

応答転送状況を用いた P2P ネットワークの繋ぎ換えアルゴリズムの評価

片山 肇[†] 中野 宏一^{††} 金子 雄[†] 福村 真哉[†] 春本 要^{†††}
西尾章治郎[†]

[†] 大阪大学大学院情報科学研究科 〒 565-0871 大阪府吹田市山田丘 1-5

^{††} 大阪大学工学部 〒 565-0871 大阪府吹田市山田丘 2-1

^{†††} 大阪大学大学院工学研究科 〒 565-0871 大阪府吹田市山田丘 2-1

E-mail: †{katayama.hajime,yu,fukumura,nishio}@ist.osaka-u.ac.jp,

††nakano.hirokazu@ise.eng.osaka-u.ac.jp, †††harumoto@eng.osaka-u.ac.jp

あらまし 近年, P2P ネットワークを利用したアプリケーションが普及している. Gnutella に代表されるフラッディングベースの P2P ネットワークはトラフィックの増大が大きな問題である. また, ピアがもつ情報を考慮せずに論理ネットワークを形成するため, 検索精度が低下する. そこで本論文では, P2P 論理ネットワークのリンク繋ぎ換えアルゴリズムを提案する. 提案するアルゴリズムは, ピアが要求する情報を保持するピアが近くに配置されるようにリンクを繋ぎ換える. これにより, フラッディングする範囲を抑え, 検索精度の高い検索を実現できる. また本論文ではシミュレーション実験によって, 提案アルゴリズムを適用することで検索精度が向上することを示す.

キーワード P2P, 情報検索, ネットワークの繋ぎ換え

A Topology Alteration Algorithm for P2P Networks Based on Response Statistics

Hajime KATAYAMA[†], Hirokazu NAKANO^{††}, Yu KANEKO[†], Shinya FUKUMURA[†], Kaname HARUMOTO^{†††}, and Shojiro NISHIO[†]

[†] Graduate School of Information Science and Technology, Osaka University

1-5 Yamadaoka, Suita, Osaka 565-0871, Japan

^{††} School of Engineering, Osaka University

2-1 Yamadaoka, Suita, Osaka 565-0871, Japan

^{†††} Graduate School of Engineering, Osaka University

2-1 Yamadaoka, Suita, Osaka 565-0871, Japan

E-mail: †{katayama.hajime,yu,fukumura,nishio}@ist.osaka-u.ac.jp,

††nakano.hirokazu@ise.eng.osaka-u.ac.jp, †††harumoto@eng.osaka-u.ac.jp

Abstract Recently, Peer-to-Peer (P2P) applications are becoming popular. The flooding-based P2P networks such as Gnutella have a problem that they generate a large amount of network traffic. Moreover, their recall ratio is generally not good because they are constructed regardless of which peer has which information. In this paper, we propose a topology alteration algorithm for flooding-based P2P networks. The proposed algorithm alters the topology of a P2P network so that a peer can be allocated near the peers that have information the peer often requests. By simulation experiments, we show that the proposed algorithm improves the recall ratio while keeping the network traffic low.

Key words Peer-to-Peer, information retrieval, topology alteration

1. ま え が き

近年, Gnutella, Napster をはじめとする P2P ネットワークを利用したアプリケーションが数多く普及している. P2P ネットワークはピアと呼ばれる端末によって構成され, 各ピアがクライアントとサーバの両方の機能をもつことによってピア自身が情報を管理し発信できる論理ネットワークアーキテクチャである. P2P ネットワークでは従来のサーバ・クライアント型アーキテクチャのようにサーバの負荷が増大せず, またあるピアがサービスを提供できない状態になった場合でも, 他のピアに配置されたキャッシュを利用できる可能性がある. これらのことから, P2P ネットワークはサーバ・クライアント型アーキテクチャに比べ, 可用性の高いデータ共有サービスの実現に非常に有効であると考えられている.

P2P ネットワークにおける代表的な検索手法として, クエリのフラッディングがある. フラッディングでは, ピアが情報を検索する場合, TTL (Time To Live) を定めたクエリを全ての隣接ピアに転送する. クエリを受け取ったピアがクエリにマッチする情報を保持している場合, その情報をレスポンスとしてクエリを転送してきたピアへ返信する. また, クエリを発行したピアからの論理ネットワーク上のホップ数が TTL の値を超えない限り, そのクエリはさらに隣接ピアへ転送される. フラッディングには, TTL を大きな値に設定するほどトラフィックが指数関数的に増加し, ネットワーク全体のパフォーマンスが低下するという問題がある. 逆に, TTL を小さな値に設定すると, クエリがフラッディングされる範囲外にある情報を取得することができない. つまり, TTL の値とレスポンス数の間にトレードオフが存在する.

TTL の値を小さく抑えながら多くのレスポンスを得るためには, 各ピアが保持する情報や各ピアが発するクエリに注目し, クエリが届く範囲内にできるだけ多くの該当情報をもつピアが存在するように P2P ネットワークを構成すればよいと考えられる. しかし従来の P2P ネットワークのトポロジは静的であるため, これを実現するためには何らかの基準によって P2P ネットワークのリンク構造を動的に変化させるアルゴリズムが必要である.

