameter)** と呼ばれるパラメータ τ の値に応じて，次の条件を満たすように構成された頂点の集合 $S \subset V$ のことをいう．つまり，$S \subset V$ について，**中心頂点**と呼ばれる頂点 $u \in S$ が存在し，閾値パラメータ τ に対して，$ML(u,v) \leqq \tau$ がすべての $v \in S$ について成立するとき，このような S を PCC と呼ぶ．

2.2 節の (ii) の設定のもとでは，少なくとも $\tau \geqq n$ ならば極大な PCC による頂点集合 V のグループ化は強連結成分分解に一致する．また，今回の実装のように (iii) の設定を採った場合も，少なくとも $\tau \geqq \sum_{v \in V} d_v^+$ ならば同じく強連結成分分解を得る．(i) の設定のもとであっても，τ を十分大きくとれば，やはり極大な PCC は強連結成分に一致する．このように，パラメータ化された連結成分への分解は，強連結成分分解の一般化とみなすことができる．ただし，与えられた τ に対する PCC への分解は，強連結成分分解とは異なり，一意に定まらない．なぜなら，頂点 u が二つの異なる頂点 v_1, v_2 から τ 以下の相互リンク距離にある場合，u は v_1 を中心頂点とする PCC にも，v_2 を中心頂点とする PCC にも属しうるからである．このとき，本研究の提案するアルゴリズムでは，u は v_1 と v_2 のうち先に中心頂点として選ばれた頂点の PCC に属するものとされる．

なお，1.3 節で述べたように，本研究の提案するグループ化手法によれば，いずれのグループも，一つの頂点からの相互リンク距離が τ 以下となるように構成される．したがって，この構成法と，相互リンク距離が三角不等式を満たすことから，グループの直径の上界が 2τ となることが帰結する．つまり，相互リンク距離が三角不等式を満たすということが，直径が一定の値以下の PCC のみが構成されることを保証する．

2.4 ランダムウォークとしてのネットサーフィン

ここでは，本研究の理論的枠組みの提示の締めくくりとして，ドリフトが，ネットサーフィンをランダムウォークとして理解する文脈においても意味をもつ概念であることを述べる．

検索エンジン Google に利用されている PageRank [16] という手法は，ネットサーフィンをランダムウォークとみなす考え方に基づいている．そこでは，ある Web ページからの遷移の確率が，基本的にはそのページの出次数の逆数と定められる．また, [2] によれば，Hub/Authority [12] もまたランダムウォークのモデルに基づく手法として整理することができる．そして，本研究の提案するドリフトという概念もまた，やはりランダムウォークの枠組みの中で理解することができる．

すなわち，枝 (u,v) に対して与えられた重み $w_{u,v}$ を，頂点 u から v へと移動する確率と解釈する．ただし，一つの頂点から出る枝の重みの総和が 1 以下となるように，重みを正規化しておく．また，一つの頂点から出る枝の重みの総和が 1 より小さい場合は，遷移の終了を表す仮想的な頂点を一つ設け，1 からその総和を引いた残りの確率でこの終了状態へ遷移すると定める．すると，歩道の重みが歩道を構成する枝の重みの積として定義されていることに鑑みれば，ドリフトは，ある頂点から別の頂点への遷移が生じる確率に依存して定まる量とみなされうる．例えば，上記 (ii) の設定のもとでは，ドリフト $Dr(u,v)$ は，u から v へと最短パスをたどって推移する確率そのものとなる．

3. アルゴリズムとその実装

3.1 アルゴリズムの詳細

本研究の提案するグループ化アルゴリズムの，現時点での実装においては，任意に選ばれた中心頂点から幅優先探索を行うことによって，それに沿った頂点の出次数の和が閾値パラメータ τ 以下である閉路をすべて列挙する．そして，列挙された閉路上にあるすべての頂点を，探索の始点である中心頂点と同じ

PCC に属するものとする．下にアルゴリズムの詳細を示す．

（1） すべての頂点がマークされていない状態にする．

（2） 任意に一つの頂点 $u \in V$ を選び出し，u 自身を中心頂点とする PCC の構成員としてマークする．

（3） u から幅優先探索を行う．ただし，幅優先探索は，u を出発点とするパス上に存在する全頂点の出次数の和が τ 以下である範囲内で継続する．つまり，τ を超えた時点で探索木の枝刈りを行う．

