

ブローカレス型ネットワークにおける 分散共有方式の提案とその性能評価

北橋 洋三郎^{†1} 星 合 擁 湖^{†2}
三 友 仁 志^{†3} 星 合 隆 成^{†4}

ブローカレス型 (P2P) ネットワークの実現技術は、中央集権的な管理者・仲介者 (ブローカ) が存在しないことを前提とするブローカレスの概念を適用するレイヤに応じて、理論的に分類・体系化することが可能である。この理論はディメンションモデルと呼ばれる。本論文は、ディメンションモデルにおける探索・グルーピングレイヤを対象とし、特にフラット型で実現する P2P 技術の基本原則を技術解説するとともに、シミュレーション実験による性能評価の結果を示す。次に、従来技術における 1 つの問題点として、ピアの参加・退去を考慮した実用的な条件下では高スケーラビリティとネットワークの収束性が両立しない点を指摘する。その解決策として、リレーションシップの双方向制御による分散共有方式を提案し、最小限のオーバーヘッドでネットワークの収束性を向上させることにより、実用的な条件下でも高い性能が得られることを示す。

A Proposal of a Decentralized Collaboration Architecture on Brokerless Network and its Evaluation

YOZABURO KITAHASHI,^{†1} YOKO HOSHIAI,^{†2} HITOSHI MITOMO^{†3}
and TAKASHIGE HOSHIAI^{†4}

Technologies for brokerless (P2P) network can be classified hierarchically in the theory called "dimension model". The theory models what layers the concept of brokerless, or no intervention of brokers such as administrators and mediators targets on. This paper focuses on the flat model in the discovery and grouping layer defined in the theory. The existing architectures are evaluated quantitatively and studied by the simulations, revealing that the high scalability and the convergence are practically not compatible features. So then we design a decentralized collaboration architecture using bidirectional relationships, and demonstrate that the architecture keeps the performance high under the practical conditions with the minimum overhead.

1. ま え が き

ブローカレス型 (P2P) ネットワークを理論的に体系化する試みとして、ディメンションモデル¹⁾ が知られている。ディメンションモデルとは「情報を一元的に管理・仲介するブローカが存在しない」ことを前提とするブローカレスの考え方を、どのようなターゲット (レイヤ) に対して適用するかを示したものである。

ディメンションモデルでは、ブローカレスの適用領域として「伝達」「探索・グルーピング」「配信」「ポリシ」の 4 つのレイヤを規定している。

探索・グルーピングレイヤを対象とする技術には、SIONet^{1),2)}, Gnutella³⁾, Freenet^{4),5)}, Winny⁶⁾, KaZaA⁷⁾, Skype⁸⁾, JXTA⁹⁾⁻¹¹⁾, Chord^{12),13)}, SOBA¹⁴⁾ などがあげられる。これらは Chord を除き、配信レイヤもターゲットとしている。さらに、JXTA は伝達レイヤ、SIONet はポリシレイヤまでを適用領域とする技術である。これらの技術は、これまで個々の製品や特定の用途について独立に論じられることが多く、各方式を共通の技術的観点から相互に比較し、評価する試みはなされてこなかった。とりわけ複数のレイヤをターゲットとする技術の場合、各レイヤごとの技術的特性が総合的な性能にそれぞれ影響を及ぼす

†1 NTT アドバンステクノロジー株式会社
NTT Advanced Technology Corporation

†2 NTT サービスインテグレーション基盤研究所
NTT Service Integration Laboratories

†3 早稲田大学大学院
Graduate School, Waseda University

†4 NTT ネットワークサービスシステム研究所
NTT Network Service Systems Laboratories

ために、特徴ある要素技術の本質が隠されてしまうことが少なくない。

本論文の第1の目的は、既存のP2Pネットワーク技術を体系的に分類し、各方式の技術的特徴の分析と定量的な性能比較を行うことである。その対象を探索・グルーピングレイヤの技術に限定することで、評価のポイントを明確化する。ここで扱う問題は、ファイルや個人情報など任意の情報を抽象化した「メタ情報」のインデックスサーチ、すなわちユニークなピアの特定である。特定のP2Pアプリケーションの評価は目的としていないため、たとえばファイル交換システムにおけるファイル複製の効果など、アプリケーション（あるいは対象とするメタ情報）特有の機構については扱わない。

探索・グルーピングレイヤの技術は、「フラットモデル（すべてのピアが対等な立場で分散・協調するモデル）」および「クラスタモデル（一部のピアに特別な役割を持たせたモデル）」と呼ぶ分散共有モデルに大別することができる。クラスタモデルの性能評価については、すでに報告済みである¹⁵⁾。2章では、「値域分割方式」を除くフラットモデルを対象として、従来方式の基本原理解説するとともに、シミュレーション実験による性能の比較・評価結果について述べる。

本論文の第2の目的は、実験結果より明らかになった従来技術の課題に対し、その解決法を考案し、有効性を評価することである。3章では、前章のシミュレーション実験の結果、最も優れた性能を示した「リレーションシップ方式」が、ピアが頻りに退去する実用的な条件下では不安定になる問題点を指摘する。さらに、その原因がリレーションシップの非対称性に起因する学習内容の喪失にあることを明らかにし、リレーションシップの双方向制御による解決法を提案する。

4章では、シミュレーション実験により提案手法の有効性を示すとともに、その効果について事例を用いた検証と考察を行う。

2. 従来手法

探索・グルーピングレイヤをターゲットとするP2Pネットワークの実現技術は、ブローカを介することなく、ピアが相互に分散・協調することによってリソースを共有するための仕組みを提供するものである。具体的には、以下の要素から構成される。

- メンバシップ管理
- アドバタイズ（公開）
- ディスカバリ（発見）
- リプライ

ここでは、これらの要素技術を総称して『情報の分散共有技術』と呼ぶ。本章では、情報の分散共有技術に対する従来の取組みについて述べる。

2.1 情報の分散共有技術の分類

情報の分散共有技術は、『フラットモデル』および『クラスタモデル』に分類することができる。フラットモデルとは、P2Pネットワークを構成するすべてのピアが対等な立場で分散・協調することによって情報を共有するモデルである。これに対してクラスタモデルとは、ベースピアと呼ばれるピアに特別な役割を持たせ、ネットワークのバックボーンとして機能させるモデルである¹⁵⁾。フラットモデルでは、クラスタモデルにおけるベースピアのような、特別な役割を担ったピアは存在しない。表1に示すように、Gnutella, Winny, Freenet, Chord, SOBA（ディレクトリサービス）がフラットモデル、KaZaA, Skype, JXTAがクラスタモデルの代表的な例である。またSIONetは、運用形態によってクラスタモデルとフラットモデルのいずれにも適用可能である。本論文では、上述の分類のうちフラットモデルを対象として論じる。

フラットモデルでは、『リンク』および『リレーションシップ』と呼ばれる情報を用いてP2Pネットワークが構築される。ここで、情報の転送先となるピアのリファレンス（IPアドレスなど）を用いて、当該ピアに対して情報の転送路を確立することをリンク確立と呼び、当該転送路をリンクと呼ぶ。たとえば図1において、ピアBにとってのリンクとは、ピアA, C, およびDにそれぞれ確立された3つのリンクが相当

表1 情報の分散共有技術の分類
Table 1 Classification of decentralized architectures.

モデル	方式	例
フラットモデル	リンク方式 リンク+リレーションシップ方式 リレーションシップ方式 値域分割方式	Gnutella Winny, SIONet Freenet Chord, SOBA
クラスタモデル	FastTrack方式 Catch-up方式 Global Index方式 SRDI方式	KaZaA SIONet Skype JXTA

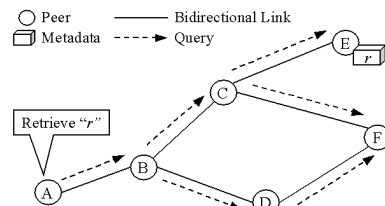


図1 リンク方式
Fig.1 Link architecture.

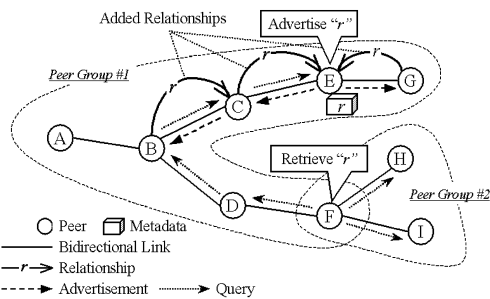


図2 リンク+リレーションシップ方式 (SIONet)

Fig. 2 Link and Relationship architecture (SIONet).

