LM-011

P2P ネットワークにおけるノード信頼性評価の効率化
Improvement of Node Trustness Evaluation in P2P Networks

吉田 雄亮 † 松本 倫子 † 吉田 紀彦 †
Yusuke Yoshida Noriko Matsumoto Norihiko Yoshida

1. はじめに

ピアツーピア (P2P) ネットワークでは中央集権的なサーバーが無く，検閲が難しいことから，著作権を無視したコンテンツ共有が行われており，それを目当てに参加するユーザが数多く存在する．また，そのようなユーザを標的として，不正コンテンツ[1]を配布して攻撃するピア (以下，不正ピアと呼ぶ) も存在する．

攻撃によっては P2P ネットワークを通じて個人情報や機密文書が流出し，多大な被害を被ることがある．このため，P2P ネットワークの利用において，不正ピアを発見し，排除することは非常に重要な課題であると言える．

これに対する解決策の 1 つに STEP (Simple Trust Exchange Protocol) [1] がある．STEP では，ファイルを送信してきたピアを評価し，“トークン”と呼ばれるファイル提供の証明書を発行する．このトークンを収集し，“knowledge”メッセージとして定期的に隣接ピア同士で交換することで，より多くのピアによる，それぞれのピアの評価を得ることができ，時間の経過とともにより正確な信頼度を得られる．

STEP では knowledge メッセージの配布に flooding を用いているため，kowledge メッセージの配布によりネットワーク帯域が圧迫され，コンテンツの検索・ダウンロードといった通常の利用におけるパフォーマンスが低下する可能性がある．

そこで本研究では knowledge メッセージの効果を維持しつつ，そのトラフィックを削減する手法を提案し，シミュレーションによる評価を行った．

2. STEP

STEP (Simple Trust Exchange Protocol) は，EigenTrust [2]，P2PRep [3]，NWAY [4] といった P2P ネットワークにおける信頼性確保の手法で提案された概念を複数取り入れた規約である．

STEP ではピアの提供するサービス[2]の評価を提供ピアの評価とし，表 1 の構造を持つトークンと呼ばれるサービス提供証明書を作成，サービス提供者，利用者の両方で保管される．トークンには有効期限があり，作成されてから一定時間過ぎると無効となる．また，公開鍵暗号による署名により，偽造・改ざんを防いでいる．

表 1: トークンの構造

要求者・提供者の ID
トークンの作成日時
評価 (good, bad)
要求者・提供者の署名

評価は正常 (good) 又は不正 (bad) のいずれかであり，ピア x の信頼度 $\mathrm{Trust}(x)$ は

$$\mathrm{Trust}(x) = \begin{pmatrix} x\text{を正常と} \\ \text{評価した} \\ \text{トークン数} \end{pmatrix} - \begin{pmatrix} x\text{を不正と} \\ \text{評価した} \\ \text{トークン数} \end{pmatrix}$$

で定義される．

他のピアに関してより正確な信頼度を得るため，各ピアが保管しているトークンは knowledge メッセージと呼ばれる構造体によって定期的に隣接ピア間で交換される．

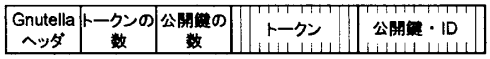

図 1: knowledge メッセージの構造

このようにして得た信頼度に基づき，隣接ピアとして接続すべき相手や，サービスを優先的に提供すべき相手を選択することで，不正ピアを正常ピアの集団から排除できる．

3. STEP における課題と対策

STEP の knowledge メッセージの配布は，それぞれのピアが隣接ピア全てに転送すること (flooding) により実現される．knowledge メッセージを転送することで，周囲のピアに正常ピア，不正ピアの存在を知らせることができるが，knowledge メッセージ転送の分だけネットワーク帯域を利用するため，コンテンツの検索・ダウンロードといった通常の利用のパフォーマンスが低下する可能性がある．