そこで本研究では, より小さな TTL 値の設定でより多くのレスポンスを得ることができるような論理ネットワークのリンク繋ぎ換えアルゴリズムを提案する. 提案するアルゴリズムでは, 各ピアは自身を経由するレスポンスの転送状況を監視する. そして転送の頻度が閾値に達したレスポンスを発見した場合, そのレスポンスの転送先のピアに, リンクを繋ぎ換えるように促す. 繋ぎ換えを促されたピアは, リンクを繋ぎ換えることによる, レスポンスによるトラフィック量の増減をネットワークへの影響として算出する. トラフィック量が減少する場合は良い影響を与えると判断し, 閾値を超えたレスポンスについてその転送先のピアに繋ぎ換えを促す. 悪い影響を与えると判断した場合は, 繋ぎ換えを促したピアに拒否を返す. 最終的に拒否を受けたピアが繋ぎ換え先のピアにリンクを繋ぎ換える. これにより情報を要求するピアとその情報を保持するピアが論理

ネットワーク上の近い位置に再配置されることになる.

以下, 本論文では第 2. 章で関連研究について述べ, 第 3. 章で提案アルゴリズムの詳細について述べる. 第 4. 章で提案アルゴリズムについての評価を行い, 第 5. 章で提案アルゴリズムの課題及び拡張案について述べる. 最後に第 6. 章で本論文をまとめる.

2. 関連研究

P2P ネットワークは主に Hybrid 型と Pure 型に分類される. Hybrid 型 P2P ネットワークではインデックスを一元管理するサーバが存在し, 検索を行う際, ピアは検索したい情報をもつピアをサーバに問い合わせる. しかしインデックスを一元管理するサーバが存在するという点でスケラビリティや可用性に問題があり, 本論文では考慮しない. Pure 型 P2P ネットワークにおける検索手法の代表的なものとして, 分散ハッシュテーブル (DHT: Distributed Hash Table) による方法とフラッディングによる方法がある.

DHT による方法には, CAN [1], Chord [2], Pastry [3], Tapestry [4] など数多くの手法が提案されている. これらの手法ではそれぞれ独自の手法で論理ネットワークが構築され, 管理すべき情報のキー値のハッシュ値に近いハッシュ値をもつピアに, キー値と対応する値 (一般的には該当情報を保持するピアの ID) が配置される. 情報検索時には検索する情報のキー値をハッシュ関数にかけ, 各ピアが保持するルーティングテーブルを参照することによって $O(\log n)$ のホップ数で該当情報を管理するピアにクエリが転送される. ただし n はピア数を表す.

このように DHT を用いれば非常に少ないメッセージ数で検索を行うことができる. しかし, DHT では検索時に単一のキー値しか指定できないため, 例えば宿泊代金 3 万円以下という条件を満たす旅館の検索のように, 一般的な条件比較などによる情報検索は不可能である.

フラッディングによる検索手法は, 複雑な条件比較などを含む任意のクエリに対応できる反面, 前章で述べたように検索トラフィックの増大の問題がある. これを解決する手法として, RI (Routing Indices) [5] や SONs (Semantic Overlay Networks) [6] などの手法が提案されている.

RI では, クエリをフラッディングするかわりに, 隣接するピアのもつ情報を考慮し, 応答の期待できるピアへのみクエリを転送する方式を提案している. ピアは隣接するピアのもつ情報を管理する. クエリを受信すると, 管理する情報をもとに隣接する各ピアから得られる応答の期待値を算出し, 期待値の一番大きいピアへクエリを転送する. これを繰り返すことにより, 要求される情報を保持するピアへクエリを転送することができる. この手法では応答の期待できるピアにのみクエリを転送するため, 不必要なクエリの発生を抑えることができる. しかし RI において論理ネットワークは静的である. これにより情報を保持するピアが論理ネットワーク上の離れたところに存在する場合, ピアは情報を取得できない可能性がある.

SONs ではピアのもつ情報を考慮し, 情報のジャンル別にネットワークを形成する. SONs では情報のファイル名とその

情報に対応するジャンル名を格納するデータベースが存在する。このデータベースを参照することにより、情報に対応するジャンルを一意に決定することができる。ジャンルは root, style, substyle の順に階層化されており、例えば音楽ファイルにおいては root は音楽ファイル全体を表し、style は JAZZ や ROCK, ROCK の substyle には HARD ROCK, POP などが含まれる。ピアは、自身が持つファイルのジャンルに該当するネットワーク (SON) に参加し、その SON 内に属する他のピアとリンクする。検索の際は、情報の分類と同様にクエリをジャンルに分類し、該当する SON へ転送する。SONs では情報のジャンルごとにネットワークを形成するため、検索精度が非常に高い。ただしこの手法は情報が明確にジャンルに分類できる場合に限る。情報が明確に分類できない場合は、一般的な条件比較などを含むクエリに対してはどの SON にそれを転送すればよいか判断できないため適用できない。また、一つの SON 内においてクエリはフラッディングされるため、トラフィック増大の問題は部分的にしか解決できていない。

本論文では一般的な条件比較などを含む任意のクエリを想定し、そのような状況で有効な方式を提案する点で既存研究とは大きく異なる。

3. 提案アルゴリズム

本章では論理ネットワークのリンクを繋ぎ換えるアルゴリズムについて述べる。

3.1 想定環境

ピアは必ず 1 つ以上のピアとリンクを保持するものとし、リンク先のピアを隣接ピアとする。ここでリンクをもつとは相手の IP アドレスを保持することを表す。ピアは全体で 1 つの論理ネットワークを形成しており、ピアのネットワークへの参加、ネットワークからの脱退、ピアの故障については考慮せず、論理ネットワークが分断することはないものとする。ピアは検索を行う際に、既存の P2P ネットワークと同様に、クエリを論理ネットワーク上でフラッディングすることによって必要な情報を得る。クエリを受信したピアが要求された情報を保持する場合は、そのクエリを転送してきたピアに対してレスポンスを送信する。レスポンスは、クエリの転送されてきた経路を逆向きに転送される。

