

M-049

スケールフリー P2P ネットワークにおけるハブの分散的検出と検索効率化 Distributed Detection of Hubs and Search Improvement in Scale-free P2P Networks

大塚 治†
Osamu Ohtsuka

畑中 壮平‡
Sohei Hatanaka

松本 倫子†
Noriko Matsumoto

吉田 紀彦†
Norihiro Yoshida

1. 目的

非構造型 P2P ネットワークにおいて、いかに効率的な検索を実現することは重要な課題になっている。この問題は、検索成功率の低下、トラフィックの抑制をどう両立させるかという問題に集約される。

ここで、非構造型 P2P ネットワークには Power-Law の性質が現れていることが報告されている [1]。これは、非構造型 P2P ネットワークはスケールフリーネットワーク [2] であり、ハブのようなノードが存在することを指している。ハブはリンク数を多数持つノードである。そのため、多くのノードに検索パケットを送信することができるので、リンク数の少ないノードに比べてコンテンツを獲得できる可能性が高い。よって、ハブ方向へ検索パケットを誘導すれば、リンク数の少ないノードの方向には検索パケットを送信せずに済むことが考えられる。これにより検索成功率の低下を抑制し、検索パケットの削減をさせることが期待できる。

これをふまえ本研究では、ハブ方向の検索を行うことによって検索の効率化を図ることを目的とする。このときの効率化とは、パケット数の削減を指す。

しかし、P2P ネットワークには、全体を管理するノードが存在しない。また、個々のノードは自分のリンク数の情報しかわからず、どのノードがハブであるかが判別できない。そこで、各々のノードが持つ局所的なリンク数の情報を利用しハブの位置を自律的に検出できる方法を考える。

しかし、どのノードもハブを知るようになると、検索がハブに集中してしまう。そして、TTL には特別な変更を加えないために、ハブの周辺にしか検索パケットは行き渡らないので、検索成功率の低下を招いている。これらを防ぐために、ハブまでの距離も考慮に入れ、リンク数は最大ではないが自ノードに近いノード（以下、準ハブと呼ぶ）の方向に検索を行うようにし検索の集中を低下させ、検索成功率の低下の抑制もできるようにする。

関連研究として、センサネットワークのノードからトポロジ情報を集約するという方式が提案されている [3]。しかしながら、本研究では完全に分散的にハブを求めており、先の研究のようにサーバなどによって求めることはしない。

以上のことにより、本研究では、スケールフリーネットワークである、P2P ネットワーク上のハブを自律的に検出し、そのハブ方向を利用して検索効率化を図る。また、距離の概念の導入により、ハブへの検索の集中の低減を図る、この3点を目的とする。以下、2. では提案手法の設計を示す。3. では、その設計に基づいて作成したシミュレータによる実験を行い、その結果を基に考察を行う。4. にて、この研究のまとめを行う。

2. 設計

2.1 ハブの検出

ハブの検出について説明する。まず各ノードには、ハブリストと呼ばれる、どのノードがどのぐらいのリンク数であるかのどの情報として以下のパラメータを持つ、表1のようなリストを持たせる。

表 1: ハブリスト

順位	Guid	リンク数	Hop 数	方向
1				
2				
3				

- 順位
自ノードが知り得る中でリンク数が多いノードの順位。
- Guid
Global Unique Identifier の略であり、ノードを識別するために使用。
- リンク数
左記の Guid を持つノードのリンク数。
- Hop 数
左記の Guid を持つノードまでの Hop 数。
- 方向
左記の Guid を持つノードがある方向の自分の隣接しているノードのアドレス。

ハブリストの初期状態は、自ノードの情報のみが格納されている。各ノードは、パケットを送信する際に、自分が持っているハブリストを付随させる。ハブリストを付随させたパケットがノードに届くと、パケット内に含まれる送信してきたノードのハブリストと自分のハブリストを比較する。送られてきたハブリストに自分がまだ位置の知らないリンク数の多いノードが存在したならば、自分のハブリストにそのノードの情報を加え更新する。この動作を繰り返し行うことで、ハブリストにはリンク数の多いノードの情報、すなわちハブの情報が集まり、スケールフリーである P2P ネットワークにおいて、自律的にハブを検出することができる。図1に動作の例を示す。