そこで本研究では正常ピアへの knowledge メッセージの効果を維持しつつ，ネットワーク全体での knowledge メッセージによる転送量を削減することを目的として不正ピアや，フリーライダーといった信頼度の低いピアへの knowledge メッセージの転送を抑制する手法を提案する．

knowledge メッセージの無駄な転送を抑制することによって，次のような効果・副作用が発生すると考えられる．

効果 knowledge メッセージのトラフィックが減少した分，コンテンツの検索・ダウンロードのパフォーマンスが向上することが考えられる．

副作用 不正ピアの近くにある正常ピアに knowledge メッセージがあまり届かなくなり，不正ピアを判別しにくくなることで，不正コンテンツのダウンロードが増加する可能性がある．

†埼玉大学，Saitama University

[1]ここでは，ウイルスに感染しているか，スパイウェアを含むアプリケーション等を指す

[2]ここでは，ファイル交換によって提供されたファイルと仮定する．

4. 提案手法

本研究では knowledge メッセージの転送を抑制する手法として次の方法を単独，あるいは組み合わせて適用する．

TTL を制御する方法

当研究室では，TTL を制御して Pure P2P ネットワークにおける不正ピアの影響を自律分散的に回避する手法を提案している [5]．これは，転送してきた検索結果で隣接ピアを評価し，不正な検索結果を送ってきた隣接ピアに検索クエリを転送する場合は，評価に応じて TTL を通常より減らして転送するものである．これにより，不正ピアに到達する検索クエリを減らし，不正ピアの影響を抑えることが可能になる．

この手法を STEP の knowledge メッセージの転送時に適用することで，不正ピアへの転送が抑制され，knowledge メッセージの配布によるネットワーク帯域使用率を減少させられると考えられる．以降ではこの手法を“TTL 制御”と呼ぶ．

flooding せず，転送するピアを選択する方法

Gnutella ではメッセージを flooding する代わりに，ランダムに隣接ピアを 1 つだけ選択して転送する Random Walk という方法も存在する．

これを knowledge メッセージの転送に適用することで，全ての隣接ピアに転送していた knowledge メッセージがいずれか 1 つの隣接ピアだけに転送されるようになるので，ネットワーク帯域使用率を減少させられると考えられる．以降ではこの手法を“選択転送”と呼ぶ．

4.1 枝 (branch) の概念の導入

詳細な説明の準備として枝 (branch) の概念を説明する．あるピアについて，1 つの隣接ピアとその隣接ピアを通じて存在を知った既知ピアの集合を枝と定義する．即ち，

$\mathrm{RootPeer}_a(x)\cdots$ ピア a の既知ピア x の存在情報を転送してきた隣接ピア

とするとき，ピア a の隣接ピア n の枝 $\mathrm{Branch}_a(n)$ を次のように定義する．

$$\mathrm{Branch}_a(n) = n \cup \{x \mid \mathrm{RootPeer}_a(x) = n\}$$

ネットワークが環状の場合であっても，pong は必ず 1 つの隣接ピアから転送されてくるため，全ての既知ピアはいずれかの枝に属することになる．例えば，図 2(左) のネットワークにおいてピア x の枝は図 2(右) のようになる．

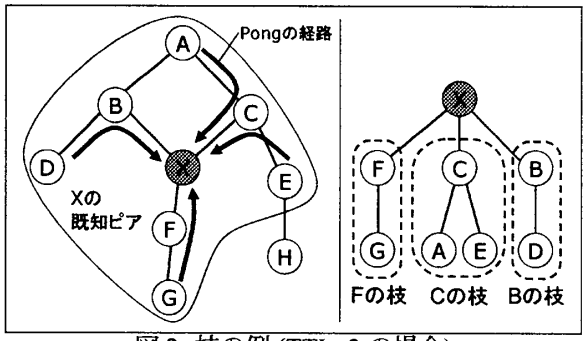

図 2: 枝の例 (TTL=2 の場合)