3.2 評価値の算出

リンクの繋ぎ換えが行われると論理ネットワークのトポロジーが変化する。ネットワークトポロジーの変化によって、ホップ数が短縮される経路もあれば、ホップ数が延長される経路もある。ホップ数が短縮される経路においては、小さいホップ数で検索できるが、ホップ数が延長される経路においては、検索時に大きなホップ数が必要となる。ホップ数の延長される経路経由のレスポンスが多い場合は、リンクを繋ぎ換えないほうが良い可能性がある。そこでリンクを繋ぎ換えた後における、得られるレスポンス数が減少する可能性を考慮するために、ピアはリンクの繋ぎ換え前に、繋ぎ換えによるレスポンス数の増減を評価値として算出する。ピアは繋ぎ換えを行うかを評価値をもとに判断する。評価値は次の式で計算する。

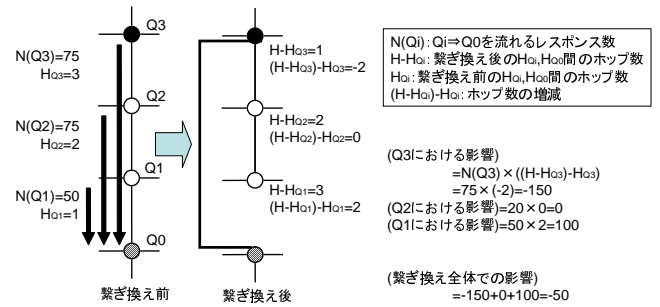


図 1 影響の算出

Fig. 1 Calculation of Influence .

$$\sum_i N(Q_i) \times ((H - H_{Q_i}) - H_{Q_i})$$

Q_i は、繋ぎ換えで注目しているレスポンスの流れる経路上にあるピアを表し、評価値を算出するピアを Q_0 として Q_0 から近い順に Q_1, Q_2, \dots と割り当てる。 $N(Q_i)$ は、ピア Q_{i+1} を経由せず、ピア $Q_i \sim Q_1$ を経由して Q_0 に流れるレスポンスの件数を表す。 H は Q_i, Q_1 間のホップ数、リンクの繋ぎ換えにより切断される Q_1, Q_0 リンクの 1 ホップ、新しくリンクを繋ぐ Q_i, Q_0 リンクの 1 ホップの合計を表す。 H_{Q_i} は、リンクを繋ぎ換える前における Q_i, Q_0 間のホップ数を表す。したがって $H - H_{Q_i}$ は、リンクを繋ぎ換えた後における Q_i, Q_0 間のホップ数を表す。そして $(H - H_{Q_i}) - H_{Q_i}$ の値が繋ぎ換えによって増減する Q_i, Q_0 間のホップ数を表す。 $(H - H_{Q_i}) - H_{Q_i}$ の値がマイナスの場合は、リンクの繋ぎ換えによって Q_i, Q_0 間のホップ数が減少することを意味し、プラスの場合は Q_i, Q_0 間のホップ数が増加することを意味する。さらにこの $(H - H_{Q_i}) - H_{Q_i}$ と $N(Q_i)$ との乗をとることで、リンクの繋ぎ換えによって増減する Q_i, Q_0 間を流れるレスポンス数を得ることができる。最後に $Q_i \sim Q_0$ 間の全てのピアについて $N(Q_i) \times ((H - H_{Q_i}) - H_{Q_i})$ の総和をとることで、リンクを繋ぎ換えることによって増減するメッセージ数を求めることができる。ピアは、算出結果がマイナスの場合、リンクの繋ぎ換えによって良い影響を与えると判断し、算出結果がプラスの場合は悪い影響を与えると判断する。

例えば図 1 のように、 Q_0 と Q_1 を結ぶリンクを切断し、新しく Q_0 と Q_3 をリンク接続する場合を考える。まず Q_3 について影響を算出すると $H = 4, H_{Q_3} = 3, H - H_{Q_3} = 1$ となり、繋ぎ換えによるホップ数の増減は $(H - H_{Q_3}) - H_{Q_3} = -2$ となる。 $N(Q_3) = 75$ であることから、繋ぎ換えによる Q_3 における影響は $N(Q_3) \times ((H - H_{Q_3}) - H_{Q_3}) = -150$ となる。同様に、 Q_2 については $N(Q_2) \times ((H - H_{Q_2}) - H_{Q_2}) = 25 \times 0 = 0$ となり、 Q_1 については $N(Q_1) \times ((H - H_{Q_1}) - H_{Q_1}) = 50 \times 2 = 100$ となる。よって $Q_3 \sim Q_1$ におけるそれぞれの評価値の合計が $-150 + 0 + 100 = -50$ であることから、ピア Q_0 は、 Q_1 とのリンクを切断して Q_3 へリンクを繋ぎ換えることはネットワークへ良い影響を与えると判断する。

3.3 転送履歴チェック

ピアはレスポンスの転送履歴を保持する。保持する転送履歴

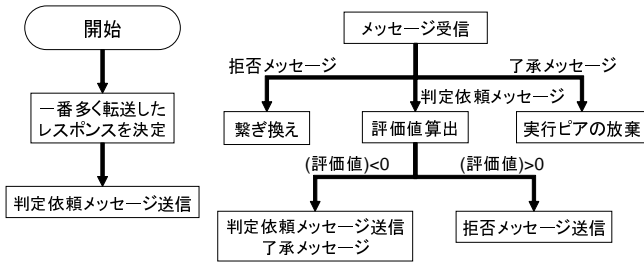


図 2 フローチャート
Fig. 2 Flow chart.

