Mem. Fac. Sci. Eng. Shimane Univ.
Series A
37, pp. 53-63（2003）

Web グラフの可視化

牧野　周平[1)]，国保　建[2)]，山﨑　美嘉[2)]，藤田　憲悦[1) 2)]

[1)]島根大学大学院　総合理工学研究科
[2)]島根大学総合理工学部　数理情報システム学科

Visualization of Web graph

Shuhei MAKINO　Tatsuru KUNIYASU　Mika YAMASAKI　Ken-etsu FUJITA
Department of Mathematics and Computer Science,
Interdisciplinary Faculty of Science and Engineering, Shimane University

Abstract

多くの Web ページは，関連性のある Web ページにリンクを張っている．Web ページをノード，Web ページ間のリンクをエッジとする有向グラフは Web グラフと呼ばれている．この Web グラフの構造を解析した結果は，Web 情報検索の高速化，効率化に応用されている．本研究では，Web グラフを可視化するシステム（Web-map）を作成した．Web-map を用いて，島根大学内部の URL を対象に実験を行った結果を報告する．

1. はじめに

インターネットの普及により様々な情報が Web 上で入手できるようになった．Web ページの検索においても，情報の量だけでなく，より質の良い情報を検索するとことが重要な課題となっている．本論文では，Web の情報検索において，検索結果の質を向上させるために，現在進行中のプロジェクトで行っている，Web グラフの可視化について紹介する．

WWW(World Wide Web)空間は非常に広く，Web ページ数は日々増加を続けている．Netcraft[1]社が毎月公表しているデータによれば，世界中の Web サーバ数は 2003 年 9 月現在，約 4300 万台である．この Web サーバ数より，NEC 北米研究所の統計調査[8]に基づくと，世界中の Web ページ数は，約 125 億と推測される．従って，Web ページ数は膨大であるため従来どおりの検索法だけでは，検索結果の質を向上させることが非常に困難であると考えられる．

Web ページにおいて，その Web ページと何らかの関連のある Web ページにリンクが張られている． Web ページをノード(頂点)とし，Web ページ間のリンクをエッジ(辺)とする有向グラフは，Web グラフと呼ばれている．この Web グラフの構造を解析した結果は，Web 情報検索の高速化，効率化に応用されている．

本研究では, Web の情報検索を支援するため, 様々な用途で利用されている Web グラフを可視化するシステム Web-map を作成し，実験を行った．本システムにおいて，ユーザが，リンク構造を調べる URL と深さを入力する．次に，入力された URL に接続し，その URL のソースを得る．ソースを解析し，それに含まれているリンク情報を抽出する．そして，抽出されたリンク情報をもとに，リンク先のリンク情報を抽出する．以上の手続きを，入力された深さまで再帰的に繰り返す．最終的に得られたリンク構造の情報を，重複のないように蓄積する．このように蓄積されたデータをもとに，Web グラフを可視化するシステムである．

JAVA 言語により作成したこの Web-map で，島根大学内部の URL を対象に実験を行った．そして，リンク構造を Web グラフとして可視化した．得られた Web グラフについて，統計的にデータ解析した結果を報告する．最後に，今後行うべきシステムの改良，データの様々な解析等の課題についても述べる．

本論文の構成は，次のとおりである．まず２章で，Web グラフについて述べる．３章では，Web グラフの構造解析に

ついて述べる．４章で Web グラフの可視化システムについて説明し，５章では，本プロジェクトの今後の計画，および，まとめについて述べる．

2．Web グラフ

2－1．サーバ数と Web ページ数

Web 空間は非常に広く，Web サーバ数は現在も増加中である．英国の Netcraft[1]社が毎月公表しているデータによれば，Web サーバ数は 2003 年 9 月現在で，約 4300 万台である．図２－１に Web サーバ数の推移を示す．この図は，Netcraft 社が毎月公表している全世界の Web サーバ数と山名早人氏のデータ[2]を元に，1995 年 7 月から 2003 年の 7 月までの Web サーバ数を半年毎にプロットした図である．Web サーバは 1996 年 1 月から急速に増えはじめ，2 年後の 1998 年 1 月には約 30 倍の約 200 万台になった．また 1998 年 1 月から 2 年後の 2000 年 1 月には約 1000 万台となり 2 年間で 5 倍にもなった．