（4） 幅優先探索を行っている最中に，中心頂点 u にもどってきたら，その閉路上にある全頂点を，u を中心頂点とする PCC の構成員としてマークする．

（5） すべての頂点が，いずれかの PCC の構成員としてマークされるまで，上の 2 から 4 を繰り返す．

上記のアルゴリズムとその実装方法についていくつかコメントを掲げる．

- ステップ 2 で，中心頂点は任意に選ばれる．中心頂点を選ぶ順が異なると，得られる PCC も異なりうる．つまり PCC への分解は一意に定まらない．しかし，2.3 節で述べたように，いずれの PCC の直径も 2τ 以下であることは保証されている．
- 本アルゴリズムは並列化できる．なぜなら，様々な中心頂点からの幅優先探索を並列に実行できるからである．ただし，異なる実行インスタンスによって同じ頂点が別々の PCC に属するとされた場合は，後処理によっていずれか一つの PCC のみに属するように定める．今回の実装では，最も近いインデックスを持つ中心頂点の PCC へ含ませるようにしている．
- 幅優先探索を行なうには，リンク情報，すなわち，どの頂点が他のどの頂点への枝を持っているかの情報が必要である．しかし，Web ページ数が膨大であると，すべてのリンク情報をメモリ上に保存することが困難となる．そこで，より少ないメモリで，より多くのリンク情報を格納できるようにする工夫が欠かせない．今回の実装では，頂点のインデックスを利用し，リンク情報を二通りに場合分けして格納した [17]．一つは，リンク先の頂点のインデックスをそのまま保存する場合．もう一つは，リンク先の頂点のインデックスを，リンク元の頂点のインデックスとの差分で保存する場合．そして，インデックスの差が 1 バイトの場合のみ，後者の仕方で保存することにした．これによって，すべてのリンク先頂点のインデックスをそのまま保存するよりも，メモリ使用量を減らすことができる．
- ステップ 3 における幅優先探索においては，同じ頂点が何度も訪問される．なぜなら，二頂点間に存在する複数のパスのうち，それに含まれる枝の数が少ないという意味でより短いものが，それに含まれる頂点の出次数の和についてもより小さい値をとるとはかぎらないからである．このように，上記のアルゴリズムでは，中心頂点からの探索において，同じ頂点を繰り返し訪問するような幅優先探索をおこなっているため，その時間計算量については，出次数の和による枝刈りを行ってはいるものの，閾値パラメータ τ を増やすにつれて実行時間が急速に増大する．そこで，時間計算量削減のための改良について，節をあらためて論じる．

3.2 APSP 問題としてのドリフト算出

3.1 節で示したアルゴリズムは，中心頂点からの探索において，特に何の戦略にも従わず，単純な幅優先探索を行っているが，この点については改良の余地がある．

今回の実装では，$\bigoplus$ を最大値をとる操作としてドリフトを定義している．このとき，対数ドリフト $\overline{Dr(u,v)}$ は，u から v への歩道のうち，それを構成する有向枝の対数重みの和が最小のものの，その最小値となる．したがって，任意の頂点間について相互リンク距離を求める問題は，対数重みによって枝が重み付けられた有向グラフ上で，APSP(all pairs shortest paths) 問題を解く問題へ帰着できる．なぜなら，APSP 問題とは，任意の順序つき頂点対 u, v について，u から v への重み最小のパスを求める問題だからである．つまり，PCC への分解としての Web ページのグループ化は，APSP 問題より難しくない．

ところで，APSP 問題は，通常，個々の頂点についての SSSP(single source shortest path) 問題へと分解して解かれる．SSSP 問題とは，与えられた頂点から他の全頂点への重み最小のパスを求める問題である．そして，SSSP 問題の古典的解法としては Dijkstra によるものがある [8]．近年，より効率的なアルゴリズムが提案されているが，基本的には Dijkstra のアイディアを反映している [9]．つまり，与えられた頂点から始めて，一定の戦略に従って，全頂点におよぶ探索を行う．このとき，効果的な戦略を採ることによって，時間計算量や空間計算量を削減できる．Dijkstra [8] は近い頂点から順に訪問するという戦略が採っていたが, [9] ではより高度な戦略を用い，枝の重みが非負の整数の場合に，時間計算量 $O(n + m \log\log n)$，空間計算量 $O(n+m)$（ただし出力サイズ分は除く）で SSSP を解くアルゴリズムが提案されている．ところで, [4] [13] によれば，Web のリンク構造を表すグラフの場合，$m = O(n)$ となる．また，今回の実装で採用した 2.2 節の設定 (iii) のもとでは，どの枝の対数重みも非負の整数値である．なぜなら，枝 (u, v) の対数重みは u の出次数だからである．よって，全頂点間の相互リンク距離を，SSSP 問題を n 個ある頂点のそれぞれについて解くことによって求めても，時間計算量は高々$O(n^2 \log\log n)$ である．これが PCC への分解としての Web ページのグループ化に要する時間計算量の上界である．