情報は、転送元、転送先、情報の種類とする。ここでは簡単のため、ピアへレスポンスを転送してきた隣接ピアを転送元とし、ピアがレスポンスを転送した先の隣接ピアを転送先とする。ピアは、自身の保持するレスポンスの転送履歴を不定期にチェックする。そして転送元、転送先、情報の種類の組み合わせの中で、一番多い組み合わせのレスポンスに関して 3.4 節に示すアルゴリズムを実行する。

3.4 アルゴリズムの実行

ピアは転送履歴チェックによって、転送した数の一番多いレスポンスに関して繋ぎ換えのアルゴリズムを実行する。以下、繋ぎ換えのアルゴリズムを実行するピアを実行ピア、リンクの繋ぎ換え先となるピアをソースピアとして、アルゴリズムの内容を図 1 を例に挙げて説明する。アルゴリズムのフローチャートは図 2 に示すとおりである。なお図 1 において初期状態として、ピア Q_1 を実行ピア、ピア Q_3 をソースピアとする。

(1) ピア Q_1 は、転送頻度が高いと判断したレスポンスについて、そのレスポンスの転送先のピア Q_0 に対して判定依頼メッセージを送信する。判定依頼メッセージはピア Q_0 に対して、ソースピアとリンクを張り、実行ピアとのリンクを切るように促すためのメッセージである。

(2) 判定依頼メッセージを受信したピア Q_0 は、自身を経由するレスポンスにおいて、ピア Q_1 とのリンクを切断し、ピア Q_3 へリンクを繋ぎ換えた場合の影響を評価値として算出する。評価値の算出方法は 3.2 節の通りである。

(3) 評価値がマイナスだった場合、ピア Q_0 はピア Q_1 に了承メッセージを送信する。了承メッセージは、ピアが実行ピアから実行ピアの権限を引き継ぐことを了承するメッセージである。このメッセージを受信したピア Q_1 はアルゴリズムの実行権限をピア Q_0 に委託する。これにより新たにピア Q_0 が実行ピアとなり、手順 1 に戻る。

(4) 評価値がプラスだった場合、ピア Q_0 はピア Q_1 に拒否メッセージを送信する。拒否メッセージは、ピアが実行ピアから実行ピアの権限を引き継ぐことを拒否するメッセージである。このメッセージを受信したピア Q_1 はピア Q_3 へリンクを繋ぎ換える。ただし実行ピアがソースピアに既にリンクしている場合、リンクの繋ぎ換えは行わない。例えば転送履歴をチェックしてアルゴリズムを実行したピアは、既にソースピアとリンクしているため、実行ピアは拒否メッセージを受信しても繋ぎ換えは行わず、アルゴリズムを終了する。また拒否メッセージ

表 1 評価環境

Table 1 Simulation Parameters .

ピア数	10000
情報の種類	400
情報のグループ数	20
保有可能な情報の種類数	10
クエリの発行確率	0.003
TTL	5~7
管理するレスポンスの転送履歴数	100
レスポンス転送履歴のチェック確率	0.003
シミュレーション時間	20000

を受信したピアがリンクを繋ぎ換える際、ソースピアの保持する隣接ピア数が 20 以上の場合は、ピアはソースピアの隣接ピアとリンクする。

こうして判定依頼メッセージを受けたピアが、実行ピアに拒否メッセージを送信し、実行ピアがソースピアへのリンクの繋ぎ換えを行うまで上記の手順が繰り返される。

4. 評価

本章では、提案アルゴリズムの性能を評価するためのシミュレーション実験結果を示す。本シミュレーションは、提案アルゴリズムの初期的な実験評価として行われたものである。

4.1 評価環境

シミュレーション実験における評価環境を表 1 に示す。まずピアおよびピアの保持する情報についての評価環境を述べる。本シミュレーションにおいて、P2P ネットワークに存在するピアの数は 10000 とし、0~9999 のピア ID を割り当てる。ネットワーク内に存在する情報の種類は 400 種類とし、0~399 の情報 ID を割り当てる。ピアの保持する情報や検索する情報に偏りをもたせるために、ピアおよび情報を 20 のグループに分類した。グループには 0~19 のグループ ID を割り当てる。各グループに属する情報数は 25 とし、グループ i に属する情報のうち 5 つはグループ $i+1$ にも属する。 i はグループ ID を表す。ピアの保持する情報は次のようにして決定する。ピアは 20 のグループの一つをランダムに選択する。ピアが保有可能な情報の種類数を 10 種類とし、そのうち 80% をピアの属するグループからランダムに選択する。残りは他のグループからランダムに選択する。