また、ハブリストの更新には、Hop 数に着目したルールを設ける。本研究の検索においては TTL に特別な設定をすることはしない。そのため、TTL 以上に離れたノードには、検索パケットが届くことがないので、ハブリストに格納しても利用することができない、そのため TTL より 1 少ない範囲までのノードを格納することにする。

2.2 評価関数に基づいた準ハブの決定

準ハブ方向の検索の際の準ハブの決定は、ハブリストの中の情報から、リンク数の順位と Hop 数の順位をもとにした評価関数を用い決定を行う。具体的には、以下のような算出方法をとる。

ハブリスト内のあるノードにおいて、ハブリスト内で

†埼玉大学, Saitama University

‡NEC ネット SI, NEC Networks & Systems Integration Corp.

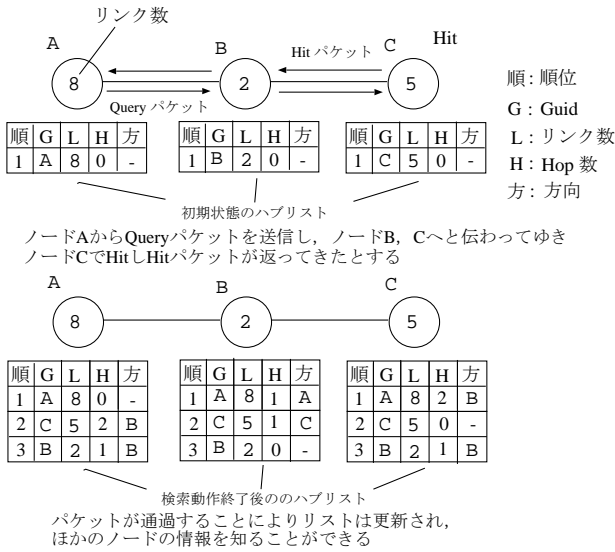


図 1: リンク情報の伝達

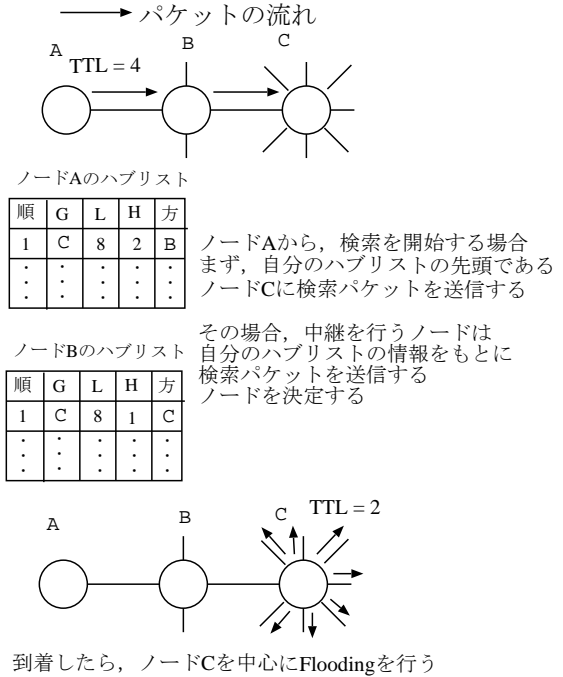


図 2: 検索の手順

のリンク数の降順の順位を L , Hop 数の昇順の順位を D とする. このときリンク数の重みを ω_L , Hop 数の重みを ω_D とし, 以下の評価関数によって値 W を得る.

$$W = \omega_L L + \omega_D D$$

そして, この W が一番小さいノードを準ハブに決定する.

たとえば, あるノードのハブリストの中にリンク数 7, Hop 数 3 のノード A, リンク数 5, Hop 数 1 のノード B の情報があったとする. この場合, ノード A のリンク数の降順の順位は $L = 1$, Hop 数の昇順の順位は $D = 2$, 同様にノード B では $L = 2, D = 1$ となる. このとき, リンク数の重み $\omega_L = 1$, Hop 数の重み $\omega_D = 2$ である評価関数に通した場合の値 W は, ノード A では $W_A = 5$, ノード B では $W_B = 4$, となり $W_A > W_B$ のためノード B を準ハブとして採用する.