しかし，$O(n^2 \log\log n)$ という値はあくまでも上界である．なぜなら，PCC の構成においては，グループの大きさを制御するパラメータとして前もって τ が与えられており，始点となっている頂点からの相互リンク距離が τ 未満の頂点へのみ，探索が及べばよいからである．つまり，中心頂点からの探索を賢い戦略に基づいて行ったうえで，さらに，中心頂点からの相互リンク距離が τ を超えた時点で枝刈りを行えば，実際の計算時間はさらに短縮されうる．

4. 実　　験

今回は，3.2 節に示した改良は実装せず，3.1 節に示した単純な幅優先型のアルゴリズムをそのまま実装し，クローリングによって集められた 15,000,000 の Web ページに対してグループ

化の実験を行った．実験環境は，Intel Xeon プロセッサを 2 つ搭載したメモリ 2GB の Solaris マシン 8 台によって構築した．並列化にともない，異なるインスタンス間でデータをやりとりする機構を実現するために，今回の実装ではマルチキャストを利用した．もちろん，単に頂点集合を等分して各インスタンスに分配し，それぞれのインスタンスが与えられた頂点集合からのみ中心頂点を選び幅優先探索をすれば，これだけで通信の機構なしに並列化は可能である．しかし，頂点集合を等分したからといって，すべての処理が同時に終わるわけではない．そこで，遅れているインスタンスがやり残している仕事を，別のインスタンスが手伝うという効果を得るために，マルチキャストを利用することにした．これにより，全体の実行時間が短縮される．実際, [5] に示されているように，Web のリンク構造のような有向グラフでは，枝をたどることによって到達可能な頂点の個数については，それが $\Theta(n)$ にも達する頂点がある一方，高々$O(\log n)$ 程度にとどまる頂点もある．したがって，幅優先探索がどの程度多くの頂点をカバーするかについては，探索の出発点としてどの頂点を選ぶかによってかなり差が生じる可能性がある．つまり，あらかじめ頂点集合を全実行インスタンスに均等に分配したからといって，処理が同時に終わるとは限らない．そこで，すでにいずれかの PCC に属している頂点の ID をマルチキャストし，他のマシン上で実行中のインスタンスが，同じ頂点を別の PCC の構成員として選ばないようにした上で，自分の担当範囲の終わったインスタンスは，本来他のインスタンスの担当分である頂点からの探索を横取りして実行するようにした．なお，実際の実行時間は，閾値パラメータが 50 の場合は 24 時間 10 分，100 の場合 24 時間 50 分，150 の場合 40 時間 45 分，300 の場合は 8 日と 8 時間だった．