ネットワーク内では、ピアは必ず一つ以上の隣接ピアをもつものとする。ネットワーク内では、予めピアが一つの大きな論理ネットワークをランダムに形成しているものとする。またピアのネットワークへの参加、ネットワークからの退出、ピアの故障は考慮せず、これによって論理ネットワークが変化することはないものとする。

次にネットワーク内でのピアの発行するクエリ、およびレスポンスについて述べる。ピアがクエリを発行する確率は、1 タイムスロットあたり 0.003 とした。ここでピアの発行するメッセージが 1 ホップ流れる時間を 1 タイムスロットとし、受信したメッセージは 1 タイムスロットの間に処理される。なおクエ

りの伝搬には、既存の Pure 型 P2P ネットワークにおける代表的な検索手法のフラディングを用いる。TTL は、4~7 のいずれかの整数値で統一する。ピアは検索するグループを選択し、そのグループからランダムに検索する情報を決定する。

クエリに付加された情報の種類に該当するピアは、クエリを発行したピアにレスポンスを送信する。レスポンスはクエリの転送された経路を逆向きに転送される。レスポンスを転送する際に、レスポンス転送先へのリンクが切断されていた場合、ピアはそのレスポンスに対応するクエリを発行したピアへレスポンスを直接転送する。レスポンスを直接転送するために、レスポンスには対応するクエリを発行したピアのアドレスが予め付加されているものとする。レスポンスを受信したピアは、転送するレスポンスの情報を転送履歴として管理する。ピアは履歴として転送元、転送先、情報の種類についてそれぞれピア ID、情報 ID で管理する。管理するレスポンス転送履歴数は新着 100 件としている。ピアがレスポンスの転送履歴をチェックする確率は 0.003 とした。

以上の環境において 20000 タイムスロットのシミュレーション実験を行い、提案方式の性能評価を行った。

評価対象は以下に示すとおりである。

- 評価対象 1

ピアの保持する情報と検索する情報に相関性があるネットワーク

- 評価対象 2

ピアの保持する情報と検索する情報に相関性がないネットワーク

評価対象 1 において、ピアは検索するグループを選択する際、80%の確率で自身の属するグループを選択し、残り20%で他のグループを選択する。ピアの検索する情報がピアの保持する情報に関連付けられていることから、実世界の想定により近い環境といえる。評価対象 2 において、ピアは以下の式を用いてグループ i を決定する。

$$P(i) = \frac{f(A(i)) - f(A(i+1))}{f(A(i)) - 0}$$

$P(i)$ はグループ i が選択される確率を表す。 $A(i)$ はグループ i の重みを表し、 $A(i) = i$ とする。 $f(x)$ は以下に示す Zipf 分布の式により決定する。 $f(x) = 10 \times ((X + 1) \times 0.014^{-1.08})$

なお評価項目は次のとおりである。

- 1 検索あたりの平均被覆率
- ピアの保持するリンク数
- ネットワークトポロジの変化
- ネットワーク内のトラフィック

以下では、シミュレーション評価の結果を示し、考察を行う。ただしネットワークトポロジの変化においては、ピア数は 100、情報の種類は 80、情報のグループ数は 4、TTL は 4 としている。

4.2 1 検索あたりの平均被覆率

1 検索あたりの平均被覆率は、ピアが 1 度の検索で得たレスポンス数と、検索に該当するネットワーク全体の情報数との商の平均によって得ることができ、検索に該当する情報の取得率を表す。提案アルゴリズムを適用しない状態で 2000 タイムス

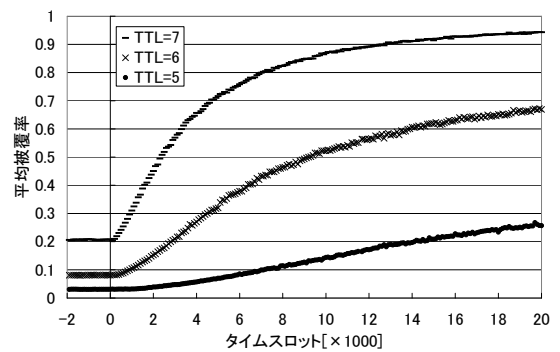


図 3 1 検索あたりの平均被覆率 (評価対象 1)

Fig. 3 Average cover rate (evaluation category 1).

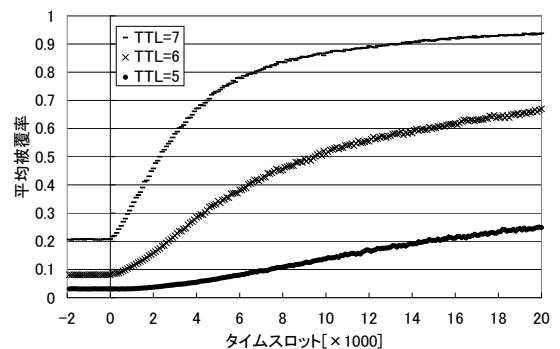


図 4 1 検索あたりの平均被覆率 (評価対象 2)

Fig. 4 Average cover rate (evaluation category 2).