ここで，枝の信頼度を，枝に含まれるピアそれぞれの信頼度の平均で定義する．つまり，ピア a の隣接ピア n の枝の信頼度 $\mathrm{BranchTrust}_a(n)$ は枝に含まれるピアの数を N，ピア a から見たピア x の信頼度を $\mathrm{Trust}_a(x)$ とする時，

$$\mathrm{BranchTrust}_a(n) = \frac{1}{N} \sum_{x \in \mathrm{Branch}_a(n)} \mathrm{Trust}_a(x)$$

である．

4.2 TTL 制御による knowledge メッセージの抑制

TTL の制御は knowledge メッセージの発行時，および受信時に行う (図 3)．

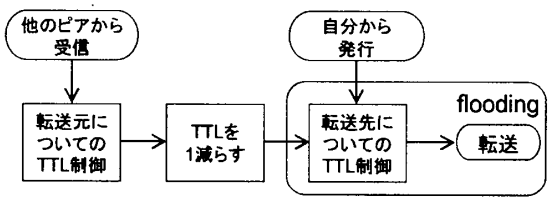

図 3: TTL 制御を使う手順

TTL 制御は具体的には図 4 のように，STEP により得られた自分の信頼度を MyTrust とする時，隣接ピア n について

$$\mathrm{BranchTrust}(n) < \mathrm{MyTrust}$$

が成り立つ場合，TTL を一定数だけ減らす．

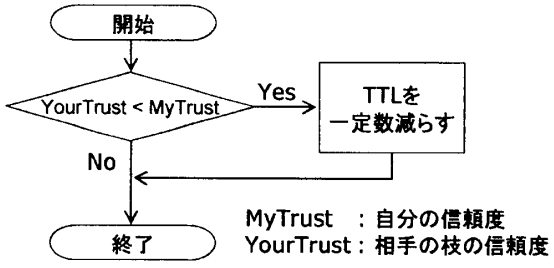

図 4: TTL 制御の処理

つまり，自分より信頼度の低い枝に属する隣接ピアから転送されてきた knowledge メッセージは信頼性が低いと言えるので，受け取った時点で TTL を一定の基準で減らす．それから通常の TTL 減算を行い，転送先の隣接ピアの枝についても同様の処理を行ったのちに転送を行う．これを転送してきた隣接ピア以外の全ての隣接ピアに対して行う (flooding)．自分から knowledge メッセージを発行する場合は全ての隣接ピアについて TTL 制御を行ってから転送する．

4.3 選択転送による knowledge メッセージの抑制

knowledge メッセージの発行，転送時に送信先を 1 つだけ選択して送信する (図 5)．ピアの選択は全くランダムに選ぶのではなく，STEP で得られたその時点での信頼性情報を利用し，枝の信頼度に比例する確率で選ぶようにする．つまり，信頼度の高い枝には高い確率で転送されるが，そうでない枝にも転送される可能性を持たせる．これは，後から参加してきた，まだ信頼度が高くない正常ピアにも knowledge メッセージが転送されるようにするためである．

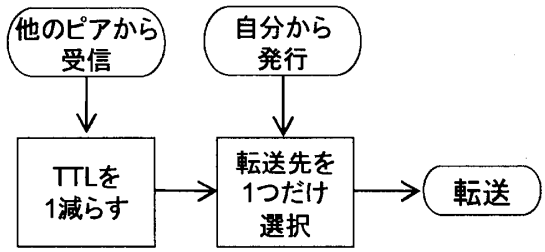

図 5: 選択転送の手順

4.4 TTL 制御と選択転送の組み合わせによる knowledge メッセージの抑制

前述の 2 つの方法を組み合わせる方法である．つまり，具体的には図 6 のように，転送元の隣接ピアの枝に対する TTL 制御を行い，選択転送で選択した隣接ピアの枝に対して TTL 制御を行う．