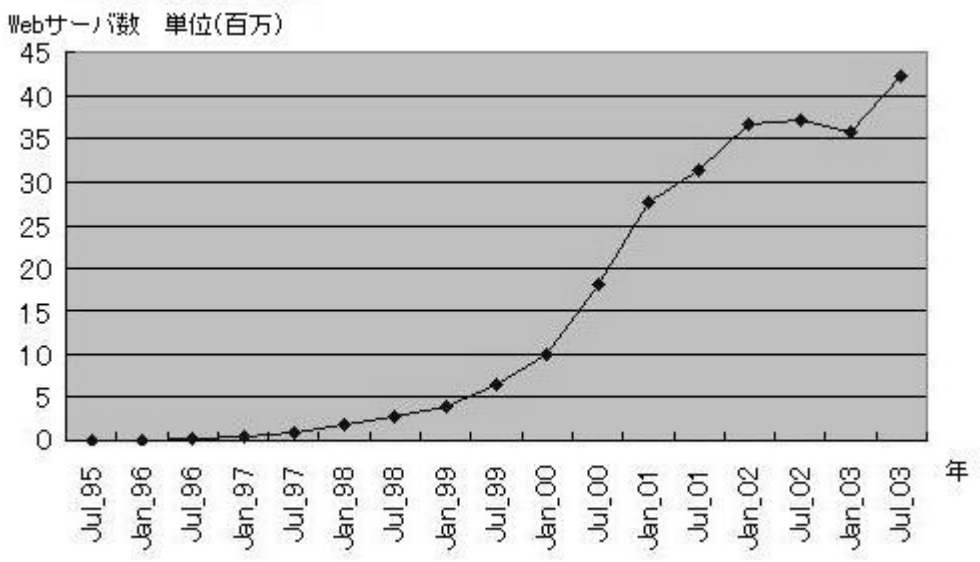

図２－１　Web サーバ数の推移

NEC 北米研究所の Steve Lawrence らの統計的調査[8]より，Web サーバ数と Web ページ数が比例すると仮定すると，2003 年 9 月現在で Web ページ数は約 125 億であると推測される．現在，最も多くの Web ページを収集しているサーチエンジン Google では，推測された Web ページ数の 25%である約 30 億の Web ページから検索が行われる．最も多くの Web ページを収集している Google の Web ページ収集率が全体の 25%であることについて，かなり収集率が低いと思われるかもしれない．これは Web の蝶ネクタイ構造[9]によるものと考えられる．Web の蝶ネクタイ構造については，2 章 4 節で詳しく述べる．

2－2．Web 情報検索の困難さ

サーチエンジンを活用した情報検索では，Web ロボットと呼ばれる Web ページのデータの収集，蓄積を自動的に行うプログラムが使用されている．この Web ロボットが収集，蓄積したデータをもとに，情報検索が行われる．しかし，2－1 で述べた通り，Web ページ数はあまりにも膨大である．また，Web ページは，不定期に更新，削除，移動される．従って，Web ロボットによって収集，蓄積されたデータをもとに，情報検索される結果の質を向上させることは，困難である．質の良い情報を検索するために Web ページのリンク構造を解析して，Web ページの Web 空間における関連性や重要性を求める手法は，数多く提案されている．この手法はリンク接続解析と呼ばれ，検索結果のランキング精度の改善や，関連情報の発見に利用されており，次の 4 種類が知られている．[3]

(1)　Link Popularity

Link Popularity は，数多くリンクされている Web ページほど重要だとみなす手法である．例えば，検索された Web ページの被リンク数，および検索された Web ページが存在するサーバの総被リンク数の多い検索結果を重要とみなす．

(2) HITS(Hypertext Induced Topic Search)

HITS は，ある特定のトピックに関する情報源であるオーソリティ(authority)と，オーソリティへのリンクの集合であるハブ(hub)と呼ばれる 2 種類の Web ページ群に分割する．そして，良いオーソリティは，多くの良いハブからリンクされ，良いハブは多くの重要なオーソリティをリンクするという相互依存する関係を求めることで，検索結果の質を改善する手法である．

(3) PageRank

PageRank は，多くの良質な Web ページからリンクされている Web ページは，良質な情報源であるという考えに基づいて重要度を決める手法である．この手法は，検索エンジン Google で使用されている． 詳しくは，3 章で説明する．

(4) Cocitation アルゴリズム

Cocitation アルゴリズムは，何らかの関連を持つ Web ページをリンク解析で発見する．二つの Web ページを同時に引用している Web ページ群を求めて，その数により関連性が高い Web ページを発見する手法である．

2－3．Web グラフ

Web ページ群を有向グラフとして見る考え方を述べる．図２－２は，Web ページ A から Web ページ B に向けてリンクが張られているという様子を示したものである．矢印がリンクとその方向を示し，Web ページ A から Web ページ B へのリンクのことを Web ページ A の Forwardlink, Web ページ B の Backlink と呼ぶことにする．ある Web ページの Forwardlink を次々と辿って行けば，何らかの関係，関連のある Web ページ群を調べることができる．Web ページをノード(頂点)，Web ページ間のリンクをエッジ(辺)とした有向グラフは，Web グラフと呼ばれている．

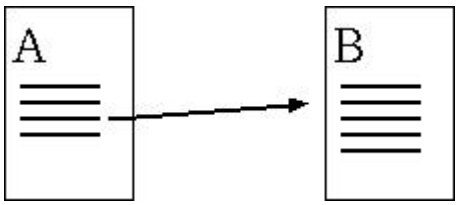

図２－２　Web ページ A の Forwardlink，Web ページ B の Backlink

2－4．Web の蝶ネクタイ構造

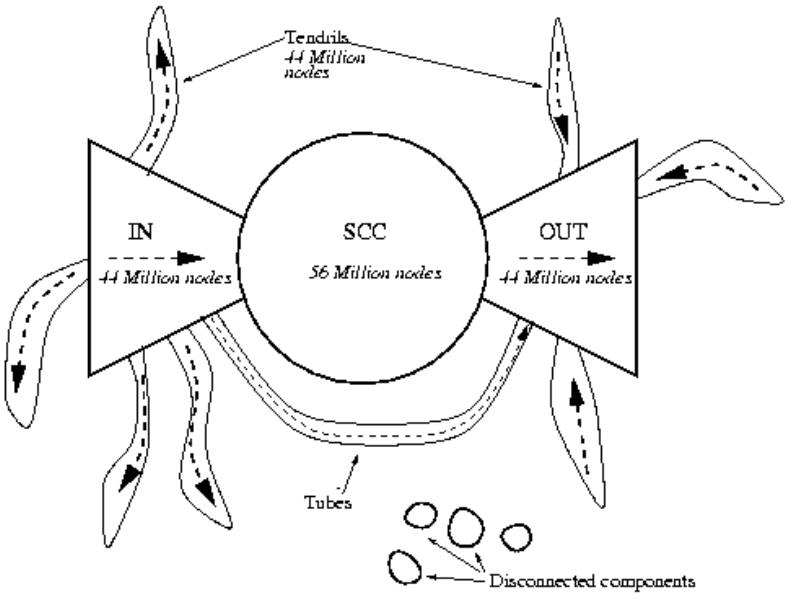

図２－３　Web の蝶ネクタイ構造[9]

