

ネットワークの自律的構築のための近傍仲介法の提案とシミュレーションによる検証

濱崎 雅弘[†] 武田 英明^{††, †}

A proposal and examination of neighborhood matchmaker method for autonomous network construction

Masahiro HAMASAKI[†] and Hideaki TAKEDA^{††, †}

あらまし 本論文で、我々はパーソナルネットワークの最適化手法として近傍仲介法 (Neighborhood matchmaker method, 以下、NMM) を提案する。パーソナルネットワークは情報収集にとって非常に有用であるが、情報収集における多様な要求に応えるためには、そのネットワークを拡張・最適化する必要がある。提案する NMM は隣接するノードのマッチングを互いに繰り返すことでネットワークを逐次的に拡張し、最適化する手法である。人のパーソナルネットワークに適用した場合は、既にある人の関係を利用した拡張・最適化が可能になる。我々はシミュレーション実験を通して提案手法の有効性を検証した。実験には仮想的に作ったデータと実際に被験者が作ったデータの両方を用いた。結果、どちらのデータにおいても十分なネットワークの最適化が行われていることが確認できた。

キーワード ネットワーク最適化, 自律分散システム, 情報共有, パーソナルネットワーク

1. はじめに

人で行なわれる情報交換は、情報過多への効果的かつ現実的な解決策の一つである。人は、情報の収集やフィルタリング、組織化などを行ってくれる知的エージェントであるといえる。我々は自分のパーソナルネットワークで繋がった人たちと情報交換することで、価値ある情報を得ている。より良い情報交換の環境を得るためには、より良いパーソナルネットワークを作る必要がある。しかし個人個人が持つパーソナルネットワークは、個人の知りうる範囲で適切であるかもしれないが、全体から見ればより適切なネットワークが存在しうる。このために本論文では、ネットワークを自律分散的に最適化する手法 (近傍仲介法) を提案し、実験により評価する。これはすでに築かれているネットワークを元に逐次的に拡張していくこと

で、最適化する手法である。

本論文の構成を以下に述べる。2 節において、情報交換のための人と人との関係をどのように作り上げるべきかという点について論じる。3 節において、我々が提案する手法を用いてネットワークがどのように最適化されるのかを説明する。4 節にて実験の概要を説明し、5 節に実験結果を示す。6 節でその考察について述べ、最後に 7 節にて本論文をまとめる。

2. 情報交換相手との関係発見

まず情報交換は、その形態の違いにより 2 種類に分けられる。一つはメールのような一対一型、もう一つはメーリングリストや掲示板のような一対多型である。我々がメールを使うのは大抵の場合、すでに友達であるかもしくはホームページが何かで見知った相手と情報交換をするときである。メールを用いた情報交換では、相手の選択や理解は比較的容易である。対してメーリングリストや掲示板では、情報共有の場を介して多くの人同士で情報交換が行なわれる。この場合、我々は情報交換の新しい相手を比較的容易に獲得することができる。

[†] 総合研究大学院大学, 東京都千代田区一ツ橋 2-1-2
Department of Informatics, The Graduate University for Advanced Studies, 2-1-2 Hitotsubashi Chiyoda-ku Tokyo

^{††} 国立情報学研究所, 東京都千代田区一ツ橋 2-1-2
National Institute of Informatics, 2-1-2 Hitotsubashi Chiyoda-ku Tokyo

前者の問題点は、新しい情報交換相手の獲得が困難であることである。このタイプでは、既に知っている相手としか情報交換が行えない。一般に、人が普通に知りうる人の数には限界があるため、情報交換にとって良い環境を築くのは難しい。対して後者は、新しい情報交換相手の選択が困難であるという問題点がある。多くの参加者の中から自分にとって良い情報交換のパートナーを見つけ出すのは、とても知的かつ時間のかかるタスクであり、人にとって大きな負担となる。

コミュニティ形成や情報交換を支援するシステムは幾つか提案されている。Kautz らの REFERRAL WEB [1] は WWW における人の関係の重要性に注目したシステムである。REFERRAL WEB は文献の引用関係などを利用して人の関係を見つけ出す。角らは美術館や研究会などで、同じ関心を持った人同士の出会いを人の関係を可視化することで支援するシステムを提案している [2] [3]。また、亀井らは参加者の関係を可視化することでコミュニティの形成を支援するシステムを提案している [4]。

