

# 非構造型 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

吉田学史<sup>†</sup> 松本倫子<sup>†</sup> 吉田紀彦<sup>†</sup>

<sup>†</sup> 埼玉大学大学院理工学研究科

## 1 はじめに

非構造型 P2P ネットワークでは、コンテンツの検索の効率化が課題となっている。この非構造型 P2P ネットワークは、スケールフリーネットワークの性質を持つことが報告されており [1]、ハブと呼ばれる多くのリンクを持つノードが存在する。また、コンテンツの人気度の分布がべき乗則に近い性質を持つことが報告されている [2]。非構造型 P2P ネットワークでは検索成功時に、コンテンツの複製や、インデックスのキャッシュが行われる。そのため、人気度の高い（検索頻度の高い）コンテンツほど発見しやすく、人気度の低い（検索頻度の低い）コンテンツほど発見しにくくなる。

本研究では、ハブを利用して検索の効率化を図ることを考える。ここでいう検索の効率化とは、高い検索成功率を維持しつつ、検索にかかるパケット数を抑制することである。ハブを利用した検索効率化の手法として、ハブの分散的検出とハブを利用した検索手法を提案する。ただし、単純な適用では、検索パケット数は抑制できても、検索成功率が低下するので、コンテンツの人気度を考慮した検索、ノードのリンク数に応じたキャッシュの住み分けという拡張を行う。

以上について、シミュレーションによる評価に基づき、有効性を示す。

## 2 提案手法

### 2.1 ハブの検出

ハブの検出方法について説明する。まず各ノードには、ハブリストというノードのリンク数等の情報を保持するリストを持たせる。表 1 にハブリストのデータ構造を示す。

表 1: ハブリストの例

順位	GUID	リンク数	Hop 数	方向
1				
2				

ハブリストには、最初是自己ノードの情報のみが格納されている。各ノードは、検索パケットやヒットパケットを送信する際にハブリストを付加させる。パケット

Search improvement in unstructured P2P networks based on their scale-free nature

Satoshi Yoshida<sup>†</sup>, Noriko Matsumoto<sup>†</sup> and Norihiko Yoshida<sup>†</sup>

<sup>†</sup> Graduate School of Science and Engineering, Saitama University

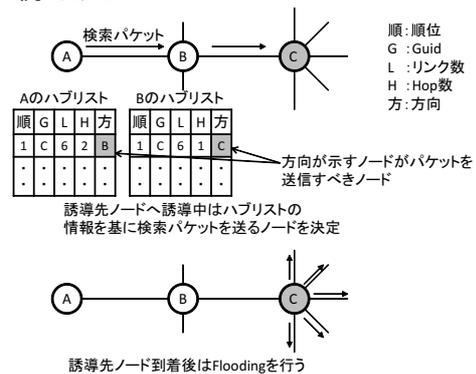
を受け取ったノードは、パケットに付加されたハブリストと自らのハブリストの比較を行う。もし、より多くのリンク数を持つ未知のノード情報があればそれを自分のハブリストに追加し更新をする。これを繰り返すことで、次第にハブリストにはリンク数の多いノード、すなわちハブの情報が集まる。

### 2.2 誘導先ノードの選択

各ノードが検索を行う際は、まず、パケットを誘導するノード（誘導先ノード）の選択を行う。誘導先ノードは、ハブリストの情報を基にしたルーレット選択で選ばれる。リンク数に比例した確率で選択されるため、リンク数の多いノードほど選ばれやすくなる。

### 2.3 誘導先ノードへの検索誘導

誘導先ノードを選択したら、検索ノードは誘導先ノードの方向へと検索パケットを送信する。検索パケットは、選択された誘導先ノードに到着するまで各ノードのハブリストに従い、リンクをたどっていく。検索パケットが誘導先ノードに到着したら、決められた TTL (Time To Live) で Flooding を行う。図 1 は、ノード A がノード C を誘導先ノードとして選択した際の検索誘導の例である。



## 3 検索成功率向上のための拡張

### 3.1 人気度の推定

提案手法の拡張ではコンテンツの人気度を利用する。しかし、非構造型 P2P ネットワークには、ネットワーク全体を管理するノードが存在しないため、各ノードはコンテンツの真の人気度を知ることはできない。各ノードは自律的に得られる情報からコンテンツの人気度を推定する必要がある。そこで本手法では、コンテ

コンテンツの人気度を、そのコンテンツの検索パケットを受信した回数とする。

### 3.2 コンテンツの人気度を考慮した検索

検索するコンテンツの人気度の高さを判断し、その結果を基に誘導先ノードの選択数や誘導先ノード到着後の Flooding の TTL を変化させることで人気度に応じた検索を行う。

