

ファイル細分型 P2P における最適化手法の提案

湯澤 孝有 服部 晃和 横田 隆史 大津 金光 馬場 敬信 †
宇都宮大学工学部情報工学科 ‡

1 はじめに

近年、ブロードバンドの普及により、コンテンツ配信等の大量データ転送が行なわれるようになってきた。これらは従来 client-server(CS) 型によって運用されたが、トラフィックの増大や、それに対処するコストの増加など拡張性による限界が指摘されるようになった。この解決方法として P2P (Peer-to-Peer) 技術が注目されている。

現在、様々な P2P 技術があるが、拡張性、信頼性、転送効率の全てを満たしている技術は開発されていない。本稿では、Fragment Pool (FP) と呼ぶ機構を導入することで上記問題点を解決する PureP2P 型ネットワーク FONet (Fragment-pool Organizing Network) を提案する。

2 目的

P2P は (1)HybridP2P, (2)PureP2P に分類できる。

(1)HybridP2P とはピアの IP アドレス、ポート番号や所持ファイル等の情報を格納している index server を設置する P2P である。静的に index server を設置する Napster^[1], eDonkey^[3] 等があり、動的に index server を設置する KaZaA^[2] がある。HybridP2P はファイルの転送効率が良く、ネットワーク全体のトラフィックが少ないが、静的に index server を設置するものでは拡張性が、動的に置く物では index server の役割になったノードへの負荷の集中がそれぞれ問題となる。

これに対し (2)PureP2P とは、index server を設置しない P2P である。Gnutella^[4] に始まり、キャッシュを生成して効率の向上を図った FreeNet^[5] や、木構造により最適化を図った Winny^[6] などがある。PureP2P は、index server を持たないため、特定のノードの信頼性が問題になることもなく、また、負荷が均等に分散するが、多数のノードが参加すると検索範囲の増加によりファイルの収集効率が極端に落ちる問題が生じる。

また、HybridP2P, PureP2P に共通する手法として、自分の嗜好情報を基に他のノードとグループを作ることでより最適化を行う SIONet^[7] などが存在する。

高速回線を持つ数 10 万ノードが参加でき、効率の良いファイル転送を環境を実現するためには次のようなことが必要となる。

- ・高い拡張性
- ・通信路の負荷分散
- ・効率的な転送

これらを満たす手法として、PureP2P において転送効率を高める手法を FONet で採用する。次節で更詳しく説明する。

3 FONet system

3.1 概要

* A new P2P optimization technique that uses fragments of file contents

† Takatomo Yuzawa, Akikazu Hattori, Takashi Yokota, Kanemitsu Ootsu and Takanobu Baba

‡ Department of Information Science, Faculty of Engineering, Utsunomiya University

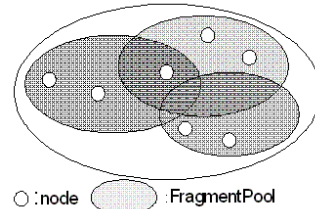


図 1: FONet の概念図

FONet では、高い拡張性を保証するために PureP2P 型とし、さらに、通信路の負荷分散と転送効率の向上のために、ファイルを細分化して転送することを基本とする。各ノードは、ノード間相互の接続情報を持ち、論理的集合を形成する。

ノードがファイルを収集する場合、各ノードが持つ接続情報を用いて、他のノードに要求メッセージを送信する。要求メッセージを受信したノードは、当該ファイルの断片を持っていれば要求ノードに送信する、そうでなければ要求メッセージを他の参加ノードに転送する。ただし、要求メッセージの無用な増殖を抑えるため、他ノードへの転送回数に上限を設ける。要求元のノードには、要求ファイルの断片が順不同で送られるが、最終的には全体が揃う。

ファイルを細分化して扱う方法では、目的ファイルの断片を持っているノードに対して効率的に要求メッセージを伝えることが、転送効率向上の鍵となる。そのため、FONet では、ファイルが属するカテゴリを設定し、カテゴリ毎にファイルを多く所持しているノードの集合 Fragment Pool (FP) を考える。ファイルは細分化されているため、FP はそのカテゴリのファイルを持っている確率の高いノードの集合となる。FONet の概念を図 1 に示す。目的のファイルを収集する時、そのファイルが属するカテゴリの FP を探し、その所属ノードに対して要求メッセージを出せば、高効率でファイル断片を収集できるものと期待できる。

3.2 Fragment Pool 実現方法

上述の FP を実現するため、FP パラメータ (FPP) を導入する。FPP とは、カテゴリ毎にファイル断片の所持率を表したものであり、カテゴリ数だけの次元からなるベクトルで表現される。各ノードはファイル断片の所持状況に応じた FPP を持つ。FONet では簡略化のため、各ノードが最大所持率のカテゴリに対応する FP に属しているとみなす。FONet をはじめ PureP2P では他のノードの接続情報を格納する接続リストを持ち、これに基づいて通信する。通常の PureP2P では、ノード間で接続情報を交換することで接続情報を保全している。FONet では接続情報とともに互いの FPP 値を交換し、同一 FP に属しているノードの接続情報を優先的に残し、そうでない (所属 FP の異なる) ノードからの接続情報を優先的に捨てる。この操作により、同一 FP に属するノードの接続情報を重点的に集めることができる。