表 1 は，閾値パラメータ τ の増加に伴う最大 PCC のサイズの変化を示している．また，図 1 は，PCC のサイズとその個数の分布である．τ の増加に伴いグループの粒度が大きくなっていることが分かる．下に，得られたグループのいくつかについて，具体的な構成員を URL で示す．

```
...
----
http://haya.town.hayakawa.yamanashi.jp/H-KANKO/YACYO/index.html
http://haya.town.hayakawa.yamanashi.jp/H-KANKO/YACYO/torisan.html
----
http://haya.town.hayakawa.yamanashi.jp/H-MATSURI/GENKIMURA/genki.html
http://haya.town.hayakawa.yamanashi.jp/H-MATSURI/event.html
----
http://haya.town.hayakawa.yamanashi.jp/H-MATSURI/indo-ten.html
http://haya.town.hayakawa.yamanashi.jp/JORYU/J-RESEARCH/RYEAR/resea1998.html
http://haya.town.hayakawa.yamanashi.jp/JORYU/J-RESEARCH/resealist.html
http://haya.town.hayakawa.yamanashi.jp/JORYU/J-RESEARCH/research.html
http://haya.town.hayakawa.yamanashi.jp/JORYU/joryu1.html
http://haya.town.hayakawa.yamanashi.jp/INDIA/yatteru.html
----
http://haya.town.hayakawa.yamanashi.jp/INDIA/dekigot2.html
http://haya.town.hayakawa.yamanashi.jp/INDIA/dekigoto.html
----
http://haya.town.hayakawa.yamanashi.jp/INDIA/mithila1.html
http://haya.town.hayakawa.yamanashi.jp/INDIA/mithila2.html
----
http://haya.town.hayakawa.yamanashi.jp/JORYU/J-2000/2000.html
http://haya.town.hayakawa.yamanashi.jp/JORYU/J-2000/2000foot.html
----
...
----
http://te.tec.fukuoka-u.ac.jp/4gou.html
http://te.tec.fukuoka-u.ac.jp/Campus.htm
http://te.tec.fukuoka-u.ac.jp/index.html
http://te.tec.fukuoka-u.ac.jp/map.html
----
http://te.tec.fukuoka-u.ac.jp/matumoto/RDEM4/dvd2avi.html
http://te.tec.fukuoka-u.ac.jp/matumoto/RDEM4/rdem4.html
----
http://te.tec.fukuoka-u.ac.jp/matumoto/labo/labo.html
http://te.tec.fukuoka-u.ac.jp/matumoto/labo/labo1.html
http://te.tec.fukuoka-u.ac.jp/matumoto/labo/labo2.html
----
http://te.tec.fukuoka-u.ac.jp/matumoto/movie/movie.html
http://te.tec.fukuoka-u.ac.jp/matumoto/movie/movielink.html
----
http://te.tec.fukuoka-u.ac.jp/matumoto/report/1999.html
http://te.tec.fukuoka-u.ac.jp/matumoto/report/report.html
----
http://te.tec.fukuoka-u.ac.jp/matumoto/student/sakanoue/daijiten.html
http://te.tec.fukuoka-u.ac.jp/matumoto/student/sakanoue/sakanoue.html
----
...
----
http://voice-inc.co.jp/aura-soma/Products/Pomanders/Application.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/DeepMagenta.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/DeepRed.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/Emerald.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomGold.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomOlive.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomOrange.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomPink.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomRoyal.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomTurqu.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomViolet.html
http://voice-inc.co.jp/aura-soma/Products/Pomanders/PomYellow.html
----
http://voice-inc.co.jp/aura-soma/Products/Quintessences/Application.html
http://voice-inc.co.jp/aura-soma/Products/Quintessences/EmerQuin.html
http://voice-inc.co.jp/aura-soma/Products/Quintessences/PaleOrange.html
http://voice-inc.co.jp/aura-soma/Products/Quintessences/PinkQuin.html
http://voice-inc.co.jp/aura-soma/Products/Quintessences/RedQuin.html
http://voice-inc.co.jp/aura-soma/Products/Quintessences/Rose.html
----
http://voice-inc.co.jp/aura-soma/Publications/Books.html
http://voice-inc.co.jp/aura-soma/Publications/Home.html
http://voice-inc.co.jp/aura-soma/Publications/Tarot.html
----
http://voice-inc.co.jp/eleaning/ecton.html
http://voice-inc.co.jp/eleaning/index.html
----
...
----
http://www.brazil.ne.jp/paraiso/portugues/apoio/index.html
http://www.brazil.ne.jp/paraiso/portugues/compra/index.html
http://www.brazil.ne.jp/paraiso/portugues/contato/index.html
http://www.brazil.ne.jp/paraiso/portugues/ecompra/index.html
http://www.brazil.ne.jp/paraiso/portugues/index.html
http://www.brazil.ne.jp/paraiso/portugues/papel/index.html
http://www.brazil.ne.jp/paraiso/portugues/proposta/index.html
----
http://www.brazil.ne.jp/port/expediente/index.html
http://www.brazil.ne.jp/port/index.html
http://www.brazil.ne.jp/port/press/001.html
http://www.brazil.ne.jp/port/press/002.html
http://www.brazil.ne.jp/port/press/003.html
http://www.brazil.ne.jp/port/press/004.html
http://www.brazil.ne.jp/port/press/005.html
http://www.brazil.ne.jp/port/press/006.html
http://www.brazil.ne.jp/port/press/index.html
----
http://www.brazil.ne.jp/present/index.html
http://www.brazil.ne.jp/present/win/index.html
----
http://www.brazil.ne.jp/press/001/index.html
http://www.brazil.ne.jp/press/002/index.html
http://www.brazil.ne.jp/press/index.html
----
http://www.brazil.ne.jp/press/003/index.html
http://www.brazil.ne.jp/press/004/index.html
----
...
----
http://www.city.shiroishi.miyagi.jp/kuzu/kuzu_dekirumade.htm
http://www.city.shiroishi.miyagi.jp/kuzu/kuzu_fukkatu.htm
http://www.city.shiroishi.miyagi.jp/kuzu/kuzu_index.htm
http://www.city.shiroishi.miyagi.jp/kuzu/kuzu_ryouri.htm
http://www.city.shiroishi.miyagi.jp/kuzu/kuzu_saibai.htm
----
http://www.city.shiroishi.miyagi.jp/product/pro_01.html
http://www.city.shiroishi.miyagi.jp/sight/sig_01.html
----
http://www.city.shiroishi.miyagi.jp/shinko/sanada/1.htm
http://www.city.shiroishi.miyagi.jp/shinko/sanada/mokuji.htm
----
http://www.city.shiroishi.miyagi.jp/sight/sig_21.html
http://www.city.shiroishi.miyagi.jp/sight/sig_22.html
----
http://www.city.shiroishi.miyagi.jp/siro_kankoumap/chizu/kamsaki.html
http://www.city.shiroishi.miyagi.jp/siro_kankoumap/chizu/map3.html
----
http://www.city.shiroishi.miyagi.jp/siro_kankoumap/chizu/map4.html
http://www.city.shiroishi.miyagi.jp/siro_kankoumap/syoukai_kankou.htm
----
...
```