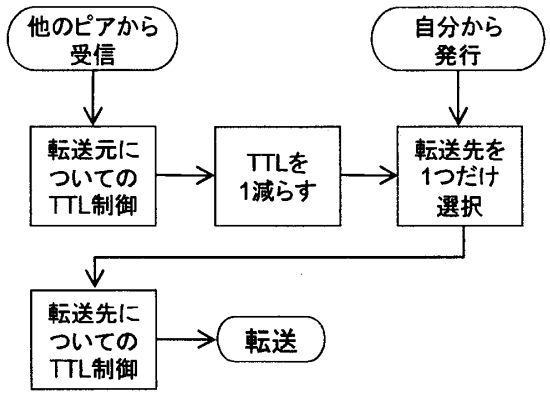

図 6: TTL 制御とピア選択を組み合わせた場合の手順

5. 実験と評価

提案手法の有効性を評価するため，シミュレータを作成し，表 2 の条件でピアを動作させ，以下のそれぞれの場合について送受信された knowledge メッセージのトラフィック，およびコンテンツのダウンロード数に占める不正コンテンツの割合を測定する．ピアの動作は全て STEP に従うものとし，正常ピアは正常コンテンツのみ，不正ピアは不正コンテンツのみを保持し，フリーライダーはコンテンツを全く持たないものとする．

1. 提案手法を使わない場合
2. TTL 制御を行う場合
3. 選択転送を行う場合
4. TTL 制御と選択転送を組み合わせて行う場合

表 2: 実験の主な条件

項目	値 (時間は仮想時間)		
シミュレーション時間	10 時間		
STEP トークン有効時間	1 時間		
初期 TTL	5		
knowledge メッセージ発行間隔	10 分		
TTL 制御で TTL を減らす量	1		
ピア総数	150		
正常ピア	50%		
不正ピア	10%	20%	30%
フリーライダー	40%	30%	20%

5.1 実験結果

knowledge メッセージの平均のトラフィックを求めると図 7 のようになった．また，コンテンツのダウンロード数に占める不正コンテンツの割合は図 8 のようになった．なお，グラフ中の Tradtional，TTLControl，SelectiveForward，TTLControl+SelectiveForward はそれぞれ提案手法を使わない場合，TTL 制御を行う場合，選択転送を行う場合，TTL 制御と選択転送を組み合わせて行う場合を意味する．また，Illegal=n%は不正ピアが n%存在する場合の実験結果であることを意味する．

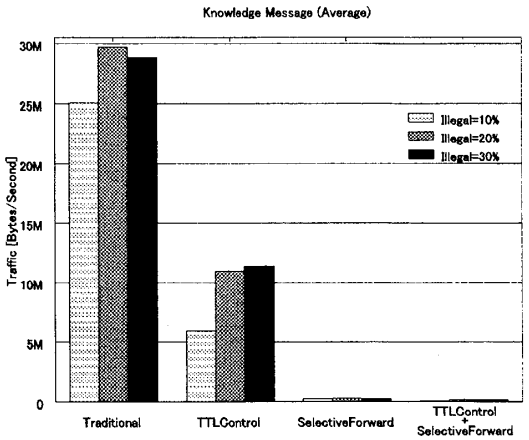

図 7: knowledge メッセージのトラフィックの平均値

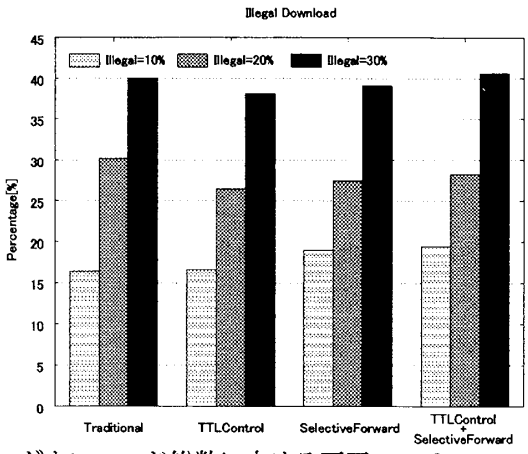

図 8: ダウンロード総数に占める不正コンテンツの割合

5.2 knowledge メッセージ削減効果の確認

knowledge メッセージのトラフィックは TTL 制御のみを使う場合は 60%程度，選択転送のみを使う場合は 99%程度，両者を組み合わせる場合は 99.6%程度減少した．このため，TTL 制御・選択転送によるトラフィック削減の効果はそれぞれ独立性が高いと考えられる．