ロット経過させた後に提案アルゴリズムを適用し、その際の時間推移による 1 検索あたりの平均被覆率の変化についての評価結果を図 3、図 4 に示す。

図 3 によると TTL を 7 に設定した際、提案アルゴリズム適用前における 1 検索あたりの平均被覆率は 0.2 で一定であった。しかし提案アルゴリズムを適用することにより、タイムスロットの経過に伴い平均被覆率は増加し、6000 タイムスロット経過後には平均被覆率が 0.8 に到達している。このことからタイムスロットの経過に伴いリンクの繋ぎ換えが実行され、リンクの繋ぎ換えによってピアの論理ネットワーク上の周辺に、検索したい情報をもつピアが集合したことが分かる。

また図 3 と 4 の差異が微少であることから、ピアの保持する情報と検索する情報の相関性によって平均被覆率が左右されることはない。これにより提案アルゴリズムはあらゆる検索傾向においても有効であるといえる。

4.3 ピアの保持するリンク数

提案アルゴリズムを適用して 20000 タイムスロット経過した後における、ピアの保持するリンク数分布を図 5、図 6 に示す。図 5 によると、20000 タイムスロット経過後は、リンク数が 2、または 3 であるピアが全体の約 75% を占めている。このことからリンクが極少数のピアに集中していることが分かる。なお図 5 と図 6 においても、4.2 節と同様に、評価対象 1 と 2 に差異はない。

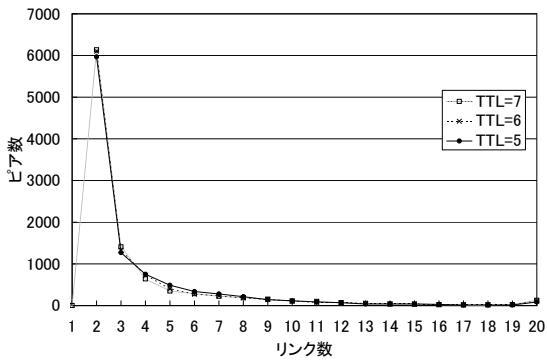


図 5 ピアの保持するリンク数 (評価対象 1)

Fig. 5 The number of links (evaluation category 1).

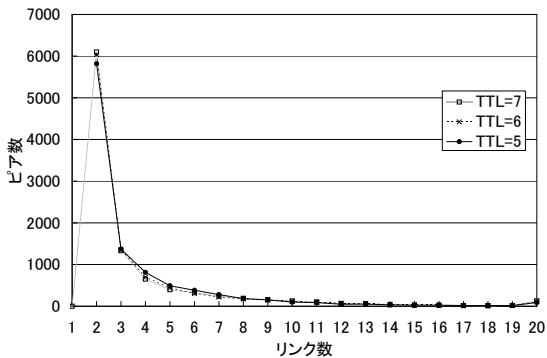


図 6 ピアの保持するリンク数 (評価対象 2)

Fig. 6 The number of links (evaluation category 2).

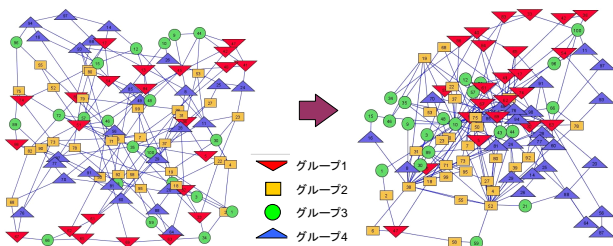


図 7 トポロジの変化

Fig. 7 Change of network topology.

4.4 ネットワークトポロジの変化の様子

評価対象 1 において提案アルゴリズムを適用した際の、ネットワークトポロジの変化を図 7 に示す。ネットワークトポロジの描画には GraphExplore [7] を利用した。左側が提案アルゴリズム適用前の論理ネットワークのトポロジを表し、右側が提案アルゴリズム適用して 20000 タイムスロット経過後の論理ネットワークのトポロジを表す。図 7 における情報のグループ別のピア数および、同じグループに属するピア同士で接続されているリンク数の変化は表 2 に示すとおりである。図 7 および表 2 より、20000 タイムスロット経過後には、提案アルゴリズム適用前にランダムに構成されていた論理ネットワークのトポロジが変化し、同じグループのピア同士が論理ネットワーク上の近くに配置されていることが分かる。

表 2 同一グループに属するピア同士のリンク数

Table 2 The number of links between peers in the same group.

グループ ID	ピア数	アルゴリズム適用前	適用後
1	23	12	21
2	30	15	29
3	21	8	11
4	26	14	11

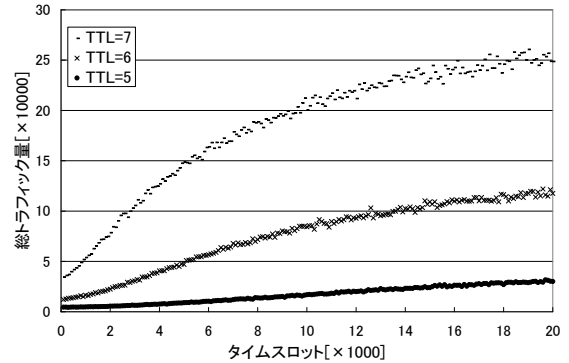


図 8 ネットワーク内のトラフィック量

Fig. 8 Traffic.