する。このときピア B は、ピア A, C, および D に対して情報を転送することができ、ピア E や F に対して直接情報を転送することはできない。クラスタモデルにおけるコネクと異なり、リンクを確立する2つのピアは互いに要求者、応答者の両者の役割を担わなければならない、両者は対等の関係を有する。

リンクの確立先となるピアの選択方法には、Gnutellaのように任意のピアが無作為に選択される方式のほか、SIONet や Winny のように、ピアの嗜好情報に基づいて特定のピアがリンクの確立先として選択される方式がある。後者の方式では、嗜好の近いピアどうしがリンクで結合され、ピアの集合がそれぞれ動的に形成される。この集合は『ピアグループ』と呼ばれ、これはSIONet のイベントブレス、および Winny のクラスタにそれぞれ相当する¹。図2において、ピアグループ #1 はたとえば「野球」に興味を持つピアの集合であり、ピアグループ #2 は「テニス」に興味を持つピアの集合である。また、ピア F は「野球」と「テニス」の両方に興味を持つピアであり、双方のピアグループに同時に所属している。このようなピアの存在により、ピアグループ間の連携が可能となる。たとえばピア F が、ピアグループ #1 のピア D から受け取った情報を、ピアグループ #2 のピア H に対して転送することにより、2つのピアグループは連携することになる。

一方、リレーションシップとは「任意のメタ情報(ファイル名、国籍、職業、嗜好など)に着目した際のピアの相関関係」を表し、その相関関係の強弱を『リ

レーションシップ強度』と呼ぶ。メタ情報を発見しようとするピアは、当該メタ情報に関して相関関係の強いピアが密集するエリア、すなわちリレーションシップ強度の強いエリアを探索する。たとえば Winny では、ピアは自身の「上流²」に対してリレーションシップを構築するため、最上流に近いピアほどリレーションシップ強度が強く、所望のメタ情報を発見できる可能性が高い。また SIONet では、ピアのメタ情報を当該ピアの「近傍³」にアドバタイズすることによって、イベントパスと呼ばれるリレーションシップを構築する。探索時には、イベントパス多重度⁴と呼ばれるリレーションシップ強度を用いて、問合せ先となるピアを選別することが可能である。

フラットモデルの分散共有技術は、リンクおよびリレーションシップの取扱いの違いに応じて、以下の4方式に分類することができる。

(1) リンク方式

ピアが自身の確立するリンクに基づいて情報をブロードキャスト転送する方式である。リンク方式ではリレーションシップを管理しないため、情報を探索する際はすべてのリンクを通じて問い合わせることによって目的のピアを見つけ出す必要がある⁵。Gnutella³⁾が本方式の代表的な例である。

(2) リンク+リレーションシップ方式

ピアが嗜好情報に基づいてリンクで結合し、その結果形成されたピアグループ内でリレーションシップを構築する方式である。本方式の代表的な例は、SIONet^{1),2)} および Winny⁶⁾ である。

(3) リレーションシップ方式

ピアがリンクを持たず、リレーションシップだけを動的に構築する方式である。リンクが存在しないため、P2P ネットワークに参加するためには自身のメタ情報をアドバタイズする必要がある。また、情報の転送先となるピアの選択もリレーションシップに基づいて行われる。本

¹ SIONet では、嗜好に合致するイベントブレスの生成・参加・退去によってピアグループが動的に形成される。一方 Winny では、ユーザが設定したクラスタワードと呼ばれる嗜好情報や、保有するメタ情報の一致度に応じてピア間の接続優先度を変化させ、優先度が低いピアとのリンクは切断し、より高いピアへとつなぎ替えていく。結果的に、同種のメタ情報を所望もしくは所有するピアどうしが安定的なリンクを確立し、緩やかなピアグループを形成する。

² Winny におけるピアの上流・下流とは、リンクを確立しようとする2者間の回線速度の違いに応じて決められる属性であり、原則として高速なピアが上流、低速なピアが下流になる。

³ SIONet におけるピアの近傍とは、当該ピアから一定のホップ数で到達可能なピアの集合である。

⁴ SIONet のイベントパス多重度は、任意のメタ情報に関するアドバタイズ回数、探索ヒット率、リプライ率、経過時間などに基いて算出され、当該メタ情報に関するピア間の結び付きの強さを表現する。

⁵ 実際の探索要求は、TTL によって転送先のピアが制限される。

方式の代表的な例は, Freenet^{4),5)}である.

(4) 領域分割方式

有限の空間を各ピアが分割管理し, メタ情報をその空間上に写像することによって, 当該メタ情報を管理するピアを一意に決定する方式である. ピアの管理領域に応じてリンクが規則的に構築されるため, 情報の転送経路もまた一意に決定される. 本方式の代表的な例として, Chord^{12),13)} および SOBA (ディレクトリサービス)¹⁴⁾ があげられる.

なお本論文では, 紙面の都合から領域分割方式は評価対象外とし, 別の機会に取り上げるものとする.

2.2 リンク方式

任意のピアが P2P ネットワークに参加し, 目的のピアを発見するまでの基本フローを以下に示す.

- (1) ピア (以下ピア A) は, Well-Known ピア に対してリンクを確立し, ピア情報の取得要求を行う.
- (2) Well-Known ピアは, 自身がリンクを確立している他のすべてのピアに対して, ピア情報の取得要求を転送する. 転送されたピアも同様に, 一定のホップ数を経由するまで要求を転送していく.
- (3) ピア情報の取得要求を受信したピア群は, それぞれ自身のリファレンスをピア A に返却する.
- (4) ピア A は, リファレンスを取得したピア群に対してそれぞれリンクを確立する.
- (5) 図 1 において, ピア E の保有するメタ情報 r は, 他のピアには公開されない. ここで, ピア A がメタ情報 r を検索キーワードとする探索を開始すると, 当該探索要求はまずピア A のリンクを通じてピア B に転送された後, さらにピア B のリンクを通じてピア C および D に転送される. 以下同様にして, 各ピアがすべてのリンクを通じて探索要求を転送することにより, 当該キーワードに合致するピア E を発見することができる.

リンク方式は, リレーションシップの構築 (アドバタイズ) がなくメンバシップ管理のコストも小さい反面,

探索要求がフラッディングされるためにディスクバリのコストが大きい. なお Gnutella については, ネットワークをよりスケーラブルにする試みとして, 低速なピアが通信の中継を担うことがないようにトポロジを制御したり, 一部にブローカを導入して探索を仲介させたりするなどの拡張が様々なベンダによってなされている^{16),17)}. その結果, サービスの運用上は支障がなくなりつつ見えるが, 本質的にはなおスケーラビリティの問題を内包していると考えられる.

2.3 リンク+リレーションシップ方式

任意のピアが P2P ネットワークに参加し, 目的のピアを発見するまでの基本フローを以下に示す.

- (1) ピア (以下ピア F) は, 探索によって発見したピア, もしくは Well-Known ピアのうち, 嗜好の近いピアを選択してリンクを確立する.
- (2) 自身のリファレンス (E) とメタ情報 r を公開しようとするピア E は, リンクを通じてアドバタイズを行うことにより, 動的にリレーションシップを構築する. 具体的な構築手順については後述する.
- (3) ここで, ピア F がメタ情報 r を検索キーワードとする探索を開始すると, 当該探索要求はメタ情報 r に関してリレーションシップ強度の強いピアに向かってリンクを通じて転送される. その結果, メタ情報 r を有するピア E を発見することができる.

SIONet におけるリレーションシップは, 公開元ピアの「近傍」に構築される. たとえば図 2 において, 自身のリファレンス (E) とメタ情報 r を新たに公開しようとするピア E は, 自身がリンクを確立するすべてのピア C および G に対して公開要求 (TTL = 2) を転送する. これを契機として, ピア C および G は公開元ピア E に対してメタ情報 r に関するリレーションシップを確立する. さらにピア C は, 自身がリンクを確立するピア B に対して公開要求を転送する. このときピア B は, 公開元のピア E ではなく, 公開要求の転送元であるピア C に対してリレーションシップを確立することに注意を要する. この方法により, ピア E の存在は隣接するピア以外には隠蔽され, 詐称や暗号化によらない, 本質的な匿名性を保証している. こ

Well-Known ピアとは, ホームページの掲載情報やあらかじめプログラム内に埋め込まれた情報などによって, あらかじめリファレンスが広く知られたピアである. フラットモデルにおける Well-Known ピアは, クライアント・サーバモデルやクラスタモデルの場合と異なり, 一般のピアとの機能的な相違はなく, また必ずしも安定的な運用 (高可用性) を前提としない. Freenet や Winny では, Well-Known ピアのリストが Web サーバなどの手段を用いて動的に更新および配布される.