文献[9]で，Broder らは AltaVista から提供された約 2 億の Web ページのリンクのグラフ構造を分析している．図２－３に示されているように，Web は巨大な強連結成分 SCC と，リンクを辿ると SCC に到達できるが SCC からは到達できない Web ページ群（IN），逆に SCC からリンクを辿ると到達できるが，SCC へは到達できない Web ページ群（OUT），それ以外の Web ページ群 Tendrils，Disconnected に大きく分類することができ，全体として蝶ネクタイ構造をしていると主張している．

Google が Web ページを収集する際の対象となる Web ページのほとんどが SCC と OUT に属する Web ページであるため，Web 全体から見ると，Web ページの収集率がそれほど高くないと推測される．

3．Web グラフの構造解析

3－1．PageRank

PageRank[4][5]とは，Page と Brin がスタンフォード大学在学中に提案した，Web ページの重要性を測定するための方法である． PageRank の基本的な考え方は，良質な Web ページからリンクされている Web ページは，やはり良質な Web ページである．つまり，ある Web ページ A から Web ページ B へのリンクを Web ページ A から Web ページ B への支持投票とみなし，その票数からその Web ページの重要さを表す指標としてランクが算出される．次に，ランクの計算方法を文献[4]に従って与える．

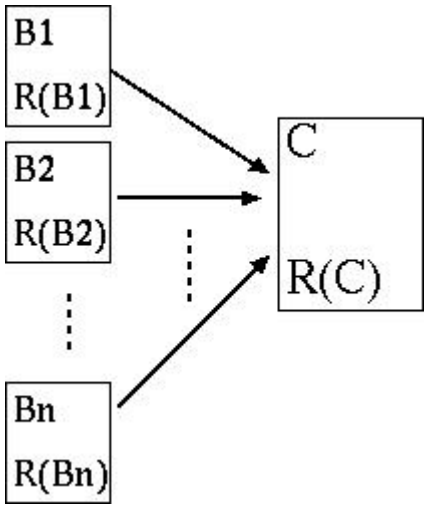

図３－１　PageRank の計算方法

$$R(C) = \sum_{(C,B_i)\in G} \frac{R(B_i)}{Forwardlink(B_i)}$$

式３－１

ランク付けの単純化された定義を式 3－1 に示す．図 3－1 において，Web ページ，B1～Bn，C に対応するランクをそれぞれ， R(B1)～R(Bn)，R(C)とする． (C，Bi)とは，Web ページ C に，Web ページ Bi からの Backlink が存在することを表す．Web ページ C の PageRank は，Web ページ C にリンクを張っている全ての Web ページ Bi のランクを，それぞれの Web ページ Bi の Forwardlink 数で割ったものの総和である．

具体的な例を図 3－2 の場合について説明する．

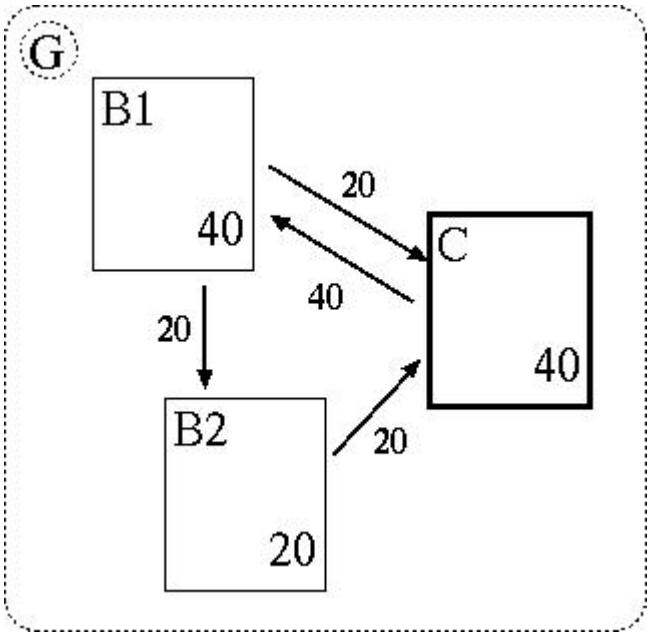

図３－２　ランク計算の例

Web ページ B1，B2，C からなる Web グラフ G における Web ページ C のランクは，Web ページ C の Backlink，つまり，Web ページ B1 と Web ページ B2 のランクと，それらの Forwardlink によって算出される．この場合，Web ページ B1 のランクは 40 であり，その Forwardlink 数は 2 である．また，Web ページ B2 のランクは 20 であり，その Forwardlink 数は 1 である．以上を式３－１に適用すると，

$$
\begin{aligned}
R(C) &= \sum_{(C,B_i)\in G} \frac{R(B_i)}{Forwardlink(B_i)} \\
&= \frac{R(B1)}{Forwardlink(B1)} + \frac{R(B2)}{Forwardlink(B2)} \\
&= \frac{40}{2} + \frac{20}{1} \\
&= 40
\end{aligned}
$$

となり，Web ページ C のランクは 40 となる．

また，Web グラフ G に n 個の Web ページが存在し，その Web ページのランクをつける問題を，(n×n)行列の固有値を求める問題に置き換えて考えることもできる．