これらのシステムは、支援対象となるグループの存在が明示的もしくは暗示的に仮定されている。グループのサイズが大きければ良い相手と情報交換ができる可能性が高くなるが、大きなグループであるほどより良い相手を見つけ出すことは難しくなる。また、そのような大きなグループを作り、運営することも困難である。特殊なグループを対象とすることでそのような問題を回避している例がある。REFERRAL WEB は研究者のグループが持つ文献データベースという質の高い情報を利用することで、効果的な支援を実現している。角らは研究会の参加者や美術館の訪問者にシステムを適用することで良い結果を得ている。しかしこれらのシステムは誰もが使えるというわけではない。

グループの形を少し異なる方法で作りに出しているものもある。人の情報共有のネットワークを介した情報伝播を利用した情報共有システムである。これは、直接的に情報を共有・交換するのは自分が選択した相手とだけだが、共有・交換対象となった情報が共有したユーザらを介してさらに他の相手に伝わるようになっている。竹内らの情報伝播システム [6] では他人から受け取ったメッセージに興味深いものがあつた場合、システムに対して関心があることを伝えるだけで、自分と共有関係にある全メンバーに対して自動的にメッセージが転送される。吉田らの WAVE [5] では自分の

情報を選択した相手にだけ公開し、その情報に興味を持ったユーザはそれを取り込み、自分が持つ情報として公開する。この行為の繰り返しによって情報が伝播していく。これらのシステムはいわば口コミのネットワークを用いて情報共有をしようというもので、自分を中心とした、境界の緩やかなグループを各ユーザがそれぞれ持っているようなイメージとなる。このモデルでは明示的にグループを作る必要は無い。しかし竹内の報告 [6] [7] では、この様な口コミによって行なわれる情報共有では、直接情報交換を行なう相手の選択が重要であると指摘されている。つまり、このような緩やかなグループ形態をとった場合でも、相手の選択は重要な課題である。

より良い相手と情報交換をしたいならば、我々は多くの相手と出会い、そして選択しなければならぬ。ゆえに情報交換を支援するシステムには、新しい相手と出会い選び出さなければならないという潜在的な問題を解決する方法を持つ必要がある。

我々は実生活において、しばしば友達の紹介を介して新しい友達を得る。紹介された友達は、出会う前には当然知らないため相手を選択することはできない。しかし、両者を知っている共通の友達がすでに選択を行った上で紹介を行ってくれているため、ある程度の選択を経た相手と出会うことができる。我々は、このような友達が新しい相手とのマッチメイカーになるというアイデアを用いた、新しいネットワーク最適化手法を提案する。この手法の説明を次節にて行う。

3. 近傍仲介法

本節では、我々が提案する自律的構築が可能なネットワーク最適化手法の紹介を行なう。この手法は実世界における手法を模倣したものである。我々はこの手法を近傍仲介法 (Neighborhood Matchmaker Method, 以下, NMM) と呼んでいる。

NMM を説明する前に、NMM の適用対象となるネットワークのモデルを定義しておく。まず人をノードとし、人と人との情報交換がなされている関係をパスとする。次に、全てのノードは、ノードとノードの関係の計算が可能であると仮定する。計算された関係の強弱を示す値を関係値と呼ぶ。

情報交換にとってより良い環境を作るには、パーソナルネットワークの最適化が必要である。このモデルにおける最適なネットワークとは各ノードにおいて関係値が高いノードと繋がっているネットワークである。

全てのノードの情報を知っている場合、ノード間の関係を総当りで計算することで最適なネットワークの発見ができる。以下ではこのような一括計算による最適化手法をサーバ型計算法と呼び、それによって生成された最適なネットワークを最適解と呼ぶ。ここではこのような全ての情報を集約するサーバがなく、個々のノードにおける計算のみで解を求める場合を考える。個々のノードは自分が繋がっているノードしか知り合えないので、最適なネットワークを形成するにはそれ以外の新しいノードを知る方法が必要である。NMMでは、隣接ノードが互いに推薦しあうことで、自律分散的にネットワークの最適化を行なう。