SIONet では, エントランス探索によって初期接続先となるピアのリファレンスを取得可能である. ユーザは発見したピアが所属するイベントスペース情報に基づき, 嗜好の合致するピアを選択してリンクを確立する. これに対し Winny では, Well-Known ピアのみサポートしている. あらかじめユーザが設定したクラスタワードなどの嗜好情報に基づき, 接続優先度の高いピアとリンクを確立する.

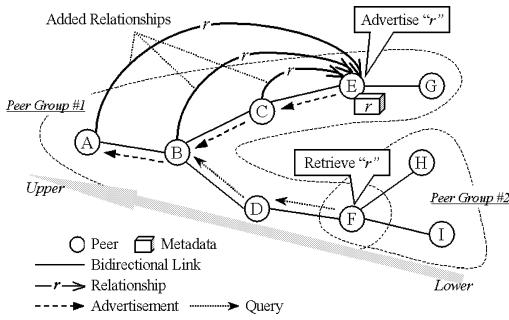


図3 リンク+リレーションシップ方式 (Winny)
Fig. 3 Link and Relationship architecture (Winny).

れに対し Winny におけるリレーションシップは、公開元ピアの「上流」に構築される。たとえば図3において、自身のリファレンス(E)とメタ情報 r を新たに公開しようとするピア E は、自身がリンクを確立するピアのうち、上流のピア C に対して公開要求を転送する。続いてピア C は、自身の上流のピア B に対して公開要求を転送する。以下同様にして、公開要求は最上流のピア A まで波及する。このとき、公開要求を受け取ったすべてのピア C, B, および A は、公開元ピア E に対してメタ情報 r に関するリレーションシップを確立する。

リンク+リレーションシップ方式では、リレーションシップ強度の強いエリアに対してディスカバリを行う。たとえば SIONet では、自身がリンクを確立するピアのうち、リレーションシップ強度(イベントパス多重度)が閾値以上のピアを選択して探索要求を転送することができる。いずれのピアもリレーションシップ強度が0の場合は、リンク方式同様、すべてのピアに対して転送を行う。図2に示すように、メタ情報 r を検索キーワードとする探索を開始したピア F は、自身がリンクを確立するピア D, H, および I のいずれに対してもリレーションシップを有さない(リレーションシップ強度が0である)ため、すべてのピアに対して探索要求を転送する。その後、ピア D を経て探索要求を受信したピア B は、リレーションシップを有するピア C だけに転送を行い、リレーションシップ強度が0のピア A には転送しないことに注意を要する。さらに、ピア C がピア E に対して探索要求を転送し、目的とするメタ情報 r を発見する。一方、Winny の

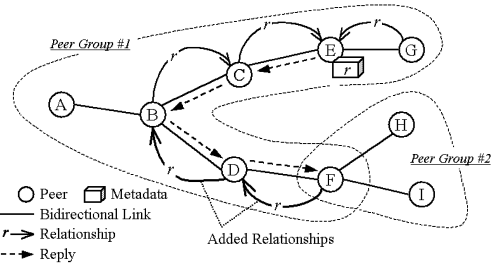


図4 SIONet におけるリプライ
Fig. 4 Reply on SIONet.

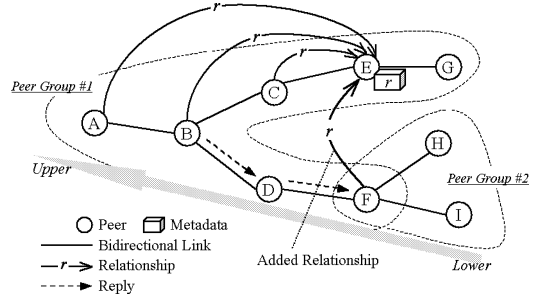


図5 Winny におけるリプライ
Fig. 5 Reply on Winny.

例においてピア F から送出された探索要求は、リレーションシップ強度の強いエリア、すなわち「上流」のピアへと転送されていく。その結果、探索元ピア F と公開元ピア E との共通の上流ピアであるピア B において、メタ情報 r を有するピアのリファレンス(E)を発見することができる。

リンク+リレーションシップ方式におけるリレーションシップの増設契機は、すでに述べたアダプタイズのほか、リプライがある。SIONet では図4に示すように、ピア E が自身のリファレンスを探索結果とするリプライを返却すると、当該リプライは探索要求を中継したピア C, B, および D を経由して探索元のピア F まで転送される。このとき、アダプタイズの場合と同様に、ピア D および F は公開元のピア E ではなく、それぞれリプライ転送元であるピア B および D に対してリレーションシップを確立する。これに対して Winny では、図5に示すように、リプライを受信した探索元のピア F が公開元のピア E に対して直接、リレーションシップを確立する。いずれの方法においても、リプライを契機としてリレーシ

実際には、匿名性の保証を目的として、ある確率の下で公開元ピアは詐称される。たとえば、ピア A は公開元のピア E ではなく、公開要求を中継したピア C に対してリレーションシップを確立する場合がある。その場合、ピア C は配信(ダウンロード)の中継を担う。

探索時に上流のピアが複数存在する場合は、より接続時間の長い、安定的なリンクを選択して探索要求を転送する。Winny では通常、複数の探索結果をまとめて返却するため、ピア B がリプライを返すとは限らない。しかし、ここでは SIONet との概念的な比較を行うためにモデルを単純化している。

ンシップを増設することにより、需要の高いメタ情報ほどリレーションシップが強化され、より発見されやすくなる効果がある。

リンク+リレーションシップ方式はリンク方式に比べ、アダプタイズのコストが加わる反面、ピアグループとリレーションシップ強度によって探索空間を限定できるため、ディスカバリのコストは小さくなる。しかしながら、ピアの嗜好情報を利用するためには、SIONNetにおけるイベントスペースの選択・生成、あるいはWinnyにおけるクラスタワードの設定、といったユーザの恣意的行為を介在させる必要がある。この問題を解決する方法として、ユーザの嗜好抽出や行動履歴に基づくポリシ制御技術 (COMNet)^{1),2)} が提案されている。

2.4 リレーションシップ方式

任意のピアがP2Pネットワークに参加し、目的のピアを発見するまでの基本フローを以下に示す。

- (1) ピア (以下ピア A) は、Well-Known ピアに対し、自身のリファレンスとメタ情報 r_0 (ピア ID など) をアダプタイズする。
- (2) Well-Known ピアは、自身がリレーションシップを確立しているピアの中から任意の 1 ピアを選択し、参加要求を転送する。転送されたピアも同様に、一定のホップ数を経由するまで要求を転送していく。
- (3) 参加要求を受け取ったピア群は、それぞれピア A に対してメタ情報 r_0 に関するリレーションシップを確立する。
- (4) 自身のリファレンス (F) とメタ情報 r を公開しようとするピア F は、既存のリレーションシップに基づいてアダプタイズを行い、新たなリレーションシップを増設する。具体的な手順については後述する。
- (5) ここで、ピア A がメタ情報 r を検索キーワードとする探索を開始すると、当該探索要求はメタ情報 r に関してリレーションシップ強度の強いピア間をリレーションシップに基づいて転送される。その結果、メタ情報 r を有するピアのリファレンス (F) を発見することができる。

リンク+リレーションシップ方式では、情報の転送先となる (固定的な) リンク、およびピアの相関関係を表すリレーションシップを個別に扱っていたのに対

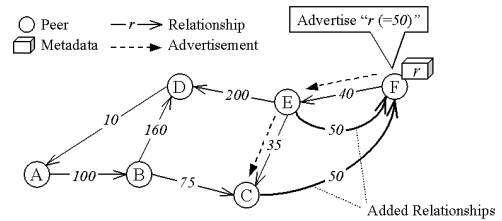


図 6 リレーションシップ方式 (アダプタイズ)
Fig. 6 Relationship architecture (Advertisement).

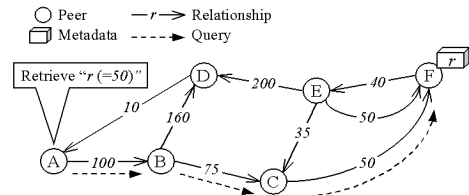


図 7 リレーションシップ方式 (ディスカバリ)
Fig. 7 Relationship architecture (Discovery).