2.3 検索動作

2.3.1 ハブ方向への検索動作

検索の方法については, まず検索を行うノードは, 自分のハブリストのリンク数が一番多いノードをハブとして, その方向にクエリーパケットを送信する. 中継ノードはその決定されたハブまで, 自らが持つハブリストによって, 順にリンクをたどっていく, その際に Flooding (全隣接ノードへのクエリーパケットのマルチキャストの連鎖) と同様に TTL を 1 ずつ減少させる. そして, パケットがハブに到達した場合には, そこから残り TTL で Flooding を行う. 図 2 に検索の手順の例を示す.

2.3.2 準ハブ方向への検索動作

検索動作としては, 2.3.1 に述べた動作とほぼ同様である. しかしながら, Flooding を行わせようとするノードは, ハブリストの中で一番多いノードではなく, 評価関数によって求められた値が一番小さい準ハブの方向にパケットを送信するようにする.

3. 実験と検証

1. で述べた目的が達成されているかを確認するため.
2. の設計をもとにシミュレータを作成し実験を行った.

3.1 実験

実験としては以下の 3 点に注目をした.

1. ハブの検出が出来ているか確認する. ハブ特定の手法は同一であるが, パケットに乗ってノードの情報は拡散するため, ハブ方向の検索, 順ハブ方向の検索両方の手法において検出を確認した. 指標としては, 一番リンク数が多いノードの存在をどれだけのノードが検出できたかを計測する.
2. 検索の効率化が行われているかを検証するために Flooding と提案手法を比較する. 提案手法は, 実験結果において, ハブ方向への検索を HubTrans と準ハブ方向への検索を PriTrans とグラフ内で記した. 比較の対象は, そのシミュレータにおいて検索時に送受信されたパケットの数である. また, 互いの手法によってほかの要素もどのように変化したかも計測を行う.
3. ハブ方向の検索および, 準ハブ方向検索での, 検索の集中度合いを比較する. この場合の集中度合いとは, 検索において, ハブや準ハブとして選択するとき 1 回でもハブや準ハブとして選択されたノードの総数, そして, 検索においてもっとも選択されたノードの選択回数之二つを基準とする.

3.2 実験条件

ネットワーク上にノードを 4000 個配置し、総数 5000 個のコンテンツの中から、各ノードは 0 から 80 個をランダムに保持する。また、ネットワークのトポロジはスケールフリーネットワークである。

このとき、ランダムなノードから用意されたコンテンツの中からランダムに検索させ、それを 8000 回行う。検索の TTL は 4 であり、検索に用いられるハブリストの最大サイズは 10 とする。

また、準ハブ方向の検索の場合に用いられる評価関数のリンク数の順位の重さは $\omega_L = 1$ また、Hop 数の順位の重さは $\omega_D = 1$ である。なお、すべての実験結果は、8000 回の検索動作を 1 セットとして、トポロジを変えずに、1 セットごとにランダムシードを変化させて得られた結果の 5 セット分の平均値で示す。

3.3 実験結果

3.3.1 実験 1

このシミュレータの動作後、シミュレータのネットワーク上でどれだけノードが最高リンク数を持つノードを検出できたかを調べた。結果としては、一番リンク数が多いノードを検出できたノード数は、どちらの検索手法においても 4000 ノード中 3597 ノードであった。

3.3.2 実験 2

各々の手法において、検索パケット数の比較実験を行った、具体的にはシミュレータによって検索のシミュレーションを行い、Flooding、ハブ方向の検索、評価関数による準ハブ方向の検索場合の総パケット数を比較した。グラフ 3 が結果となる。グラフにおいて、検索 100 回にシミュレータ上でやりとりされた総パケット数の平均を縦軸にした。