NMMにおける、各ノードの振る舞いの例を図1に示す。まず最初に各ノードが自身と隣接しているノード間の関係を計算する(図1-1)。良い関係にあるノードのペアを発見した場合、それぞれのノードにお互いを推薦する(図1-2)。推薦を受け取ったノードは新しくパスをはるかどうか決める。推薦を受け入れると推薦されたノードが隣接ノードとして追加される(図1-3)。これらの動作の繰り返しにより、ネットワークは改善されていく。この手法は、前節で述べた友人関係の推薦を模倣したものである。この特徴は自分の友達が選択したものが推薦される点である。このため、推薦されたノードは比較的適切なものであることが期待される。

次節以降において、この手法のシミュレーション実験の解説を行なう。

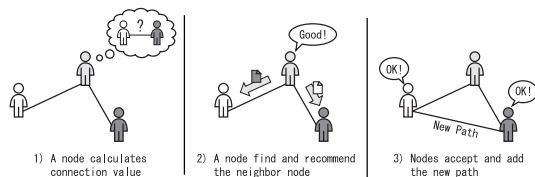


図1 Behavior of nodes

4. 実験概要

4.1 シミュレーションの概要

本実験の目的は、提案する手法がどの程度パーソナルネットワークの最適化に有効であるかを調べることにある。そこで実データと仮想データを用いて、NMMの最適化をシミュレーションにより評価する。

本シミュレーションでは、各ノードは極めて単純な戦略をとる。各ノードは、推薦されたノードとの関係

値が、今すでにパスを張っている相手との関係値よりも高い場合はパスの張替えを行なう。つまり、推薦されたノードとの関係値が、隣接ノードとの関係値の中で最低のものよりも高い場合、最低の関係値を持つ隣接ノードとのパスを切り、新たに推薦されたノードとのパスを張る。

図2はこのシミュレーションのフローチャートである。まず最初に各ノードにデータを与える。今回は、10次元の特徴ベクトルとWWWブックマークの2種類のデータを用いた。次に、ノード間にランダムにパスを張る。パス数はパラメータとして設定する(1)。

ネットワークの再構築は以下の手順で行なう。まず、全てのノードは自分と隣接する(パスが張られている)ノード間の関係値を計算する。計算を行なったノードは、各隣接ノードに対して最も高い関係値のパスを張ることができるノードをそれぞれ一つ選び、そのノードに推薦する(2)。次にパスの張替えを行なうノードをランダムに選択する(3)。選択されるノードは1ターンに1つだけで、そのノードはかならず新しいパスを生成可能でなくてはならない。選ばれたノードは隣接ノードから推薦されたノードの中から最も関係値の高いものを選び、新しくパスを張る。そして既に張っているパスの中から最も関係値の低いものを選び、そのパスを切る(4)(5)。1ターンに1つのパスが新しく生成され、1つのパスが削除されるので、全体としてはパス数は常に一定に保たれる。

各ターンに選択されるノードは新しいパスを生成可能でなくてはならない。そして、新しいパスの関係値は、既に自分が持っている削除候補となるパスよりも高くなくてはならない。ネットワークを構成する全てのノードが新しいパスを生成できなくなれば、そのネットワークは収束したとみなし(6)、シミュレーションは終了する(7)。

図3はパスの張替えの例を示している。図中でノードAはすでにノードB、C、D、Fとのパスを持っている。ノードAが新たに生成可能なパスは、ノードBまたはFがマッチメイキングしてくれたノードEまたはGとのパスである。ノードAにとってノードEおよびGは未知のノードであり、それらと自身とのパスの関係値は計算不可能である。だが両ノードを知るマッチメイカーであるノードBまたはFがそれぞれ関係値を計算しており、ノードAには伝えられている。この値を元にノードAは新しいパスの選択を行なう。

まず、ノードAはすでにあるパスのなかで最も関係

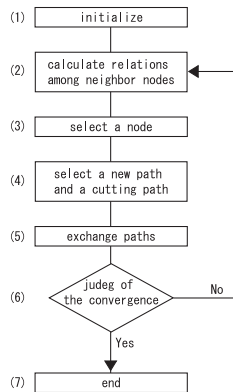


図 2 Flow chart of simulation

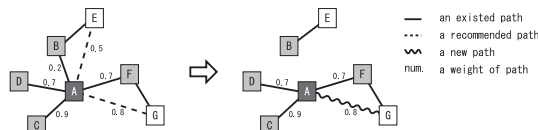


図 3 How to make a new path

値の低いパスを選ぶ。この例では、ノード B とのパスは 0.2 で最も低い。次にマッチメイカーが推薦するノードの中から最も関係値の高いパスを持つノードを選ぶ。この例では、ノード G とのパスとなる。ノード B との関係値よりもノード G との関係値の方が高いので、このパスの張り替えは成立する。具体的にはノード B とのパスは削除され、新たにノード G とのパスが生成される(図 3-右)。なお、パスの追加および削除はパスの一端のノードの判断によって行なわれる。そのため、場合によってはどこからもパスの無い孤立ノードが生まれる可能性もある。本シミュレーションでは、いったん孤立したノードが新たにパスを張る方法はない。