し、リレーションシップ方式ではこれらを一体的に扱い、動的に増設されるリレーションシップを情報の転送先とすることで、ピアの相関関係に応じたトポロジの局所化を可能としている。

Freenetにおけるリレーションシップ強度は、リレーションシップに関連付けられたメタ情報のハッシュ値と、着目するメタ情報のハッシュ値との「値の近さ」である。図 6 に示すように、ピア F から自身のリファレンス (F) とメタ情報 r の公開要求を受けたピア E は、自身が確立する 2 つのリレーションシップのうち、メタ情報 r のハッシュ値「50」により近い、「35」に関するリレーションシップを選択してピア C に公開要求を転送する。一方、図 7 に示すように、ピア A からメタ情報 r を検索キーワードとする探索要求を受けたピア B は、メタ情報 r のハッシュ値「50」により近い、「75」に関するリレーションシップを通じてピア C に探索要求を転送する。その結果、メタ情報 r を有するピアのリファレンス (F) を発見することができる。

リレーションシップの増設契機は、リンク+リレーションシップ方式と同様、アダプタイズおよびプライである。図 6 において、ピア F による公開要求を転送されたピア E および C は、それぞれ公開元のピア F に対してメタ情報 r のハッシュ値「50」に関するリレーションシップを追加する。また図 7 に示す探

Freenet では、複数のピアが生成したシードを組み合わせで計算された擬似乱数をアダプタイズする。ピアが相互に改竄をチェックすることにより、悪意あるユーザによる作動的なリレーションシップの構築を防止している。

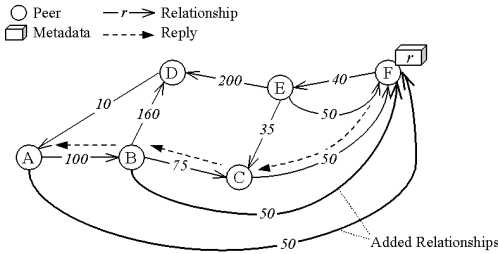


図 8 リレーションシップ方式 (リプライ)
Fig. 8 Relationship architecture (Reply).

索の結果、ピア F がリプライを返却すると、図 8 に示すように、当該リプライは探索要求を中継したピア C および B を経由して探索元のピア A まで転送される。これを契機として、探索元のピア A およびリプライを中継したピア B は、公開元のピア F に対して「50」に関するリレーションシップを確立する。

このように、Freenet ではメタ情報のハッシュ値に最も近い値に関するリレーションシップを選択して情報を転送することにより、ある範囲内のハッシュ値を有するメタ情報の公開・探索要求が特定のピアに集まりやすくなる効果がある。さらに、時間の経過とともに各ピアが公開・探索を繰り返すに従い、ハッシュ値の近接するメタ情報を所望もしくは所有するピア間にショートカット・パスが生成され、P2P ネットワークが動的に局所化されていく。

リレーションシップ方式は、アダバタイズとディスカバリーに係るコストの一部を、メンバシップ管理 (局所化) のコストに転嫁したものであるといえる。そのため、P2P ネットワークが十分に局所化されていない状況 (初期状態、ピアが頻繁に出入りする状態など) では探索が振動し、ターンアラウンド時間が増大する問題がある。

なお Freenet では、具体的にはファイル識別子のハッシュ文字列 (ファイルキー) に関するリレーションシップを構築し、「ファイルキーの辞書的な近さ」を情報転送先の選択基準としている。ファイルキーの値はファイルの内容とは無関係であるため、Freenet における局所化は必ずしもユーザの嗜好に基づくものではない。しかしながら、メタ情報が持つ「意味」をリレーションシップに反映することができれば、本方式

は COMNet^{1),2)} などが目指すポリシ制御を実現するための 1 つのアプローチになりうる。

2.5 性能評価および考察

本節では、これまでに述べた既存技術の性能について、定量的な評価を与える。以下に性能評価モデルおよび性能諸元を定義する。

- **ピア数 n** : ネットワークを構成するピアの総数。各ピアは、自身のメタ情報として「ピア ID」を公開する。ピア ID の値は、ピアの生起順によるシーケンス番号 (0, 1, 2, ...) とする。
- **イベント**: 各ピアが独立に発行する情報の公開要求および探索要求 (有効イベントと呼ぶ) のほか、これらに起因する転送要求 (制御イベントと呼ぶ) の総称。ネットワーク・トラフィックに相当する。
- **イベント生起率 λ** : 各ピアにおいて、単位時間あたりに生起する有効イベントの数。制御イベントの生起は、これに含まない。特に公開要求の生起率を公開率、探索要求の生起率を探索率と呼ぶ。

$$\lambda = \text{公開率} + \text{探索率}$$

- **イベント処理時間 h** : 各ピアにおいて、1 つのイベントを処理するのに要する時間。ここでは、すべてのイベントについて $h = 1$ (単位時間) とする。
- **イベント終了率 μ** : 各ピアにおいて、単位時間あたりに処理可能なイベントの数。ここでは、 $\mu = h^{-1} = 1$ である。
- **イベント比率 k** : 生起したイベントが公開要求であるか探索要求であるかを示す比率。

$$k = \frac{\text{公開率}}{\lambda} = \frac{\text{公開率}}{\text{公開率} + \text{探索率}}$$

- **利用率 ρ** : 1 つのピアにおいて、単位時間あたりに処理されたイベントの数。
- **スループット λ_s** : ネットワーク全体で、単位時間あたりに処理された有効イベントの数。制御イベントはオーバーヘッドであるため、スループットには含まない。
- **待合せ時間 $E[W]$** : 各ピアにおける、イベントの平均処理待ち時間。ピアにおけるイベントの同時処理数はたかだか 1 であり、イベント処理中に他のイベントが到着した場合、当該イベントは先着のイベント処理が終了するまで待ち合わせる。
- **ターンアラウンド時間**: ピアが探索要求を発行してから、リプライを受信するまでの時間。
- **探索ヒット率 θ** : 各ピアが発行した探索要求のうち、目的とするメタ情報を発見できたものの割合。
- **ピアの変動率 q** : 単位時間あたりに退去するピア

Freenet の場合、アダバタイズの際にピア C にファイルの複製が置かれるため、探索要求はピア F まで波及しない。しかし、本論文はファイル交換システムの評価を目的としたものではなく、P2P ネットワークにおけるインデックスサーチ (ユニークなピアの特定) を題材として各方式間の比較・評価を行うことが主眼であるため、ここでは意図的に複製の効果を除去している。

の数．なお任意のピアが退去するとすぐに別のピアが新たに参加し，ネットワーク全体のピア数 n はつねに一定に保たれるものとする．

- リンク数 m : 各ピアが確立するリンクの数．
- TTL : イベント転送の最大ホップ数．ここでは，各方式において探索ヒット率 50% を満足する最小の TTL を使用する．

ピアの利用率 ρ は，イベント生起率 λ および終了率 μ を用いて，以下の式で表される．

$$\rho = \frac{\lambda_s + \alpha}{n\mu} = \frac{n\lambda + \alpha}{n\mu} \quad (1)$$

ここで α は，スループットの増加に寄与しない制御イベントの数であり，オーバーヘッドの大きさを表す．(仮想的な)理想性能は $\alpha = 0$ であり，このとき $\rho = \lambda\mu^{-1}$ となる．またスループットは $\lambda_s = n\lambda$ であるから， ρ と λ_s との関係を示すグラフは，傾き $n\mu$ の直線になる．一方，現実のネットワーク・システムではオーバーヘッド α が加わる分，同一のスループットにおける利用率は，理想性能より大きくなる．いい換えれば，同一の利用率 $\rho = \rho_0$ においてスループット λ_s が大きいシステムほど理想性能に近く，よりスケラビリティが高い方式である．

また，P2P ネットワーキングにおける主なコスト要因には，以下のものがあげられる．

- イベント処理時間 h
- 待合せ時間 $E[W]$
- 内部キャッシュの探索時間

本論文では，内部キャッシュの探索時間は他の要因に比べて十分小さいと見なし，考慮しないものとする．また，各ピアの内部キャッシュサイズは無制限とし，キャッシュあふれによる性能劣化は考えない．

本節では，前節までに述べたリンク方式，リンク+リレーションシップ方式，およびリレーションシップ方式について，それぞれ P2P ネットワーク・シミュレータを作成し，探索ヒット率 50% を満足する条件下でのスループットおよびターンアラウンド時間について比較評価を行った．また比較の参考とするため，クライアント・サーバ (C/S) 方式についても同様の測定を行った．

理想性能とは，各ピアが公開・探索要求を他のピアへと転送することなく，自身が保持するキャッシュ情報によって目的の情報を発見することができる場合の性能である．

探索ヒット率を他の値に固定した場合でも，以下の評価結果には影響しないことを確認済みである．ここでは，シミュレーションに要する時間，および結果の明瞭さの観点から 50% を選択した．なお，C/S モデルの性能は探索ヒット率にかかわらず一定である．

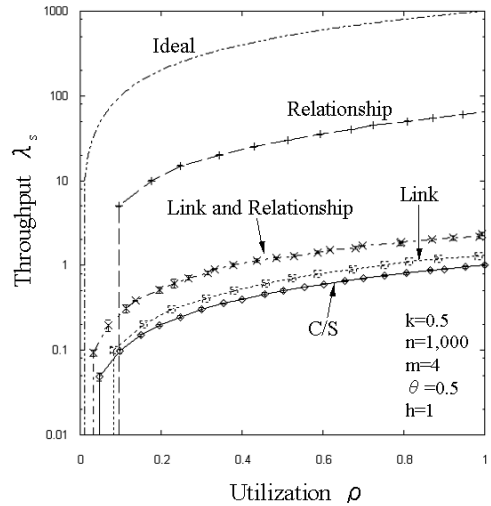


図 9 利用率対スループット ($k = 0.5$)
Fig. 9 Utilization vs. Throughput ($k = 0.5$).