4.5 ネットワーク内のトラフィック量

評価対象 1 において提案アルゴリズムを適用した際のネットワーク内のトラフィック量を図 8 に示す。図 8 では、提案アルゴリズムの適用と同時にトラフィックが増加する。TTL を 7 に設定した場合、20000 タイムスロット経過後にはトラフィック量が 250000 にも達している。これは図 5、図 6、図 7 から分かるように、隣接ピアを多数保持しているピアが存在し、さらにそれらのピアが論理ネットワーク上の近くに存在するためである。これによってピアはクエリを受信するとクエリをフラッディングし、全ての隣接ピアにクエリを転送するためトラフィックが膨大になる。トラフィックを軽減するための拡張案は 5.6 節で述べる。

5. 提案アルゴリズムの拡張

提案アルゴリズムにおいて、現状では様々な問題が発生する。本章では、それらの問題を解決するための提案アルゴリズムの拡張案について考察する。

5.1 繋ぎ換えによるホップ数増減の決定方法

評価値算出において、リンクの繋ぎ換え後レスポンスはその経路および繋ぎ換えによって接続されるリンクを経由するという前提で、ピアは転送頻度の一番高いレスポンスの転送経路のみを利用してピア間のホップ数増減を決定する。しかし実際は、レスポンスをその経路以外の経路に転送することによって、レスポンスを転送するホップ数がさらに抑えることができる場合がある。図 1 で例を挙げると、リンクの繋ぎ換え後において、 Q_1 から Q_0 へレスポンスを送信する場合、図 1 では 3 ホップ必要としている。ところが Q_1 が Q_2 以外の隣接ピアへレスポンスを転送することにより、2 ホップで Q_0 へ転送できる可能性がある。このような経路を迂回経路とする。もし迂回経路へ

転送することによって2ホップで転送できる場合は、 Q_0 が評価値を算出する際に、リンク繋ぎ換え後の Q_1, Q_0 間のホップ数は2とする必要があったと考える。そこでピアはより正確な評価値を算出するために、ピア周辺のネットワークポロジを予め把握しておくことが必要である。

拡張案としては、評価値を算出する際に、評価値を算出するピアが、迂回経路を考慮しない場合のホップ数増減の値を TTL に設定し、メッセージをフラッディングすることを検討している。そのメッセージが、指定した TTL 未満でピア $Q_i (i > 1)$ へ到達した場合は、迂回経路を考慮する必要があると判断し、ホップ数増減の値を迂回経路経由時の値に更新する。これによってより正確な評価値の算出が可能となる。

5.2 転送履歴チェックの処理

ピアは不定期にレスポンスの転送履歴をチェックし、転送されたレスポンス数が最大のレスポンスに関して提案アルゴリズムが実行される。ただしピアは最大数のレスポンスに関して、その数については考慮しない。他のレスポンスとの転送数の差が微少の場合でも、ピアは提案アルゴリズムを実行する。そのため現状では提案アルゴリズムの実行を開始したとしても、提案アルゴリズムを実行したピアの送信する判定依頼のほとんどが拒否され、繋ぎ換えは行われずアルゴリズムが終了する。

そこで拡張案として、転送数が最大のレスポンスと他のレスポンスについて、それぞれの転送数を統計的に検討する。まずレスポンスの転送される経路は等確率で選択されるものと仮定する。このときピアの管理するレスポンス 100 件について、各転送経路を経由するレスポンス数は正規分布に依存する。これを利用して転送経路を選択する確率の確からしさを算出する。確からしさが否定された場合、転送数が最大のレスポンスは他のレスポンスに比べて過剰であると判断する。

5.3 評価値の算出に必要なレスポンスの取得方法

提案アルゴリズムでは、転送頻度の一番高いレスポンスの転送される経路上で提案アルゴリズムが実行される。実行ピアから判定依頼メッセージを受信したピアは、3.4 節で述べた処理を行う。その際他のレスポンスも含めて、リンクの繋ぎ換えによってホップ数が短縮されるかどうかの期待値を、評価値として算出する。評価値を算出するためには、転送頻度の高いレスポンスの転送経路上において、ソースピアと評価値を算出するピア間にあるピアの管理するレスポンスの転送経路および転送数を必要とする。図 1 を例に挙げると、ピア Q_0 は評価値を算出する際、 Q_2 を経由せず Q_1 を経由して Q_0 に転送されるレスポンス数や、 Q_3 を経由せず Q_2, Q_1 を経由して Q_0 に転送されるレスポンス数を取得する必要がある。しかし現状では、提案アルゴリズムにおいてそれらの情報の取得方法は考慮されていない。ピアはレスポンスの転送経路の情報を予め取得しているという前提のもとで提案アルゴリズムを実行している。

そこで拡張案として、レスポンスの転送経路情報を取得する方法について検討する。取得方法として、実行ピアが判定依頼メッセージを送信する際に、実行ピア自身を経由するレスポンスの情報を付加してメッセージを送信する方法が考えられる。同時に、実行ピアが実行ピアの権限を取得する以前において、

実行ピアの権限を保持していたピアから判定依頼メッセージを受信した際に、メッセージに付加されていたレスポンスの転送の情報もあわせて付加する。図 1 で例を挙げると、 Q_3 が Q_2 に判定依頼メッセージを送信する場合、 Q_3 は自身を経由して Q_2 に転送されるレスポンスについて、転送元、転送先、レスポンス数を付加して送信する。続いて Q_2 が Q_1 に判定依頼メッセージを送信する場合、 Q_2 は、 Q_3 の送信した判定依頼メッセージに付加されていた情報と Q_2 自身を経由して Q_1 に転送されるレスポンスの情報を付加してメッセージを送信する。以下 Q_2 が行った処理を繰り返すことによってピアはレスポンスの転送されてきた経路とその転送数を得ることができ、評価値を計算できる。