ここでは，Web グラフ G 上の，Web ページ C が Web ページ Bi からリンクを張られているとき，つまり Web ページ C が Web ページ Bi からの Backlink を持つ場合，(C, Bi)の成分を 1／Forwardlink(Bi)とし，そうでないときは 0 であるような(n×n)の正方行列とおく．その正方行列の固有値求めることが，Web ページのランクを算出することに相当する．図３－２の場合は，

$$
\begin{pmatrix} (B1,B1) & (B1,B2) & (B1,C) \\ (B2,B1) & (B2,B2) & (B2,C) \\ (C,B1) & (C,B2) & (C,C) \end{pmatrix}
\begin{pmatrix} R(B1) \\ R(B2) \\ R(C) \end{pmatrix}
= \begin{pmatrix} 0 & 0 & 1 \\ 1/2 & 0 & 0 \\ 1/2 & 1 & 0 \end{pmatrix}
\begin{pmatrix} R(B1) \\ R(B2) \\ R(C) \end{pmatrix}
= \begin{pmatrix} R(C) \\ 1/2 \times R(B1) \\ 1/2 \times R(B1) + R(B2) \end{pmatrix}
$$

となり，

$$
\begin{pmatrix} R(C) \\ 1/2 \times R(B1) \\ 1/2 \times R(B1) + R(B2) \end{pmatrix}
= \begin{pmatrix} R(B1) \\ R(B2) \\ R(C) \end{pmatrix}
$$

が得られる．Web グラフ G における全ての Web ページのランクの合計が 1 となるように正規化すると，Web ページ B1 のランクは 0.4，Web ページ B2 のランクは 0.2，Web ページ C のランクは 0.4 となる．

現在，実装されている Google のシステムでは，上記の方法が改良して使用されている．改良されている点は以下の通りである．

Web グラフ G 上のある Web ページ C について，

1．ある Web ページのリンクを辿ることなく，直接 URL が入力されて Web ページ C が訪問される

2．様々なリンクを辿って，Web ページ C が訪問される

という，1，2 の場合の確率を，ランク計算の一部に取り入れている点である．これらを考慮したランク付けの定義が，式３－２である．

$$R(C)=\frac{\varepsilon}{n}+(1-\varepsilon)\sum_{(C,B_i)\in G}\frac{R(B_i)}{Forwardlink(B_i)} \qquad \text{式３－２}$$

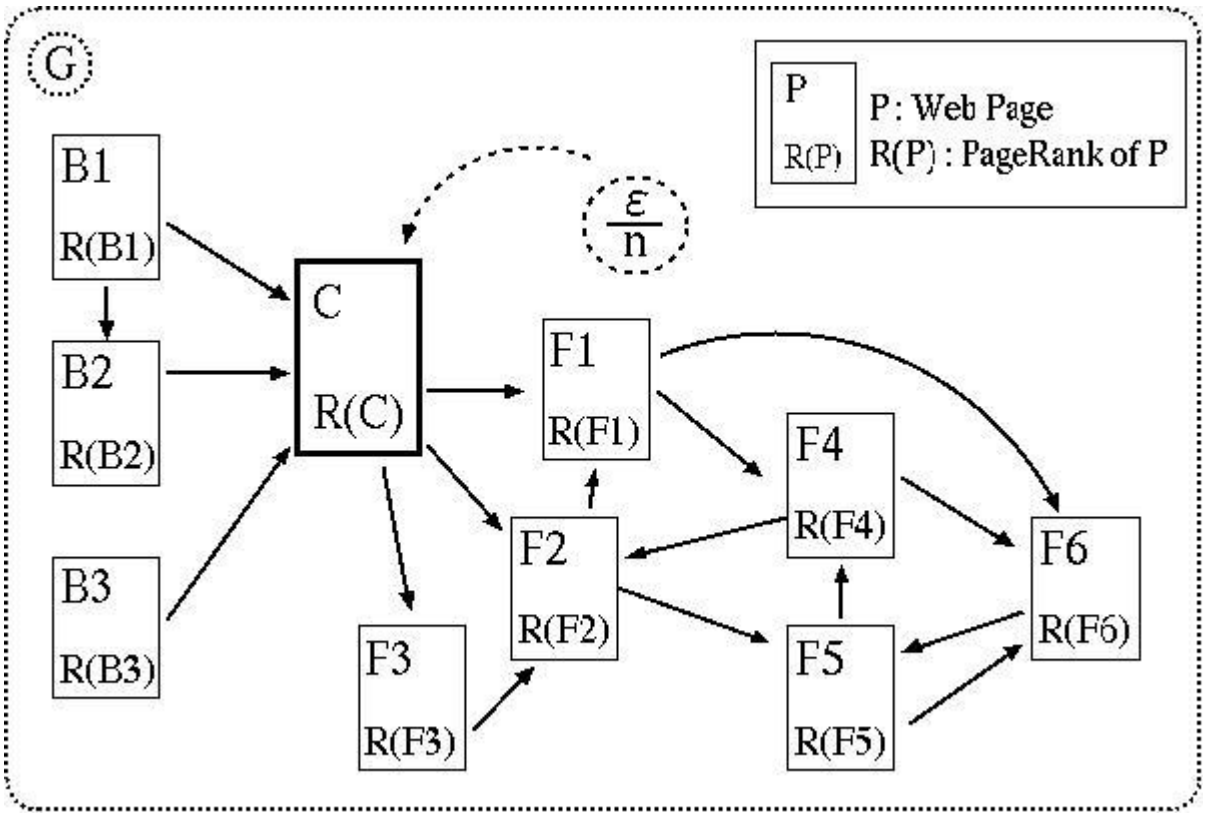

図 3－3　実際のランク計算法

(1)　Web ページ C の URL が直接指定されて，Web ページ C が訪問される場合

Web グラフ G 上の各 Web ページが，直接 URL を指定して訪問される確率を ε とし，Web グラフ G 上の Web ページ数を，n 個とすると，Web グラフ G 上の，ある Web ページ C が URL を指定され訪問される確率は ε ／ n となる．従って，Backlink を一つも持たない Web ページのランクは ε ／ n となる．これで，Backlink を持たない Web ページにもランクを付けることができる．

(2)　様々なリンクを辿って，Web ページ A が訪問される場合

この場合の確率は，(1)のときの確率を ε としたので，(1－ ε)となる．この確率を式 3-1 のランク計算方法で算出されたランクに掛けて，リンクを辿って Web ページが訪問された場合の Web ページのランクとする．