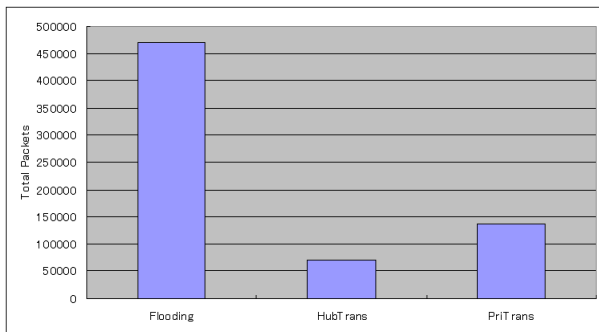


図 3: 総パケット数

また、検索成功率、コンテンツヒット時の Hop 数についても計測を行った。このときの検索成功率とは、ネットワーク上に存在しているコンテンツが実際に検索において発見された割合である。検索成功率はグラフ 4、コンテンツヒット時の Hop 数はグラフ 5 が結果となる。

3.3.3 実験 3

検索集中についての実験を行った。具体的にはハブ方向の検索、準ハブ方向の検索の集中度合いを計測した。グラフ 6 とグラフ 7 が結果となる。このときグラフ 6 は、

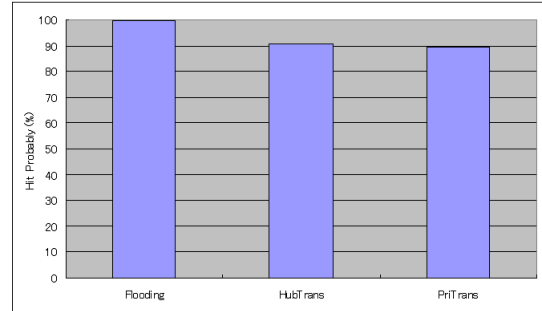


図 4: 検索成功率

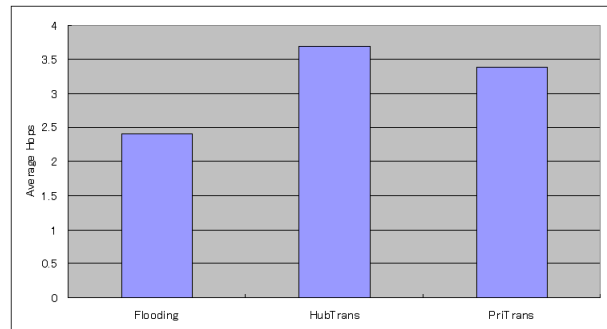


図 5: コンテンツ Hit 時の Hop 数

検索 100 回当たりにハブや準ハブに 1 回でも選択されたノードの数の平均を縦軸とした。また、グラフ 7 は、検索 100 回当たりの最も多くハブや準ハブに選択されたノードの選択回数を縦軸とした。

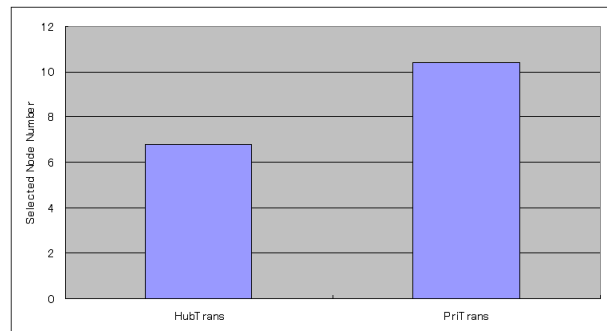


図 6: 1 回でもハブや準ハブとして選択されたノードの総数

3.4 考察

まず、検索の効率化について考察する。パケット数について、Flooding に対してハブ方向の検索では約 $1/7$ 、準ハブ方向の検索では約 $1/3$ という結果となった。この結果については、リンク数の少ないノードへのパケットを削ったために、この結果が出たとみられる。

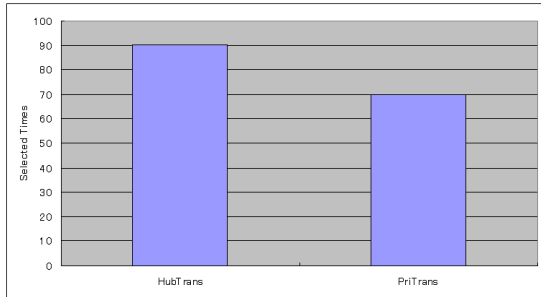


図 7: 一番選択されたノードの選択回数

また、ハブ方向の場合と準ハブ方向の場合のパケットの差について考える。この差であるが、準ハブ方向の検索では Hop 数も考慮に入れられているため、ハブ方向の検索よりは Flooding を開始するノードが近くになりやすくなる。そのために、Flooding に費せる残りの TTL がハブ方向の検索より大きくなるために準ハブ方向の検索がハブ方向の検索よりもパケット数が多くなったのである。