我々はシミュレーション結果を可視化するために、Java アプレットを実装した。図 4 がシミュレーションの結果を Java アプレットによって可視化したものである。それぞれ、初期状態 (a)、収束状態 (b)、最適解 (c) のネットワークを示している。図 4-b(converged) のネットワークが図 4-a(initial) よりもずいぶん図 4-c(ideal) に似ていることがわかる。

4.2 評価値

全てのノード間の関係値が一箇所で計算可能であるならば、関係値において最適なネットワークというものを作ることができる。そこで、サーバ型計算法で計算した最適なネットワークと提案手法で最適化した

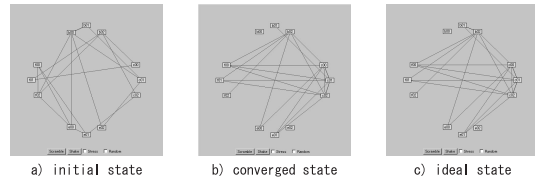


図 4 the networks at the some states

ネットワークと比較することで、我々の提案する手法の有効性を評価する。評価は二つの評価値を用いて行う。一つは被覆率という指標で測るもので、これは最適解に対してどの程度似たようなネットワークが提案手法によって作られているかを示す割合である。もう一つは到達率という指標で、これはノード間の関係値の総和における割合である。ネットワークがどれだけ関係値の高いパスから構成されているかという指標となる。被覆率は形式的に二つのネットワークの類似を示すのに対し、到達率は実質的に類似していることを示している。被覆率と到達率は直接関係していないので、被覆率が悪くても到達率が良いことや、その逆もありうる。被覆率、到達率はそれぞれ以下の数式により求められる。なお、最適解の被覆率、到達率はともに 1.0 である。

$$\text{被覆率} = \frac{|\{P_{current} \cap P_{best}\}|}{N}$$

$$\text{到達率} = \frac{\sum_{i=1}^N f(p_i | p_i \in \{P_{current}\})}{\sum_{m=1}^N f(p_m | p_m \in \{P_{best}\})}$$

- p : パス
- N : パス数
- $\{P\}$: パスの集合
- $\{P_{best}\}$: 最適解のパスの集合
- $\{P_{current}\}$: その時点でのパスの集合
- $f(p)$: パスの関係値

5. 実験

5.1 仮想データを用いた場合

本シミュレーションでは、ノード間の関係値を計算するためにデータが必要である。仮想データの場合、各ノードは乱数を用いて生成された 10 次元の特徴ベクトルを持つ。ノード間の関係値はノードがそれぞれ持つ特徴ベクトルの内積によって求められる。特徴ベクトルの要素は 0.0~1.0 の乱数である。ただし各要素は 0.5 の確率で 0.0 になるようにした。これはノード

間の関係にばらつきを作るためである。

実験のために二つのパラメータを設定する必要がある。一つはノード数であり、もう一つはパス数である。本実験では、ノード数を 10~100 とし、パス数をノード数の 1~5 倍とした。実験の諸設定を表 1 に示す。シミュレーションは各設定ごとに 10 試行し、その平均値を結果とした。

表 1 The set of parameters

ノード	10 次元特徴ベクトル
ノード間の関係値	特徴ベクトルの内積
ノード数	10, 20, 30, 40, 50, 60, 70, 80, 90, 100
パス数	$\times 1, \times 2, \times 3, \times 4, \times 5$

図 5 の二つのグラフはともに横軸がターン数で縦軸が被覆率のグラフである。図 5-上はパス数をノード数の 3 倍に固定したときの結果であり、図 5-下はノード数を 60 に固定したときの結果である。