図３－３において，(1)の場合は点線，(2)の場合を実線で表した．実際の Web ページのランクは，(1)と(2)の場合で算出された数値を足したものになる．

PageRank を計算する際に，Forwardlink を持たない Web ページ(Dangling Links)は，直接いかなる Web ページのランキングにも，響しないため取り除いて計算を行い，グラフを構築するときに再び戻す．

特殊な場合として，図３－４に示すように Web ページ間のリンク構造が小さなループになっていることが考えられる．この場合，Random Surfer はループから抜け出すことができず，永遠にそのループを回り続けてしまう．この問題を解決するために Rank Sink という機能が備えられている．Rank Sink は，Random Surfer が，Web 空間を巡っていて，リンクのループに入ってしまった場合，上の(1)で述べたように ε の確率で，Web 空間上のある Web ページにジャンプさせるということでループから抜け出せるようにしている．

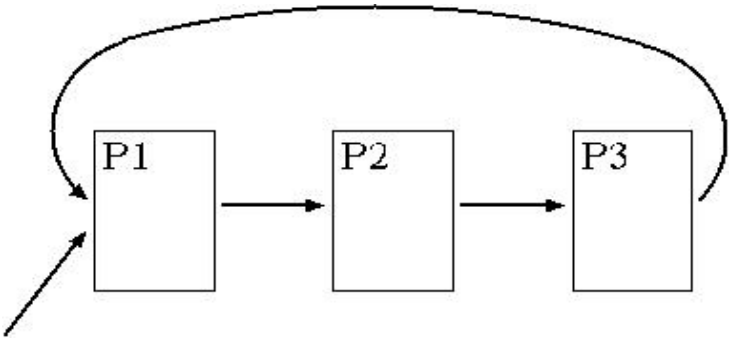

図３－４　Web ページのループ

PageRank によって、全ての Web ページをランク付けすることができる．Google では，Web 情報の検索結果を，このランクに基づく順序通りに表示することにより，ユーザにより質の良い検索結果を提供している。Web グラフを可視化する際，この PageRank によって測定される Web ページの重要度をもとに，Web のグラフの頂点(ノード)に重みをつけることで、重要なページを容易に見つけることができる．

3－2．Web ページとリンクの関係

文献[8]で，Albert らは，NotreDame 大学の全ての Web ページ（約 33 万ページ，146 万リンク）からなる Web 空間を分析し，Web ページとリンクの関係がベキ分布に従うことを示した．Forwardlink 数が x である Web ページの個数 y は，ある定数 a，b について log(y) = a – b * log(x)となる．(log(x)，log(y))をプロットすると， x 切片が a，傾きが－b の直線となる．この係数 b の値は，ある Web ページから k 個のリンクが出ている確率が k の－ b 乗であることを意味する．Backlink についても Forwardlink の場合と同様である．これにより，文献[8]の場合，たかだか 0.35+2.06*log(N)回リンクを辿ることで，どの Web ページにも辿り着くことができる．ただし N は総 Web ページ数である．

3－3．Jaccard 関数

ある二つの Web ページの関連性を調べる方法である．サーチエンジンを利用し，二つの Web ページを個々に検索した結果を足したものを分母に，同時に検索した結果を分子にして，その値の大きさにより二つの Web ページの関連性を調べる．Jaccard 関数により求められる二つの Web ページ間の関連性をもとにして，より関連性の高い Web ページ間のリンクを表す Web グラフのエッジを短くすることで，Web ページ間の関連性がわかりやすい Web グラフを作成することができると考えられる．

3－4．コミュニティの発見

コミュニティ（Web コミュニティ）という単語は，様々な文脈において用いられているが，ここでは Web ページのリンクによって蜜に結合した関連の強い Web ページの集合という意味で用いることにする．コミュニティを発見するためには，Web グラフを探索しリンクによって蜜に結合している部分的な Web グラフをコミュニティとする方法が考えられる．コミュニティを発見し，各コミュニティ毎に色分け等をすることにより，Web グラフにおける Web ページの関係がわかりやすくなると考えられる．

4．Web グラフの可視化システム

4－1．Web グラフの可視化

Web 情報検索の補助として，Web グラフを可視化する．その利点を，次に挙げる．

(１) Web の検索目的のページを起点として個々のページの局所的情報だけでなく，周囲を見渡すことのできる広域的な探索の支援となる．

(２) Web グラフを可視化し，Web ブラウザの機能を拡張することにより，より良い支援が期待できる．

(３) Web ページの内容とページ間の関係を関連付けさせることが容易になる．

4－2．システムの説明

本システムは，入力された URL の Web ページに含まれるリンク(URL)を再帰的に抽出し，その結果に基づき Web ページをノード, ページ間のリンクをエッジとする Web グラフを表示するためのシステムである. Web-map の作成において，リンク構造の解析には JAVA 言語，Web グラフの作成には dot を使用した.