5.4 リンク接続のみによる繋ぎ換えアルゴリズム

提案アルゴリズムにおいて繋ぎ換えは、実行ピアとのリンクを切断し、ソースピアとリンクすることにより実現される。

しかしピアの保持できる隣接ピア数に余裕があり、実行ピアとのリンクを切断する必要がない場合もある。今後はリンク接続のみのアルゴリズムについて検討し、提案アルゴリズムと併用する。

5.5 ピアの参加、退出、故障時のアルゴリズム

提案アルゴリズムでは、ピアのネットワークへの参加、ネットワークからの退出、ピアの故障を考慮していない。実際はピアの参加、退出、故障によって論理ネットワークのトポロジも変化するため、それらを考慮する必要がある。例えばピアのネットワークへの参加を考慮する場合を考える。ピアはネットワークへ参加する際、ピアの検索したい情報をもつピアおよびその周辺のピアに接続することができれば、高い検索成功率を実現できる。さらに検索したい情報が予め近くにあるため、提案アルゴリズムの実行回数を減らすことができ、ピアの負荷を軽減することができる。

ピアのネットワークへの参加を実現するためには以下のようなことが必要となる。ネットワーク内の一部のピアを IP アドレスを既知のピアとし、ピアのネットワークへの参加を受け付けるピアに設定する。ネットワークへ参加するピアは、ピアのネットワークへの参加を受け付けるピアを通じて周辺にあるピアのもつ情報を知り、必要なピアと接続できる。

5.6 ネットワーク内のトラフィックの削減

提案アルゴリズムにおいては、ピアの保持する隣接ピア数の上限を 20 と制限している。これによりピアへのリンクの集中を避けることはできる。しかしクエリが隣接ピア数の多いピアを経由するとクエリは膨大になる。

そこでトラフィックを削減する方法として、以下の3つの方法が考えられる。

- 不必要なリンクの切断
- フラッディング以外によるクエリの転送
- TTL の動的な設定

まず不必要なリンクを切断するリンク切断アルゴリズムを検討する。ピアは、レスポンスの転送数が少ない、またはレスポンスが一定期間転送されていないリンクを保持している可能性がある。これらのリンクを切断することによってピアの保持す

る隣接ピア数を削減できる．これによってネットワーク内のトラフィックを削減できる．

次にフラッディング以外によるクエリの転送方法として，一部の隣接ピアへのみクエリ転送する方法を検討する．クエリをフラッディングするのではなく，クエリの転送先を応答の期待できる隣接ピアのみに限定することによってトラフィックを削減できる．ピアはレスポンスの転送履歴を管理しているため，レスポンスの転送履歴を利用し，応答の期待できる隣接ピアを決定することは容易である．

最後にクエリの TTL を動的に決定する方法を検討する．提案アルゴリズムのシミュレーションでは TTL が静的であったため，トラフィックが膨大に発生したと考えられる．そこで 1 検索あたりの平均被覆率をもとにして，被覆率に応じて TTL を動的に変化させる．これにより被覆率の向上に伴ってクエリをフラッディングする範囲を縮小することができトラフィックの増大を抑制できる．また被覆率が低下した場合は TTL を大きく設定することによって被覆率を向上することができるため，一定の被覆率を維持できる．

6. ま と め

本論文ではレスポンスの転送状況を監視し，転送頻度の高いレスポンスを発見した場合に，そのレスポンスの経路に関して論理ネットワークを繋ぎ換えるアルゴリズムを提案した．これにより情報を要求するピアと，その情報を保持するピアを論理ネットワーク上の近傍に再配置することができた．またシミュレーションによる評価を行った．その結果から，提案アルゴリズムを適用することにより，論理ネットワークトポロジが組織化され，検索時に得られる情報数が増加することを確認した．なお提案アルゴリズムはまだ初期的なアルゴリズムであり，今後はピアの参加，退出，故障時のアルゴリズム，リンク接続切断アルゴリズム，トラフィックの削減などを考慮したアルゴリズムへの拡張を行う予定である．

謝 辞

本研究の一部は，平成 15 年度総務省「ユビキタスネットワーク認証・エージェント技術の研究開発」の研究助成によるものである．また，本研究の一部は，文部科学省 21 世紀 COE プログラム「ネットワーク共生環境を築く情報技術の創出」(研究拠点形成費補助金)の研究助成によるものである．ここに記して謝意を表す．

文 献

- [1] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker: "CAN: A scalable content-addressable network," in *Proc. SIGCOMM '01*, pp. 161-172.
- [2] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan: "Chord: A scalable peer-to-peer lookup service for internet applications," in *Proc. SIGCOMM '01*, pp. 149-160.
- [3] A. Rowstron, and P. Druschel: "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems," in *Proc. Middleware 2001*, pp. 329-350.
- [4] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph: "Tapestry: An infrastructure for wide-area fault-tolerant

location and routing," *U. C. Berkeley Technical Report UCB//CSD-01-1141*.

- [5] A. Crespo and H. Garcia-Molina, Routing indices for peer-to-peer systems, In *ICDCS*, 2002, pp. 23-33.
- [6] A. Crespo and H. Garcia-Molina: Semantic overlay networks for p2p systems, Technical report, Computer Science Department, Stanford University, 2002.
- [7] W. Quanli, Y. Guang, N. Joseph, W. Mike and D. Adrian: "GraphExplore: A Software Tool for Network Visualization," Duke University, 2004.