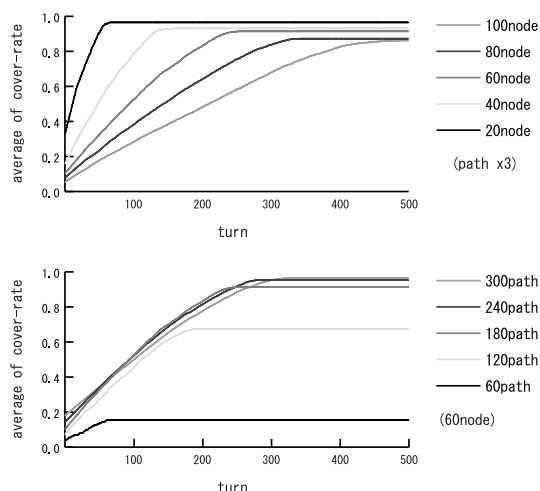


図 5 Cover-Rate in the virtual data

図 6 は横軸にターン数、縦軸に到達率をとったグラフである。先ほどの被覆率に関するグラフと同様、それぞれパス数とノード数を固定した場合の結果を示している。グラフを見ると、パス数やノード数が到達率に与える影響は、被覆率の場合よりも比較的小さいことがわかる。

図 7 および図 8 のグラフは評価値とノード数およびパス数の関係を示している。図 7-a のグラフは初期状態、図 7-b のグラフは収束状態におけるノード数と被覆率の関係をそれぞれ示している。可能なパス数はノード数の 2 乗に比例するので、確率的に生成される

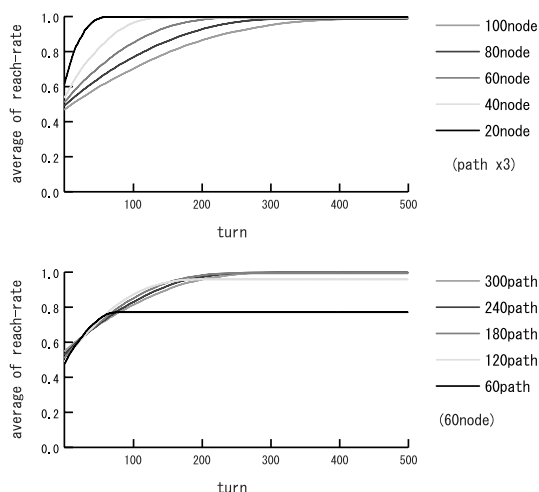


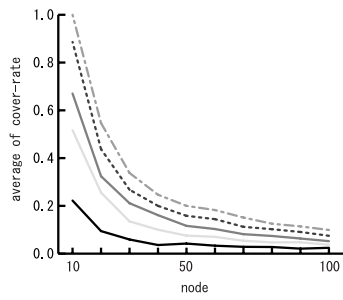
図 6 Reach-Rate in the virtual data

初期状態における被覆率はノード数に反比例する(図 7-a)。図 7-b を見ると、この初期状態における差はパス数が 2 倍程度までは影響を受けるが、それ以上では問題がないことがわかる。同様に図 8 の二つのグラフはそれぞれの状態のノード数と到達率の関係を示している。到達率も初期状態においてノード数と反比例の関係にあるが(図 8-a)、その影響が現れるのはパス数が 1 倍程度のときだけであることが図 8-b からわかる。以上のようにパス数がノード数の 2~3 倍以上ある場合、ほぼ最適解に近い解に至ることがわかった。

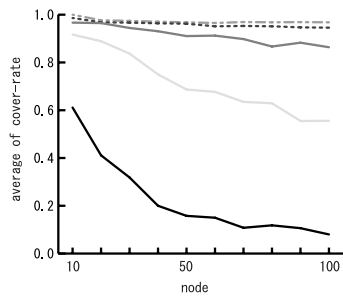
図 9 のグラフは、パスおよびノード数と収束ターン数の関係を示している。グラフの横軸はノード数、縦軸は収束ターン数である。なお、本実験では全てのケースにおいて収束が確認された。グラフから、ノード数の増加にしたがって収束ターン数は線形に増加していることがわかる。

以上が仮想データにおける NMM の最適化の性能についての実験結果である。被覆率および到達率という二つの指標で NMM の最適化過程を評価したが、両指標において十分な結果が得られている。