多くの場合，あたかも URL について [20] に提案されているようなヒューリスティクスを適用したかのように，ディレクトリ構造に比較的忠実にグループ化が行われている．また，異なるサーバの Web ページは，ほとんどの場合，異なるグループに属している．以上の結果は，今回のグループ化が，URL に関するヒューリスティクスを場当たり的に案出する手間を省きつつ，組織的にサーバ内のグループ化を実現するための手法としての利用価値も持つことを意味する．さらにこの結果は，どのようなサーバについてもそれを一つの単位とみなす考え方が必ずしも正しくなく，同じサーバ内であっても，希薄な関連性を表すリンクと密接な関連性を表すリンクとの区別があることを予想させる．

次に，得られたグループの評価に関しても，一つの試みを

行った．今回の評価に際しては，次のような基準を仮に設定した．つまり，リンク構造上をランダムウォークするとき,（1）異なるグループ間のリンクを優先して辿る場合と,（2）どのリンクも差別なく辿る場合とで，決まったステップ数内でどれだけ遠くのページまで行けるかにおいて，本質的な差が出るか否か，という基準である．この基準を設けた根拠は以下のとおりである．

- ネットサーフィンは，今いる Web ページ上にあるリンクをランダムに選んでクリックするというランダムウォークによって近似できる．
- 特定の分野のページだけを閲覧したいという目的ではなく，どんな分野のページであれ可能なかぎり雑多なページを閲覧したいという目的で行われるネットサーフィンもある．
- 上述の種類のネットサーフィンにおいては，同じページに何度も戻ってくるのを避けることが望ましい．
- ランダムウォークにおいて，同じステップ数内で，出発点となるページからより遠くにあるページ（＝そこへ至るまでに辿るべきリンクの数がより多いページ）へ行けるならば，そのようなランダムウォークのほうが，同じページに戻ってくることがより少ない．

評価のための実験においては，出発点となるページを 15,000,000 の Web ページからランダム選び，そこから，ステップ数が 100 のランダムウォークを，次の二通りの方法で実行した．

（1） 未訪問の**グループ**につながるリンクがあれば，それらの中からたどるべきリンクをランダムに選ぶ．そうでなければ，たどるべきリンクをすべてのリンクからランダムに選ぶ．

（2） 未訪問の**ページ**につながるリンクがあれば，それらの中からたどるべきリンクをランダムに選ぶ．そうでなければ，たどるべきリンクをすべてのリンクからランダムに選ぶ．

前者をランダムウォーク１，後者をランダムウォーク２と呼ぶ．なお，グループ化の効果を反映しているのは，ランダムウォーク１である．また，リンクのないページに行き着いたら，前にいたページに戻ることとする．