ノード間の接続状況は、この接続リストで定義されるため、結局、特定のカテゴリのファイル断片を多く含むノードがFPとして互いに密に接続される。

3.3 FPの検索

上記のように、各ノードは自身が現在保持しているファイルの断片の状況によってFPを形成している。目的のファイルが同じFPに属していれば、上記の方法で高効率な転送が出来るが、異なるFPに属する場合には、当該FPの情報(参加ノードなど)を効率的に得る方法が必要となる。このため、FONetでは、以下の操作によって目的のFPの検索を行なう。

ノードは、収集しようとするファイルのカテゴリの断片に対し重み付けを行ない、FPPを再計算する。同時に、収集しているカテゴリのFPPが断片の収集に伴い増加する。結果、FPPは収集中のFPに偏ることになり、収集前と比べて現在の所属FP内ノードでの優先度は減少し収集対象FP内ノードでの優先度が増加する。これにより、収集対象FP内ノードとの繋がりが密となり、そのFPの新規参加ノードなり、高効率な転送が行なえる。

3.4 断片集積

細分化されたファイルが分散配置されている状況では、必ずしも全てのファイル断片がFP内に保持されているとは限らない。このため、例えFPが正しく検索できても、FP外にある断片が収集できないために、結果として収集効率が著しく悪化する。また、ファイル収集中は要求ノードに対して集中的な通信が発生するために、輻輳が発生する可能性もある。

この問題を避けるため、要求ノードへの転送が輻輳の為に長時間ブロックされる状況で、FPの外辺部にあるノードがファイル要求を受けた場合には、要求元ノード自身ではなく、同一FPにある周辺のノードに送ることを許す。このノードが中継ノードとなり、効率的な収集が期待できる。また、このようにファイル断片を集積することにより、FP内のファイル存在確率をより高めることできる。これによって、以降のファイル収集効率が向上する。

4 評価

我々はファイル収集時間を評価するためにFONetシミュレータCsimとCS型ネットワークモデルシミュレータSsimを開発した。

4.1 シミュレータ概要

Csimでは、予めファイルを細分化した断片をランダムに選んだノードに分配する。各ノードはランダムに初期接続ノードが与えられる。全てのノードは、全てのFPのカテゴリ、ファイル名、ファイルの分割数を既知とする。シミュレーションの開始から一定時間経過したのちに所在FPと同カテゴリのファイルの収集を開始し、この収集時間を測定した。主なシミュレーションパラメータを表1に示す。

Ssimでは、高速通信路を持つ、常に安定したサーバノードが存在していると想定し、全クライアントノードはこれらのサーバの所在、それらの所持しているファイルを既知とする。基本的にCsimと同じパラメータを用いるが、サーバの通信速度をクライアントノードの10倍に設定している。Csim同様に、1ファイルの収集時間を測定する。

表 1: シミュレーションパラメータ

FP種類	3
ファイル種類	FP毎に6
最大メッセージ転送回数	100
1発呼メッセージ数	10
バンド幅	2000[Kbyte/s]
断片サイズ	256[Kbyte]
断片数	平均60

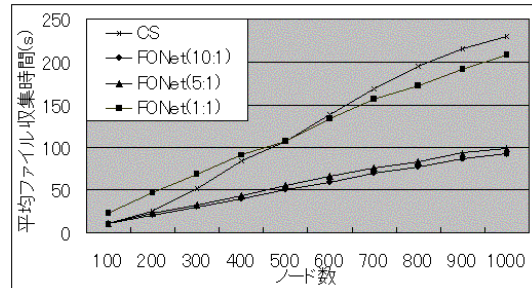


図 2: FONetとCSモデルとの比較

両シミュレータの共通事項として、どのノード(観測ノード、サーバノード除く)も一定時間毎に一定確率でオンラインのノードはオフラインとなり、また、オフラインのノードはオンラインに復帰する。同時に一定確率の元でランダムに選定したファイルの収集を開始する。

4.2 実験結果

CsimとSsimにおいて、ノード数を100~1000と変化させた時の収集時間の平均を比較する。また、Csimにおいて、FP内外のノードに対する要求メッセージ量の比率を(内:外)=1.(1:1) 2.(5:1) 3.(10:1)と変化させ、その収集時間の平均を比較する。図2はノード数に対するファイル収集時間の平均値を表すグラフである。横軸がノード数、縦軸が平均収集時間を示す。

図2より、FONetにおいて、特にFPへの要求メッセージを出す比率を(5:1)、(10:1)とするとときに非常に効率が良いことが分かる。

FONetではそのファイル収集効率はFP内のノードの構成状況のみによるため、他のPureP2Pシステムのように、ノード数が増加しすぎたためにファイルの収集効率が極端に低下する、と言う状況を防ぐことができる。

5 おわりに

本論文では、PureP2Pにおけるファイル転送の効率化手法としてFONetを提案し、シミュレーションによりその有効性を示した。

今後の予定として、FONetの最適化手法をシミュレータで評価し、これらの最適化手法を実装し、評価する。

謝辞 本研究は、一部日本学術振興会科学研究費補助金(基盤研究(B)14380135, 同(C)14580362, 若手研究14780186)の援助による。

参考文献

- [1] <http://www.napster.com/>
- [2] <http://www.kazaa.com/>
- [3] <http://www.edonkey2000.com/>
- [4] <http://welcome.to/gnutella>
- [5] <http://freenetproject.org/>
- [6] <http://winny.info/>
- [7] 星合隆成:“ブローカレス型検索モデルと意味情報ネットワーク SIONet”第2回 Jnutella Workshop 資料(2002)