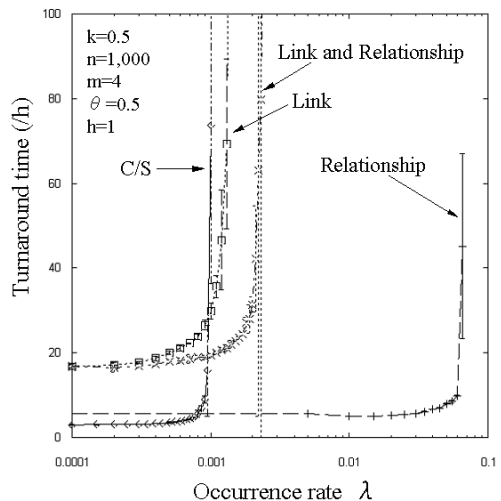
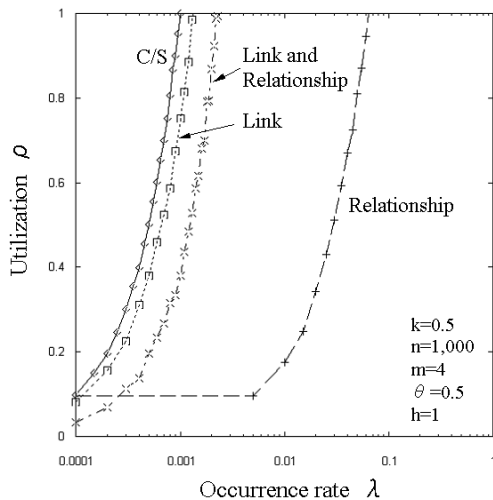
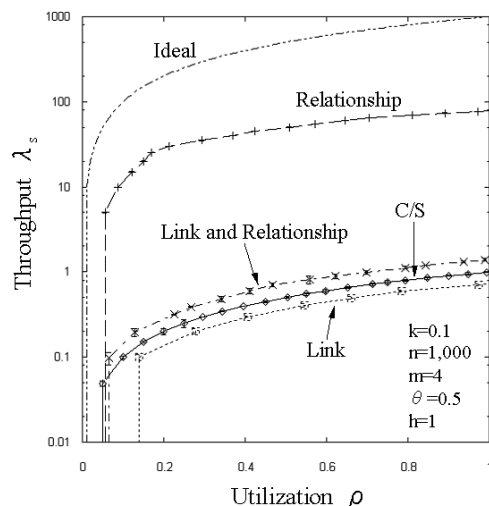


図 10 生起率対ターンアラウンド時間 ($k = 0.5$)
Fig. 10 Occurrence rate vs. Turnaround time ($k = 0.5$).

$n = 1,000, m = 4, q = 0, k = 0.5$ (公開と探索の占める割合が同じ) とした場合の測定結果を図 9, 図 10, 図 11 に示す．なお TTL の値は，リンク方式では 16，リンク+リレーションシップ方式では 5，リレーションシップ方式では 20 を用いた．この設定により，探索ヒット率 50%，すなわち単位時間 h あたりに生起する $(1 - k)\lambda$ 個の探索要求に対し， $((1 - k)\lambda)/2$ 個以上が目的のメタ情報を発見する状況における，各方式のスループットおよびターンアラ

Gnutella では TTL = 7 としているが，これは探索の対象がファイルである (いたるところに複製が存在する) ことを前提としている．

図 11 生起率対利用率 ($k = 0.5$)Fig. 11 Occurrence rate vs. Utilization ($k = 0.5$).図 12 利用率対スループット ($k = 0.1$)Fig. 12 Utilization vs. Throughput ($k = 0.1$).

ウンド時間を比較することができる。また、同条件で $k = 0.1$ (公開の占める割合が 1 割, 探索の占める割合が 9 割) とした場合のスループットの測定結果を図 12 に示す (ターンアラウンド時間の測定結果は, $k = 0.5$ の場合に比べ特段の変化がないため割愛する)。なお本論文中のグラフは, それぞれ同一の条件による実験を 3 回ずつ試行した結果の平均値, および 95%信頼区間を示したものである。

リンク方式は分散共有に係る大部分のコストが探索

要求の転送処理に起因する特徴を持つため, k が小さくなる (探索の占める割合が増加する) ほどオーバーヘッドが増大し, スループットが低下する。図 12 に示すように, $k = 0.1$ におけるリンク方式のスループットは, ループの多発により C/S モデルをも下回る性能であることが分かる。一方, k が 1 に近い (公開の占める割合が探索を大きく上回る) ケースではリンク方式のスループットが向上すると推測できるが, このような条件は P2P ネットワークの適用モデルとして現実的とはいえない。

一方, リレーションシップ方式は, 他の方式に比べて同一の利用率におけるスループットが顕著に高い。え, ターンアラウンド時間も抑制され, 総合的に性能が優れていることが分かる。その理由として, 以下の 2 点が考えられる。1 点目は, イベントの波及方法の違いである。リンク方式やリンク+リレーションシップ方式では通常, 複数のピアに対して同時に情報を転送するため, イベントが面状に波及する傾向を持つ。これに対しリレーションシップ方式では, 最適なりレーションシップを有するピアに対して選択的に情報を転送するため, イベントが線状に波及する。前者の方式ではイベントの複製・転送回数, すなわち制御イベント数が指数関数的に増加するのに対し, 後者の方式では線形に増加する。2 点目の理由は, 動的なネットワーク局所化の有無である。リレーションシップ方式では, アドバタイズおよびリプライを契機として増設されるリレーションシップを情報の転送先として使用するため, 時間の経過とともに任意のピア間のイベント転送距離 (最小ホップ数) は短縮される傾向がある。これに対し他の方式では情報の転送先となるリンクの増設をしないため, ピア間の距離が変化することはない。これらの性質から, リレーションシップ方式では他の方式に比べてオーバーヘッドが小さく抑えられている。

以上のシミュレーション実験では, 既存の P2P アプリケーションが使用する一般的な値として, リンク方式およびリンク+リレーションシップ方式のリンク数を $m = 4$ とした。この値をある程度変動させても, 性能に与える影響は軽微である。ただし m の値を極端に大きくする ($n - 1$ に近付ける) と, リンク方式は (プル型の) メッシュモデル¹⁵⁾, リンク+リレーションシップ方式は C/S モデルの性能に近付くと考えられる。両方式について, それぞれ $m = 500$, TTL = 1 とした場合のスループットの測定結果を図 13 に示す。リンク方式のスループットが上昇する一方, リンク+リレーションシップ方式のスループットは低下し, $m = 4$ のケースとは優劣が逆転してい

2.5 節では安定状態に収束後の性能値をサンプリングしているため, 誤差範囲はおおむね 1%程度に収まっており, ほとんどの誤差棒は視認できないほど小さい。

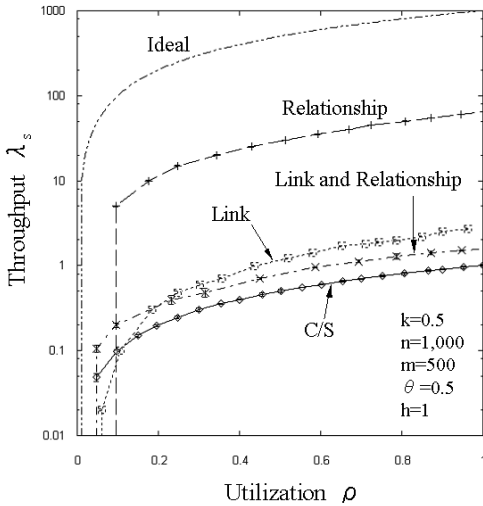


図 13 利用率対スループット ($m = 500$)
Fig. 13 Utilization vs. Throughput ($m = 500$).