ここで興味深いのは被覆率と到達率に差があるケースである。これは今回の実験においてパス数が 2 倍の時に顕著であった。例えばノード数が 100、パス数が 200 の時に、被覆率は低い(0.56)にもかかわらず到達率は比較的高い(0.95)ネットワークが作られている(図 7-b および図 8-b)。これは形式的には異なるが、実質的には類似したネットワークであることを示している。このネットワークは NMM 特有の興味深い最適

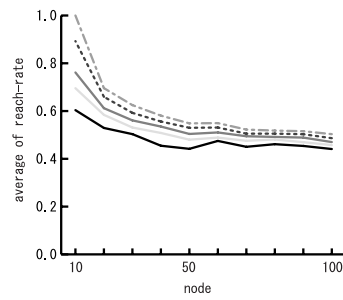


a) start

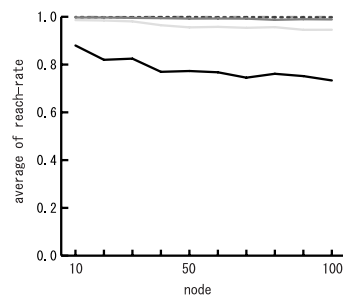


b) end

図 7 The average of cover-rate



a) start



b) end

図 8 The average of reach-rate

化の結果といえる。

このネットワークの詳細を調べると次のようであった。図 10 にノード数が 100, パス数が 200 の場合の, 各ノードが持つパス数のヒストグラムを示す。図 10-a が最適解, 図 10-b が NMM による収束解である。100 ノードに対して 200 のパスなので, 平均して 1 ノードあたり 4 つのパスを持つ。最適解では最頻値が 4 となっているが, 収束解では最頻値は 4 ではなく, ヒストグラムもより平坦である。必要以上に多くのパスを持っているノードが多数存在し, それによって必要なパス数を持ってないノードができていく。これは, 局所的に優れたノードにパスが集まり, ネットワークが収束した(局所解に陥った)結果といえる。

5.2 実データを用いた場合

前節では乱数で作った仮想データを用いてシミュレーション実験を行なった。次の実験では, 実際に人が用いているデータを使って実験を行なう。実験データとしてここでは WWW ブックマークを用いる。WWW ブックマークはユーザが興味を持った Web ページを集めたデータであり, 利用者の関心が表れているデータであるといえる。本実験では各ノードが WWW ブックマークをデータとして一つずつ持つ。ノード間の関

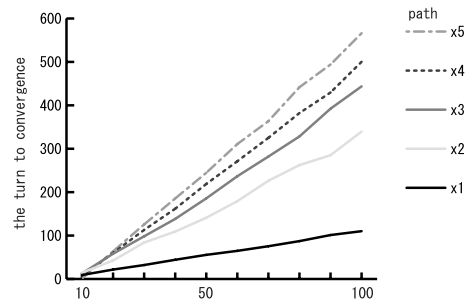
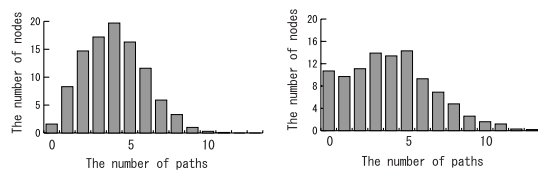


図 9 Average of Convergence Turn



a) best b) converged

図 10 The histogram of the number of paths

係値はカテゴリズ近似度 [8] と呼ばれる値を用いる。これは WWW ブックマーク中にある Web ページの類似と分類の仕方の類似を元にした, WWW ブックマークを持つ人同士の関係を示す値である。

我々は 12 人の被験者から WWW ブックマークを

収集し [8], システムによってそれぞれの WWW ブックマークの関係値 (カテゴライズ近似度) を計算した。収集した WWW ブックマーク数が 12 であるため, ノード数は 12 に固定し, パス数を 12, 16, 20, 24 とした。シミュレーション実験の諸設定を表 2 に示す。シミュレーションは各設定ごとに 10 試行し, その平均値を結果とした。

表 2 The set of parameters (2)