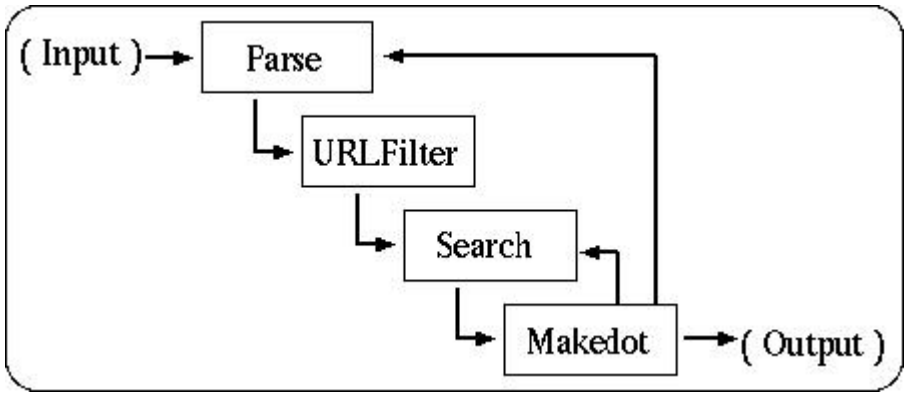

図４－１　システムの流れ

図 4－1 は，本システムの流れ図である．最初に，ユーザが情報を抽出したい URL と調べたい深さを入力する．次に，Parse クラスの Parse メソッドで，入力された URL が示す Web ページのソースからリンク情報を抽出する．抽出されたリンク情報は絶対パス，相対パス，アンカーといった種類のパスで表現されているが，そのリンク情報を URLFilter クラスの Filter メソッドで，すべて絶対パスに変換する．その結果をもとに，Search クラスの Search メソッドで URL の重複をチェックする. 重複の無い URL のデータをファイルに保存し, dot ファイルを作成するために Node 番号を返す. 最後に，Makedot メソッドが Node 番号を受け取り，dot ファイルを作成する．次に各メソッドの詳細について述べる.

・Parse メソッド
入力された URL に接続し，そのソースからリンク情報を抜き出し，リンク情報の配列を作成する.

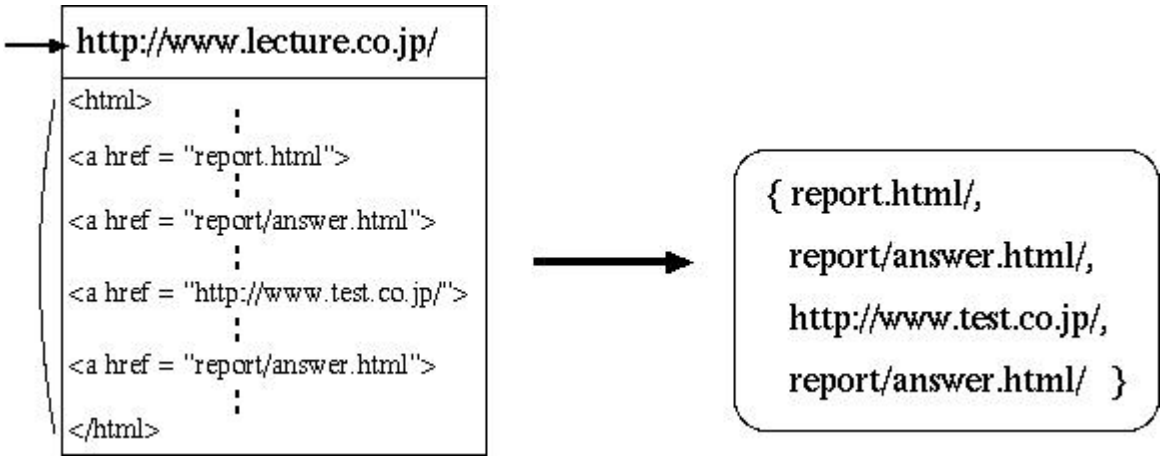

図４－２　Parse の処理

・URLFilter メソッド
Parse で作成したリンク情報の配列を，すべて絶対パスにする.

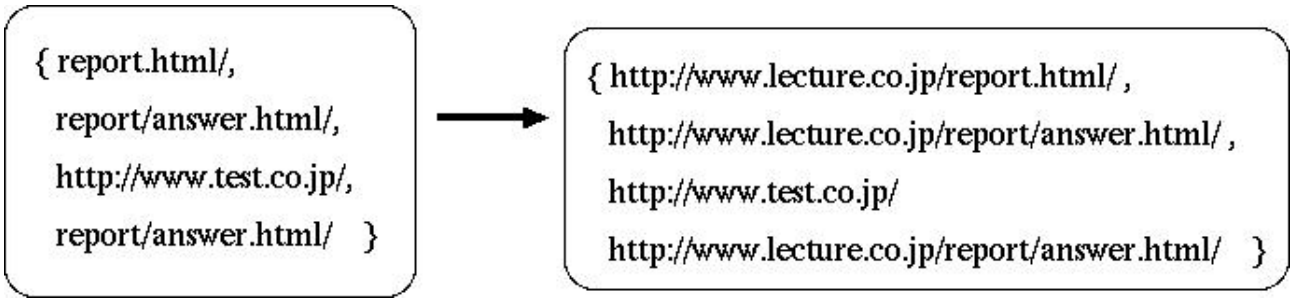

図４－３　URLFilter の処理

・Search メソッド

URLfilter で作成したリンク情報の配列を，一つずつ受け取り重複の無いリンク情報のデータベースを作成する．また，dot ファイル作成のための Node 番号を返す．最初に入力された URL の Node 番号を 0 とする．