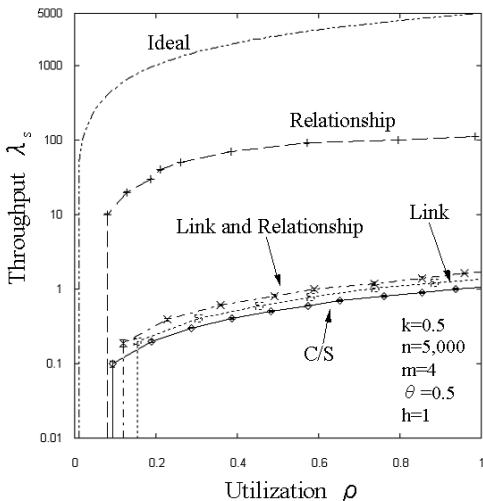


図 14 利用率対スループット ($n = 5,000$)
Fig. 14 Utilization vs. Throughput ($n = 5,000$).

る様子が見られる。なお、 $k = k_0$ における (プル型の) メッシュモデルのスループットは、 $k = 1 - k_0$ における Catch-up 方式のスループットに等しく、その最大値は $1/((1 - k_0)h)$ である。

また、ピア数 n を変動させた場合は、個々の方式について性能値の上下はあるものの、方式間の相対的な関係には変化が見られなかった。一例として、 $n = 5,000$ とした場合のスループットの測定結果を図 14 に示す。

3. 提案方式

2.5 節の評価結果から分かるように、リレーションシップ方式 (Freenet) はいずれの条件下でも安定的

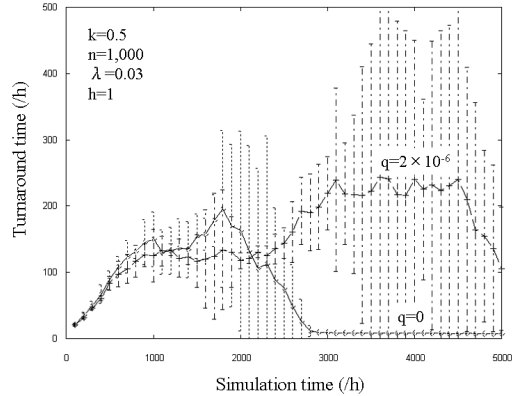


図 15 Freenet におけるターンアラウンド時間の推移
Fig. 15 Time evolution of the turnaround time on Freenet.

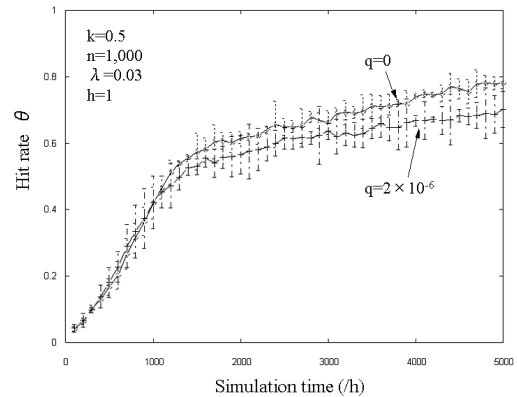


図 16 Freenet における探索ヒット率の推移
Fig. 16 Time evolution of the hit rate on Freenet.

に高いスループットを示しており、スケラビリティが高い方式であるといえる。しかしながらこの評価結果は、 $q = 0$ (ピアの参加・退去が起こらない) とする仮定の下、安定状態に収束したネットワークを前提としていることに注意を要する。現実の P2P ネットワークでは、ピアは過渡的・短寿命であり、それらが頻繁に参加または退去する状態を考える必要がある。

そこで、ピアの変動率 q を変化させ、Freenet の挙動に与える影響を調べた。結果を図 15 および図 16 に示す。図 15 から分かるように、 $q = 0$ の場合、ターンアラウンド時間が 10 程度に収束し、ネットワークは安定状態となる。2.5 節において示したデータは、このような安定状態のネットワークにおける測定値 (収束解) を示したものである。ところが、 $q = 2 \times 10^{-6}$ を超えるとターンアラウンド時間が振動して収束しなくなるほか、図 16 に示すように探索ヒット率が低下するなど、顕著な性能の劣化が見られる。なお $q = 2 \times 10^{-6}$

とは、各ピアが $5 \times 10^5 h$ の間に 1 回の頻度で他のピアと入れ替わることを表している。イベント処理時間 h が実時間の 100 ミリ秒に相当するものと仮定すると、1 ピアあたりの平均的な滞在時間はおよそ 14 時間である。この値は、ピアの短寿命を前提とする P2P ネットワークの運用条件としては過剰に安定的なものであり、現実にはより頻繁にピアが入り出る状況を想定しなければならない。安定状態のネットワークにおいて高い性能を示しながら、Freenet が実用上直面している課題の 1 つは、この収束の難しさにあると考えられる。

そこで本論文では、ピアの参加・退去を考慮した局面においても安定状態に収束させるための改良を Freenet に加え、実用的な条件下でも高い性能を維持可能であることを示す。

初めに、実験的に観測された Freenet における「安定状態」とは、ネットワークのどのような状態を反映したものであるかを考える。探索時にターンアラウンド時間を増大させる最大の要因は、TTL の制約によって、目的とするピアに到達不可能なケースである。このケースでは、不発見が判明するまで探索要求を転送し続けるため、探索ホップ数の増加はもとより、中継ピアにおける待合せ時間の蓄積によって、ターンアラウンド時間が相乗的に増大する。ピアの変動率 q にかかわらず、多くの探索要求が不発見で終わるネットワークの初期状態（図 15、図 16 における 0~約 2,000h の時間帯）において、ターンアラウンド時間が極端に増加するのはそのためである。その後、時間の経過とともにアダプタイズやリプライが繰り返され、ピア間にリレーションシップが増設されていくと、初期状態では到達不可能であったピアに対しても、やがて到達可能な経路が確立される。このようなショートカット・パスが多数追加されることによって、目的のピアを早期に発見できる可能性が高まり、ターンアラウンド時間の増大は抑制され収束へと至る。ところが、ピアの退去によってこれらのパスが頻繁に分断されると、ネットワークの部分的な初期化が繰り返され、ターンアラウンド時間は収束しなくなる。以上のことから、Freenet における（究極的な）安定状態とは、「すべてのピアから他のすべてのピアに対し、TTL の制約下で到達可能な経路が確立された状態」であると考えられる。

以下に例を示す。いま $TTL = 3$ とするとき、図 17

ピアが退去すると、当該ピアにおいて処理待機中のイベントがロストするため、理想的な安定状態においても探索ヒット率が 100% になるとは限らない。

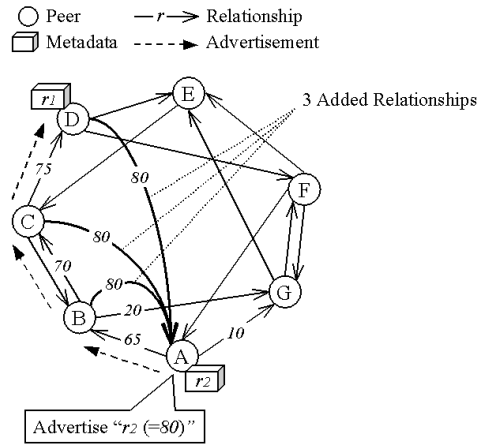


図 17 Freenet における公開と収束
Fig. 17 Advertise and converge on Freenet.

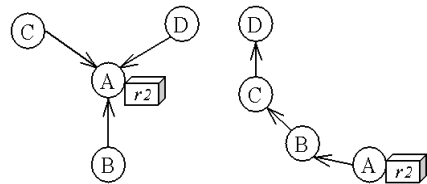


図 18 Freenet におけるリレーションシップ（抜粋）
Fig. 18 Relationships on Freenet.