ノード	WWW ブックマーク
ノード間の関係値	カテゴライズ近似度
ノード数	12
パス数	12, 16, 20, 24

図 11 は横軸にターン数, 縦軸に被覆率および到達率をとったグラフである。仮想データを用いた場合と同様の傾向が現れている。

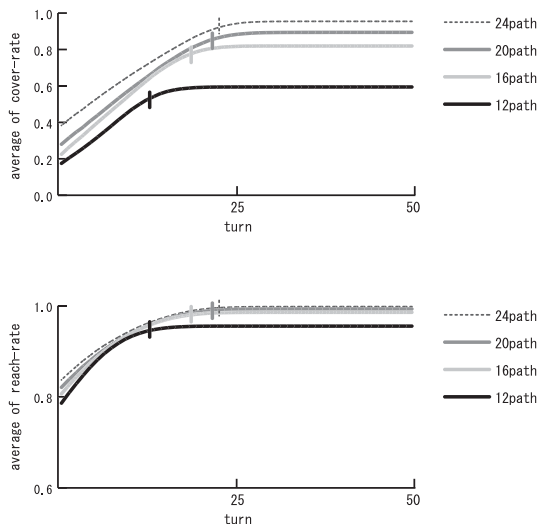


図 11 the average of cover-rate in the real data

6. 考察

6.1 評価値

図 7 および図 8 のグラフを見ると, ノード数が増えるに従い評価値は減少傾向にあり, 逆にパス数が増えると評価値は増加傾向にある。このことから, たくさんのノードによって構成されたネットワークを最適化するためにはたくさんのパス数が必要になるということがわかる。被覆率の場合, パス数がノード数の 1 倍である場合, かなり悪い結果となったが, 到達率の場合は 0.8 程度であり, ほぼ十分な結果が得られてい

るといえる。特にパス数が 2 倍以上の場合はほとんど最適解と同じような値となっている。到達率はネットワークを構成するノード間の関係値の高さを示す値である。これらのグラフから, パス数がノード数の 2~3 倍以上である場合, 提案手法は十分に効果を発揮することがわかった。

6.2 収束性

図 5 および図 6 を見るとが正しくネットワークが収束していることがわかる。仮想データではなく, 実際に人が作ったデータ (Web ブックマーク) を用いた場合でも収束していることが図からわかる。仮想データにおける収束ターン数とパス数およびノード数については図 11 において説明しているが, どのようなノード数, パス数であっても十分なターン数があれば問題なく収束している。NMM による最適化は収束を保証していないが, これらの結果から現実にはほぼ確実に収束するといえる。

6.3 計算量

図 9 はノード数およびパス数と収束に必要なターン数の関係を示したグラフである。ノード数の増加にしたがって収束ターン数は線形に増加していることがわかる。線形増加であるため, ノード数が n 倍になった場合は収束に必要なターン数も n 倍になる。本シミュレーションではパスの張替えは 1 ターンで 1 回行なわれる。よって新しいパスが追加されるノードは 1 ターンにつき 2 つとなる。パスが追加されたノードは, 推薦のために新たな隣接ノード間の関係値計算を行なわねばならない。ここで 1 つのノードが持つ平均パス数を r とすると, 1 ターンで平均 $2r$ 回の関係値計算が行なわれることになる。

ノード数が N の時に収束ターン数が kN で求められるとすると, NMM による最適化に必要な関係値計算回数は $2r(kN)$ となる。対してサーバ型計算法では全てのノード間の関係値を計算するので計算回数は N^2 となる。 r および k は定数であるため, NMM による最適化に必要な計算量は $O(N)$ となりサーバ型計算法の $O(N^2)$ を下回る。

6.4 Anytime Algorithm

NMM による最適化は常にその時点で計算できた最高の解を出力し続ける Anytime Algorithm である。これは動的にノードの状態が変化するようなネットワーク, 例えば人のネットワークなど, において重要な特徴であると考えられる。動的にノードの状態が変化する環境では, ある瞬間において最適であったネット

ワーク構造が、次の瞬間においても最適とは限らない。ゆえに時間をかけて完全な最適解を求める方法は有効であるとはいえない。その点、Anytime Algorithm である NMM は動的に変化するネットワークに対しても柔軟に追従することができると考えられる。また、必ずしも収束するまで計算する必要性がないという点も計算量軽減に大きく役立つといえる。

6.5 自律分散モデル

NMM では個々のノードに対して色々なパス張り替えの戦略を組み合わせることが可能である。人のネットワークに対して適用することを想定した場合、ノードに該当する人は、それぞれ異なる目標を持つはずである。各ノードで異なる評価尺度を持つネットワークの最適化はサーバ型計算法では実現困難であると考えられる。