さて，以上二種類のランダムウォークを，閾値パラメータ τ が 50,100,150,200,300 の場合のそれぞれについて，4000 回行った．そして，100 ステップ内で訪れたページのうち，出発点からの最短パス長（＝そこへ至るまでに辿るべきリンク数の最小値）が最大のもののその最大値を，ランダムウォーク１とランダムウォーク２とで比較した．表 2 に結果を示す.「グループ化を利用」の行は，グループ化の結果を利用したランダムウォーク１のほうが，そうでない場合よりも遠くのページに辿り着けた場合が，全 4000 回のランダムウォークのなかで何回あったかを，それぞれの閾値パラメータの値について示している．同様に,「グループ化を利用せず」の行は，グループ化を利用しないランダムウォーク２のほうが，より遠くのページに辿り着けた場合の数を示している.「差が出なかった」の行は，どちらの仕方でランダムウォークをしても，どれだけ遠くのページに辿り着けたかにおいて違いのなかった場合の数を示している．

実際に行われたランダムウォークの履歴を見ると，出ていくリンクのないページが相当数あるため，前にいたページに戻っては再び訪問済みのページに行く，という繰り返しが頻繁に起こっていた．そのため，差が出なかった場合の数が非常に多くなってしまった．この点については，評価方法の再検討が必要であろう．また，閾値パラメータの影響については，それが増大するにともなって，グループ化の効果が顕著になっていることが観察された．だが，より大きな閾値パラメータでの評価も今後必要と思われる．

閾値パラメータ τ	50	100	150	200	300
最大 PCC のサイズ	633	1206	1292	1286	1890

表 1　**最も大きい PCC のサイズと閾値パラメータとの相関**

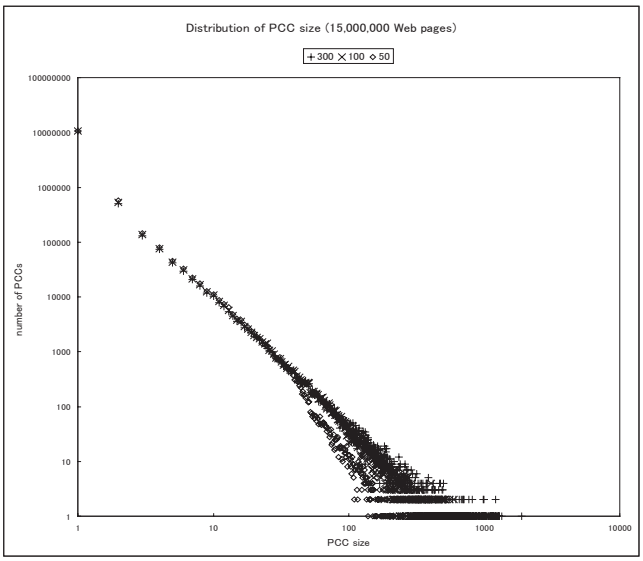

図 1　**PCC のサイズと個数の分布:** グラフは閾値パラメータが 300, 100, 50 それぞれの場合のグループのサイズと個数の分布を表す．横軸が PCC のサイズ．縦軸が PCC の個数．

5.　まとめと今後の課題

本研究では，リンク情報のみに基づいて Web ページのグループ化を実現する手法を提案した．また，実装レベルでは，マルチキャストを利用した並列化にも成功した．しかし，アルゴリズムの高速化のためには，3.2 節で述べた工夫を盛り込むことが今後の課題として残されている．また，実験の結果からは，同一サーバ内であっても，リンク構造上区別すべきいくつかの部分構造が含まれていることが示唆された．さらに，得られたグループ化について，ネットサーフィンにおいてどれだけ遠くまで辿り着けるか，という基準の下に評価を試みた．しかし，今回の評価基準がそもそも妥当であるかという点については，再検討する予定である．最後に，提案手法をテキスト情報に基づく Web 検索に組み合わせて使う場合，どのようにすれば検索性能の向上につながるかを考えることが，現実的な課題として残されていることを述べておきたい．

謝　　辞

本論文の査読に貴重な時間を割いてくださった査読者の皆様

閾値パラメータ τ	50	100	150	200	300
グループ化を利用	1243	1280	1390	1384	1394
グループ化を利用せず	1097	1038	981	971	949
差が出なかった	1660	1682	1629	1645	1657

表 2　**グループ化の効果と閾値パラメータとの相関**

に，心より感謝申し上げます，また，国立情報学研究所の大山敬三教授のご協力がなければ，今回の計算機実験は実現できませんでした．なお，本研究は文部科学省科学研究費補助金特定領域研究「情報学」の助成のもとに行われています．

文　献

[1] Michael W. Berry, Susan T. Dumais, and Gavin W. O'Brien. Using linear algebra for intelligent information retrieval. Technical Report UT-CS-94-270, Department of Computer Science, University of Tennessee, 1994.