に示すネットワークは、ピア E からピア A に到達可能でないため安定状態ではない。ここでピア A が自身のリファレンス (A) とメタ情報 r_2 (ハッシュ値 80) を新たに公開すると、2.4 節で述べた手順により、ピア B, C, および D からピア A に対して「80」に関するリレーションシップが追加される。このとき公開先の選定に使用されたリレーションシップの抜粋を図 18 右、および公開の結果追加されたリレーションシップの抜粋を図 18 左に示す。その結果、ピア E からピア A に至るパス (E→C→A, E→C→B→A, または E→C→D→A) が確保され、ネットワークは安定状態に収束する。

いま図 19 に示すように、安定状態のネットワークにおいてピア B が退去した場合を考える。このときピア A からピア D への到達可能性が失われるため、ネットワークは再び非収束状態となる。ここでピア F が、メタ情報 r_1 (ハッシュ値 60) を検索キーワードとする探索を開始すると、当該探索要求はまずピア A に転送される。次にピア A は、要求を本来転送すべきピア B が退去したため、「60」とはかけ離れた「10」に関するリレーションシップを通じてピア G に要求を転送する。このように過去の学習結果とは反する経路を選択した結果、探索は F→A→G→E→C→D と

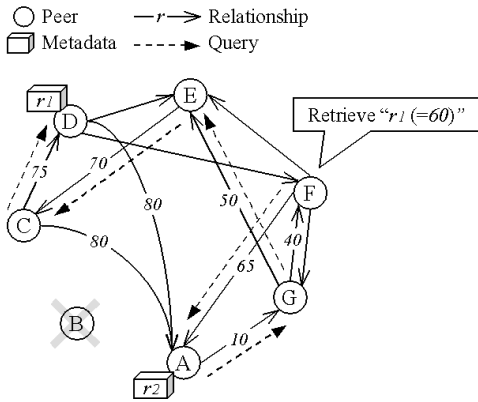


図 19 Freenet における退去と発散
Fig. 19 Depart and diverge on Freenet.

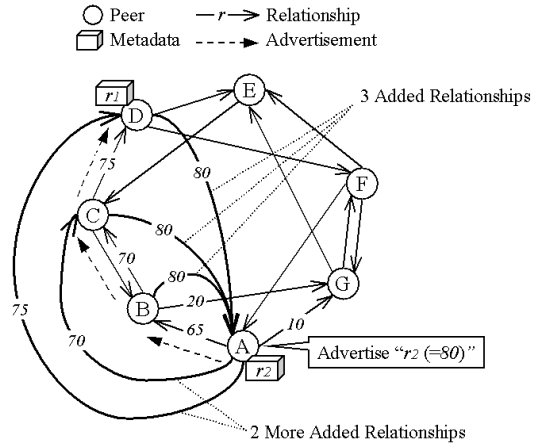


図 20 リレーションシップの双方向制御 (アドバタイズ)
Fig. 20 Proposal architecture (Advertisement).

振動し、TTL = 3 の制約下では発見に至ることができない。

この問題の原因としては、あるメタ情報に着目した際のリレーションシップの非対称性が考えられる。図 18 で示したように、公開先ピア (B, C, D) から公開元ピア (A) に対するリレーションシップは、公開元ピアを中心とするスター型トポロジを構成するのに対し、公開元ピアから公開先ピアに対するリレーションシップは線形トポロジである。前者のリレーションシップは公開元ピアが退去しない限り維持される一方、後者のリレーションシップは両端を除く任意のピアが退去することによって分断を生じ、収束を阻害する要因になる。たとえばピア B が退去すると、ピア A はピア C および D との間接的なリレーションシップを有しているにもかかわらず、それらのピアに情報を転送することが困難になる。

そこで上記の問題を解決する方策として、リレーションシップの双方向制御を提案する。たとえば図 20 において、ピア A が自身のリファレンス (A) とメタ情報 r_2 (ハッシュ値 80) を新たに公開したとき、Freenet ではピア B, C, および D からそれぞれピア A に対するリレーションシップを追加するだけであるが、それに加え、ピア A からピア C および D に対するリレーションシップを追加し、双方向のリレーションシップを構築する。このときピア A がピア C および D に対して確立するリレーションシップに関連付けられる値は、それぞれ当該ピアに対して公開要求を転送する際に使用されたリレーションシップに関連付けられた値、すなわち「70」および「75」とする。この方法により、ピア間のリレーションシップは図 21 に示すような双方向のスター型トポロジを構成するため、Freenet に比べより独立性の高い (第三者の障害の影

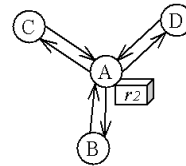


図 21 双方向制御適用後のリレーションシップ (抜粋)
Fig. 21 Relationships on the proposal architecture.

響を受けにくい) 関係性を築くことが可能になる。

4. 提案方式に関する評価および考察

2.5 節で使用したリレーションシップ方式 (Freenet) のシミュレータをベースとして、前章で提案したリレーションシップの双方向制御機構を追加し、ネットワークの挙動がどのように変化するかを調べた。その結果を図 22, 図 23, 図 24, 図 25 に示す。ピアの変動率を $q = 2 \times 10^{-5}$ としたとき、Freenet は測定時間内を通じて振動状態であるのに対し、提案方式では時間の経過とともにターンアラウンド時間が安定的に推移し、収束している様子が分かる。また探索ヒット率についても、提案方式において顕著な改善が見られる。なお、 $q = 2 \times 10^{-5}$ とは、イベント処理時間 h が 100 ミリ秒相当と仮定した場合、1 ピアあたり平均 1 時間程度の滞在を意味しており、P2P ネットワークにおけるピアの振舞いとして現実的な値といえる。ピアの変動率 q の値をさらに大きくすると提案方式でも発散を示すケースが見られたが、これは公開元のピア自身 (たとえば図 20 のピア A) が退去するようなケースが多発するためと考えられる。

また、本提案方式はアドバタイズ処理に対する改良であるため、 $k = 0.33$ (公開率と探索率の比が 1 : 2)

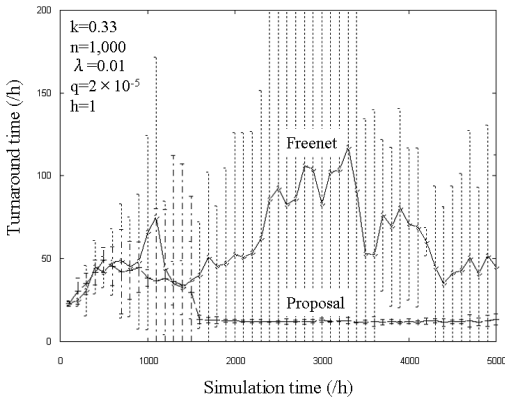


図 22 ターンアラウンド時間の比較 ($k = 0.33$)

Fig. 22 Comparison on the turnaround time ($k = 0.33$).

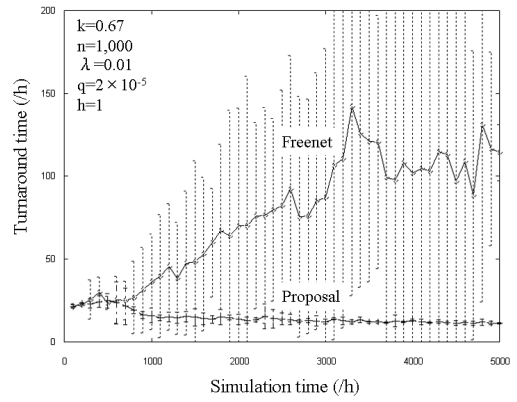


図 24 ターンアラウンド時間の比較 ($k = 0.67$)

Fig. 24 Comparison on the turnaround time ($k = 0.67$).

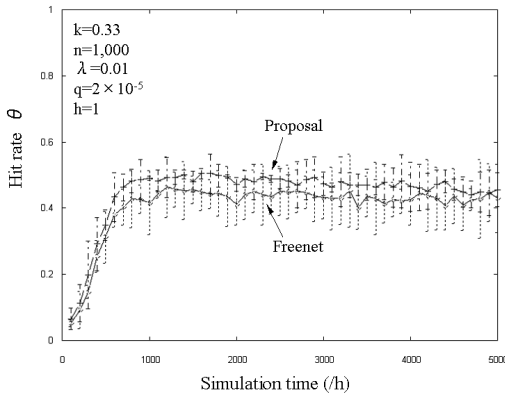


図 23 探索ヒット率の比較 ($k = 0.33$)

Fig. 23 Comparison on the hit rate ($k = 0.33$).

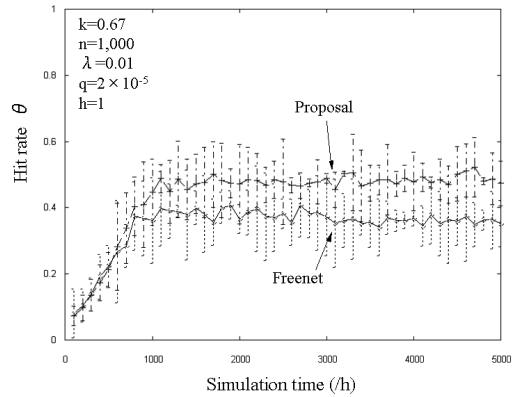


図 25 探索ヒット率の比較 ($k = 0.67$)

Fig. 25 Comparison on the hit rate ($k = 0.67$).