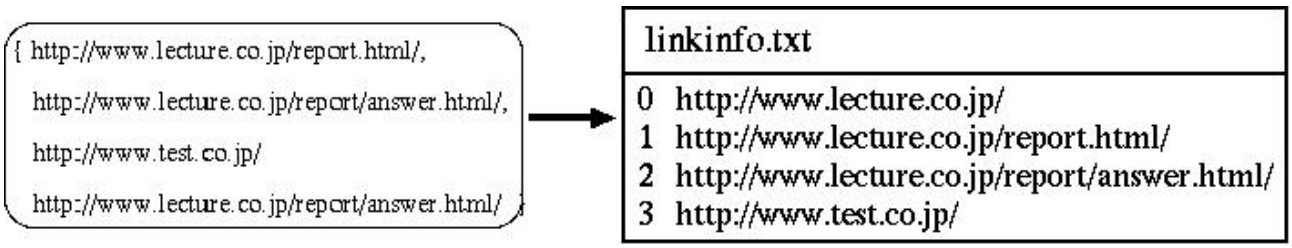

図４－４　Search の処理

・Makedot メソッド

Search メソッドから Node 番号を受け取り，Web グラフを可視化するための dot ファイルを作成する．

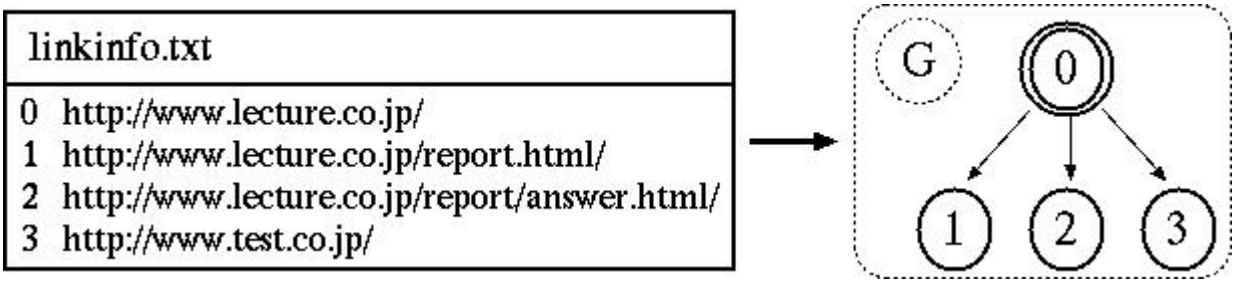

図４－５　Makedot の処理

4－3．実行例

島根大学総合理工学部数理・情報システム学科情報分野の URL を起点とし，Web-map を用いて作成した結果の Web グラフを次ページに示す．

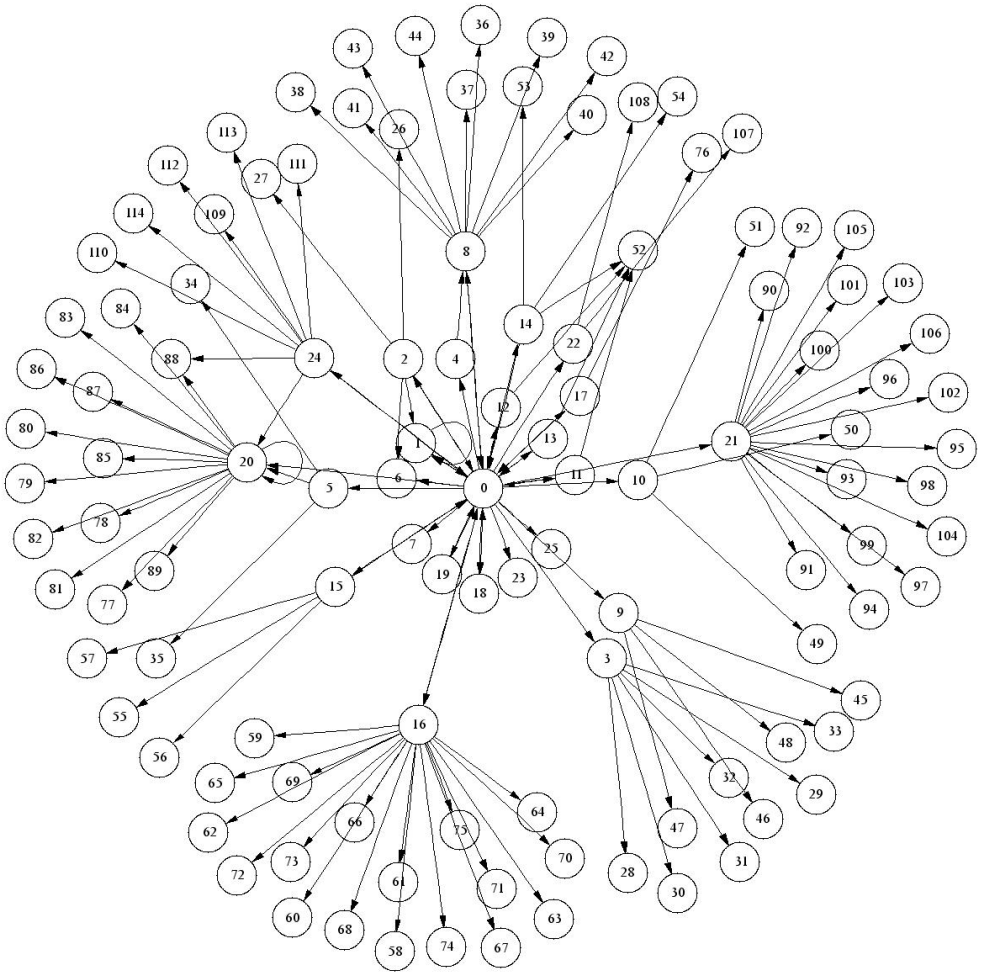

図４－６　可視化された Web グラフ

島根大学の総合理工学部数理･情報システム学科情報分野の URL を Web-map に入力し，Web グラフを深さ２まで表示したものが図４－６である．この Web グラフにおいて，ノード 0 が情報分野のページ（http://www.cis.shimane-u.ac.jp/）を指している．また，ノード 16（http://www.cis.shimane-u.ac.jp/jabee/ ），20（http://www.ipc.shimane-u.ac.jp/），および 21（http://www.cis.shimane-u.ac.jp/systeminfo/）が多くの Forwardlik を持つ．これらは，それぞれ情報分野の Jabee のページ，島根大学総合情報処理センター，および情報分野のシステムインフォメーションのページとなっている．