[2] Alan Borodin, Gareth O. Roberts, Jeffrey S. Rosenthal, and Panayiotis Tsaparas. Finding authorities and hubs from link structures on the world wide web. In *Proceedings of the 10th International World Wide Web Conference*, pp. 415–429, 2001.

[3] A. Z. Broder, S. R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. Wiener. Graph structure in the web. In *Proceedings of the 9th International World Wide Web Conference*, pp. 309–320, 2000.

[4] Fan Chung, Bill Aiello, and Linyuan Lu. A random graph model for power law graphs. *Experimental Math.*, Vol. 10, pp. 53–66, 2001.

[5] Colin Cooper and Alan Frieze. The size of the largest strongly connected component of a random digraph with a given degree sequence, *pre-print*. available at `http://www.math.cmu.edu/~af1p/papers.html`, 2002.

[6] Douglass R. Cutting, Jan O. Pedersen, David Karger, and John W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In *Proceedings of the Fifteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 318–329, 1992.

[7] Scott C. Deerwester, Susan T. Dumais, Thomas K. Landauer, George W. Furnas, and Richard A. Harshman. Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, Vol. 41, No. 6, pp. 391–407, 1990.

[8] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numer. Math.*, Vol. 1, pp. 269–271, 1959.

[9] Torben Hagerup. Improved shortest paths on the word ram. In U. Montanari et al., editor, *ICALP 2000, LNCS 1853*, pp. 61–72. Springer-Verlag, 2000.

[10] Ravi Kannan, Santosh Vempala, and Adrian Vetta. On clusterings - good, bad and spectral. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, pp. 367–377, 2000.

[11] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, Vol. 18, pp. 39–43, 1953.

[12] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, Vol. 46, No. 5, pp. 604–632, 1999.

[13] Linyuan Lu. The diameter of random massive graphs. In *Proceedings of the Twelfth ACM-SIAM Symposium on Discrete Algorithms*, pp. 912–921, 2001.

[14] Raymond T. Ng and Jiawei Han. Clarans: A method for clustering objects for spatial data mining. *IEEE Transactions on Knowledge and Data Engineering*, Vol. 14, No. 5, pp. 1003–1016, 2002.

[15] Esko Nuutila and Eljas Soisalon-Soininen. On finding the strongly connected components in a directed graph. *Information Processing Letters*, Vol. 49, No. 1, pp. 9–14, 1994.

[16] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[17] K. Randall, R. Stata, R. Wickremesinghe, and J. Wiener. The link database: Fast access to graphs of the web. Technical Report 175, Compaq Systems Research Center, Palo Alto, CA, 2001.

[18] Leonard J. Schulman. Clustering for edge-cost minimization. *Electronic Colloquium on Computational Complexity (ECCC)*, Vol. 6, No. 035, 1999.

[19] Keishi Tajima, Yoshiaki Mizuuchi, Masatsugu Kitagawa, and Katsumi Tanaka. Cut as a querying unit for WWW, Netnews, and E-mail. In *Proc. of ACM Hypertext '98*, pp. 235–244, June 1998.

[20] Loren Terveen, Will Hill, and Brian Amento. Constructing, organizing, and visualizing collections of topically related Web resources. *ACM Transactions on Computer-Human Interaction*, Vol. 6, No. 1, pp. 67–94, 1999.

[21] 安村賢英, 川原稔, 岩下武史, 金澤正憲. Web コミュニティ発見のための大規模有向グラフに対するデータ圧縮計算手法の vpp への実装. 京都大学大型計算機センター研究開発部 研究発表報告集, 第 17 号, pp. 71–80, 2002.

[22] 浅野泰仁, 今井浩, 豊田正史, 喜連川優. Web リンクの可視化によるグラフ構造の発見. 第 13 回データ工学ワークショップ (DEWS2002), 2002.