6.6 人のネットワークへの考察

本手法は人のネットワークでの適用を想定しているが、逆にこのシミュレーション実験から実際の人のネットワークに対して考察を試みる。一般に人間社会においては小さなコミュニティが数多く築かれている。このような状態は図 10 で示した 100 ノード 200 パスの状況に該当する。これは NMM によって十分な最適化を行なうにはパス数が足りない状態である。100 ノード 200 パスの場合、1 ノードあたりのパス数は平均 4 つとなるが、これは現実の人のネットワークを考えると少ない数である。しかしながら、一般に人のネットワークは SmallWorld [9] を形成すると言われている。Small World とはネットワークの構造を示す語で、隣接ノード間が直接パスで結ばれている割合が高く、かつ平均パス長が短いようなネットワーク構造を指す。前者の特徴から、SmallWorld においてはマッチメイキングを行なえるノードが少なくなる。それが結果的に NMM におけるパス数の少なさと同等の効果を現したのだと考えられる。

7. 結 論

本論文において、我々は新しい情報交換のネットワークを拡張する手法として近傍仲介法 (NMM) を提案し、シミュレーションによるサーバ型計算法との比較実験により有効性を検証した。NMM は各ノードが協調的かつ自律的にマッチングを行なうだけで最適化を行なう。しかしながら、それでも十分なネットワークを構築することが可能であることが実験からわかった。本最適化手法は自律分散的であり中央サーバを必要

としていない。これには二つの利点がある。一つは簡単にこの手法を利用できるという点である。ユーザはインターネット上にコミュニティをいつでもどこからでも動的に生成していくことができる。もう一つは、この手法はどのようなサイズのグループに対しても適用可能という点である。なぜなら全ての計算やデータの記録は各ノードが自分の周囲に対して行なうだけで、全てのデータを集め全ての計算をするような中央サーバを必要としないためである。参加するノード数が多いほど、良いペアが生まれる可能性は高くなるので、これは大きな利点であるといえる。

今後は、本手法を用いたシステムを開発し、実際に運用してみることにより有効性を評価したい。

文 献

- [1] Henry Kautz, Bart Selman, Mehul Shah. ReferaWeb: Combining Social Networks and Collaborative Filtering. In *the Communications of the ACM*, Vol. 40, No. 3, 1997.
- [2] Yasuyuki. Sumi, Kenji. Mase. Collecting, Visualizing, and Exchanging Personal Interests and Experiences in Communities. *the 2001 International Conference on Web Intelligence (WI-01)*, 2001.
- [3] Yasuyuki. Sumi, Kenji. Mase. Digital Assistant for Supporting Conference Participants: An Attempt to Combine Mobile, Ubiquitous and Web Computing, *UbiComp 2001*, in G.D.Abowd, B.Brumitt, S.A.N.Shafer (Eds.), *UbiComp 2001*, LNCS 2201, pp.156-175, Springer, Sep. 2001.
- [4] Koji Kamei, Eva Jettmar, Kunihiko Fujita, Sen Yoshida, Kazuhiro Kuwabara. Community Organizer: Supporting the Formation of Network Communities through Spatial Representation. In *Proceedings of the 2001 Symposium on Applications and the Internet (SAINT'01)*, 2001.
- [5] 吉田匡志, 伊藤雄介, 沼尾正行. 口コミによる分散型情報収集システム. *Workshop on Multi-Agent and Cooperative Computation*, 2001.
- [6] 竹内亨, 鎌原淳三, 下條真司, 宮原秀夫. ユーザの関連性に基づいた情報伝播モデル. 第 12 回データ工学ワークショップ, 2001.
- [7] 竹内亨, 鎌原淳三, 佐伯勇, 寺岡伸悟, 原田隆司, 下條真司, 宮原秀夫. 携帯端末を用いた情報伝播モデルによる実験に基づいた情報伝達力の評価. 第 13 回データ工学ワークショップ, 2002.
- [8] 濱崎雅弘, 武田英明, 松塚健, 谷口雄一郎, 河野恭之, 木戸出正継. Bookmark からの共通話題ネットワークの発見手法の提案とその評価. *人工知能学会論文誌*, Vol.17, No.3, pp.276-284, 2001.
- [9] Duncan J. Watts. *Small Worlds*. PRINCETON UNIVERSITY PRESS, 1999.