検索成功率については、Flooding に対しては両手法ともに、約 9% 程度の低下がみられた。また、ハブ方向の検索、および、準ハブ方向の検索について、検索成功率の点では差異がみられなかった。しかしながら、検索を失敗した場合には、ハブ方向の検索では、その次にリンク数が多いノードに、準ハブ方向の検索では、次に評価関数の値が小さいノードの方向に検索を行うように最高で 3 位まで繰り返させた場合の検索成功率がグラフ 8 になる。このグラフを見るとハブ方向、準ハブ方向に関わらず、検索方式を Flooding にしたときとほぼ変わらない検索成功率を出すことができた。

また、コンテンツヒット時の Hop 数におけるハブ方向の検索と準ハブ方向の検索の差であるが、それは準ハブ方向の方が距離を考慮に入れている分だけ検索ノードから近い位置で Flooding を行っているため準ハブ方向の検索の方が近いところからコンテンツを取得できていると考えられる。

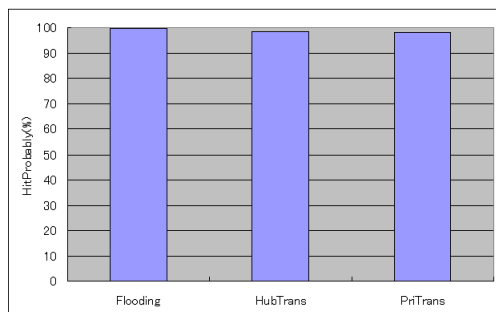


図 8: ハブリストの上位 3 位までみたときの検索成功率

次に、ハブ方向の検索と準ハブ方向の検索についても

考察を行う。今回の実験結果では、評価関数で距離の概念を入れることにより一番選択されたノードで選択回数を約 20% 強下げる結果となった。また、検索時に選択されたノードの数も、6 ノードから 10 ノードへと増加している。これにより一番リンク数がノードへの検索の集中が分散されていると考えられる。この結果について考察をすると、ハブ方向の検索では、一番リンク数が多いノードを検出できるすべてのノードは、そのノードをハブとして検索を実行するため検索が集中する。しかし、評価関数により、距離も含めて算出した方法では、一番リンク数が多いノードではなく、それよりもリンク数が少ないが検索ノードに近いノードが準ハブに選ばれやすくなるので今回のように、集中度が低下したのであると考えられる。

また、このとき Hop 数の重みをより重くすれば、近隣のノードがより選ばれやすくなるので、より検索は分散されると考えられる。

4. まとめ

本研究では、非構造型 P2P ネットワークがスケールフリーネットワークの性質を持つという報告から、ハブを自律的に検出し、そのハブを利用した、P2P ネットワークにおける検索を効率的に行う手法を提案した。これにより、検索時のトラフィックを抑制することができた。

また、距離の概念を導入することによって、より検索の集中を低下させることも可能となった。

本研究では、ハブを検索のみに利用したが、ハブをさらに利用し、コンテンツ配置やインデクス配置など、検索動作以外のところでも活用することが考えられる。つまり、よりハブを利用した検索効率化が課題となる。また、準ハブ方向の検索においては、評価関数を重みで静的に決定して行ったが、理想的な重みの計算、また動的に評価関数に更新するという方法も考えられる。つまり検索の分散と検索の効率化をより両立させることも今後の課題となる。

参考文献

- [1] L. A. Adamic, R. M. Lukose, A. R. Puniyani and B. A. Huberman, "Search in Power-Law Networks", Physical Review E, Vol.64, 46135-46143, 2001.
- [2] アルバート=ラズロ=バラバシ, "新ネットワーク思考~世界の思考を読み解く~", NHK 出版, 2002.
- [3] 茂木 信二, 吉原 貴仁, 堀内 浩規, "センサネットワークのためのトポロジ情報収集方式の提案", 情報処理学会第 67 回全国大会, 2005
- [4] 大塚 治, "P2P ネットワークのスケールフリー性を活用した検索効率化", 埼玉大学卒業論文, 2007