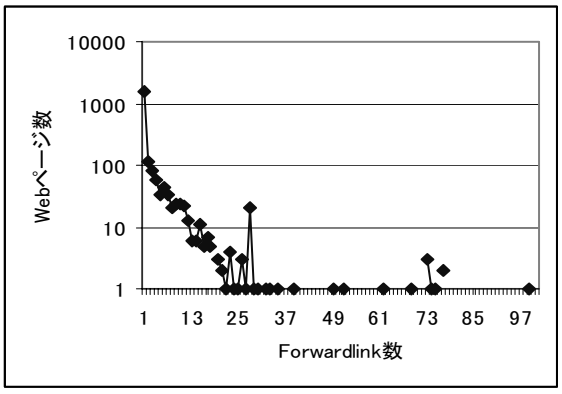

図４－７　Web ページの Forwardlink 数

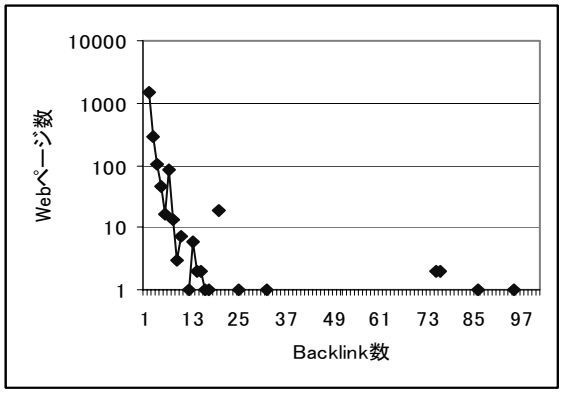

図４－８　Web ページの Backlink 数

図４－７，図４－８は，Web-map に島根大学総合理工学部数理・情報システム学科の Web ページを入力し，深さを５としたときに得られる Web グラフの Forward リンク数，Backlink 数と Web ページ数の関係を解析した結果である．縦軸が Web ページ数を，横軸が Forwardlink 数（図４－７），Backlink 数（図４－８）を示す．また，入力した Web ページから深さ５で繋がっている Web ページの総数は，2115 ページである．図４－７において，Forwardlink 数が多くなるにしたがって Web ページ数が，ほぼ指数関数的に減少しており，文献[8]で述べられている結果と一致している．図４－８は，図４－７ほど明らかではないが，やはり Backlink 数の多い Web ページ数がほぼ指数関数的に少なくなっている．図４－８の結果が図４－７と比べて少し不明快な理由は，収集しているデータが少ないためであると考えられる．ここで，文献[8]によると，Web ページと Forwardlink，Backlink の関係はベキ分布に従うことが知られている．得

られたデータの詳細を統計的に解析することにより，Web グラフの直径を求めることができると考えられる．

5. まとめと今後の計画

本研究で作成した Web-map を用いて Web グラフを可視化すると，入力した URL の Web ページとその周辺の Web ページ群の関係を，一目で把握できる．また，Web グラフを可視化する際に得られるリンク情報を解析することによって，入力した URL の Web ページと，その周辺の Web ページ群の関係，関連性を知ることができるため，Web 情報検索の支援となる．

関連のある Web ページを 3 次元的に表示するシステムとして，NTT 研究所の InfoLead[11]が知られている．Web グラフを見ただけでは，ノードの示す Web ページがどのようなページか具体的にはわからない．そこで，InfoLead のように Web ページを直接見ることができるように，Web グラフのノードをクリックすることにより，そのノードに対応する Web ページを表示する機能を追加する．また，統計的解析の結果も Web グラフに反映させる予定である．

更に，今後の計画，課題としては，次の 5 点が挙げられる．

（1）Web グラフの可視化について，Web グラフを解析して Web ページの重要度を測定し，Web グラフの重要な Web ページだけを表示させる．

（2）（1）で得られる重要な Web ページの番号を再び入力し，入力された Web ページの周辺の Web グラフの詳細を表示させる．

（3）Web グラフの解析結果をもとに，よりわかりやすく Web グラフを可視化させる．

（4）Web グラフに Backlink も追加する．そのため，Backlink を見つけるための効率のよい方法を考える．

（5）対象を島根大学全体にして，３章 2 節で行われている，Web グラフの直径を求める．

謝 辞

本研究の一部は柏森情報科学振興財団の援助を受けて遂行された．

参考文献

[1] Netcraft (http://www.netcraft.co.uk/)

[2] 山名早人: 検索エンジンと高速ページ収集技術 －分散型 WWW ロボット実験－, bit vol.32, No.12, 2000, pp.72-99.

[3] 風間一洋，原田昌紀: Web サーチエンジン技術の高度化，人工知能学会誌, 16 巻, 4 号, 2001, pp.503-508.

[4] Page,L. , Brin,S. , Motowani, R. and Winograd,T: The PageRank Citation Ranking: Bringing Order to the Web, Stanford Digital Library Technologies, Working Paper SIDL-WP-1999-0120 , 1998 .

[5] 山名早人，近藤秀和: サーチエンジン Google，情報処理 , Vol.42, No.8, 2001 , pp.775-780.

[6] 廣川佐千男，池田大輔: Web グラフの構造解析，人工知能学会誌，16 巻, 4 号, 2001, pp.525-529.

[7] 村田剛志: Web コミュニティ，情報処理，Vol44，No.7，2003，pp702-706.

[8] Steve Lawrence, C. Lee Giles: Accessibility of information on the web, Nature, Vol.400, 1999, pp107-109.

[9] Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Ranjagopalan, S., Stata, R., Tomkins, A. and Wiener : Graph Structure in the Web: Experiments and Models, Proc. of the 9th WWW Conference, 2000, pp309-320.

[10] Albert,R., Jeong, H., Wtaki, A., Fujino, R., and Barabasi, A.: Diameter of the World Wide Web, Nature, Vol.401, 1999, pp.130-131.

[11] http://www.ntt-infolead.net/