の場合に比べ、より公開の占める割合が高い $k = 0.67$ (公開率と探索率の比が 2 : 1) の場合の方がより効果的であることも分かる。

これらの実験結果は、提案方式における以下のような効果を実証するものといえる。いま図 20 のネットワークからピア B が退去した状態を図 26 に示す。ここで、ピア F がメタ情報 r_1 (ハッシュ値 60) を検索キーワードとする探索を開始した場合を考える。当該探索要求を転送されたピア A は、要求を本来転送すべきピア B が退去したため、「60」に最も近い「70」に関するリレーションシップを通じてピア C に要求を転送する。さらにピア C はピア D に対して要求を転送し、メタ情報 r_1 を発見することができる。このように本方式では、ピア A がメタ情報 r_2 の公開を契機として学習した経路、すなわちピア C および D に対して確立したリレーションシップが、本来ピア B へ転送されるべき要求のバックアップ経路として機能することが分かる。この方法により、ピアの退去にとも

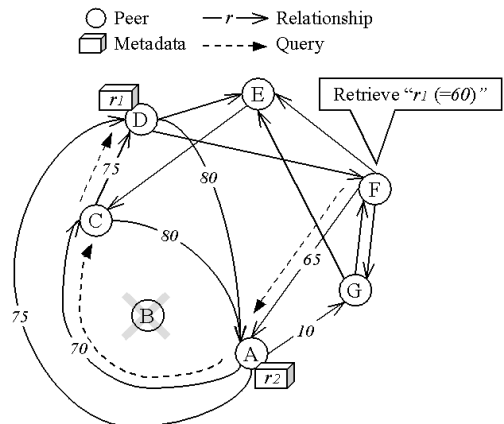


図 26 リレーションシップの双方向制御 (ディスカバリ)

Fig. 26 Proposal architecture (Discovery).

う学習内容の喪失を回避し、ピア間の到達可能性、すなわちネットワークの収束性を堅固に維持することが可能になる。

なお提案方式では, Freenet に比較して追加的なトラフィックは発生しない. 具体的には, Freenet におけるファイル挿入要求 (Request.Insert), およびその応答 (Reply.Insert) に対するデータの追加だけで実現可能である. したがって, Freenet が収束する条件の下では, 提案方式と Freenet が示すスループットの値に有意な差はなく, 両方式のスケラビリティは同等である. しかしながら, Freenet に比べ収束条件が大きく緩和される点において, 提案方式は適用領域が広く, より実用性が高い方式であるといえる.

5. む す び

ブローカレス型ネットワークの実現技術において, 探索・グルーピングレイヤを対象とする技術は「フラットモデル(すべてのピアが対等な立場で分散・協調するモデル)」と「クラスタモデル(一部のピアに特別な役割を持たせたモデル)」に分類される. 本論文はフラットモデルに焦点を絞り, 従来技術の体系的分類と方式間比較, およびシミュレーション実験による性能評価を行った. さらにその結果から, 従来技術の中では Freenet がスケラビリティに秀でている反面, ピアの変動(参加・退去)を考慮した実用的な条件下では, ネットワークを安定状態に収束させることが困難である問題を明らかにした. そこでその解決策として, Freenet にリレーションシップの双方向制御機構を追加した方式を考案し, 実用的な条件下においてもネットワークの収束性が顕著に改善され, 高い性能が安定的に得られることを示した.

本論文では紙面と時間の制約から割愛した, 値域分割方式についても分析と性能評価を行い, 機会を改めて報告する予定である. 本論文中で取り上げたリンク方式, リンク+リレーションシップ方式, およびリレーションシップ方式は, いずれもピアが相互にリンクまたはリレーションシップを自由に構築した結果として, 直接認知しないピアどうしも緩やかな関係性を築くものであった. これらの方式は, 全世界の人々と直接友達でなくとも, 友達の友達を手繰っていくことでたいのひとと知り合うことができるとする Small-World 理論¹⁸⁾ を具現化したものといえる. これに対して値域分割方式は, 初めに全世界の人々と確実に知り合うことを保証する関係性(分散ハッシュ・アルゴリズム)を定め, そのルールに従って友達関係を厳密に構築・維持する方式であり, アプローチが大きく異なっている.

また, Freenet や Winny などのファイル交換システムの実装に見られるコンテンツの複製処理が, 分散共有方式の性能に与える影響についても評価・考察を行

い, 別の機会に報告したい. 今後は, デイメンションモデルにおける配信レイヤ, ポリシレイヤなど, ブローカレスの他の適用領域についても研究を進めていく.

参 考 文 献

- 1) 星合隆成: ブローカレスモデルと SIONet, 電気通信協会 (July 2003).
- 2) Sunaga, H., Hoshiai, T., Kamei, S. and Kimura, S.: Technical trends in P2P-based communications, *IEICE Trans. Commun.*, Vol.E87-B, No.10, pp.2831-2846 (Oct. 2004).
- 3) <http://www.jnutella.org/>
- 4) Clarke, I., Sandberg, O., Wiley, B. and Hong, T.W.: Freenet: A distributed anonymous information storage and retrieval system, *ICSI Workshop on Design Issues in Anonymity and Unobservability* (July 2000).
<http://freenetproject.org/freenet.pdf>
- 5) The Free Network Project.
<http://freenetproject.org/index.php?page=index>
- 6) <http://winny.info/>
- 7) <http://www.kazaa.com/>
- 8) <http://skype.com/>
- 9) Traversat, B., Arora, A., Abdelaziz, M., Duigou, M., Haywood, C., Hugly, J.C., Pouyoul, E. and Yeager, B.: Project JXTA 2.0 Super-Peer Virtual Network (May 2003).
<http://www.jxta.org/project/www/docs/JXTA2.0protocols1.pdf>
- 10) Project JXTA. <http://www.jxta.org/>
- 11) Wilson, B.J.: JXTA のすべて, 日経 BP 社 (Mar. 2003).
- 12) Stoica, I., Morris, R., Kaashoek, M.F. and Balakrishnan, H.: Chord: A scalable peer-to-peer lookup service for internet applications, *Proc. Conference of the ACM Special Interest Group on Data Communication (SIGCOMM'01)*, pp.149-160 (Aug. 2001).
- 13) The Chord Project.
<http://www.pdos.lcs.mit.edu/chord/>
- 14) SOBA Project.
<http://www.soba-project.org/>
- 15) 星合隆成, 北橋洋三郎, 星合擁湖, 原田輝俊, 三友仁志: ブローカレス型ネットワーク構成技術の性能評価, 電子情報通信学会論文誌, Vol.J98-D-I, No.11, pp.1608-1621 (Nov. 2005).
- 16) <http://www.bearshare.com/>
- 17) <http://www.limewire.com/>
- 18) Milgram, S.: The small world problem, *Psychology Today*, pp.60-67 (May 1967).

(平成 17 年 10 月 21 日受付)

(平成 18 年 5 月 9 日採録)



北橋洋三郎 (正会員)

2001年慶應義塾大学大学院理工学研究科計算機科学専攻修士課程修了。同年NTTアドバンステクノロジー(株)入社。P2P、グリッドをはじめとするネットワークシステムの設計開発に従事。電子情報通信学会会員。



星合 擁湖

1988年お茶の水女子大学大学院理学研究科修了。同年NTT交換システム研究所入所。トラフィック解析の業務に従事。現在、NTTサービスインテグレーション基盤研究所勤務。日本オペレーションズリサーチ学会会員。



三友 仁志

早稲田大学大学院国際情報通信研究科教授、早稲田大学デジタル・ソサエティ研究所長。博士(工学)。横浜国立大学経営学部卒業。筑波大学大学院環境科学研究科、筑波大学大学院社会工学研究科、日本学術振興会特別研究員、専修大学商学部助教授、教授等を経て、2000年4月より現職。郵政省郵政研究所特別研究官、電気通信審議会専門部会委員、国土審議会専門部会委員等を歴任。専門分野：情報通信経済学、情報通信の社会的影響分析。



星合 隆成

1986年NTT通信網第一研究所入所。工学博士(1994年度)。1995～1997年ベルコミュニケーション研究所(米国)客員研究員。現在、NTTネットワークサービスシステム研究所主幹研究員。1998年にブローカレスモデルを提唱して以来、その実現技術である意味情報ネットワークアーキテクチャSIONet(シオネット)を考案。2000年以降は、ブローカレス型ポリシモデルやコミュニティコラボレーションの実現技術であるCOMNet(コムネット)の研究を推進するとともに、コミュニティとP2P技術の融合、コミュニティ活性化のための活動や、P2P・ユビキタスコンピューティングの普及・啓蒙活動に深くかかわる。早稲田大学客員研究員。群馬大学客員教授。主な著書『ブローカレスモデルとSIONet』(電気通信協会/オーム社)。