5.3 knowledge メッセージの効果維持の確認

コンテンツダウンロード数に占める不正コンテンツの割合については提案方式の有無による大幅な変化は認められなかった．このため，knowledge メッセージの効果を維持できていると言える．

6. まとめ

本研究では，P2P ネットワーク上での信頼性構築プロトコルである STEP において，knowledge メッセージの効果を維持しつつそのトラフィックを削減することを目的として，信頼度の低いピアへの knowledge メッセージ転送を抑制する手法を提案した．提案手法は自分の信頼度と転送元・転送先の枝の信頼度に基づいて TTL を制御する方法，枝の信頼度に比例した確率で 1 つだけ転送先を選択する方法，およびそれらを組み合わせた方法である．

シミュレーションにより提案手法の評価を行った結果，TTL 制御・選択転送の効果は独立性が高く，両者を組み合わせることで knowledge メッセージのトラフィックを 99.6%程度削減できた．また，トラフィック削減により不正コンテンツのダウンロードが増加するという副作用は確認されなかった．

以上より，提案手法は STEP の knowledge メッセージ転送における効率化を実現する手法であると言える．

今後の課題は，他のトラフィック削減手法との比較および組み合わせによる効果の確認が挙げられる．本研究のシミュレーションではネットワークトポロジはランダムとしたが，実際の Pure P2P のネットワークは少数のピアが多くの隣接ピアを持ち，残りの多数のピアはほとんど隣接ピアを持たないという，スケールフリーネットワークになっていると言われる．

ネットワークのスケールフリー性を利用した効率化の手法は [6] で行われており，それを STEP にも適用できると考えられる．例えば，ネットワークの“ハブ”となっているピアには多くのピアからの knowledge メッセージが集まってくると考えられる．そしてそのハブからネットワーク全体に knowledge メッセージが転送されていくので，ハブでないピアが knowledge メッセージを発行する際はハブの方にだけ転送すれば良いことになる．これによって更なるトラフィック削減が実現できると考えられる．

また，knowledge メッセージが抑制されることによるコンテンツの検索・取得のパフォーマンスの向上の程度も確認する必要があるほか，knowledge メッセージ発行間隔とトークン有効期間の設定についても議論の余地があると言える．今回はピアの総数が 150 と小規模なシミュレーションだったため，より大規模なシミュレーションによる効果の確認も必要であると考えられる．

参考文献

[1] I. Martinovic, C. Leng, F. A. Zdarsky, A. Mauthe, R. Steinmetz, and J. B. Schmitt. Self-protection in P2P Networks:Choosing the Right Neighbourhood. In Proceedings of the International Workshop on Self-Organizing Systems (IWSOS 2006), Passau, Germany. Springer LNCS, pp.23-33, September 2006.

[2] S. D. Kamvar, M. T. Schlosser, and H. Garcia-Molina. The EigenTrust Algorithm for Reputation Management in P2P Networks. In Proceedings of the 12th International Conference on World Wide Web (WWW2003) , Budapest, Hungary. pp.640-651, May 2003.

[3] E. Damiani, S. De Capitani di Vimercati, S. Paraboschi, and P. Samarati. Managing and Sharing Servents' Reputation in P2P Systems. IEEE Transactions on Data and Knowledge Engineering, Vol.15, Issues.4, pp.840-854, July 2003.

[4] E. C. Efstathiou, P. A. Frangoudis, and G. C. Polyzos. Stimulating Participation in Wireless Community Networks. In Proceedings of 25th IEEE International Conference on Computer Communications. Barcelona, Spain, pp.1-13, April 2006.

[5] 佐藤，吉田，ピアツーピアシステムにおける不正ノードの動的回避方式，情報処理学会/電子情報通信学会 情報科学技術レターズ，Vol.4，pp.307-309，September 2005.

[6] 畑中，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，埼玉大学卒業論文，February 2006.

[7] 吉田，ピアツーピアネットワークにおけるノード信頼性の自律分散的な評価，埼玉大学卒業論文，February 2007.