人気度の高さを判断するための基準として各コンテンツの人気度の平均  $P_a$  を用いる。基準となる選択数を  $S$ 、基準となる TTL を  $T$ 、検索コンテンツの人気度を  $p$  として、以下のように 5 パターンに場合分けして選択数と TTL を決定する。 $m$  はパラメータである。

$$\left\{ \begin{array}{ll} \text{選択数} = S - 2, \text{ TTL} = T & (p > m^2 P_a) \\ \text{選択数} = S - 1, \text{ TTL} = T & (m P_a < p \leq m^2 P_a) \\ \text{選択数} = S, \text{ TTL} = T & (\frac{P_a}{m} < p \leq m P_a) \\ \text{選択数} = S + 1, \text{ TTL} = T & (\frac{P_a}{m^2} < p \leq \frac{P_a}{m}) \\ \text{選択数} = S + 2, \text{ TTL} = T + 1 & (p \leq \frac{P_a}{m^2}) \end{array} \right.$$

これにより、人気度の低いコンテンツを検索する際は、選択数や TTL が増え、多くのノードに検索パケットが行き渡り検索成功率が向上すると考えられる。また、人気度の高いコンテンツを検索する際は、選択数が減少し無駄なパケットの発生を防げると考えられる。

### 3.3 キャッシュの住み分け

ハブとハブ以外のノードで異なるキャッシュ置換アルゴリズムを用いることでキャッシュの住み分けを行う。各ノードでキャッシュの置換えが必要になった場合、まず自身がハブかどうかの判定を行う。自身の持つハブリストのリンク数の平均  $L_a$  と自身のリンク数  $l$  を比較し、自身がハブかどうかを判断する。 $n$  をパラメータとして

$$l > n L_a$$

の場合、自身をハブであると判断する。また、

$$l \leq n L_a$$

の場合は自身はハブでないと判断する。

この判定結果を基に使用するキャッシュ置換アルゴリズムを決定し、キャッシュの置換えを行う。

- 自身をハブであると判断した場合  
キャッシュ内で最も人気度の低いコンテンツのインデックスをキャッシュから削除する。
- 自身をハブでないと判断した場合  
キャッシュ内で最も人気度の高いコンテンツのインデックスをキャッシュから削除する。

つまり、ハブのキャッシュには人気度の高いコンテンツのインデックス、ハブ以外のノードのキャッシュには人気度の低いコンテンツのインデックスが残ることになる。これにより、ハブ周辺に多様なコンテンツのインデックスのキャッシュが集まるため、検索成功率の向上が期待できる。

## 4 実験

提案手法が検索効率の向上に有効であることを確かめるため、シミュレーション実験を通して Flooding、提

案手法 (Hub Trans)、提案手法の拡張 (Ex-Hub Trans) の比較を行う。比較の対象となるのは、検索によって生じたパケット数と検索成功率である。

また、提案手法のパラメータ  $m, n$  と基準となる選択数  $S$ 、基準となる TTL である  $T$  はそれぞれ、 $m = 1.5, n = 0.5, S = 3, T = 3$  として実験を行った。

図 2 は各手法のパケット数の推移を図示したグラフである。提案手法、提案手法の拡張とともに Flooding と比べてパケット数を削減できていることが確認できる。

図 3 は各手法の検索成功率の推移を図示したグラフである。提案手法は Flooding と比べて検索成功率が低下しているが、提案手法の拡張によって Flooding と同等の検索成功率を達成していることが確認できる。

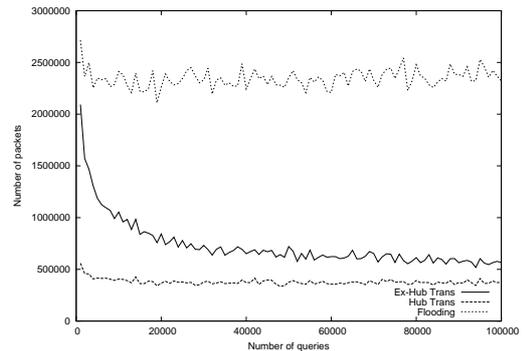


図 2: パケット数の推移

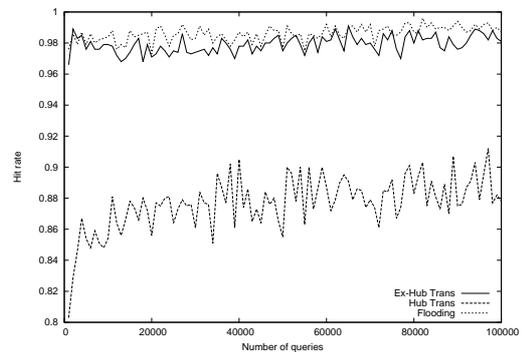


図 3: 検索成功率の推移

## 5 まとめ

本研究では、非構造型 P2P ネットワークにおける検索効率化の手法としてハブを利用した検索効率化手法を提案した。また、人気度を考慮した検索手法、キャッシュの住み分けという提案手法の拡張を行った。シミュレーションにより、提案手法によって高い検索成功率を保ちつつ、パケット数が抑制されていることが確認できた。

## 参考文献

- [1] L. A. Adamic et al., "Search in power-law networks", Physical Review E, Vol 64 046135, 2001
- [2] K. P. Gummadi et al., "Measurement, modeling, and analysis of a peer-to-peer file-sharing workload", SOSP'03, 2003.